



**UNIVERSIDAD NACIONAL  
AUTÓNOMA DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

**SISTEMA DE REINSCRIPCIÓN PARA LA CARRERA DE  
ARQUITECTURA VÍA WEB**

**T E S I S**

**QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN**

**P R E S E N T A N**

**JOSÉ RICARDO MANRÍQUEZ BETANZOS  
MIRIAM VALDÉS RODRÍGUEZ**

**DIRECTOR DE TESIS:  
M.C. ALEJANDRO VELÁZQUEZ MENA**



CIUDAD UNIVERSITARIA 2009

## **Agradecimientos:**

A Dios por la vida.

A México por las oportunidades.

A la Universidad Nacional Autónoma de México, a la Facultad de Ingeniería, Facultad de Arquitectura, Facultad de Medicina Veterinaria y Zootecnia, y a PROTECO, por permitirme superarme y crecer.

A mis padres María de Lourdes Betanzos Mendoza y Antonio Manríquez Marín por su amor, paciencia, su ejemplo y todo lo que han hecho por mí.

A mi hermano Juan Carlos Manríquez Betanzos por ser mi primer maestro y ejemplo a seguir.

A mi director de tesis M. C. Alejandro Velázquez Mena por su tiempo y asesoría.

Al Ing. Rodrigo Rosas por su apoyo, a nuestros sinodales: Arq. María Todd Alvarez, Ing. Laura Sandoval Montaña, M.I. Aurelio Adolfo Millán Nájera y al Ing. Jorge Ontiveros Junco.

A mi compañera de tesis, Miriam, por su constancia y compañerismo durante todo el desarrollo de esta tesis.

A mi amigo el Ing. Edgar García Cano Castillo por su amistad, asesoría y ayuda.

**Dedicatorias:**

A la familia Betanzos y a la familia Manríquez, especialmente a mis padres María de Lourdes Betanzos y Antonio Manríquez; y a mi hermano Carlos Manríquez.

A Enrique Gutiérrez, Luis Dzib, Víctor Barzalobre, Jonathan Hidalgo, Nancy Paz, Victoria Jiménez, Edith Ayala, Alejandra Moreno, Iván Figueroa, Edgar García Cano, Susana Betanzos, Angie Aguilar, Xóchitl Delgado, Janet López, Elizabeth Concha, Andrea Leyva, Elena León, Gloria López Ruiz, a amistades, profesores, compañeros de trabajo y a todos aquellos que, de una manera u otra, han contribuido para forjar el ser humano que soy.

A todos los que, en algún momento, han formado, forman o formarán parte de mi vida.

**Pensamientos:**

*“Los seres humanos no nacen para siempre el día en que sus madres los alumbran, sino que la vida los obliga a parirse a sí mismos una y otra vez”.*

Gabriel García Márquez. Escritor colombiano.

*“Divide las dificultades que examinas en tantas partes como sea posible para su mejor solución”.*

René Descartes. Filósofo y matemático francés.

*“Vivir no es sólo existir,  
sino existir y crear,  
saber gozar y sufrir  
y no dormir sin soñar.  
Descansar, es empezar a morir”.*

Gregorio Marañón. Médico y escritor español.

## **Dedicatoria**

A todos y cada una de las personas e Instituciones coparticipes de mi formación académica y personal.

## **Agradecimientos**

### **A Dios**

Como testimonio de infinito agradecimiento por darme la oportunidad de haber logrado uno de mis más profundas metas.

### **A la Universidad Nacional Autónoma de México y a la Facultad de Ingeniería**

Mi más profundo agradecimiento a mi Alma Mater, Universidad Nacional Autónoma de México y a la Facultad de Ingeniería, a los distintos miembros del cuerpo docente de esta Institución que constituyeron parte de mi formación académica, así como a los compañeros de carrera coadjutores de mi trayectoria escolar.

### **A mi familia**

Mi más eterno agradecimiento a mis padres Matilde y Daniel, ejemplos de vida y fortaleza, cuyos inmortales conocimientos y esencias han forjado mi ser. Gracias por su apoyo y comprensión incondicional en cada etapa de mi vida.

A mis hermanos: Armando, Tlanex y Citlali Guadalupe elementos fundamentales en mi vida, gracias por su apoyo y compañía incondicional en cada etapa emprendida.

A mi familia materna: Guadalupe Rivero, Valentín Rodríguez, Ángela Rodríguez y Tlacaoel Valentín López, así como a mi familia paterna: Manuela Martínez, Secundino Valdés, Martha Valdés, Carmen Valdés, Socorro Valdés y Antonio Valdés; partícipes en el proceso inherente de crecimiento humano e intelectual.

## **A mis Amigos**

A mis amigos Ma. Fernanda, Sergio, Angie, Verónica, Mauricio, Gerardo, Gil y Omar coadyuvantes de mí formación académica y personal, cuyas experiencias de vida han quedado plasmadas en mi ser.

Al M.C. Alejandro Velázquez Mena, al Ing. Rodrigo Rosas, al Ing. Edgar Eduardo García Cano, a la Facultad de Arquitectura, a Ricardo Manríquez compañero de Tesis, a los sinodales: Arq. María Tood Álvarez, Ing. Laura Sandoval Montaña, M.I Aurelio Adolfo Millán Nájera y al Ing. Jorge Ontiveros Junco por su asesoría, apoyo, aportaciones, y comentarios en el desarrollo de este trabajo.

*“Somos lo que hacemos día a día, de modo que la excelencia no es un acto, sino un hábito”*

Aristóteles. Filósofo griego

*“Por mi raza hablará el espíritu”*  
*Ciudad Universitaria, D.F. Agosto 2009*  
*Miriam Valdés Rodríguez*

# Contenido

Introducción.....	1
I.1 Secretaría de Asuntos Escolares.....	3
I.1.1 Objetivo de la Secretaría de Asuntos Escolares.....	3
Capítulo 1 Conceptos básicos.....	6
1.1 Ingeniería de software .....	6
1.1.1 Modelo de ciclo de vida en cascada .....	7
1.1.2 Modelo de ciclo de vida en espiral .....	8
1.2 Bases de datos.....	10
1.2.1 Propósito de los sistemas de bases de datos. ....	10
1.2.2 Abstracción de datos.....	10
1.2.3 Modelo de Datos.....	11
1.2.4 Lenguajes de Bases de Datos .....	12
1.2.5 Bases de Datos Relacionales .....	13
1.3 SQL.....	14
1.3.1 Componentes de SQL.....	14
1.3.2 Integridad.....	15
1.3.3 Integridad de datos.....	15
1.4 Arquitectura de aplicaciones multinivel .....	16
1.4.1 Arquitectura de dos capas.....	16
1.4.2 Arquitectura de tres capas.....	18
1.5 Lenguajes De Programación.....	19
1.5.1 Paradigma de programación .....	20
1.5.2 Clasificación por paradigmas de programación .....	20
1.6 Herramientas de programación.....	21
1.6.1 UML .....	22
1.7 Arquitectura MVC (Modelo/Vista/Controlador).....	22
Capítulo 2. Análisis .....	24
2.1 Proceso de Reinscripción.....	24
2.2 Inscripción a Taller de Arquitectura.....	24
2.3 Consideraciones generales para inscripción a Taller.....	24
2.3.1 Casos posibles al realizar la inscripción al Taller de Arquitectura.....	25
2.4 Registro de asignaturas.....	27
2.4.1 Consideraciones generales para el Registro de asignaturas: .....	27
2.5 Asignación de grupos .....	29
2.5.1 Consideraciones generales para asignación de grupos .....	29
2.5.2 Jerarquía de la asignación de grupos .....	31
2.6 Etapa de altas, bajas y cambios .....	32
2.6.1 Consideraciones generales para la etapa de altas, bajas y cambios.....	32

2.7 Alcance del sistema .....	34
2.8 Escenarios .....	35
2.8.1 Actor Alumno .....	35
2.8.2 Para el usuario Mantenimiento .....	37
2.8.3 Para el actor Coordinador de Taller.....	40
2.8.4 Para el actor Coordinador de Carrera .....	41
2.8.5 Para el actor Jefe de Asuntos Escolares.....	42
Capítulo 3 Diseño del sistema .....	44
3.1 Alumno .....	44
3.2 Mantenimiento.....	45
3.3 Coordinador de Taller.....	46
3.4 Coordinador de Carrera .....	47
3.5 Jefe de Asuntos Escolares .....	47
3.6 Diagrama Entidad Relación.....	48
3.6.1 Diagrama Entidad – Relación.....	49
3.7 Diccionario de datos .....	50
3.8 Diagramas de Modelado de Comportamiento UML .....	55
3.9 Diagramas de casos de uso .....	55
3.9.2 Mantenimiento.....	56
3.9.3 Coordinador de Taller.....	57
3.9.4 Coordinador de Carrera .....	57
3.9.5 Jefe de Asuntos Escolares.....	58
3.10 Diagramas de Actividades .....	59
3.10.1 Ingreso a Trámites Escolares .....	59
3.10.2 Alumno .....	60
3.10.3 Mantenimiento.....	61
3.10.4 Coordinador de Taller.....	62
3.10.5 Coordinador de Carrera .....	63
3.10.6 Jefe de Asuntos Escolares.....	63
3.11 Diagramas de Secuencia.....	65
3.11.1 Alumno .....	65
3.11.2 Mantenimiento.....	66
3.11.3 Coordinador de Taller.....	72
3.11.4 Coordinador de Carrera .....	74
3.11.5 Jefe de Asuntos Escolares.....	75
3.12 Vistas .....	77
3.12.1 Secuencia de pantallas .....	77
3.12.1.2 Ingreso General Trámites Académicos.....	82
3.12.1.3 Mantenimiento.....	83
3.12.1.4 Coordinador de Taller.....	94
3.12.1.5 Coordinador de Carrera .....	99

3.12.1.6 Jefe de Asuntos Escolares.....	103
Capítulo 4 Programación .....	106
4.1 PHP.....	106
4.1.1 Características de PHP .....	106
4.2. SQL.....	107
4.2.1 Estándares ANSI / ISO .....	107
4.3. PostgreSQL.....	108
4.3.1 Características de PostgreSQL .....	109
4.4 Javascript .....	109
4.4.1 Características de Javascript.....	109
4.5. HTML.....	110
4.6. CSS .....	111
Capítulo 5 Implementación .....	112
5.1 Instalación de Apache.....	112
5.2 Instalación de PHP .....	113
5.3 Instalación de PostgreSQL .....	114
5.4 Base de datos ARQESCOLAR .....	116
6. Pruebas y Mantenimiento .....	119
6.1. Mantenimiento al sistema.....	119
6.1.1. Alumno .....	119
6.1.2. Actor Mantenimiento.....	122
6.1.3. Coordinador de Taller.....	125
6.1.5. Jefe de Asuntos Escolares.....	127
6.2. Código .....	129
6.2.1 Estándares utilizados en la nomenclatura de las funciones .....	129
6.2.2 Estándares utilizados en la nomenclatura para scripts de PHP .....	130
6.3 Casos de prueba .....	132
6.3.1 Alumno .....	132
6.3.2 Coordinador de Taller.....	135
6.3.3 Mantenimiento.....	138
6.3.4 Coordinador de Carrera .....	147
6.3.5 Jefe de asuntos Escolares.....	147
Conclusiones.....	151
Glosario .....	153
Referencia.....	306

# Figuras

Figura I. 1 Mapa curricular arquitectura.....	2
Figura I. 2 Seriación de las etapas .....	3
Figura 1.1 Modelo en cascada .....	7
Figura 1.2 Modelo en espiral.....	8
Figura 1.3 Los tres niveles de abstracción de datos .....	11
Figura 1.4 Arquitectura de dos capas .....	16
Figura 1.5 Arquitectura de tres capas .....	18
Figura 1.6 Arquitectura Modelo Vista Controlador .....	22
Figura 2.1 Inscripción al Taller .....	25
Figura 2.2 Registro de Asignaturas .....	28
Figura 2.3 Registro de Asignaturas .....	28
Figura 2.4 El Alumno ingresará al registro tantas veces lo requiera dentro del periodo de tiempo estipulado.....	29
Figura 2.5 Asignación de grupos: Alumnos del taller por promedio .....	31
Figura 2.6 Asignación de grupo: alumnos regulares por promedio .....	31
Figura 2.7 Asignación de grupo: alumnos irregulares por promedio.....	32
Figura 2.8 Etapa de Altas, Bajas y Cambios .....	33
Figura 2.9 El alumno ingresará tantas veces lo requiera dentro del periodo de tiempo estipulado a partir de su horario para cambios.....	33
Figura 3.1 Carta de estructura del usuario Alumno.....	44
Figura 3.2 Carta de estructura del usuario Mantenimiento .....	45
Figura 3.3 Carta de estructura del usuario Coordinador de Taller .....	46
Figura 3.4 Carta de estructura del usuario Coordinador de Carrera.....	47
Figura 3.5 Carta de estructura del usuario Jefe de Asuntos Escolares .....	47
Figura 3.6 Diagrama Entidad -Relación.....	49
Figura 3.7 Diagrama de caso de uso, usuario Alumno.....	55
Figura 3.8 Diagrama de caso de uso, usuario Mantenimiento.....	56
Figura 3.9 Diagrama de caso de uso, usuario Coordinador de Taller .....	57

Figura 3.10 Diagrama de caso de uso, usuario Coordinador de Carrera .....	57
Figura 3.11 Diagrama de caso de uso, Jefe de Asuntos Escolares .....	58
Figura 3.12 Diagrama de actividades Ingreso a Trámites Escolares .....	59
Figura 3.13 Diagrama de actividades Alumno Actualiza datos .....	60
Figura 3.14 Diagrama de actividades Cancelar inscripción al Taller.....	62
Figura 3.15 Diagrama de actividades Ingreso Consultar Historial Académico .....	64
Figura 3.16 Diagrama de secuencia Actualizar Datos.....	65
Figura 3.17 Diagrama de secuencia Asignación de grupos.....	66
Figura 3.18 Diagrama de secuencia Cambiar calificación de un alumno .....	67
Figura 3.19 Diagrama de secuencia Cancelar grupo de cursos .....	68
Figura 3.20 Diagrama de secuencia Modificar cupo de cursos .....	69
Figura 3.21 Diagrama de secuencia Actualizar Curso de Cómputo.....	70
Figura 3.22 Diagrama de secuencia Actualización de etapas mínimas .....	71
Figura 3.23 Diagrama de secuencia Inscribir Alumno a Taller.....	72
Figura 3.24 Diagrama de secuencia Cancelar Registro a Taller.....	73
Figura 3.25 Diagrama de secuencia Actualizar Cupo de un Taller .....	74
Figura 3.26 Diagrama de secuencia Inscribir a Alumnos.....	75
Figura 3.27 Diagrama de secuencia Consultar Historial Académico .....	76
Figura 3.28 Pantalla de Ingreso al Sistema.....	77
Figura 3.29 Pantalla para ingresar login y password para el usuario Alumno .....	78
Figura 3.30 Pantalla que muestra las asignaturas registradas por el Alumno .....	79
Figura 3.31 Pantalla que muestra las asignaturas asignadas por el Sistema así como los errores durante el proceso de Asignación.....	80
Figura 3.32 Pantalla para ingresar la información personal del Alumno .....	81
Figura 3.33 Pantalla para ingresar login y password para los usuarios de Asuntos Escolares .....	82
Figura 3.34 Menú de opciones para el usuario Mantenimiento .....	83
Figura 3.35 Pantalla para ingresar un nuevo cupo de un grupo .....	84
Figura 3.36 Pantalla que muestra que el cupo de un grupo ha sido actualizado con éxito ..	85
Figura 3.37 Pantalla que muestra los grupos de una asignatura seleccionada .....	86
Figura 3.38 Pantalla que muestra que el grupo seleccionado ha sido cancelado .....	87
Figura 3.39 Pantalla para ingresar una calificación de un Alumno.....	88
Figura 3.40 Pantalla que muestra la actualización de una calificación de un Alumno .....	89
Figura 3.41 Pantalla correspondiente al proceso de actualización de Etapas Mínimas .....	90

Figura 3.42	Página correspondiente al proceso de Asignación de grupos .....	91
Figura 3.43	Pantalla para ingresar el número de cuenta de un Alumno .....	92
Figura 3.44	Pantalla que despliega que el requisito de curso de cómputo ha sido actualizado .....	93
Figura 3.45	Menú de opciones del usuario Coordinador de Taller .....	94
Figura 3.46	Pantalla para ingresar el número de cuenta de un Alumno .....	95
Figura 3.47	Pantalla que despliega las opciones para inscripción a un Taller de un Alumno .....	96
Figura 3.48	Pantalla que despliega el resultado de una inscripción a Taller .....	97
Figura 3.49	Pantalla que muestra que un Alumno ha sido dado de baja de un Taller .....	98
Figura 3.50	Menú de opciones del usuario Coordinador de Carrera.....	99
Figura 3.51	Pantalla para ingresar nuevo cupo de un grupo de Taller .....	101
Figura 3.52	Pantalla que muestra que el cupo de un grupo de Taller ha sido actualizado. ....	102
Figura 3.53	Menú de opciones para el usuario Jefe de Asuntos Escolares .....	103
Figura 3.54	Página que muestra que un alumno se ha inscrito en una asignatura .....	104
Figura 3.55	Página que muestra el Historial Académico de un Alumno .....	105
Figura 6. 1	Módulos del usuario Alumno.....	119
Figura 6. 2	Pantalla para ingresar login y password para el usuario Alumno .....	119
Figura 6. 3	Pantalla que despliega las asignaturas registradas por el Alumno .....	120
Figura 6. 4	Pantalla que muestra los errores ocurridos en el proceso de Asignación .....	121
Figura 6. 5	Módulos del usuario Mantenimiento .....	122
Figura 6. 6	Pantalla que permite cancelar el grupo de la asignatura seleccionada.....	123
Figura 6. 7	Pantalla para ingresar una nueva calificación para un Alumno .....	124
Figura 6. 8	Módulo del usuario Coordinador de Taller .....	125
Figura 6. 9	Pantalla que muestra que un Alumno ha sido inscrito a un Taller.....	126
Figura 6. 10	Módulo del usuario Coordinador de Carrera .....	127
Figura 6. 11	Módulo del usuario Jefe de Asuntos Escolares.....	127
Figura 6. 12	Pantalla que muestra que un Alumno ha sido inscrito en una asignatura .....	128
Figura 6. 13	Pantalla para ingresar información personal del Alumno .....	135
Figura 6. 14	Pantalla que muestra que el Alumno ha sido inscrito en un grupo de Taller. ....	136
Figura 6. 15	Pantalla que muestra que el Alumno ha sido dado de baja de un grupo de Taller.....	137

Figura 6. 16 Pantalla que muestra que el cupo de un grupo ha sido actualizado .....	139
Figura 6. 17 Pantalla que muestra que un grupo ha sido cancelado.....	141
Figura 6. 18 Pantalla que permite seleccionar el intervalo de tiempo en el que se realizará la Asignación de Horarios para Altas, Bajas y Cambios.....	144
Figura 6. 19 Pantalla que muestra el resultado del proceso de Asignación de grupos.....	145
Figura 6. 20 Página que muestra que el requisito de Curso de Cómputo ha sido actualizado .....	146
Figura 6. 21 Pantalla que muestra que un Alumno ha sido inscrito en una asignatura .....	149
Figura 6. 22 Pantalla que despliega el Historial Académico de un Alumno .....	150

# Tablas

Tabla 1.1 .....	19
Tabla 3.1 Diccionario de datos correspondiente a la tabla horarios_grupos .....	50
Tabla 3.2 Diccionario de datos correspondiente a la tabla profesores_grupos .....	50
Tabla 3.3 Diccionario de datos correspondiente a la tabla profesores .....	50
Tabla 3.4 Diccionario de datos correspondiente a la tabla aulas.....	50
Tabla 3.5 Diccionario de datos correspondiente a la tabla actores_asuntos_escolares .....	50
Tabla 3.6 Diccionario de datos correspondiente a la tabla tipos_actores_asuntos_escolares .....	51
Tabla 3.7 Diccionario de datos correspondiente a la tabla carreras .....	51
Tabla 3.8 Diccionario de datos correspondiente a la tabla talleres .....	51
Tabla 3.9 Diccionario de datos correspondiente a la tabla grupos .....	51
Tabla 3.10 Diccionario de datos correspondiente a la tabla etapas .....	52
Tabla 3.11 Diccionario de datos correspondiente a la tabla asignaturas_seriaciones .....	52
Tabla 3.12 Diccionario de datos correspondiente a la tabla asignaturas .....	52
Tabla 3.13 Diccionario de datos correspondiente a la tabla historias_academicas_completas .....	52
Tabla 3.14 Diccionario de datos correspondiente a la tabla alumnos_carreras.....	52
Tabla 3.15 Diccionario de datos correspondiente a la tabla alumnos_errores .....	53
Tabla 3.16 Diccionario de datos correspondiente a la tabla registros .....	53
Tabla 3.17 Diccionario de datos correspondiente a la tabla inscripciones .....	53
Tabla 3.18 Diccionario de datos correspondiente a la tabla horarios_abc .....	53
Tabla 3.19 Diccionario de datos correspondiente a la tabla situaciones_academicas.....	53
Tabla 3.20 Diccionario de datos correspondiente a la tabla alumnos .....	54
Tabla 3.21 Diccionario de datos correspondiente a la tabla errores.....	54
Tabla 3.22 Diccionario de datos correspondiente a la tabla tipos_alumnos.....	54
Tabla 3.23 Diccionario de datos correspondiente a la tabla fechas.....	54
Tabla 3.24 .....	77
Tabla 3.25 .....	78
Tabla 3.26.....	79
Tabla 3.27A, B.....	80

Tabla 3.28A,B .....	81
Tabla 3.29 .....	82
Tabla 3.30A, B .....	83
Tabla 3.31A, B .....	84
Tabla 3.32 .....	85
Tabla 3.33 A,B .....	86
Tabla 3.34 .....	87
Tabla 3.35 A,B .....	88
Tabla 3.36 .....	89
Tabla 3.37 .....	90
Tabla 3.38 .....	91
Tabla 3.39 .....	92
Tabla 3.40 .....	93
Tabla 3.41 .....	94
Tabla 3.42 .....	95
Tabla 3.43 .....	96
Tabla 3.44 .....	97
Tabla 3.45 A, B .....	98
Tabla 3.46 .....	100
Tabla 3.47 .....	101
Tabla 3.48 .....	102
Tabla 3.49 .....	103
Tabla 3.50 .....	104
Tabla 3.51 .....	105
Tabla 6.1 .....	120
Tabla 6.2 .....	120
Tabla 6.3 .....	121
Tabla 6.4 .....	121
Tabla 6.5 .....	122
Tabla 6.6 .....	122
Tabla 6.7 .....	123
Tabla 6.8 .....	124

Tabla 6.9 .....	124
Tabla 6.10 .....	125
Tabla 6.11 .....	125
Tabla 6.12 .....	126
Tabla 6.13 .....	126
Tabla 6.14 .....	127
Tabla 6.15 .....	128
Tabla 6.16 .....	128
Tabla 6.17 .....	130

# Introducción

La Facultad de Arquitectura de la UNAM tiene una tradición de más de dos siglos, heredada de la Antigua Academia de San Carlos.

Cuenta con cuatro licenciaturas: Licenciatura en Arquitectura, Licenciatura en Diseño Industrial, Licenciatura en Urbanismo y Licenciatura en Arquitectura de paisaje.

La Licenciatura en Arquitectura es coordinada por el Colegio Académico de Arquitectura, el cual está conformado por los Coordinadores de los 16 Talleres de Arquitectura, los Seminarios de Área de Proyectos, Tecnología, Teoría, Historia e Investigación, Urbano-Ambiental, Extensión Universitaria y la propia Coordinación del Colegio Académico.

El Plan de estudios 1999 representado gráficamente en la figura I.1, con una duración de diez semestres, estructura la carrera en cinco etapas de formación: Básica, Desarrollo, Profundización, Consolidación y Demostración, ver figura I.2; el eje curricular es el Taller de Arquitectura.

El Taller de Arquitectura es la forma de organización pedagógica que contiene los elementos que relacionan al Área de proyecto con sus componentes teóricos, tecnológicos y constructivos, urbano ambientales y de vinculación social. En torno a él se estructuran todas las actividades académicas que constituyen la base formativa del estudiante para plantear propuestas coherentes con el ámbito en que se ubicarán; permitiendo a sus miembros identificarse con un interés común.

Se considera que existen conocimientos básicos e indispensables de las disciplinas que integran la arquitectura; las cuales se imparten en cursos obligatorios y se organizan en cinco áreas de conocimiento: Proyecto, Teoría; Historia e investigación; Tecnología; Urbano Ambiental y Extensión Universitaria. Los cursos obligatorios donde se proporcionan los conocimientos significativos de las diferentes áreas de conocimiento se imparten durante las tres primeras etapas de formación.

Además existen cursos selectivos de donde los alumnos eligen 12 cursos en función de su vocación, intereses, habilidades de tal modo que el alumno se hace copartícipe del diseño curricular de su formación.

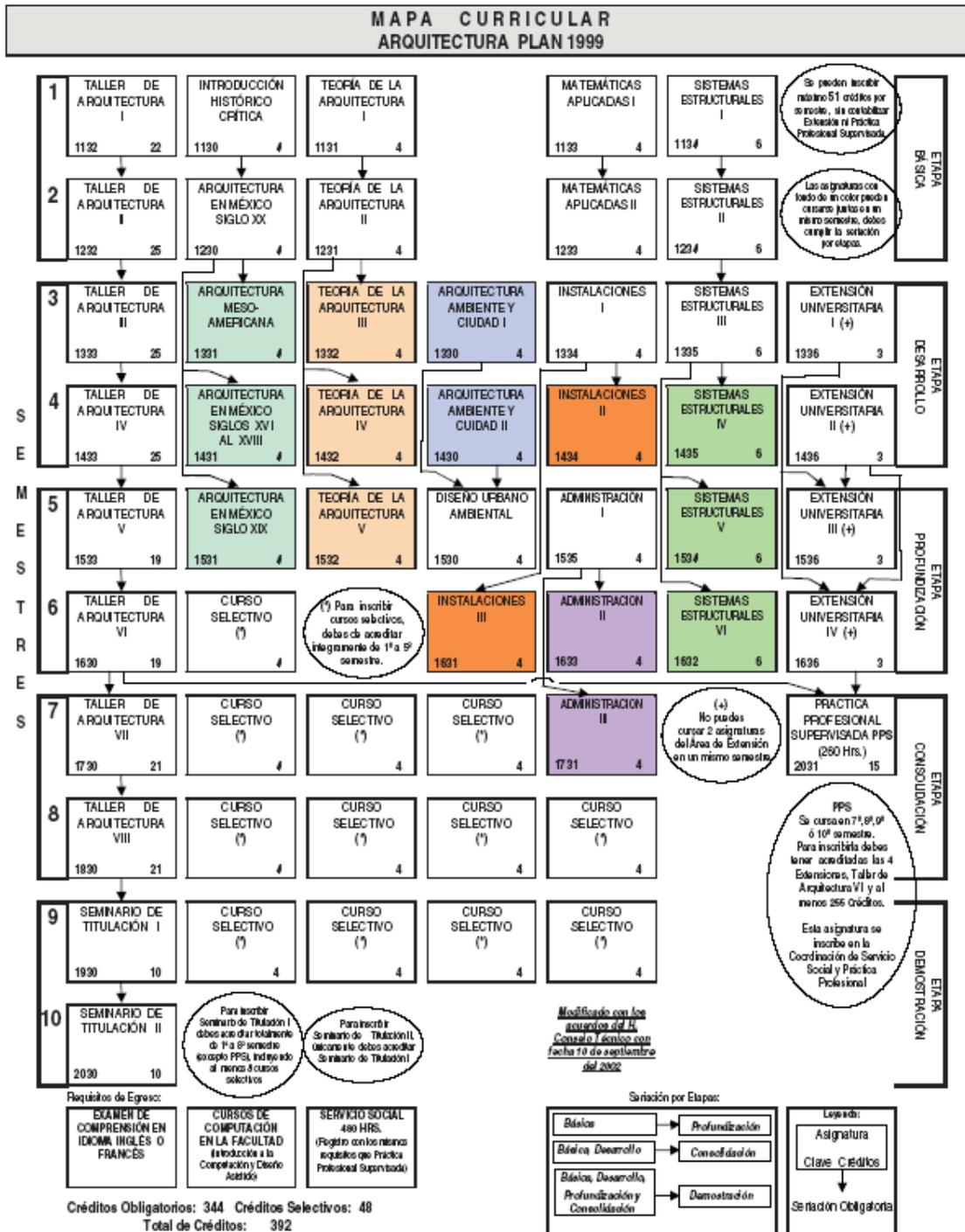


Figura I. 1 Mapa curricular arquitectura

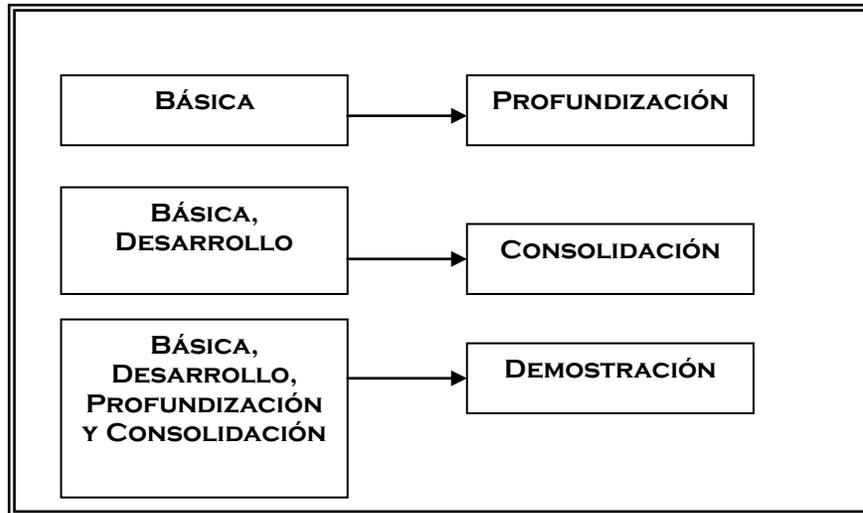


Figura I. 2 Seriación de las etapas

## **I.1 Secretaría de Asuntos Escolares**

La estructura Académica Administrativa de la Facultad de Arquitectura está integrada por el H. Consejo Técnico, Dirección, las Secretarías General, Académica y de Asuntos Escolares, la Oficina de Servicios Escolares, las Coordinaciones de Apoyo a la Dirección, de Apoyo a Cuerpos Colegiados, de Planeación y de Apoyo a la Docencia, por citar algunos.

### **I.1.1 Objetivo de la Secretaría de Asuntos Escolares**

La Secretaría de Asuntos Escolares tiene como objetivo programar, coordinar y supervisar las actividades relacionadas con la administración escolar, llevar el control administrativo de los alumnos desde su ingreso a la licenciatura y hasta que concluyan con el 100% de los créditos que marca el plan de estudios correspondiente.

Parte de las actividades administrativas de la Secretaría de Asuntos Escolares realizadas cada semestre dentro de la Facultad de Arquitectura comprende el proceso de reinscripción, cuyo objetivo es el de inscribir en las diferentes asignaturas impartidas en esta institución educativa, a los estudiantes de las cuatro licenciaturas que integran esta Facultad.

El sistema que actualmente realiza las reinscripciones está desarrollado en PHP, donde la información es consultada y administrada bajo Microsoft Access, apoyada con un sistema de archivos.

Debido a las necesidades de delegar tareas entre los usuarios de Asuntos Escolares de la Facultad de Arquitectura, de proporcionar información que es inaccesible para los usuarios, centralizar la información correspondiente a la población estudiantil de Arquitectura, así como de robustecer el proceso de reinscripciones vía Internet se desarrolló un nuevo sistema de reinscripciones que contempla únicamente la licenciatura en Arquitectura.

En el capítulo 1 se presenta una visión general de la Ingeniería de Software, el ciclo de vida en el que se basó el desarrollo del sistema, así como la descripción de bases de datos, arquitectura de sistemas y lenguajes de programación.

En el capítulo 2 se describe el proceso de reinscripciones, detallando los estatutos de reinscripción de la Secretaría de Asuntos Escolares del mismo. Se muestran los módulos que integran la estructura del sistema y se presentan los escenarios correspondientes a los actores finales del sistema.

En el capítulo 3 se describe el comportamiento del sistema mediante diagramas de secuencia, diagramas de casos de uso, diagramas de actividades, mostrando la interacción entre sus componentes, diagrama entidad/relación y el diccionario de datos. Se detalla la estructura de cada módulo que compone al Sistema a través de cartas de estructura. Se presenta la secuencia de pantallas que forma parte de las vistas (front-end).

En el capítulo 4 se describen las características y el uso de estándares de programación utilizados en el desarrollo del sistema.

En el capítulo 5 se describe el proceso de instalación del software que requiere el Sistema para ser puesto en marcha: PostgreSQL como DBMS, PHP como lenguaje de programación y Apache como servidor.

En el capítulo 6 se muestran los diferentes casos de prueba realizados en cada módulo del sistema para verificar su funcionalidad. Finalmente se describe el mantenimiento preventivo del sistema.

# Capítulo 1 Conceptos básicos

## 1.1 Ingeniería de software

Ingeniería de software es la aplicación de un enfoque sistemático, disciplinado y cuantificable hacia el desarrollo, operación y mantenimiento del software. Un modelo de ciclo de vida es una descripción de los pasos que deben realizarse cuando se construye un producto de software, el cual se divide en fases:

- Fase de requisitos. Se explora y afina el concepto, y se determinan los requisitos del cliente.
- Fase de análisis. Se analizan y presentan los requisitos del cliente en forma del documento de especificaciones. Al término de esta fase se redacta el plan de gestión del proyecto del software, en el que se describe una propuesta del desarrollo del software con amplio detalle.
- Fase de diseño. Las especificaciones atraviesan dos procesos de diseño consecutivos durante esta fase. Primero surge el diseño arquitectónico, en el cual el producto como un todo se divide en sus componentes, denominados módulos. Después se diseña cada módulo; este proceso se denomina diseño detallado.
- Fase de implementación. Cada uno de los diferentes componentes atraviesa por procesos de codificación y pruebas. Después los componentes del producto se combinan y prueban como un todo; esto se denomina integración. Posteriormente el cliente realiza pruebas (pruebas de aceptación). Esta fase termina cuando el cliente acepta el producto y se instala en su computadora.
- Mantenimiento. Consiste en todos los cambios al producto una vez que se ha entregado e instalado en el equipo del cliente y pasa su prueba de aceptación.

### 1.1.1 Modelo de ciclo de vida en cascada

Este modelo de ciclo de vida se retroalimenta dependiendo de la necesidad de regresar o avanzar a la siguiente etapa, en este modelo se indica que en cualquier etapa del desarrollo del sistema se puede regresar a la etapa requerida dependiendo de cómo se desarrolle el proyecto. En la figura 1.1 se observan las etapas que componen el modelo de ciclo de vida en cascada.

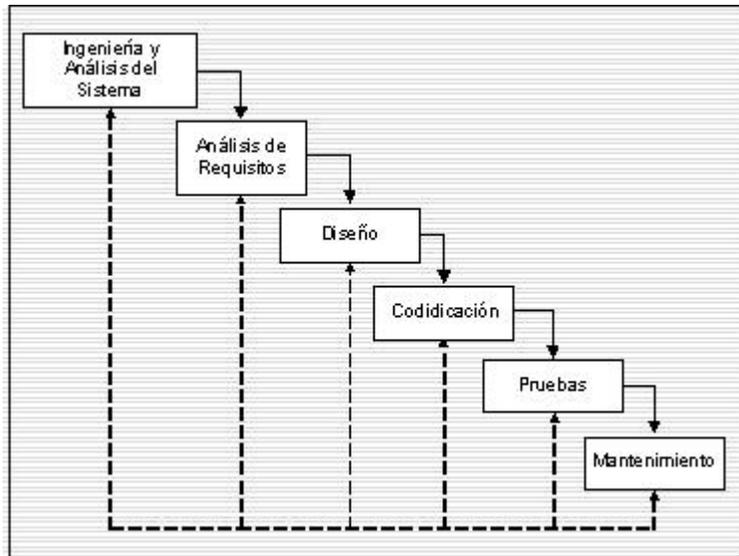


Figura 1.1 Modelo en cascada

**Ingeniería de Sistemas.** En esta etapa se abarcan los requerimientos globales del sistema, además de establecer los requerimientos del mismo, se realiza un análisis a grandes rasgos y se genera un diseño en el ámbito global.

**Análisis.** Es el proceso de recolectar los requerimientos de los usuarios, así como el entendimiento del dominio del problema y el entorno en que se encuentra éste. Las técnicas utilizadas para obtener los requerimientos son: entrevistas con el cliente, la observación, el análisis de tareas problemáticas y el desarrollo de reglas.

**Diseño.** Esta etapa se enfoca sobre tres atributos del sistema: la estructura de datos, la arquitectura del sistema y los detalles de los procedimientos. Es aquí donde se detalla el análisis de los requerimientos para posteriormente traducirlos a un lenguaje de programación.

**Codificación.** En esta etapa el diseño se traduce en líneas de código de algún lenguaje de programación de computadora.

**Pruebas.** Es el proceso en donde se realizan las pruebas al sistema, se realizan cada vez que se va generando el código de algún módulo, aquí se verifica la lógica del programa para asegurar que todas las sentencias del programa se ejecuten correctamente.

**Mantenimiento.** Se realiza posteriormente a la entrega del sistema, en ésta se realizan adecuaciones a nuevos requerimientos de los usuarios o modificaciones solicitadas posteriormente a la entrega.

### 1.1.2 Modelo de ciclo de vida en espiral

Es un modelo de proceso de software evolutivo que conjuga la naturaleza iterativa de construcción de prototipos con los aspectos controlados y sistemáticos del modelo lineal secuencial. En este modelo el software se desarrolla en una serie de versiones incrementales. Puede adaptarse y aplicarse a lo largo de la vida del software de computadora.

Utiliza la construcción de prototipos como mecanismo de reducción de riesgos y permite a quién lo desarrolla aplicar el enfoque de construcción de prototipos en cualquier etapa de evolución del producto.

Se divide en un número de actividades de marco de trabajo conocidas como regiones de tareas. En la figura 1.2 se muestran las distintas etapas que comprende el modelo en espiral.



Figura 1.2 Modelo en espiral

**Comunicación con el cliente.** La tarea requerida para establecer comunicación entre el desarrollador y el cliente: es el proceso de fijar los requerimientos de los usuarios, basados en las necesidades del cliente.

**Planificación.** Las tareas requeridas para definir recursos, el tiempo y otra información relacionadas con el proyecto: predecir la duración de las actividades y tareas de nivel individual, recursos requeridos, concurrencia y solapamiento de tareas para el desarrollo en paralelo y camino crítico a través de la red de actividades. Estimar recursos: Predicción de personal, esfuerzo y costo que se requerirán para terminar las actividades y productos conocidos asociados con el proyecto. Planificar tareas y solapamiento de actividades y tareas. Definir y desarrollar los requerimientos de software. Definir los requisitos de interfaz. Priorizar e integrar los requisitos de software.

**Análisis de riesgos.** Las tareas requeridas para evaluar riesgos técnicos y de gestión. Las actividades a realizar son: Identificar ideas o necesidades, formular soluciones potenciales, conducir estudios de viabilidad. Planificar la transición del sistema, refinar y finalizar la idea o necesidad. Analizar las funciones del sistema, desarrollar la arquitectura del sistema. Descomponer los requisitos del sistema y hacer la planificación de contingencias.

**Ingeniería.** Las tareas requeridas para construir una o más representaciones de la aplicación son: Realizar el diseño arquitectónico. Analizar el flujo de información. Diseñar la base de datos. Seleccionar o desarrollar algoritmos. Realizar el diseño detallado de la etapa. Crear el código fuente.

**Construcción y acción.** Las tareas requeridas para construir, probar, instalar y proporcionar soporte al usuario. Las actividades a realizar en esta etapa son: Crear los datos de prueba de código fuente. Ejecutar las tareas de verificación y validación. Recoger y analizar los datos de la métrica. Planificar las pruebas. Desarrollar las especificaciones de las pruebas. Ejecutar las pruebas. Generación de los aspectos de mejora, errores, efectos, ampliaciones.

**Evaluación del cliente.** Las tareas requeridas para obtener la reacción del cliente según la evaluación de la etapa de ingeniería e implementación durante la etapa de instalación. El cliente evalúa el trabajo de ingeniería y sugiere modificaciones. Sobre la base de los comentarios del cliente se produce la siguiente fase de planificación y de análisis de riesgo. En cada bucle alrededor de la espiral, la culminación del análisis de riesgo resulta en una decisión de "seguir o no seguir".

## **1.2 Bases de datos**

Un sistema gestor de bases de datos (SGBD) consiste en una colección de datos (llamados base de datos) interrelacionados y un conjunto de programas para acceder a ellos. El objetivo principal de un SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera práctica y eficiente.

### **1.2.1 Propósito de los sistemas de bases de datos.**

Los sistemas de bases de datos se diseñan para gestionar grandes cantidades de datos. La gestión de los datos implica la definición de estructuras para almacenar la información así como la provisión de mecanismos para la manipulación de la misma.

### **1.2.2 Abstracción de datos.**

Se usan estructuras de datos complejos para la representación de los ítems en la base de datos. Esta complejidad se oculta a los usuarios mediante varios niveles de abstracción, lo cual simplifica la interacción de los usuarios con el sistema, como lo muestra la figura 1.3

- Nivel físico. Describe cómo se almacenan realmente los datos. Describe en detalle las estructuras de datos complejas de bajo nivel. Los registros se pueden describir como bloques de posiciones consecutivas de almacenamiento.
- Nivel lógico. Describe qué datos se almacenan en la base de datos y qué relaciones existen entre esos datos. Describe toda la base de datos en términos de un número pequeño de estructuras relativamente simples. Los registros de este tipo se describen mediante una definición de tipo así como la relación entre estos tipos de registros.
- Nivel de vistas. El nivel más elevado de abstracción sólo describe parte de la base de datos. Aunque el nivel lógico usa estructuras más simples, queda algo de complejidad debido a la variedad de información almacenada en las grandes bases de datos. Este nivel existe para simplificar su interacción con el sistema, puede proporcionar muchas vistas para la misma base de datos. Los usuarios ven un conjunto de programas de aplicación que ocultan los detalles de los tipos de datos.

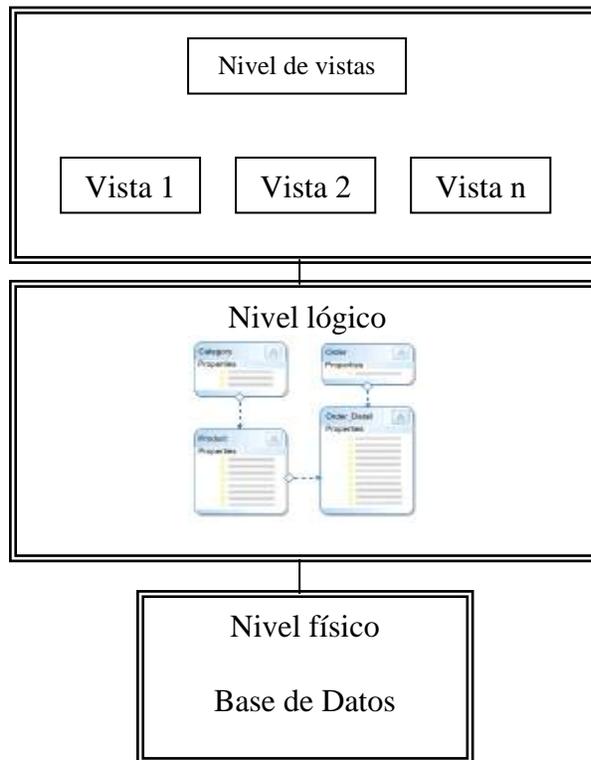


Figura 1.3 Los tres niveles de abstracción de datos

### 1.2.3 Modelo de Datos

El modelo de datos es una colección de herramientas conceptuales para describir los datos, sus relaciones, su semántica y las restricciones de inconsistencia. Ofrecen un modo de describir el diseño de las bases de datos en los niveles físico, lógico y de vistas.

Se pueden clasificar en cuatro categorías diferentes:

- **Modelo relacional:** Usa una colección de tablas para representar tanto los datos como sus relaciones. Cada tabla tiene varias columnas y cada columna tiene un nombre único. Es un ejemplo de un modelo basado en registros ya que la base de datos se estructura en registros de formato fijo de varios tipos.
- **Modelo entidad-relación (E-R):** Se basa en una percepción del mundo real que consiste en una colección de entidades y de las relaciones entre ellos. Las entidades se describen en las bases de datos mediante un conjunto de atributos. Una relación es una asociación entre varias entidades.

- Modelo de datos orientado a objetos: Se puede considerar como una extensión del modelo E-R con los conceptos de la encapsulación, los métodos y la identidad de los objetos.
- Modelo de datos semiestructurado: Permite la especificación de datos donde los elementos de datos individuales del mismo tipo pueden tener diferentes conjuntos de atributos.

#### **1.2.4 Lenguajes de Bases de Datos**

Los sistemas de bases de datos proporcionan un lenguaje de definición de datos para especificar el esquema de la base de datos y un lenguaje de manipulación de datos para expresar las consultas y las modificaciones de la misma.

- **Lenguaje de manipulación de datos.**

Un lenguaje de manipulación de datos (LMD) es aquel lenguaje que permite a los usuarios manipularlos o tener acceso a los datos organizados mediante el modelo de datos correspondientes. Los tipos de acceso son:

La recuperación de la información almacenada en la base de datos.

La inserción de información nueva en la base de datos.

El borrado de la información de la base de datos.

La modificación de la información almacenada.

Existen fundamentalmente dos tipos:

Los LMDs procedimentales necesitan que el usuario especifique qué datos se necesitan y cómo obtenerlos.

Los LMDs declarativos necesitan que el usuario especifique qué datos se necesitan sin que haga falta que especifique cómo obtenerlos.

Una consulta es una instrucción que solicita que se recupere información. La parte de los LMDs implicada en la recuperación de información se denomina lenguaje de consultas.

- **Lenguaje de definición de datos**

La estructura de almacenamiento y los métodos de acceso usados por el sistema de bases de datos se especifican mediante un conjunto de instrucciones en un tipo especial de LDD

denominado lenguaje de almacenamiento y definición de datos, cuyas instrucciones definen los detalles de implementación de los esquemas de las bases de datos.

### **1.2.5 Bases de Datos Relacionales**

Una base de datos relacional es un conjunto de tablas, bajo una misma identificación, que trabajan en base a relaciones las cuales pueden ser entre dos o más tablas, generando una nueva tabla a partir de los registros que cumplen con el criterio de correspondencia.

- **Componentes de una base de datos relacional y términos técnicos.**

Una base de datos relacional consiste de varios componentes.

**Tablas y relaciones.** Cada base de datos consiste de componentes fundamentales llamados relaciones. Las relaciones son las asociaciones entre tablas, las tablas consisten en varias líneas y en un número fijo de columnas.

**Llaves primarias.** Para identificar una fila de datos, cada tabla debe tener una columna que contenga valores únicos. Esta columna es conocida como la llave primaria de una tabla. La columna que es llave primaria no debe tener valores duplicados. Cada base de datos sofisticados deberá de poner atención en esto, para satisfacer las demandas del concepto de una base de datos relacional.

**Llaves foráneas.** Las llaves foráneas son columnas que están relacionadas a ciertas columnas de otra tabla. Los valores de la llave foránea en la columna son tratados de acuerdo a las restricciones definidas para la columna.

**Constraints** Las constraints son atributos definidos para un cierto objeto. Constraints pueden ser incluso restricciones o reglas para definir como se llevarán a cabo ciertas operaciones.

**Tipo de datos.** Cada columna en una tabla debe de tener un tipo de dato predefinido.

**Índices.** Los índices son utilizados para acelerar las consultas a la base de datos. Internamente los índices son estructuras complejas de árbol que permiten el rápido acceso a registros individuales.

**Secuencias** Las secuencias son utilizadas para generar una lista consecutiva de números que son únicos

**Joins** En el modelo de Base de datos relacional, los datos están almacenados en tablas. Si la información de más de una tabla necesita ser recuperada de la base de datos, los joins necesitan ser llevados a cabo. Joining significa que las tablas están conectadas con otras tablas utilizando una columna en común.

**Disparadores** Todas las bases de datos sofisticadas soportan los disparadores. Los disparadores son una característica que puede ser usada para empezar ciertas funciones cada vez que un evento predefinido ocurra. Los disparadores son definidos en las tablas y pueden ser disparados cuando alguna de estas operaciones ocurra: INSERT, UPDATE, o DELETE.

**Vistas** Las vistas son tablas virtuales. Esto significa que los usuarios pueden seleccionar datos de una vista justo como si lo seleccionaran de una tabla. Sin embargo, en realidad una vista no es una tabla real, si no que es el resultado de ejecutar una declaración SQL.

**Expresiones combinadas.** Son utilizadas para realizar operaciones con múltiples líneas devueltas por una consulta.

**Funciones combinadas.** Son utilizadas para realizar cálculo de datos, como obtener el máximo, el mínimo o el promedio.

## 1.3 SQL

Es una abreviatura de *Structured Query Language* (lenguaje de consulta estructurado). SQL es utilizado para controlar todas las funciones que provee un DBMS para los usuarios.

Es una herramienta para organizar, manejar y consultar información almacenada en una base de datos.

### 1.3.1 Componentes de SQL

El DDL (lenguaje de definición de datos), DML (lenguaje de manipulación de datos) y DCL (lenguaje de control de datos)

**DDL:** Permite a los usuarios definir la estructura y organización de los datos almacenados así como sus relaciones entre los ítems de los datos ya almacenados. No sólo permite la especificación de un conjunto de relaciones, sino también proporcionar la información relativa a éstas, conteniendo:

El esquema de cada relación.

El dominio de valores asociado a cada atributo.

Las restricciones de integridad.

El conjunto de índices que se deben de mantener para cada relación.

La información de seguridad y de autorización de cada relación.

La estructura de almacenamiento físico de cada relación en el disco.

**DML:** Permite al usuario o a una aplicación actualizar la base de datos agregando nueva información, eliminando viejos datos y modificando previamente los datos almacenados.

**DCL:** Se refiere a la autorización que el usuario puede dar a otros para ver o manipular información de su perfil.

### **1.3.2 Integridad**

El LDD de SQL incluye comandos para especificar las restricciones de integridad que deben cumplir los datos almacenados en la base de datos.

**Definición de vistas** El LDD contiene comandos para la definición de vistas.

**Control de transacciones.** SQL tiene comandos para especificar el comienzo y el final de las transacciones.

SQL puede ser utilizado para restringir la capacidad de los usuarios para recuperar, agregar y modificar datos por lo cual se protege almacenada de usuarios no autorizados la integridad de la información.

**Compartir información.** SQL es utilizado para coordinar el proceso de compartir datos entre los usuarios, asegurando que ninguno de ellos interfiere con el otro.

**1.3.3 Integridad de datos.** SQL define una característica de integridad en la base de datos, protegiéndolos de una posible corrupción debido a una inconsistencia de actualizaciones o fallos en el sistema. SQL es, de este modo, un lenguaje comprensivo para controlar e interactuar con un sistema gestor de base de datos.

## 1.4 Arquitectura de aplicaciones multinivel

La arquitectura de los sistemas de bases de datos se ve muy influida por el sistema informático subyacente sobre el que se ejecuta el sistema. Pueden estar centralizados o ser de tipo cliente-servidor, en los que una máquina servidora ejecuta el trabajo en nombre de multitud de máquinas clientes. Las aplicaciones de bases de datos suelen dividirse en dos o tres partes.

### 1.4.1 Arquitectura de dos capas

En una arquitectura de dos capas, la aplicación se divide en un componente que reside en la máquina cliente, que llama a la funcionalidad del sistema de base de datos en la máquina servidora mediante instrucciones del lenguaje de consultas, como se observa en la figura 1.4. Los estándares de programas de aplicación como ODBC y JDBC se usan para la interacción cliente-servidor.

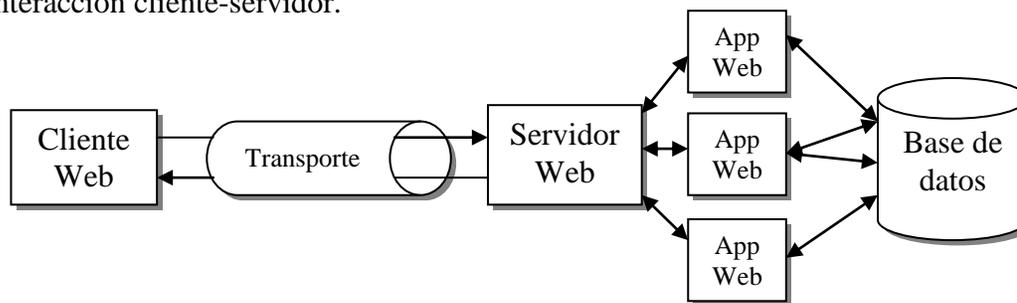


Figura 1. 4 Arquitectura de dos capas

- **Arquitectura cliente/servidor**

Esta arquitectura consiste básicamente en un cliente que realiza peticiones a otro programa -el servidor- que le da respuesta. Aunque esta idea se puede aplicar a programas que se ejecutan sobre una sola computadora es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

- **Front-end y Back-end**

En los sistemas desarrollados bajo la arquitectura cliente-servidor el software se divide en dos niveles: al software con el que se construye la interfaz que ve el usuario final

(programas de aplicación que opera el usuario) se le denomina front-end y al software con el que se crea y administra la BD se le conoce como back-end.

- **Características que debe de cumplir un software manejador de base de datos para arquitectura cliente-servidor.**

Todo servidor de bases de datos para esta arquitectura debe contar con ciertas características que lo hagan seguro, confiable y capaz de poder administrar bases de datos que se encuentren en un solo punto o distribuidas para poder asegurar la integridad de la información.

La capacidad de tener acceso a sitios remotos y transmitir consultas y datos entre los diversos nodos de la red.

La capacidad de seguir la pista todos los procesos y transacciones en el catálogo del sistema administrador de la base de datos.

La capacidad de elaborar estrategias de ejecución para consultas y transacciones que tienen acceso a datos de más de un sitio.

La capacidad de decidir a cual copia de un elemento de información replicado se tendrá acceso.

La capacidad de mantener la consistencia de las copias de un elemento de información replicado.

La capacidad de recuperarse de caídas de nodos individuales sin afectar a toda la red.

### **Software para arquitectura cliente-servidor**

En la actualidad existe una gran cantidad de software disponible en el mercado tanto de sistemas operativos como de lenguajes de programación y servidores de bases de datos que pueden trabajar y aprovechar las ventajas de la arquitectura cliente-servidor.

### **Software para el servidor**

Sistemas operativos	SOLARIS, UNIX, NOVELL, OS/2, WINDOWS
Sistemas de gestión de base de datos (SGBD) (back-end)	PostgreSQL, ORACLE, SQL-SERVER, SYBASE, Microsoft Access

## Software para el cliente

Sistemas operativos

DOS, NOVELL, UNIX, OS/2, WINDOWS

Servidor de base de datos (front-end)

C++, VISUAL BASIC, VISUAL FOXPRO

### 1.4.2 Arquitectura de tres capas

Una arquitectura de tres capas está constituida por las capas de presentación, lógica y de datos.

La capa de presentación es la interfaz del usuario de la aplicación, la cual recopila los datos de entrada y visualiza los resultados.

La capa lógica toma las entradas de la capa de presentación y realiza algunas operaciones sobre los datos. Finalmente devuelve el resultado a la presentación.

La capa de datos permite el almacenamiento de la información, misma que la capa lógica podrá consultar o actualizar, proporcionando abstracción de forma que los datos no necesiten ser copiados en la capa lógica y se puedan actualizar fácilmente.

La máquina cliente actúa simplemente como una parte visible al usuario y no contiene ninguna llamada directa a la base de datos. En vez de eso el cliente se comunica con un servidor de aplicaciones. Este a su vez se comunica con el sistema de bases de datos para tener acceso a la información. En la figura 1.5 se ilustra esta arquitectura.

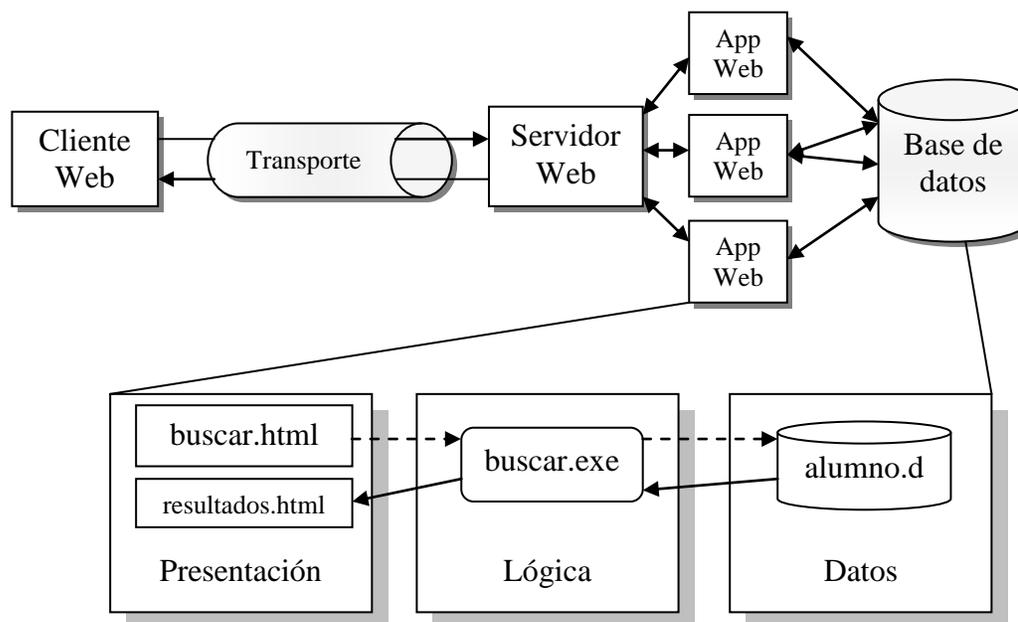


Figura 1.5 Arquitectura de tres capas

Muchas de las tecnologías para construir aplicaciones integran la funcionalidad de una o más de estas capas. En la tabla 1.1 se clasifican algunas de las más utilizadas.

Fabricante	Tecnología
Microsoft	Active Server Pages (ASP) ASP.NET ISAPI Common Object Model (COM)
Sun Microsystems IBM Websphere BEA Weblogic	Java Enterprise Edition, incluyendo Java Servlets Java Server Pages (JSP) CORBA JavaScript
Apache Software Foundation	PHP (Hypertext-Preprocessor) Jakarta (Java del lado del servidor)
(ninguno)	HTML CGI (incluyendo perl)

Tabla 1. 1

## 1.5 Lenguajes De Programación

Un lenguaje de programación es un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Es utilizado para controlar el comportamiento físico y lógico de una máquina. El lenguaje de programación debe ser universal. Es decir, cualquier problema debe tener una solución que puede ser programada en el lenguaje y dicha solución ser implementada en cualquier computadora. El lenguaje de programación debe ser implementable en una computadora, es decir; debe ser posible ejecutar un programa en términos del lenguaje en cualquier máquina.

Cada lenguaje tiene sintaxis y semántica:

- La sintaxis de un lenguaje de programación está relacionada con la forma de los programas, por ejemplo, cómo es que las expresiones, comandos, declaraciones, etc. son puestos juntos en un programa.

- La semántica de un lenguaje de programación está relacionada con el significado de los programas; por ejemplo, cómo se comportarán cuando se ejecutan en una computadora.

La sintaxis de un lenguaje influye en cómo los programas son escritos por el programador, leídos por otro programador y traducidos por la computadora. La semántica de un lenguaje determina cómo los programas son compuestos por el programador, entendidos por otros programadores e interpretados por la computadora.

### **1.5.1 Paradigma de programación**

Al estudio de los lenguajes en cuanto al enfoque del proceso de programación se le denomina paradigma de la programación. Es el modelo básico de construcción de programas. Permite producir programas conforme a directrices tales como diseñar un programa mediante una secuencia de instrucciones que operan sobre unos datos de entrada y producen un resultado de salida.

### **1.5.2 Clasificación por paradigmas de programación**

#### **Paradigma Imperativo o paradigma por procedimientos**

El paradigma por procedimientos, es tal vez el más conocido y utilizado en el proceso de programación, donde los programas se desarrollan a través de procedimientos. Debido a la estrecha relación con la arquitectura de la máquina, los lenguajes de programación imperativa pueden ser implementados muy eficientemente.

#### **Paradigma Declarativo**

El paradigma declarativo o paradigma de programación lógica se basa en el hecho que un programa implementa una relación antes que una correspondencia. Debido a que las relaciones son mas generales que las correspondencias (identificador - dirección de memoria), la programación lógica es potencialmente de más alto nivel que la programación funcional o la imperativa. El auge del paradigma declarativo se debe a que el área de la lógica formal de las matemáticas ofrece un sencillo algoritmo de resolución de problemas adecuado para usarse en un sistema de programación declarativo de propósito general.

## **Paradigma Estructurado**

La programación se divide en bloques (procedimientos y funciones) que pueden o no comunicarse entre sí. Además la programación se controla con secuencia, selección e iteración. Permite reutilizar código programado y otorga una mejor comprensión de la programación.

## **Paradigma Orientado a Objetos**

El paradigma orientado a objetos se basa en los conceptos de objeto y clase. El primero de ellos es una variable equipada con un conjunto de operaciones y atributos que le pertenecen o están definidas para ellos. Una de las características importantes de los lenguajes orientados a objetos es que las definiciones de los objetos pueden usarse una y otra vez para construir múltiples objetos con las mismas propiedades o modificarse para construir nuevos objetos con propiedades similares pero no exactamente iguales.

## **Paradigma Funcional**

La programación funcional se caracteriza por el uso de expresiones y funciones. Un programa dentro del paradigma funcional, es una función o un grupo de funciones compuestas por funciones más simples estableciéndose que una función puede llamar a otra, o el resultado de una función puede ser usado como argumento de otra función.

### **1.6 Herramientas de programación**

Son aquellas que permiten a los desarrolladores de software editar, interpretar, compilar, probar y depurar todos los programas fuentes de una aplicación o un sistema, con la finalidad de que cada vez sea más rápido y fácil el desarrollo de los programas. Existen diversas herramientas desde editores y compiladores hasta generadores de código automático, como lo es CASE (Ingeniería de Software Asistida por Computadora). Esto es la representación de sistemas a un nivel de abstracción más alto que el de los lenguajes de programación convencionales.

Estas herramientas proporcionan métodos para diseñar y documentar procesos de desarrollo en la ingeniería de software, incluyendo el análisis, diseño y programación.

### 1.6.1 UML

El Lenguaje de Modelado Unificado (UML) es un lenguaje gráfico para visualizar, especificar, construir y documentar los componentes de un sistema de software. EL UML ofrece una forma estándar para describir un plano del sistema, funciones del sistema, así como también expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

Se usa para definir un sistema de software; para detallar los componentes que forman parte del sistema. Puede utilizarse en una gran variedad de formas para soportar una metodología de desarrollo de software pero no especifica en sí mismo qué metodología o proceso.

El UML se usa como una parte de un proceso de desarrollo de software, con el soporte de la herramienta CASE apropiada, para definir los requisitos, las interacciones y los elementos del sistema de software propuesto. La naturaleza exacta del proceso depende de la metodología de desarrollo usada.

### 1.7 Arquitectura MVC (Modelo/Vista/Controlador)

El Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

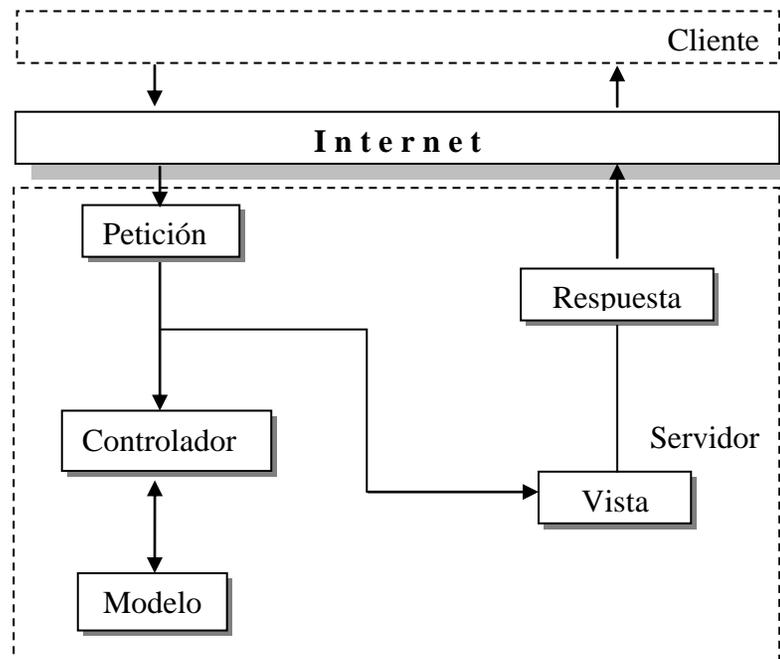


Figura 1.6 Arquitectura Modelo Vista Controlador

**Modelo:** Es la representación específica de la información con la cual el sistema opera. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene la responsabilidad de mantener enlaces entre el Modelo y sus Vistas, y notificar a las Vistas cuando cambia el Modelo.

**Vista:** Es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario.

**Controlador:** Es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del Modelo o por alteraciones de la Vista. Interactúa con el Modelo a través de una referencia al propio Modelo.

# Capítulo 2. Análisis

## 2.1 Proceso de Reinscripción

El proceso de reinscripción en la Facultad de Arquitectura para su licenciatura homóloga está integrado por las siguientes etapas:

- 1) Inscripción a Taller de Arquitectura.
- 2) Registro de Asignaturas.
- 3) Asignación de grupos.
- 4) Etapa de altas, bajas y cambios.

## 2.2 Inscripción a Taller de Arquitectura

Es el proceso en el que el Alumno elige uno, de los dieciséis talleres, para desarrollar su formación académica de acuerdo a sus intereses y aptitudes.

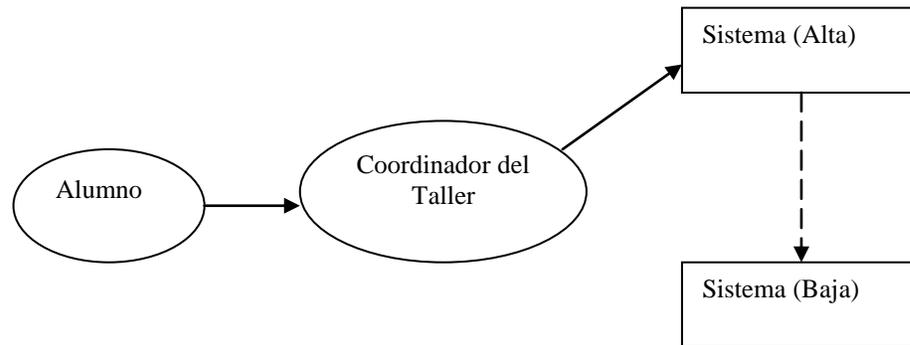
Es una inscripción directa a un grupo de la asignatura Taller de Arquitectura realizada por el coordinador del mismo Taller.

El alumno que se haya inscrito en un grupo de Taller tiene garantizada su inscripción a los grupos compatibles a dicho Taller, si y solo si cumple con los requisitos necesarios que establezca cada asignatura.

## 2.3 Consideraciones generales para inscripción a Taller

- El Alumno de primer ingreso selecciona el Taller al que quiere inscribirse, proceso que no forma parte del Sistema.
- En caso de reprobación 2 veces el Taller de Arquitectura, el alumno debe de inscribirse nuevamente con el Coordinador de Taller, quedando asentada la inscripción como extraordinario largo (EA).
- En la etapa de inscripción a talleres, el Alumno puede cancelar esa inscripción sólo con el coordinador del Taller con el que se inscribió.

- En la etapa de altas, bajas y cambios, el Alumno puede cancelar él mismo su inscripción al Taller; e inscribirse en algún otro siempre y cuando haya cupo.



La inscripción a Taller se realiza con el Coordinador del mismo, el cual da de alta al alumno directamente en el sistema. Una vez inscrito el Taller, se podrá darlo de baja sólo con el Coordinador. (Etapa de inscripción a Taller)

Figura 2. 1 Inscripción al Taller

- Para realizar cambio de Taller, el Alumno debe de haber aprobado dos asignaturas consecuentes (impar y par) de Taller de Arquitectura correspondientes a un mismo Taller.

### 2.3.1 Casos posibles al realizar la inscripción al Taller de Arquitectura

#### a) Primer caso

Un Alumno estuvo inscrito en el mismo Taller en el que solicita una inscripción, en un semestre impar.

Ejemplo: El coordinador del Taller Juan O' Gorman va a inscribir al alumno, el sistema encuentra que el alumno estuvo inscrito en ese mismo Taller correspondiente al tercer semestre.

Las opciones de taller para este alumno serán:

- Taller Juan O Gorman en el tercer semestre
- Taller Juan O Gorman en el cuarto semestre

#### b) Segundo caso

Un Alumno estuvo inscrito en el mismo Taller en el que solicita una inscripción, en un semestre par.

Ejemplo: El coordinador del Taller Juan O' Gorman va a inscribir al alumno, el sistema encuentra que el alumno estuvo inscrito en ese mismo Taller correspondiente al cuarto semestre.

Las opciones de taller para este alumno serán:

- Taller Juan O Gorman en el cuarto semestre
- Taller Juan O Gorman en el quinto semestre

### **c) Tercer caso**

Un Alumno estuvo inscrito en un distinto Taller en el que solicita una inscripción, en un semestre impar.

Ejemplo: El coordinador del Taller Juan O' Gorman va a inscribir al alumno, el sistema encuentra que el alumno estuvo inscrito en otro Taller correspondiente al quinto semestre.

No habrá opciones para que éste Alumno se inscriba en el taller Juan O'Gorman, sólo se mostrará un mensaje explicando el problema.

### **d) Cuarto caso**

Un Alumno estuvo inscrito en un distinto Taller en el que solicita una inscripción, en un semestre par.

Ejemplo: El coordinador del Taller Juan O' Gorman va a inscribir al alumno, el sistema encuentra que el alumno estuvo inscrito en otro Taller correspondiente al cuarto semestre.

Las opciones de taller para este alumno serán

- Taller Juan O Gorman en el quinto semestre

Al finalizar este proceso se tendrán Alumnos inscritos en los grupos de la materia Taller de Arquitectura.

## **2.4 Registro de asignaturas.**

El proceso que corresponde al registro de asignaturas que no son de tipo Taller (*obligatorias, selectivas internas y exámenes extraordinarios largos*) es realizado por el Alumno, el cual ingresa a la página de Asuntos Escolares, para iniciar el proceso de reinscripciones.

Las asignaturas obligatorias son aquellas que forman parte del mapa curricular, pero se imparten de manera complementaria a la clase de Taller.

Las asignaturas selectivas internas son materias optativas que se imparten dentro de la misma Facultad de Arquitectura.

El examen extraordinario largo es la inscripción como Oyente, de un Alumno durante la etapa de registro de asignaturas.

El registro de Asignaturas no consiste en una inscripción real a los cursos, se utiliza como una manera de conocer la demanda que existe para los grupos y evaluar si el cupo es adecuado.

### **2.4.1 Consideraciones generales para el Registro de asignaturas:**

En la etapa de registro:

- No se cuenta aún con todas las calificaciones de los Alumnos en el semestre en curso, por lo cual no se revisa la seriación por asignatura.
- No se manejan vacantes y el registro no garantiza la inscripción a los grupos solicitados.
- Se valida la seriación de etapas.
- Se valida que no ocurra traslape de los horarios registrados.
- Se valida que el Alumno no registre más de un curso de Extensión Universitaria por semestre.
- Los cursos selectivos manejados son sólo los impartidos dentro de la propia Facultad de Arquitectura.

- Se valida que el Alumno cumpla el requisito de curso de cómputo, para los cursos selectivos que así lo requieran.

En esta etapa el Alumno puede realizar los siguientes movimientos:

- Dar de alta y/o baja su registro en materias, con excepción del Taller (inscrito previamente en el proceso de inscripción de Taller de Arquitectura).
- Ingresar al sistema de registro tantas veces sea necesario, dentro del rango de tiempo determinado para esto.
- Imprimir comprobantes de asignaturas registradas.

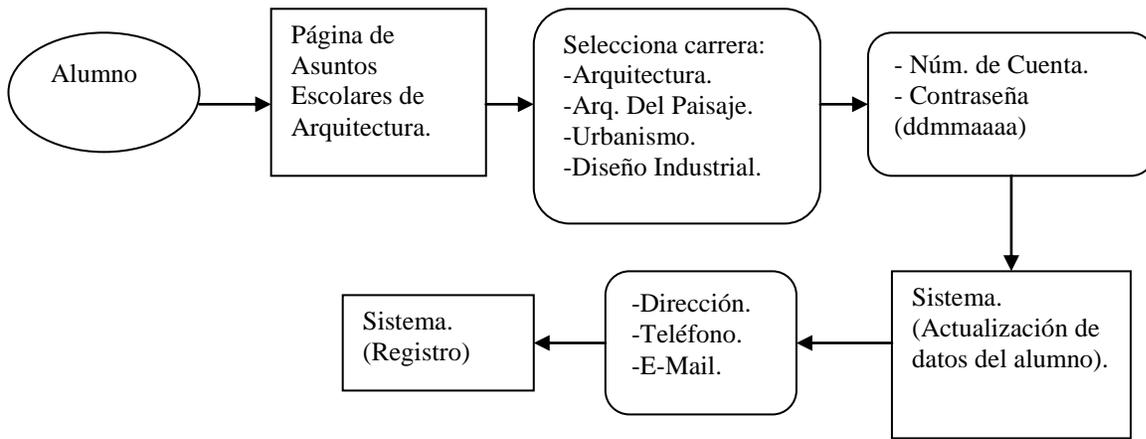
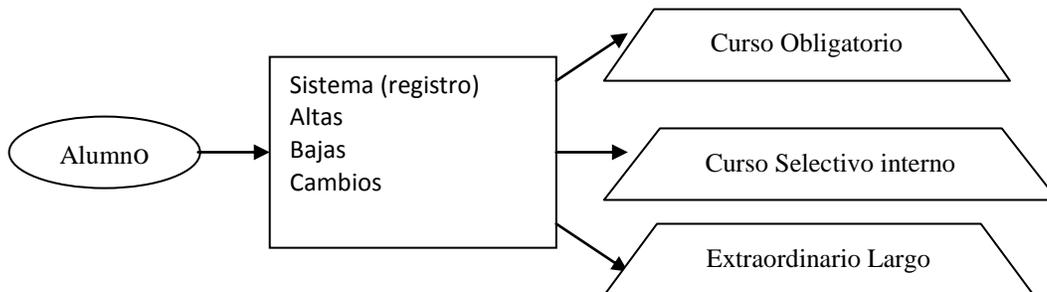


Figura 2.2 Al momento de empezar el registro de asignaturas aparecerá una pantalla solicitando el número de cuenta y contraseña del Alumno. A continuación se desplegará un formato con los datos personales del estudiante, en este paso se deberá actualizar la información.

Figura 2. 2 Registro de Asignaturas



En este proceso el Alumno puede dar de alta y/o baja materias, con excepción del Taller.

Figura 2. 3 Registro de Asignaturas

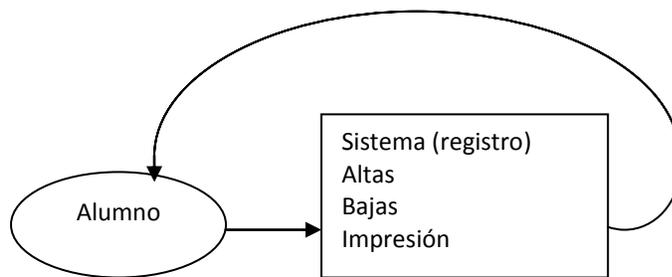


Figura 2. 4 El Alumno puede ingresar al registro tantas veces lo requiera dentro del periodo de tiempo determinado

El resultado de este proceso son los registros de peticiones para inscripción de los alumnos, los cuales son utilizados en la etapa de Asignación de grupos.

## 2.5 Asignación de grupos

En esta fase el sistema obtiene el avance de etapas de cada Alumno y procesa la información que el Alumno haya solicitado previamente en el Registro de asignaturas, con el fin de asignar grupos para el próximo semestre.

Este proceso formaliza la inscripción de los Alumnos en los grupos solicitados en la etapa de Registro, siempre y cuando el Alumno cumpla los requisitos necesarios de cada asignatura.

### 2.5.1 Consideraciones generales para asignación de grupos

El proceso es ejecutado por el sistema tras finalizar el periodo de Registro de Asignaturas.

Al realizar el proceso de asignación se toma en cuenta a toda la población estudiantil de la licenciatura en Arquitectura que se encuentre en estado Activo.

- A los alumnos que solicitan una asignatura compatible con su taller, no se les valida el cupo; para el resto de los Alumnos sí se les toma en cuenta.
- Al conocer los Alumnos que se registraron para un grupo determinado se maneja una jerarquía de Alumnos: Se ordenan a los alumnos de manera descendente empezando por aquellos de los semestres más adelantados y los promedios más altos.

- Solamente se puede cursar un máximo de 51 créditos por semestre, sin tomar en cuenta los créditos de la asignatura Extensión Universitaria.
- El registro a Práctica Profesional Supervisada es manejado en este sistema, no obstante, debe terminarse la inscripción del mismo en las ventanillas de la Secretaría de Asuntos Escolares.
- Se valida la seriación por Etapa.
- Se valida la seriación por Asignatura.
- Se valida los requisitos para Cursos Selectivos.
- Se valida los requisitos para Práctica Profesional Supervisada.
- Se establece el tipo de inscripción: Ordinaria o Extraordinaria (EA).

Días después de la asignación de grupos los alumnos pueden imprimir un comprobante provisional de inscripción, en el cual se especifican los problemas encontrados durante la asignación de grupos (porque no entró a cierto grupo: no hay cupo, traslape de horarios, no cumple seriación por etapa, etc). Así mismo indica los horarios para altas, bajas y cambios de grupos. Para generar este horario se toman en cuenta los promedios de los Alumnos.

**a) Consideraciones para asignación de cursos obligatorios.**

La jerarquía de asignación para estos cursos es la siguiente:

- Alumnos del Taller por promedio: aquellos Alumnos que realizaron inscripción a un Taller tienen garantizada su inscripción a las asignaturas solicitadas en la etapa de registro, siempre y cuando los grupos de éstas sean compatibles con el Taller en que se inscribieron.
- Alumnos regulares por promedio: Alumnos que no se han rezagado en el avance de sus asignaturas. Si el alumno no acredita Taller o no lo inscribe ya no es considerado Alumno regular.
- Alumnos irregulares por promedio: Alumnos que se han rezagado en el avance de sus asignaturas sin que entren en la aplicación de los artículos 22 y 33 del Reglamento General de Inscripciones. Es decir, que estén rezagados, pero que aún puedan cursar todas sus materias de forma ordinaria.

- Alumnos en examen extraordinario largo “EA” por promedio: Alumnos que se encuentren en Artículo 22.

### 2.5.2 Jerarquía de la asignación de grupos



Figura 2. 5 Asignación de grupos: Alumnos del taller por promedio

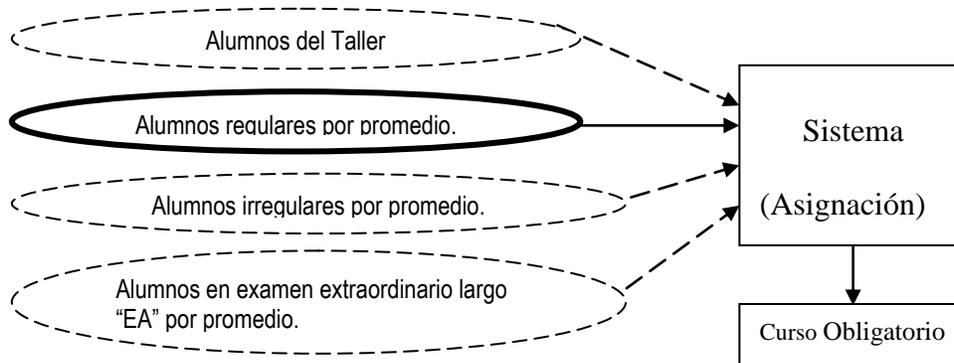


Figura 2. 6 Asignación de grupo: alumnos regulares por promedio

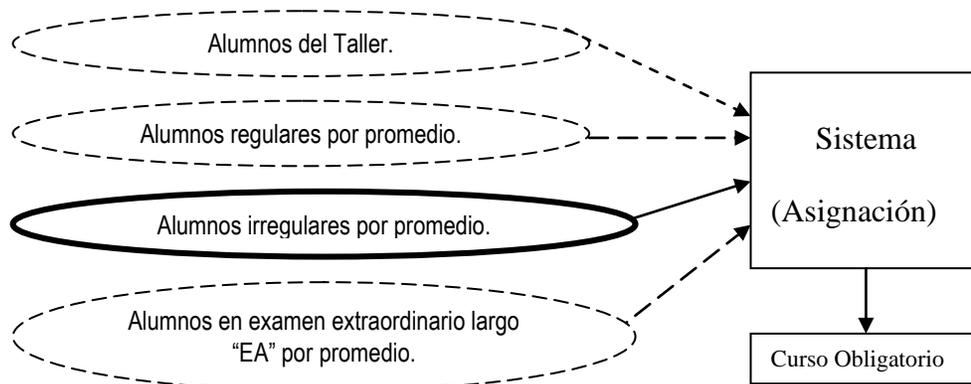


Figura 2. 7 Asignación de grupo: alumnos irregulares por promedio.

Al finalizar el proceso de asignación se tendrán Alumnos inscritos en los grupos de las asignaturas de Arquitectura.

## 2.6 Etapa de altas, bajas y cambios

En este proceso el Alumno puede realizar movimientos de altas o bajas de asignaturas obligatorias, selectivos internos y/o talleres, de acuerdo a sus necesidades.

### 2.6.1 Consideraciones generales para la etapa de altas, bajas y cambios

El sistema realiza las siguientes validaciones en la etapa de altas, bajas y cambios:

- Un Alumno puede ingresar al sistema a partir de la hora asignada en adelante.
- En esta etapa se manejan todas las validaciones presentes tanto en la etapa de Registro como en la de Asignación:
  - ✓ Se valida el cupo para todas las altas, sean o no compatibles los grupos con el Taller en que está inscrito el Alumno.
  - ✓ Se valida la seriación por etapas.
  - ✓ Se valida la seriación por asignaturas.
  - ✓ Se valida que no ocurra traslape de los horarios registrados.
  - ✓ Se valida que el Alumno no registre más de un curso de Extensión Universitaria por semestre.

- ✓ Los cursos selectivos manejados son sólo los impartidos dentro de la propia Facultad de Arquitectura.
- ✓ Se valida que el Alumno cumpla el requisito de curso de cómputo, para algunos cursos selectivos.
- ✓ Solamente se pueden inscribir cursos siempre que no se exceda un máximo de 51 créditos por semestre, sin tomar en cuenta Extensión Universitaria.

Después de realizar cambios alta/baja de asignaturas, el Alumno puede imprimir otro comprobante de reinscripción provisional (el definitivo lo dará Servicios Escolares después de iniciar las clases).

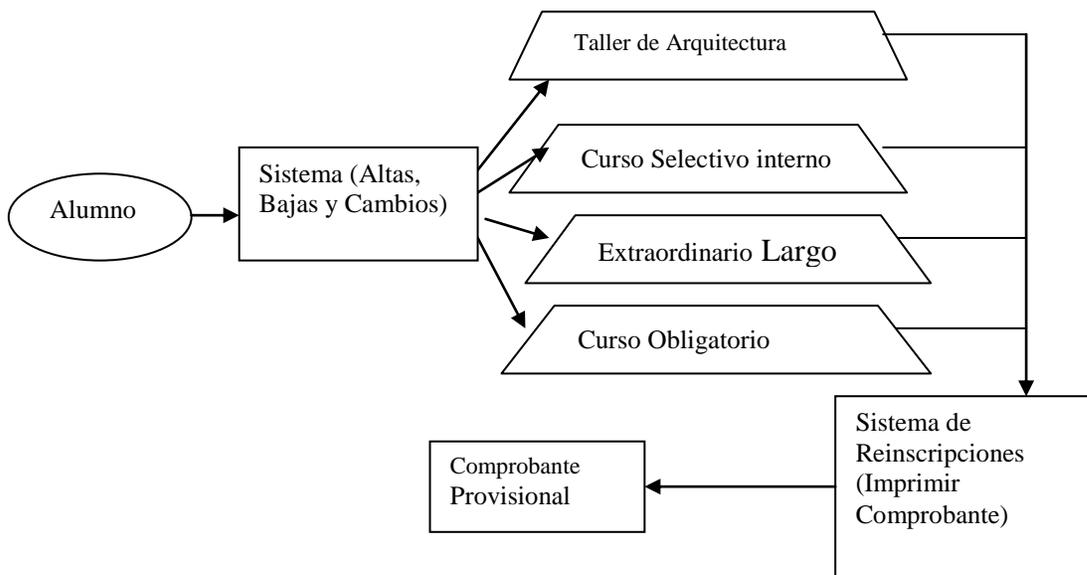


Figura 2. 8 Etapa de Altas, Bajas y Cambios

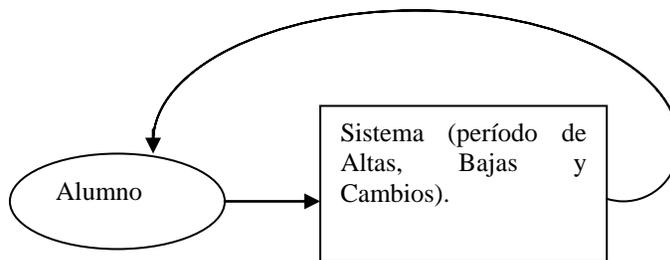


Figura 2. 9 El alumno puede ingresar tantas veces lo requiera dentro del periodo de tiempo determinado a partir de su horario para cambios.

## 2.7 Alcance del sistema

El presente trabajo contempla únicamente la reinscripción de la Licenciatura en Arquitectura, se espera que el trabajo realizado y la documentación generada sirvan como modelo de desarrollo para el proceso de reinscripción de las otras carreras que se imparten en la Facultad de Arquitectura.

A partir de los requerimientos descritos en el capítulo 2 Análisis, el sistema a desarrollar contempla una Base de Datos y una aplicación que realice el proceso de reinscripción, éste último que comprenda las siguientes fases:

- Inscripción a Taller. Correspondiente al Coordinador de Taller.
- Etapa de Registro. Correspondiente al usuario Alumno.
- Etapa de Asignación. Correspondiente al usuario Mantenimiento.
- Etapa de altas, bajas y cambios. Correspondiente al usuario Alumno

Se considera el desarrollo de aquellos módulos para las tareas que ejecuta el personal de Asuntos Escolares que, aunque no forman parte de las etapas anteriores, son necesarias para el correcto funcionamiento de ellas.

a) Funciones adicionales al usuario Mantenimiento:

- Cambiar calificación de un Alumno.
- Cancelar grupos de cursos. En este módulo se contemplan *cursos obligatorios* y *cursos selectivos internos*.
- Modificar cupo de grupos. En este módulo se contemplan *cursos obligatorios* y *cursos selectivos internos*.
- Actualización de curso de cómputo. Este curso se pide como requisito para algunos *cursos selectivos internos*.
- Actualización de etapas mínimas. Este se ocupa de actualizar el avance de la etapa cubierta en su totalidad por un Alumno (avance de bloques).
- Asignación de Horarios para Altas, Bajas y Cambios.

b) Funciones del usuario Jefe de Asuntos Escolares

- Inscribir de manera extemporánea a un Alumno.
- Consultar historial académico de un Alumno.

c) Funciones del usuario Coordinador de Carrera

- Cambiar cupo de Taller.

d) Funciones adicionales al usuario Alumno

- Actualizar datos. En éste módulo el Alumno debe actualizar sus datos personales para poder comenzar su proceso de reinscripción.

## **2.8 Escenarios**

### **2.8.1 Actor Alumno**

**Nombre del escenario: Realizar Altas, Bajas y Cambios (tras la etapa de asignación de horarios)**

**Descripción:** El Alumno realiza cambios en tiempo real en las asignaturas asignadas a partir de un horario establecido cuantas veces lo requiera.

#### **Flujo Básico:**

1. El escenario Ingresar a Trámites Académicos ha concluido con éxito previamente.
2. Se presentan en pantalla los datos de cursos y grupos en donde el Alumno está inscrito así como las opciones de Alta, Baja, Imprimir Comprobante y Cerrar sesión.
3. El Alumno ingresa las claves de asignatura y grupo.

4. El Alumno elige la opción de Alta/Baja de acuerdo a sus necesidades.
5. El sistema valida los requisitos necesarios para clave de curso ingresado.
6. Se muestra la información actualizada de todos los cursos y grupos en que el Alumno está inscrito.

**Nombre del escenario: Actualizar datos**

**Descripción:** El Alumno actualiza sus datos personales.

**Flujo Básico:**

1. El escenario Ingresar a Trámites Académicos ha concluido con éxito previamente.
2. La siguiente información del Alumno: domicilio, teléfono y correo electrónico se presenta en pantalla, distribuida en campos junto con la opción “Actualizar”.
3. El alumno actualiza sus datos y elige la opción “Actualizar”
4. Se actualiza la información y se muestra un mensaje informando que los datos han sido actualizados.

**Nombre del escenario: Registro de Asignaturas**

**Descripción:** El Alumno registra las peticiones de los grupos y asignaturas de su preferencia.

**Flujo Básico:**

1. El escenario Actualizar datos ha concluido con éxito previamente.
2. Se muestra el menú de opciones.
3. El Alumno selecciona la opción “Comenzar inscripción”.
4. Se valida que la fecha actual corresponda dentro del periodo determinado para la etapa de Registro.
5. Se obtienen los datos de asignaturas y de grupos registrados previamente.
6. Se muestran los cursos registrados previamente, las opciones Alta, Baja, Imprimir Comprobante y Cerrar sesión.
7. Se solicitan la clave de la asignatura y el grupo.
8. El Alumno proporciona una clave de asignatura y el grupo y elige la opción de “Alta/Baja”.

9. Se valida que Alumno cumpla los requisitos para la asignatura, de acuerdo a la opción elegida.
10. El sistema muestra en pantalla los cursos registrados junto con las opciones.

### **2.8.2 Para el usuario Mantenimiento**

**Nombre del escenario: Modificar el cupo de cursos.**

**Descripción:** El usuario Mantenimiento cambia el cupo de los cursos obligatorios.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestra un menú de opciones al actor Mantenimiento.
3. El usuario Mantenimiento selecciona la opción “Modificar cupo de cursos”.
4. Se muestran las etapas correspondientes a la carrera de Arquitectura.
5. Se selecciona una de estas etapas.
6. El sistema muestra las asignaturas correspondientes a la etapa elegida.
7. Se selecciona uno de los cursos.
8. Se muestran los grupos del curso elegido.
9. Se selecciona uno de los grupos.
10. Se muestra la información del cupo y demanda además de un campo para establecer el nuevo cupo de dicho grupo.
11. El usuario Mantenimiento proporciona el cupo y elige la opción “Actualizar cupo”.
12. El sistema muestra un mensaje informando que el cupo ha sido actualizado.

**Nombre del escenario: Cancelar grupos de Cursos.**

**Descripción:** El usuario Mantenimiento elimina el registro de un grupo de cursos obligatorios.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestra un menú de opciones al actor Mantenimiento.

3. El usuario Mantenimiento selecciona la opción “Cancelar grupos de Cursos Obligatorios”.
4. Se muestran las etapas que componen la carrera de Arquitectura.
5. Se selecciona una de las etapas.
6. Se muestran los cursos de la etapa seleccionada.
7. Se selecciona uno de los cursos.
8. Se muestran en pantalla los grupos.
9. Se selecciona uno de los grupos.
10. Se muestra el nombre, clave de grupo ordinario así como del extraordinario y del cupo del grupo seleccionado.
11. Se elige la opción “Cancelar grupo”.
12. El Sistema muestra un mensaje informando que la cancelación del grupo se ha realizado.

**Nombre del escenario: Cambiar calificación de un Alumno**

**Descripción:** El usuario Mantenimiento cambia la calificación de una asignatura de la historia académica de un Alumno.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestra un menú de opciones al actor Mantenimiento.
3. El usuario Mantenimiento selecciona la opción “Cambiar calificación de un Alumno”.
4. Se solicitan el número de cuenta del Alumno y la clave de la asignatura.
5. Se obtienen los datos de la historia académica.
6. Se muestran los datos del Alumno, la calificación obtenida en la asignatura y el período.
7. Se confirma que los datos son correctos.
8. Se solicita la nueva calificación.
9. Se ingresa la nueva calificación.
10. Se actualiza la nueva calificación y la etapa mínima del Alumno.

11. El Sistema muestra un mensaje informando que el cambio de calificación se ha realizado.

**Nombre del escenario: Actualización de etapas mínimas**

**Descripción:** El usuario Mantenimiento inicia el proceso mediante el cual se actualiza el avance de etapas de los Alumnos de Arquitectura.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestra un menú de opciones al actor Mantenimiento.
3. El usuario Mantenimiento selecciona la opción “Actualización de etapas mínimas”.
4. Se obtienen la seriación de asignaturas de Arquitectura.
5. Se escoge un Alumno de Arquitectura.
6. Se obtienen las asignaturas de la historia académica del Alumno.
7. Se verifica el caso del Alumno en turno.
8. Se actualiza la etapa mínima correspondiente al caso del Alumno en turno.
9. Se repite el proceso a partir del paso 5 del flujo básico hasta terminar con todos los alumnos.
10. El Sistema muestra un mensaje informando que el proceso fue finalizado.

**Nombre del escenario: Asignación de grupos**

**Descripción:** El usuario Mantenimiento inicia el proceso mediante el cual se formalizan las inscripciones de los Alumnos de Arquitectura a las asignaturas solicitadas en la etapa de registro.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestra un menú de opciones al actor Mantenimiento.
3. El usuario Mantenimiento selecciona la opción “Asignación de grupos”.
4. Se valida que la fecha corresponda a la asignada a la etapa de Asignación.
5. Se obtienen las peticiones de los Alumnos realizadas en la etapa de Registro.

6. Se validan las peticiones según la historia académica y el Taller de cada Alumno.
7. Se muestra un mensaje indicando que el proceso de Asignación de grupos ha finalizado.

**Nombre del escenario: Actualización de curso de cómputo**

**Descripción:** El usuario Mantenimiento marca como cumplido el requisito de curso de cómputo para un Alumno.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestra un menú de opciones al actor Mantenimiento.
3. El usuario Mantenimiento selecciona la opción “Actualización de curso de cómputo”.
4. Se solicita el número de cuenta del Alumno.
5. Se actualiza el requisito de curso de cómputo para el Alumno.
6. Se muestra un mensaje indicando que el requisito fue actualizado para el Alumno.

### **2.8.3 Para el actor Coordinador de Taller**

**Nombre del escenario: Inscribir a un Alumno**

**Descripción:** El usuario Coordinador de Taller inscribe a un Alumno al Taller a su cargo.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestran un menú de opciones al Coordinador de Taller.
3. El Coordinador de Carrera escoge la opción “Inscripción a Taller”.
4. Se solicita el número de cuenta del Alumno.
5. Se ingresa el número de cuenta.
6. Se muestran los datos del Alumno.
7. Se indica que los datos son correctos.
8. Se verifica el caso del Alumno y se muestran opciones de acuerdo a su caso.
9. Se elige una opción de grupo.

10. Se muestra un mensaje indicando que el Alumno se ha inscrito en el grupo seleccionado del Taller del Coordinador.

**Nombre del escenario: Cancelar inscripción al Taller**

**Descripción:** El usuario Coordinador de Taller cancela la inscripción de un Alumno en el Taller a su cargo.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestran un menú de opciones al Coordinador de Taller.
3. El Coordinador de Carrera escoge la opción “Cancelar Inscripción a Taller”.
4. Se solicita el número de cuenta del Alumno.
5. El Coordinador de Taller proporciona el número de cuenta proporcionado del Alumno.
6. Se valida que el Alumno se haya inscrito previamente a un grupo del taller correspondiente al Coordinador en turno.
7. Se muestran nombre, número de cuenta, carrera y Taller en que está inscrito el Alumno.
8. El Coordinador confirma que los datos son correctos.
9. Se cancela la inscripción del Alumno al Taller del Coordinador en turno.
10. Se actualiza la demanda del grupo del Taller en que Alumno estaba inscrito.
11. Se muestra un mensaje informando que la inscripción del Alumno al Taller fue cancelada.

**2.8.4 Para el actor Coordinador de Carrera**

**Nombre del escenario: Modificar el cupo del Taller.**

**Descripción:** El usuario Coordinador de Carrera cambia el cupo de un grupo de Taller.

**Flujo básico:**

1. El escenario Ingresar a trámites académicos ha concluido con éxito previamente.
2. Se muestran los datos correspondientes al nombre de cada uno de los Talleres de la carrera de Arquitectura.
3. El Coordinador de Carrera escoge un determinado Taller.

4. Se muestran las etapas de la carrera de Arquitectura.
5. El Coordinador de Carrera elige una etapa.
6. El sistema muestra los grupos correspondientes a la etapa elegida.
7. El coordinador de Carrera selecciona el grupo al cual quiere realizar el cambio de cupo.
8. El sistema muestra el cupo y demanda del grupo del Taller elegido junto con un campo para ingresar el cupo deseado.
9. El actor Coordinador de Carrera ingresa nuevo cupo y selecciona “Actualizar cupo”.
10. El sistema muestra un mensaje indicando que se ha realizado el cambio de cupo.

### **2.8.5 Para el actor Jefe de Asuntos Escolares**

#### **Nombre del escenario: Realizar inscripciones de casos extemporáneos**

**Descripción:** El Jefe de Asuntos Escolares inscribe de manera extemporánea al Alumno en un curso, en grupo ordinario o extraordinario (asignaturas de tipo obligatorio y/o selectivo interno), validando únicamente su número de cuenta y su historial en dicho curso.

#### **Flujo Básico:**

1. El escenario Ingresar a Trámites Académicos ha concluido con éxito previamente.
2. Se muestra el menú de opciones para el actor Jefe de Asuntos Escolares.
3. El Jefe de Asuntos Escolares selecciona la opción de “Inscribir Cursos”.
4. Se muestra un campo para el número de cuenta del Alumno.
5. El Jefe de Asuntos Escolares ingresa el número de cuenta del Alumno y selecciona la opción “Enviar”.
6. Se muestran los datos del alumno, junto con las opciones de “Datos son correctos” o “Datos incorrectos” y “Consultar historial”.
7. El Jefe de Asuntos Escolares selecciona la opción “Datos son correctos”.
8. Se muestran los datos de las etapas al Jefe de Asuntos Escolares.
9. El Jefe de Asuntos Escolares selecciona una etapa.
10. Se muestran en pantalla las asignaturas de la etapa elegida.
11. El Jefe de Asuntos Escolares selecciona la asignatura.
12. El Jefe de Asuntos Escolares elige un grupo de la asignatura, selecciona la opción “Inscribir”.

13. El sistema muestra un mensaje indicando que el Alumno ha sido inscrito.

**Nombre del escenario: Consultar Historial Académico del Alumno**

**Descripción:** El Jefe de Asuntos Escolares consulta el Historial Académico de un Alumno.

**Flujo Básico:**

1. El escenario Ingresar a Trámites Académicos ha concluido con éxito previamente.
2. Se muestra el menú de opciones para el actor Jefe de Asuntos Escolares.
3. El Jefe de Asuntos Escolares selecciona la opción de “Inscribir Cursos”.
4. Se muestra un campo para el número de cuenta del Alumno.
5. El Jefe de Asuntos Escolares ingresa el número de cuenta del Alumno y selecciona la opción “Enviar.
6. Se muestran los datos del alumno, junto con las opciones de “Datos son correctos” o “Datos incorrectos” y “Consultar historial”.
7. El Jefe de Asuntos Escolares selecciona la opción “Consultar Historial”.
8. Se muestran las asignaturas, sus claves, calificaciones y periodo en el que fueron cursadas por el Alumno.

# Capítulo 3 Diseño del sistema

El Sistema de reinscripciones de Arquitectura esta estructurado en Módulos correspondientes a las diferentes actividades que cada usuario lleva a cabo.

## 3.1 Alumno

El Módulo del Alumno esta representado por la siguiente carta de estructura:

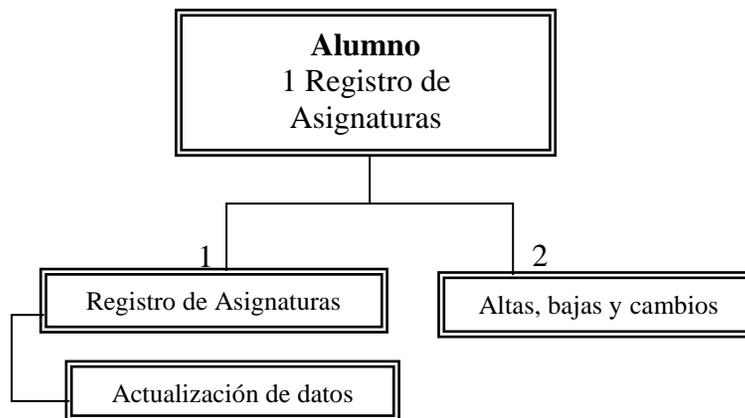


Figura 3. 1 Carta de estructura del usuario Alumno

**Registro de Asignaturas\*** Módulo que realiza el proceso de registro de las peticiones de asignaturas de un Alumno.

**Altas, Bajas y Cambios\*** Módulo cuyo objetivo es el de realizar movimientos de altas, bajas y/o cambios de cursos (horario de asignaturas para semestre en curso) de un Alumno.

**Actualización de datos\*** Módulo que corresponde a la actualización de información personal de un Alumno.

\*Para mayor información acerca de los módulos consultar capítulo 2 Análisis

### 3.2 Mantenimiento

El Módulo del usuario Mantenimiento esta representado por la carta de estructura de la figura 3.2:

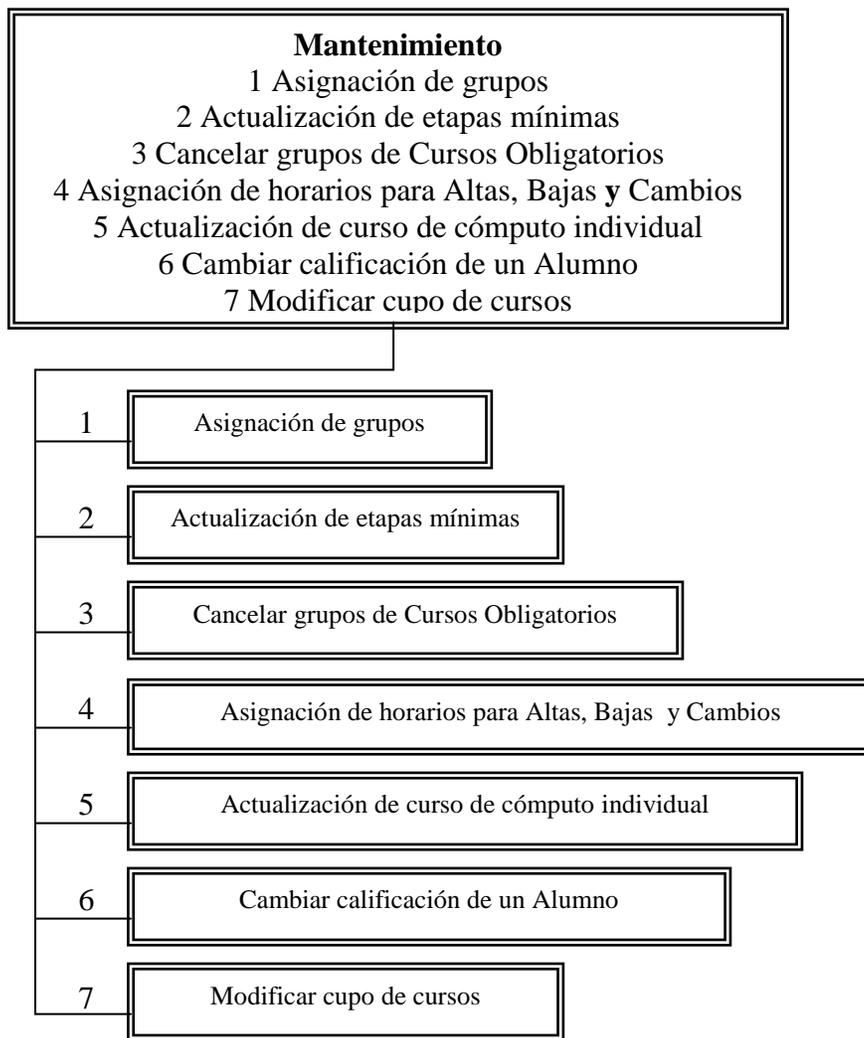


Figura 3. 2 Carta de estructura del usuario Mantenimiento

#### **Asignación de horarios para Altas, Bajas Cambios.\***

Módulo que corresponde a establecer un horario a cada Alumno después del periodo de Asignación de grupos para realizar altas, bajas o cambios en su horario asignado.

#### **Modificar cupo de cursos.\***

Módulo cuya función es la de modificar el cupo de algún grupo correspondiente a una asignatura interna de la Facultad de Arquitectura.

\*Para mayor información acerca de los módulos consultar capítulo 2 Análisis

**Cancelar grupos de Cursos Obligatorios.\*** Módulo correspondiente a cancelar grupos de asignaturas obligatorias que se encuentren activos en el semestre en curso.

**Cambiar calificación de un Alumno.\*** Módulo cuya función es la de cambiar la calificación de una determinada asignatura de un Alumno.

**Actualización de etapas mínimas.\*** Módulo que realiza la actualización de avance de créditos de un Alumno.

**Actualización de curso de cómputo individual.\*** Módulo que realiza la actualización del requisito de curso de cómputo para los Alumnos que así lo requieran.

### 3.3 Coordinador de Taller

El Módulo del usuario Coordinador de Taller esta representado por la carta de estructura de la figura 3.3:

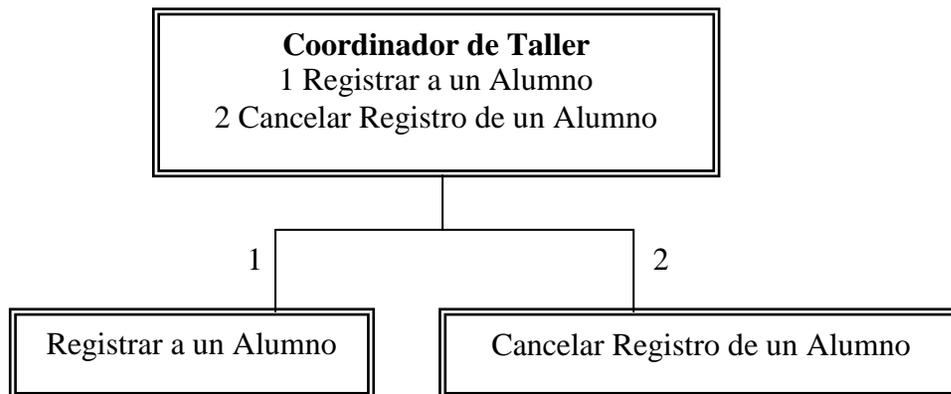


Figura 3. 3 Carta de estructura del usuario Coordinador de Taller

**Registrar a un Alumno\*** Módulo cuya función es la de inscribir a un Alumno a determinado Taller.

**Cancelar Registro de un Alumno.\*** Módulo que se encarga de cancelar el registro previo de un Alumno en un Taller.

\*Para mayor información acerca de los módulos consultar capítulo 2 Análisis

### 3.4 Coordinador de Carrera

El Módulo del usuario Coordinador de Carrera esta representado por la carta de estructura de la figura 3.4:

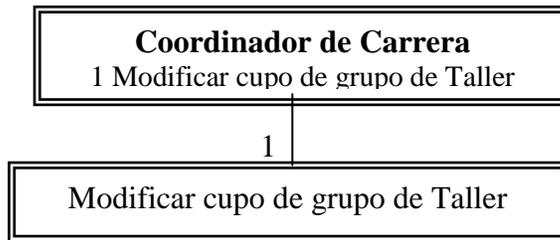


Figura 3. 4 Carta de estructura del usuario Coordinador de Carrera

**Modificar cupo de grupo de Taller.\*** Módulo cuya función es la de cambiar el cupo de algún grupo correspondiente a uno de los Talleres de la carrera de Arquitectura.

### 3.5 Jefe de Asuntos Escolares

El Módulo del usuario Jefe de Asuntos Escolares está representado por la carta de estructura de la figura 3.5:

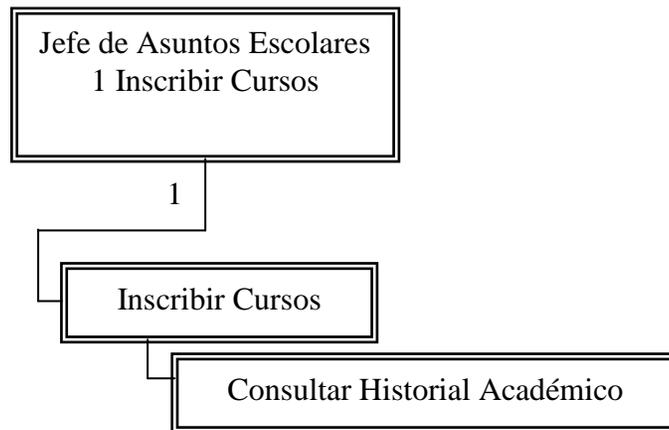


Figura 3. 5 Carta de estructura del usuario Jefe de Asuntos Escolares

**Inscribir Cursos.\*** Módulo que realiza inscripciones de un Alumno a las diferentes asignaturas obligatorias o selectivas internas de la Facultad de Arquitectura de manera extemporánea. Así mismo puede, de manera opcional, consultar dentro de este mismo Módulo el Historial Académico de un Alumno.

\*Para mayor información acerca de los módulos consultar capítulo 2 Análisis

### **3.6 Diagrama Entidad Relación**

Para diseñar la base de datos se debe utilizar el modelo entidad relación, el cual está basado en una percepción del proceso real que se lleva a cabo en las reinscripciones, donde se describen cada una de las entidades mediante sus atributos y las relaciones entre estas. Los cuales están caracterizados en la estructura lógica de la base de datos ARQESCOLAR que se representa gráficamente mediante el diagrama entidad-relación de la figura 3.6.

### 3.6.1 Diagrama Entidad – Relación

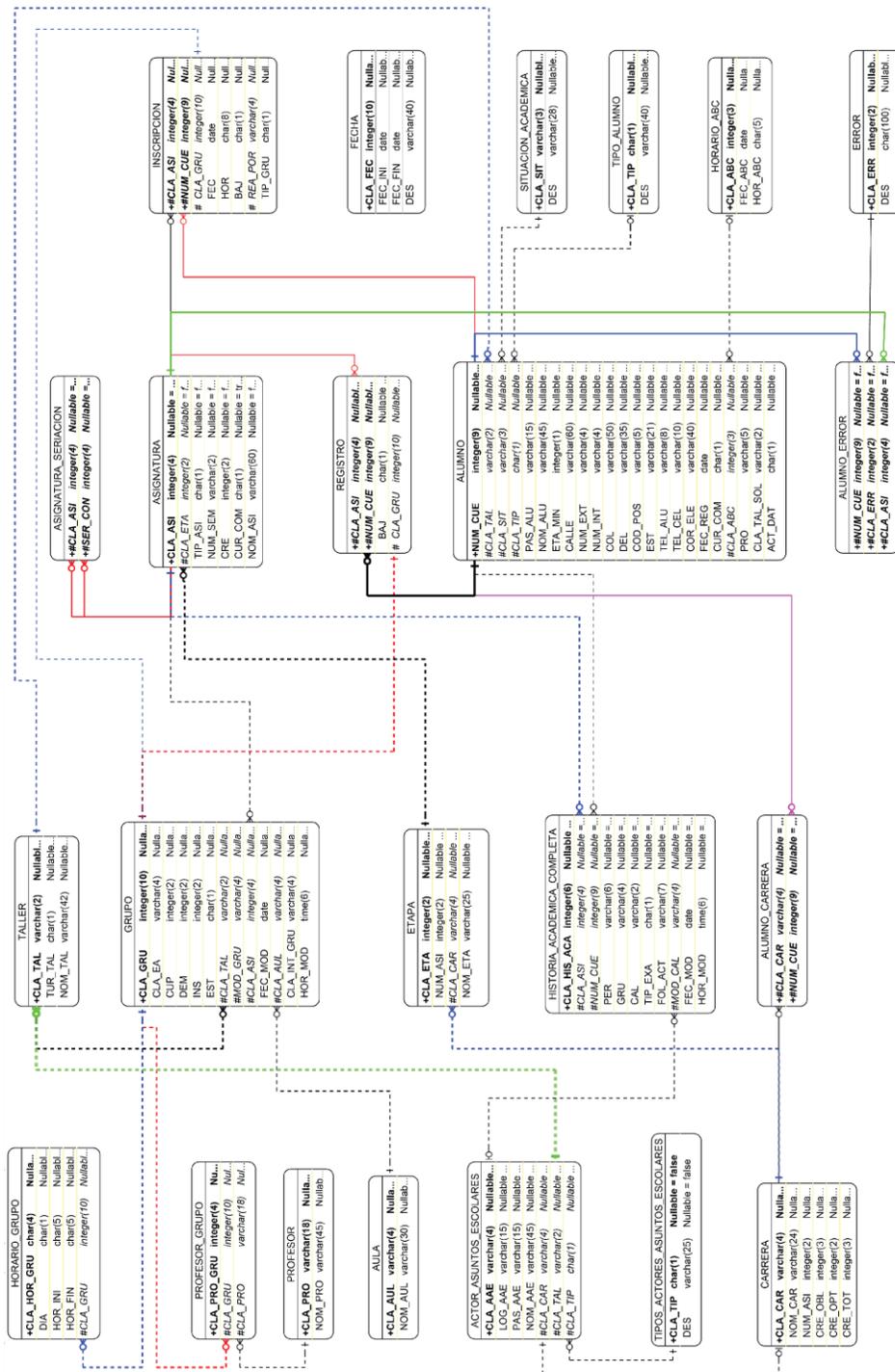


Figura 3. 6 Diagrama Entidad -Relación Base de datos ARQESCOLAR

### 3.7 Diccionario de datos

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_hor_gru	integer		primaria
cla_gru	integer		
dia	character	1	
hor_ini	character	5	
hor_fin	character	5	

Tabla 3. 1 Diccionario de datos correspondiente a la tabla horarios\_grupos

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_pro_gru	integer		Primaria
cla_pro	character varying	18	Foránea
cla_gru	integer		Foránea

Tabla 3. 2 Diccionario de datos correspondiente a la tabla profesores\_grupos

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_pro	character varying	18	primaria
nom_pro	character varying	45	

Tabla 3. 3 Diccionario de datos correspondiente a la tabla profesores

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_aul	character varying	4	primaria
nom_aul	character varying	30	

Tabla 3. 4 Diccionario de datos correspondiente a la tabla aulas

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_aae	character varying	4	primaria
log_aae	character varying	15	
pas_aae	character varying	15	
nom_aae	character varying	45	
cla_car	character varying	4	foránea
cla_tal	character varying	2	foránea
cla_tip	character	1	foránea

Tabla 3. 5 Diccionario de datos correspondiente a la tabla actores\_asuntos\_escolares

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_tip	character	1	primaria
des	character varying	25	

Tabla 3. 6 Diccionario de datos correspondiente a la tabla tipos\_actores\_asuntos\_escolares

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_car	character varying	4	primaria
nom_car	character varying	24	
num_asi	integer		
cre_obl	integer		
cre_opt	integer		
cre_tot	integer		

Tabla 3. 7 Diccionario de datos correspondiente a la tabla carreras

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_tal	character varying	2	primaria
nom_tal	character varying	42	
tur_tal	character	1	

Tabla 3. 8 Diccionario de datos correspondiente a la tabla talleres

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_gru	integer		primaria
cla_asi	integer		
cla_tal	character varying	2	
cla_aul	character varying	4	
cla_int_gru	character varying	4	
cla_tal	character varying	2	
cla_ea	character varying	4	
cup	integer		
dem	integer		
ins	integer		
mod_gru	character varying	4	
est	character	1	
fec_mod	date		
hor_mod	time without time zone		

Tabla 3. 9 Diccionario de datos correspondiente a la tabla grupos

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_eta	integer		primaria
cla_car	character varying	4	foránea
num_asi	integer		
nom_eta	character varying	25	

Tabla 3. 10 Diccionario de datos correspondiente a la tabla etapas

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_asi	integer		primaria
ser_con	integer		primaria

Tabla 3. 11 Diccionario de datos correspondiente a la tabla asignaturas\_seriasiones

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_asi	integer		primaria
cla_eta	integer		
tip_asi	character	1	
num_sem	character varying	2	
cre	integer		
nom_asi	character varying	60	
cur_com	character	1	

Tabla 3. 12 Diccionario de datos correspondiente a la tabla asignaturas

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_his_aca	integer		primaria
cla_asi	integer		Foránea
num_cue	integer	2	Foránea
per	character varying	6	
gru	character varying	4	
cal	character varying	2	
tip_exa	character	1	
fol_act	character varying	7	
fec_mod	date		
hor_mod	time without time zone		
mod_cal	character varying	4	

Tabla 3. 13 Diccionario de datos correspondiente a la tabla historias\_academicas\_completas

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_car	character varying	4	Primaria
num_cue	integer		Primaria, foránea

Tabla 3. 14 Diccionario de datos correspondiente a la tabla alumnos\_carreras

Nombre	Tipo de dato	Longitud	Tipo de llave
num_cue	integer		Primaria, Foránea
cla_err	integer		Primaria, Foránea
cla_así	integer		Primaria, Foránea

Tabla 3. 15 Diccionario de datos correspondiente a la tabla alumnos\_errores

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_así	integer		Primaria, foránea
num_cue	integer		Primaria, foránea
cla_gru	integer		Foránea
baj	character	1	

Tabla 3. 16 Diccionario de datos correspondiente a la tabla registros

Nombre	Tipo de dato	Longitud	Tipo de llave
num_cue	integer		Primaria, foránea
cla_así	integer		Primaria, foránea
cla_gru	integer		Foránea
rea_por	character	4	
fec	date		
hor	character	8	
baj	character	1	
tip_gru	character	1	

Tabla 3. 17 Diccionario de datos correspondiente a la tabla inscripciones

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_abc	integer		Primaria
fec_abc	date		
hor_abc	char	5	

Tabla 3. 18 Diccionario de datos correspondiente a la tabla horarios\_abc

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_sit	character varying	3	Primaria
des	character varying	28	

Tabla 3. 19 Diccionario de datos correspondiente a la tabla situaciones\_academicas

Nombre	Tipo de dato	Longitud	Tipo de llave
num_cue	integer		Primaria
cla_tal	character varying	2	Foránea
cla_sit	character varying	3	Foránea
cla_tip	character	1	Foránea
pas_alu	character varying	15	
nom_alu	character varying	45	
eta_min	integer		
calle	character varying	60	
num_ext	character varying	4	
num_int	character varying	4	
col	character varying	50	
del	character varying	35	
cod_pos	character varying	5	
est	character varying	21	
tel_alu	character varying	8	
tel_cel	character varying	10	
cor_ele	character varying	40	
fec_reg	date		
cla_abc	integer		
cur_com	character	1	
cla_tal_sol	character varying	2	
act_dat	character	1	

Tabla 3. 20 Diccionario de datos correspondiente a la tabla alumnos

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_err	integer		Primaria
des	character varying	100	

Tabla 3. 21 Diccionario de datos correspondiente a la tabla errores

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_tip	character	1	Primaria
des	character varying	40	

Tabla 3. 22 Diccionario de datos correspondiente a la tabla tipos\_alumnos

Nombre	Tipo de dato	Longitud	Tipo de llave
cla_fec	integer		Primaria
fec_ini	date		
fec_fin	date		
des	character varying	40	

Tabla 3. 23 Diccionario de datos correspondiente a la tabla fechas

### 3.8 Diagramas de Modelado de Comportamiento UML

Los diagramas de comportamiento capturan las variedades de interacción: visualización, especificación, construcción y documentación de los aspectos del sistema, además del estado instantáneo dentro de un modelo mientras se ejecuta a través del tiempo.

### 3.9 Diagramas de casos de uso

Los diagramas de casos de uso describen la interacción entre los actores y el Sistema de reinscripciones.

#### 3.9.1 Alumno

Diagrama de caso de uso correspondiente al usuario Alumno:

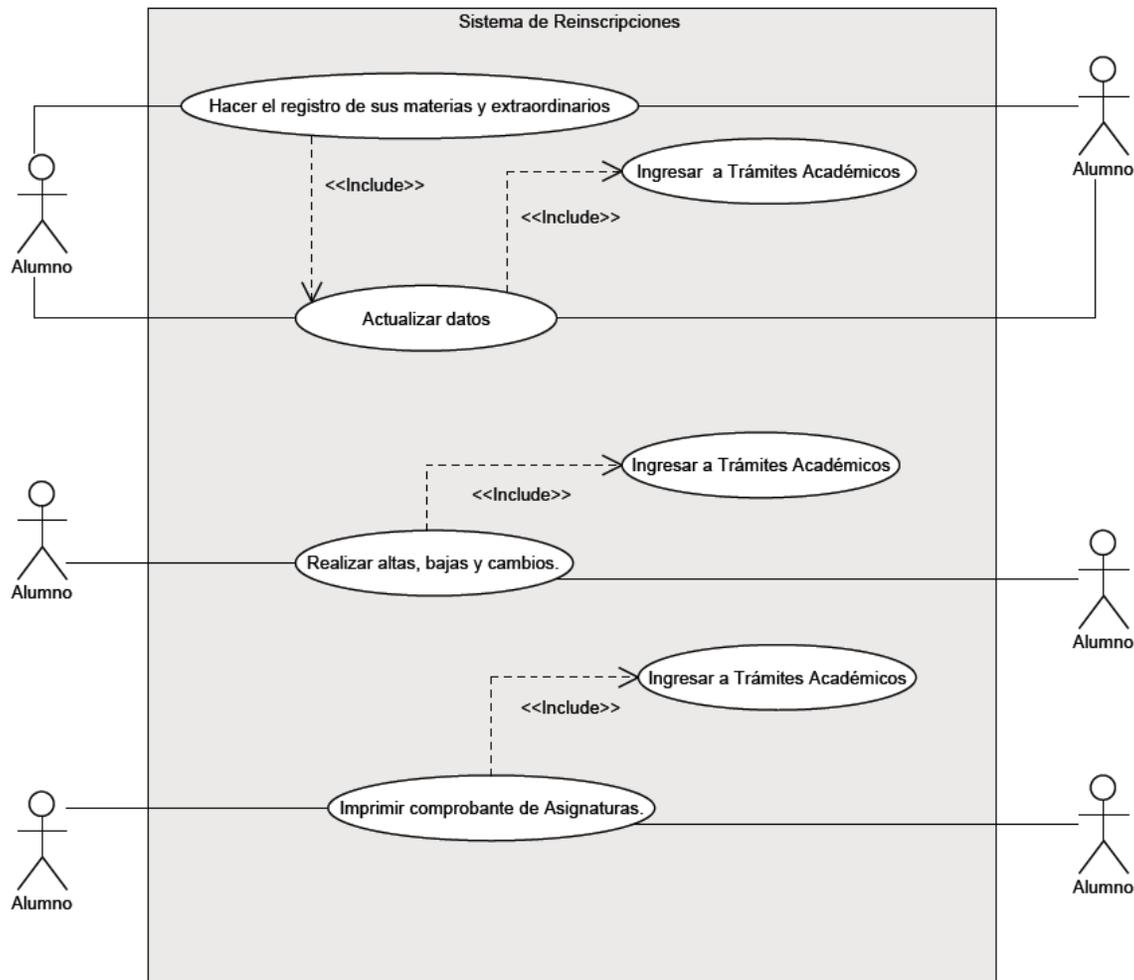


Figura 3. 7 Diagrama de caso de uso, usuario Alumno

### 3.9.2 Mantenimiento

Diagrama de caso de uso correspondiente al usuario Mantenimiento:

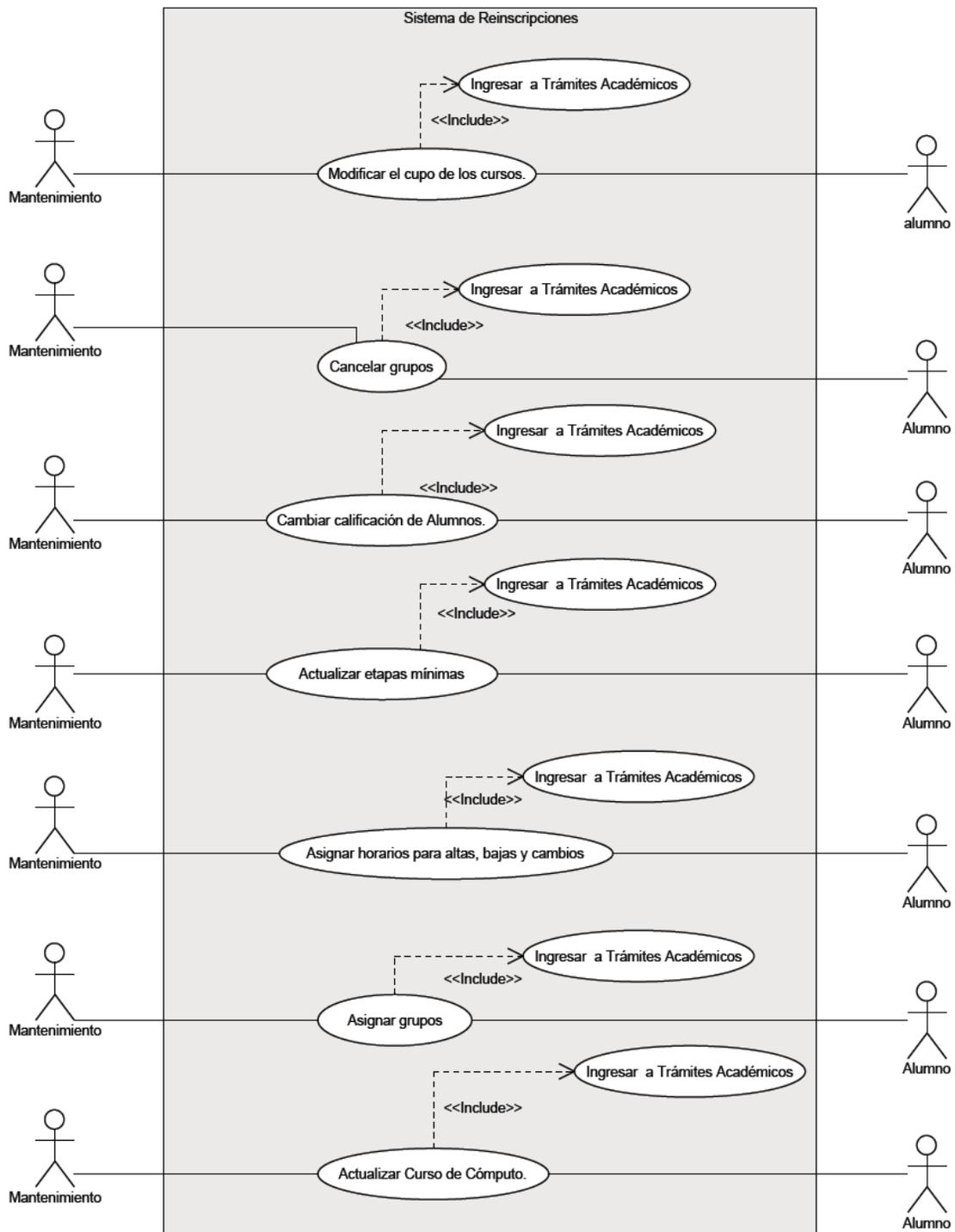


Figura 3. 8 Diagrama de caso de uso, usuario Mantenimiento

### 3.9.3 Coordinador de Taller

Diagrama de caso de uso correspondiente al usuario Coordinador de Taller:

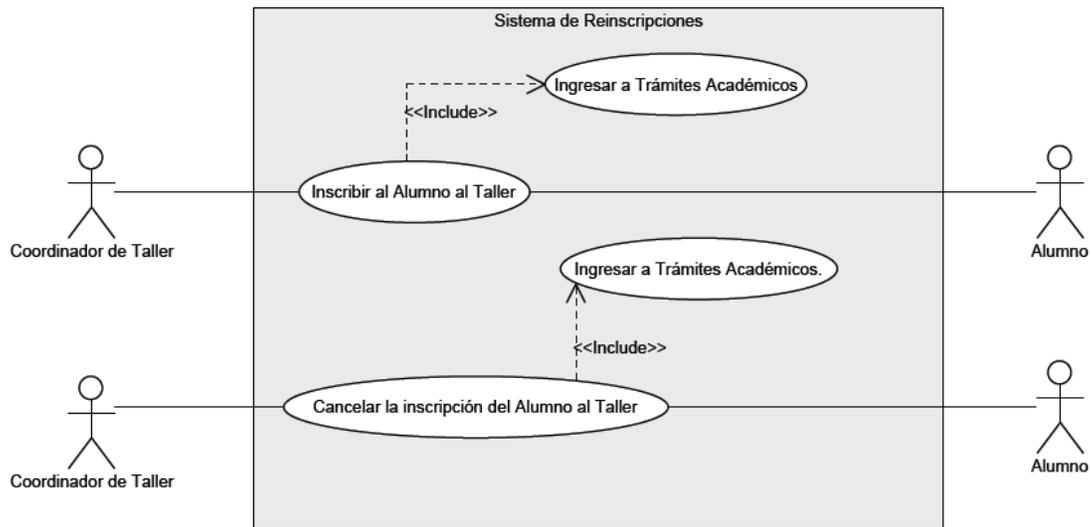


Figura 3. 9 Diagrama de caso de uso, usuario Coordinador de Taller

### 3.9.4 Coordinador de Carrera

Diagrama de caso de uso correspondiente al usuario Coordinador de Carrera:

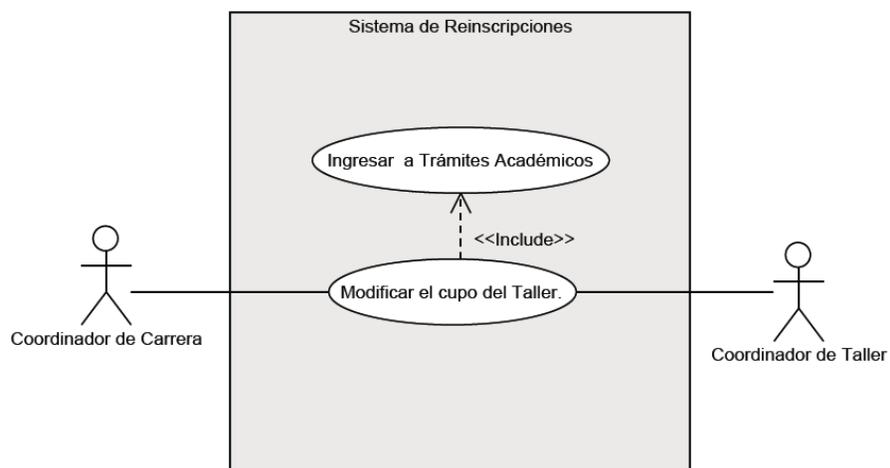


Figura 3. 10 Diagrama de caso de uso, usuario Coordinador de Carrera

### 3.9.5 Jefe de Asuntos Escolares

Diagrama de caso de uso correspondiente al usuario Jefe de Asuntos Escolares:

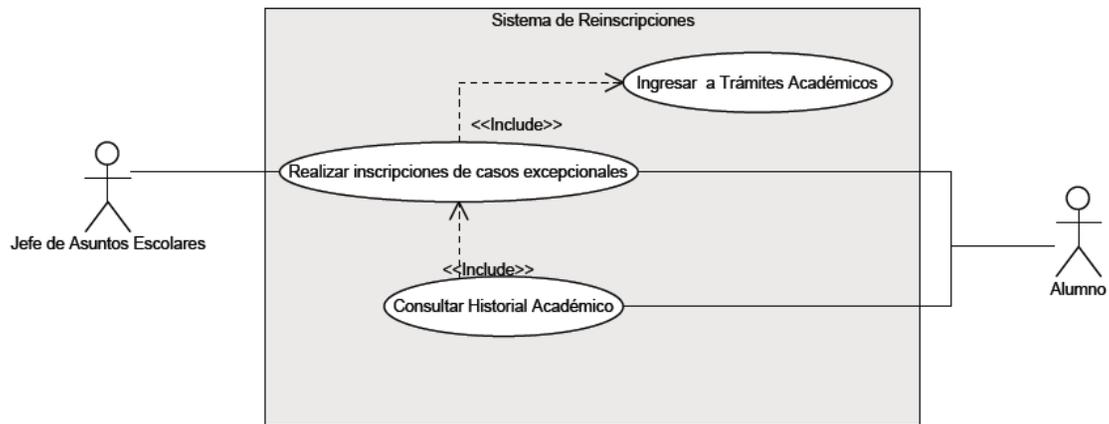


Figura 3. 11 Diagrama de caso de uso, Jefe de Asuntos Escolares

### 3.10 Diagramas de Actividades

En los diagramas de actividades se muestra el flujo de trabajo desde el punto de inicio hasta el punto final indicando las rutas de decisiones que existen en el progreso de sucesos contenidos en la actividad.

#### 3.10.1 Ingreso a Trámites Escolares

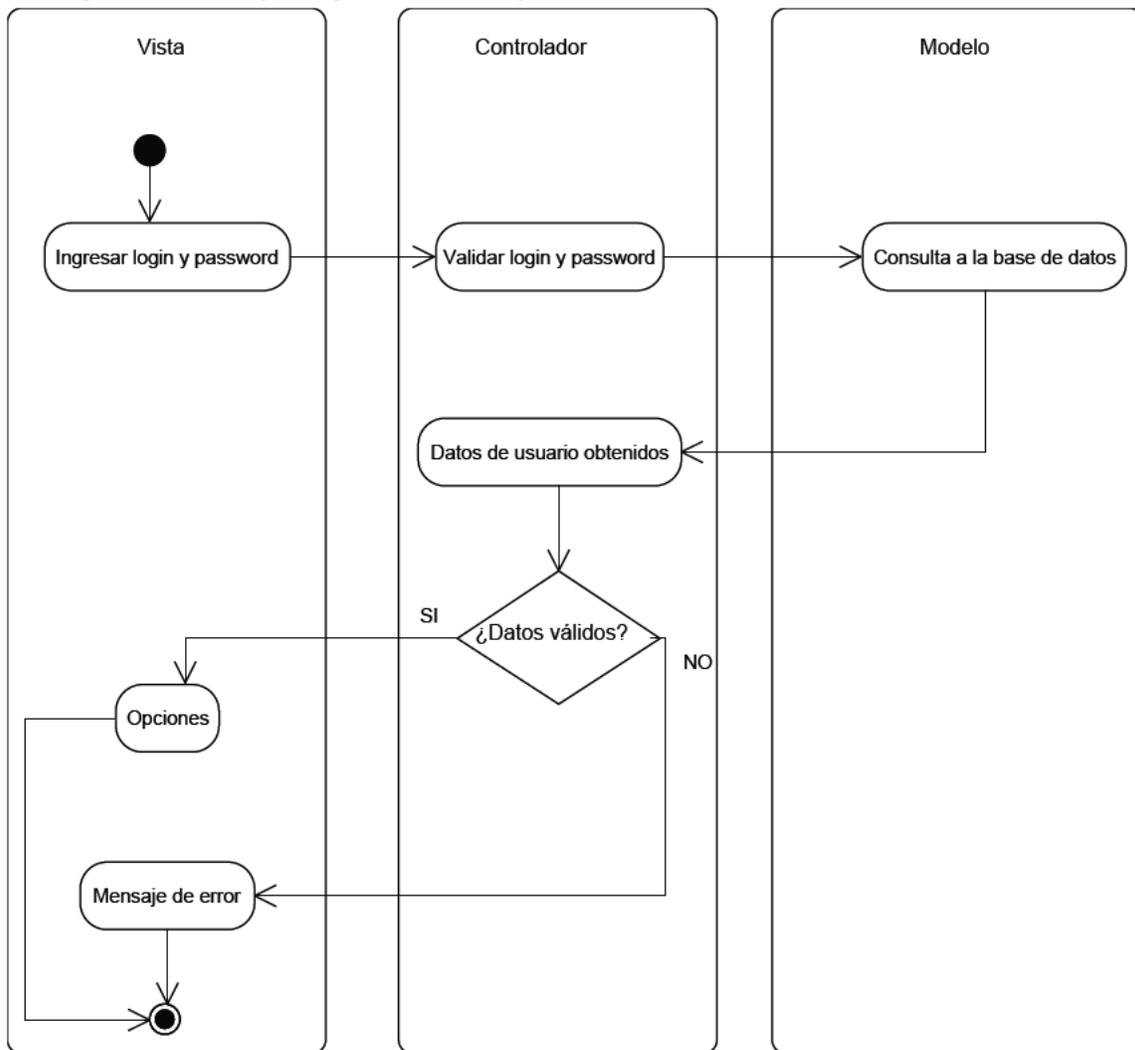


Figura 3. 12 Diagrama de actividades Ingreso a Trámites Escolares

### 3.10.2 Alumno

**Registro** Ver Diagrama de Actividades Anexo C, Figura C.1

**Altas, Bajas y Cambios** Ver Diagrama de Actividades Anexo C, Figura C.2

#### Alumno Actualiza datos

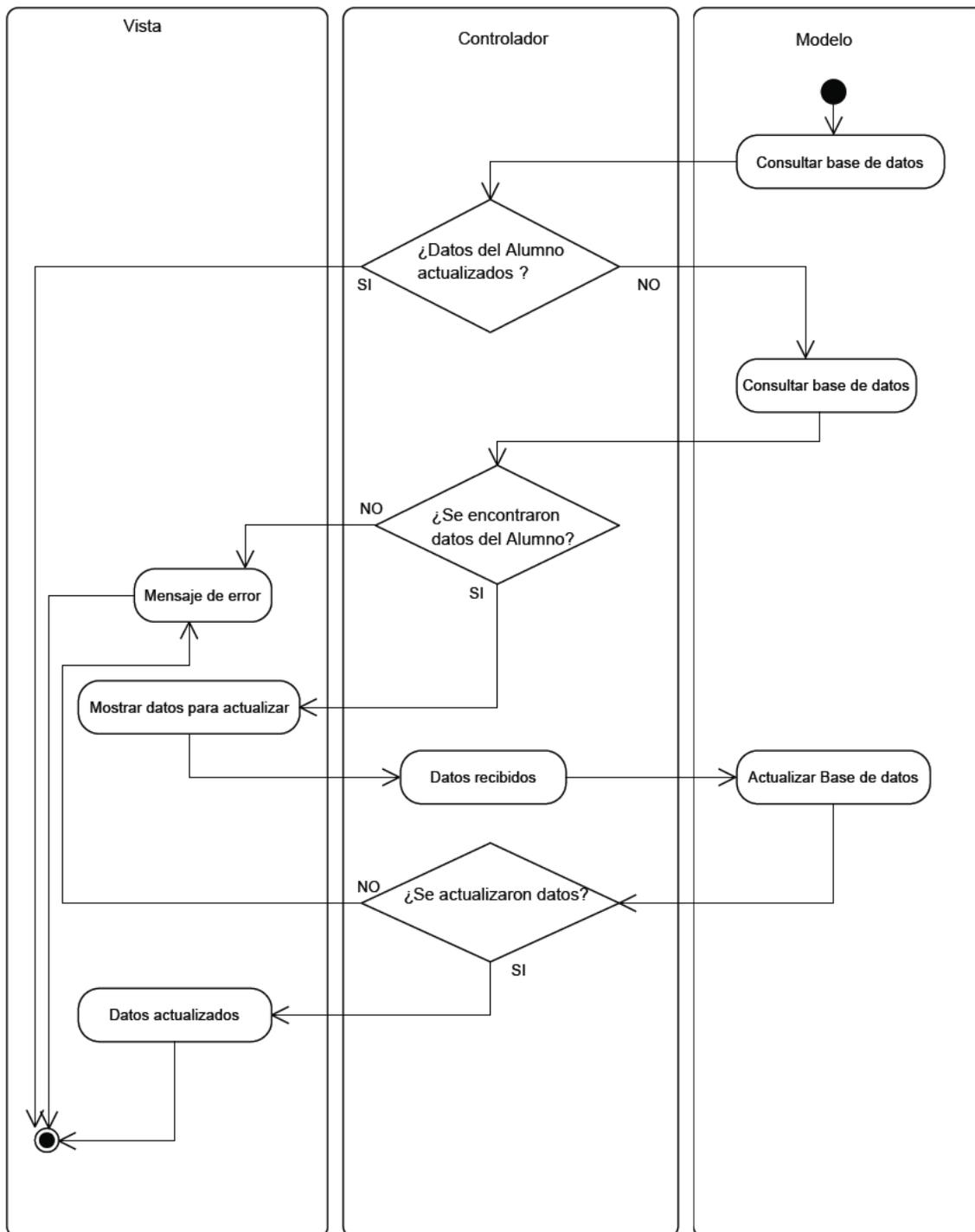


Figura 3. 13 Diagrama de actividades Alumno Actualiza datos

### **3.10.3 Mantenimiento**

**Cambiar calificación del Alumno** Ver Diagrama de Actividades Anexo C, Figura C.4

**Cancelar grupos de cursos** Ver Diagrama de Actividades Anexo C, Figura C.5

**Modificar Cupo de cursos** Ver Diagrama de Actividades Anexo C, Figura C.6

### 3.10.4 Coordinador de Taller

#### Cancelar inscripción al Taller

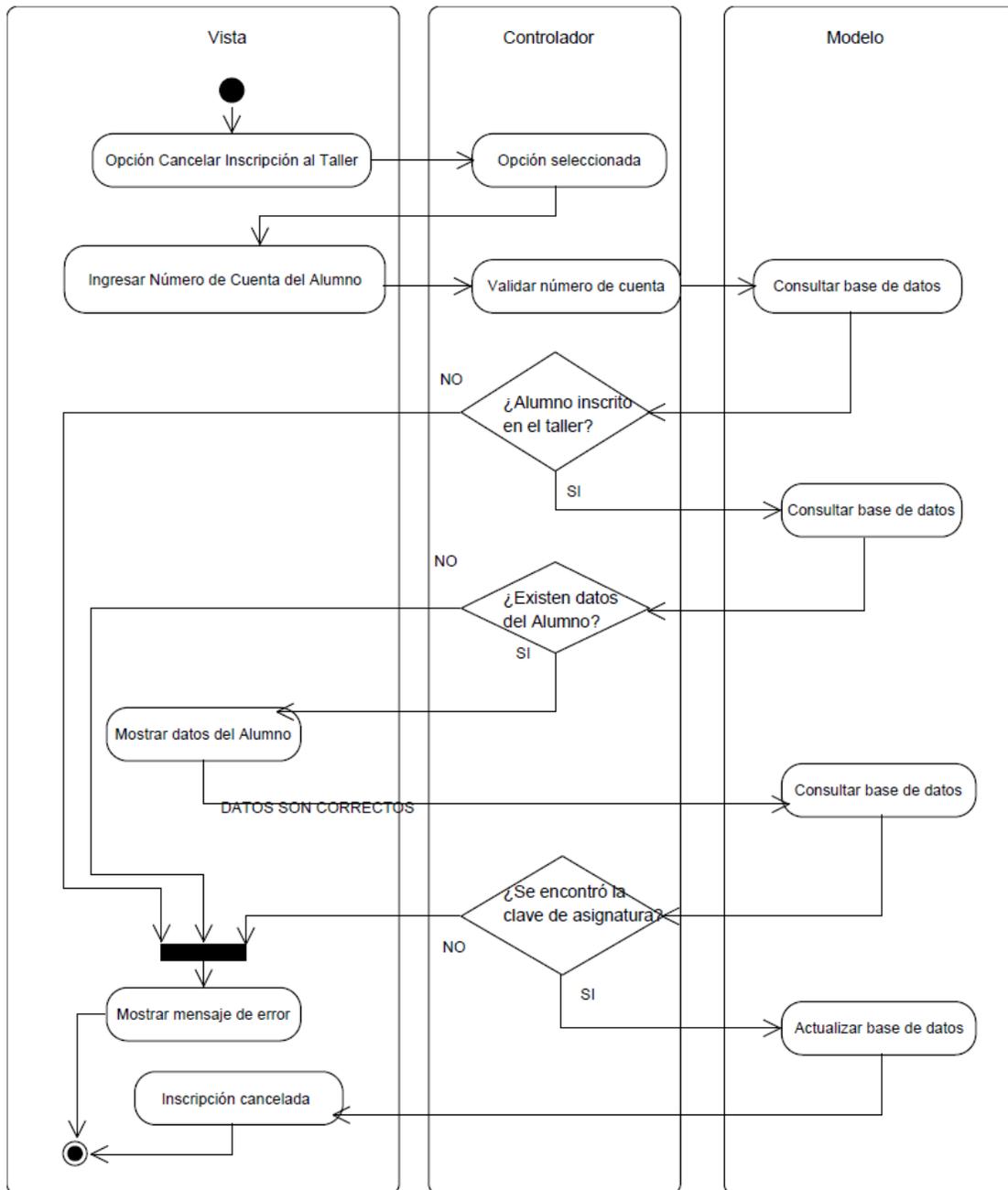


Figura 3. 14 Diagrama de actividades Cancelar inscripción al Taller

### **3.10.5 Coordinador de Carrera**

**Actualizar cupo de Taller** Ver Diagrama de Actividades Anexo C, Figura C.8

### **3.10.6 Jefe de Asuntos Escolares**

**Inscribir Alumnos a cursos** Ver Diagrama de Actividades Anexo C, Figura C.9

## Consulta historial académico del Alumno

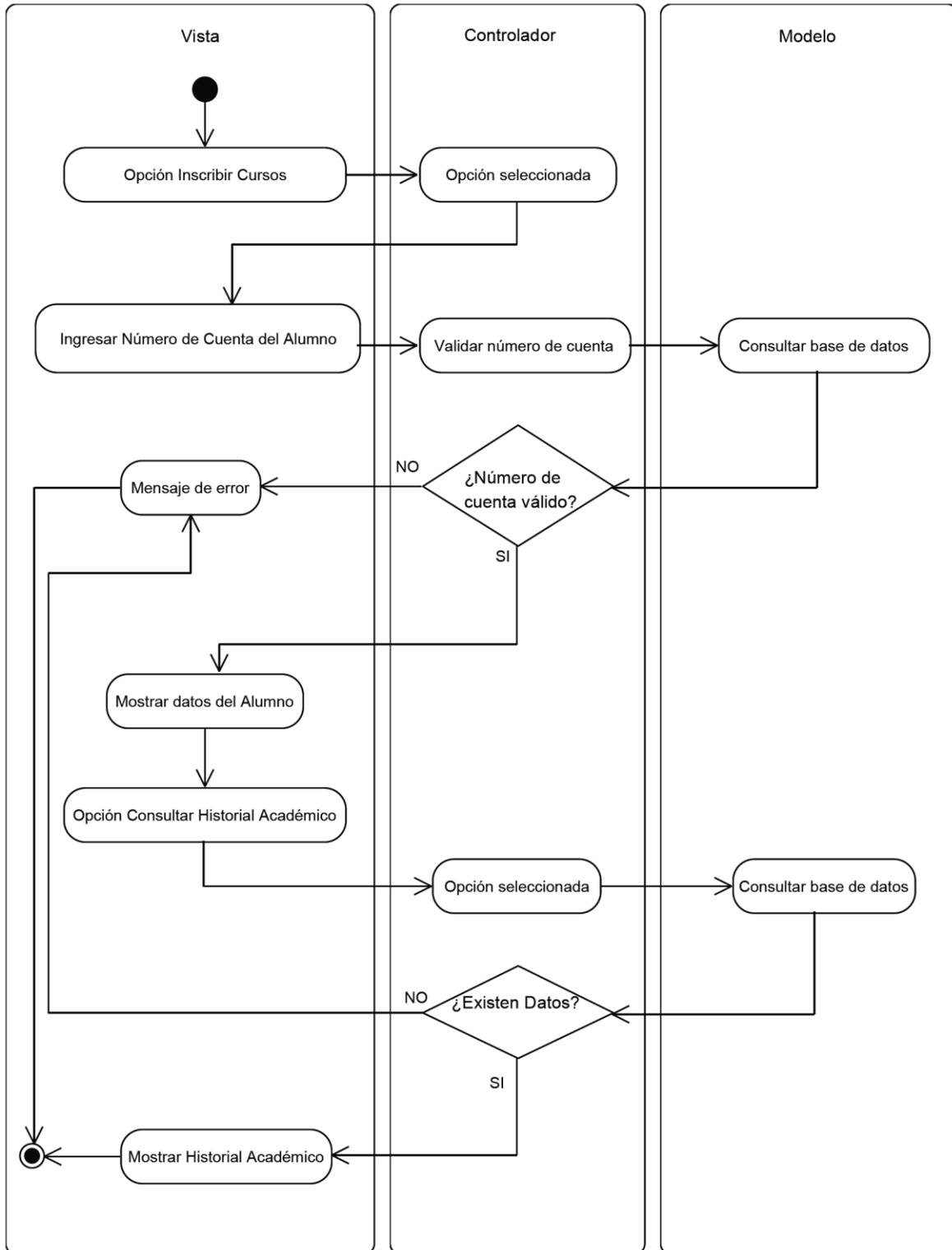


Figura 3. 15 Diagrama de actividades Ingreso Consultar Historial Académico

### 3.11 Diagramas de Secuencia

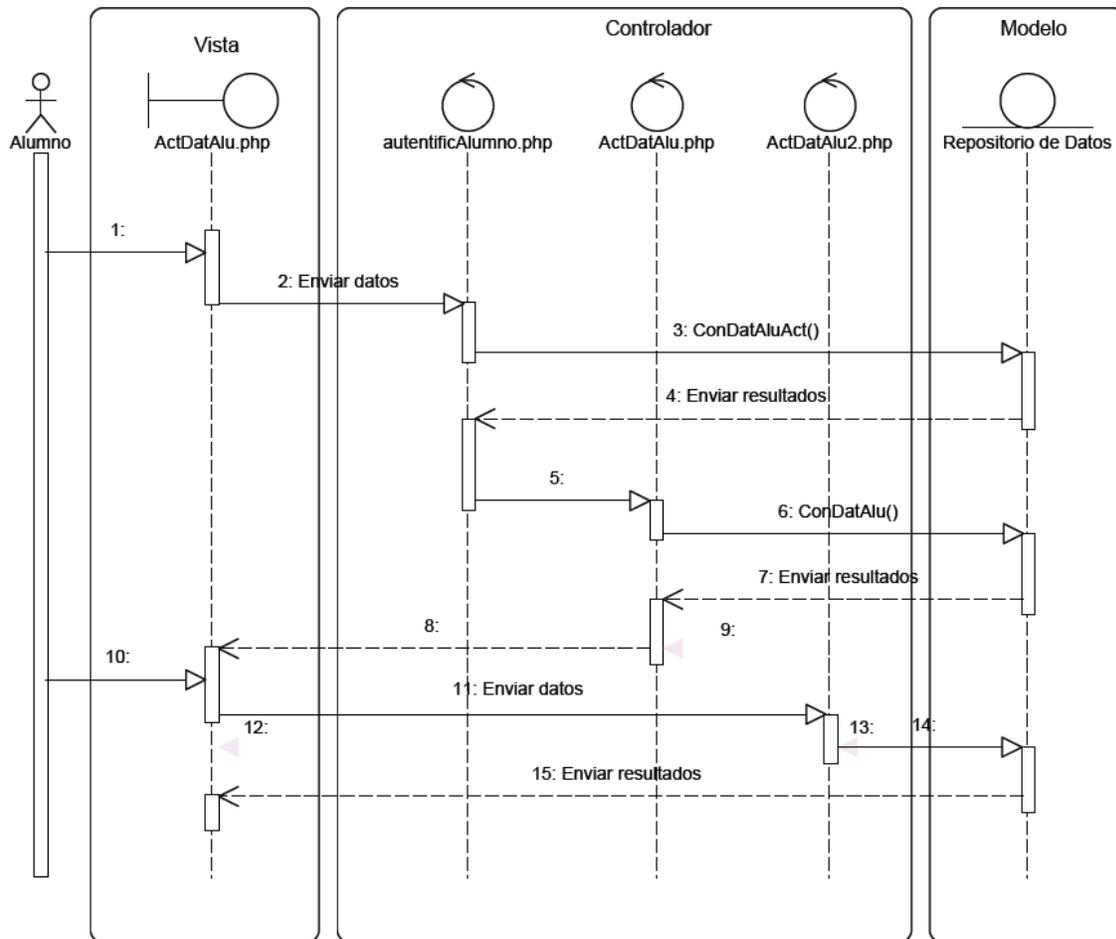
En los diagramas de secuencias se presenta el conjunto de componentes de los módulos del sistema así como los mensajes enviados y recibidos por ellos.

#### 3.11.1 Alumno

**Registro de asignaturas** Ver Diagrama de Secuencia Anexo B, Figura B.1

**Altas, bajas y cambios** Ver Diagrama de Secuencia Anexo B, Figura B.2

**Actualizar datos**

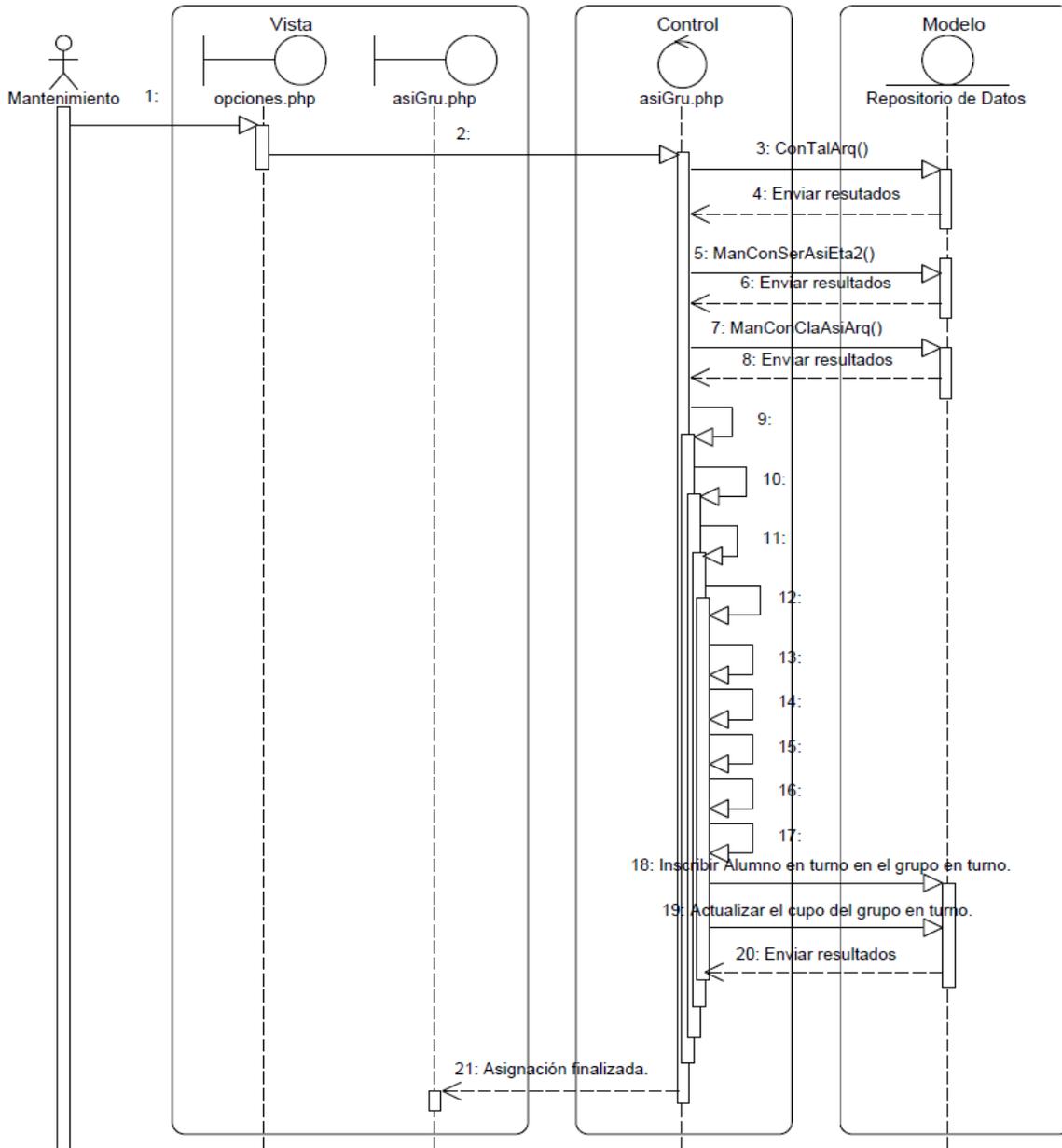


- Del escenario Ingresar a Trámites Académicos.
1. Ingresar login y password.
  3. Consultar que el Alumno haya actualizado sus datos.
  6. Consultar información personal del Alumno.
  8. Mostrar datos personales.
  10. Alumno actualiza sus datos.
  14. Actualizar Datos.
  15. Información actualizada.

Figura 3. 16 Diagrama de secuencia Actualizar Datos

### 3.11.2 Mantenimiento

#### Asignación de grupos



Del escenario Ingresar a Trámites Académicos.

1: Seleccionar Asignación de grupos.

9: Mientras haya Talleres.

10: Mientras haya asignaturas.

11: Mientras haya grupos compatibles al taller en turno.

12: Mientras haya peticiones al grupo en turno.

13: Asignatura en turno es Práctica Profesional Supervisada.

14: Revisar que Alumno cumpla requisitos para Práctica Profesional Supervisada.

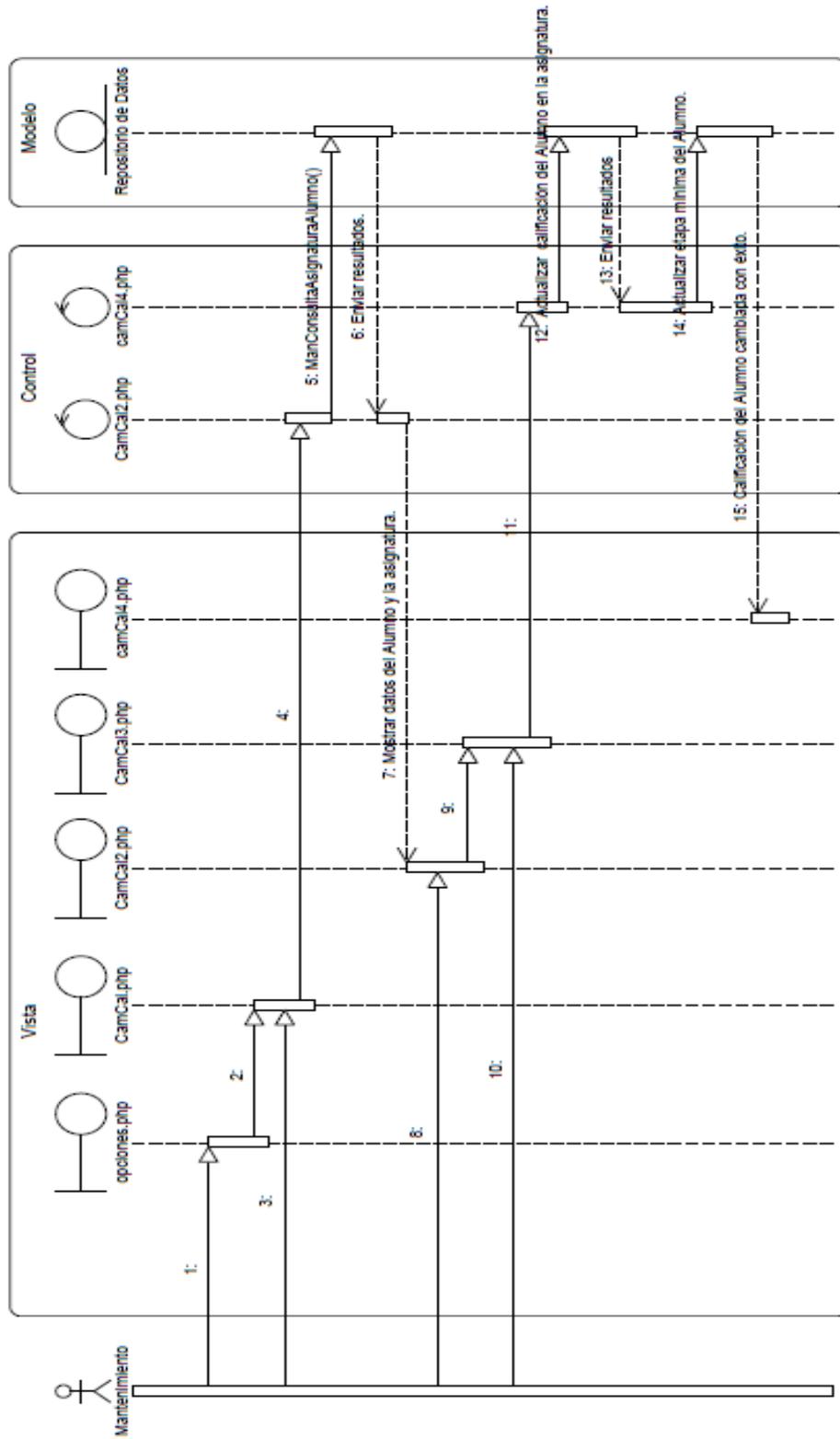
15: Revisar tipo de Alumno.

16: Revisar que Alumno sea del Taller en turno.

17: Alumno es del Taller en turno.

Figura 3. 17 Diagrama de secuencia Asignación de grupos

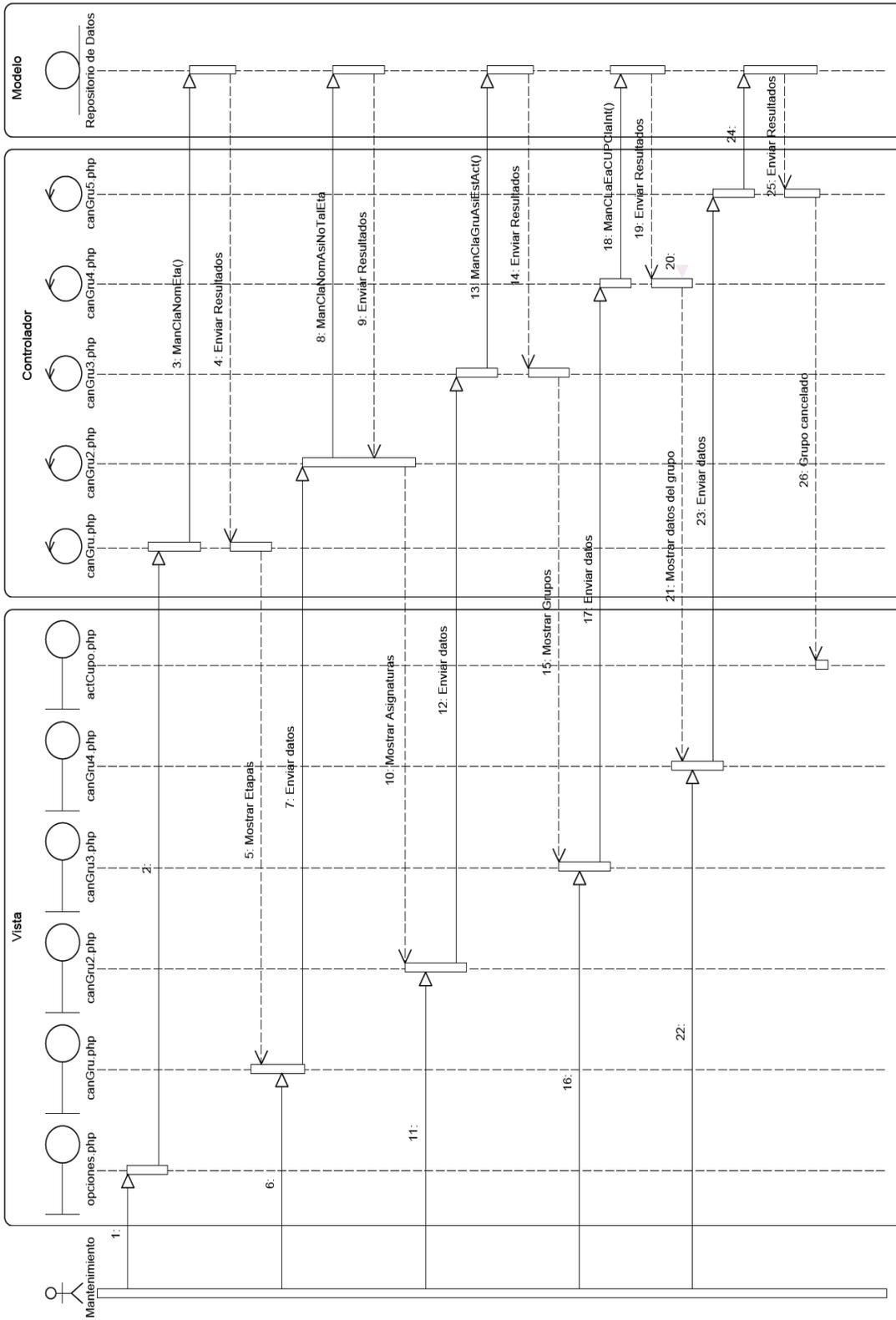
## Cambiar calificación de un Alumno



- Del escenario Ingresar a Trámites Académicos.
- 1: Seleccionar Cambiar calificación de un Alumno.
  - 3: Proporcionar número de cuenta y clave de asignatura.
  - 8: Seleccionar datos correctos.
  - 10: Proporcionar nueva calificación.

Figura 3. 18 Diagrama de secuencia Cambiar calificación de un alumno

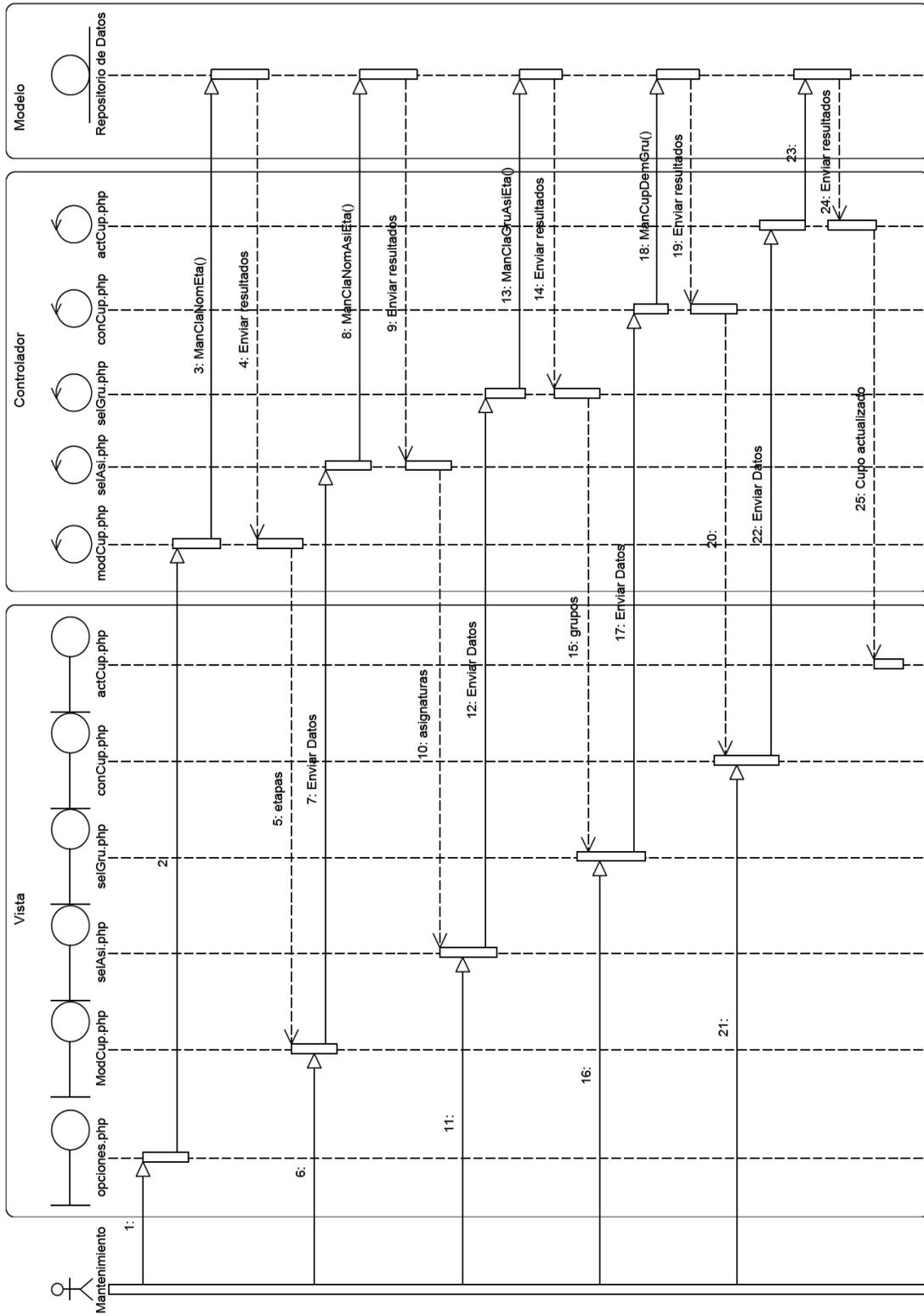
# Cancelar grupo de cursos



- Del escenario Ingresar a Trámites Académicos.
1. Seleccionar Cancelar grupos de cursos.
  6. Elegir una Etapa.
  11. Seleccionar Asignatura.
  16. Seleccionar grupo.
  22. Seleccionar Cancelar grupo.
  24. Actualizar bandera (baja) del grupo.
  26. El grupo ha sido cancelado.

Figura 3. 19 Diagrama de secuencia Cancelar grupo de cursos

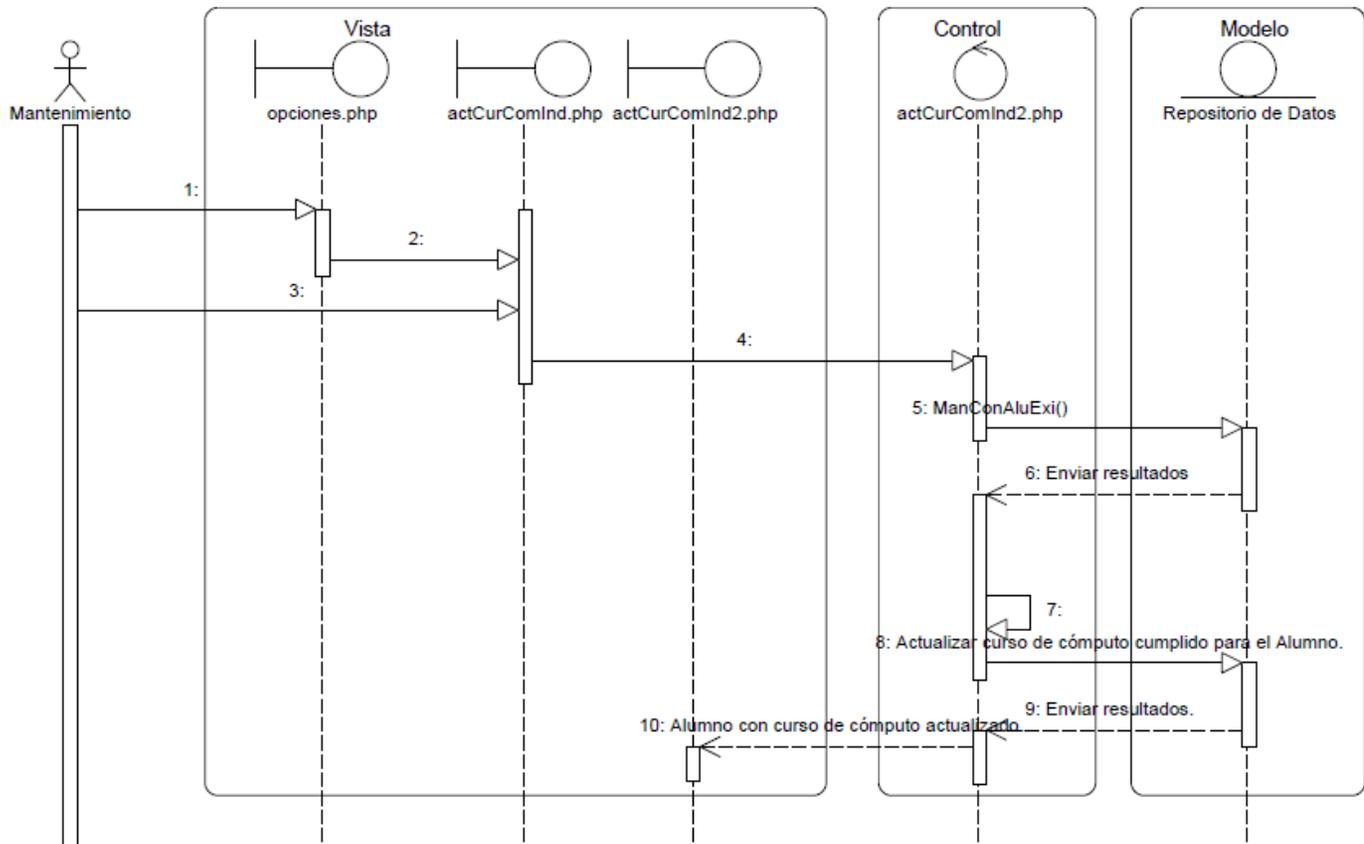
# Modificar cupo de cursos



- Del escenario Ingresar a Trámites Académicos.
- 1 Selección Modificar cupo de cursos.
  - 6. Elegir una Etapa.
  - 11. Selección Asignatura.
  - 16. Selección grupo.
  - 21. Proporcionar nuevo cupo.
  - 23. Actualizar cupo del grupo.
  - 25. Cupo del grupo actualizado.

Figura 3. 20 Diagrama de secuencia Modificar cupo de cursos

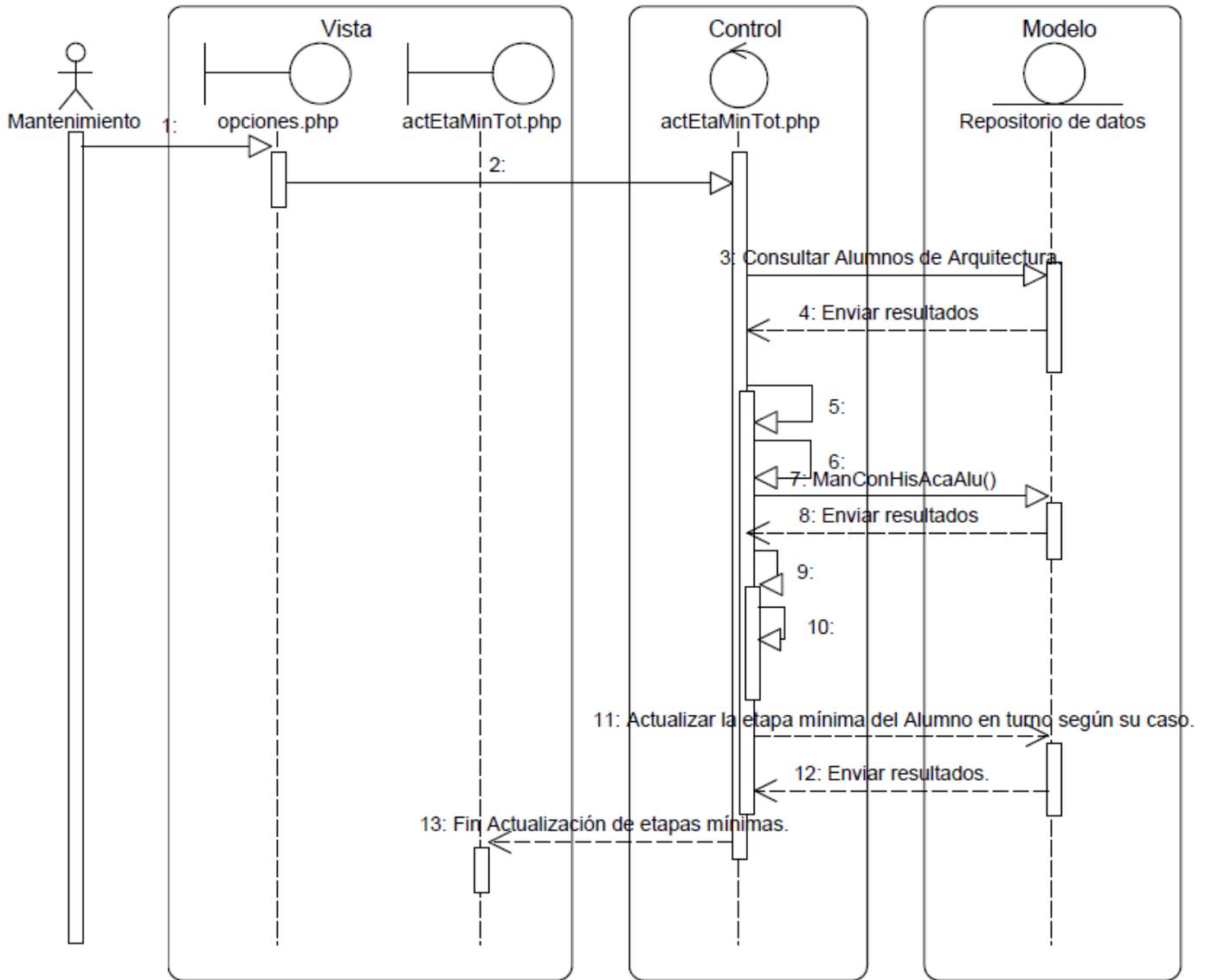
## Actualizar Curso de Cómputo



Del escenario Ingresar a Trámites Académicos.  
 1: Seleccionar Actualización de curso de cómputo  
 3: Ingresar número de cuenta.

Figura 3. 21 Diagrama de secuencia Actualizar Curso de Cómputo

## Actualización de etapas mínimas



Del escenario ingresar a Trámites Académicos

1: Seleccionar Actualización de etapas mínimas

3: Consultar Alumnos de Arquitectura. ManConClaAsiEtaCar(). ManConNumAsiOblCar()

5: Mientras haya Alumnos.

6: Escoger Alumno en turno.

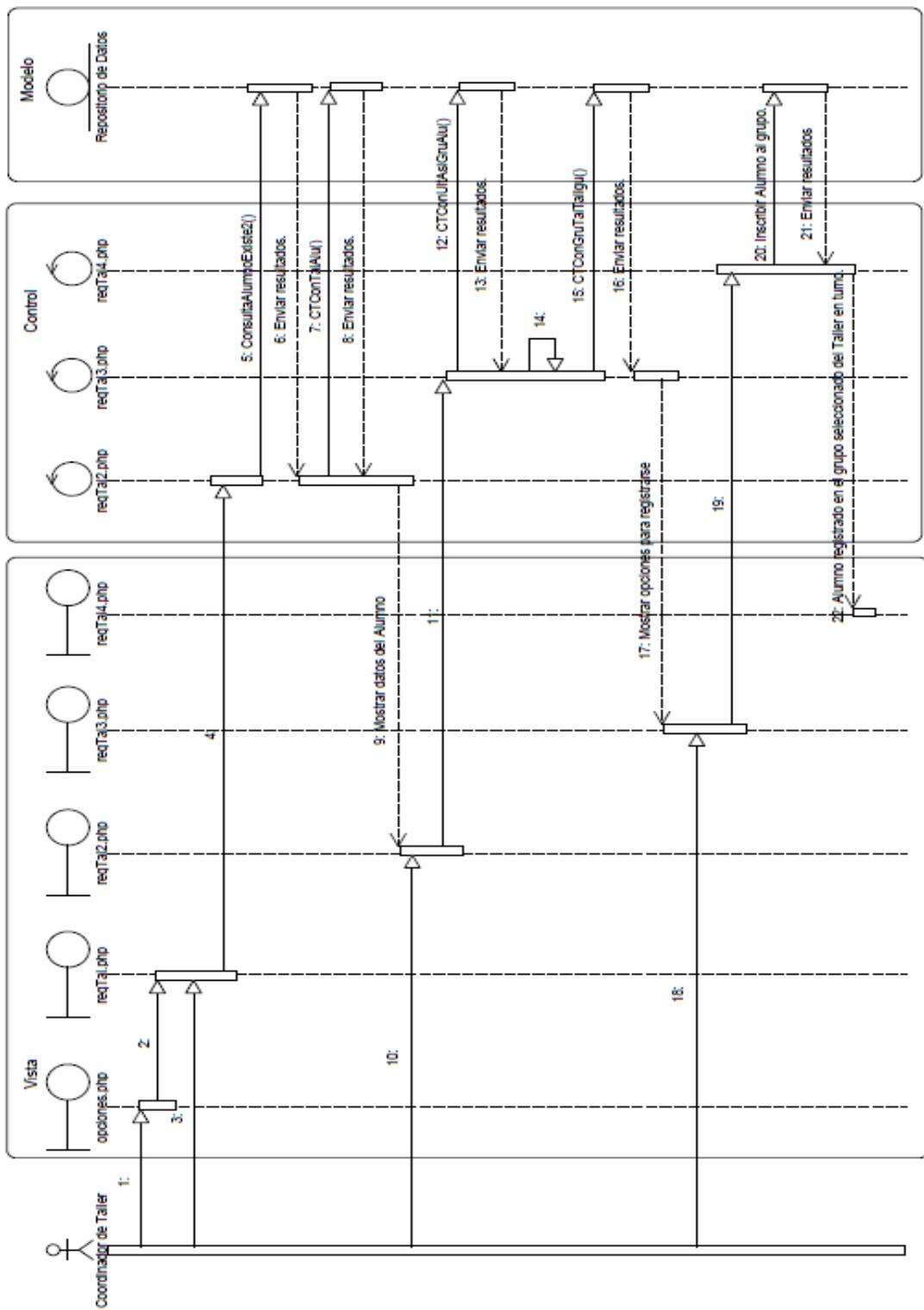
9: Mientras el Alumno en turno tenga asignaturas en su historial académico.

10: Verificar el caso del Alumno en turno.

Figura 3. 22 Diagrama de secuencia Actualización de etapas mínimas

### 3.11.3 Coordinador de Taller

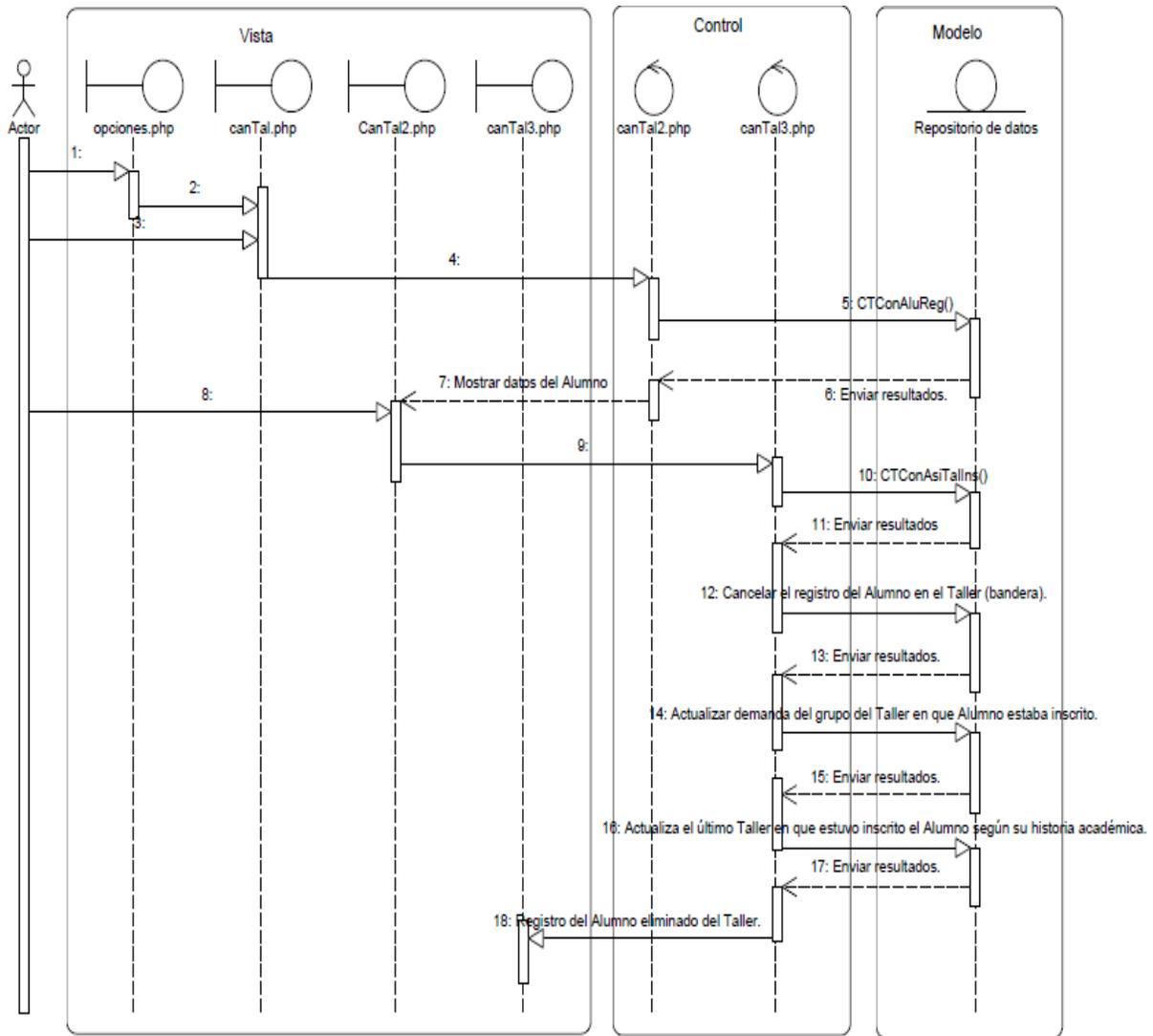
#### Inscribir Alumno a Taller



Del escenario Ingresar a Trámites Académicos  
 1: Seleccionar inscripción a Taller.  
 3: Proporcionar número de cuenta del Alumno.  
 9: Mostrar datos del Alumno.  
 10 Seleccionar Datos son correctos.  
 14: Definir el caso del Alumno.  
 18: Seleccionar Grupo elegido.  
 20: Inscribir Alumno al grupo.  
 Actualizar el cupo del grupo.  
 Actualizar Taller a que el Alumno se registró.

Figura 3. 23 Diagrama de secuencia Inscribir Alumno a Taller

## Cancelar Registro a Taller

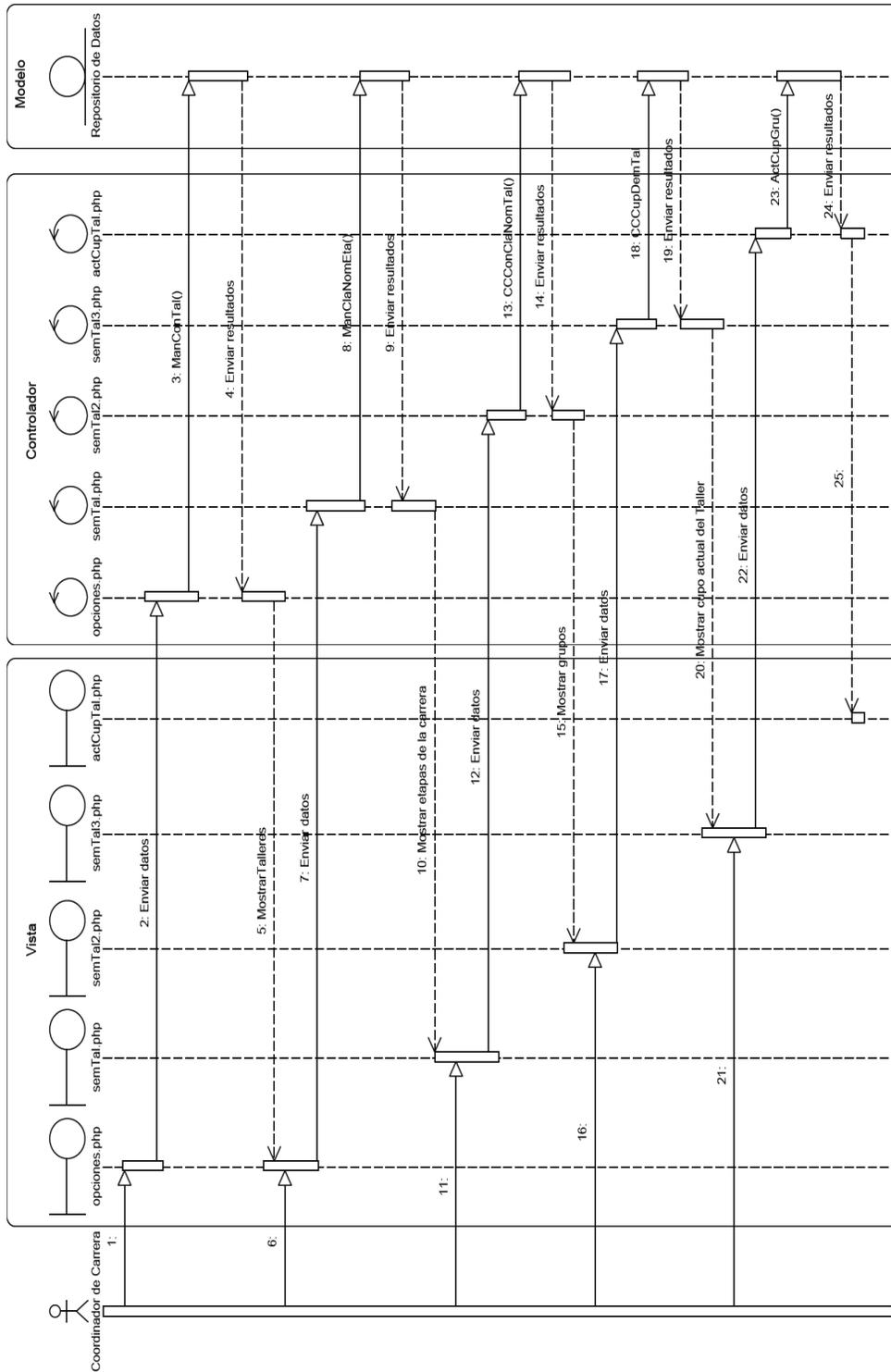


Del escenario Ingresar a Trámites Académicos  
 1: Seleccionar Cancelar inscripción al Taller.  
 3: Proporcionar número de cuenta del Alumno  
 8: Seleccionar Datos correctos.

Figura 3. 24 Diagrama de secuencia Cancelar Registro a Taller

### 3.11.4 Coordinador de Carrera

#### Actualizar Cupo de un Taller

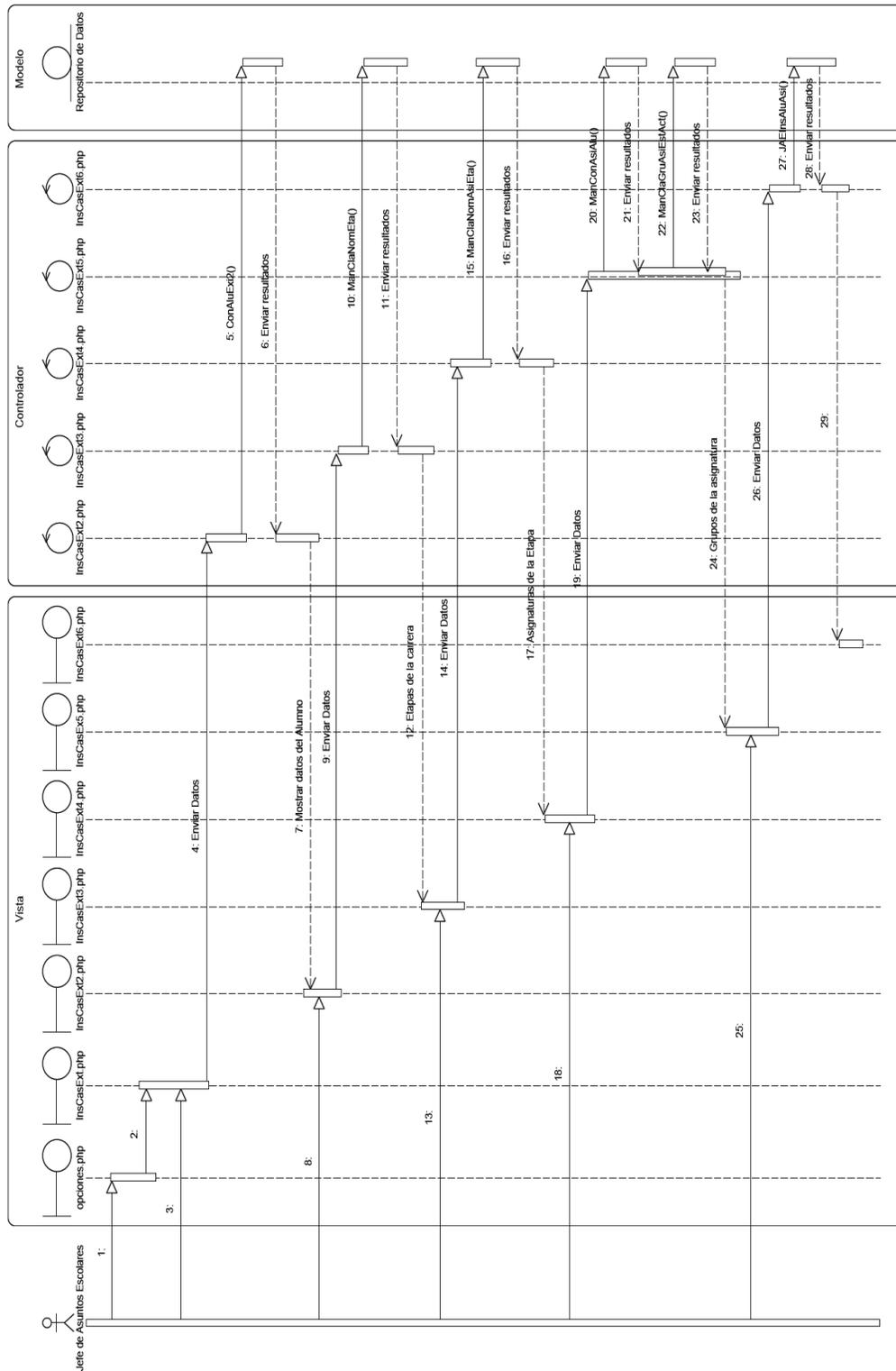


- Del escenario Ingresar a Trámites Académicos.
3. Consultar los Talleres correspondientes a la carrera de Arquitectura.
  6. Seleccionar un Taller.
  11. Seleccionar Etapa.
  16. Elegir grupo.
  21. Proporcionar nuevo cupo.
  23. Actualizar cupo del grupo del Taller seleccionado.
  25. Cupo del Taller elegido actualizado.

Figura 3. 25 Diagrama de secuencia Actualizar Cupo de un Taller

### 3.11.5 Jefe de Asuntos Escolares

#### Inscribir a Alumnos

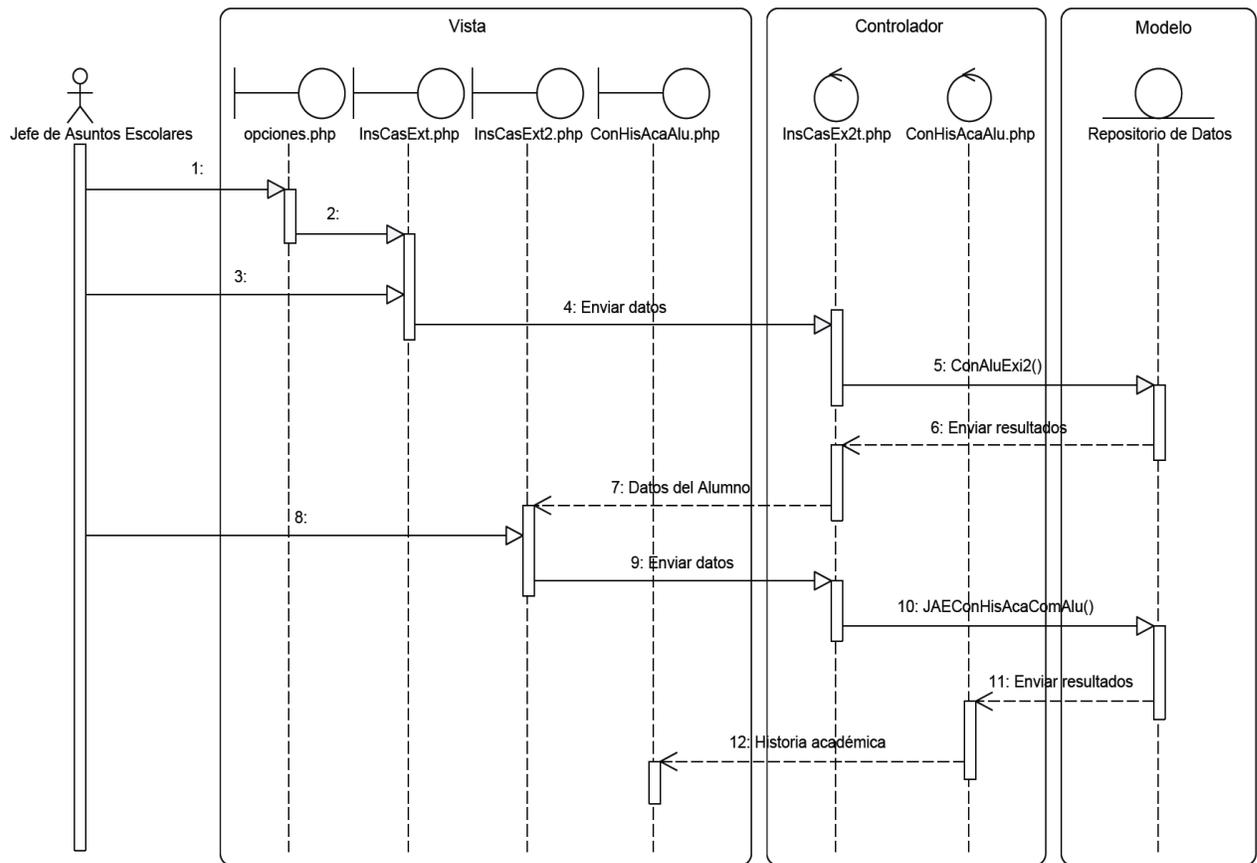


Del escenario Ingresar a Trámites Académicos.

2. Seleccionar Inscribir cursos.
3. Ingresar número de cuenta.
8. Elegir Datos son correctos.
18. Elegir una Etapa.
20. Establecer tipo de grupo (ordinario/extraordinario)
25. Seleccionar Asignatura.
27. Elegir un grupo para inscribir Alumno.
27. Insertar un registro del Alumno al grupo seleccionado.  
Actualizar el cupo del grupo.
29. Alumno inscrito.

Figura 3. 26 Diagrama de secuencia Inscribir a Alumnos

## Consultar Historial Académico



- 2. Seleccionar Inscribir cursos.
- 3. Ingresar número de cuenta.
- 7. Mostrar datos del Alumno.
- 8. Elegir Consultar Historial Académico.
- 12. Mostrar Historia Académica del Alumno.

Figura 3. 27 Diagrama de secuencia Consultar Historial Académico

## 3.12 Vistas

### 3.12.1 Secuencia de pantallas

- **Alumno**

#### Registrar asignaturas

Nombre: index.html

Página anterior: ninguna

Página siguiente: IngresoTramitesEscolaresAlumno.php

Descripción: Página que permite el ingreso de los actores de Asuntos Escolares a sus respectivos formularios de autenticación.

Pantalla:



Figura 3. 28 Pantalla de Ingreso al Sistema

Campo:	Representación
Actores Asuntos Escolares	Link
Alumno	Link

Tabla 3. 24

Nombre: IngresoTramitesEscolaresAlumno.php  
 Página anterior: index.html  
 Página siguiente: registroAlumno.php  
 Descripción: Página para ingresar datos del Alumno para su autenticación.  
 Pantalla:

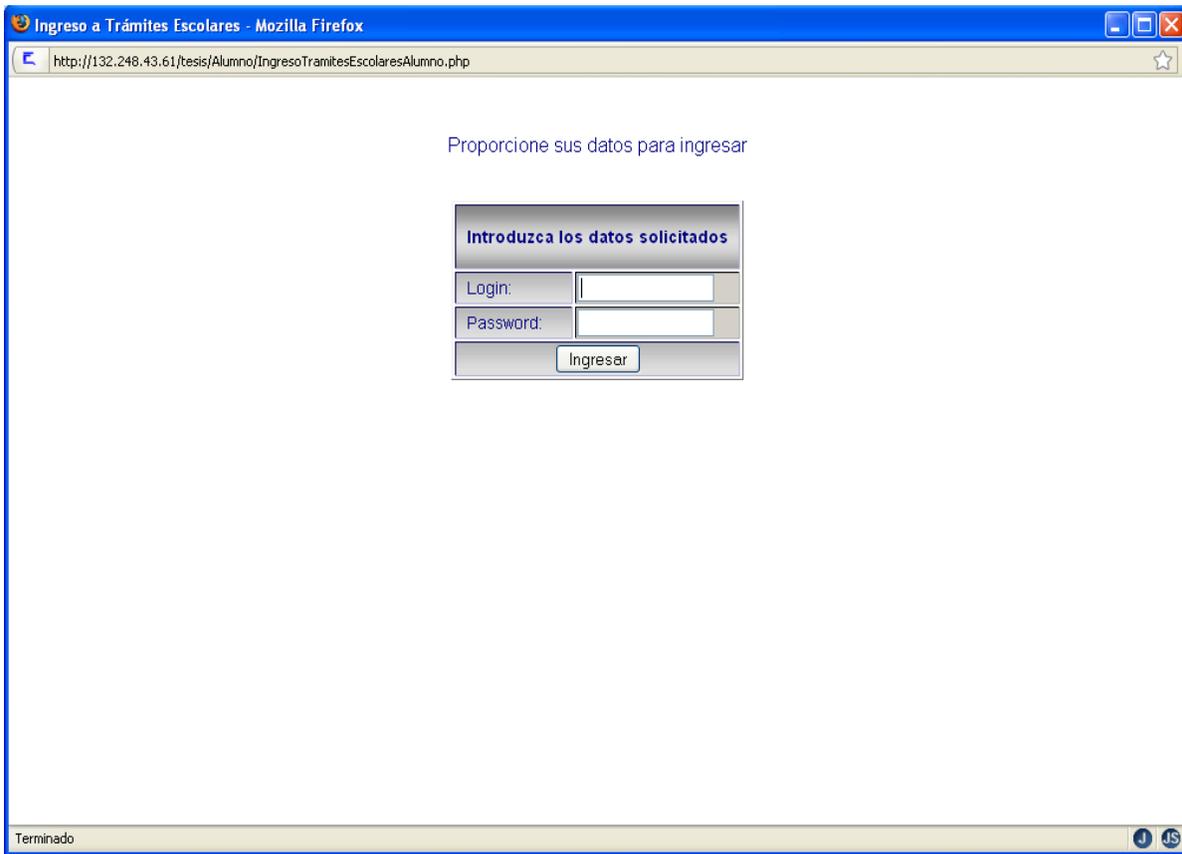


Figura 3. 29 Pantalla para ingresar login y password para el usuario Alumno

Campo:	Representación
Login	Text Field
Password	Text Field
Ingresar	Button

Tabla 3. 25

## Registro de asignaturas

Nombre: registroAlumno.php

Página anterior: IngresoTramitesEscolaresAlumno.php

Página siguiente: registroAlumno.php

Descripción: Página para que el Alumno ingrese los datos de las asignaturas y grupos en que desea registrarse, así como para mostrar los datos de aquellos en que ya se registró.

Pantalla:

Clave	Asignatura	Grupo	Profesor	Aula	Créditos	Horario							
						L	M	J	V	S	Hora inicio	Hora fin	
1333	TALLER DE ARQUITECTURA III	5316	Ver profesores	C102	25		*					07:00	15:00
										*		07:00	14:00
											*	09:00	14:00
1131	TEORIA DE LA ARQUITECTURA I	5121	Ver profesores	F101	4							00:00	00:00
1133	MATEMATICAS APLICADAS I	5129	Ver profesores	F101	4		*					09:00	11:00
										*		07:00	09:00
1134	SISTEMAS ESTRUCTURALES I	5129	Ver profesores	C102	6		*					09:00	12:00
1233	MATEMATICAS APLICADAS II	5232	Ver profesores	C201	4				*			18:00	20:00

Figura 3. 30 Pantalla que muestra las asignaturas registradas por el Alumno

Campo:	Representación
Clave de la asignatura	Text Field
Grupo:	Text Field
Alta	Button
Baja	Button
Imprimir Comprobante	Link
Cerrar sesión	Link

Tabla 3. 26

## Altas, Bajas y cambios

Nombre: ABCAlumno.php  
 Página Anterior: IngresoTramitesEscolares.php  
 Página Siguiente: ABCAlumno.php  
 Descripción: Página en donde se muestran los cursos asignados al Alumno previamente por el sistema en el periodo correspondiente a Asignación, así como las opciones para realizar altas, bajas y cambios de cursos.

Pantalla:

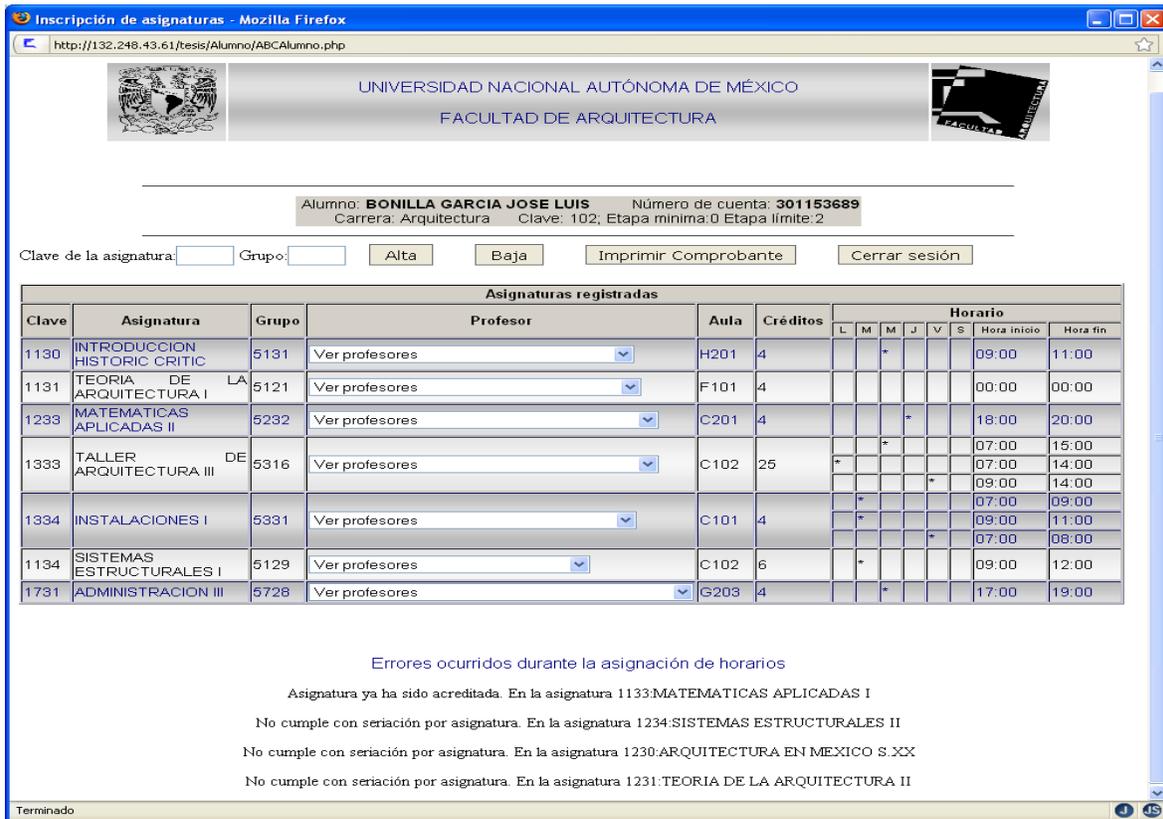


Figura 3. 31 Pantalla que muestra las asignaturas asignadas por el Sistema así como los errores durante el proceso de Asignación

Campos:	Representación
Clave de asignatura	Text Field
Grupo	Text Field
Alta	Button
Baja	Button

Tabla 3. 27A

Campos:	Representación
Actualizar datos	Link
Imprimir comprobante	Link
Cerrar sesión	Link

Tabla 3.27B

### Alumno actualizar datos

Nombre: ActDatAlu.php  
 Página Anterior: IngresoTramitesEscolares.php  
 Página Siguiente: ActDatAlu2.php  
 Descripción: Página en donde se muestra la información personal del Alumno.  
 Pantalla:

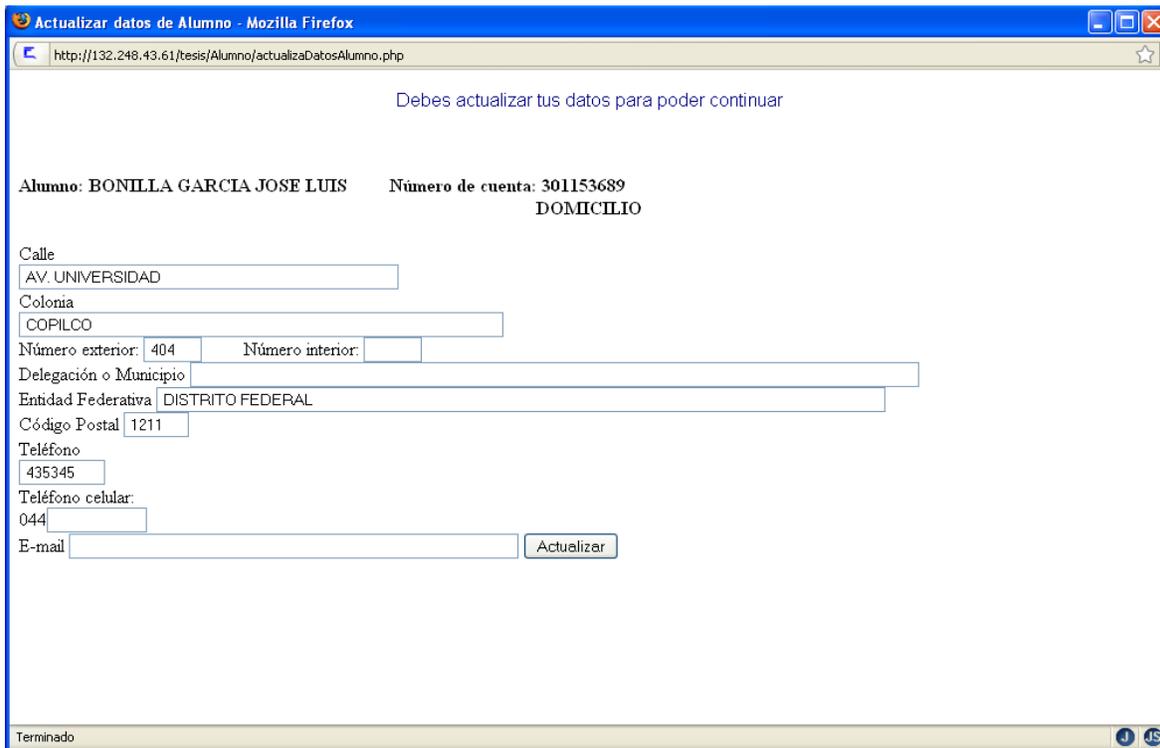


Figura 3. 32 Pantalla para ingresar la información personal del Alumno

Campos:	Representación
Calle	Text Field
Colonia	Text Field
Número exterior	Text Field
Número interior	Text Field
Delegación o Municipio	Text Field
Entidad Federativa	Text Field

Tabla 3. 28A

Campos:	Representación
Código Postal	Text Field
Teléfono	Text Field
Teléfono Celular	Text Field
E-mail	Text Field
Actualizar	Button

Tabla 3.28B

### 3.12.1.2 Ingreso General Trámites Académicos

Nombre: IngresoTramitesEscolaresGeneral.php

Página anterior: index.html

Página siguiente: opciones.php

Descripción: Página que permite la autenticación de los actores de asuntos escolares.

Pantalla:

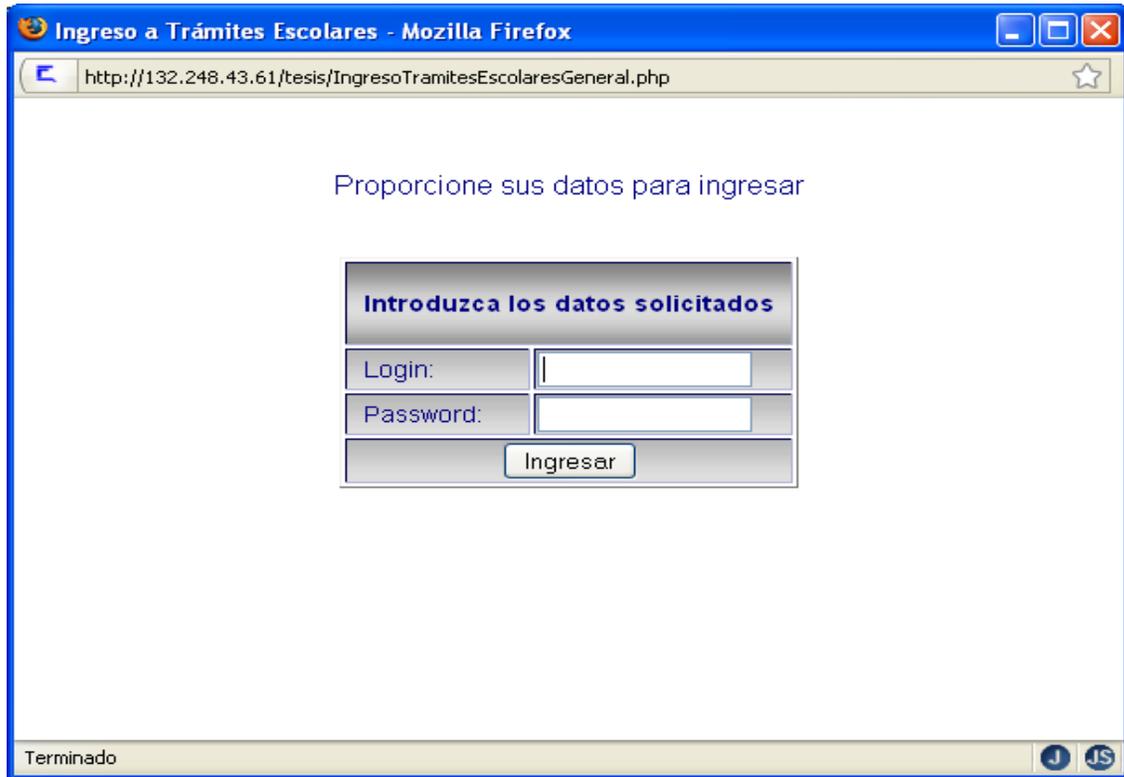


Figura 3. 33Pantalla para ingresar login y password para los usuarios de Asuntos Escolares

Campo:	Representación
Login	Text Field
Password	Text Field
Ingresar	Button

Tabla 3. 29

### 3.12.1.3 Mantenimiento

#### Menú general para Mantenimiento

Nombre: opciones.php  
Página Anterior: IngresoTramitesEscolares.php  
Página Siguiete: ModCup.php  
CanGru.php  
CamCal.php  
AsiGru.php  
ActCurComInd.php

Descripción: Página que despliega las opciones para el usuario Mantenimiento.

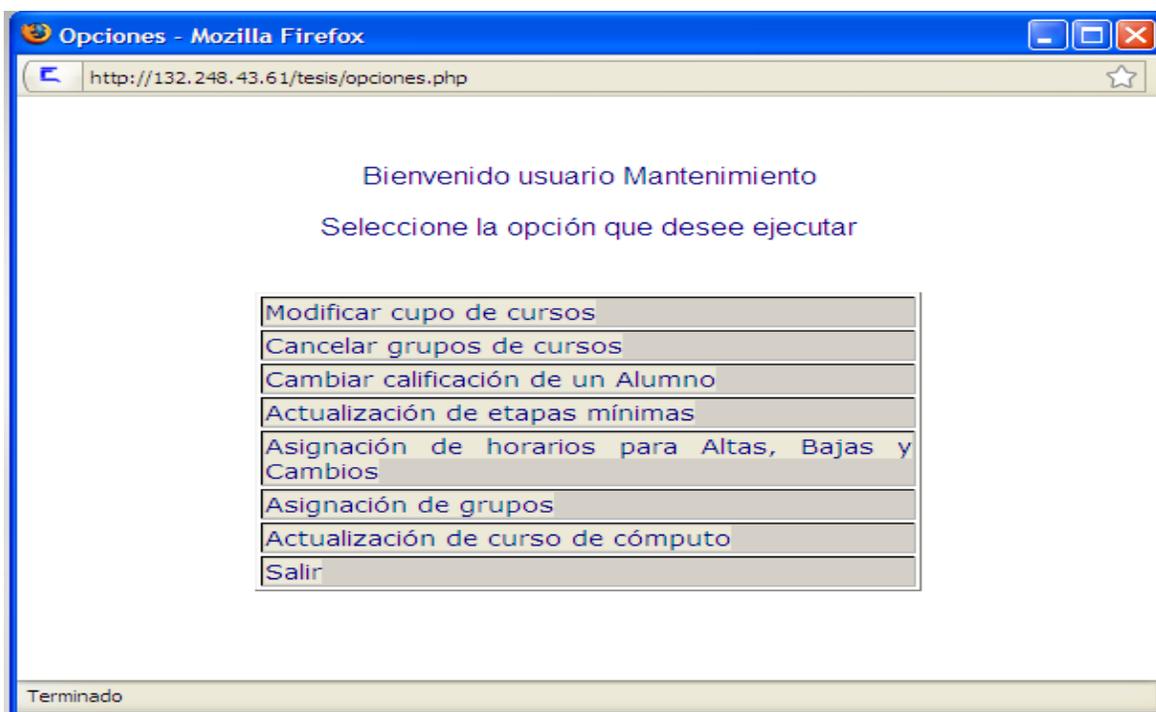


Figura 3. 34 Menú de opciones para el usuario Mantenimiento

Campo:	Representación
Modificar cupo de cursos	Link
Cancelar grupos de cursos	Link
Cambiar calificación de un Alumno	Link
Actualización de etapas mínimas	Link

Tabla 3. 30

Campo:	Representación
Asignación de horarios para Altas, Bajas y Cambios	Link
Asignación de grupos	Link
Actualización de curso de cómputo	Link
Salir	Link

Tabla 3.30B

## Modificar cupo de cursos

Al seleccionar la opción “Modificar cupo de cursos” se mostrarán las etapas de la carrera de Arquitectura, al elegir una de éstas etapas se desplegarán las asignaturas correspondientes a la etapa elegida. Posteriormente al seleccionar uno de los cursos se mostrarán los grupos y al elegir uno de estos aparecerá la pantalla 3.35

Nombre: ConCup.php

Página Anterior: SelGru.php

Página Siguiente: ActCup.php  
opciones.php  
salir.php

Descripción: Página en donde se muestran la información del grupo seleccionado además de un campo para proporcionar el nuevo cupo del mismo.

Pantalla:

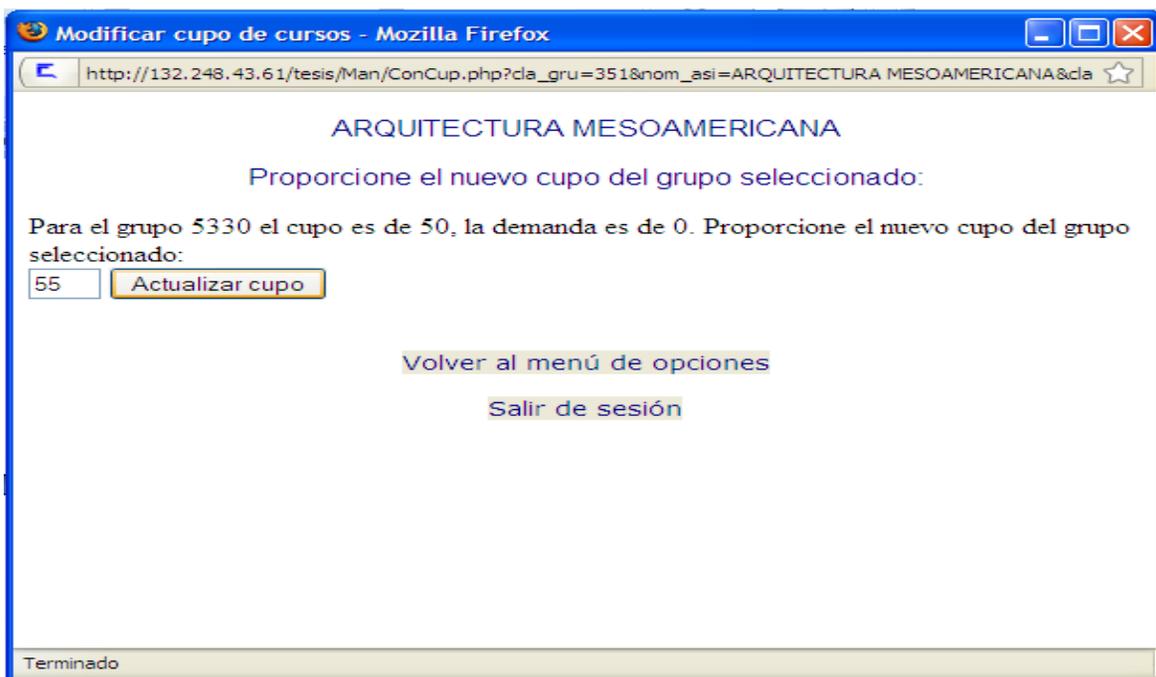


Figura 3. 35 Pantalla para ingresar un nuevo cupo de un grupo

Campos:	Representación
Cupo	Text Field
Actualizar Cupo	Button

Tabla 3. 31A

Campos:	Representación
Volver al menú de opciones	Link
Salir de sesión	Link

Tabla 3.31B

Nombre: ActCup.php

Página Anterior: ConCup.php

Página Siguiente: SelGru.php  
opciones.php  
salir.php

Descripción: Página en donde se muestra que el cupo de la asignatura seleccionada ha sido actualizada.

Pantalla:

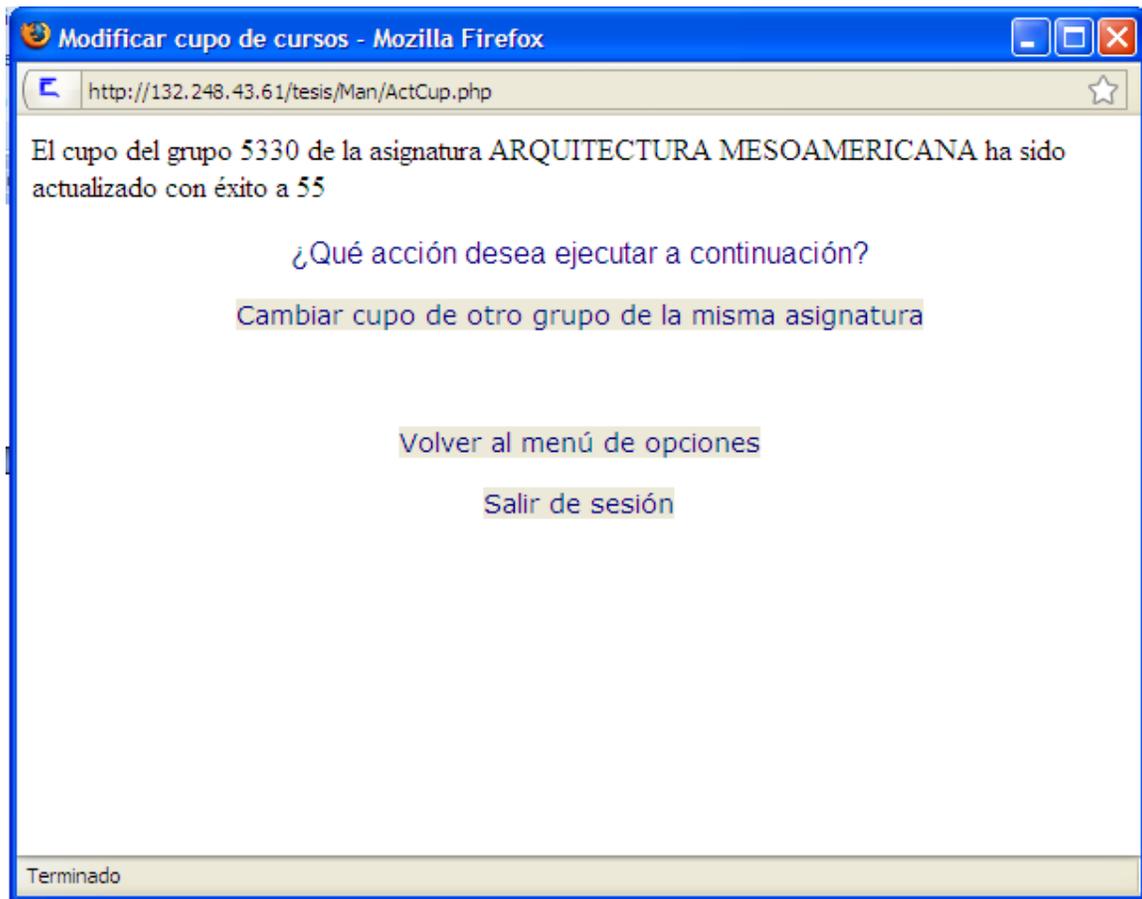


Figura 3. 36 Pantalla que muestra que el cupo de un grupo ha sido actualizado con éxito

Campos:	Representación
Cambiar cupo de otro grupo de la misma asignatura	Link
Volver al menú de opciones	Link
Salir de sesión	Link

Tabla 3. 32

### Cancelar grupos de Cursos Obligatorios

Al seleccionar la opción “Cancelar grupos de Cursos Obligatorios” se mostrarán las etapas de la carrera de Arquitectura, al elegir una de las etapas se desplegarán las asignaturas correspondientes a esta etapa. Posteriormente al seleccionar uno de los cursos se mostrarán los grupos y al elegir uno de estos aparecerá la pantalla 3.37

Nombre: CanGru4.php

Página Anterior: CanGru3.php

Página Siguiente: CanGru5.php

opciones.php

salir.php

Descripción: Página en donde se despliegan las claves de los grupos ordinarios así como de los extraordinarios del grupo seleccionado.

Pantalla:

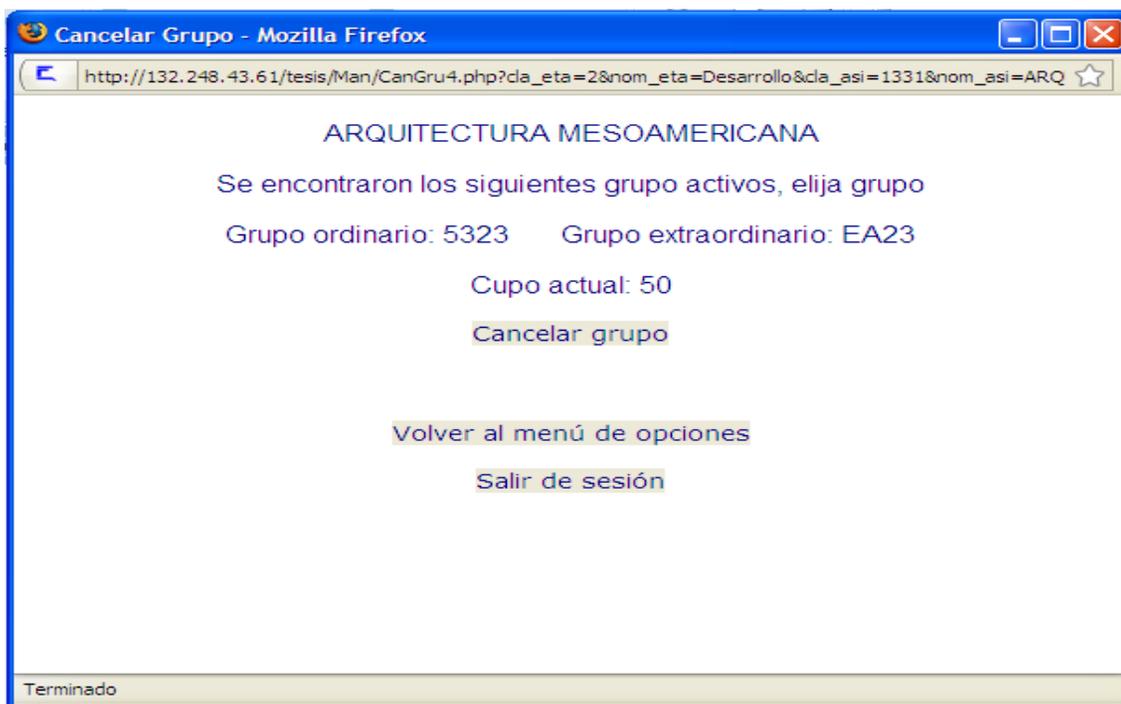


Figura 3. 37 Pantalla que muestra los grupos de una asignatura seleccionada

Campos:	Representación
Cancelar grupo	Link
Volver al menú de opciones	Link

Tabla 3. 33 A

Campos:	Representación
Salir de sesión	Link

Tabla 3.33B

Nombre: CanGru5.php

Página Anterior: CanGru4.php

Página Siguiente: opciones.php  
salir.php

Descripción: Página en donde se muestra que el grupo seleccionado ha sido cancelado.

Pantalla:



Figura 3. 38 Pantalla que muestra que el grupo seleccionado ha sido cancelado

Campos:	Representación
Volver al menú de opciones	Link
Salir de sesión	Link

Tabla 3. 34

## Cambiar calificación de un Alumno

Al seleccionar la opción “Cambiar calificación de un Alumno” se solicitará el número de cuenta del Alumno y la clave de la asignatura, al ingresar estos datos se mostrará la calificación obtenida en la asignatura proporcionada. Al confirmar que los datos son correctos se mostrará la información correspondiente a la pantalla 3.39

Nombre: CamCal3.php

Página anterior: CamCal2.php

Página siguiente: CamCal4.php

Descripción: Página que permite el ingreso de la nueva calificación del Alumno en el último grupo de la asignatura cursada.

Pantalla:

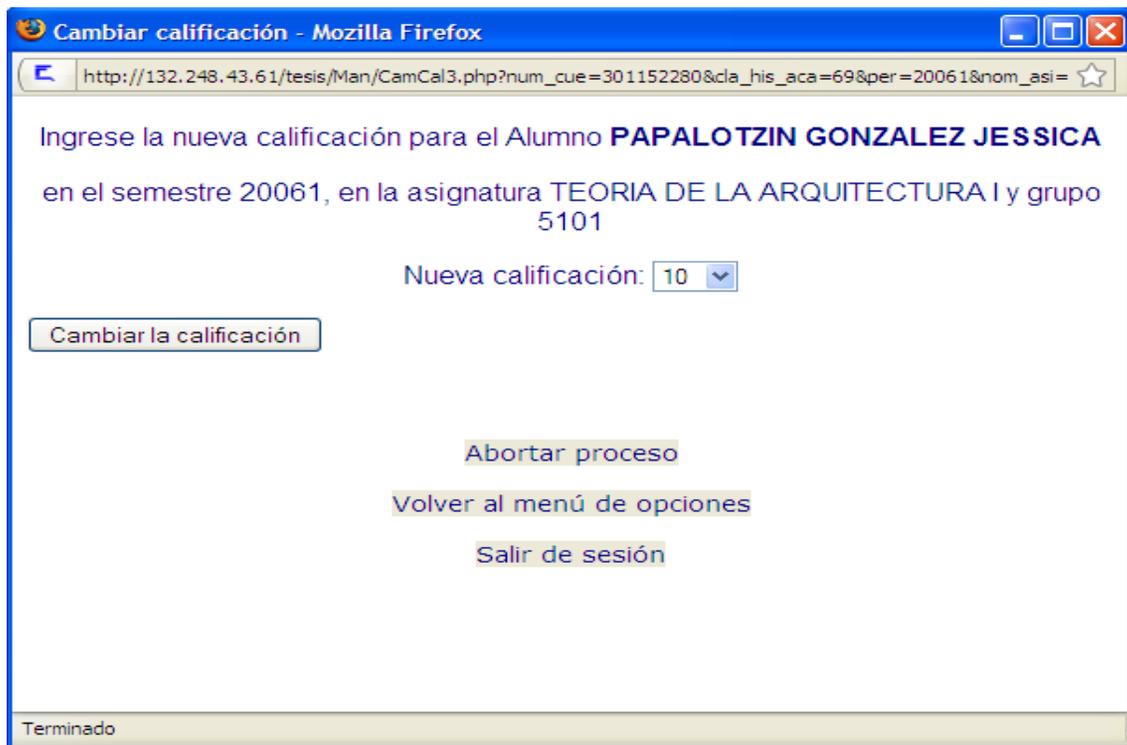


Figura 3. 39 Pantalla para ingresar una calificación de un Alumno

Campo	Representación
Nueva calificación	Menu list
Cambiar calificación	Button
Abortar proceso	Link

Tabla 3. 35 A

Campo	Representación
Volver al menú de opciones	Link
Salir de sesión	Link

Tabla 3.35B

Nombre: CamCal4.php  
 Página anterior: CamCal3.php  
 Página siguiente: ninguna  
 Descripción: Página que muestra que la calificación de la asignatura seleccionada ha sido cambiada.

Pantalla:

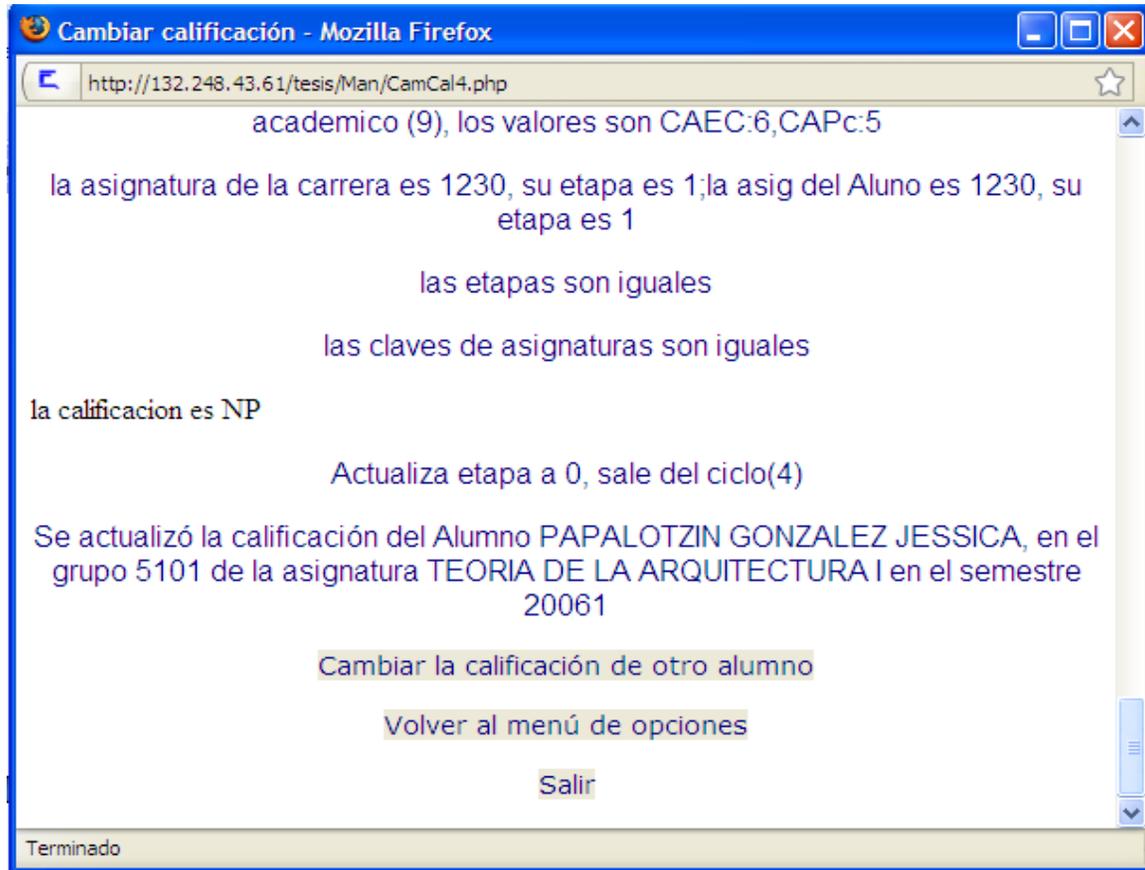


Figura 3. 40Pantalla que muestra la actualización de una calificación de un Alumno

Campo	Representación
Cambiar la calificación de otro Alumno	Link
Volver al menú de opciones	Link
Salir	Link

Tabla 3. 36

## Actualización de etapas mínimas

Nombre: ActEtaMinTot.php

Página anterior: opciones.php

Página siguiente: ninguna

Descripción: Página que muestra el fin de la actualización de la etapa mínima (avance por bloques) para todos los Alumnos de Arquitectura.

Pantalla:

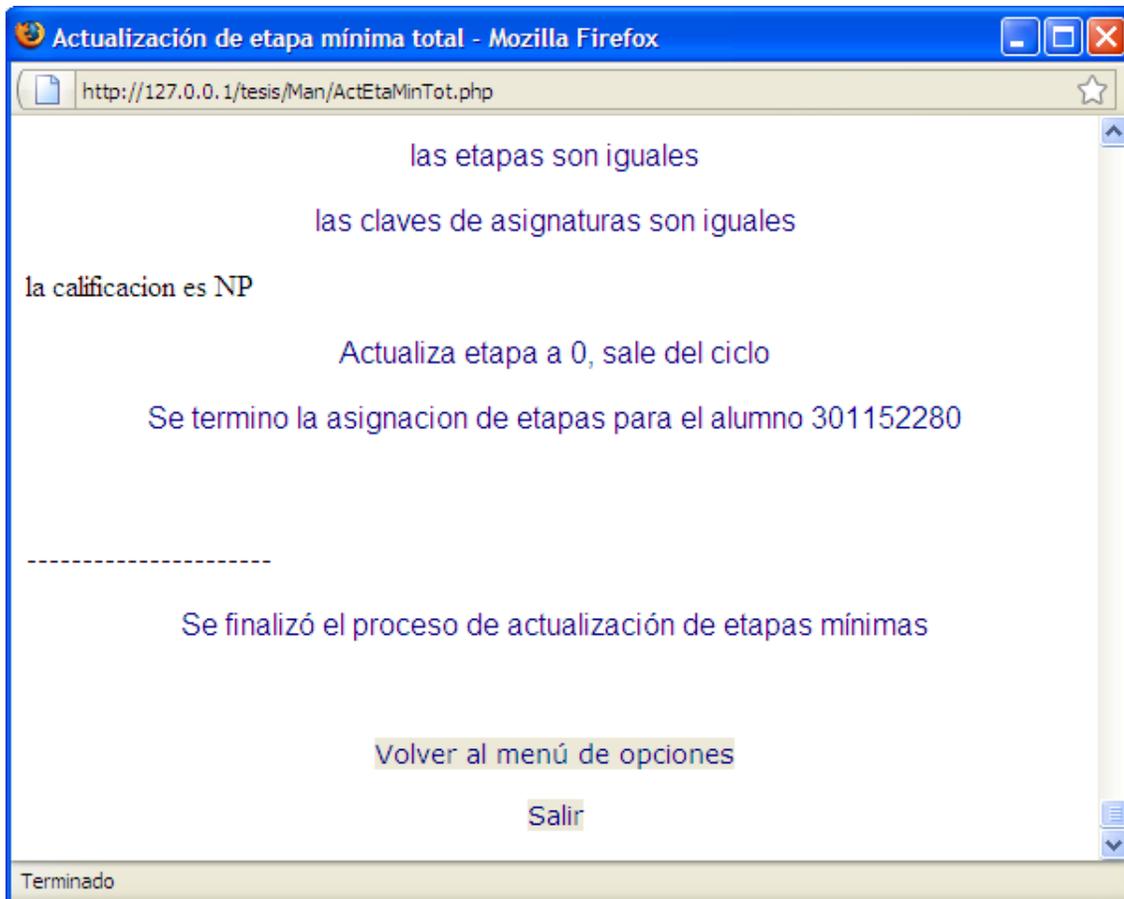


Figura 3. 41 Pantalla correspondiente al proceso de actualización de Etapas Mínimas

Campo:	Representación
Volver al menú de opciones	Link
Salir	Link

Tabla 3. 37

## Asignación de grupos

Nombre: AsiGru.php

Página anterior: opciones.php

Página siguiente: ninguna

Descripción: Página que muestra el fin del proceso de asignación de grupos para toda la población estudiantil de Arquitectura.

Pantalla:

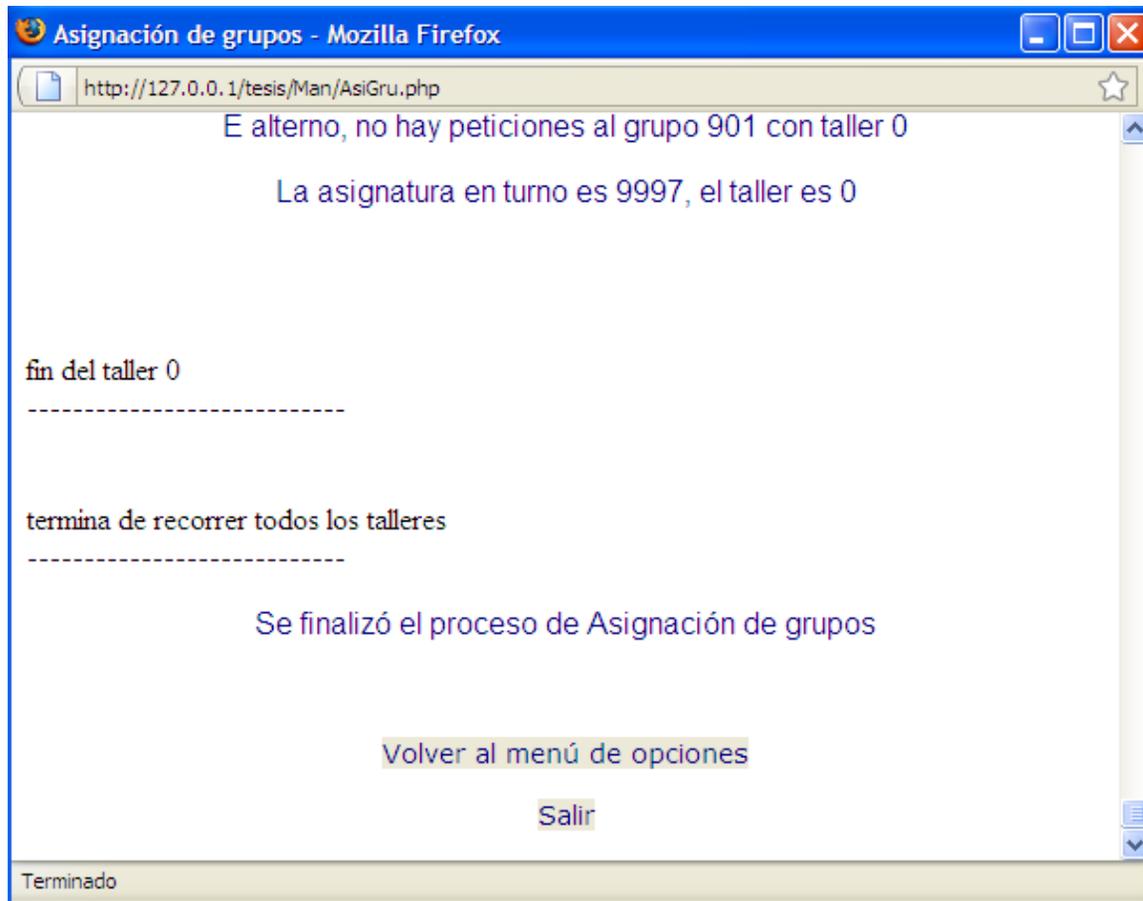


Figura 3. 42 Página correspondiente al proceso de Asignación de grupos

Campo:	Representación
Volver al menú de opciones	Link
Salir	Link

Tabla 3. 38

## Actualización de curso de cómputo individual

Nombre: ActCurComInd.php

Página anterior: opciones.php

Página siguiente: ActCurComInd2.php

Descripción: Página para ingresar el número de cuenta de un alumno.

Pantalla:

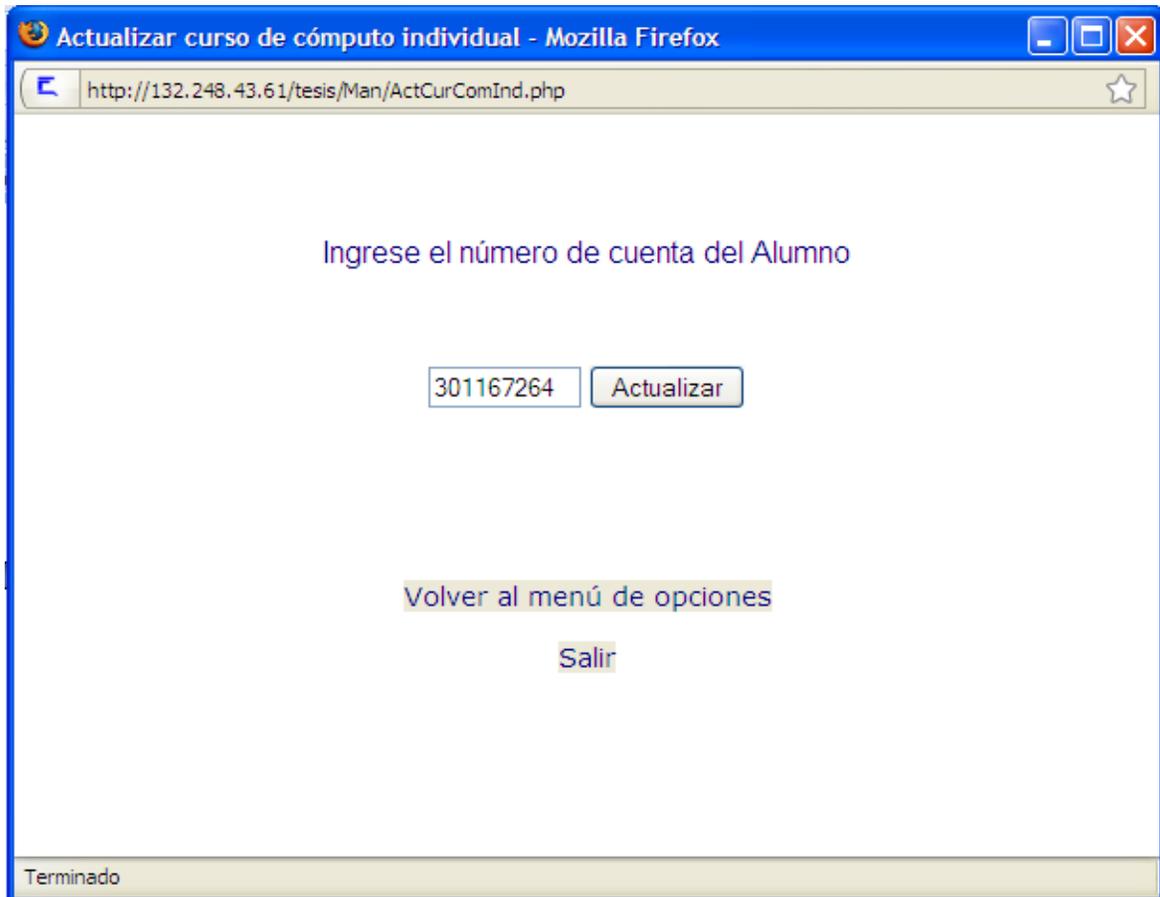


Figura 3. 43 Pantalla para ingresar el número de cuenta de un Alumno

Campo:	Representación
Número de cuenta	Text Field
Actualizar	Button
Volver al menú de opciones	Link
Salir	Link

Tabla 3. 39

Nombre: ActCurComInd2.php  
 Página anterior: ActCurComInd.php  
 Página siguiente: ninguna  
 Descripción: Página que muestra que el requisito de curso de cómputo se ha actualizado para el alumno.

Pantalla:

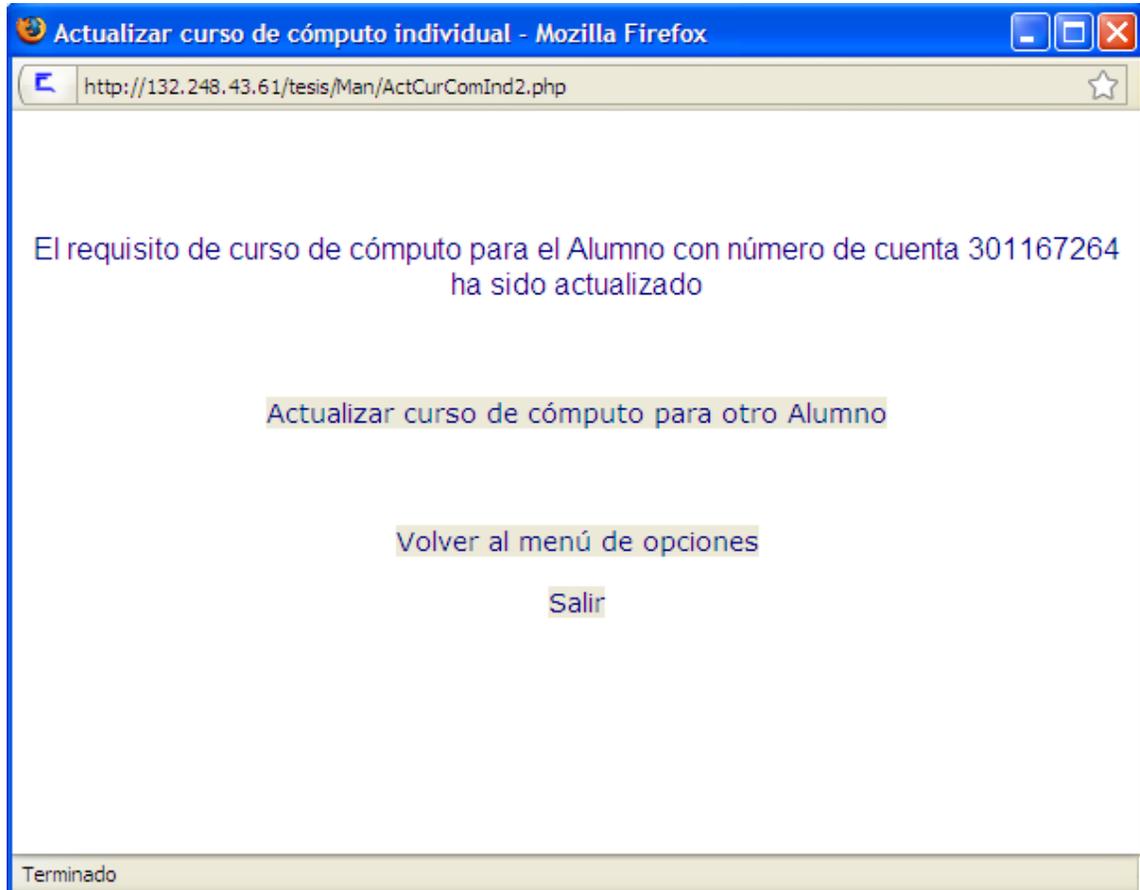


Figura 3. 44 Pantalla que despliega que el requisito de curso de cómputo ha sido actualizado

Campo:	Representación
Actualizar curso de cómputo para otro Alumno	Link
Volver al menú de opciones	Link
Salir	Link

Tabla 3. 40

### 3.12.1.4 Coordinador de Taller

Nombre: opciones.php  
Página anterior: IngresoTramitesEscolaresGeneral.html  
Página siguiente: CanTal.php  
Descripción: Página que sirve para que el actor Coordinador de Taller seleccione la opción que desea ejecutar.  
Pantalla:

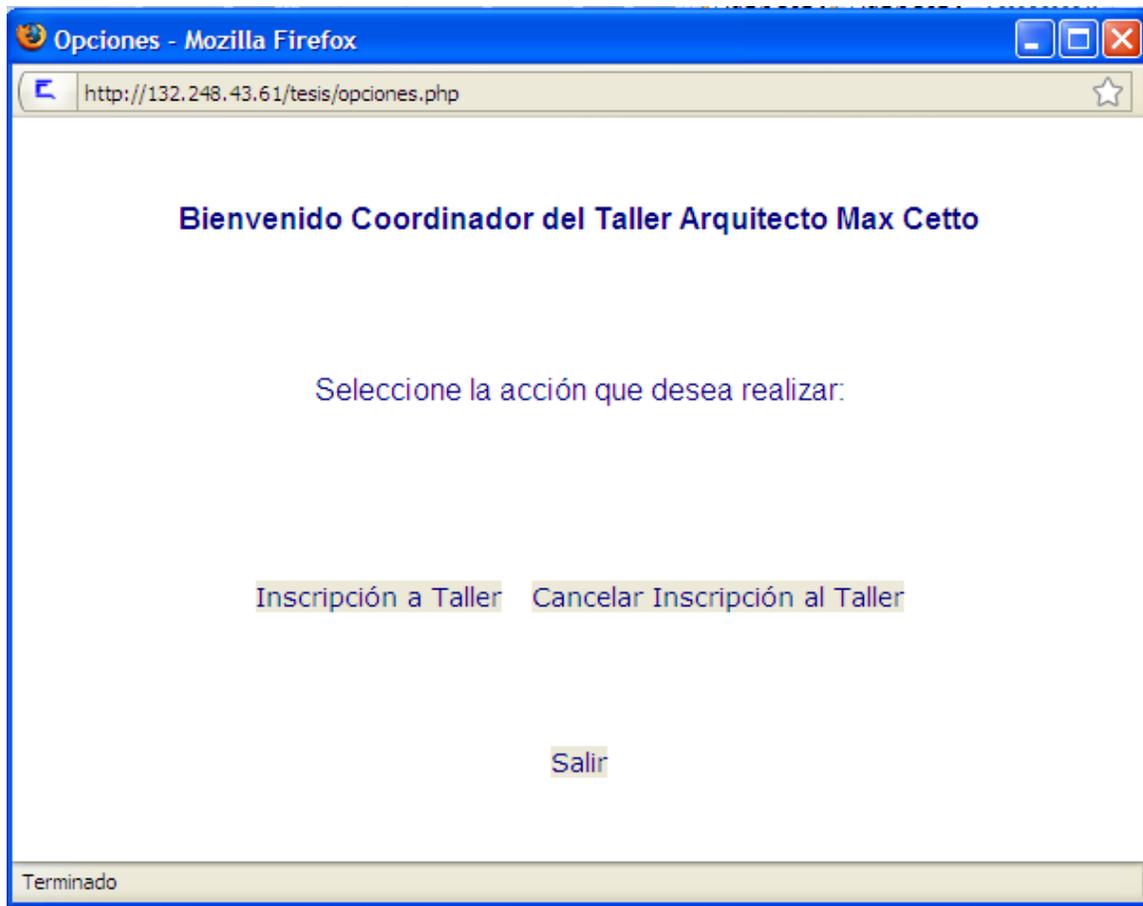


Figura 3. 45 Menú de opciones del usuario Coordinador de Taller

Campo:	Representación
Inscripción a Taller	Link
Cancelar inscripción al Taller	Link
Salir	Link

Tabla 3. 41

Nombre: RegTal.php  
 Página anterior: opciones.php  
 Página siguiente: RegTal2.php  
 Descripción: Página para que el actor Coordinador de Taller ingrese el número de cuenta del Alumno.

Pantalla:

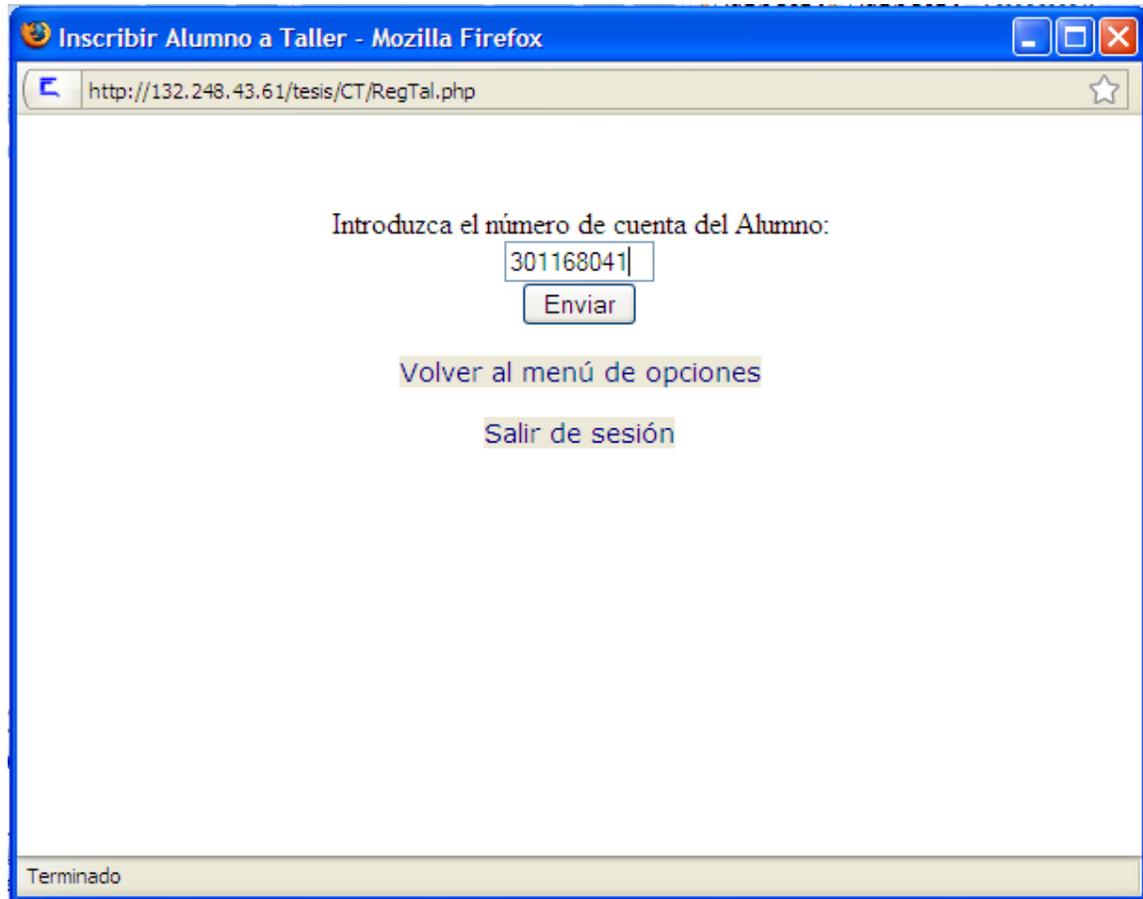


Figura 3. 46 Pantalla para ingresar el número de cuenta de un Alumno

Campo:	Representación
Número de cuenta	Text Field
Enviar	Button
Volver al menú de opciones	Link
Salir de sesión	Link

Tabla 3. 42

Al ingresar los datos solicitados y siendo estos correctos, se verificará el caso del Alumno y se mostrarán las opciones de acuerdo a su caso, como la información de la figura 3.47

Nombre: RegTal3.php  
 Página anterior: RegTal2.php  
 Página siguiente: RegTal4.php  
 Descripción: Página que muestra al Coordinador de Taller las opciones que el Alumno tiene para poder inscribir Taller.

Pantalla:



Figura 3. 47 Pantalla que despliega las opciones para inscripción a un Taller de un Alumno

Campo:	Representación
Clave Grupo:885,Cupo:49,Demanda:1,Clave_int_grupo:5002 *	Link
Volver al menú de opciones	Link
Salir de sesión	Link

Tabla 3. 43

\* NOTA: El número de opciones para inscribir, así como los datos del grupo pueden variar, ya que dependen del caso del Alumno y de cuántos Alumnos más se hayan inscrito en éste.

Nombre: RegTal4.php  
 Página anterior: RegTal3.php  
 Página siguiente: ninguna  
 Descripción: Página que muestra al Coordinador de Taller que el proceso de Inscribir a un Alumno a su Taller fue finalizado con éxito.

Pantalla:

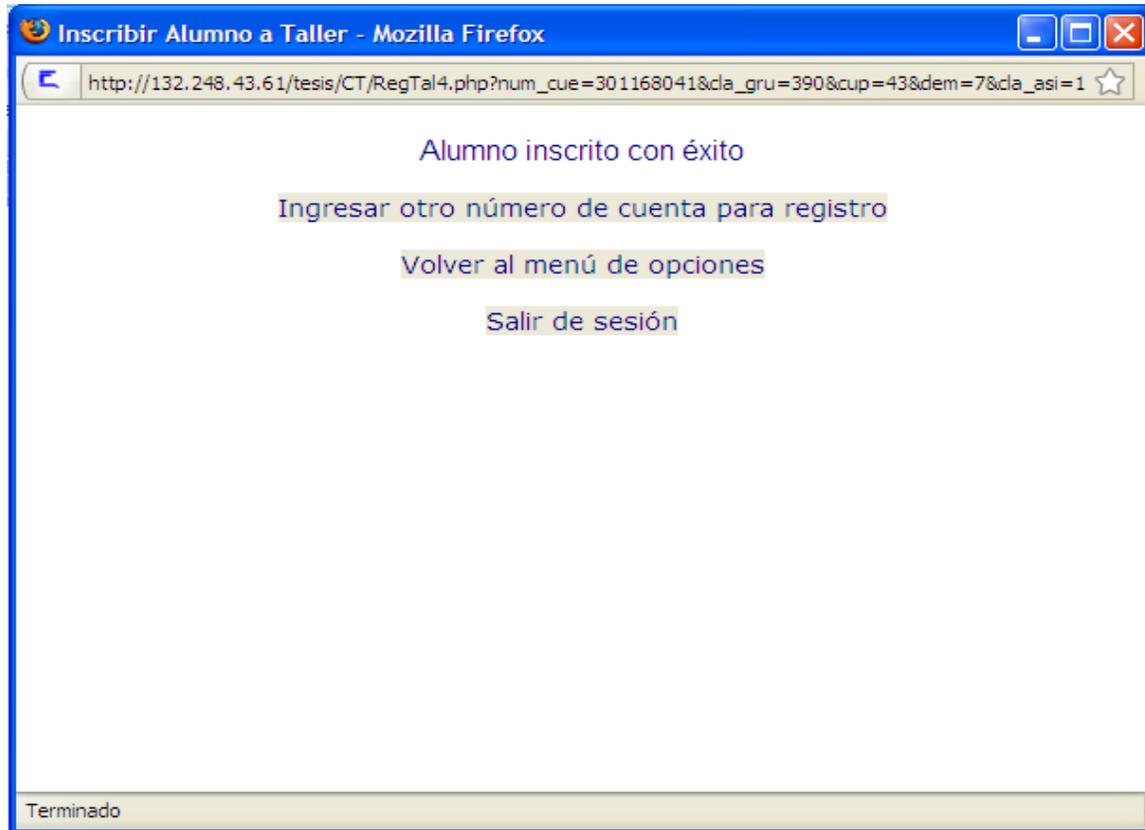


Figura 3. 48 Pantalla que despliega el resultado de una inscripción a Taller

Campo:	Representación
Ingresar otro número de cuenta para registro	Link
Volver al menú de opciones	Link
Salir de sesión	Link

Tabla 3. 44

## Cancelar inscripción a Taller

Al seleccionar la opción “Cancelar inscripción a Taller” se solicita el número de cuenta del Alumno, al ingresar este dato se mostrará nombre, número de cuenta, carrera y Taller en que está inscrito el Alumno. Se confirma que los datos estén correctos y posteriormente se mostrará la información correspondiente a la pantalla 3.49

Nombre: CanTal2.php  
Página anterior: CanTal.php  
Página siguiente: CanTal3.php  
Descripción: Página que muestra al actor Coordinador de Taller los datos del Alumno, cuando éste está inscrito en la asignatura de su Taller.

Pantalla:

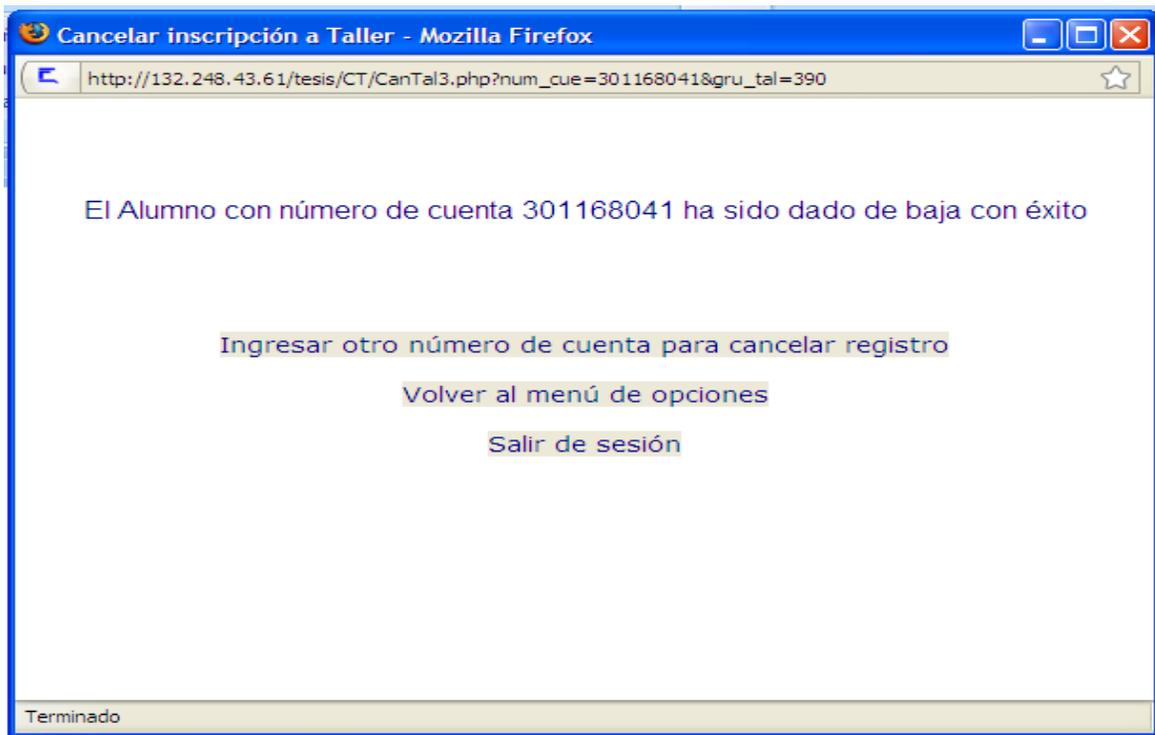


Figura 3. 49 Pantalla que muestra que un Alumno ha sido dado de baja de un Taller

Campo:	Representación
Ingresar otro número de cuenta para cancelar registro	Link
Volver al menú de opciones	Link

Tabla 3. 45 A

Campo:	Representación
Salir de sesión	Link

Tabla 3.45B

### 3.12.1.5 Coordinador de Carrera

Nombre: opciones.php

Página Anterior: IngresoTramitesEscolares.php

Página Siguiente: SemTal.php

salir.php

Descripción: Página en donde se muestran los diferentes Talleres que componen la carrera de Arquitectura, de los cuales se actualizará el cupo de uno de sus grupos.

Pantalla:



Figura 3. 50 Menú de opciones del usuario Coordinador de Carrera

Campos:	Representación
Taller Arquitecto Luis Barragán	<a href="#">Link</a>
Taller Arquitecto Federico Mariscal y Piña	<a href="#">Link</a>
Taller Arquitecto Hannes Meyer	<a href="#">Link</a>
Taller Arquitecto Juan O' Gorman	<a href="#">Link</a>
Taller José Revueltas	<a href="#">Link</a>
Taller Tres	<a href="#">Link</a>
Taller Uno	<a href="#">Link</a>
Taller Arquitecto José Villagrán García	<a href="#">Link</a>
Taller Arquitecto Max Cetto	<a href="#">Link</a>
Taller Ehécatl XXI	<a href="#">Link</a>
Taller Arquitecto Juan García Gayoú	<a href="#">Link</a>
Taller Arquitecto Domingo García Ramos	<a href="#">Link</a>
Taller Arquitecto Jorge González Reyna	<a href="#">Link</a>
Taller Arquitecto Carlos Lazo Barreiro	<a href="#">Link</a>
Taller Arquitecto Carlos Leduc Montaña	<a href="#">Link</a>
Taller Arquitecto Ramón Marcos Noriega	<a href="#">Link</a>
Salir	<a href="#">Link</a>

Tabla 3. 46

### **Modificar cupo de Taller Etapa grupo**

Al seleccionar uno de los Talleres se muestran las etapas de la carrera de Arquitectura, se selecciona una y se desplegarán los grupos correspondientes a la etapa elegida. Se escoge el grupo al cual quiere realizar el cambio de cupo y se mostrará la información como se observa en la figura 3.51

Nombre: SemTal3.php

Página Anterior: SemTal2.php

Página Siguiente: ActCupTal.php  
opciones.php  
salir.php

Descripción: Página en donde se muestra la información del grupo elegido: cupo y demanda del mismo, y un campo para proporcionar el nuevo cupo del grupo.

Pantalla:



Figura 3. 51 Pantalla para ingresar nuevo cupo de un grupo de Taller

Campos:	Representación
Cupo	Text Field
Actualizar Cupo	Button

Tabla 3. 47

Nombre: ActCupTal.php

Página Anterior: SemTal3.php

Página Siguiente: SemTal.php  
opciones.php  
salir.php

Descripción: Página en donde se muestra que el cupo del grupo seleccionado ha sido actualizado.

Pantalla:

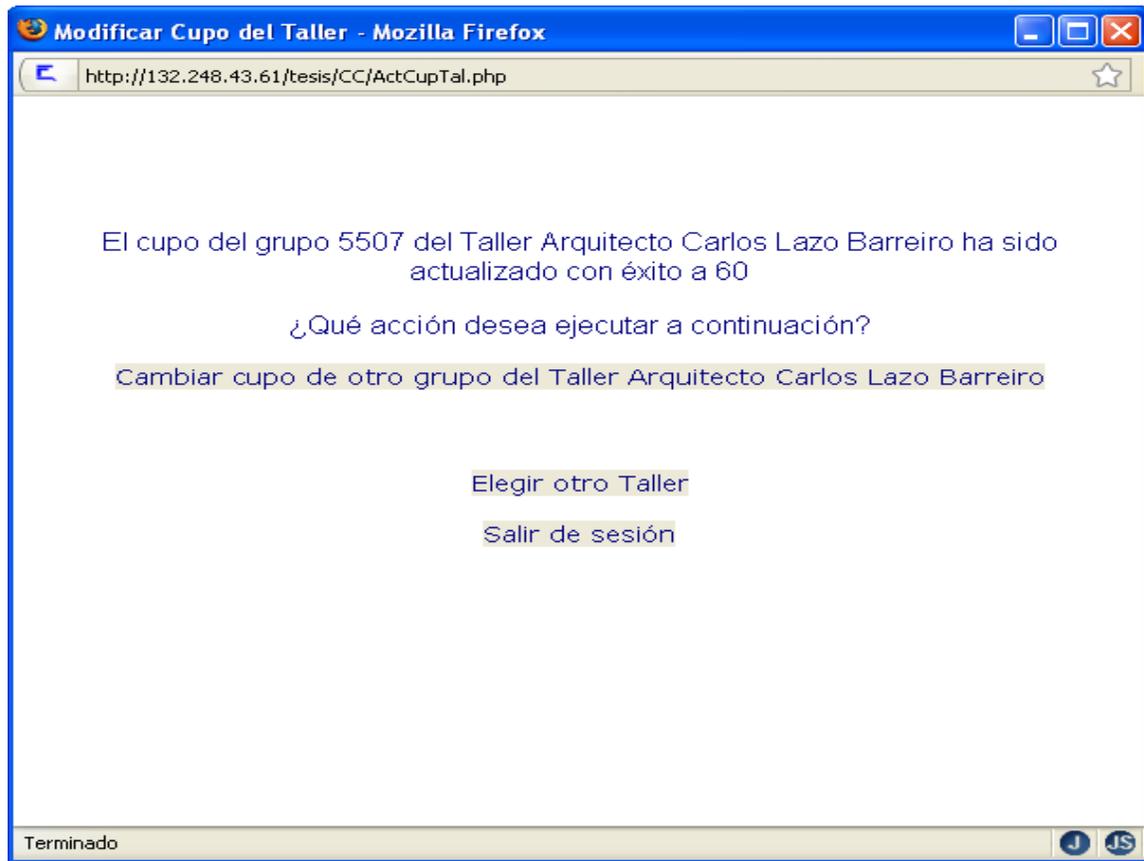


Figura 3. 52 Pantalla que muestra que el cupo de un grupo de Taller ha sido actualizado

Campos:	Representación
Cambiar cupo de otro grupo del mismo Taller	Link
Elegir otro Taller	Link
Salir de sesión	Link

Tabla 3. 48

### 3.12.1.6 Jefe de Asuntos Escolares

Nombre: opciones.php

Página Anterior: IngresoTramitesEscolares.php

Página Siguiente: InsCasExt.php  
salir.php

Descripción: Página en donde se muestran las opciones para el Jefe de Asuntos Escolares.

Pantalla:

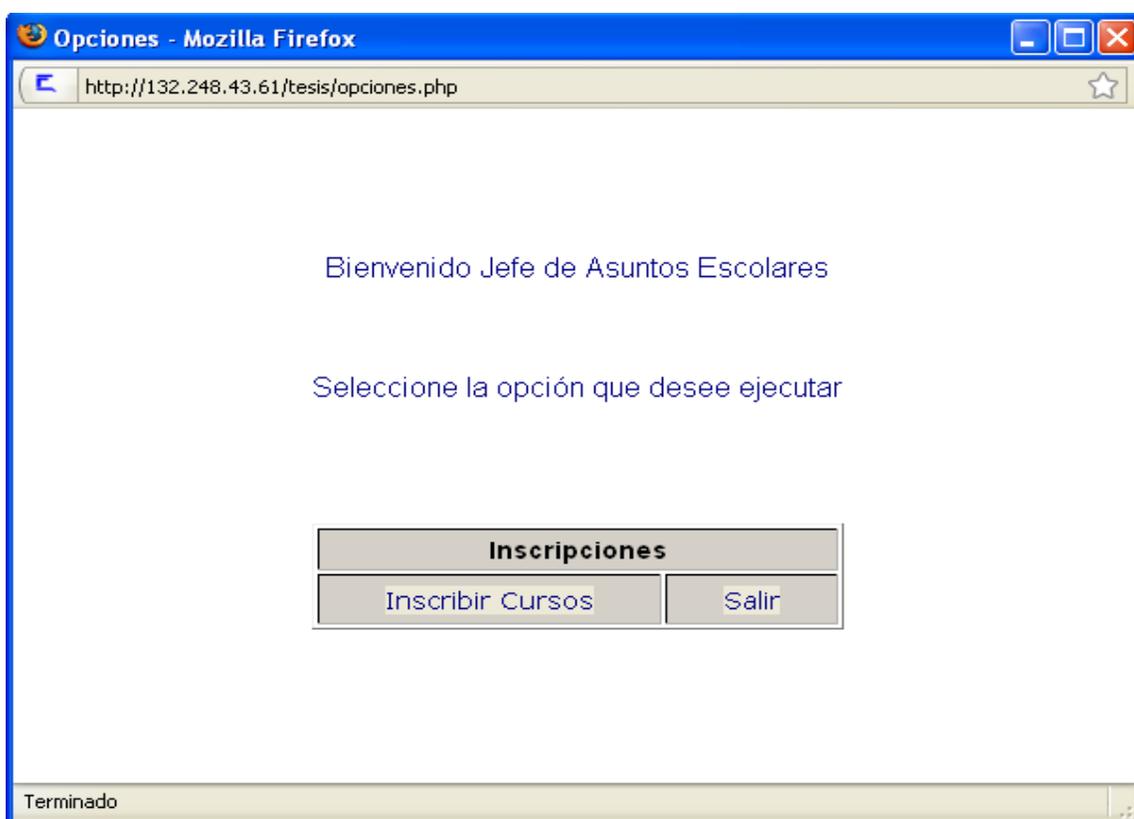


Figura 3. 53 Menú de opciones para el usuario Jefe de Asuntos Escolares

Campos:	Representación
Inscribir Cursos	Link
Salir	Link

Tabla 3. 49

## Inscribir cursos

Al seleccionar la opción “Inscribir cursos” se solicitará el número de cuenta del Alumno, al ingresar este dato se mostrará nombre y número de cuenta del Alumno. Se confirma que los datos estén correctos se mostrarán las etapas de la carrera de Arquitectura, se debe seleccionar una, en donde se desplegará las asignaturas de esa etapa y finalmente se elige un grupo para inscribir al Alumno. Posteriormente se desplegará una pantalla correspondiente a la figura 3.54

Nombre: InsCasExt6.php

Página Anterior: InsCasExt5.php

Página Siguiente: opciones.php  
salir.php

Descripción: Página en donde se muestra que el Alumno ha sido inscrito al grupo de la asignatura seleccionada.

Pantalla:

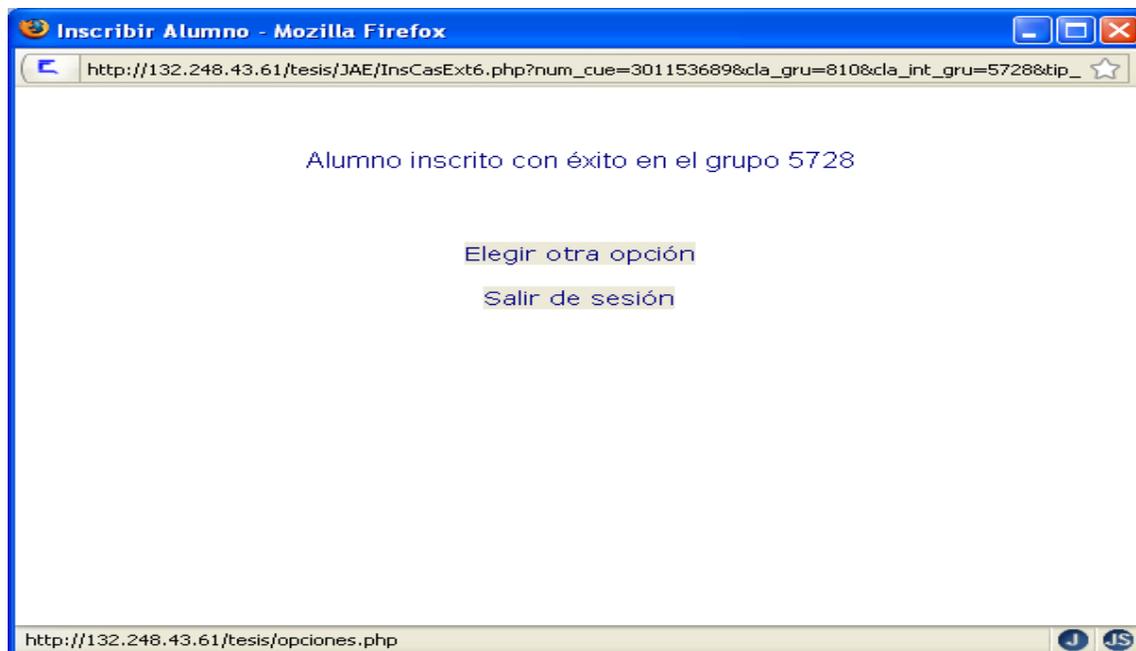


Figura 3. 54 Página que muestra que un alumno se ha inscrito en una asignatura

Campos:	Representación
Elegir otra opción	Link
Salir de sesión	Link

Tabla 3. 50

## Consultar historial académico completo

Nombre: ConHisAcaAlu.php

Página Anterior: InsCasExt2.php

Página Siguiente: Ninguna

Descripción: Página en donde se despliega la Historia Académica del Alumno: asignaturas con calificación, clave, periodo y tipo de examen.

Pantalla:



Figura 3. 55 Página que muestra el Historial Académico de un Alumno

Campos:	Representación
Cerrar ventana	Link

Tabla 3. 51

Nota: Para ver la secuencia completa de pantallas ver anexo **D Vistas Secuencia de Pantallas.**

# Capítulo 4 Programación

## 4.1 PHP

PHP es un lenguaje interpretado del lado del servidor, caracterizado por su potencia, versatilidad, robustez y modularidad. Los programas escritos en PHP son embebidos directamente en el código HTML y ejecutados por el servidor web a través de un intérprete antes de transferir al cliente un resultado en forma de código HTML. Al ser un lenguaje que sigue la corriente open source, tanto el intérprete como su código fuente son totalmente accesibles de forma gratuita en la red. La dirección oficial en la que puede descargarse es <http://www.php.net>

### 4.1.1 Características de PHP

Los programas escritos en PHP funcionan sobre diferentes plataformas, trabajando sobre la mayoría de los servidores web, el lenguaje está preparado para interactuar con más de 20 tipos de manejadores de bases de datos.

Actualmente PHP permite realizar una multitud de tareas útiles para el desarrollo web como:

- Funciones para programar sistemas de correo electrónico vía web.
- Funciones de administración y gestión de bases de datos específicas para la mayoría de los gestores comerciales y funciones para conexiones ODBC con bases de datos en sistemas Microsoft.
- Funciones de gestión de directorios y archivos, incluso para transferirse mediante el protocolo FTP.
- Funciones para el manejo de imágenes y bibliotecas de funciones gráficas.
- Funciones de generación y lectura de cookies.
- Funciones para la generación de documentos PDF.

Sumando además la innumerable cantidad de funciones predefinidas en PHP y todas aquellas propias de cada programador, que pueden ser reutilizadas e intercambiadas a través de foros específicos con otros programadores.

## **Desarrollo en la Tesis**

El sistema se desarrolló en PHP ya que es un lenguaje específico para desarrollar páginas web además puede interactuar con distintas bases de datos y funcionar con diferentes plataformas utilizando el mismo código fuente, pudiendo ser compilado y ejecutado en diversos sistemas operativos, lo que lo transforma en una herramienta eficiente y flexible.

Un programa escrito en PHP se ejecuta desde un cliente web de cualquier plataforma; donde el usuario puede escoger un determinado sistema operativo y su cliente web preferido, sin mayor problema ya que se puede acceder al programa desde cualquier nodo de una red.

### **4.2. SQL**

SQL (Structured Query Language) es un lenguaje estandarizado y *de facto* usado para el manejo de bases de datos.

El surgimiento de SQL está ligado al desarrollo de las bases de datos relacionales, cuyo concepto surgió en un artículo publicado en 1970 por el Doctor E. F. Codd, un investigador de IBM.

#### **4.2.1 Estándares ANSI / ISO**

El estándar oficial adoptado por el Instituto de Estándares Nacionales de América (ANSI, por sus siglas en inglés) y por la Organización de Estándares Internacionales (ISO) es al que comúnmente se le llama el “estándar SQL”.

El primer estándar, surgido en 1989, es el que comúnmente se denomina como estándar “SQL-89” o estándar “SQL1”.

En 1992 surgió el estándar SQL2 (oficialmente nombrado estándar SQL92), el cuál definió tres niveles de observación:

El primer nivel añadió un mínimo de utilidades al estándar SQL-89.

El nivel intermedio representó un paso adelante del estándar SQL-89, evitando la dependencia de las marcas de DBMS's.

El tercer nivel requiere una implementación total de las utilidades de SQL2.

A pesar del estándar SQL2, no existe un producto comercial de SQL que implemente todas sus características, ni existen dos productos de SQL que soporten el mismo dialecto SQL.

Con los fabricantes de software diversificando las extensiones de SQL, el desarrollo del estándar SQL3 ha tomado menor importancia comercial que el de sus predecesores.

El estándar “real” de SQL es aquel que se implementa en los productos aceptados por el mercado, mientras los programadores y usuarios tienden a trabajar con las partes del lenguaje que sean similares en diversos productos.

### **Desarrollo en la Tesis**

El uso del estándar ANSI / ISO de SQL en el manejo de datos sobre el DBMS tales como inserción, actualización, consulta además de múltiples vistas de los datos, garantiza su portabilidad en cualquier tipo de plataforma.

### **4.3. PostgreSQL**

Es un sistema gestor de bases de datos relacionales de objeto de código abierto. Ofrece características como las consultas complejas, las claves externas, los disparadores, las vistas, la integridad transaccional y el control de concurrencia de varias versiones. Los usuarios pueden ampliar PostgreSQL con tipos de datos, funciones, operadores y métodos de indexación nuevos. Trabaja con gran variedad de lenguajes de programación: C, C++, Java, Perl, Tcl, PHP y Python. Es un sistema de bases de datos relacionales de código abierto, su licencia es BSD, que concede libre cargo permiso para el uso, modificación y distribución del código y de la documentación de PostgreSQL con cualquier propósito.

Se puede ejecutar bajo prácticamente todos los sistemas operativos tipo UNIX, incluidos Linux y OS X para Apple de Macintosh. También puede ejecutarse bajo Microsoft Windows en el entorno Cygwin, el cual proporciona emulación de Linux bajo Windows.

Incluye soporte para varios tipos no normalizados, que resultan útiles para dominios de aplicaciones concretas.

### **4.3.1 Características de PostgreSQL**

Soporta las restricciones y los disparadores de SQL además reglas de reescritura de consultas que pueden declararse en el servidor.

PostgreSQL almacena la información sobre las bases de datos, las tablas, las columnas, tipos de datos, las funciones, los métodos de acceso, etc., en los catálogos del sistema, que se presentan al usuario como tablas normales.

Permite que los usuarios definan tipos compuestos, así como que amplíen los tipos básicos disponibles. Además permite que los usuarios definan funciones que se almacenen y ejecuten en el servidor. Soporta la sobrecarga de funciones, se pueden declarar funciones que utilicen el mismo nombre pero con argumentos de tipos diferentes. Éstas pueden escribirse como instrucciones de SQL.

PostgreSQL es el gestor de bases de datos de código abierto más avanzado hoy en día, ofreciendo control de concurrencia multi-versión, soportando casi toda la sintaxis SQL (incluyendo subconsultas, transacciones, tipos y funciones definidas por el usuario), contando también con un amplio conjunto de enlaces con lenguajes de programación (incluyendo PHP, C, C++, Java, perl, tcl y python).

### **Desarrollo en la Tesis**

Se trabajó en PostgreSQL como DBMS debido a que trabaja bajo diferentes plataformas, su código es extensible y posee herramientas de diseño para administrar y diseñar las bases de datos. De la misma forma las características que ofrece PostgreSQL, tales como funciones, secuencias, relaciones, tipos de datos definidos por el programador así como vistas lo hacen un robusto DBMS para el uso del sistema propuesto.

## **4.4 Javascript**

JavaScript es un lenguaje interpretado basado en scripts que son integrados directamente en el código HTML. El código es transferido al cliente para que este lo interprete al cargar la página. Fue creado por Netscape específicamente para su uso en el desarrollo de sitios web.

### **4.4.1 Características de Javascript**

- Es un lenguaje interpretado.
- No necesita compilación.

- Multiplataforma.
- Lenguaje de alto nivel
- Admite programación estructurada.
- Basado en objetos.
- Maneja la mayoría de los eventos que se pueden producir sobre la página web.
- No se necesita ningún kit o entorno de desarrollo.

### **Desarrollo en la Tesis**

JavaScript se utilizó como herramienta para consumir menos recursos del servidor además de ofrecer una interacción entre el usuario y el sistema.

### **4.5. HTML**

Para publicar información y distribuirla globalmente, se necesita un lenguaje entendido universalmente, una especie de lengua franca de publicación que todas las computadoras puedan comprender potencialmente. El lenguaje de publicación usado por la World Wide Web es el HTML (acrónimo de HyperText Markup Language, Lenguaje para el Formato de Documentos de Hipertexto).

HTML es un lenguaje de descripción de hipertexto compuesto por una serie de comandos o etiquetas, también denominadas “tags” que permiten definir la estructura lógica de un documento web y establecer los atributos del mismo.

### **Desarrollo en la Tesis**

Se utilizó HTML 4.01 para especificar la estructura lógica de su contenido: textos, tablas y encabezados, así como diferentes aplicaciones, entre las que destacan el diseño de formularios para realizar consultas a la base de datos y establecer enlaces con otras páginas. Debido a que se utilizó el formato estándar HTML 4.01 se asegura que las páginas serán capaces de ser leídas con cualquier cliente web.

Las ventajas que ofrece HTML 4.01, como son su alta capacidad para soportar el uso de hojas de estilo, la ejecución de scripts y el manejo de formularios lo convierten en una herramienta para el desarrollo del sistema.

## 4.6. CSS

Las Hojas de Estilo en Cascada (Cascading Style Sheets), son un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

### **Desarrollo en la Tesis**

Al diseñar y desarrollar las vistas de cada módulo del sistema, se contempló primero la estructura de la información de cada una de ellas y posteriormente se planteó el formato que llevaría utilizando las Hojas de Estilo en Cascada (CSS). Se recurrió a las CSS ya que tienen una estructura simple, flexible y potente, definiendo así la apariencia de cada elemento, lo cual permite modificar la presentación del diseño del Sistema posteriormente si es necesario, sin tener que modificar el código.

# Capítulo 5 Implementación

La instalación de Apache, PHP y PostgreSQL en un entorno Linux se lleva a cabo conectándose al servidor como usuario root (administrador), salvo que se indique lo contrario.

## 5.1 Instalación de Apache

El software de la última versión del servidor Apache se descarga en el sitio <http://httpd.apache.org/download.cgi>

Es necesario extraer el código fuente del archivo descargado (con extensión .tar.gz) utilizando los comandos enunciados a continuación (las letras *NN* hacen referencia a la versión descargada) :

```
$ gzip -d httpd-2_0_ NN.tar.gz  
$ tar xvf httpd-2_0_ NN.tar
```

El resultado es un nuevo directorio (dentro del actual), al cual es preciso moverse para realizar la compilación.

Se localiza el script llamado *configure* para poder configurar la estructura de directorios.

En este script la opción `--prefix` indica el directorio en el que Apache será instalado tras la compilación. Posteriormente Apache tendrá que ser configurado, empleando dicho directorio, para poder funcionar de forma adecuada.

Para configurar la estructura de Apache, usando las opciones por default, se debe ejecutar el script *configure* :

```
$ ./configure
```

Una vez ejecutado, se comprobarán características o funcionalidades disponibles en el sistema y se crearán *Makefiles*, los cuales se usarán para compilar el servidor, esta acción tomará algunos minutos.

El siguiente paso es la compilación de módulos del paquete Apache, empleando el comando *make*:

```
$ make
```

Dependiendo del hardware del equipo en que se lleve a cabo la compilación, y del número de módulos seleccionados, el tiempo que toma la compilación es variable.

El paquete quedará finalmente instalado en el directorio especificado en el script *configure* tras ejecutarse el siguiente comando:

```
$ make install
```

Al finalizar debe personalizarse el servidor editando el archivo *httpd.conf*, entendiendo como *PREFIX* el directorio en donde se ha instalado Apache (su valor por defecto es */usr/local/apache2* ).

```
$ vi PREFIX/conf/httpd.conf
```

Para cerciorarse que el servidor Apache puede ponerse en funcionamiento se corre la sentencia:

```
$ PREFIX/bin/apachectl start
```

En un navegador web debe poder accederse a la dirección *http://localhost*

La ejecución del servidor se detiene con el comando siguiente:

```
$ PREFIX/bin/apachectl stop
```

## 5.2 Instalación de PHP

PHP debe descargarse del sitio <http://www.php.net/downloads.php>, donde se encuentra disponible para una gran variedad de sistemas operativos.

El siguiente paso es descomprimir el código de PHP, refiriéndose a la versión descargada de PHP con las letras *xxx*:

```
$ gunzip php-xxx.tar.gz
```

```
$ tar -xvf php-xxx.tar
```

La configuración de PHP se lleva a cabo empleando las siguientes sentencias:

```
$ ./configure --with-config-file-path=/etc  
--with-pdo-pgsql=/usr/local/pgsql/bin/pg_config  
--with-apxs2=/usr/local/apache2/bin/apxs2
```

Terminada la configuración de PHP, se necesita la creación de los módulos y/o ejecutables:

```
$ make  
$ make install
```

Se debe copiar el script `php.ini-recommended` a la misma dirección especificada en la opción `--with-config-file-path` durante la configuración:

```
$ cp php.ini-recommended /etc/php.ini
```

En el script `httpd.conf` (dentro del directorio de configuración de Apache), en la sección de `LoadModules` se debe agregar la siguiente línea (es posible que el comando `make install` pueda haberla agregado):

```
LoadModule php5_module modules/libphp5.so
```

Dentro del mismo script tendrá que buscarse o agregarse la línea siguiente a fin de que Apache sea capaz de analizar las extensiones `php` de los scripts como código PHP:

```
AddType application/x-httpd-php .php
```

De igual manera, en el mismo script, habrá que localizarse la opción `DirectoryIndex` y colocar el valor `index.php` al final de la línea:

```
DirectoryIndex index.html index.php
```

Se sugiere reiniciar el servidor Apache como ya se indicó anteriormente, y a fin de asegurarse que la instalación de PHP fue correcta, crear un script llamado `prueba.php` en el directorio `/usr/local/apach2/htdocs/`, el cual ha de contener las siguientes líneas:

```
<?php  
phpinfo();  
?>
```

Por último, acceder en un explorador web a la dirección `http://localhost/prueba.php`. El resultado es una pantalla con información de la configuración de PHP bajo Apache.

### **5.3 Instalación de PostgreSQL**

Lo primero es crear al usuario `postgres` con el comando `adduser`:

```
$ adduser postgres
```

El código fuente de PostgreSQL se obtiene en el sitio <http://www.postgresql.org/ftp/source>. Para propósitos de esta guía se emplea postgresql-8.1.5.tar.gz, no obstante se puede descargar la versión más actualizada.

Se debe descomprimir el código fuente descargado con la siguiente instrucción:

```
$ tar -zxvf postgresql-8.1.5.tar.gz
```

Posteriormente deberá cambiar de directorio de trabajo:

```
$ cd postgresql-8.1.5
```

Una vez que se encuentre en el nuevo directorio se procede a configurar PostgreSQL para instalarlo en el directorio /usr/local/pgsql:

```
$ ./configure --prefix=/usr/local/pgsql
```

La compilación se realiza con el comando make, como se indica a continuación:

```
$ make install
```

Por razones de seguridad, PostgreSQL no tiene permitido al administrador root iniciar el servidor, por ello hay que cambiar los archivos de PostgreSQL para que pasen a ser propiedad del usuario postgres:

```
$ chown -R postgres:postgres /usr/local/pgsql
```

El paso siguiente implica cambiar de usuario y conectarse como el usuario postgres, cambiando también de directorio de trabajo a /usr/local/pgsql:

```
$ cd /usr/local/pgsql
```

Tras realizar el cambio es necesario iniciar un cluster de base de datos empleando la siguiente sentencia:

```
$ bin/initdb -D ./data
```

Finalmente, se inicia el DBMS PostgreSQL con la sentencia que se enuncia a continuación:

```
$ bin/pg_ctl -D ./data -l data/logfile start
```

Entre las sentencias útiles a partir de este momento, se encuentra la siguiente para crear una base de datos especificando su nombre como basededatos:

```
$ /usr/local/pgsql/bin/createdb -owner=usuario basededatos
```

Otra de las sentencias que se pueden emplear es la siguiente, para crear a un usuario:

```
$ /usr/local/pgsql/bin/createuser
```

## **5.4 Base de datos ARQESCOLAR**

Una vez instalado PHP, APACHE y PostgreSQL, se crea la base de datos ARQESCOLAR, la cual alberga la estructura mostrada en el DER ARQESCOLAR (ver figura 3.6):

```
=# CREATE DATABASE ARQESCOLAR with encoding 'UTF8';
```

De acuerdo al análisis de las características que se requieren para el funcionamiento del Sistema se determina la estructura de las tablas que conforman la base de datos, mediante un Modelo Entidad-Relación, el cual se basa en la teoría de la normalización.

La información necesaria para insertar datos en ARQESCOLAR es proporcionada por personal de Asuntos Escolares de la Facultad de Arquitectura, la cual, en diferentes formatos, fue procesada hasta obtener la estructura apropiada para el llenado de tablas.

Posteriormente se deben crear las tablas indicadas en el DER:

Ejemplo: Creación de la tabla aulas

```
CREATE TABLE aulas (  
  cla_aul character varying(4) NOT NULL,  
  nom_aul character varying(30)  
);
```

Resulta útil indicar que es posible insertar datos en una tabla desde un archivo de texto, para ello es necesario subirlo al servidor y ejecutar el comando copy desde el prompt psql, indicando la ruta absoluta del archivo y un delimitador:

```
argescolar=# copy AULAS from '/tesis/nuevoaul.txt' with  
delimiter ';';COPY
```

En donde el contenido del archivo de texto aulas.txt es el siguiente:

```
1P ;ENRIQUE YAÑEZ  
2P ;PEPITA SAISO  
3P ;HONORATO CARRASCO  
5P ;ENRIQUE DEL MORAL  
6P ;MARIO PANI  
7P ;FRANCISCO CENTENO  
A.P.;ARQUITECTURA DE PAISAJE  
A001;A001  
A002;A002  
A003;A003
```

```

A004;A004
A005;A005
A101;A101
A102;A102
A103;COORDINACION VILLAGRAN G
A104;A104 SALA DE MAESTROS
A105;COORDINACION DOMINGO G.R
A201;A201
A202;A202
A203;C.C. DOMINGO G. R.
A204;A204
B001;B001
B002;B002
B004;B004
B005;B005
B006;B006
B101;B101
B102;B102
B103;B103
B104;B104
B105;COORDINACION MAX CETTO
B106;B106 AUDITORIO
B107;B107 C.C. MAX CETTO
B201;B201 MAQUETAS MAX CETTO
B202;B202

```

Otra manera muy utilizada para crear y/o poblar las tablas es usando archivos que contienen sentencias SQL, dentro del prompt de psql se cuenta con el comando \i:

```
arquescolar=# \i etapas.txt
```

En donde el contenido del archivo de texto etapas.txt es el siguiente

```

CREATE TABLE etapas (
    cla_eta integer NOT NULL,
    cla_car character varying(4) NOT NULL,
    num_asi integer NOT NULL,
    nom_eta character varying(20) NOT NULL
);

```

En la siguiente instrucción se muestra la manera de poblar la tabla etapas haciendo uso del comando \i en el prompt de psql:

```
arquescolar=# \i etapas.txt
```

Donde el contenido del archivo etapas.txt es el que se muestra a continuación:

```

INSERT INTO etapas VALUES (1, '102', 10, 'Básica');
INSERT INTO etapas VALUES (2, '102', 14, 'Desarrollo');
INSERT INTO etapas VALUES (3, '102', 13, 'Profundización');

```

```
INSERT INTO etapas VALUES (4, '102', 4, 'Consolidación');  
INSERT INTO etapas VALUES (5, '102', 2, 'Demostración');  
INSERT INTO etapas VALUES (6, '102', 20, 'Selectivas');
```

Para mayor información consultar el anexo E Diccionario de datos

# 6. Pruebas y Mantenimiento

## 6.1. Mantenimiento al sistema

Para realizar el mantenimiento de la vista o de la lógica interna de los módulos que componen el sistema, es necesaria la modificación los scripts que lo integran, los cuales constan de archivos php, javascript y css. La persona encargada de realizar esta tarea debe poseer conocimientos de programación en PHP, sentencias de SQL, manejo del DBMS PostgreSQL, etiquetas HTML y CSS.

A fin de facilitar esa tarea, a continuación se muestran los módulos que componen el sistema y se especifican cuáles scripts manejan la vista y la lógica de los mismos.

### 6.1.1. Alumno

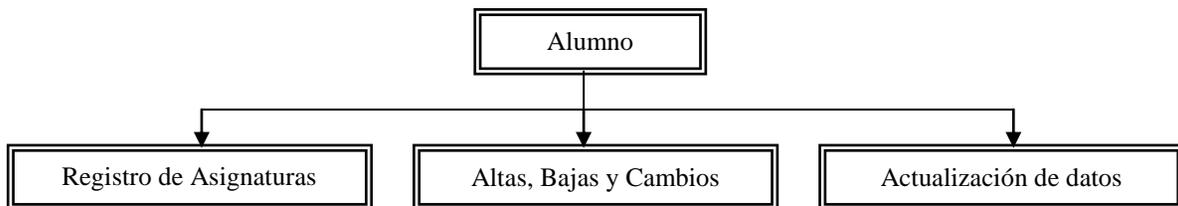


Figura 6. 1 Módulos del usuario Alumno

### Establecer sesión de usuario Alumno

Ingreso a Trámites Escolares - Mozilla Firefox  
http://132.248.43.61/tesis/Alumno/IngresoTramitesEscolaresAlumno.php

Proporcione sus datos para ingresar

**Introduzca los datos solicitados**

Login:

Password:

Ingresar

Terminado

Figura 6. 2 Pantalla para ingresar login y password para el usuario Alumno

Vista	Lógica
IngresoTramitesEscolaresAlumno	autenticAlumno.php
	redirección.php

Tabla 6. 1

**Registro de Asignaturas.** Módulo que realiza el proceso de registro de peticiones de asignaturas de un Alumno.



Figura 6. 3 Pantalla que despliega las asignaturas registradas por el Alumno

Vista	Lógica
registroAlumno.php	registroAlumno.php
	registroAlumno2.php
	bajaRegistroAlumno.php

Tabla 6. 2

**Altas, Bajas y Cambios.** Módulo cuyo objetivo es el de realizar movimientos de altas, bajas y/o cambios de cursos (horario de asignaturas para semestre en curso) de un Alumno.

Inscripción de asignaturas - Mozilla Firefox  
 http://132.248.43.61/tesis/Alumno/ABCAlumno.php

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
 FACULTAD DE ARQUITECTURA

Alumno: **BONILLA GARCIA JOSE LUIS** Número de cuenta: **301153689**  
 Carrera: Arquitectura Clave: 102; Etapa mínima:0 Etapa límite:2

Clave de la asignatura:  Grupo:

Asignaturas registradas													
Clave	Asignatura	Grupo	Profesor	Aula	Créditos	Horario					Hora inicio	Hora fin	
						L	M	M	J	V			S
1130	INTRODUCCION HISTORIC CRITIC	5131	Ver profesores	H201	4			*				09:00	11:00
1131	TEORIA DE ARQUITECTURA I	5121	Ver profesores	F101	4							00:00	00:00
1233	MATEMATICAS APLICADAS II	5232	Ver profesores	C201	4				*			18:00	20:00
1333	TALLER ARQUITECTURA III	5316	Ver profesores	C102	25			*				07:00	15:00
										*			07:00
1334	INSTALACIONES I	5331	Ver profesores	C101	4		*					09:00	14:00
								*				07:00	09:00
1134	SISTEMAS ESTRUCTURALES I	5129	Ver profesores	C102	6		*					09:00	11:00
1731	ADMINISTRACION III	5728	Ver profesores	G203	4			*				17:00	19:00

Errores ocurridos durante la asignación de horarios

Asignatura ya ha sido acreditada. En la asignatura 1133:MATEMATICAS APLICADAS I

No cumple con seriación por asignatura. En la asignatura 1234:SISTEMAS ESTRUCTURALES II

No cumple con seriación por asignatura. En la asignatura 1230:ARQUITECTURA EN MEXICO S.XX

No cumple con seriación por asignatura. En la asignatura 1231:TEORIA DE LA ARQUITECTURA II

Terminado

Figura 6. 4 Pantalla que muestra los errores ocurridos en el proceso de Asignación

Vista	Lógica
ABCAlumno.php	ABCAlumno.php
	ABCAlumno2.php
	ABCAlumno3.php

Tabla 6. 3

**Actualización de datos.** Módulo que corresponde a la actualización de información personal de un Alumno.

Vista	Lógica
ActDatAlu.php	ActDatAlu.php
	ActDatAlu2.php

Tabla 6. 4

### 6.1.2. Actor Mantenimiento

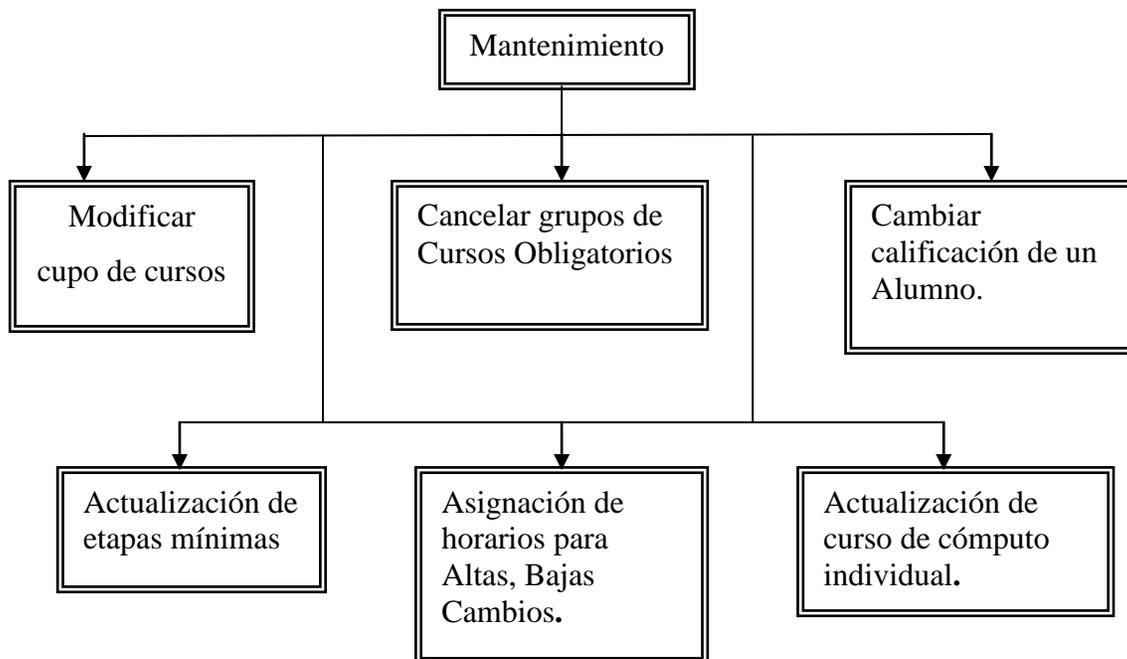


Figura 6. 5 Módulos del usuario Mantenimiento

**Asignación de grupos.** Módulo cuya función es la de inscribir a los Alumnos en las asignaturas y grupos, tomando como referencia los resultados de la etapa de Registro.

Vista	Lógica
opciones.php	AsiGru.php
AsiGru.php	

Tabla 6. 5

**Modificar cupo de cursos.** Módulo cuya función es la de modificar el cupo de algún grupo correspondiente a una asignatura interna de la Facultad de Arquitectura.

Vista	Lógica
opciones.php	ModCup.php
ModCup.php	SelAsi.php
SelAsi.php	SelGru.php
SelGru.php	ConCup.php
ConCup.php	ActCup.php
ActCup.php	

Tabla 6. 6

**Cancelar grupos de Cursos Obligatorios.** Módulo correspondiente a la cancelación de grupos de asignaturas obligatorias que se encuentren activos durante el semestre en curso.

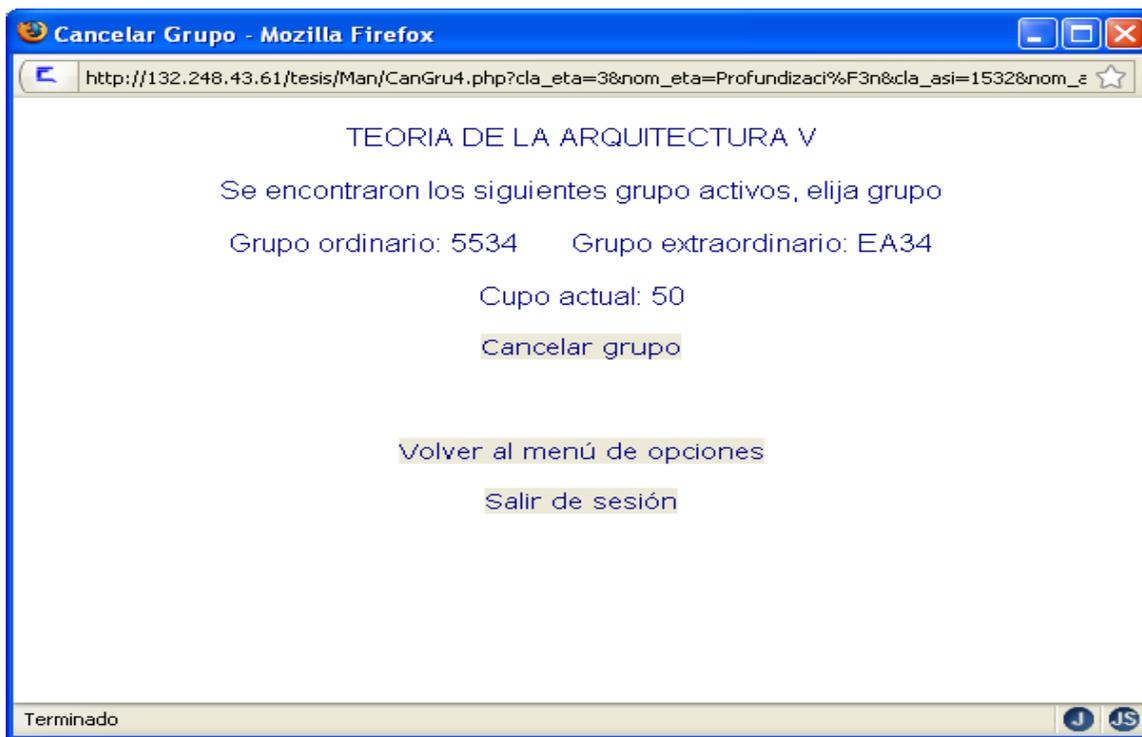


Figura 6. 6 Pantalla que permite cancelar el grupo de la asignatura seleccionada

Vista	Lógica
opciones.php	CanGru2.php
CanGru.php	CanGru3.php
CanGru.2php	CanGru4.php
CanGru.3php	CanGru5.php
CanGru.4php	
ActCup.php	

Tabla 6. 7

**Cambiar calificación de un Alumno.** Módulo cuya función es la de modificar la calificación de un Alumno en alguna asignatura que haya cursado con anterioridad.

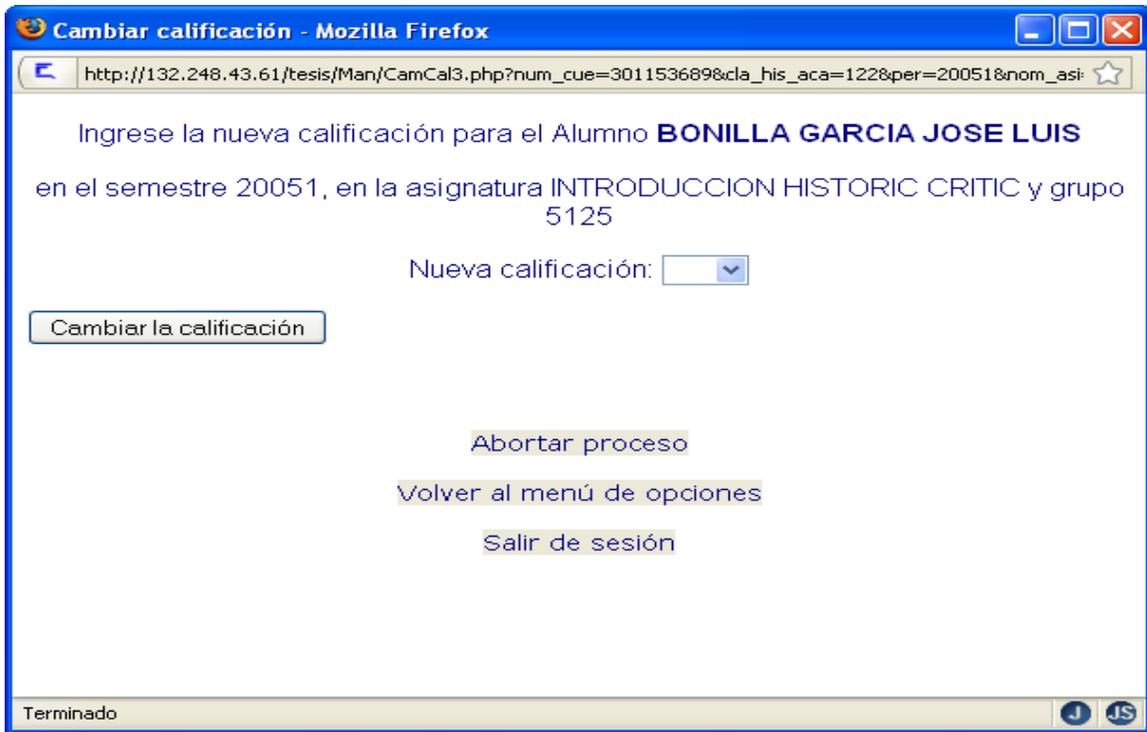


Figura 6. 7 Pantalla para ingresar una nueva calificación para un Alumno

Vista	Lógica
opciones.php	CamCal2.php
CamCal.php	CamCal4.php
CamCal2.php	
CamCal3.php	
CamCal4.php	

Tabla 6. 8

**Actualización de etapas mínimas.** Módulo que realiza la actualización del avance de los créditos de un Alumno.

Vista	Lógica
opciones.php	ActEtaMinTot.php
ActEtaMinTot.php	

Tabla 6. 9

**Asignación de horarios para Altas, Bajas Cambios.** Módulo cuya función es establecer un horario para realizar Altas, Bajas y Cambios a cada Alumno que haya realizado el registro de asignaturas.

Vista	Lógica
opciones	AsiHorABC.php
AsiHorABC.php	AsiHorABC2.php
AsiHorABC2.php	

Tabla 6. 10

**Actualización de curso de cómputo individual.** Módulo que realiza la actualización del requisito de curso de cómputo para el Alumno que así lo requiera.

Vista	Lógica
opciones.php	ActCurComInd2.php
ActCurComInd.php	
ActCurComInd2.php	

Tabla 6. 11

### 6.1.3. Coordinador de Taller

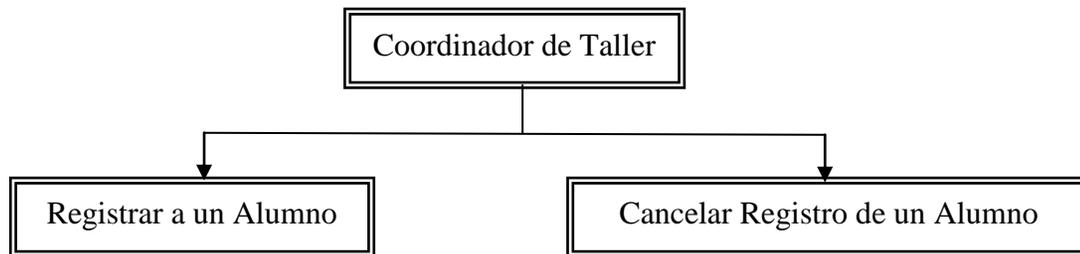


Figura 6. 8 Módulo del usuario Coordinador de Taller

**Registrar a un Alumno.** Módulo cuya función es la de inscribir a un Alumno a un determinado Taller.

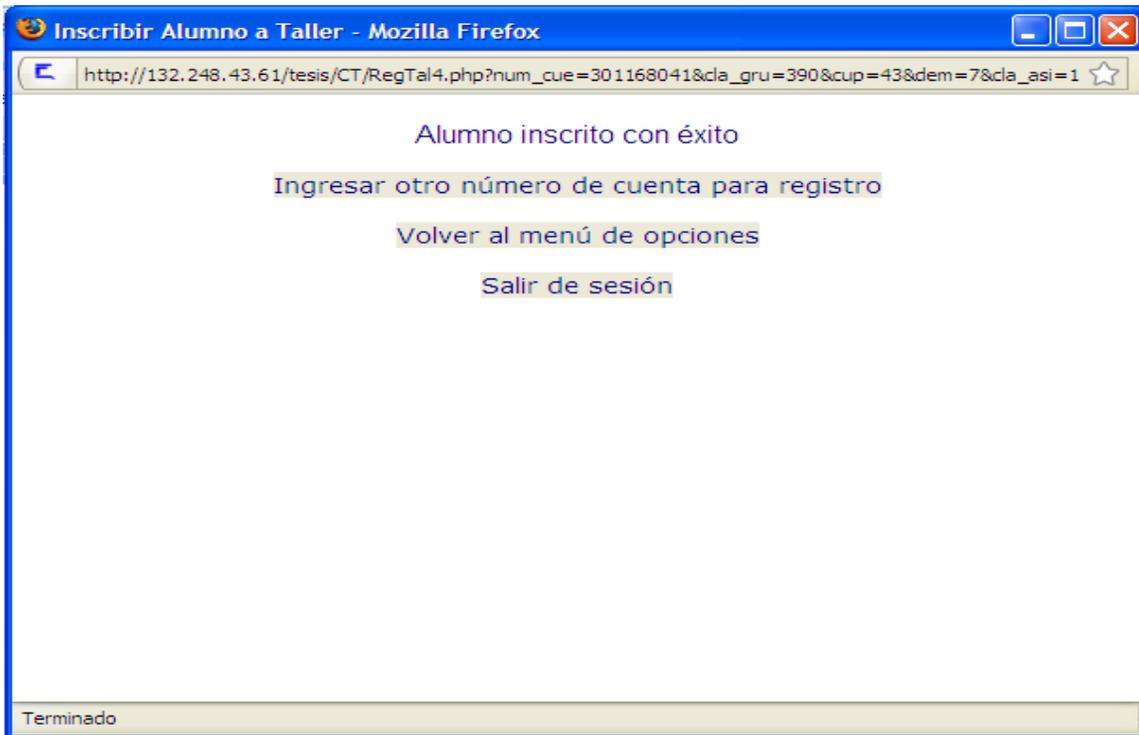


Figura 6. 9 Pantalla que muestra que un Alumno ha sido inscrito a un Taller

Vista	Lógica
opciones.php	RegTal2.php
RegTal.php	RegTal3.php
RegTal2.php	RegTal4.php
RegTal3.php	
RegTal4.php	

Tabla 6. 12

**Cancelar Registro de un Alumno.** Módulo que se encarga de cancelar el registro de un Alumno en un Taller específico.

Vista	Lógica
opciones.php	CanTal2.php
CanTal.php	CanTal3.php
CanTal2.php	
CanTal3.php	

Tabla 6. 13

#### 6.1.4. Coordinador de Carrera

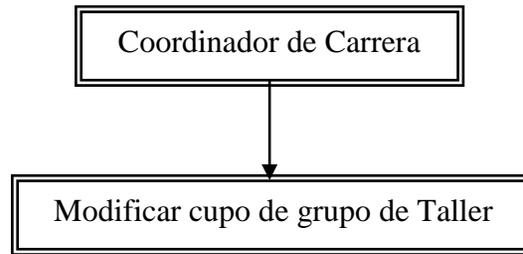


Figura 6. 10 Módulo del usuario Coordinador de Carrera

**Modificar cupo de grupo de Taller.** Módulo cuya función es la de cambiar el cupo de algún grupo correspondiente a uno de los Talleres de la carrera de Arquitectura en sus diferentes semestres.

Vista	Lógica
opciones.php	opciones.php
SemTal.php	SemTal.php
SemTal.php	SemTal.php
SemTal.php	SemTal.php
ActCupTal.php	ActCupTal.php

Tabla 6. 14

#### 6.1.5. Jefe de Asuntos Escolares

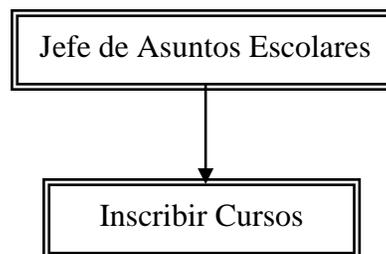


Figura 6. 11 Módulo del usuario Jefe de Asuntos Escolares

**Inscribir Cursos.** Módulo que realiza inscripciones de un Alumno a las diferentes asignaturas obligatorias o selectivas internas de la Facultad de Arquitectura de manera extemporánea. De manera opcional puede consultarse el Historial Académico del Alumno.

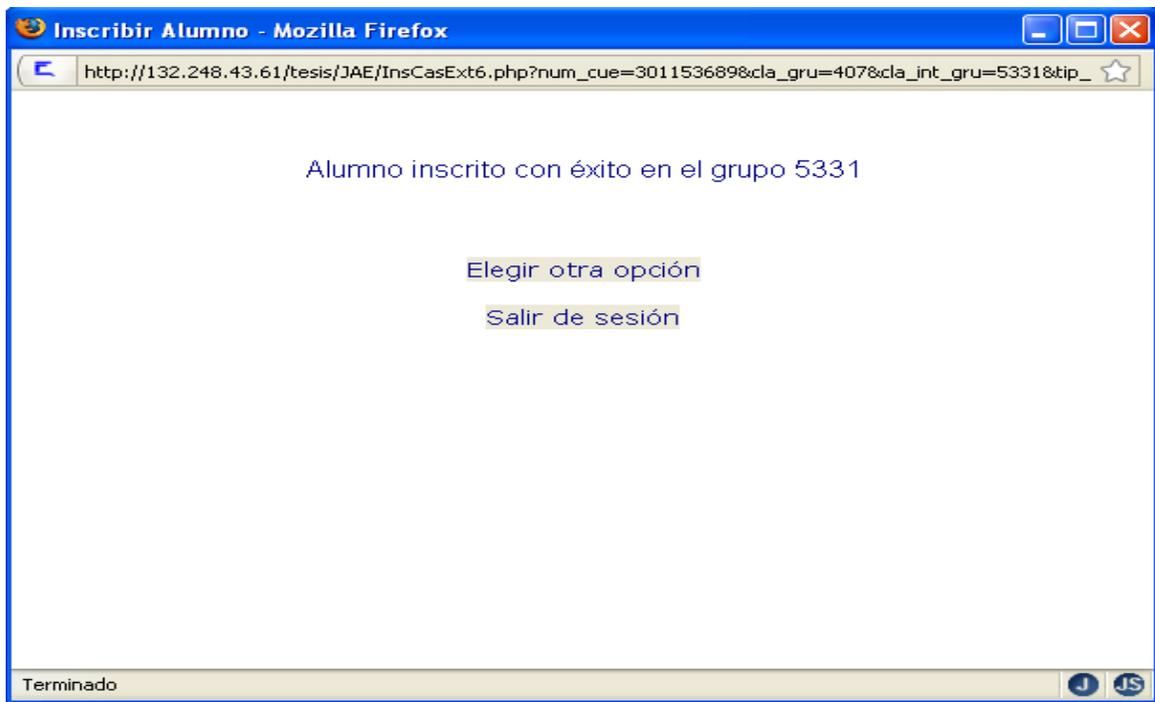


Figura 6. 12 Pantalla que muestra que un Alumno ha sido inscrito en una asignatura

Vista	Lógica
opciones.php	InsCasExt2.php
InsCasExt.php	InsCasExt3.php
InsCasExt2.php	InsCasExt4.php
InsCasExt3.php	InsCasExt5.php
InsCasExt4.php	InsCasExt6.php
InsCasExt5.php	
InsCasExt6.php	

Tabla 6. 15

**Consultar el Historial Académico.** Módulo cuya función es consultar la historia académica completa de un Alumno.

Vista	Lógica
opciones.php	InsCasExt2.php
InsCasExt.php	ConHisAcaAlu.php
InsCasExt2.php	
ConHisAcaAlu.php	

Tabla 6. 16

## 6.2. Código

El código fuente de cada módulo del Sistema está basado en programación estructurada, con llamadas a funciones, las cuales están contenidas en las utilerías llamadas `libreria.php` y `libreriaAlumno.php`

Se empleó reutilización de código a fin de hacer un producto con menor carga para el servidor.

Con respecto a la vista, el archivo `index1.css` contiene la hoja de estilo externa, la cual se encarga de establecer el formato y apariencia de cada interfaz del sistema.

Respecto a la lógica de cada módulo del sistema es necesario consultar el capítulo 2 “Análisis”, en donde se describe el proceso que sigue cada elemento del mismo, así como consultar el capítulo 3 “Diseño”, el cual contiene diagramas UML que sirven de apoyo para visualizar el flujo, intercambio e interacción de cada uno de los elementos del Sistema.

Teniendo en cuenta la velocidad a la que cambian o se actualizan las tecnologías informáticas empleadas en el sistema se facilitan las direcciones en internet de sus sitios oficiales, a fin de obtener la documentación, especificaciones y recursos más recientes:

PostgreSQL	<a href="http://www.postgresql.org/">http://www.postgresql.org/</a>
PHP	<a href="http://www.php.net/">http://www.php.net/</a>
HTML	<a href="http://www.w3.org/html/">http://www.w3.org/html/</a>
W3C	<a href="http://www.w3.org/">http://www.w3.org/</a>

A fin de hacer más comprensible la lógica del sistema se siguieron algunos estándares propios en las nomenclaturas.

### 6.2.1 Estándares utilizados en la nomenclatura de las funciones

Se nombraron las funciones de acuerdo a expresiones que describen la actividad que realizan, empleando las tres primeras letras de cada palabra, comenzando la primera letra de cada palabra con mayúscula.

En el caso de funciones que tienen interacción con la base de datos se empleó una abreviatura del actor de asuntos escolares que ejecutaba dicha función.

Actor	Abreviatura
Coordinador de Carrera	CC
Coordinador de Taller	CT
Jefe de Asuntos Escolares	JAE
Mantenimiento	Man

Tabla 6. 17

En el caso de algunas funciones muy generales, compartidas por más de un actor, éstas se nombraron sin agregar la abreviatura de ningún actor, por ejemplo la función que se conecta al servidor.

Ejemplo:

La función que se encarga de consultar la historia académica del Alumno y que es ejecutada por el actor Mantenimiento se nombró: ManConHisAcaAlu()

### **6.2.2 Estándares utilizados en la nomenclatura para scripts de PHP**

Se nombraron los scripts de acuerdo a expresiones que describen la actividad que realizan, empleando las tres primeras letras de cada palabra, comenzando la primera letra de cada palabra con mayúscula.

En el caso de que una actividad de un módulo fuese realizada por más de un script se finalizó el nombre de los sucesivos scripts con una secuencia numérica que comenzó con el número 2.

Ejemplos:

1. El script que realiza la asignación de asignatura se nombró AsiAlu.php
2. Los scripts que se encargan de realizar inscripciones de casos extemporáneos se nombraron InsCasExt.php, InsCasExt2.php y así sucesivamente.

A fin de ejemplificar lo anteriormente explicado, se muestra un ejemplo de código en un script php:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Consultar Historia Académica de Alumno</title>
<link href=" ../index.css" rel="stylesheet" type="text/css">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>
<body>
<?php
if(isset($_GET["num_cue"]))
{
    include("../libreria.php");
    $num_cue=$_GET['num_cue'];
    $conecta=conexionServer();
    if($conecta!=1)
    { // si no hubo conexion al servidor
        echo "falló; la conexión al servidor";
    }
    else if($conecta==1)
    { //hubo conexión

        $num_resultados=JAECOnHisAcaComAlu($num_cue);
        //print_r(array_values($num_resultados));
        $resultados=$num_resultados[0][0];

        if($resultados>0) //revisa que se encontraran resultados
        {
            echo "<table border='1'>";
            echo
            "<tr><td>Asignatura</td><td>Clave</td><td>Periodo</td><td>Calificación</td><td>Tipo
de examen</td></tr>";
            for($i=1;$i<=$resultados;$i++)
            {
                $nom_asi=$num_resultados[$i][0];
                $cla_asi=$num_resultados[$i][1];
                $per=$num_resultados[$i][2];
                $scal=$num_resultados[$i][3];
                $tip_exa=$num_resultados[$i][4];
                echo "<tr><td>$nom_asi</td> <td>$cla_asi</td> <td>$per</td> <td>$scal</td>
<td>$tip_exa</td></tr>";
            } //fin for
            echo "</table>";

            } //fin se encontraron resultados
            else
            { //escenario alternativo, si no se encontraron resultados
                echo "<p align='center'>No se encontró registro del Alumno con
número de cuenta $num_cue.</p>";
            } //fin escenario alternativo, si no se encontraron resultados
            pg_close();
        } //fin hubo conexión

    }
    else
    {
        echo "<br><br><p align='center'>No se recibió; el número de cuenta</p>";
    }
    ?>
<p align='center'><a href="" onclick='window.close();'> Cerrar ventana</a></p>
</body>
</html>

```

## **6.3 Casos de prueba**

### **Ingresar a Trámites Académicos**

#### **Funcionalidad o características:**

Al proporcionar los datos correctos se otorga acceso al módulo de opciones correspondientes a cada usuario.

#### **Secuencia:**

- El usuario ingresa sus datos.
- El usuario elige “Ingresar”.
- Se muestra el menú de opciones de acuerdo al usuario que se haya autenticado.

#### **Prerrequisitos:**

- Login: (según el usuario)
- Password: (según el usuario).

#### **Resultados de la prueba:**

El usuario es autenticado y las opciones correspondientes al mismo son desplegadas.

#### **Evaluación de resultados:**

Se autentifica al usuario solamente cuando tanto el login y password son correctos, en caso de que alguno de estos sea erróneo el Sistema impide el acceso y solicita nuevamente ambos datos.

### **6.3.1 Alumno**

#### **Registro de Asignaturas**

#### **Funcionalidad o característica:**

El Alumno realiza su registro en los grupos y asignaturas de su preferencia.

#### **Secuencia:**

- Se muestran los cursos registrados previamente, junto con las opciones Alta, Baja, Imprimir Comprobante y Cerrar sesión.
- Se deben ingresar las claves de asignatura y grupo.
- Se elige la opción de “Alta/Baja” para registrarse o para darse de baja en el grupo de la asignatura proporcionada respectivamente.
- Finalmente, se muestra la información actualizada de todos los cursos y grupos en que el Alumno está registrado.

**Prerrequisitos:**

- El Alumno debió haberse autenticado.
- El proceso de Actualizar datos debe haber concluido con éxito previamente.

**Resultado:**

Se muestran las asignaturas registradas en semestre en curso.

**Evaluación de resultados:**

Se verifica que las claves de asignatura y grupo existan en la base de datos, además que el Alumno cumpla con requisitos de la asignatura si es que así lo requiere para la etapa de Registro. Ésta será desplegada en pantalla junto con las que ya tenga registradas. En caso de que el Alumno no cumpla con algún requisito para la asignatura se desplegará un mensaje informándole acerca del problema.

**Realizar Altas, Bajas y Cambios (tras la etapa de asignación de horarios)**

**Funcionalidad o característica:**

El Alumno puede realizar cambios en tiempo real en las asignaturas asignadas a partir de un horario establecido, realizando estos movimientos cuantas veces así lo requiera.

**Secuencia:**

- Se muestran la información de cursos y grupos en donde el Alumno está inscrito así como las opciones de Alta, Baja, Imprimir Comprobante y Cerrar sesión.

- Se ingresan las claves de asignatura y grupo.
- Se elige la opción de “Alta/Baja” para inscribirse o para darse de baja en el grupo de la asignatura proporcionada respectivamente.
- Finalmente se muestra la información actualizada de todos los cursos y grupos en que el Alumno está inscrito.

**Prerrequisitos:**

- El Alumno debió de haberse autenticado.

**Resultado:**

Las asignaturas inscritas y actualizadas para el semestre en curso son mostradas.

**Evaluación de resultados:**

Al realizarse el proceso de Alta o Baja se verifica que las claves de la asignatura y del grupo existan en la base de datos, además que el Alumno cumpla con requisitos de la asignatura si es que así lo requiere, ésta será desplegada en pantalla junto con las que ya tenga inscritas. En caso de que el Alumno no cumpla con algún requisito para la asignatura se despliega un mensaje informándole acerca del problema.

**Actualizar datos****Funcionalidad o característica:**

El Alumno tiene la posibilidad de actualizar sus datos personales.

**Secuencia:**

- Se muestran los datos de domicilio, teléfono y correo electrónico, distribuida en campos junto con la opción “Actualizar”.
- Se actualizan los datos del Alumno hasta que el Alumno hace clic en “Actualizar”.

**Prerrequisitos:**

- El Alumno debió de haberse autenticado.

**Resultado:**

La información personal del Alumno queda actualizada en la base de datos.

### Evaluación de resultados:

Tras elegir el botón “Actualizar” se realiza la actualización de la información del Alumno, el proceso tiene éxito cuando el sistema informa que se ha llevado a cabo la acción elegida.

Pantalla:

Actualizar datos de Alumno - Mozilla Firefox

http://132.248.43.61/tesis/Alumno/actualizaDatosAlumno.php

Debes actualizar tus datos para poder continuar

Alumno: BONILLA GARCIA JOSE LUIS      Número de cuenta: 301153689  
DOMICILIO

Calle  
AV. UNIVERSIDAD

Colonia  
COPILCO

Número exterior: 404      Número interior:

Delegación o Municipio COYOACÁN

Entidad Federativa DISTRITO FEDERAL

Código Postal 44545

Teléfono

Teléfono celular:  
044 154878898

E-mail |      Actualizar

Terminado

Figura 6. 13 Pantalla para ingresar información personal del Alumno

### 6.3.2 Coordinador de Taller

#### Inscribir Alumno a Taller.

##### Funcionalidad o característica:

Se inscribe un Alumno a un Taller de Arquitectura.

##### Secuencia:

- Se selecciona la opción “Inscripción a Taller”.
- Se ingresa el número de cuenta del Alumno.
- Se muestran datos del Alumno.
- Se elige la opción “datos son correctos”.
- Se elige opción de grupo.
- El Alumno queda finalmente inscrito.

**Prerrequisitos:**

El Coordinador de Taller debió de haberse autenticado.

**Resultados de la prueba:**

El Alumno quedó inscrito al grupo del Taller de Arquitectura escogido.

**Evaluación de resultados:**

Se busca que el alumno esté inscrito en un grupo del Taller, tras mostrar los datos del Alumno se elige la opción “Datos son correctos” y se realiza la inscripción. El proceso es exitoso cuando se muestra un mensaje informando que se ha llevado a cabo la inscripción.

Pantalla:

Tras haber seleccionado la opción “Datos son correctos”:

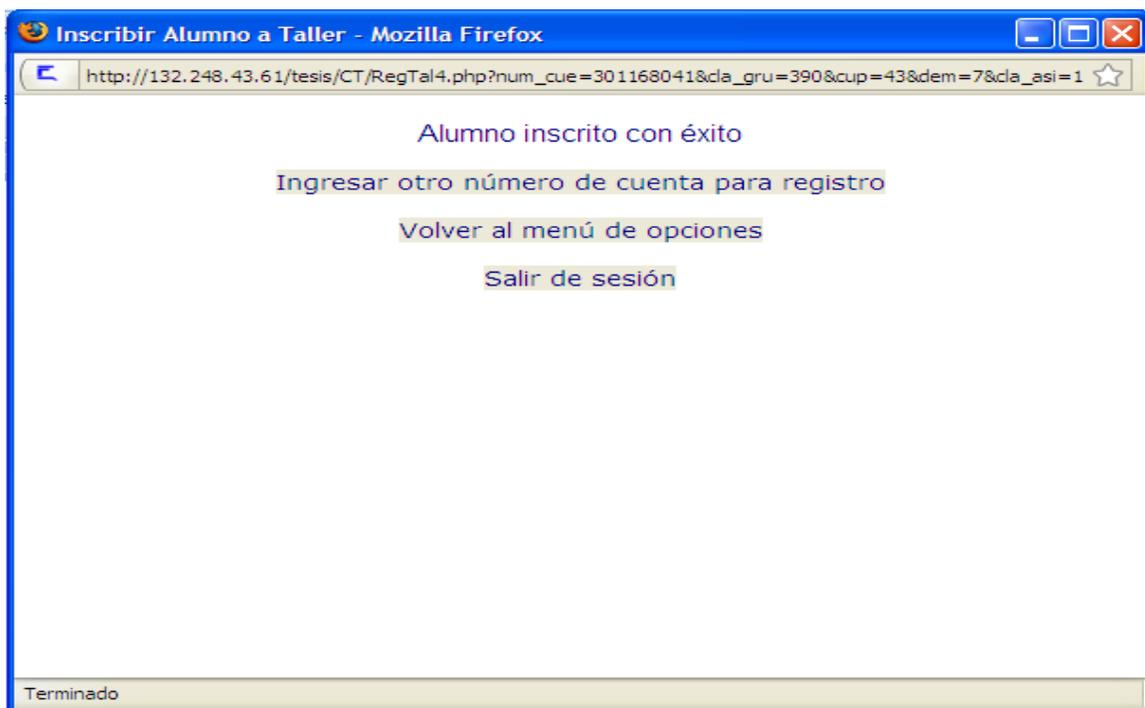


Figura 6. 14 Pantalla que muestra que el Alumno ha sido inscrito en un grupo de Taller

**Cancelar inscripción****Funcionalidad o característica:**

Se cancela la inscripción de un Alumno a un grupo de un Taller de Arquitectura.

**Secuencia:**

- El Coordinador de Taller selecciona la opción “Cancelar inscripción al Taller”.
- Se ingresa el número de cuenta del Alumno.
- Se muestran datos del Alumno.
- Se elige la opción “Datos son correctos”.
- La inscripción del Alumno a un grupo del Taller es cancelada.

#### **Prerrequisitos:**

- El Coordinador de Taller debió de haberse autenticado.

#### **Resultados de la prueba:**

Se cancela la inscripción del Alumno al grupo de Taller de Arquitectura (solo si fue realizada previamente y en el mismo Taller).

#### **Evaluación de resultados:**

En caso de encontrarse inscrito, se muestran los datos del Alumno y, como paso final, se elige “Datos correctos”, se inhabilita la inscripción y el proceso es exitoso cuando el proceso informa de la cancelación.

Pantalla:

Tras seleccionar la opción “Datos son correctos”.

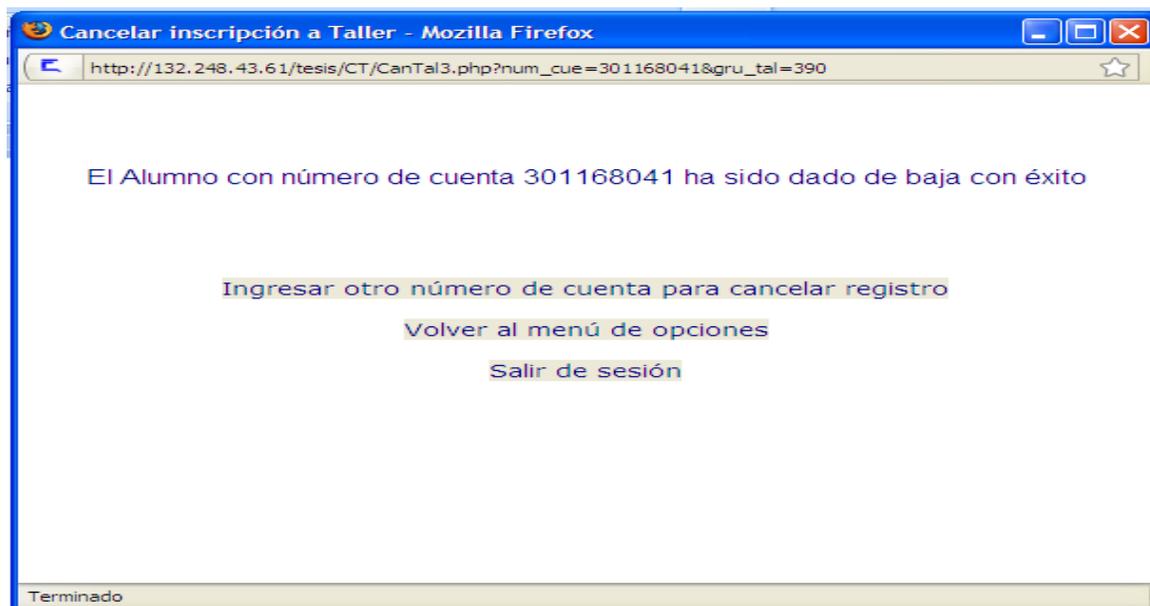


Figura 6. 15 Pantalla que muestra que el Alumno ha sido dado de baja de un grupo de Taller

### **6.3.3 Mantenimiento**

#### **Modificar cupo de cursos.**

##### **Funcionalidad o característica:**

Se modifica el cupo de un grupo de una asignatura obligatoria o selectiva interna.

##### **Secuencia:**

- Se selecciona la opción “Modificar cupo de cursos”.
- Se muestran las etapas de la carrera de Arquitectura, además de una adicional para los cursos de carácter selectivo interno.
- Se elige una opción de etapa.
  
- Se muestran las asignaturas correspondientes a la etapa seleccionada.
- Se elige una asignatura.
- Se muestran los grupos de la asignatura.
- Se elige un grupo.
- Se ingresa el nuevo cupo y se elige el botón “Actualizar cupo”.
- El cupo del grupo elegido se actualiza.

##### **Prerrequisitos:**

- El actor Mantenimiento debió de haberse autenticado.

##### **Resultados de la prueba:**

Se actualiza el cupo del grupo de la asignatura seleccionada.

##### **Evaluación de resultados:**

Tras ingresar el nuevo cupo y seleccionar la opción “Actualizar cupo” se reemplaza el cupo anterior, el proceso es exitoso cuando el sistema informa de la actualización.

##### **Pantalla:**

Tras seleccionar “Actualizar cupo”.

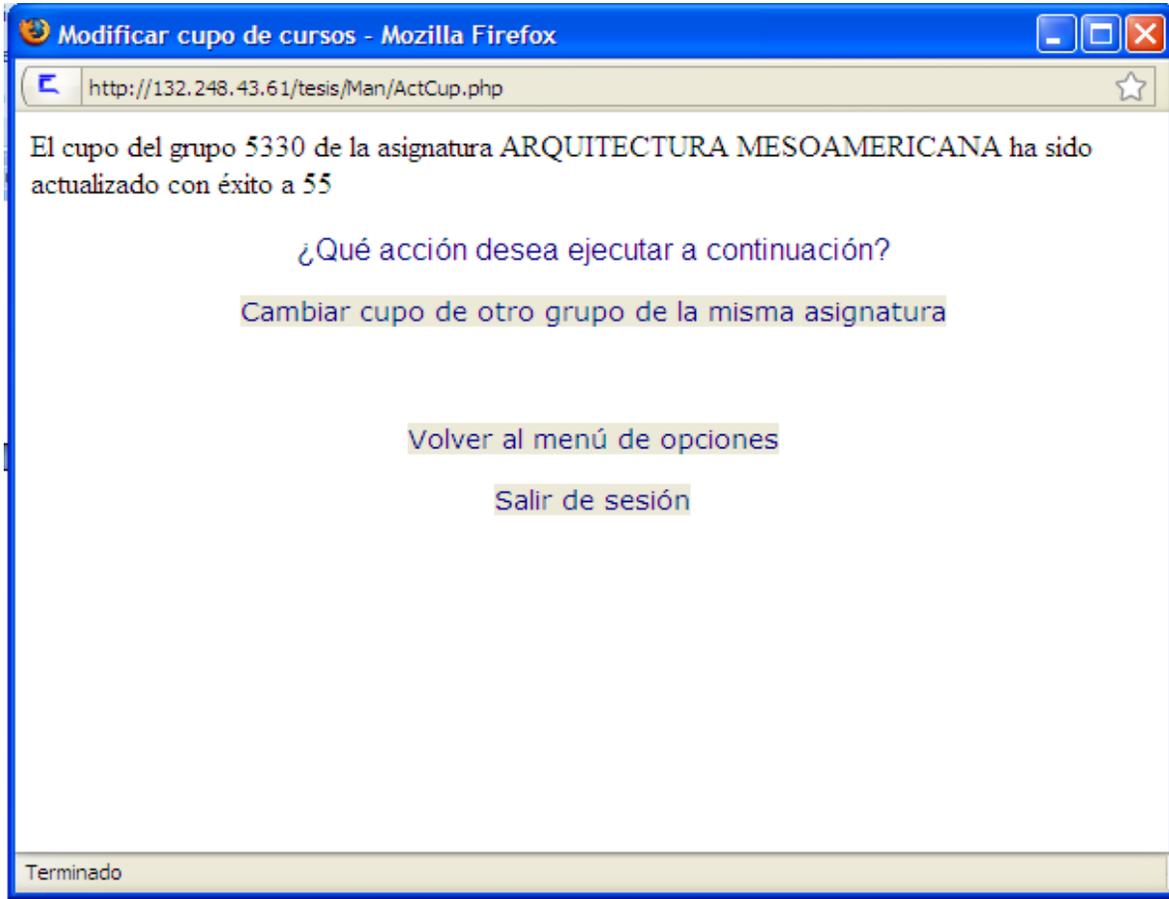


Figura 6. 16 Pantalla que muestra que el cupo de un grupo ha sido actualizado

## Cancelar grupos de cursos

### Funcionalidad o característica:

Se cancela un grupo de una asignatura obligatoria o selectiva interna.

### Secuencia:

- Se selecciona la opción "Cancelar grupo de cursos".
- Se muestran las etapas de la carrera de Arquitectura, además de una adicional para los cursos de carácter selectivo interno.
- Se elige una opción de etapa.
- Se muestran las asignaturas correspondientes a la etapa seleccionada.
- Se elige una opción de asignatura.
- Se muestran los grupos de la asignatura.
- Se elige la opción correspondiente a un grupo.

- Se muestran datos del grupo seleccionado.
- Se selecciona la opción “Cancelar grupo”.
- El grupo elegido queda cancelado.

**Prerrequisitos:**

- El actor Mantenimiento debió de haberse autenticado.

**Resultados de la prueba:**

El grupo elegido de la asignatura seleccionada queda cancelado.

**Evaluación de resultados:**

Tras seleccionar la opción “Cancelar grupo” se inhabilita el grupo, el proceso es exitoso cuando el sistema informa de la cancelación.

Pantalla:

Tras la seleccionar “Cancelar grupo”:

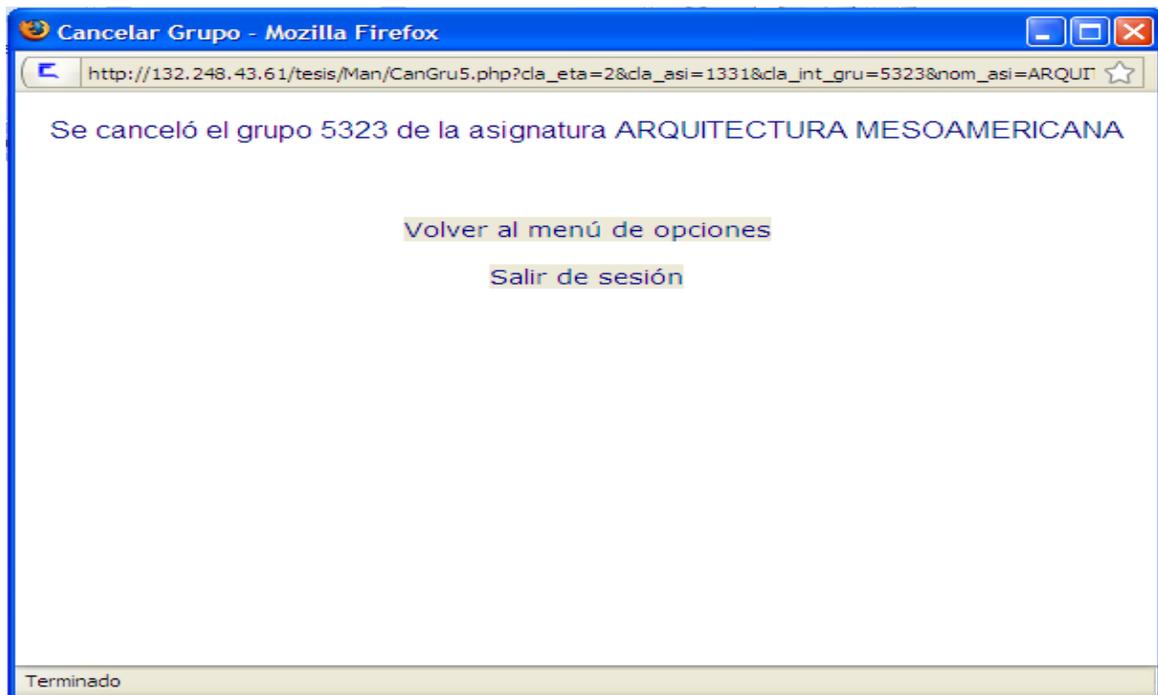


Figura 6. 17 Pantalla que muestra que un grupo ha sido cancelado

### **Cambiar calificación de un Alumno.**

#### **Funcionalidad o característica:**

Se cambia la calificación obtenida por un Alumno en una asignatura obligatoria o selectiva interna.

#### **Secuencia:**

- Se selecciona la opción “Cambiar calificación de un Alumno”.
- Se ingresan el número de cuenta del Alumno y la clave de la asignatura.
- Se elige la opción “Actualizar”.
- Se muestran los datos del Alumno, así como las veces que ha cursado dicha asignatura y la calificación obtenida.
- Se elige la opción “Datos son correctos”.
- Se ingresa la nueva calificación y se elige la opción “Cambiar la calificación”.
- Se actualiza la calificación en el último grupo inscrito de la asignatura y la etapa mínima cubierta en su totalidad por el Alumno.

**Prerrequisitos:**

- El actor Mantenimiento debió de haberse autenticado.

**Resultados de la prueba:**

Se cambia la calificación del Alumno en la asignatura proporcionada. En caso de que haya necesidad de cambio en la etapa mínima del Alumno, ésta se actualiza.

**Evaluación de resultados:**

Tras ingresar la nueva calificación, y elegir el botón “Cambiar la calificación”, se realiza la actualización de ésta. El proceso es exitoso cuando el sistema informa que la acción seleccionada ha sido ejecutada.

**Actualización de etapas mínimas****Funcionalidad o característica:**

Se actualizan las etapas que todos los alumnos de Arquitectura han cubierto en su totalidad.

**Secuencia:**

- Se selecciona la opción “Actualización de etapas de etapas mínimas”.

**Prerrequisitos:**

- El actor Mantenimiento debió de haberse autenticado.

**Resultados de la prueba:**

Se actualizan las etapas que los Alumnos de Arquitectura han cubierto en su totalidad.

**Evaluación de resultados:**

Tras ingresar seleccionar la opción “Actualización de etapas de etapas mínimas”, se llevan a cabo las validaciones y actualizaciones para cada caso; el proceso es exitoso una vez que se informa que éste ha concluido.

## **Asignación de Horarios para Altas, Bajas y Cambios**

### **Funcionalidad o característica:**

Se asignan los turnos en que los Alumnos realizarán su proceso en la etapa de Altas, Bajas y Cambios.

### **Secuencia:**

- Se selecciona la opción “Asignación de Horarios para Altas, Bajas y Cambios”.

### **Prerrequisitos:**

- El actor Mantenimiento debió de haberse autenticado.

### **Resultados de la prueba:**

Se asignan los turnos para Altas, Bajas y Cambios.

### **Evaluación de resultados:**

Tras ingresar seleccionar la opción “Asignación de Horarios para Altas, Bajas y Cambios”, y de acuerdo a la jerarquía de asignación para este proceso (ver capítulo 2 “Análisis”); el proceso es exitoso una vez que se informa que éste ha concluido.

Pantalla:

Tras seleccionar la opción “Asignación de Horarios para Altas, Bajas y Cambios”:

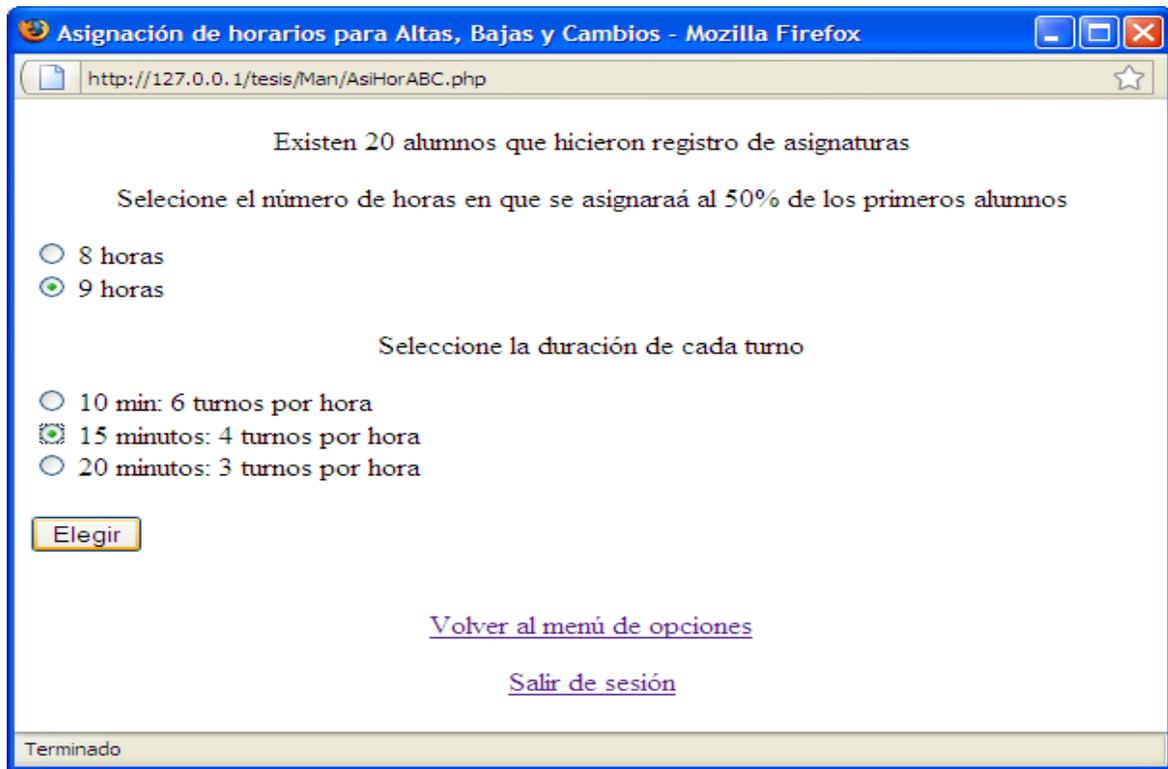


Figura 6. 18 Pantalla que permite seleccionar el intervalo de tiempo en el que se realizará la Asignación de Horarios para Altas, Bajas y Cambios

### Asignación de grupos

#### Funcionalidad o característica:

Se asignan los grupos de las asignaturas que los Alumnos solicitaron durante el periodo de Registro.

#### Secuencia:

- Se selecciona la opción “Asignación de grupos”.

#### Prerrequisitos:

- El actor Mantenimiento debió de haberse autenticado.

#### Resultados de la prueba:

Los grupos son asignados y con ello se formaliza la inscripción de los Alumnos a las asignaturas para el siguiente semestre.

### **Evaluación de resultados:**

Tras seleccionar la opción “Asignación de Grupos”, se realizan las validaciones necesarias para las asignaciones; el proceso es exitoso una vez que se informa que éste ha concluido.

Pantalla:

Tras elegir la opción “Asignación de Grupos”.

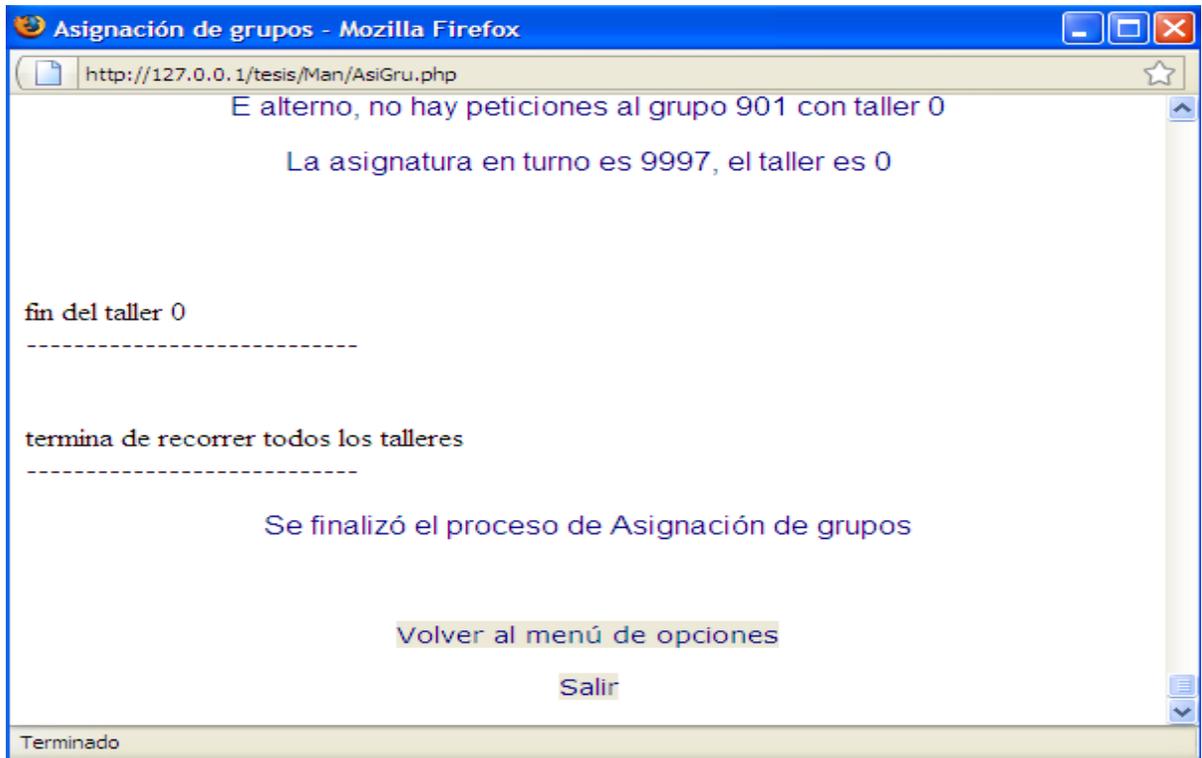


Figura 6. 19 Pantalla que muestra el resultado del proceso de Asignación de grupos

### **Actualización de curso de cómputo**

#### **Funcionalidad o característica:**

Se actualiza el requisito de curso de cómputo para un Alumno.

#### **Secuencia:**

- Se selecciona la opción “Actualización de curso de cómputo”.
- Se ingresa el número de cuenta del Alumno.

- Finalmente se muestra que se ha actualizado el requisito de curso de cómputo para el Alumno.

### **Prerrequisitos:**

- El actor Mantenimiento debió de haberse autenticado.

### **Resultados de la prueba:**

Se actualiza el requisito de curso de cómputo para el Alumno.

### **Evaluación de resultados:**

Tras ingresar el número de cuenta del Alumno y elegir la opción “Actualizar” se actualiza el requisito, el proceso es exitoso una vez que se informa que se ha realizado.

Pantalla:

Tras elegir la opción “Actualizar”:

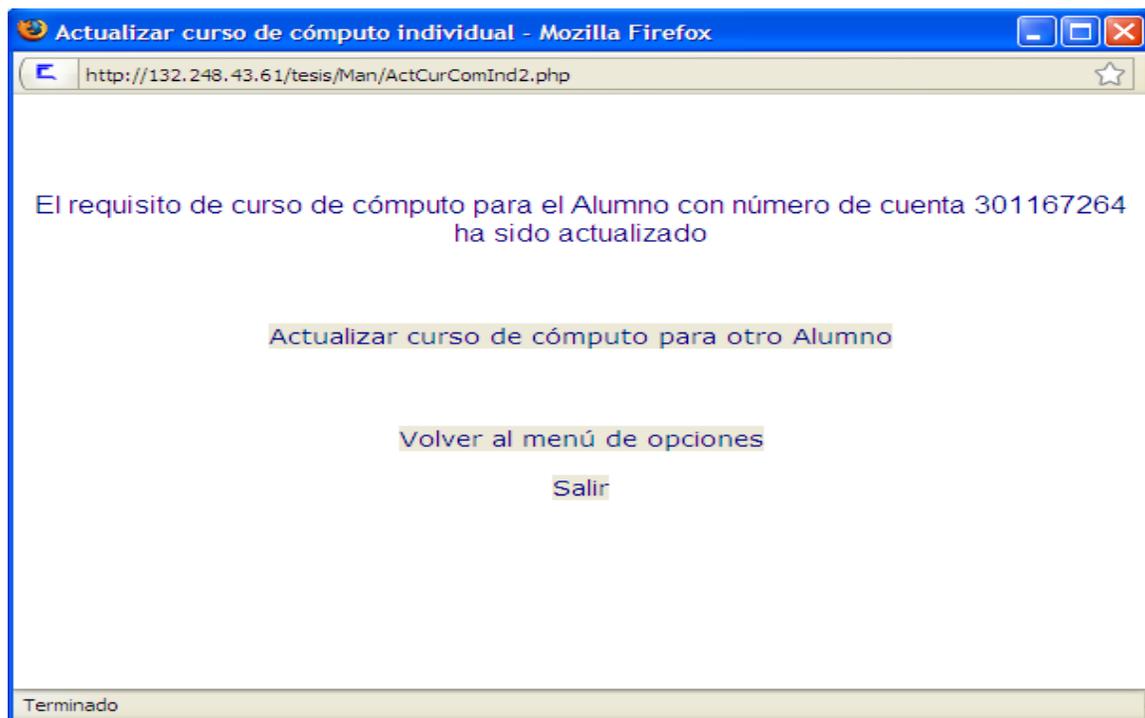


Figura 6. 20 Página que muestra que el requisito de Curso de Cómputo ha sido actualizado

### **6.3.4 Coordinador de Carrera**

#### **Modificar el cupo del Taller**

##### **Funcionalidad o característica:**

El Coordinador de Carrera puede cambiar el cupo de un grupo de Taller.

##### **Secuencia:**

- Se muestran todos los Talleres de la carrera de Arquitectura.
- Se elige un determinado Taller.
- Se muestran las etapas de la carrera de Arquitectura.
- Se elige una etapa.
- Se muestran los grupos correspondientes a la etapa elegida.
- Se elige el grupo al cual quiere realizar el cambio de cupo.
- Se muestra el cupo y demanda del grupo del Taller elegido junto con un campo para ingresar el cupo deseado.
- Se ingresa nuevo cupo y se selecciona la opción “Actualizar cupo”.

##### **Prerrequisitos:**

- El actor Coordinador de Carrera debió de haberse autenticado.

##### **Resultado:**

Se actualiza el cupo del grupo de Taller seleccionado.

##### **Evaluación de resultados:**

Tras ingresar el nuevo cupo y elegir la opción “Actualizar cupo”, el cambio es realizado y el proceso tiene éxito al informar que el cupo ha sido actualizado.

### **6.3.5 Jefe de asuntos Escolares**

#### **Realizar inscripciones de casos extemporáneos**

**Funcionalidad o característica:**

El Jefe de Asuntos Escolares tiene la facultad de inscribir de manera extemporánea a un Alumno en un curso, ya sea en un grupo ordinario o extraordinario (asignaturas de tipo obligatorio y/o selectivo interno).

**Secuencia:**

- Se elige la opción de “Inscribir Cursos”.
- Se ingresa el número de cuenta del Alumno y selecciona la opción “Enviar.
- Se muestran los datos del alumno, junto con las opciones de “Datos son correctos” o “Datos incorrectos” y “Consultar historial”.
- Se elige la opción “Datos son correctos”.
- Se muestran las etapas de la carrera de Arquitectura.
- Se elige una etapa.
- Se muestran las asignaturas de la etapa elegida.
- Se selecciona una asignatura.
- Se muestran los grupos de la asignatura seleccionada.
- Se elige un grupo de la asignatura y se selecciona la opción “inscribir”.

**Prerrequisitos:**

- El actor Jefe de Asuntos Escolares debió de haberse autenticado.

**Resultado:**

Alumno se inscribe en el grupo de la asignatura seleccionada.

**Evaluación de resultados:**

Para inscribir al Alumno se selecciona un grupo de la asignatura elegida, el proceso es exitoso una vez que el sistema informa que la inscripción se ha realizado.

**Pantalla:**

Tras seleccionar la opción “Ingresar”:

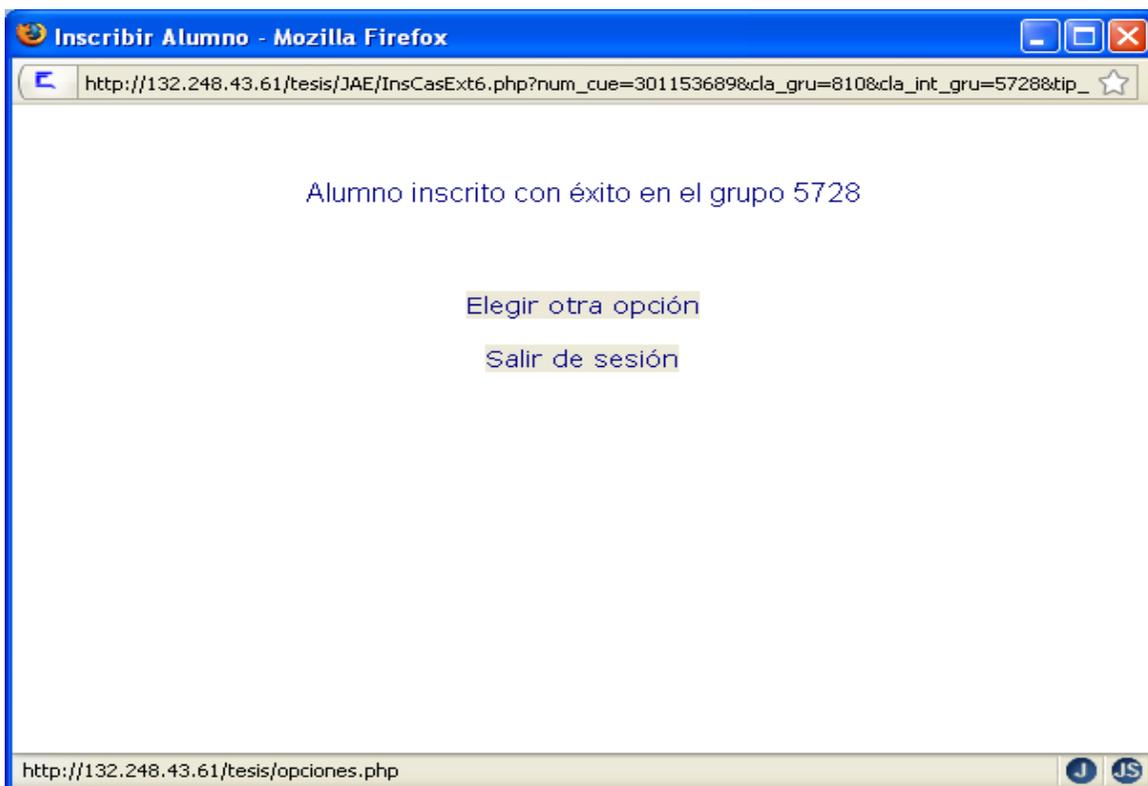


Figura 6. 21 Pantalla que muestra que un Alumno ha sido inscrito en una asignatura

## **Consultar Historial Académico Completo**

### **Funcionalidad o característica:**

El Jefe de Asuntos Escolares tiene la facultad de poder consultar el Historial Académico Completo de un Alumno.

### **Secuencia:**

- Se selecciona la opción de “Inscribir Cursos”.
- Se ingresa el número de cuenta del Alumno y se selecciona la opción “Enviar”.
- Se muestran los datos del alumno, junto con las opciones de “Datos son correctos”, “Datos incorrectos” y “Consultar Historial”.
- Se elige la opción “Consultar Historial”.
- Se muestra el Historial Académico Completo del Alumno.

### **Prerrequisitos:**

- El actor Jefe de Asuntos Escolares debió de haberse autenticado.

### Resultado:

Se muestra el Historial Académico del Alumno.

### Evaluación de resultados:

Se consultan de acuerdo al número de cuenta ingresado en la base de datos, las asignaturas cursadas por el Alumno. El proceso tiene éxito al mostrar la historia académica completa del Alumno.

Pantalla:

Asignatura	Clave	Periodo	Calificación	Tipo de examen
INTRODUCCION HISTORIC CRITIC	1130	20051	NP	O
TEORIA DE LA ARQUITECTURA I	1131	20051	NP	O
TALLER DE ARQUITECTURA I	1132	20051	NP	O
MATEMATICAS APLICADAS I	1133	20051	08	O
SISTEMAS ESTRUCTURALES I	1134	20051	NP	O

Cerrar ventana

Terminado

Figura 6. 22 Pantalla que despliega el Historial Académico de un Alumno

# Conclusiones

La aplicación de sistemas que conlleven a aprovechar los recursos de software, hardware y humano de una mejor manera, con el objetivo de que la información manejada por éste sea oportuna, relevante y exacta, resulta de suma importancia. Siguiendo estas particularidades y atributos fue desarrollado un sistema para optimizar el proceso de reinscripciones de la licenciatura en Arquitectura.

El sistema esta basado conforme a los estatutos generales de reinscripciones de la Secretaría de Asuntos Escolares de la Facultad de Arquitectura, utilizando para su desarrollo la metodología de ingeniería de software de un modelo de ciclo de vida en cascada.

Como técnicas para determinar los requerimientos del sistema y de los usuarios, tomando en cuenta el entorno en el que llevan a cabo sus actividades se realizaron entrevistas y presentaciones con las cuales se describieron los escenarios de los diversos actores en el sistema.

Posteriormente se detalló el análisis de requerimientos en el diseño del sistema, mediante herramientas CASE utilizando diferentes diagramas UML: diagramas de casos de uso, diagramas de secuencia y, finalmente, diagramas de actividades, complementados con el Diagrama Entidad Relación, cuya función es la de esquematizar el comportamiento del sistema.

El diseño del Sistema reviste especial importancia debido a que un buen diseño se traduce en un software con un adecuado nivel de calidad.

Aunado a ello, el uso de estándares de programación: PHP5 como lenguaje de programación orientado a la creación de páginas web dinámicas, SQL ANSI/ISO como lenguaje de consulta, PostgreSQL como DBMS para brindar integridad y flexibilidad a los datos, HTML 4.01 para dar una estructura al contenido de la información, y finalmente CSS para proporcionar un formato unánime al cuerpo de la página, en el desarrollo del Sistema, implica que un navegador será capaz de interpretar la aplicación que se lleva a cabo.

La importancia de que el lenguaje PHP se ejecute del lado del servidor y que sea un lenguaje Web que trabaje sobre distintos DBMS's radica en el hecho de que en la actualidad no existe ningún navegador web que implemente todos los estándares del World Wide Web Consortium, así como tampoco existe ningún producto SQL que implemente todas las especificaciones de los estándares ANSI / ISO.

Al realizar pruebas de unidad a las distintas interfaces, se aseguró que la información fluyera de forma adecuada entre los módulos que integran el sistema, por lo que se implementaron diversos caminos para el manejo de errores, en este caso se utilizaron pruebas límites para asegurar que no existieran errores en los elementos iniciales y/o finales de los arreglos manejados.

Se ocupó una integración descendente en profundidad, para verificar que, a partir de los menús de opciones (módulo de control principal de cada actor) y conforme se incorporarán nuevos módulos se fueran aislando y corrigiendo los errores.

Se efectuaron pruebas de aceptación de todos los módulos del sistema, dentro de un entorno controlado estando presentes el Ing. Rodrigo Rosas de la Secretaría de Asuntos Escolares de la Facultad de Arquitectura y los desarrolladores del sistema.

Como resultado de las pruebas de validación ya mencionadas se realizó un mantenimiento perfectivo, es decir, se efectuaron algunas modificaciones a funciones, así como de la presentación de la información extraída de la base de datos.

Cabe resaltar que la documentación generada a lo largo del proceso de desarrollo del sistema, es una herramienta que facilita la comprensión, uso y mantenimiento del mismo, cuya finalidad es la de lograr con cada actualización la consecución de un software de mejor calidad y que pueda entonces adaptarse para responder a nuevos requerimientos.

# Anexo A

## Oracle

Oracle es una herramienta cliente/servidor para la gestión de bases de datos, tiene una gran potencia, y su elevado precio hace que solo se vea en empresas muy grandes y multinacionales.

Soporta acceso por SQL y por lenguajes de programación, posee un lenguaje de procedimientos llamado PL/SQL.

Su base de datos tiene estructuras lógicas y físicas, gracias a que son separadas el almacenamiento físico de datos puede ser manejado sin afectar el acceso a las estructuras lógicas del almacenamiento

## Java

Es un lenguaje de programación de alto nivel orientado a objetos, con sintaxis basada en lenguaje C.

Con Java se pueden crear dos tipos de programas:

Applets: programas que se integran en las páginas web y que, residiendo en el servidor, son ejecutados por el cliente. La ejecución necesita de la interpretación del código compilado por el software del cliente.

Aplicaciones: programas autónomos que se pueden ejecutar en cualquier equipo. En este último caso se puede optar por generar código compilado similar al de los applets y que para su ejecución necesita de un intérprete o código compilado ejecutable directamente como en cualquier otro lenguaje de programación.

Algunas de sus características son:

Es un lenguaje orientado a objeto.

Admite programación concurrente.

Dispone de clases de objetos para la generación de interfaces gráficas de usuario.

Tiene prestaciones multimedia.

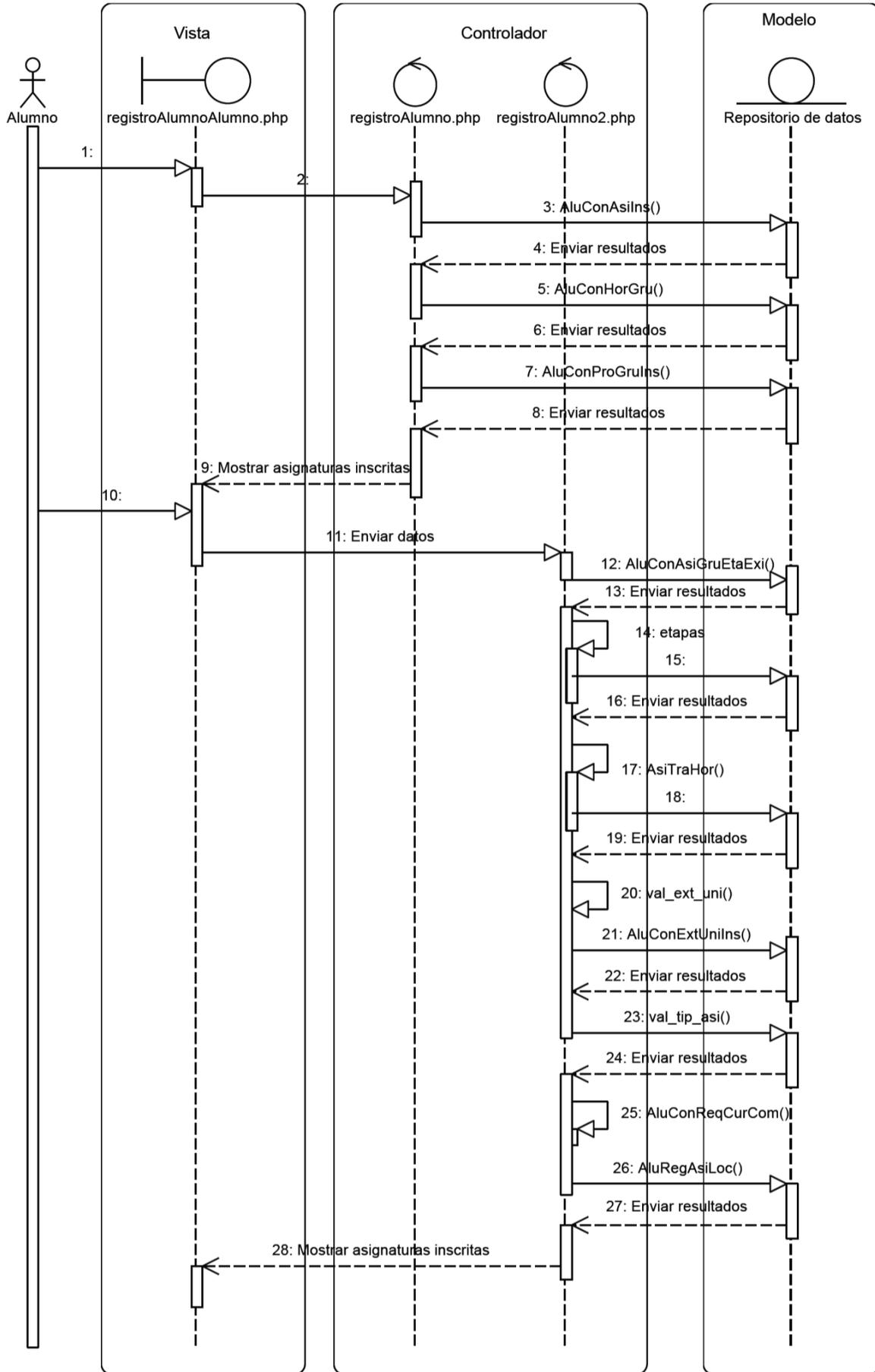
A través de Internet se puede acceder a todo lo necesario para desarrollar applets Java.

# Anexo B

## Diagramas de Secuencia

Alumno

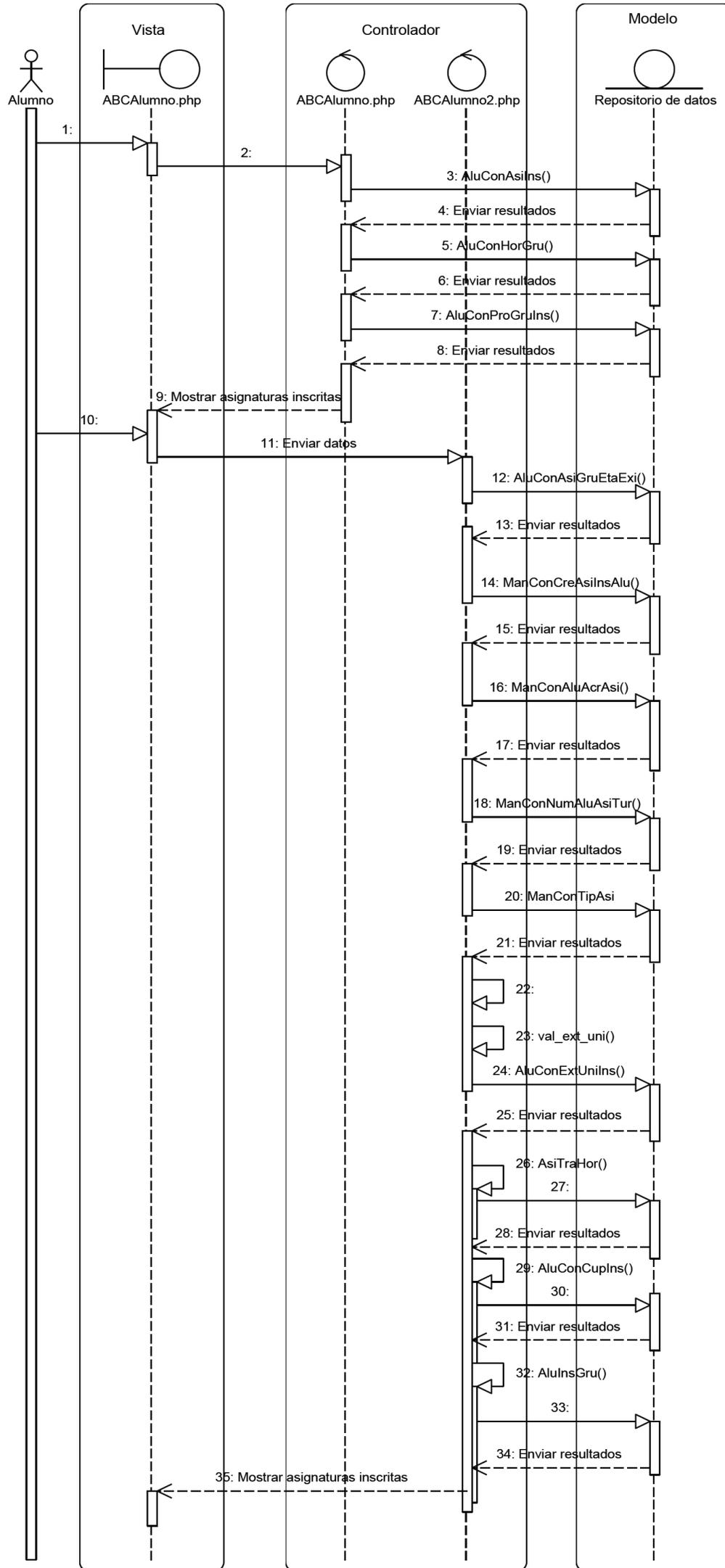
Registro de asignaturas



- Del escenario Ingresar a Trámites Académicos.
3. Consultar las asignaturas inscritas previamente.
  5. Consultar los horarios de las asignaturas inscritas.
  7. Consultar los profesores que imparten los grupos de las asignaturas inscritas.
  9. Mostrar cursos y grupos en que Alumno está inscrito previamente.
  10. Alumno ingresa clave de asignatura y grupo
  12. Revisar que clave de asignatura y grupo sean válidas.
  14. Revisar seriación por etapas
  17. Revisar que no hay traslape de horarios con asignaturas previamente registradas.
  20. Revisar que clave de asignatura corresponda a una Extensión Universitaria.
  21. Consultar que Alumno no haya inscrito previamente un grupo de Extensión Universitaria.
  23. Revisar que tipo de asignatura sea selectiva.
  24. Revisar que Alumno cumpla con el requisito de curso de Cómputo.
  26. Insertar un registro en el grupo de la asignatura proporcionada.
  28. Mostrar cursos y grupos registrados por Alumno.

Figura B.1 Diagrama de Secuencias Registro de asignaturas

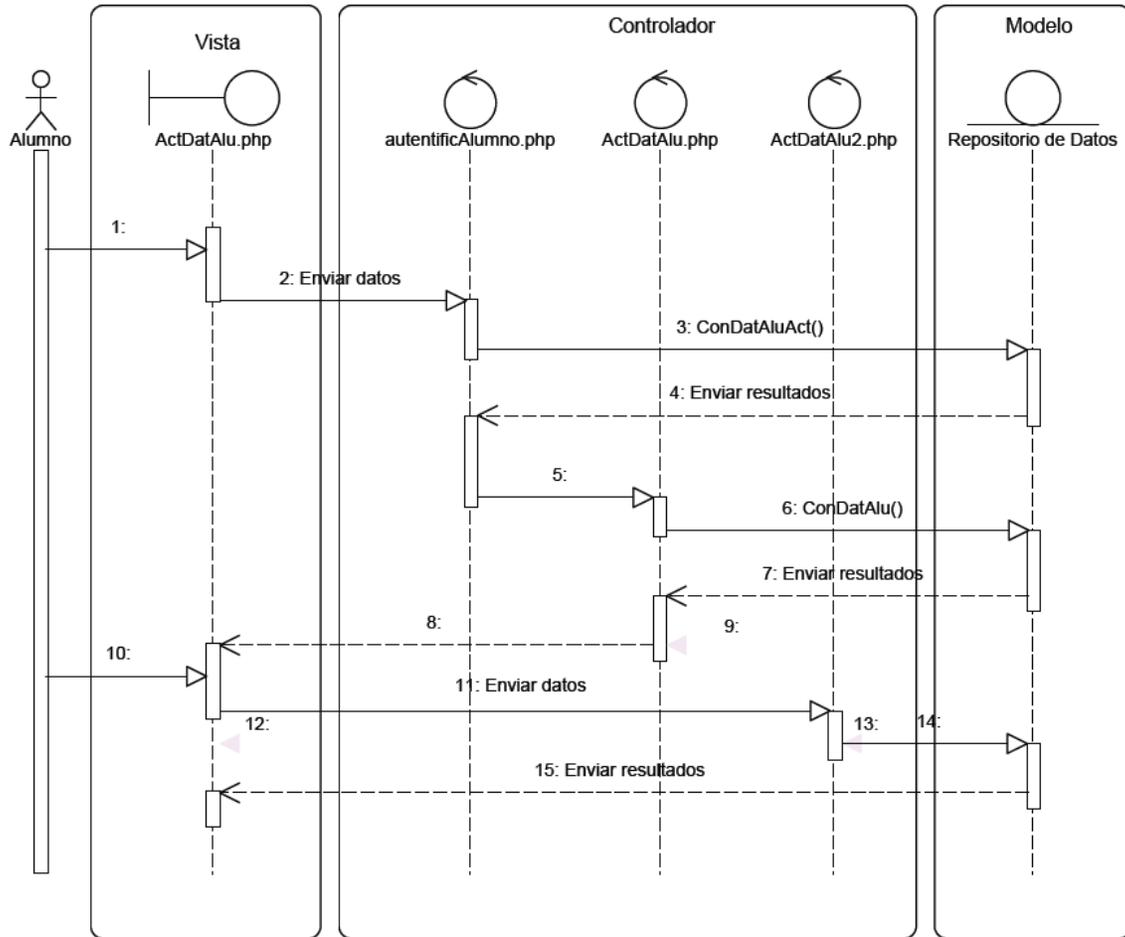
## Altas, bajas y cambios



- Del escenario Ingresar a Trámites Académicos.
3. Consultar las asignaturas inscritas previamente.
  5. Consultar los horarios de las asignaturas inscritas.
  7. Consultar los profesores que imparten los grupos de las asignaturas inscritas.
  9. Mostrar cursos y grupos en que Alumno está inscrito previamente.
  10. Alumno ingresa clave de asignatura y grupo
  12. Consultar clave de asignatura, grupo y etapa de asignatura.
  14. Consultar créditos totales para inscripción actual.
  16. Consultar que Alumno haya acreditado asignatura proporcionada.
  18. Consultar el número de veces que Alumno ha cursado asignatura proporcionada.  
Establecer el tipo de grupo ordinario/extraordinario.
  20. Consultar el tipo de asignatura de curso seleccionado.
  22. Revisar seriación por etapas o que tipo de asignatura sea selectiva.
  23. Revisar que clave de asignatura corresponda a una Extensión Universitaria.
  24. Consultar que Alumno no haya inscrito previamente un grupo de Extensión Universitaria.
  26. Revisar que no hay traslape de horarios con asignaturas previamente inscritas.
  29. Revisar que haya cupo disponible para grupo seleccionado.
  32. Insertar un registro en el grupo de la asignatura proporcionada.  
Actualizar cupo del grupo correspondiente.
  35. Mostrar cursos y grupos en que Alumno está inscrito previamente.

Figura B.2 Diagrama de Secuencias Altas, Bajas y Cambios

## Actualizar datos

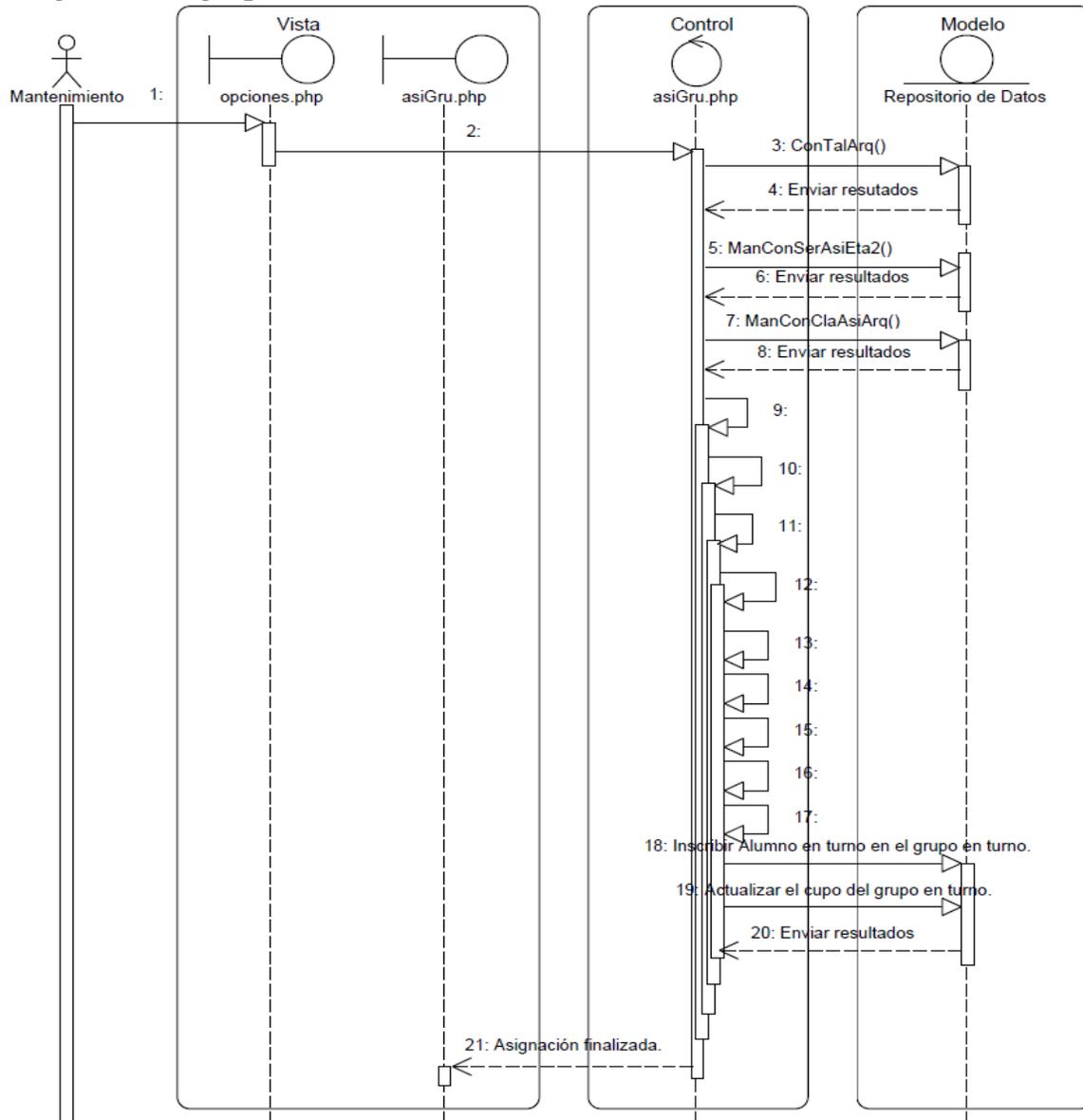


Del escenario Ingresar a Trámites Académicos.

1. Ingresar login y password.
3. Consultar que el Alumno haya actualizado sus datos.
6. Consultar información personal del Alumno.
8. Mostrar datos personales.
10. Alumno actualiza sus datos.
14. Actualizar Datos.
15. Información actualizada.

Figura B.3 Diagrama de Secuencias Actualizar datos

## Mantenimiento Asignación de grupos



Del escenario Ingresar a Trámites Académicos.

1: Seleccionar Asignación de grupos.

9: Mientras haya Talleres.

10: Mientras haya asignaturas.

11: Mientras haya grupos compatibles al taller en turno.

12: Mientras haya peticiones al grupo en turno.

13: Asignatura en turno es Práctica Profesional Supervisada.

14: Revisar que Alumno cumpla requisitos para Práctica Profesional Supervisada.

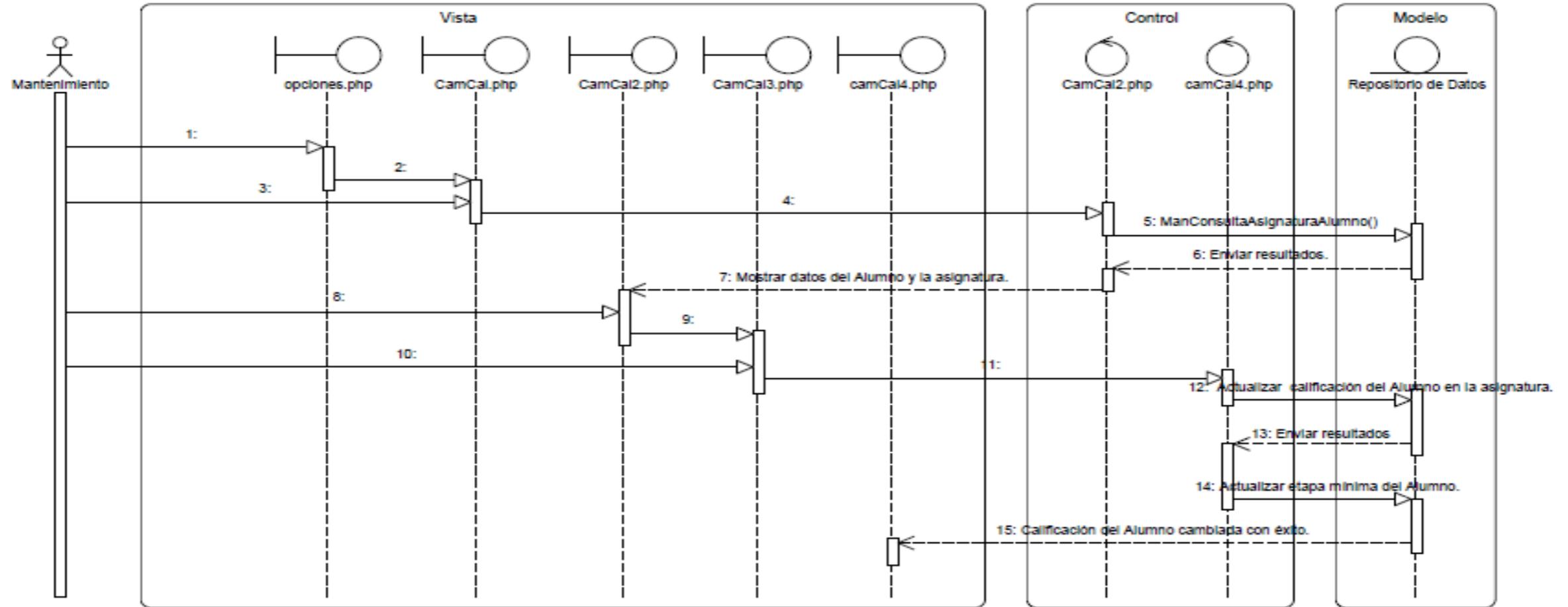
15: Revisar tipo de Alumno.

16: Revisar que Alumno sea del Taller en turno.

17: Alumno es del Taller en turno.

Figura B.4 Diagrama de Secuencias Asignación de grupos

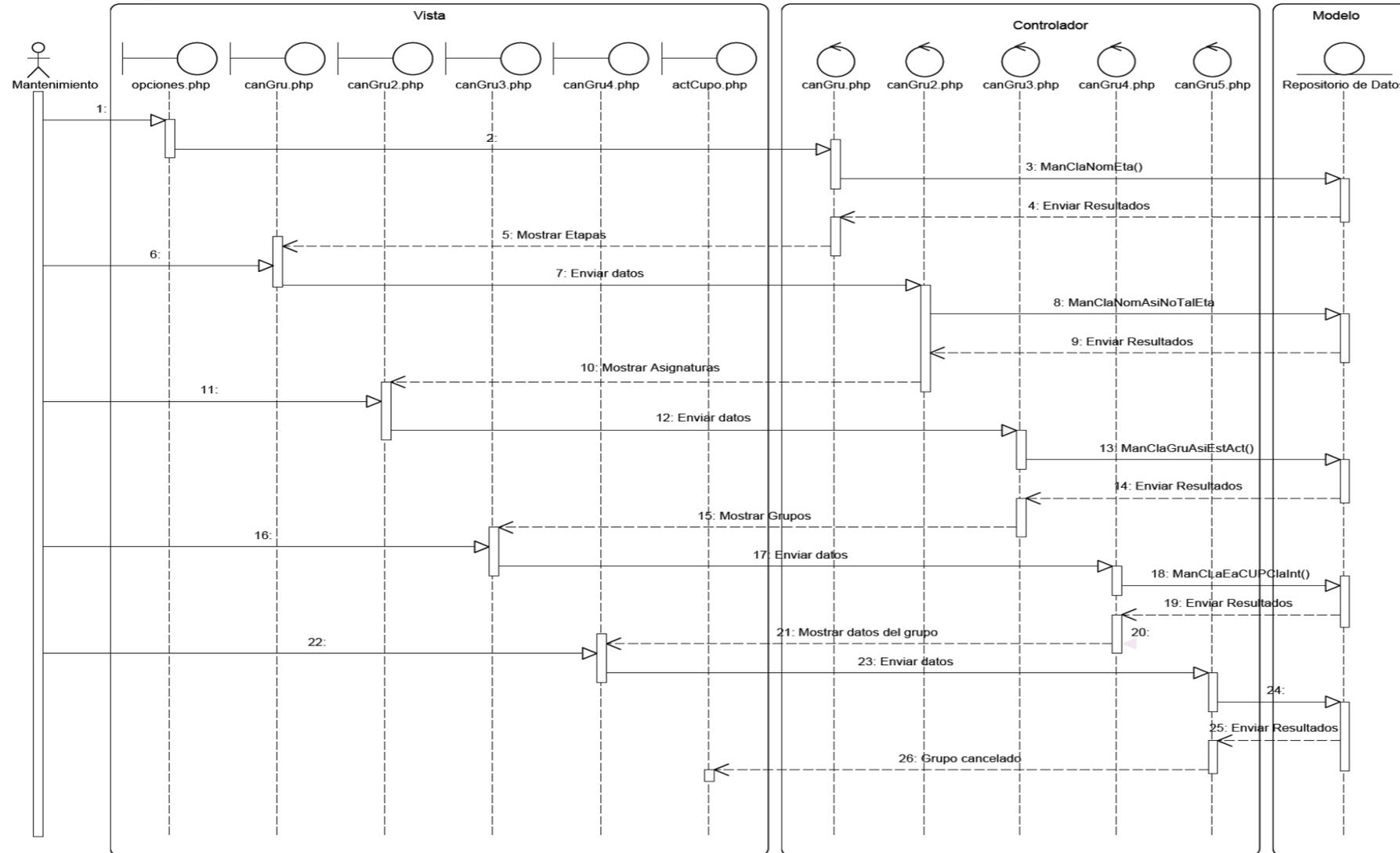
## Cambiar calificación de un Alumno



- Del escenario Ingresar a Trámites Académicos.
- 1: Seleccionar Cambiar calificación de un Alumno.
  - 3: Proporcionar número de cuenta y clave de asignatura.
  - 8: Seleccionar datos correctos.
  - 10: Proporcionar nueva calificación.

Figura B.5 Diagrama de Secuencias Cambiar calificación de un Alumno

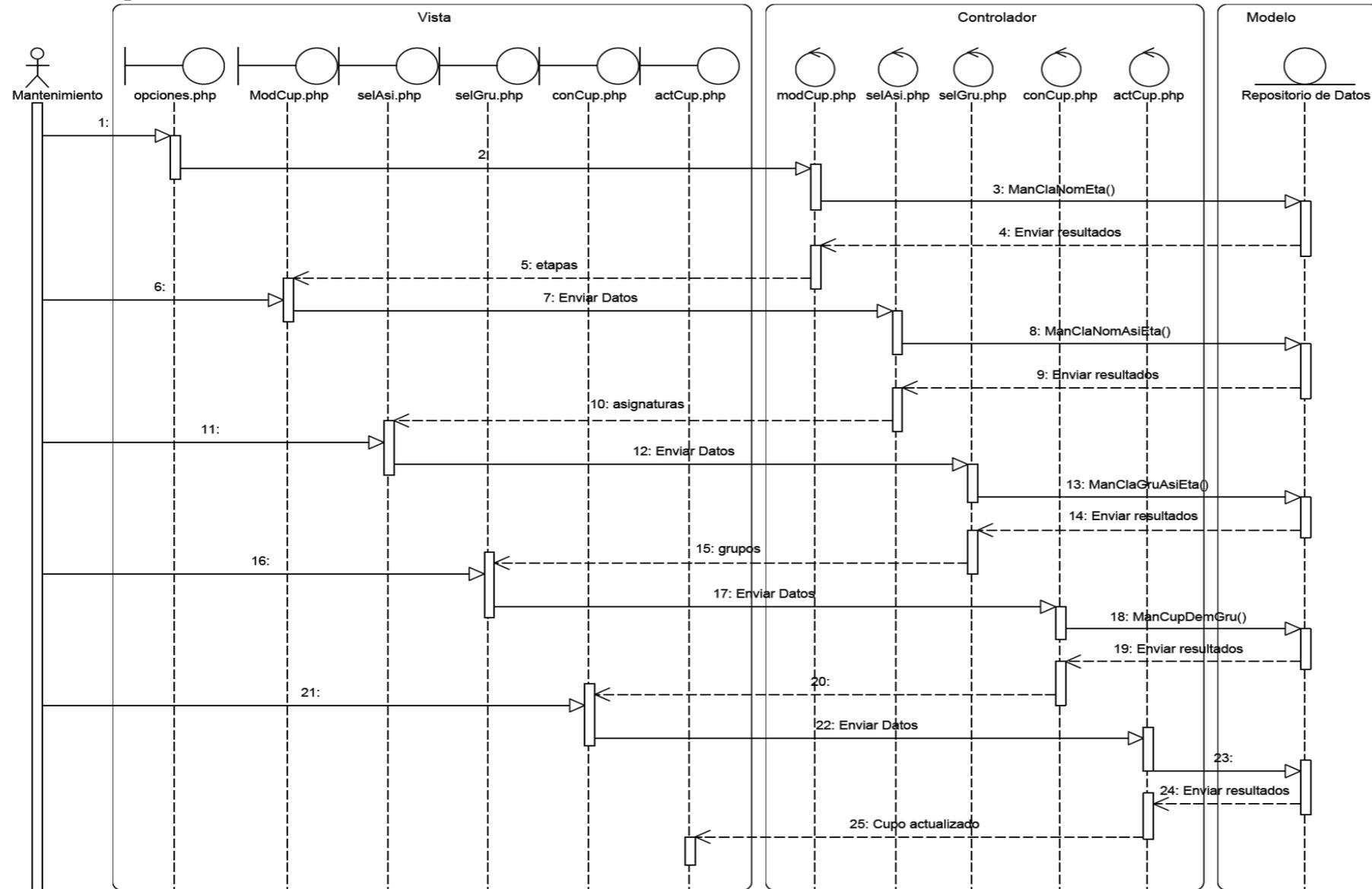
## Cancelar grupo de cursos



Del escenario Ingresar a Trámites Académicos.  
 1. Seleccionar Cancelar grupos de cursos.  
 6. Elegir una Etapa.  
 11. Seleccionar Asignatura.  
 16. Seleccionar grupo.  
 22. Seleccionar Cancelar grupo.  
 24. Actualizar bandera (baja) del grupo.  
 26. El grupo ha sido cancelado.

Figura B.6 Diagrama de Secuencias Cancelar grupo de cursos

## Modificar cupo de cursos



Del escenario Ingresar a Trámites Académicos.  
 1 Seleccionar Modificar cupo de cursos.  
 6. Elegir una Etapa.  
 11. Seleccionar Asignatura.  
 16. Seleccionar grupo.  
 21. Proporcionar nuevo cupo.  
 23. Actualizar cupo del grupo.  
 25. Cupo del grupo actualizado.

Figura B.7 Diagrama de Secuencias Modificar cupo de cursos

## Actualizar Curso de Cómputo

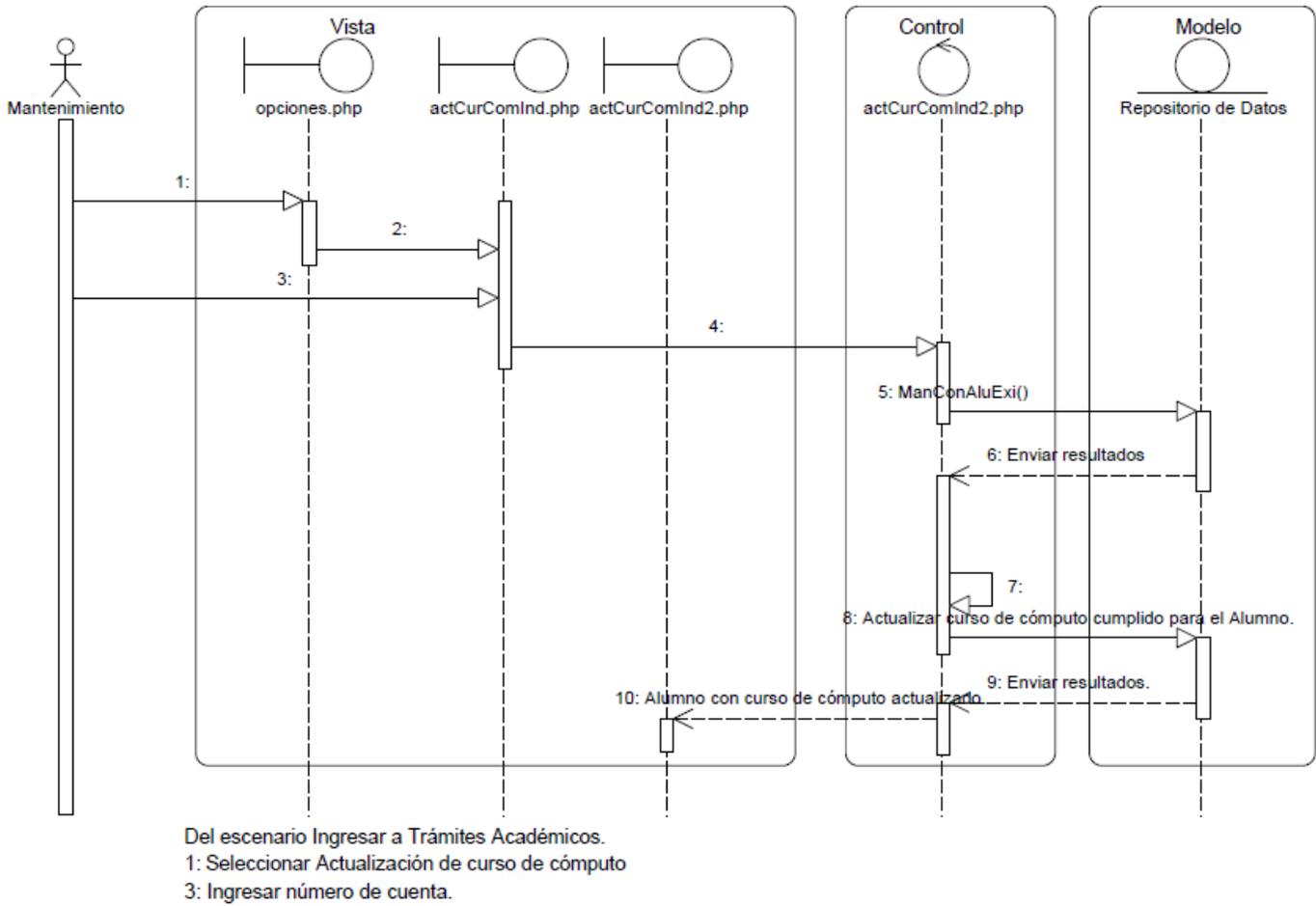
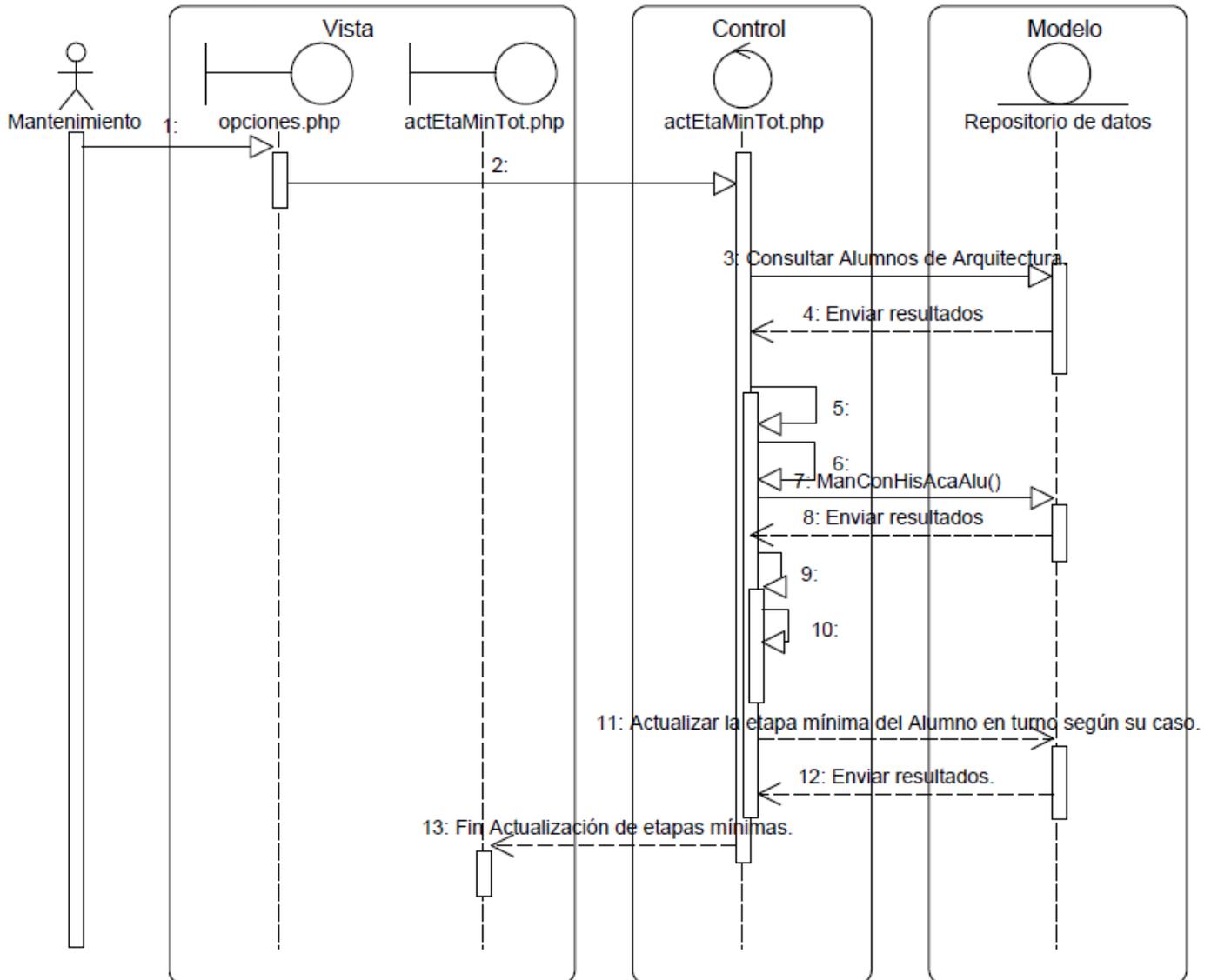


Figura B.8 Diagrama de Secuencias Actualizar Curso de Cómputo

## Actualización de etapas mínimas



Del escenario ingresar a Trámites Académicos

1: Seleccionar Actualización de etapas mínimas

3: Consultar Alumnos de Arquitectura. `ManConClaAsiEtaCar()`. `ManConNumAsiOblCar()`

5: Mientras haya Alumnos.

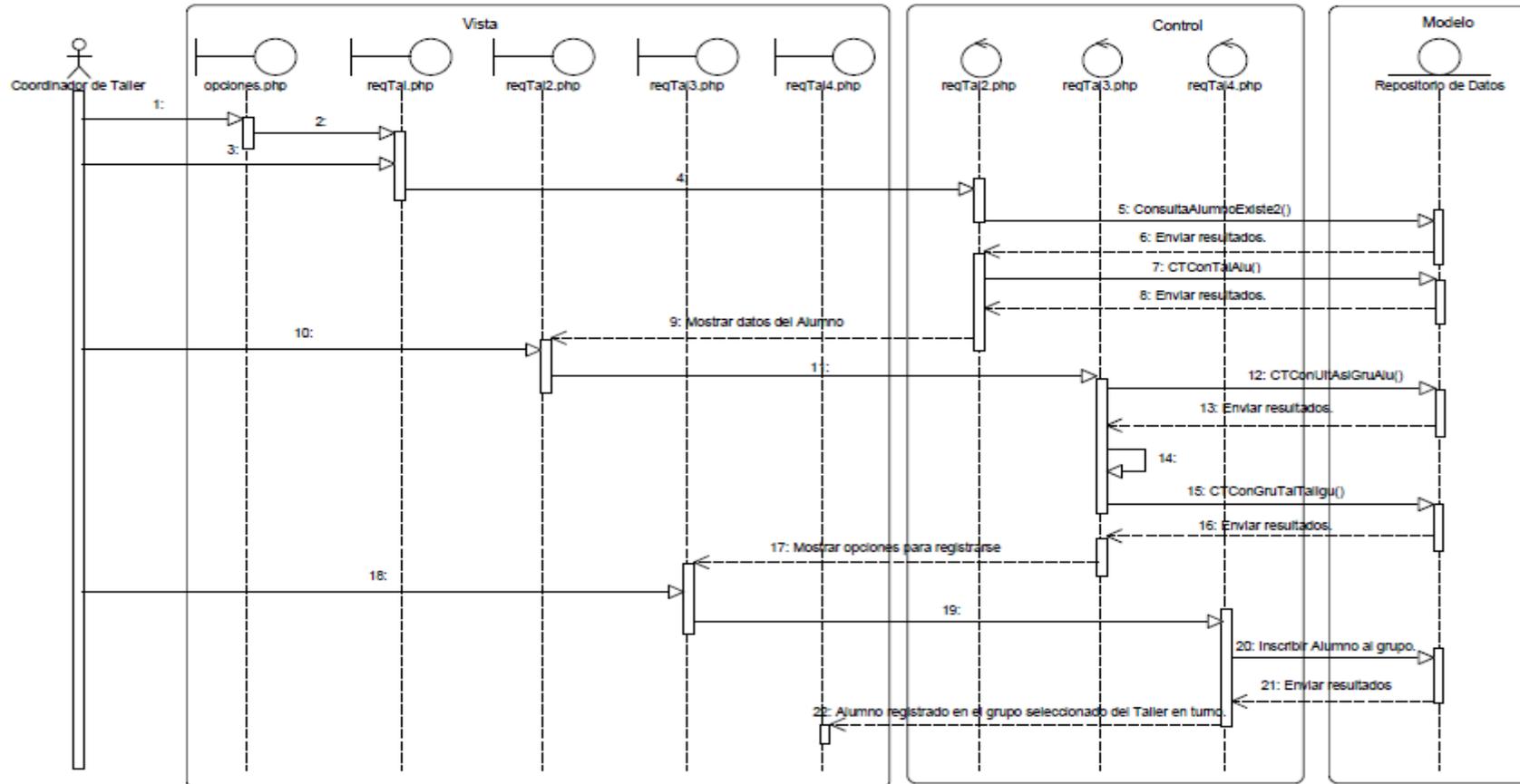
6: Escoger Alumno en turno.

9: Mientras el Alumno en turno tenga asignaturas en su historial académico.

10: Verificar el caso del Alumno en turno.

Figura B.9 Diagrama de Secuencias Actualización de Etapas Mínimas

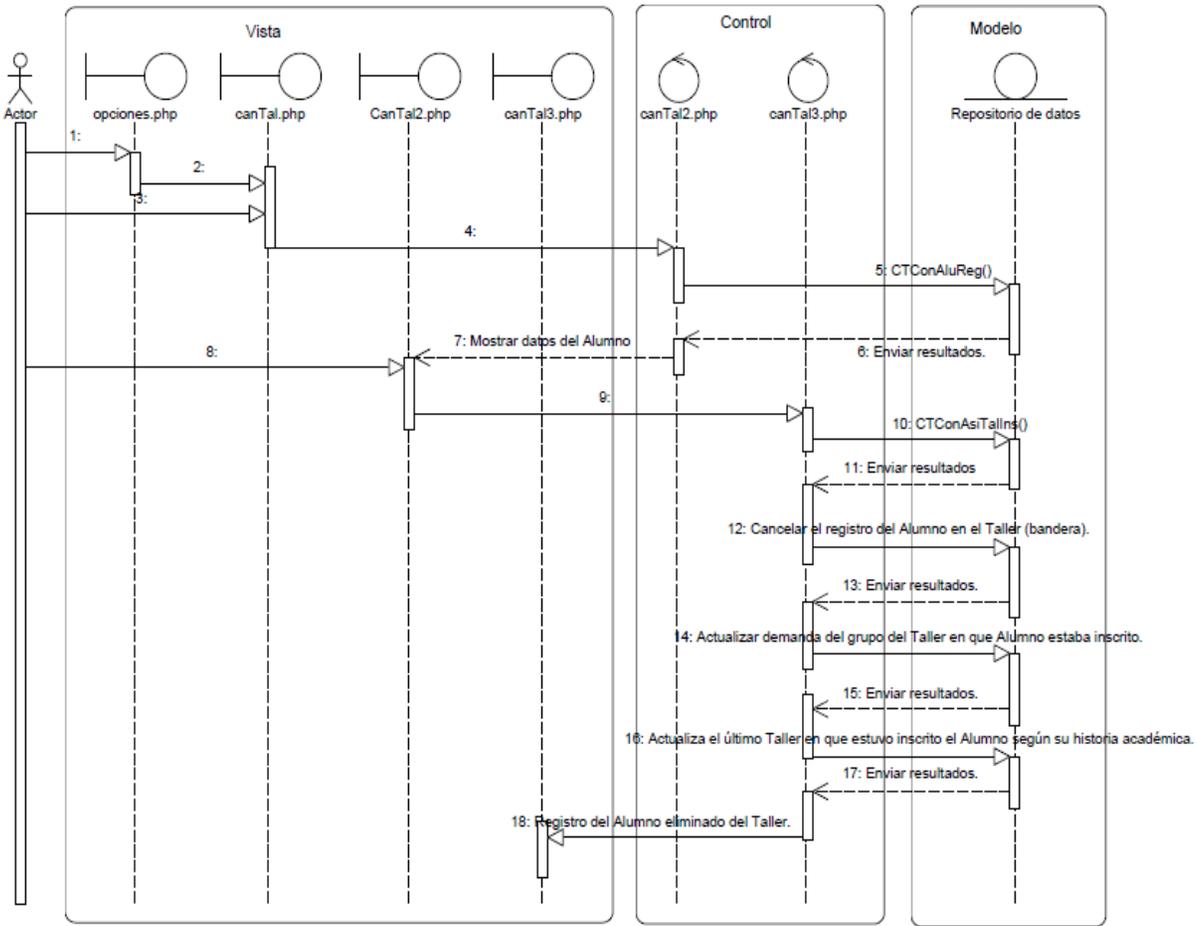
## Coordinador de Taller Inscribir Alumno a Taller



- Del escenario ingresar a Trámites Académicos
- 1: Seleccionar Inscripción a Taller.
  - 3: Proporcionar número de cuenta del Alumno.
  - 9: Mostrar datos del Alumno.
  - 10 Seleccionar Datos son correctos.
  - 14: Definir el caso del Alumno.
  - 18: Seleccionar Grupo elegido.
  - 20: Inscribir Alumno al grupo. Actualizar el cupo del grupo. Actualizar Taller al que el Alumno se registró.

Figura B.10 Diagrama de Secuencias Inscribir Alumno a Taller

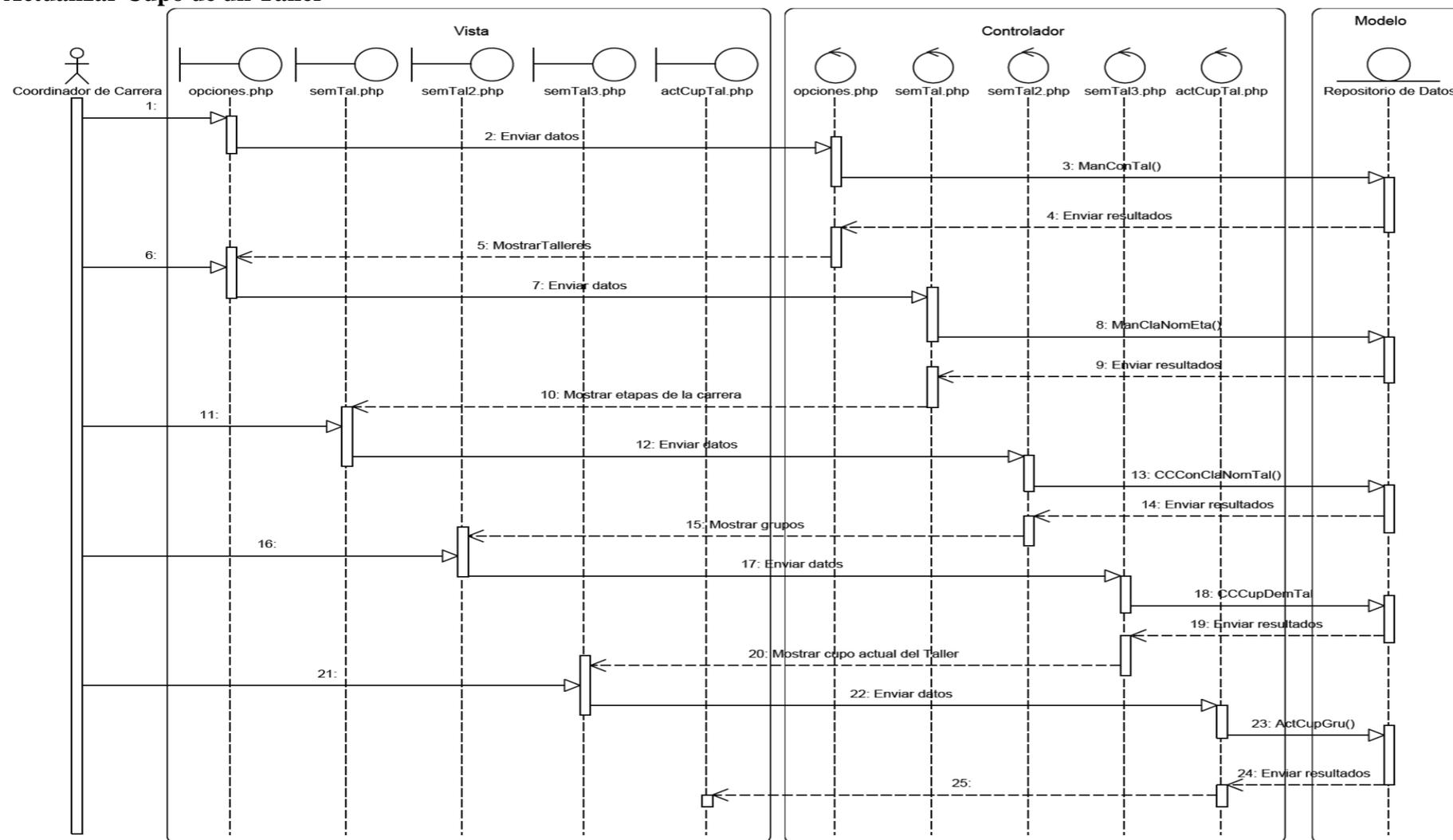
## Cancelar Registro a Taller



Del escenario Ingresar a Trámites Académicos  
 1: Seleccionar Cancelar inscripción al Taller.  
 3: Proporcionar número de cuenta del Alumno  
 8: Seleccionar Datos correctos.

Figura B.11 Diagrama de Secuencias Cancelar Registro a Taller

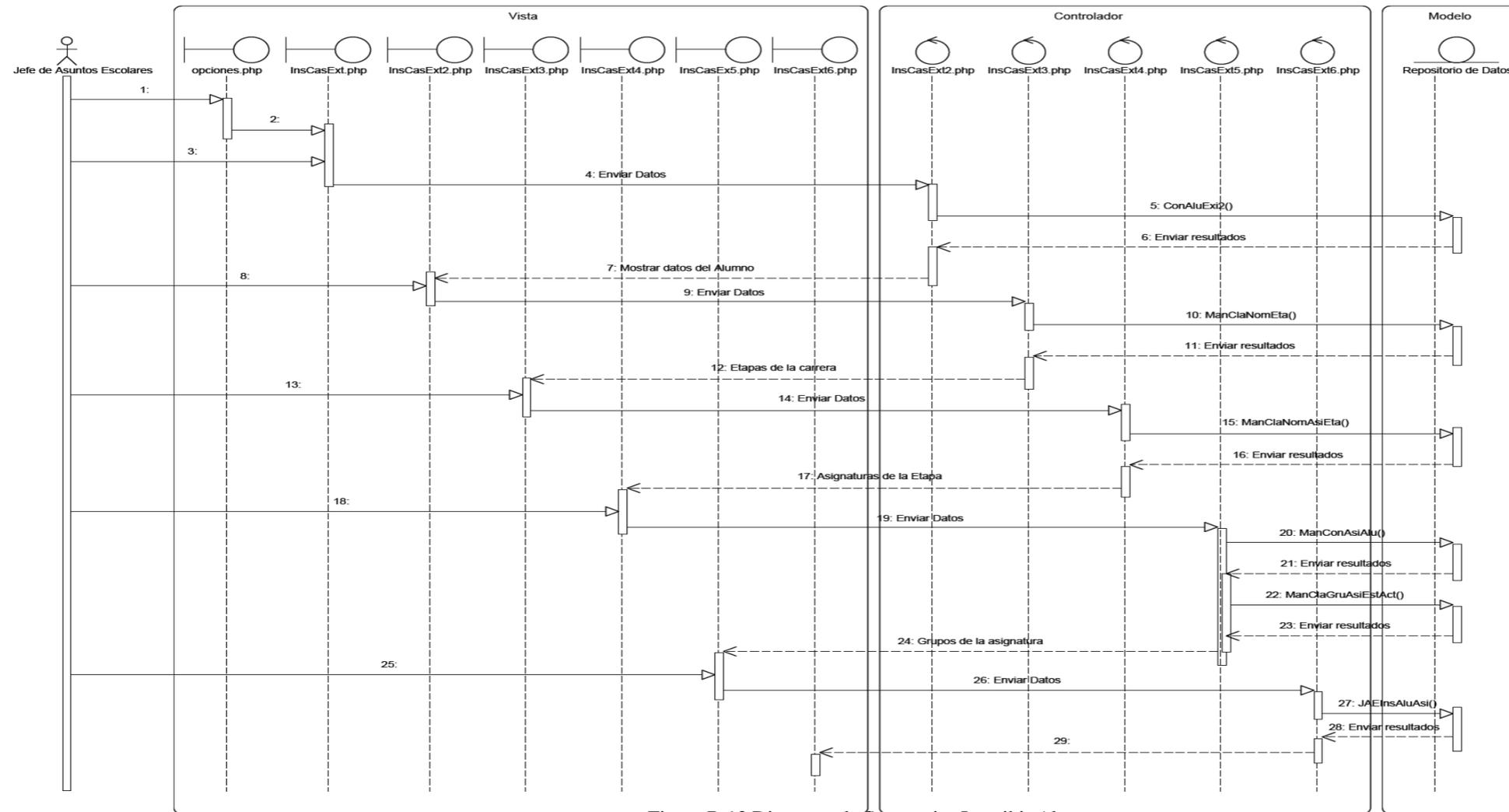
## Coordinador de Carrera Actualizar Cupo de un Taller



- Del escenario Ingresar a Trámites Académicos.  
 3. Consultar los Talleres correspondientes a la carrera de Arquitectura.  
 6. Seleccionar un Taller.  
 11. Seleccionar Etapa.  
 16. Elegir grupo.  
 21. Proporcionar nuevo cupo.  
 23. Actualizar cupo del grupo del Taller seleccionado.  
 25. Cupo del Taller elegido actualizado.

Figura B.12 Diagrama de Secuencias Actualizar cupo de un Taller

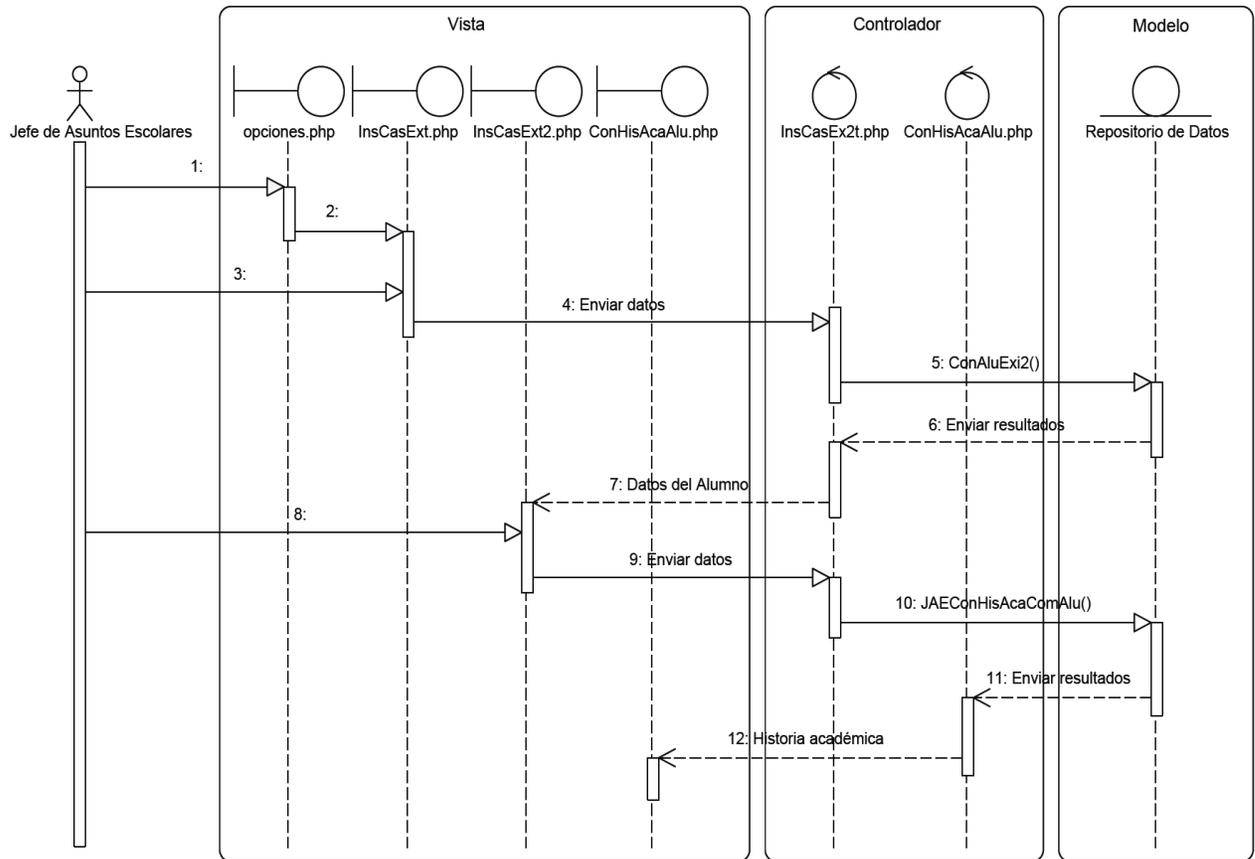
## Jefe de Asuntos Escolares Inscribir Alumno



- Del escenario Ingresar a Trámites Académicos.
- 2. Seleccionar Inscribir cursos.
- 3. Ingresar número de cuenta.
- 8. Elegir Datos son correctos.
- 18. Elegir una Etapa.
- 20. Establecer tipo de grupo (ordinario/extraordinario)
- 25. Seleccionar Asignatura.
- 27. Elegir un grupo para inscribir Alumno.
- 27. Insertar un registro del Alumno al grupo seleccionado.
- Actualizar el cupo del grupo.
- 29. Alumno inscrito.

Figura B.13 Diagrama de Secuencias Inscribir Alumno

## Consultar Historial Académico



- 2. Seleccionar Inscribir cursos.
- 3. Ingresar número de cuenta.
- 7. Mostrar datos del Alumno.
- 8. Elegir Consultar Historial Académico.
- 12. Mostrar Historia Académica del Alumno.

Figura B.14 Diagrama de Secuencias Consultar Historial Académico

# Anexo C

## Diagramas de Actividades

Alumno

Registro de asignaturas

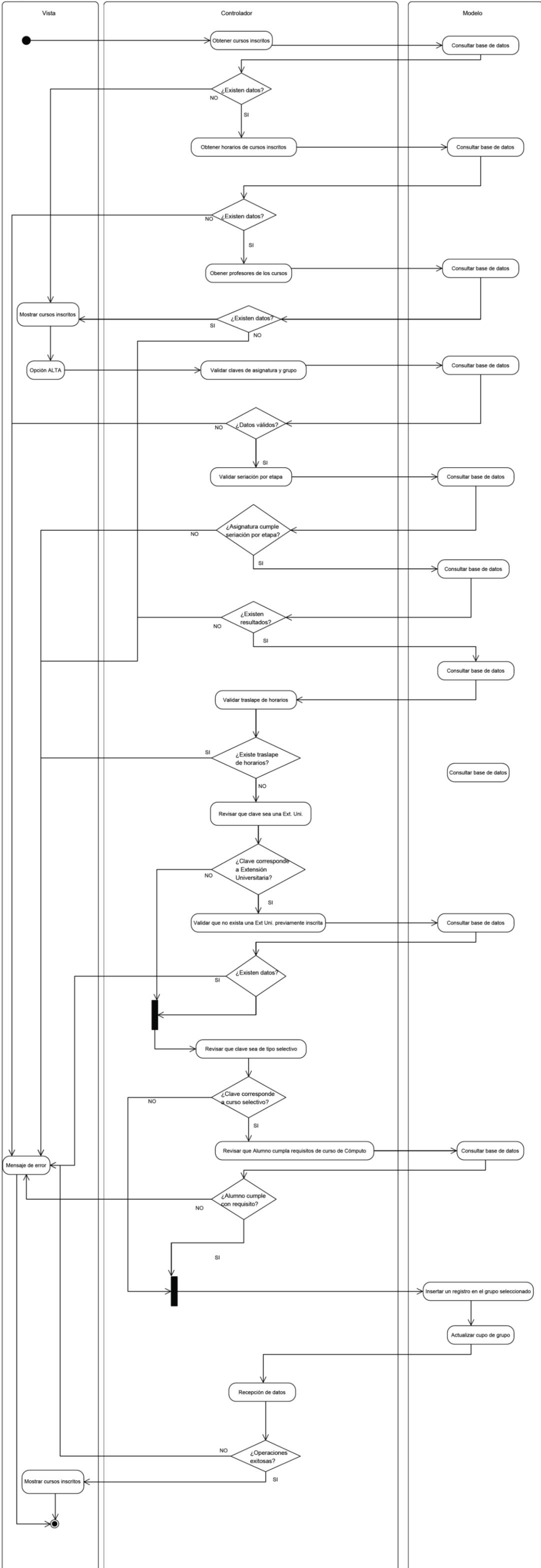


Figura C.1 Diagrama de Actividades Registro de Asignaturas

# Altas, Bajas y Cambios

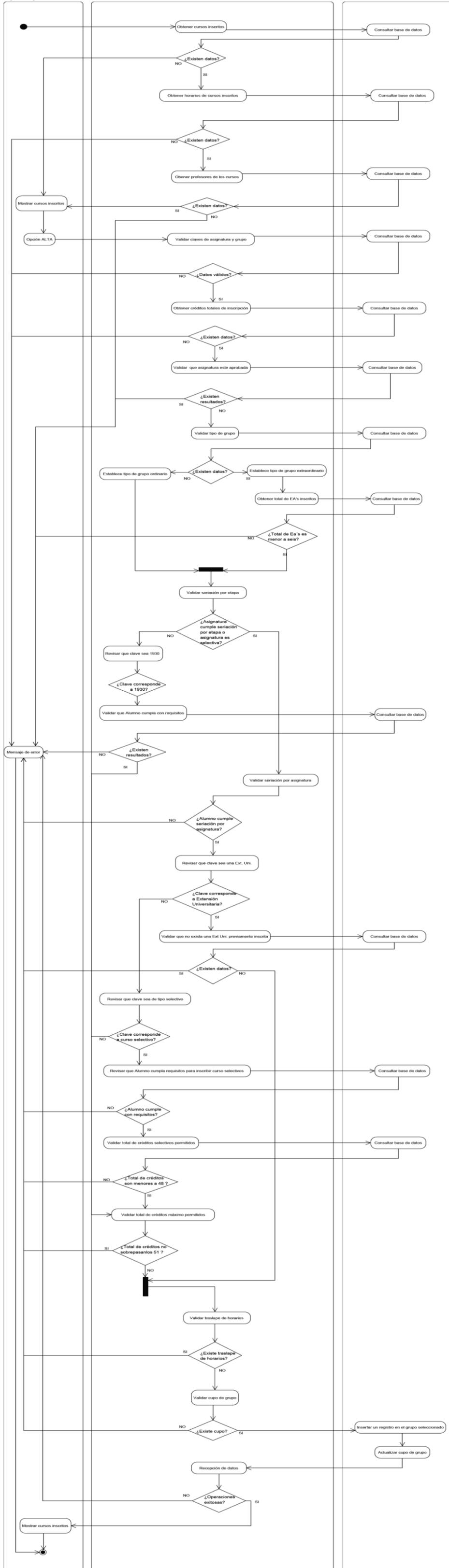


Figura C.2 Diagrama de Actividades Altas, Bajas y Cambios

## Actualizar datos

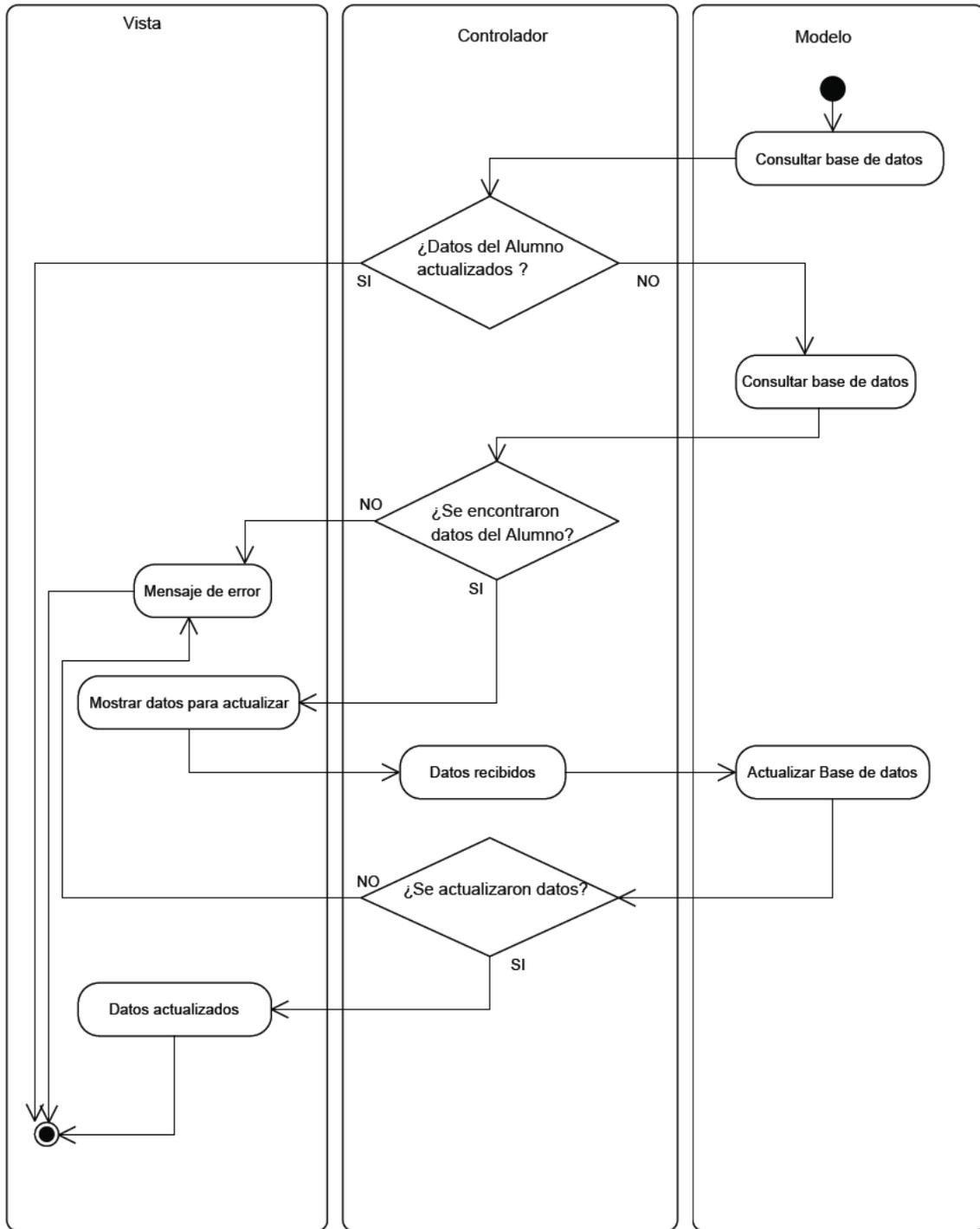


Figura C.3 Diagrama de Actividades Actualizar datos

**Mantenimiento**  
**Cambiar calificación del Alumno**

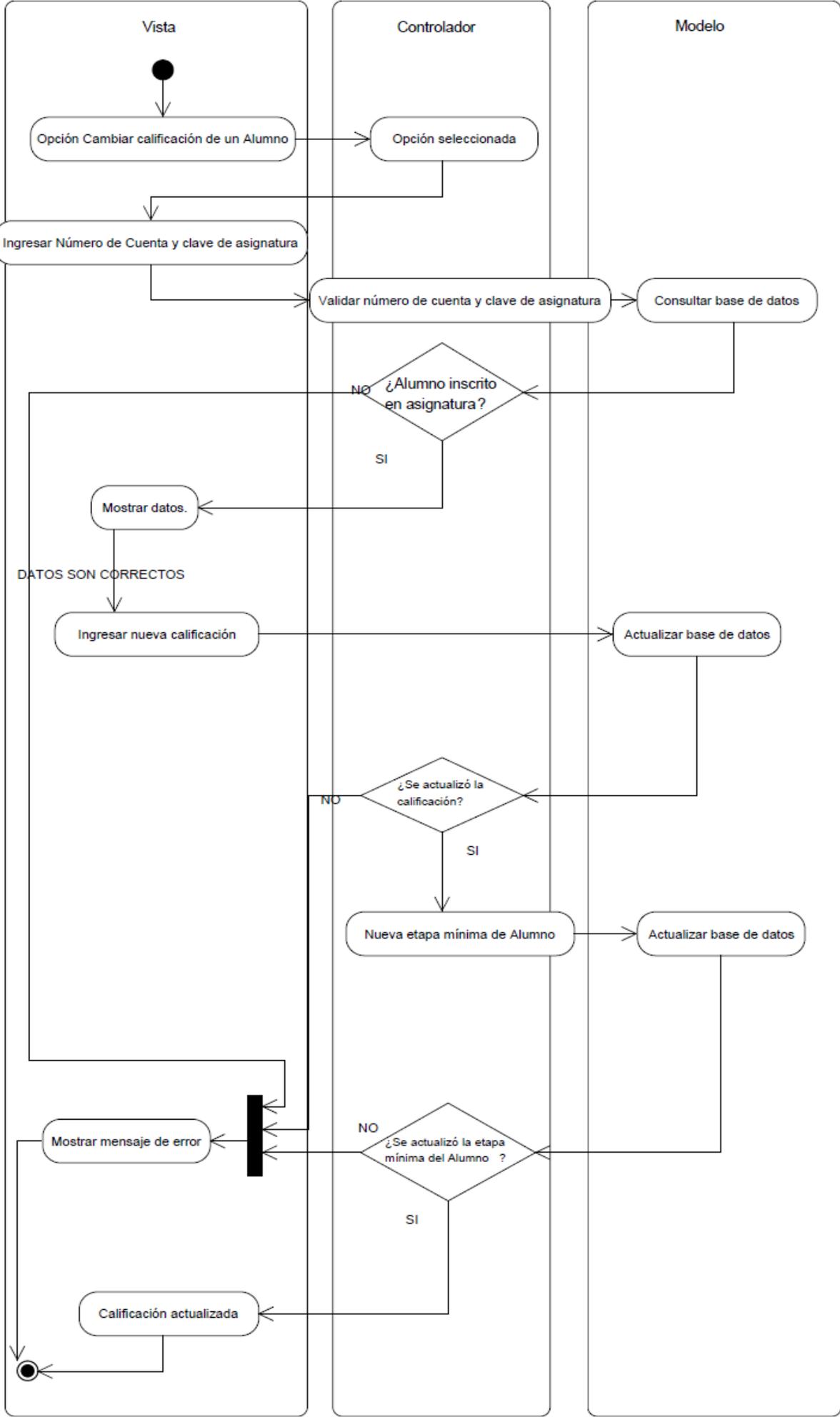


Figura C.4 Diagrama de Actividades Cambiar calificación de un Alumno

### Cancelar grupos de cursos

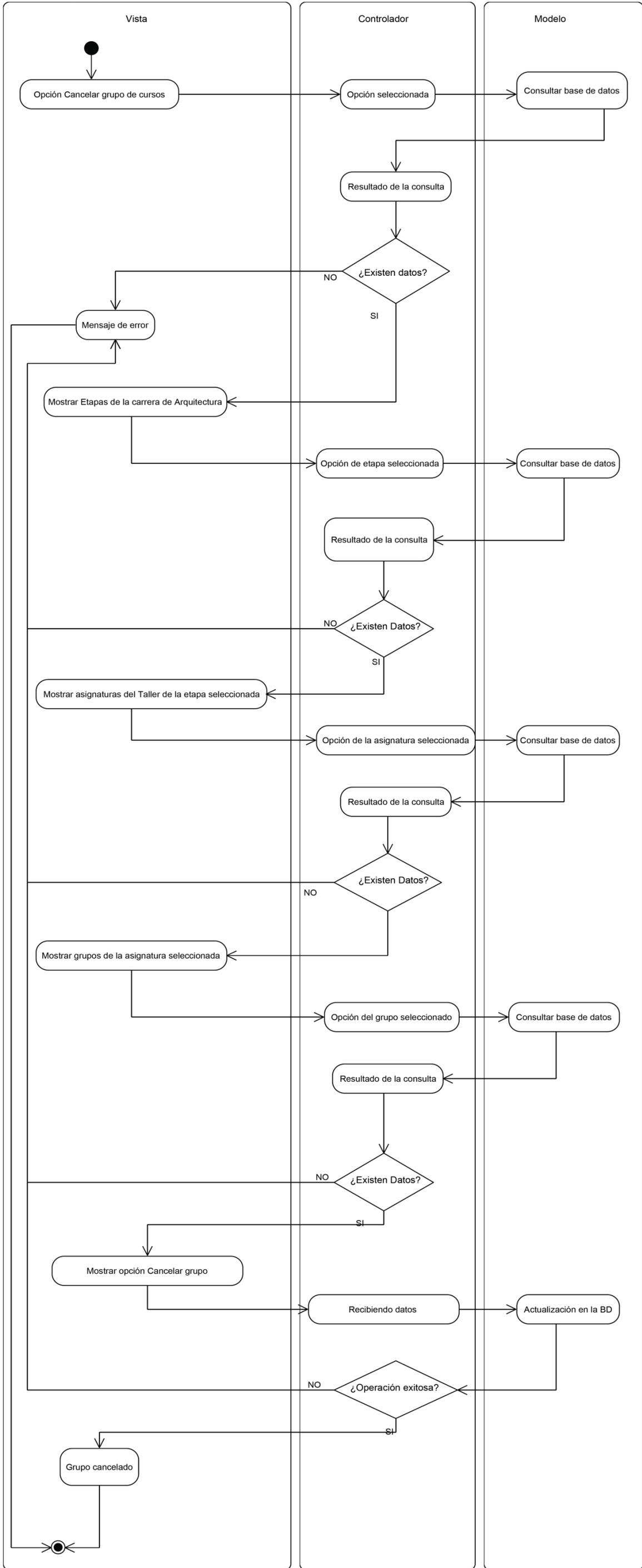


Figura C.5 Diagrama de Actividades Cancelar grupos de cursos

# Modificar Cupo de cursos

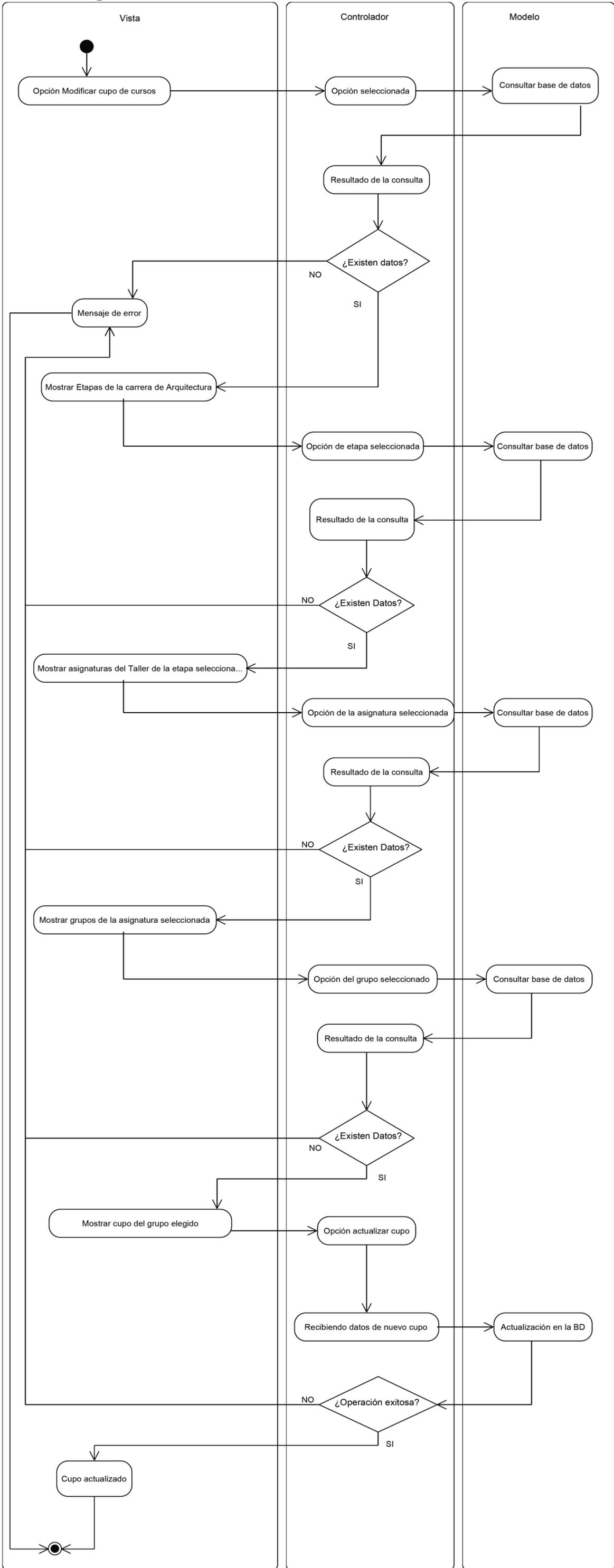


Figura C.6 Diagrama de Actividades Modificar cupo de cursos

## Coordinador de Taller Cancelar Inscripción al Taller

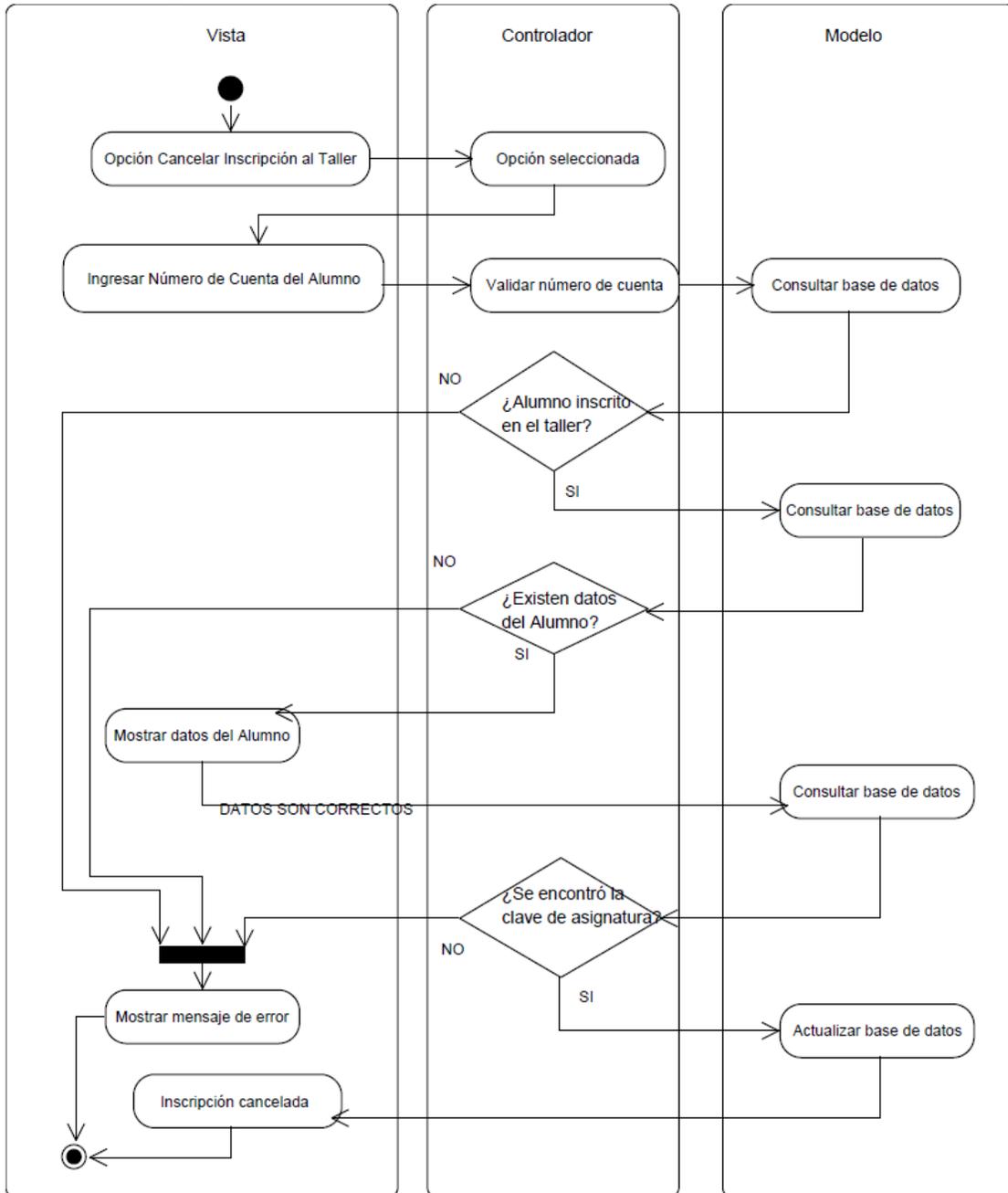


Figura C.7 Diagrama de Actividades Cancelar inscripción al Taller

**Coordinador de Carrera  
Actualizar cupo de Taller**

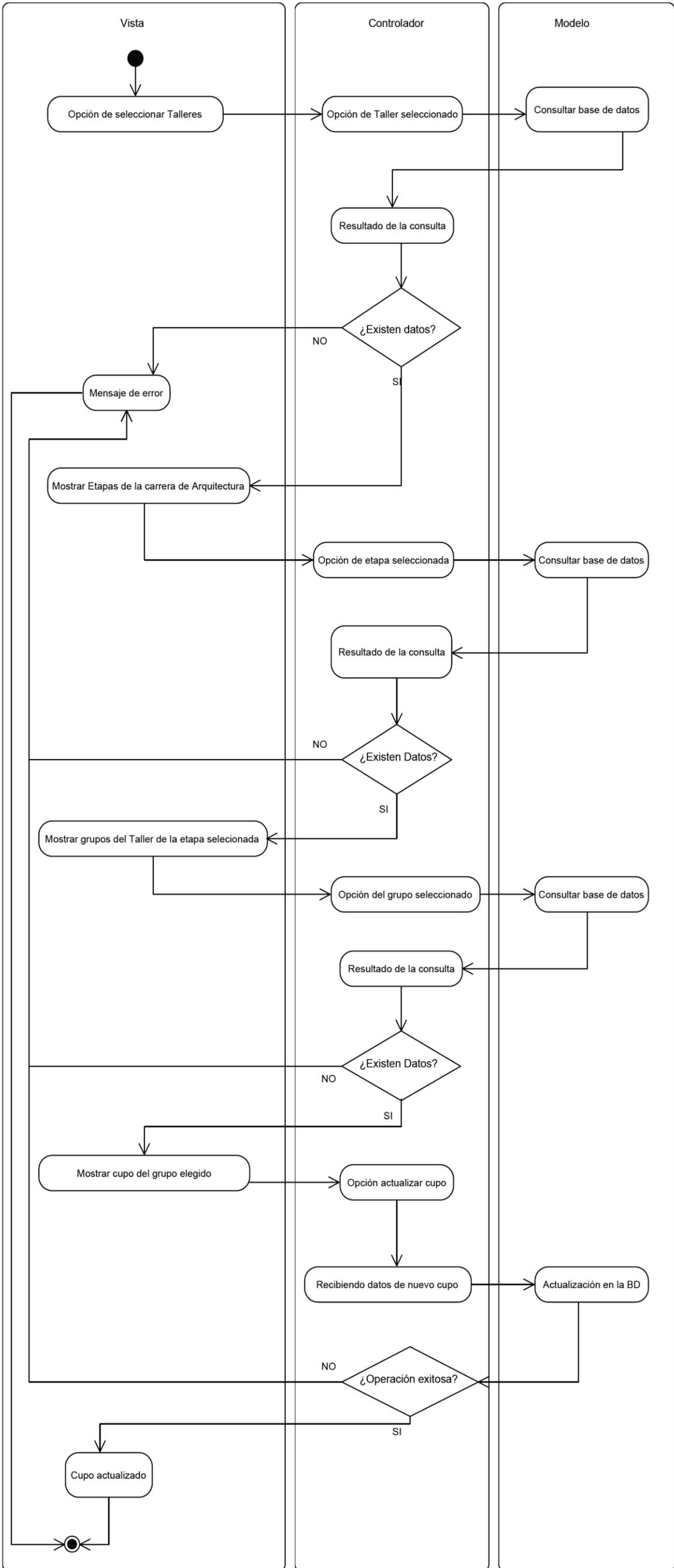


Figura C.8 Diagrama de Actividades Actualizar cupo de Taller



## Consulta historial académico del Alumno

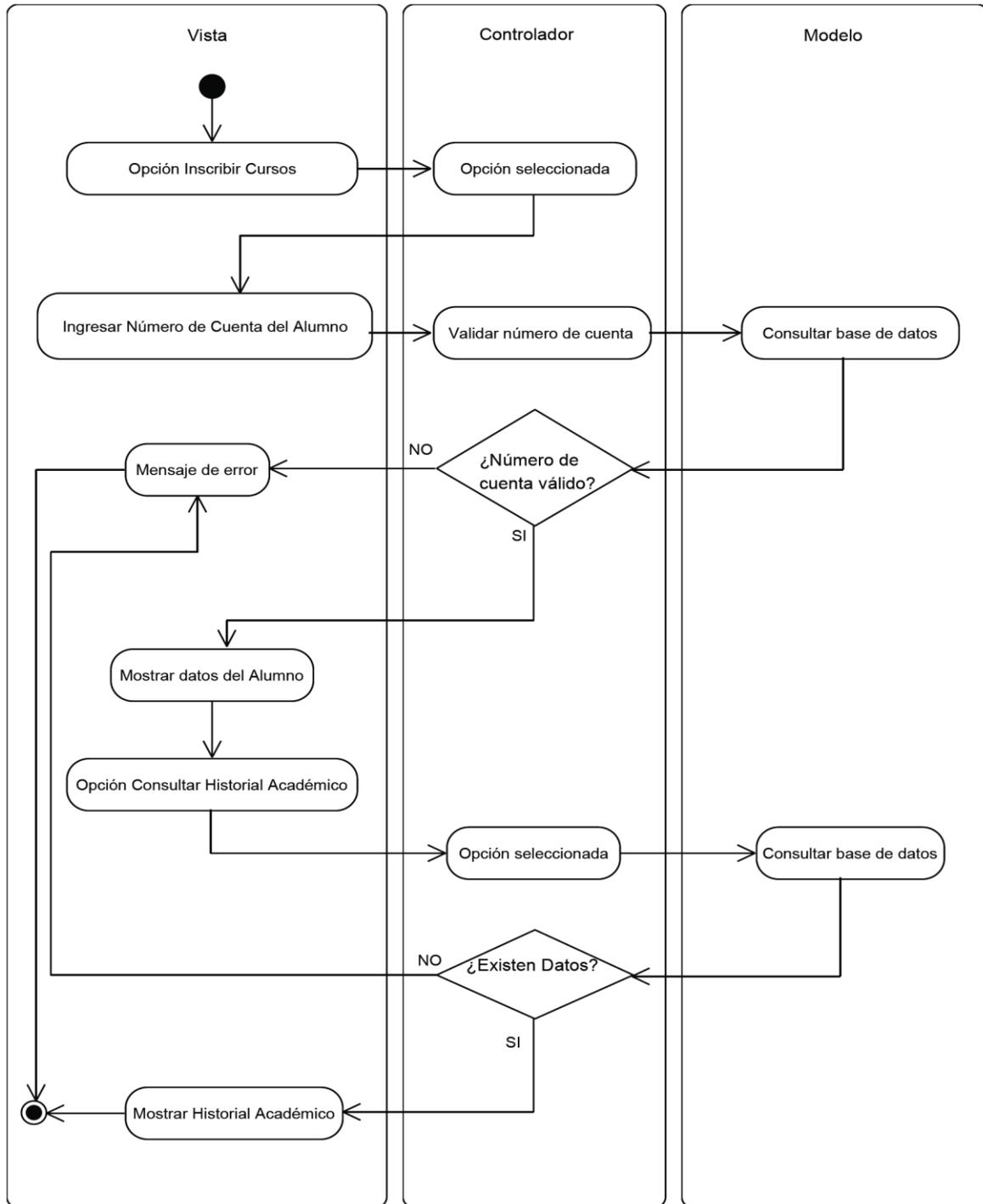


Figura C.10 Diagrama de Actividades Consultar Historial Académico del Alumno

# Anexo D

## Vistas

### Secuencia de pantallas

#### Alumno

#### Registrar asignaturas

Nombre: **index.html**

Página anterior: ninguna

Página siguiente: IngresoTramitesEscolaresAlumno.php

Descripción: Página que permite dirigir el ingreso de los actores de Asuntos Escolares a sus respectivos formularios de autenticación.

Pantalla:

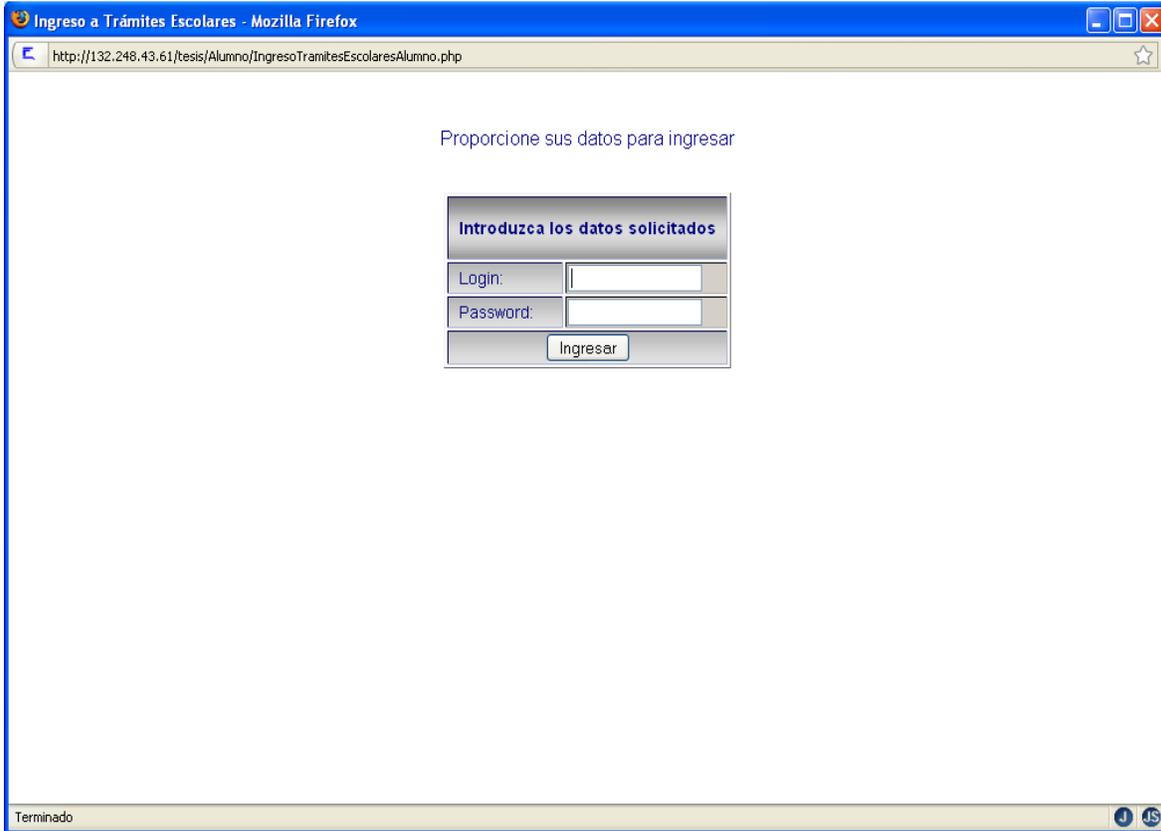


Campos:

Campo	Representación
Actores Asuntos Escolares	Link
Alumno	Link

Nombre: **IngresoTramitesEscolaresAlumno.php**  
 Página anterior: index.html  
 Página siguiente: registroAlumno.php  
 Descripción: Página que sirve para que el Alumno ingrese sus datos para su autenticación.

Pantalla:



Campos:

Campo	Representación
Login	Text Field
Password	Text Field
Ingresar	Button

Nombre: **registroAlumno.php**  
 Página anterior: IngresoTramitesEscolaresAlumno.php  
 Página siguiente: registroAlumno.php  
 Descripción: Página que sirve para que el Alumno ingrese los datos de las asignaturas y grupos en que desea registrarse, así como para mostrarle los datos de aquellos en que ya se registró.

Pantalla:

Campos:

Campo	Representación
Clave de la asignatura	Text Field
Grupo	Text Field
Alta	Button
Baja	Button
Imprimir Comprobante	Link
Cerrar sesión	Link

## Altas, Bajas y cambios

Nombre: **ABCAlumno.php**

Página Anterior: **IngresoTramitesEscolares.php**

Página Siguiente: **ABCAlumno.php**

Descripción: Página en donde se muestran los cursos asignados al Alumno previamente por el sistema en el periodo correspondiente a Asignación, así como las opciones para realizar altas, bajas y cambios de cursos.

### Pantalla:

Clave	Asignatura	Grupo	Profesor	Aula	Créditos	Horario							
						L	M	J	V	S	Hora inicio	Hora fin	
1333	TALLER DE ARQUITECTURA III	5316	Ver profesores	C102	25		*					07:00	15:00
						*						07:00	14:00
									*			09:00	14:00
1131	TEORIA DE LA ARQUITECTURA I	5121	Ver profesores	F101	4							00:00	00:00
1133	MATEMATICAS APLICADAS I	5129	Ver profesores	F101	4		*					09:00	11:00
									*			07:00	09:00
1130	INTRODUCCION HISTORIC CRITIC	5131	Ver profesores	H201	4			*				09:00	11:00
1134	SISTEMAS ESTRUCTURALES I	5129	Ver profesores	C102	6		*					09:00	12:00
1233	MATEMATICAS APLICADAS II	5232	Ver profesores	C201	4				*			18:00	20:00
1231	TEORIA DE LA ARQUITECTURA II	5235	Ver profesores	E002	4		*					07:00	09:00
1230	ARQUITECTURA EN MEXICO S.XX	5221		E201	4	*						07:00	09:00

### Campos:

Campos	Representación
Clave de asignatura	Text Field
Grupo	Text Field
Alta	Button
Baja	Button
Imprimir comprobante	Link
Cerrar sesión	Link

## Alumno actualizar datos

Nombre: **ActDatAlu.php**  
Página Anterior: IngresoTramitesEscolares.php  
registroAlumno.php  
Página Siguiente: ActDatAlu2.php  
Descripción: Página en donde se muestra la información personal del Alumno.

Pantalla:

Actualizar datos de Alumno - Mozilla Firefox

http://132.248.43.61/tesis/Alumno/actualizaDatosAlumno.php

Debes actualizar tus datos para poder continuar

Alumno: BONILLA GARCIA JOSE LUIS      Número de cuenta: 301153689  
DOMICILIO

Calle  
AV. UNIVERSIDAD

Colonia  
COPILCO

Número exterior: 404      Número interior:

Delegación o Municipio

Entidad Federativa DISTRITO FEDERAL

Código Postal 1211

Teléfono  
435345

Teléfono celular:  
044

E-mail

Terminado

Campos:

Campos	Representación
Calle	Text Field
Colonia	Text Field
Número exterior	Text Field
Número interior	Text Field
Delegación o Municipio	Text Field
Entidad Federativa	Text Field
Código Postal	Text Field
Teléfono	Text Field
Teléfono Celular	Text Field
E-mail	Text Field
Actualizar	Button

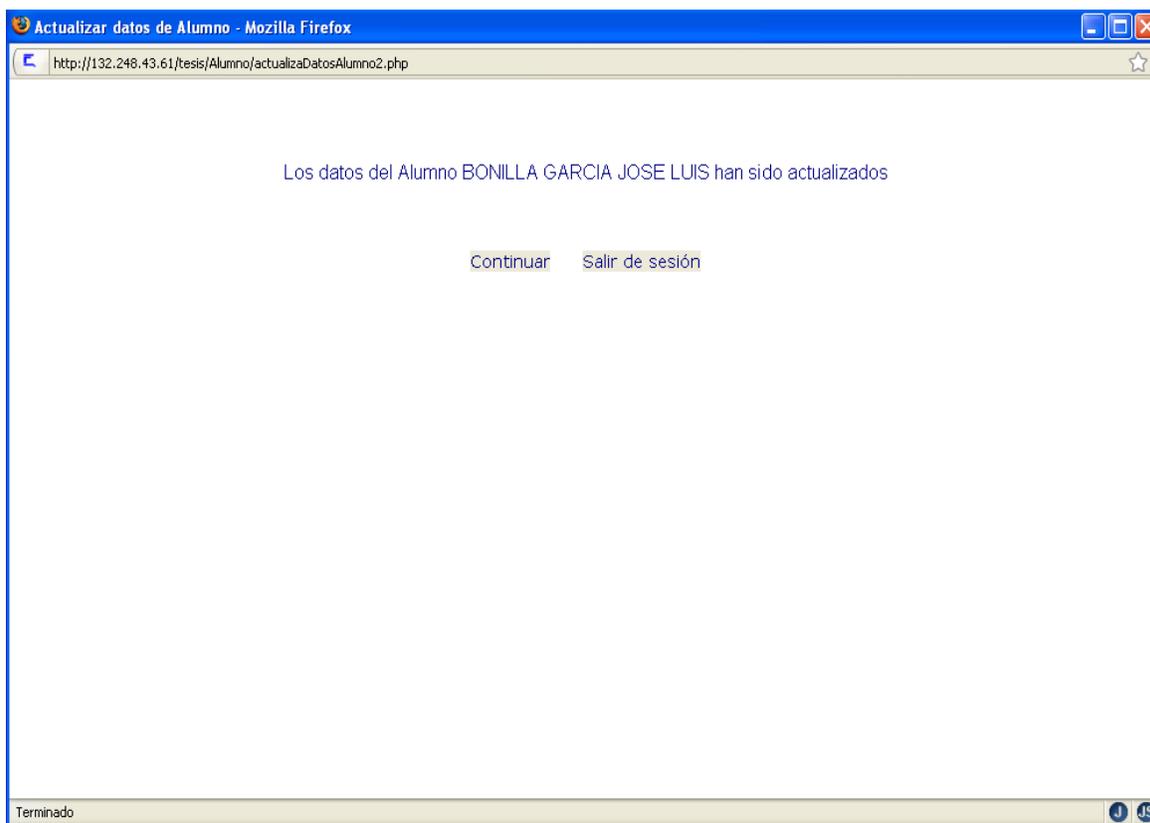
Nombre: **ActDatAlu2.php**

Página Anterior: ActDatAlu.php

Página Siguiente: Ninguna  
registroAlumno.php

Descripción: Página en donde se muestra que los datos del Alumno han sido actualizados.

Pantalla:

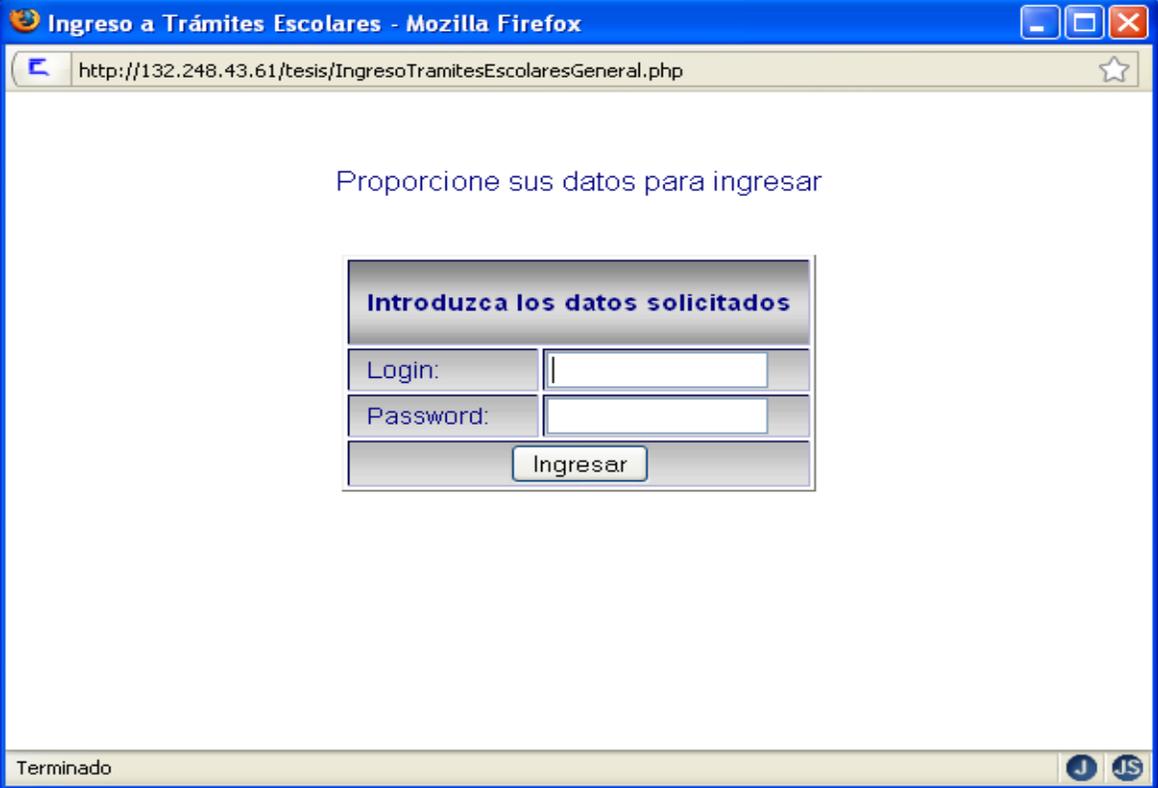


Campos:

Campos	Representación
Continuar	Link
Salir de sesión	Link

## Mantenimiento

Nombre: **IngresoTramitesEscolaresGeneral.php**  
Página anterior: index.html  
Página siguiente: opciones.php  
Descripción: Página que permite la autenticación de los actores de asuntos escolares.  
Pantalla:



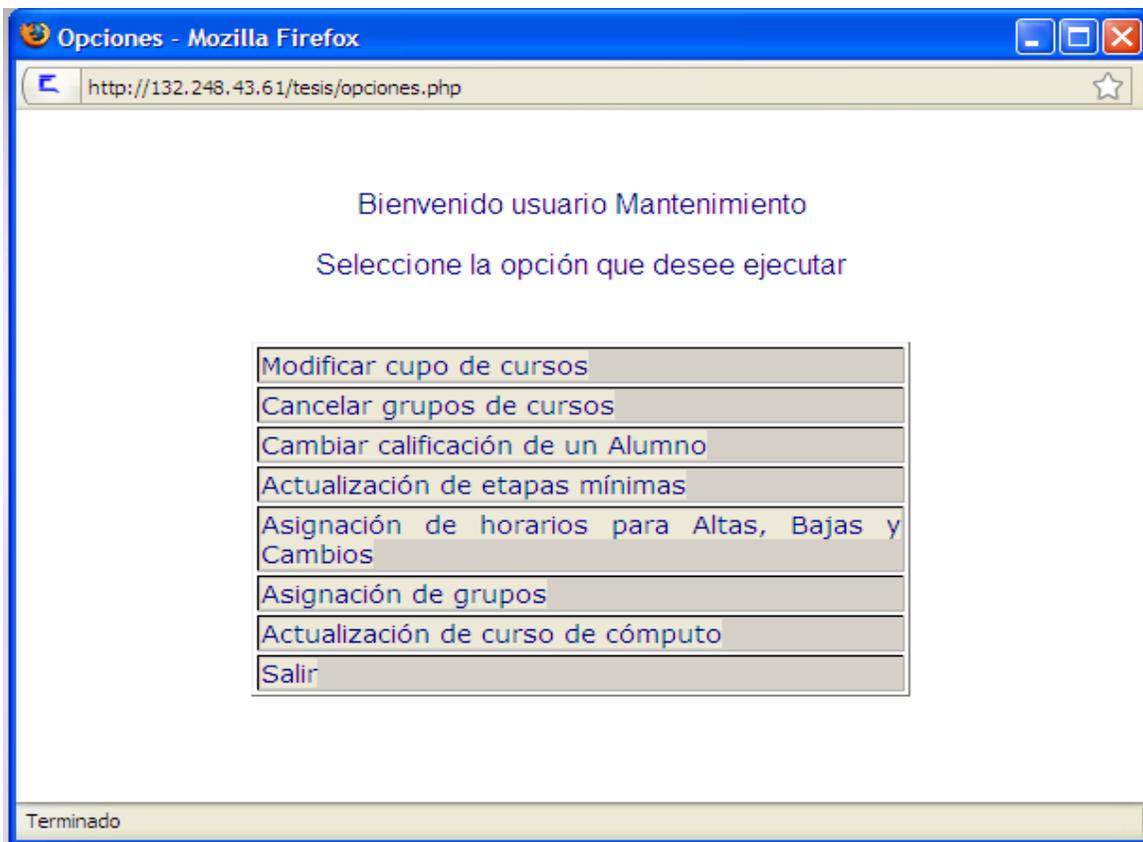
The screenshot shows a Mozilla Firefox browser window titled "Ingreso a Trámites Escolares - Mozilla Firefox". The address bar contains the URL "http://132.248.43.61/tesis/IngresoTramitesEscolaresGeneral.php". The main content area displays the text "Proporcione sus datos para ingresar" and a form titled "Introduzca los datos solicitados". The form includes a "Login:" label with a text input field, a "Password:" label with a text input field, and an "Ingresar" button. The status bar at the bottom indicates "Terminado" and shows Java and JavaScript icons.

Campos:

Campo	Representación
Login	Text Field
Password	Text Field
Ingresar	Button

## Menú general para Mantenimiento

Nombre: **opciones.php**  
Página Anterior: IngresoTramitesEscolares.php  
Página Siguiente: ModCup.php  
CanGru.php  
CamCal.php  
AsiGru.php  
ActCurComInd.php  
Descripción: Página que sirve para que el actor Mantenimiento elija la acción que desea ejecutar.  
Pantalla:



Campos:

Campo	Representación
Modificar cupo de cursos	Link
Cancelar grupos de cursos	Link
Cambiar calificación de un Alumno	Link
Actualización de etapas mínimas	Link
Asignación de horarios para Altas, Bajas y Cambios	Link
Asignación de grupos	Link
Actualización de curso de cómputo	Link
Salir	Link

## Modificar cupo de cursos

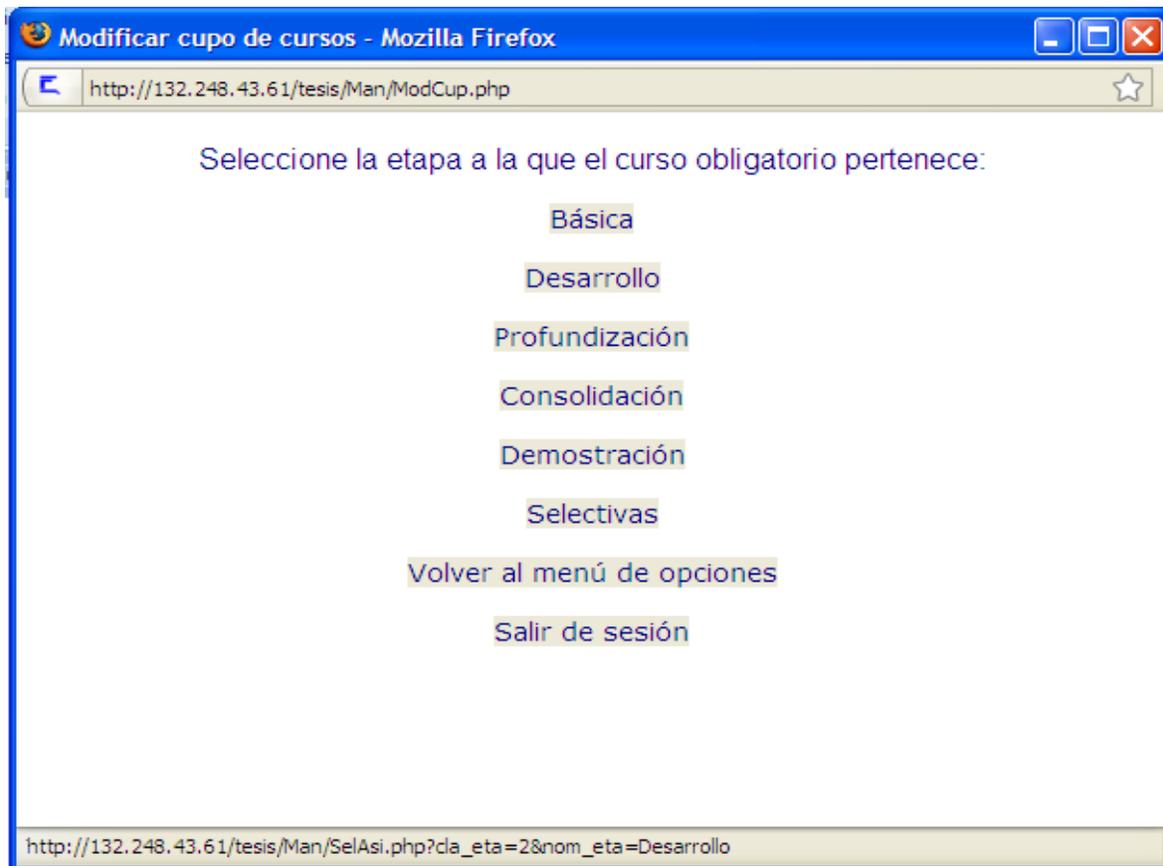
Nombre: **ModCup.php**

Página Anterior: opciones.php

Página Siguiente: SelAsi.php  
opciones.php  
salir.php

Descripción: Página que despliega las etapas correspondientes a la carrera de Arquitectura, así como la etapa de cursos selectivos.

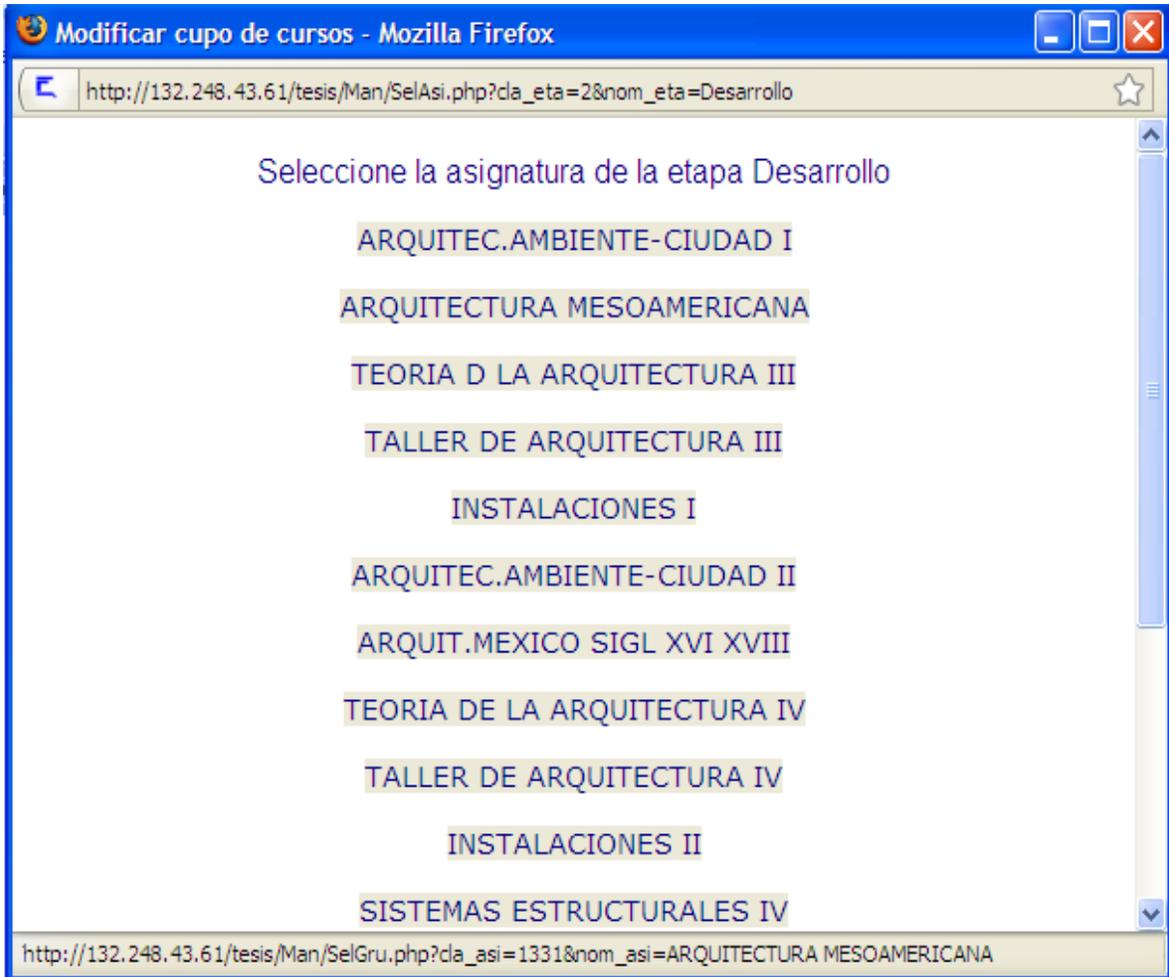
Pantalla:



Campos:

Campos	Representación
Básica	Link
Desarrollo	Link
Consolidación	Link
Demostración	Link
Selectivas	Link
Volver al menú de opciones	Link
Salir de sesión	Link

Nombre: **SelAsi.php**  
Página Anterior: ModCup.php  
Página Siguiente: SelGru.php  
opciones.php  
salir.php  
Descripción: Página en donde se muestran los cursos de la etapa elegida.  
Pantalla:



Campos:

Campos	Representación
ARQUITEC-AMBIENTE CIUDAD I	<a href="#">Link</a>
ARQUITECTURA MESOAMERICANA	<a href="#">Link</a>
TEORÍA DE LA ARQUITECTURA III	<a href="#">Link</a>
TALLER DE ARQUITECTURA III	<a href="#">Link</a>
INSTALACIONES I	<a href="#">Link</a>
ARQUITEC-AMBIENTE CIUDAD II	<a href="#">Link</a>
ARQUIT. MEXICO SIGLOS XVI XVIII	<a href="#">Link</a>
TEORIA DE LA ARQUITECTURA IV	<a href="#">Link</a>
TALLER DE ARQUITECTURA IV	<a href="#">Link</a>
INSTALACIONES II	<a href="#">Link</a>
SISTEMAS ESTRUCTURALES IV	<a href="#">Link</a>
Volver al menú de opciones	<a href="#">Link</a>
Salir de sesión	<a href="#">Link</a>

Nombre: **SelGru.php**  
Página Anterior: SelAsi.php  
Página Siguiente: ConCup.php  
opciones.php  
salir.php

Descripción: Página que muestra los grupos ordinarios así como los extraordinarios de la asignatura seleccionada.

Pantalla:



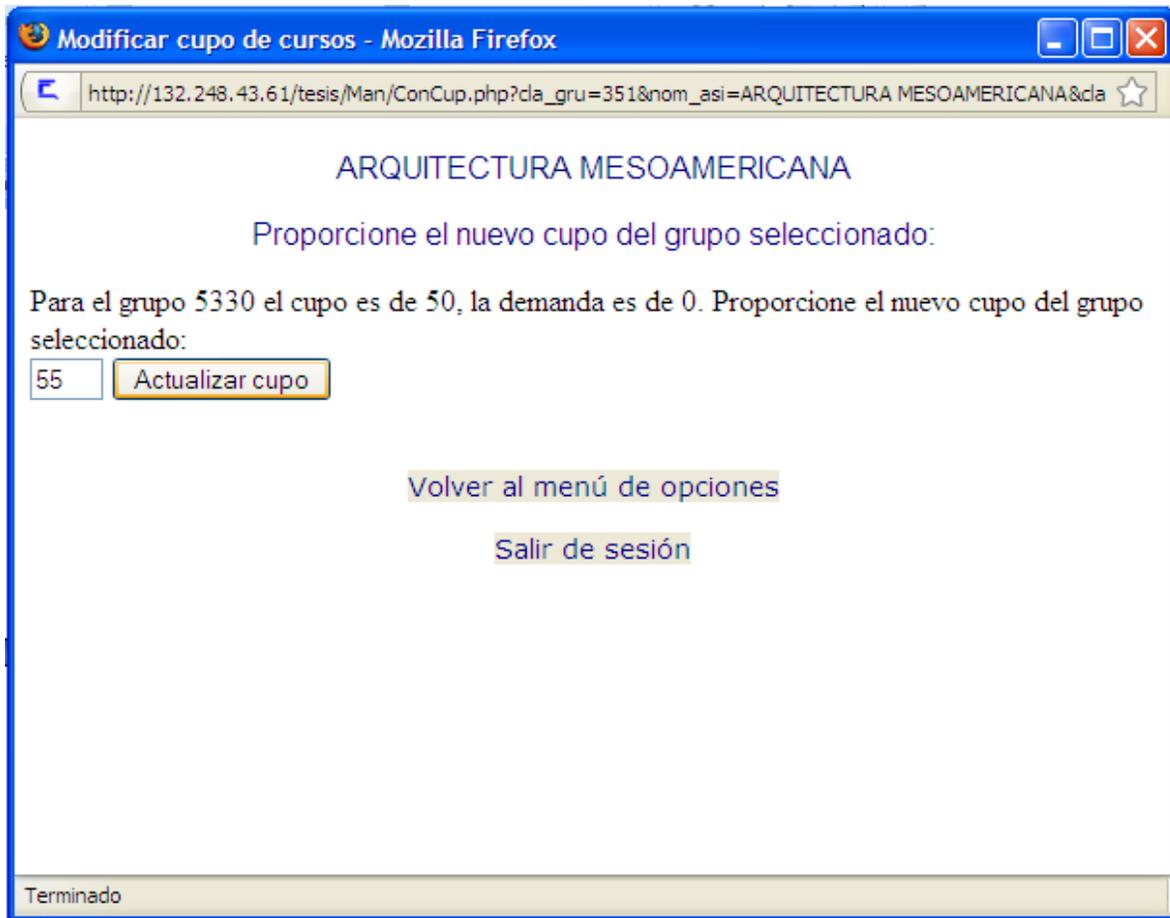
Campos:

Campo						Representación
INTRODUCCIÓN extraordinario EA00	HISTÓRICA	CRÍTICA	grupo	1000	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA30	HISTÓRICA	CRÍTICA	grupo	5330	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA23	HISTÓRICA	CRÍTICA	grupo	5323	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA36	HISTÓRICA	CRÍTICA	grupo	5336	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA34	HISTÓRICA	CRÍTICA	grupo	5334	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA24	HISTÓRICA	CRÍTICA	grupo	5324	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA27	HISTÓRICA	CRÍTICA	grupo	5327	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA25	HISTÓRICA	CRÍTICA	grupo	5325	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA33	HISTÓRICA	CRÍTICA	grupo	5333	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA22	HISTÓRICA	CRÍTICA	grupo	5322	grupo	<a href="#">Link</a>
INTRODUCCIÓN extraordinario EA26	HISTÓRICA	CRÍTICA	grupo	5326	grupo	<a href="#">Link</a>
Volver al menú de opciones						<a href="#">Link</a>
Salir de sesión						<a href="#">Link</a>

Nombre: **ConCup.php**  
Página Anterior: SelGru.php  
Página Siguiente: ActCup.php  
opciones.php  
salir.php

Descripción: Página en donde se muestran la información del grupo seleccionado así como un campo para proporcionar el nuevo cupo del mismo.

Pantalla:



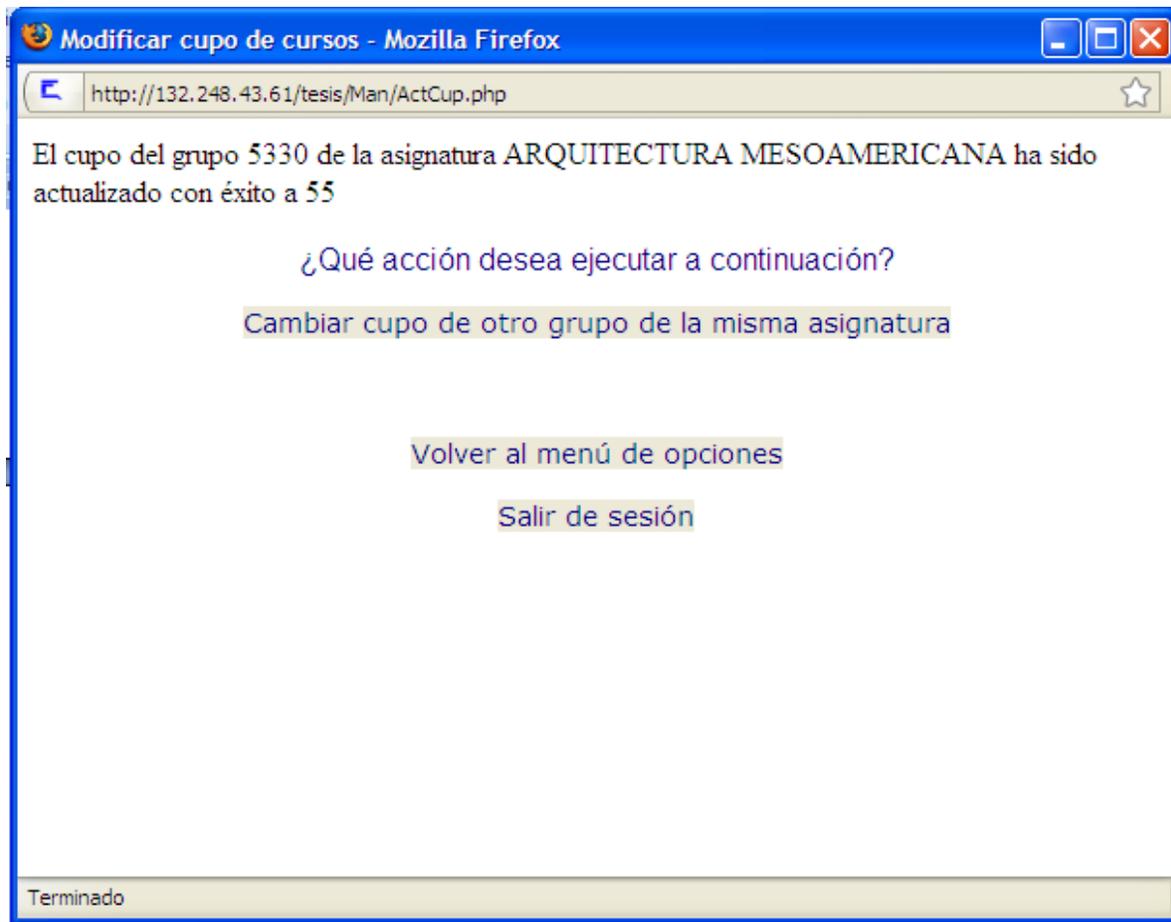
Campos:

Campo	Representación
Cupo	Text Field
Actualizar Cupo	Button
Volver al menú de opciones	Link
Salir de sesión	Link

Nombre: **ActCup.php**  
Página Anterior: ConCup.php  
Página Siguiente: SelGru.php  
opciones.php  
salir.php

Descripción: Página en donde se muestra que el cupo de la asignatura seleccionada ha sido actualizada.

Pantalla:



Campos:

Campo	Representación
Cambiar cupo de otro grupo de la misma asignatura	Link
Volver al menú de opciones	Link
Salir de sesión	Link

## Cancelar grupos de Cursos Obligatorios

Nombre: **CanGru.php**

Página Anterior: opciones.php

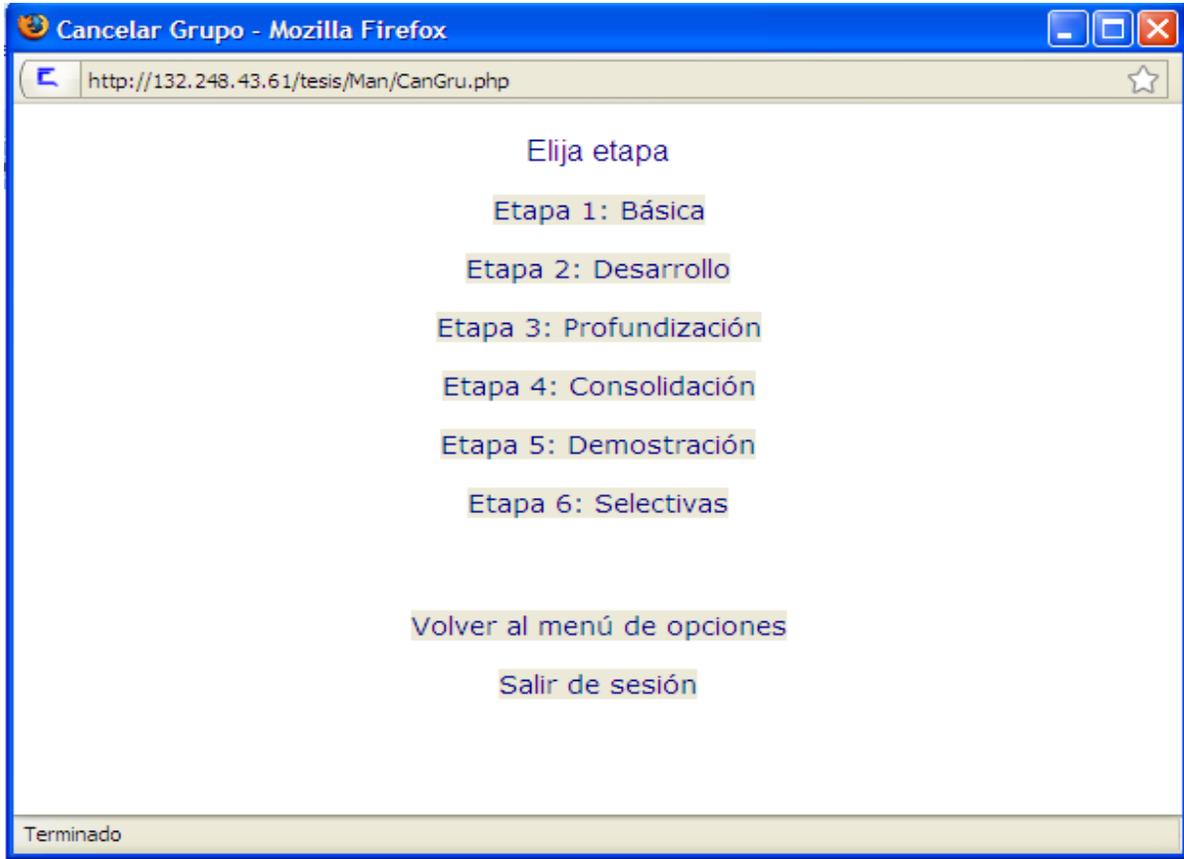
Página Siguiente: CanGru2.php

opciones.php

salir.php

Descripción: Página que despliega las etapas correspondientes a la carrera de Arquitectura.

Pantalla:



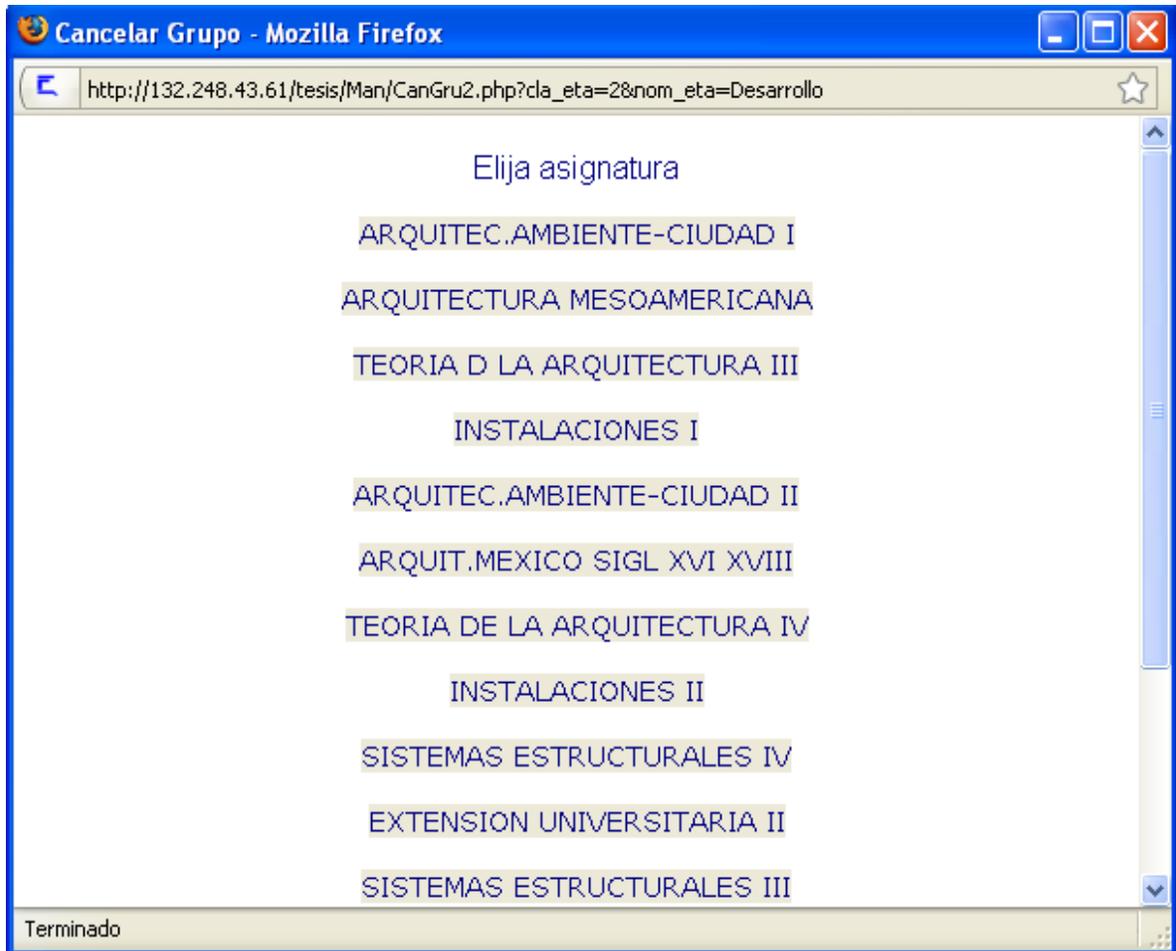
Campos:

Campo	Representación
Etapa 1: Básica	Link
Etapa 2: Desarrollo	Link
Etapa 3: Consolidación	Link
Etapa 4: Demostración	Link
Etapa 5: Selectivas	Link
Volver al menú de opciones	Link
Salir de sesión	Link

Nombre: **CanGru2.php**  
Página Anterior: CanGru.php  
Página Siguiente: CanGru3.php  
opciones.php  
salir.php

Descripción: Página en donde se muestran los cursos de la etapa seleccionada.

Pantalla:



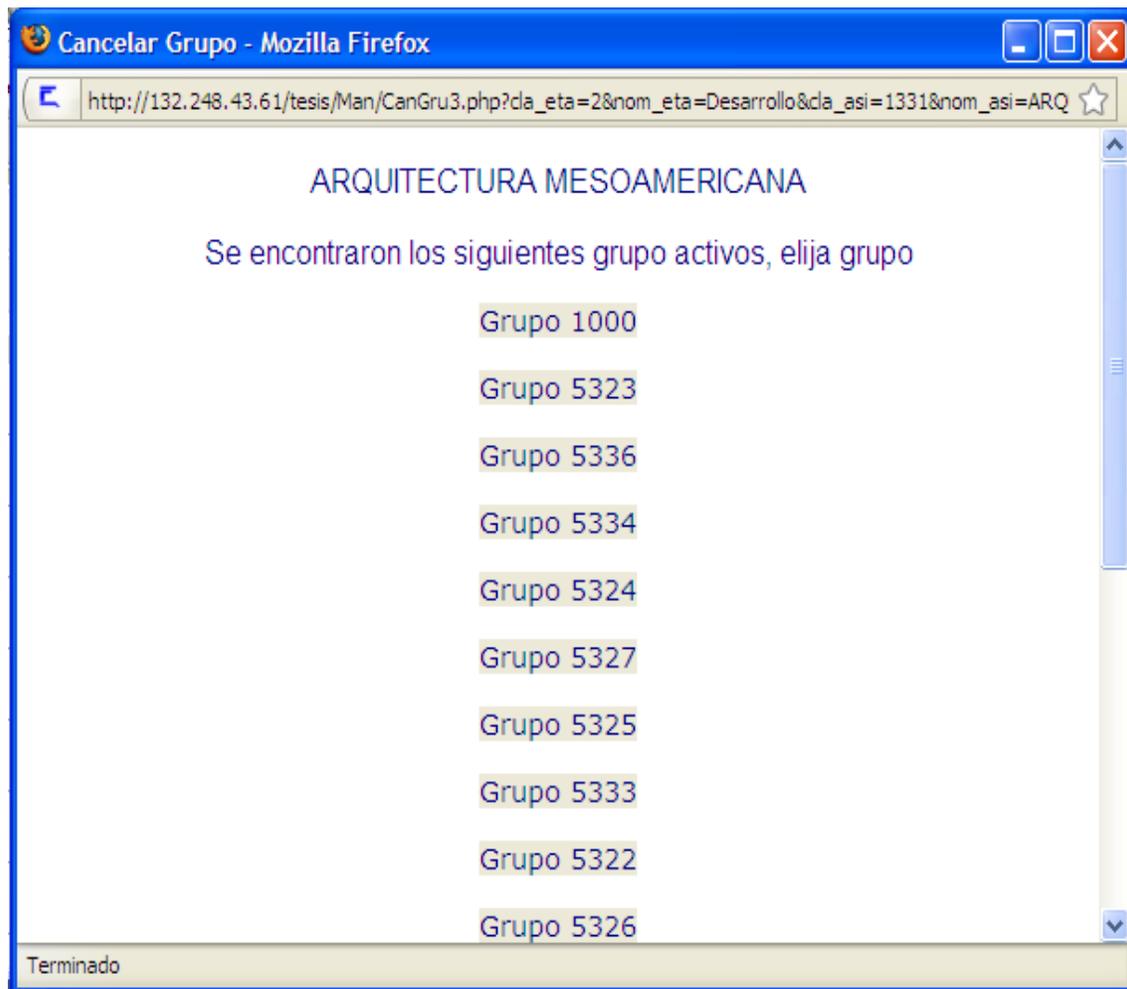
Campos:

Campos	Representación
ARQUITEC-AMBIENTE CIUDAD I	<a href="#">Link</a>
ARQUITECTURA MESOAMERICANA	<a href="#">Link</a>
TEORÍA DE LA ARQUITECTURA III	<a href="#">Link</a>
TALLER DE ARQUITECTURA III	<a href="#">Link</a>
INSTALACIONES I	<a href="#">Link</a>
ARQUITEC-AMBIENTE CIUDAD II	<a href="#">Link</a>
ARQUIT. MEXICO SIGLOS XVI XVIII	<a href="#">Link</a>
TEORIA DE LA ARQUITECTURA IV	<a href="#">Link</a>
TALLER DE ARQUITECTURA IV	<a href="#">Link</a>
INSTALACIONES II	<a href="#">Link</a>
SISTEMAS ESTRUCTURALES IV	<a href="#">Link</a>
EXTENSION UNIVERSITARIA II	
SISTEMAS ESTRUCTURALES III	<a href="#">Link</a>
<a href="#">Volver al menú de opciones</a>	<a href="#">Link</a>
<a href="#">Salir de sesión</a>	<a href="#">Link</a>

Nombre: **CanGru3.php**  
Página Anterior: CanGru2.php  
Página Siguiente: CanGru4.php  
opciones.php  
salir.php

Descripción: Página en donde se muestran los grupos activos de la asignatura elegida.

Pantalla:



Campos:

Campos	Representación
Grupo 1000	Link
Grupo 5323	Link
Grupo 5334	Link
Grupo 5324	Link
Grupo 5327	Link
Grupo 5325	Link
Grupo 5333	Link
Grupo 5322	Link
Grupo 5326	Link
Volver al menú de opciones	Link
Salir de sesión	Link

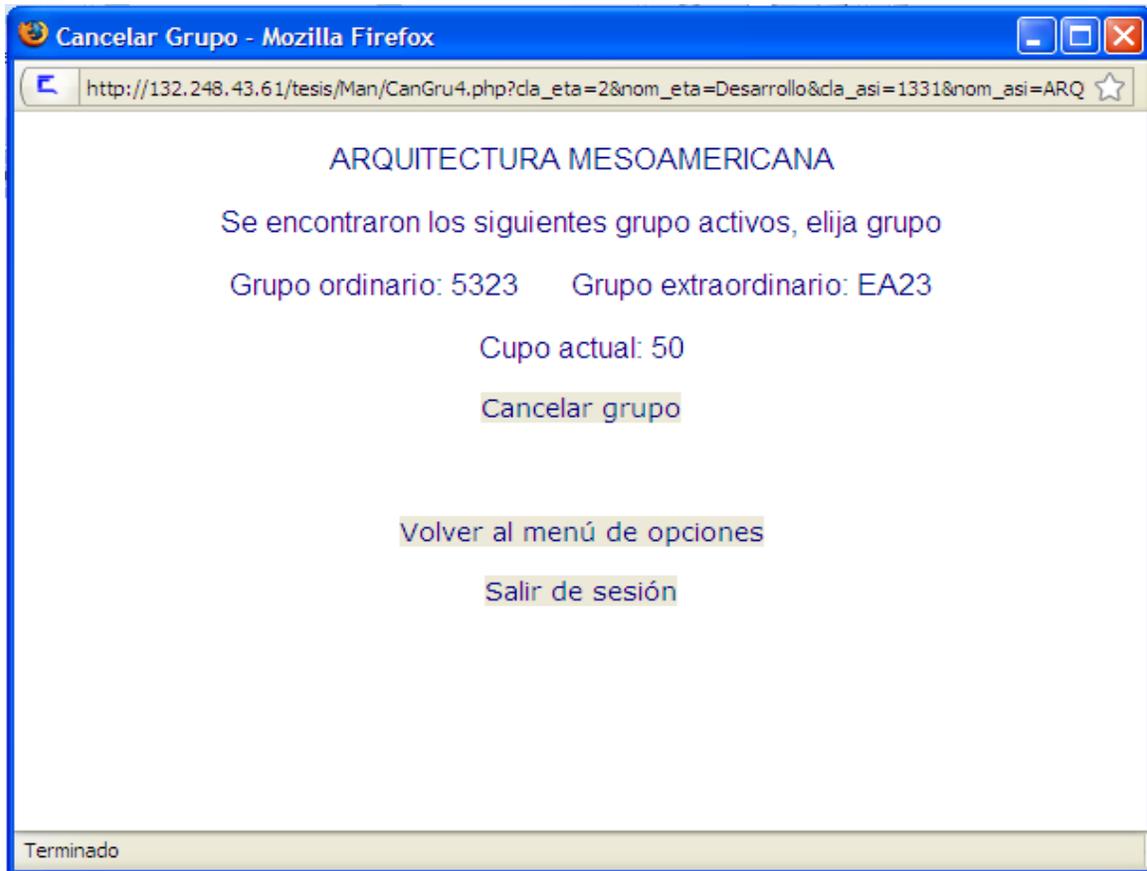
Nombre: **CanGru4.php**

Página Anterior: CanGru3.php

Página Siguiente: CanGru5.php  
opciones.php  
salir.php

Descripción: Página en donde se despliegan las claves de los grupos ordinarios así como de los extraordinarios del grupo seleccionado.

Pantalla:



Campos:

Campo	Representación
Cancelar grupo	Link
Volver al menú de opciones	Link
Salir de sesión	Link

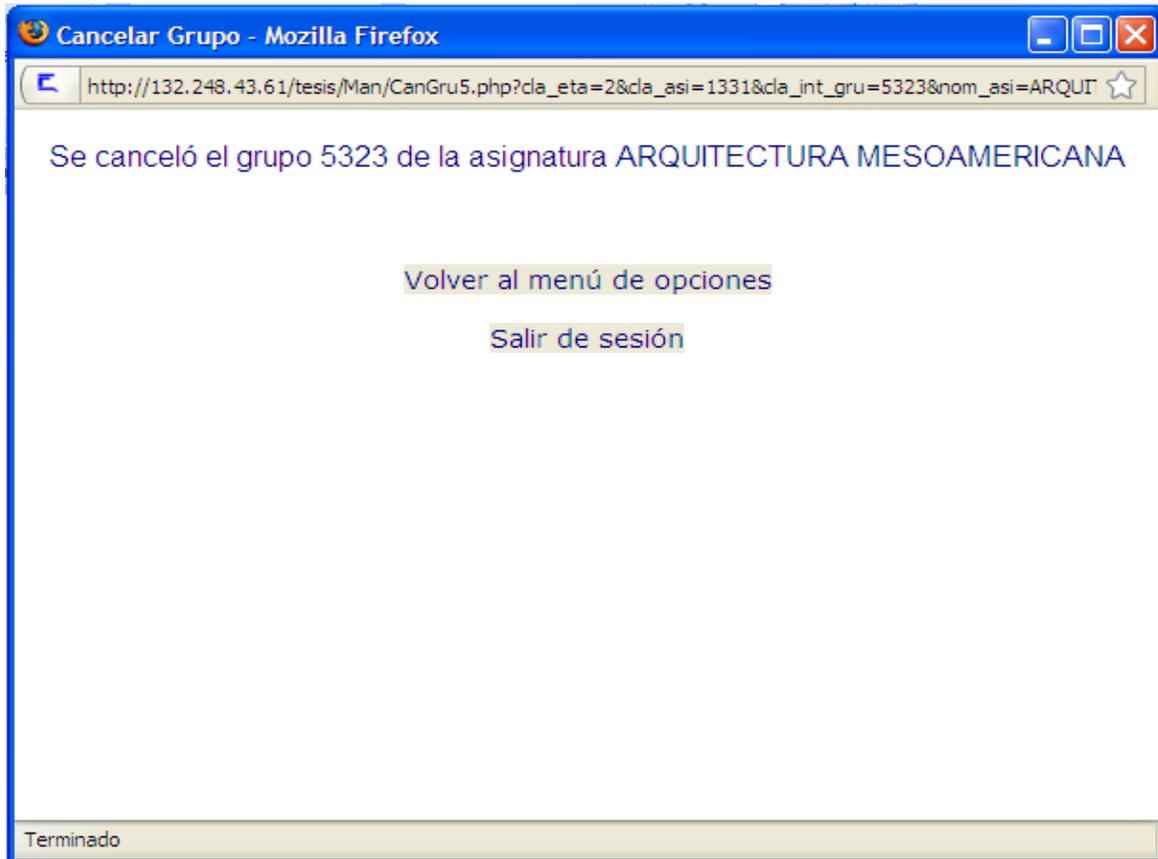
Nombre: **CanGru5.php**

Página Anterior: CanGru4.php

Página Siguiente: opciones.php  
salir.php

Descripción: Página en donde se muestra que el grupo seleccionado ha sido cancelado.

Pantalla:



Campos:

Campo	Representación
Volver al menú de opciones	Link
Salir de sesión	Link

## Cambiar calificación de un Alumno

Nombre: CamCal.php  
Página anterior: opciones.php  
Página siguiente: CamCal2.php  
Descripción: Página que recibe el número de cuenta del Alumno y la clave de la asignatura de la que se desea cambiar calificación.  
Pantalla:

Proporcione los datos que solicitan

Número de cuenta:

Clave de asignatura:

[Volver al menú de opciones](#)

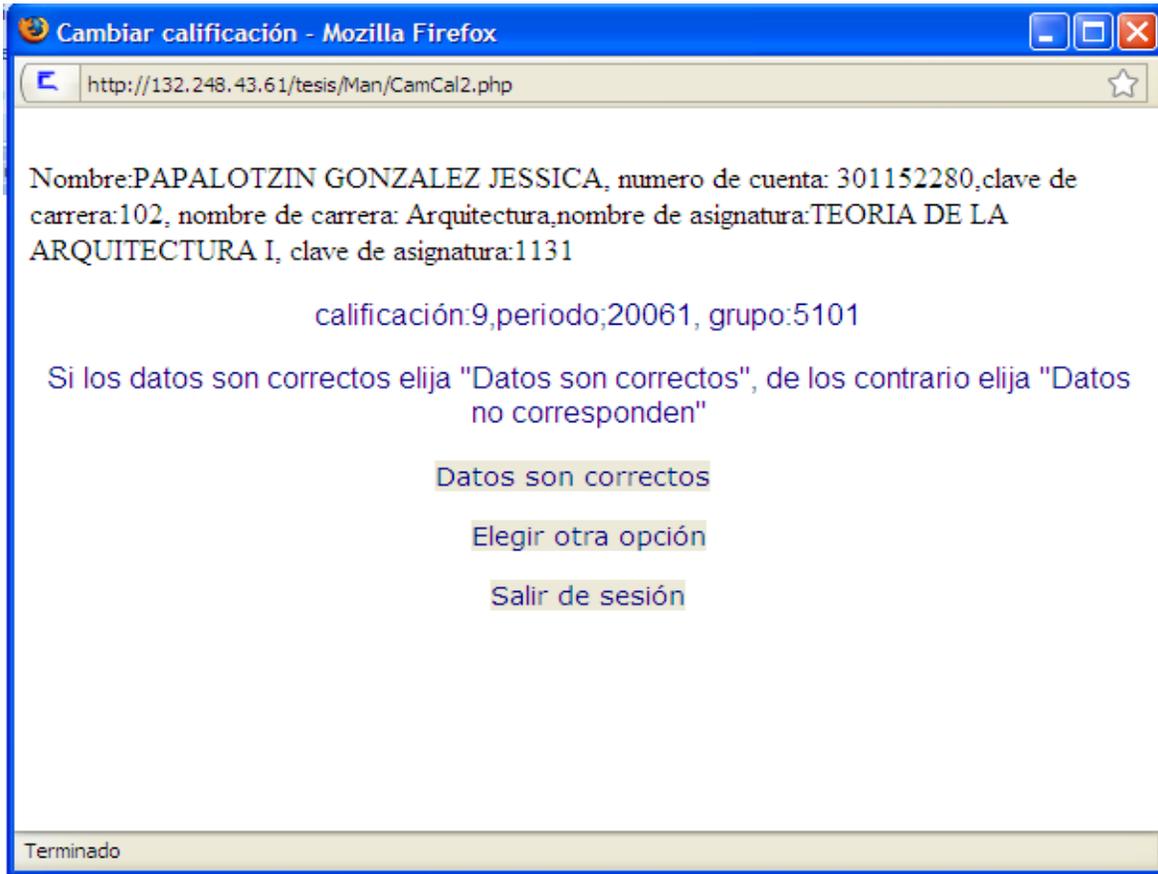
[Salir de sesión](#)

Terminado

Campos:

Campo	Representación
Número de cuenta	Text Field
Clave de asignatura	Text Field
Actualizar	Button
Volver al menú de opciones	Link
Salir de sesión	Link

Nombre: **CamCal2.php**  
 Página anterior: CamCal.php  
 Página siguiente: CamCal3.php  
 Descripción: Página que muestra los datos del Alumno, así como la(s) calificación(es) de su historial académico en la asignatura.  
 Pantalla:

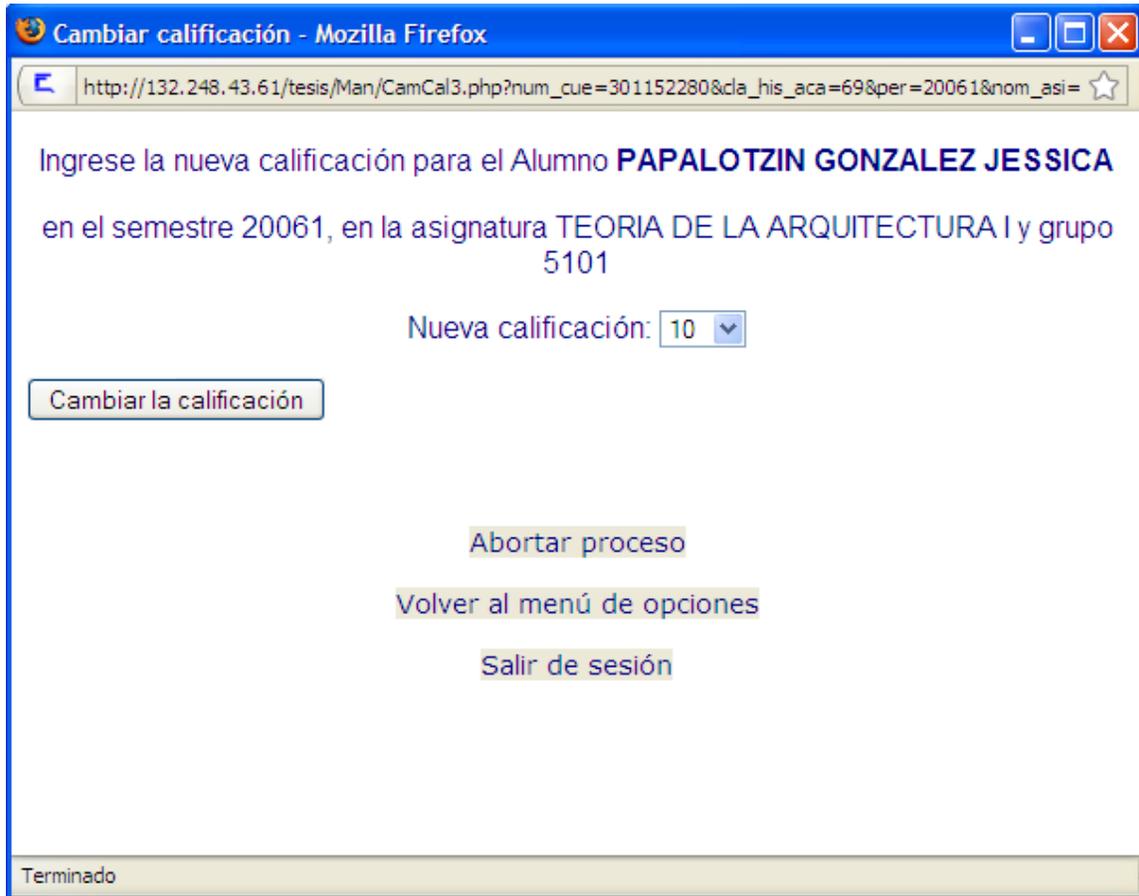


Campos:

Campo	Representación
Datos son correctos	Link
Elegir otra opción	Link
Salir de sesión	Link

Nombre: **CamCal3.php**  
 Página anterior: CamCal2.php  
 Página siguiente: CamCal4.php  
 Descripción: Página que permite el ingreso de la nueva calificación del Alumno en el último grupo de la asignatura cursada.

Pantalla:

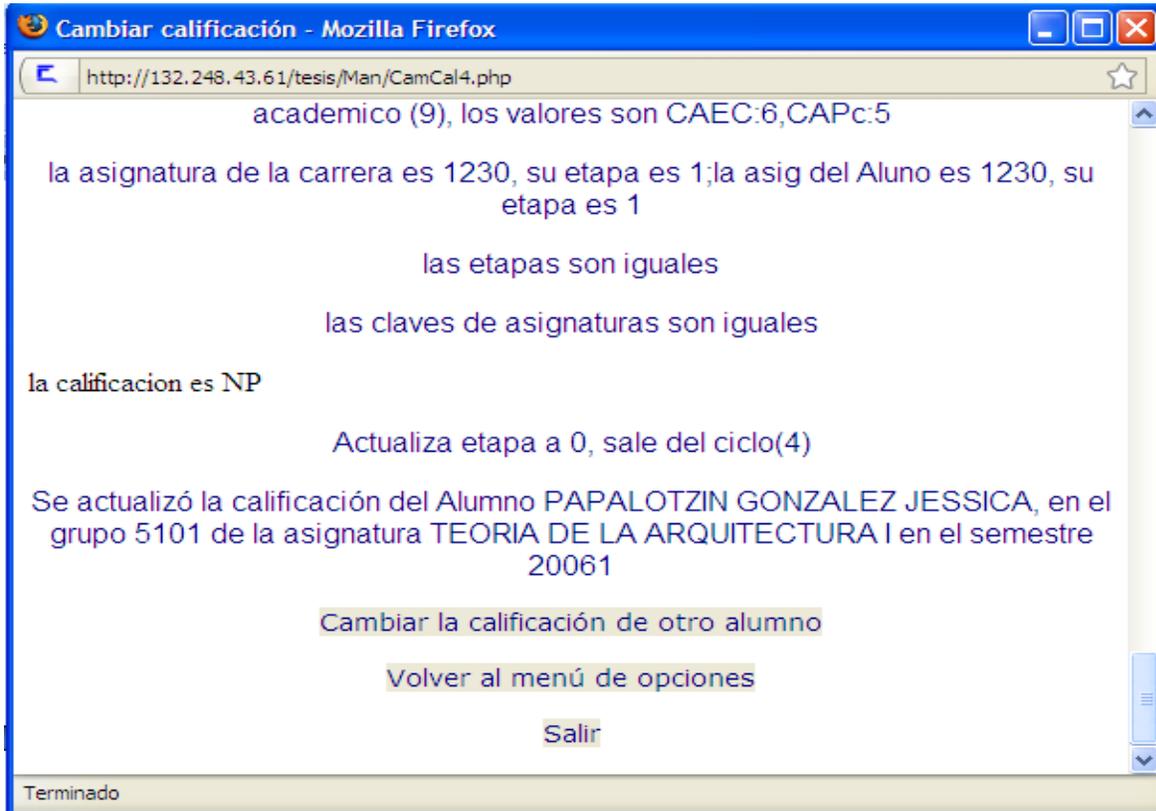


Campos:

Campo	Representación
Nueva calificación	Menu list
Cambiar calificación	Button
Abortar proceso	Link
Volver al menú de opciones	Link
Salir de sesión	Link

Nombre: **CamCal4.php**  
 Página anterior: CamCal3.php  
 Página siguiente: ninguna  
 Descripción: Página que muestra el éxito en el proceso de actualizar la calificación del Alumno en el último grupo cursado de la asignatura en su historia académica .

Pantalla:

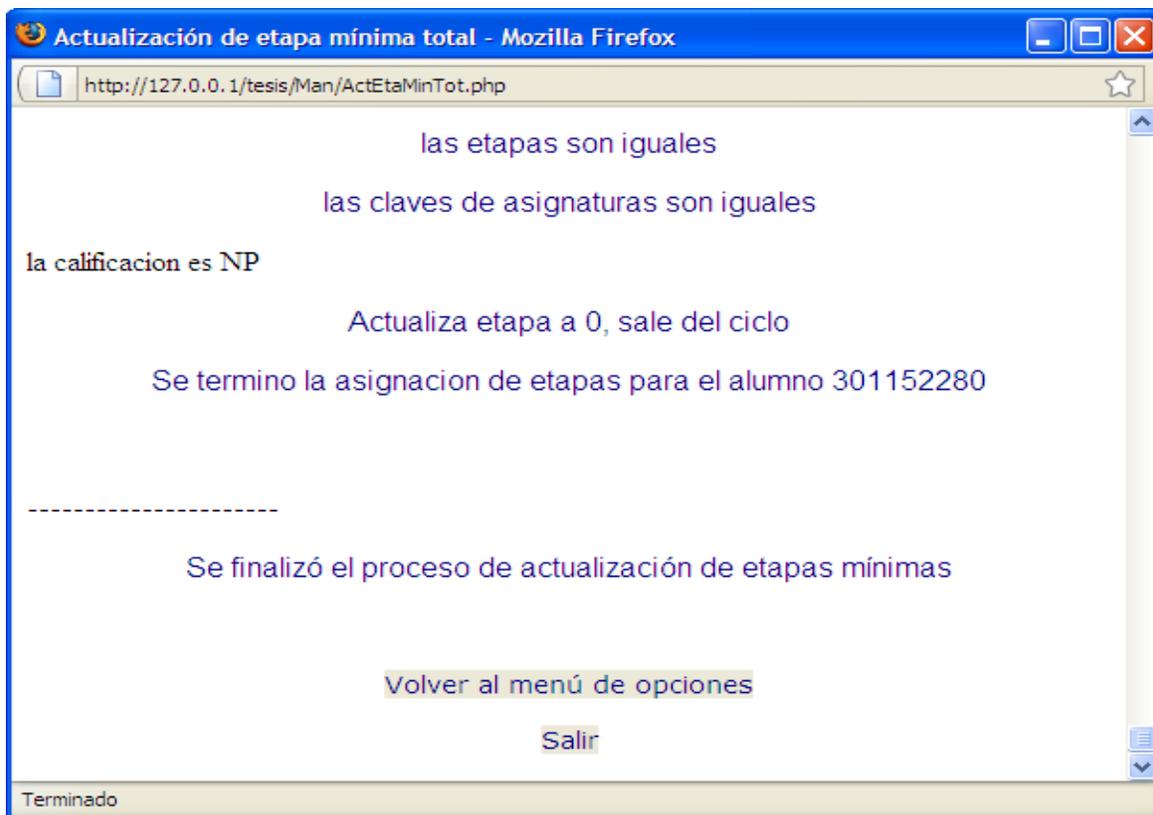


Campos:

Campo	Representación
Cambiar la calificación de otro Alumno	Link
Volver al menú de opciones	Link
Salir	Link

## Actualización de etapas mínimas

Nombre: **ActEtaMinTot.php**  
Página anterior: opciones.php  
Página siguiente: ninguna  
Descripción: Página que muestra el fin de la actualización de la etapa mínima (avance por bloques) para todos los Alumnos de Arquitectura.  
Pantalla:

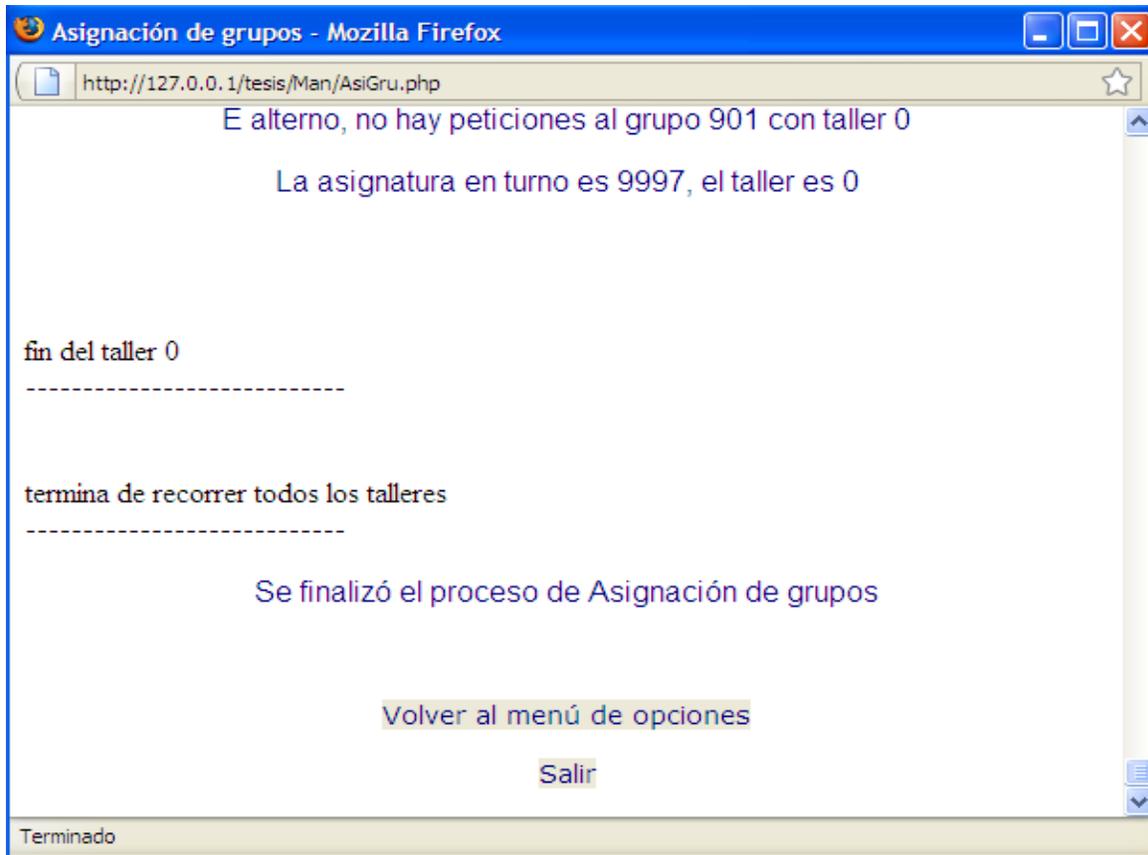


Campos:

Campo	Representación
Volver al menú de opciones	Link
Salir	Link

## Asignación de grupos

Nombre: AsiGru.php  
Página anterior: opciones.php  
Página siguiente: ninguna  
Descripción: Página que muestra el fin del proceso de asignación de grupos para todos los Alumnos de Arquitectura.  
Pantalla:

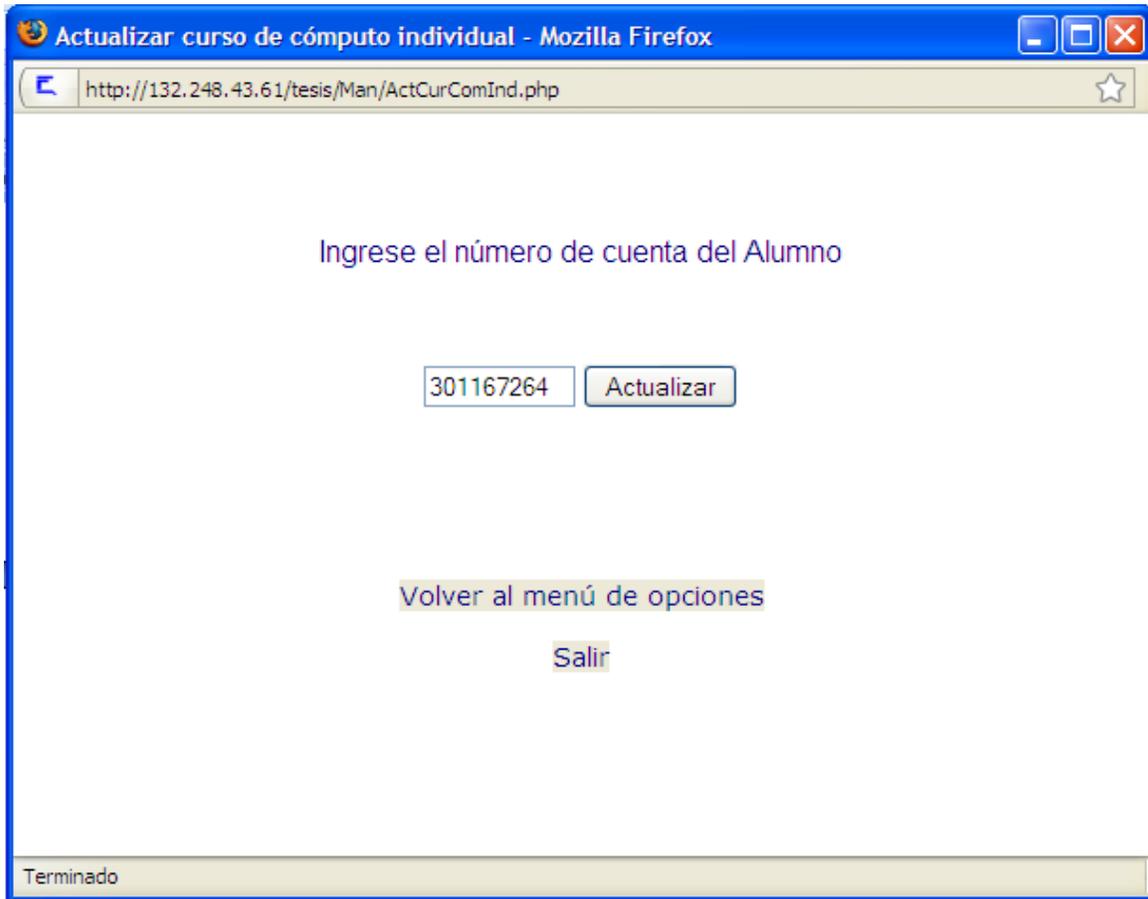


Campos:

Campo	Representación
Volver al menú de opciones	Link
Salir	Link

## Actualización de curso de cómputo individual

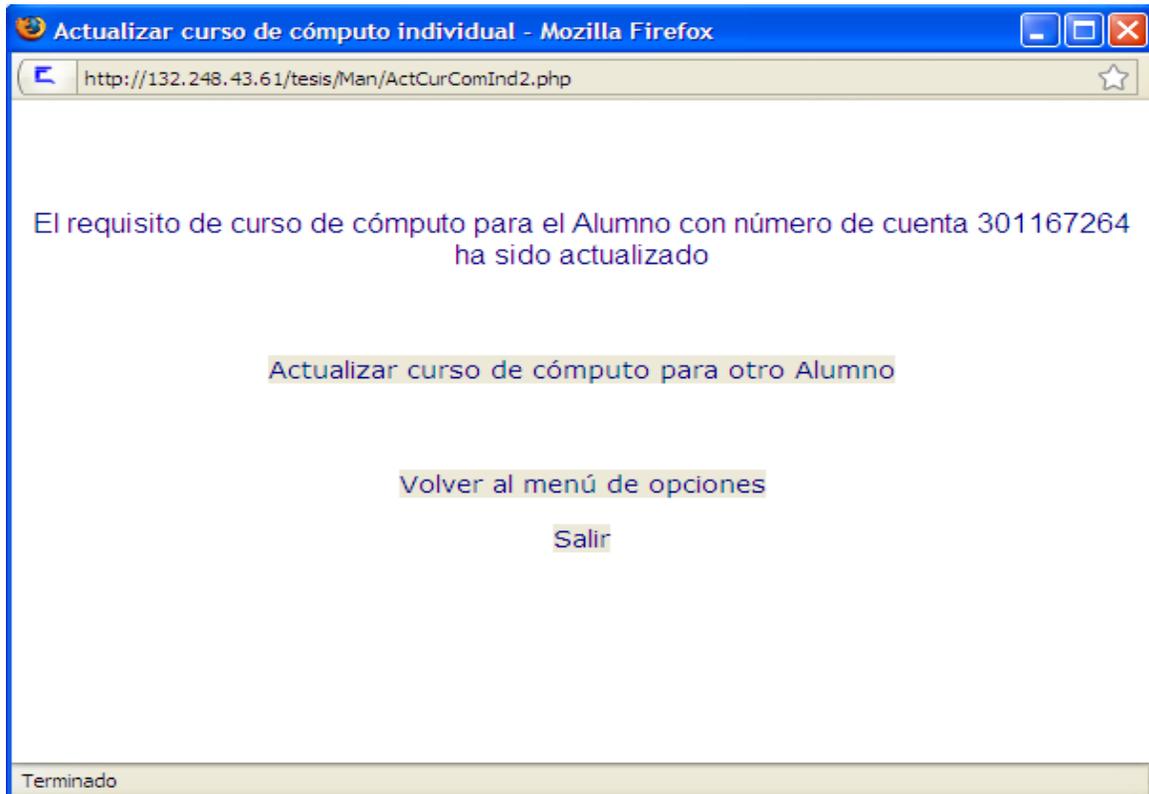
Nombre: **ActCurComInd.php**  
Página anterior: opciones.php  
Página siguiente: ActCurComInd2.php  
Descripción: Página que sirve para buscar a un Alumno de Arquitectura.  
Pantalla:



Campos:

Campo	Representación
Número de cuenta	Text Field
Actualizar	Button
Volver al menú de opciones	Link
Salir	Link

Nombre: **ActCurComInd2.php**  
 Página anterior: ActCurComInd.php  
 Página siguiente: ninguna  
 Descripción: Página que muestra que el requisito de curso de cómputo se ha actualizado para el alumno.  
 Pantalla:

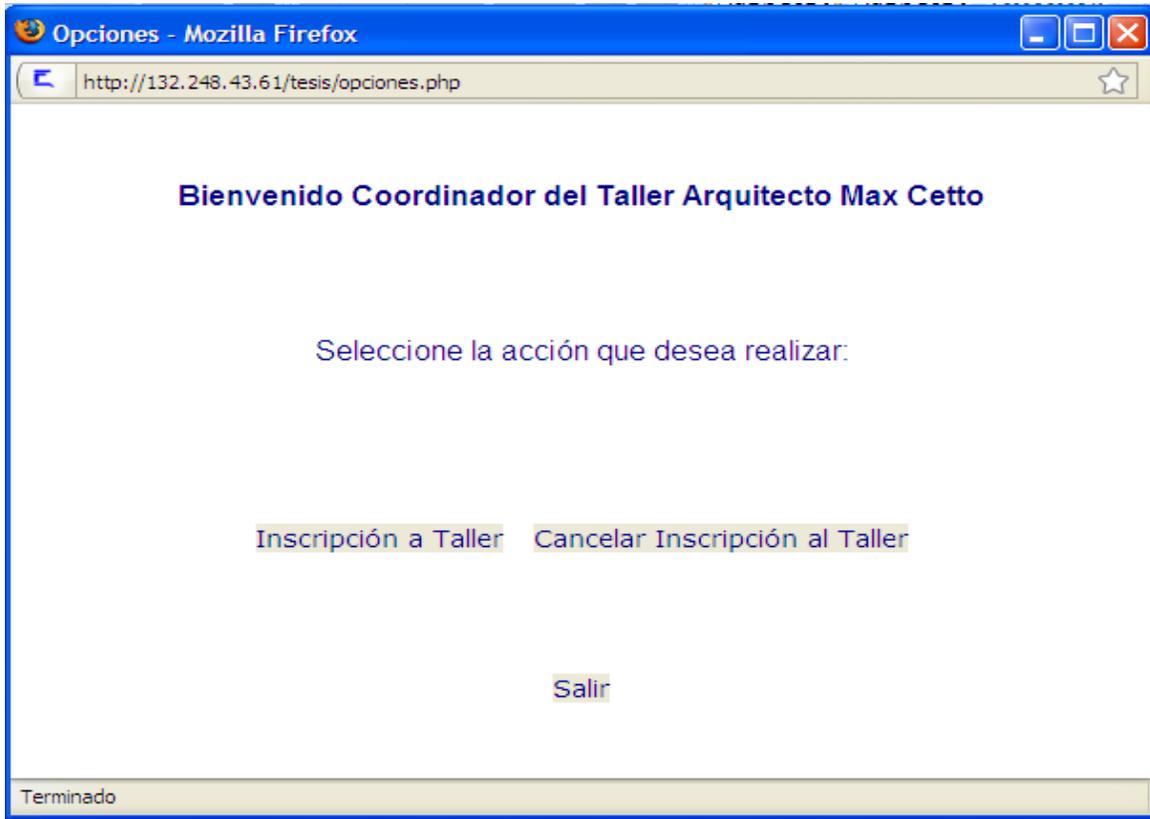


Campos:

Campo	Representación
Actualizar curso de cómputo para otro Alumno	Link
Volver al menú de opciones	Link
Salir	Link

## Coordinador de Taller

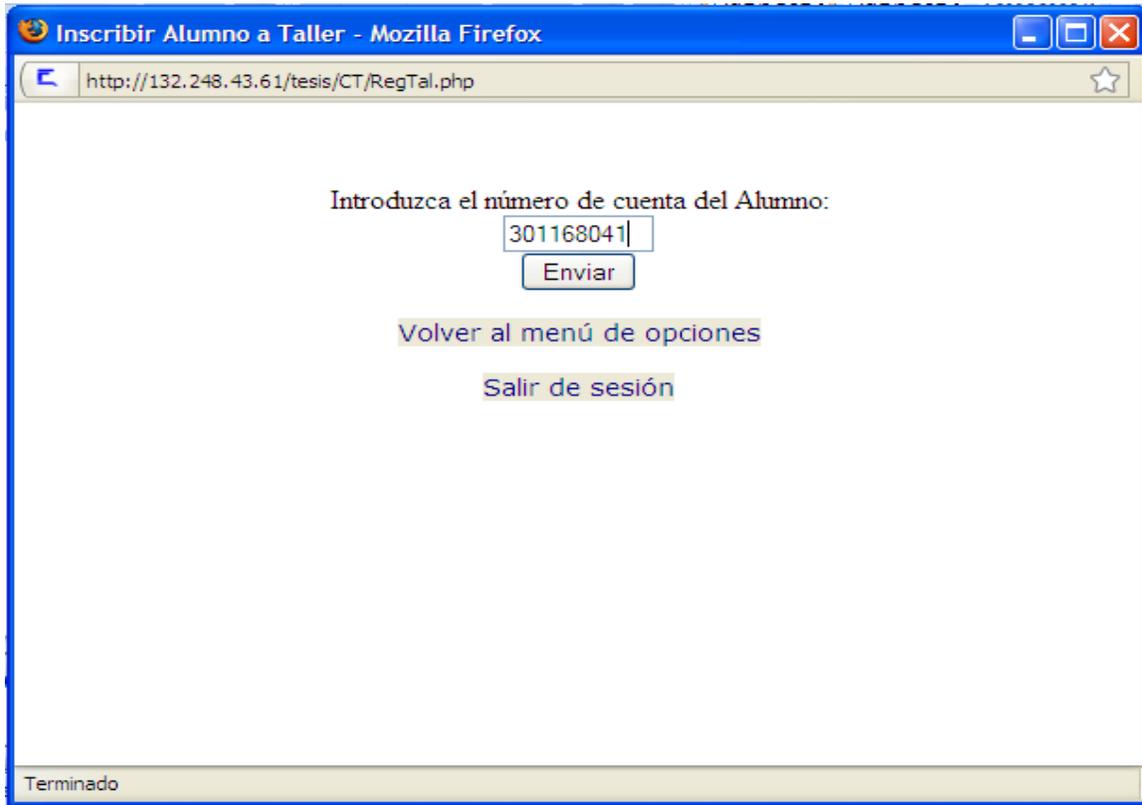
Nombre: opciones.php  
Página anterior: IngresoTramitesEscolaresGenera.html  
Página siguiente: CanTal.php  
Descripción: Página que sirve para que el actor Coordinador de Taller elija la acción que desea ejecutar.  
Pantalla:



Campos:

Campo	Representación
Inscripción a Taller	Link
Cancelar inscripción al Taller	Link
Salir	Link

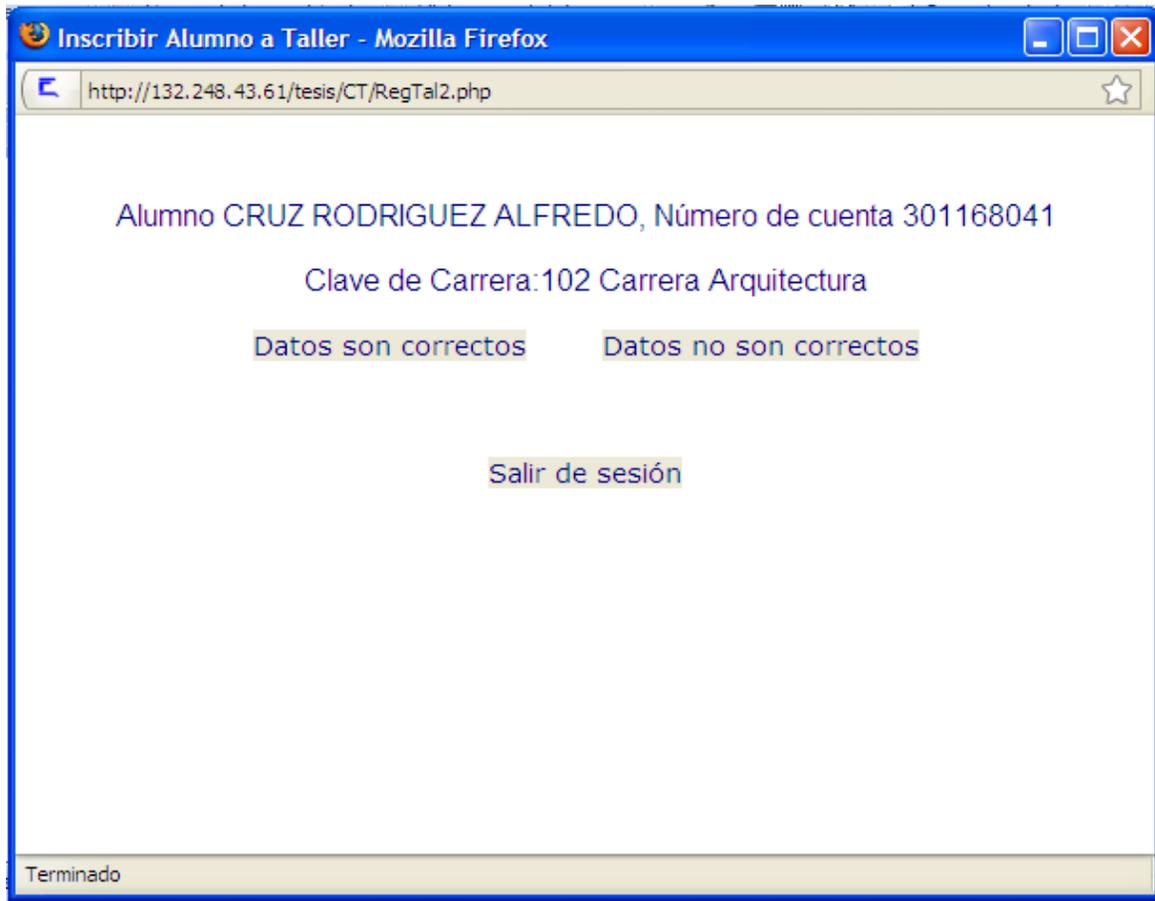
Nombre: **RegTal.php**  
 Página anterior: opciones.php  
 Página siguiente: RegTal2.php  
 Descripción: Página que sirve para que el actor Coordinador de Taller ingrese el número de cuenta del Alumno.  
 Pantalla:



Campos:

Campo	Representación
Número de cuenta	Text Field
Enviar	Button
Volver al menú de opciones	Link
Salir de sesión	Link

Nombre: **RegTal2.php**  
 Página anterior: RegTal.php  
 Página siguiente: RegTal3.php  
 Descripción: Página que muestra al Coordinador de Taller los datos del Alumno.  
 Pantalla:



Campos:

Campo	Representación
Datos son correctos	Link
Datos no son correctos	Link
Salir de sesión	Link

Nombre: **RegTal3.php**  
 Página anterior: RegTal2.php  
 Página siguiente: RegTal4.php  
 Descripción: Página que muestra al Coordinador de Taller las opciones que el Alumno tiene para poder inscribir Taller.  
 Pantalla:

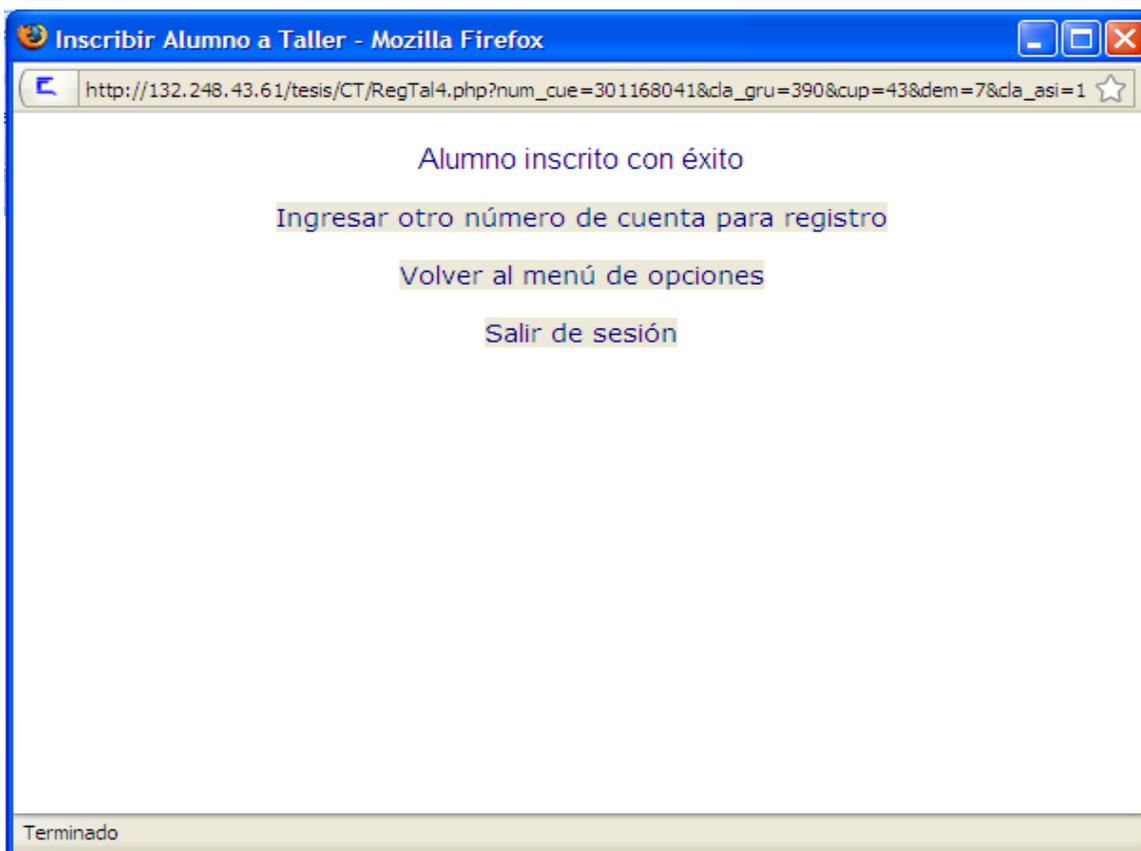


Campos:

Campo	Representación
Clave Grupo:885,Cupo:49,Demanda:1,Clave_int_grupo:5002 *	Link
Volver al menú de opciones	Link
Salir de sesión	Link

\*El número de opciones para inscribir, así como los datos del grupo pueden variar, ya que dependen del caso del Alumno y de cuántos Alumnos más se hayan inscrito en éste

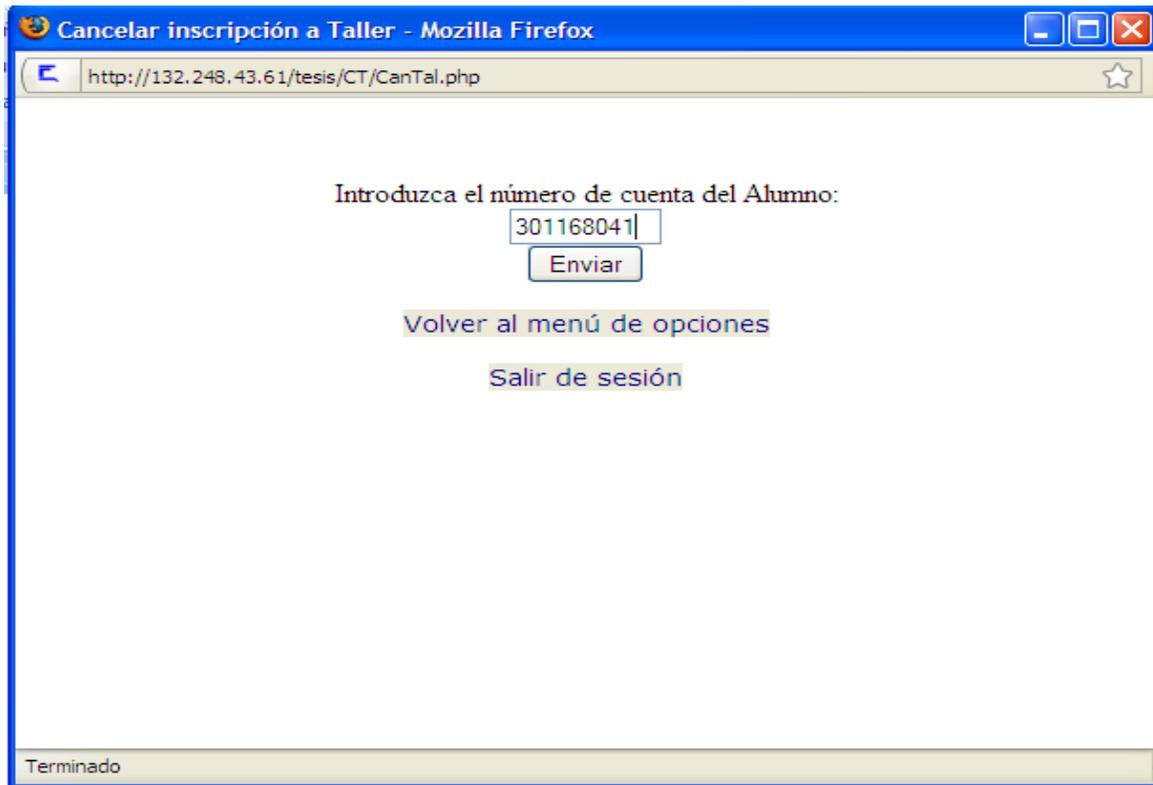
Nombre: **RegTal4.php**  
 Página anterior: RegTal3.php  
 Página siguiente: ninguna  
 Descripción: Página que muestra al Coordinador de Taller el proceso de Inscribir a un Alumno a su Taller fue finalizado con éxito.  
 Pantalla:



Campos:

Campo	Representación
Ingresar otro número de cuenta para registro	Link
Volver al menú de opciones	Link
Salir de sesión	Link

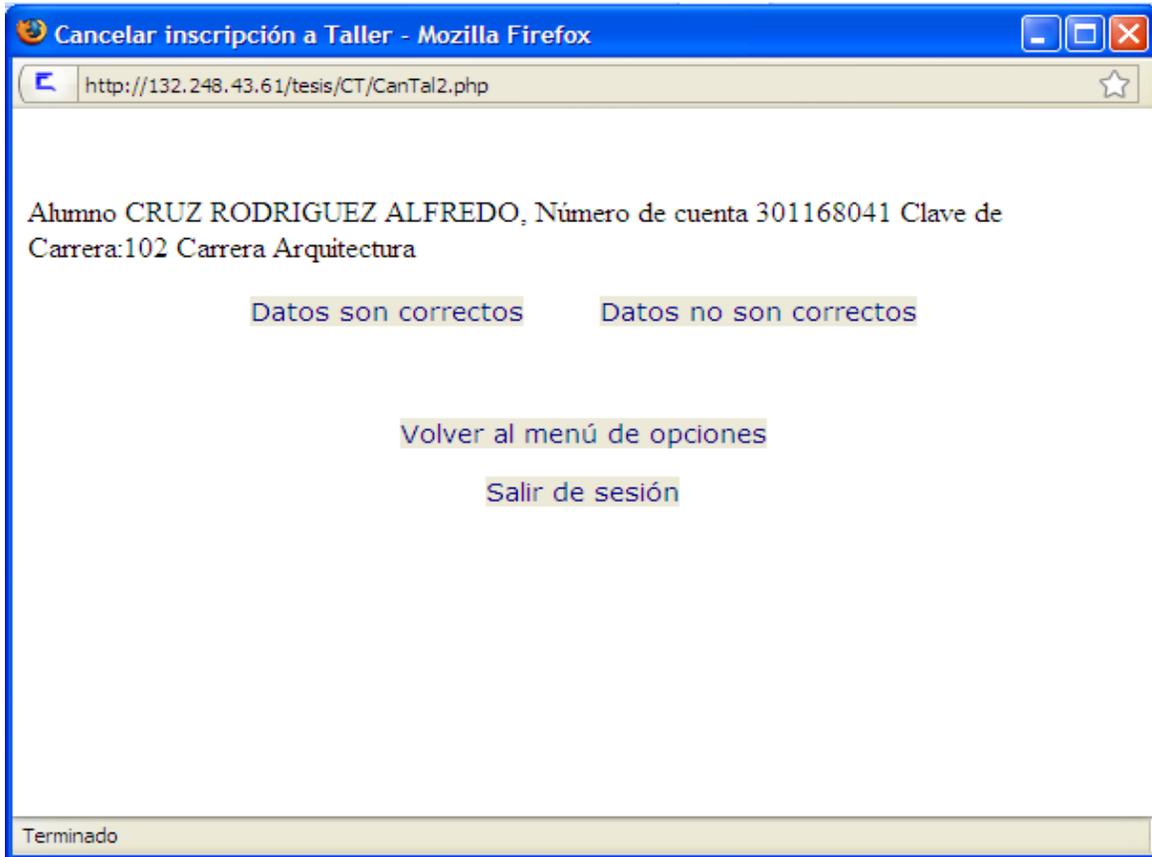
Nombre: **CanTal.php**  
 Página anterior: opciones.php  
 Página siguiente: CanTal2.php  
 Descripción: Página que sirve para que el actor Coordinador de Taller ingrese el número de cuenta del Alumno que cancelará su inscripción al Taller.  
 Pantalla:



Campos:

Campo	Representación
Número de cuenta	Text Field
Enviar	Button
Volver al menú de opciones	Link
Salir de sesión	Link

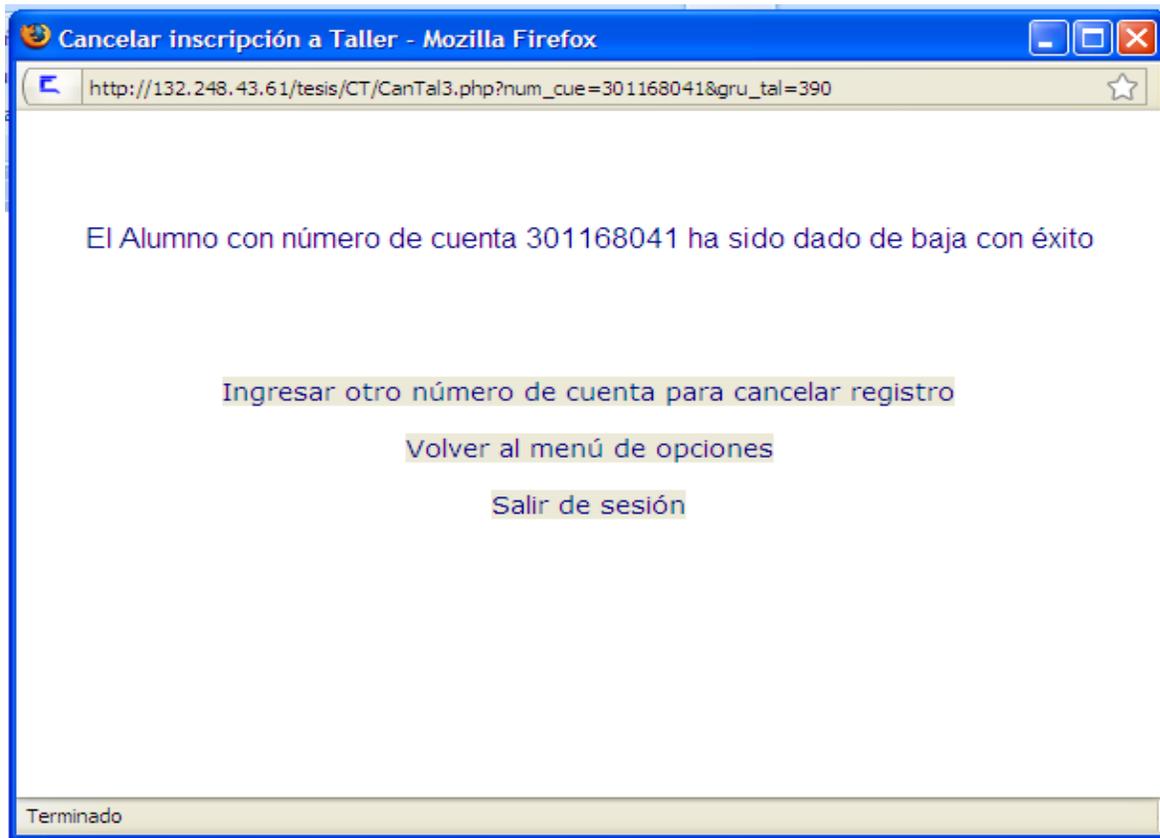
Nombre: **CanTal2.php**  
 Página anterior: CanTal.php  
 Página siguiente: CanTal3.php  
 Descripción: Página que muestra al actor Coordinador de Taller los datos del Alumno, cuando éste está inscrito en una asignatura de su Taller.  
 Pantalla:



Campos:

Campo	Representación
Datos son correctos	Link
Datos no son correctos	Link
Volver al menú de opciones	Link
Salir de sesión	Link

Nombre: **CanTal2.php**  
 Página anterior: CanTal.php  
 Página siguiente: CanTal3.php  
 Descripción: Página que muestra al actor Coordinador de Taller los datos del Alumno, cuando éste está inscrito en una asignatura de su Taller.  
 Pantalla:



Campos:

Campo	Representación
Ingresar otro número de cuenta para cancelar registro	Link
Volver al menú de opciones	Link
Salir de sesión	Link

### Coordinador de Carrera

Nombre: **opciones.php**

Página Anterior: IngresoTramitesEscolares.php

Página Siguiente: SemTal.php  
salir.php

Descripción: Página en donde se muestran los diferentes Talleres que componen la carrera de Arquitectura, al cual se va a actualizar el cupo de un grupo.

Pantalla:



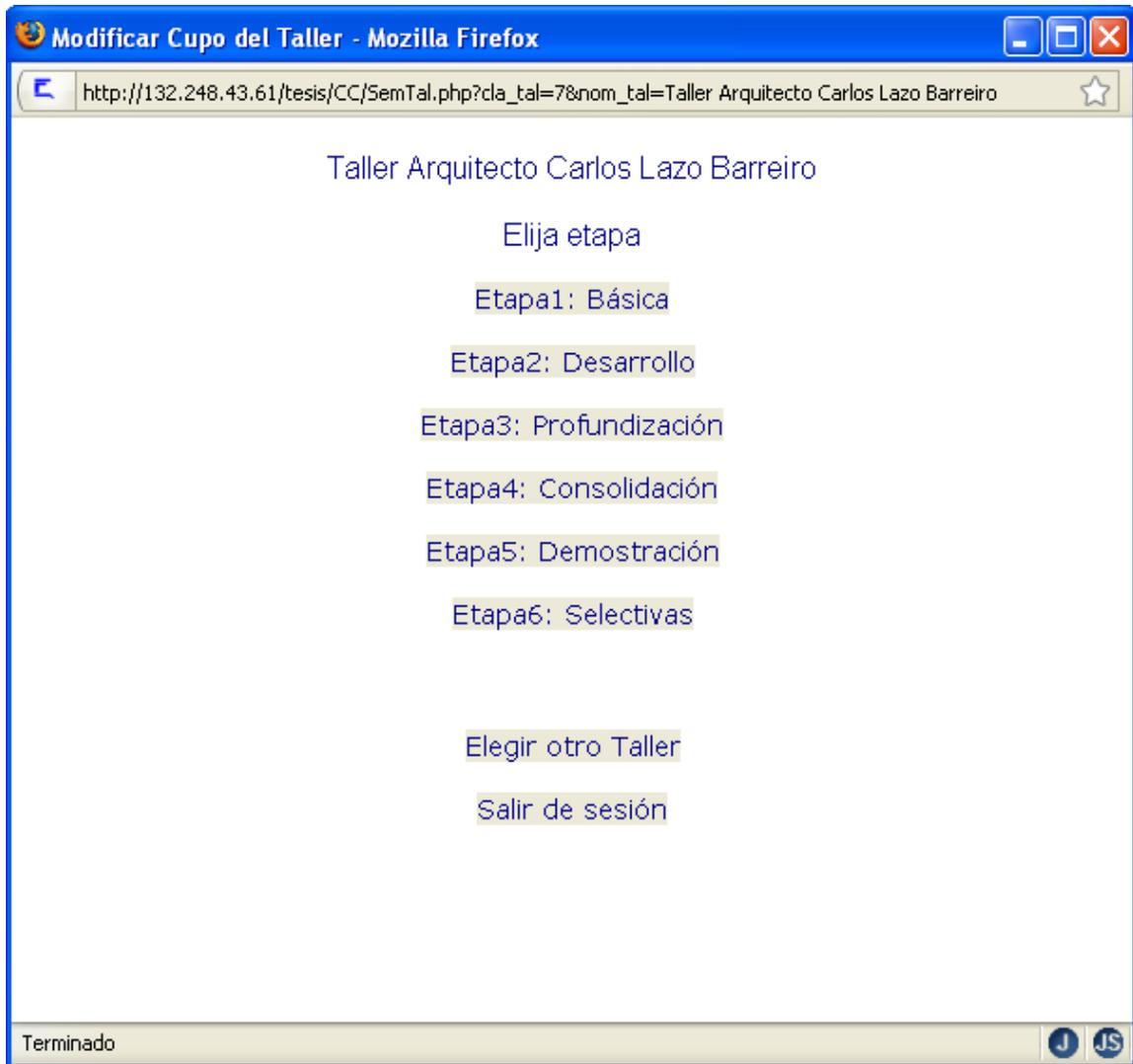
Campos:

Campo	Representación
Taller Arquitecto Luis Barragán	<a href="#">Link</a>
Taller Arquitecto Federico Mariscal y Piña	<a href="#">Link</a>
Taller Arquitecto Hannes Meyer	<a href="#">Link</a>
Taller Arquitecto Juan O' Gorman	<a href="#">Link</a>
Taller José Revueltas	<a href="#">Link</a>
Taller Tres	<a href="#">Link</a>
Taller Uno	<a href="#">Link</a>
Taller Arquitecto José Villagrán García	<a href="#">Link</a>
Taller Arquitecto Max Cetto	<a href="#">Link</a>
Taller Ehécatl XXI	<a href="#">Link</a>
Taller Arquitecto Juan García Gayoú	<a href="#">Link</a>
Taller Arquitecto Domingo García Ramos	<a href="#">Link</a>
Taller Arquitecto Jorge González Reyna	<a href="#">Link</a>
Taller Arquitecto Carlos Lazo Barreiro	<a href="#">Link</a>
Taller Arquitecto Carlos Leduc Montaña	<a href="#">Link</a>
Taller Arquitecto Ramón Marcos Noriega	<a href="#">Link</a>
Salir	<a href="#">Link</a>

Nombre: **SemTal.php**  
Página Anterior: opciones.php  
Página Siguiente: SemTal2.php  
opciones.php  
salir.php

Descripción: Página en donde se muestran las etapas correspondientes a la carrera de Arquitectura así como los cursos selectivos.

Pantalla:



Campos:

Campo	Representación
Etapa1: Básica	Link
Etapa2: Desarrollo	Link
Etapa3: Profundización	Link
Etapa4: Consolidación	Link
Etapa5: Demostración	Link
Etapa6: Selectivas	Link
Elegir otro Taller	Link
Salir de sesión	Link

Nombre: **SemTal2.php**  
 Página Anterior: SemTal1.php  
 Página Siguiente: SemTal3.php  
 opciones.php  
 salir.php  
 Descripción: Página en donde se despliegan los grupos del semestre actual del Taller seleccionado.  
 Pantalla:



Campos:

Campo	Representación
Grupo 5707	Link
Grupo 5607	Link
Elegir otro Taller	Link
Salir de sesión	Link

Nombre: **SemTal3.php**  
Página Anterior: SemTal2.php  
Página Siguiente: ActCupTal.php  
opciones.php  
salir.php

Descripción: Página en donde se muestra la información del grupo elegido: cupo y demanda del mismo, así como un campo para proporcionar el nuevo cupo del grupo.

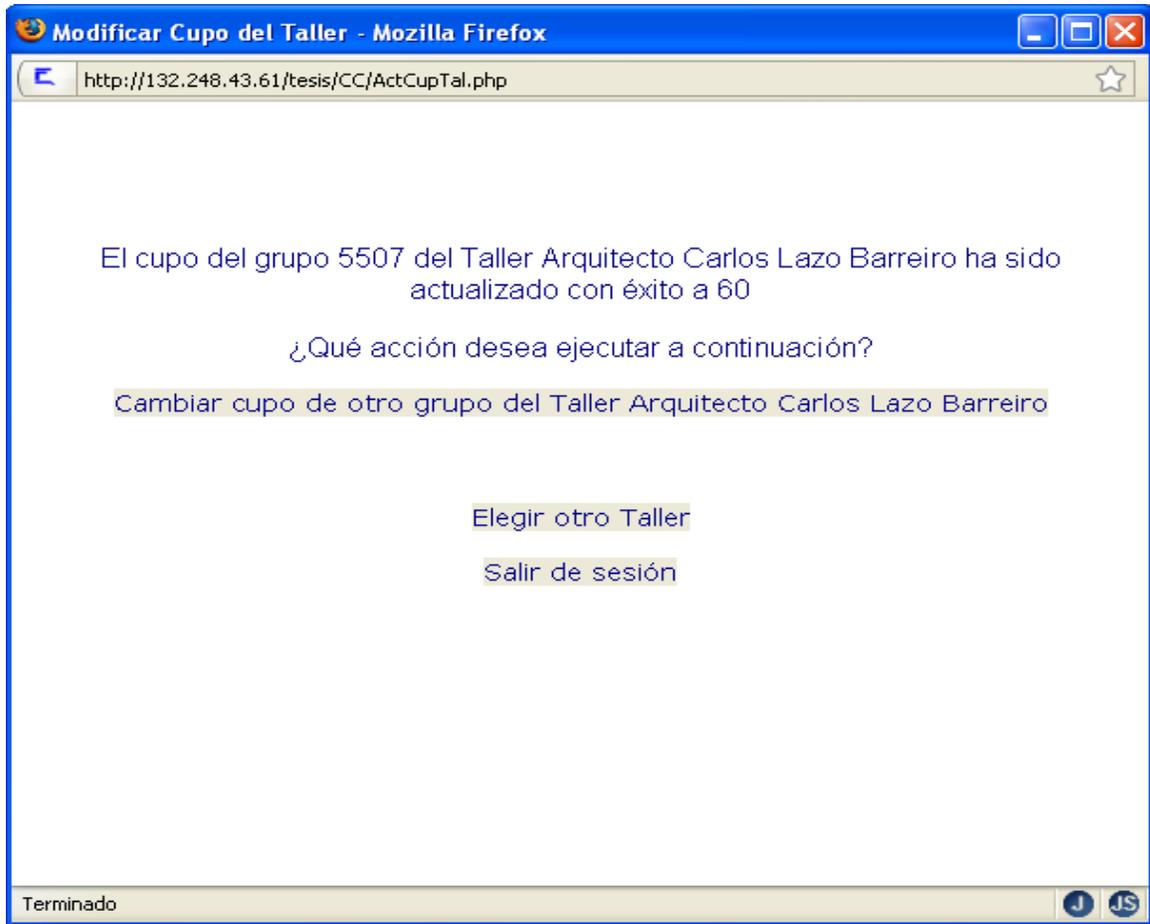
Pantalla:



Campos:

Campo	Representación
Cupo	Text Field
Actualizar Cupo	Button

Nombre: **ActCupTal.php**  
 Página Anterior: SemTal3.php  
 Página Siguiente: SemTal.php  
 opciones.php  
 salir.php  
 Descripción: Página en donde se muestra que el cupo del grupo seleccionado ha sido actualizado.  
 Pantalla:

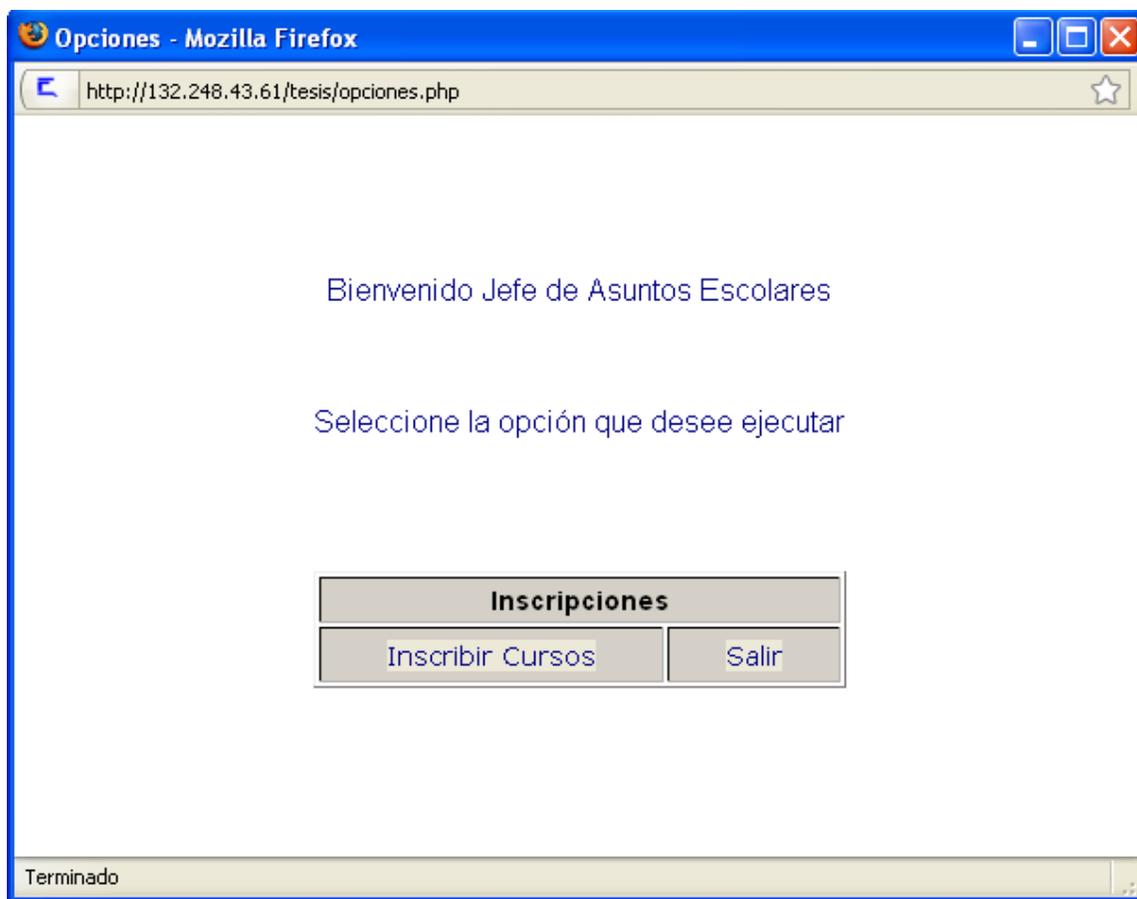


Campos:

Campo	Representación
Cambiar cupo de otro grupo del mismo Taller	Link
Elegir otro Taller	Link
Salir de sesión	Link

## Jefe de Asuntos Escolares

Nombre: **opciones.php**  
Página Anterior: IngresoTramitesEscolares.php  
Página Siguiente: InsCasExt.php  
salir.php  
Descripción: Página en donde se muestran las opciones para el Jefe de Asuntos Escolares.  
Pantalla:



Campos:

Campo	Representación
Inscribir Cursos	Link
Salir	Link

## Inscribir cursos

Nombre: InsCasExt.php  
Página Anterior: opciones.php  
Página Siguiente: InsCasExt2.php  
opciones.php  
salir.php

Descripción: Página en donde se ingresa el número de cuenta del Alumno.  
Pantalla:

Inscribir Alumno - Mozilla Firefox

http://132.248.43.61/tesis/JAE/InsCasExt.php

Proporcione los datos que solicitan

Número de cuenta:

Ingresar

Volver al menú de opciones

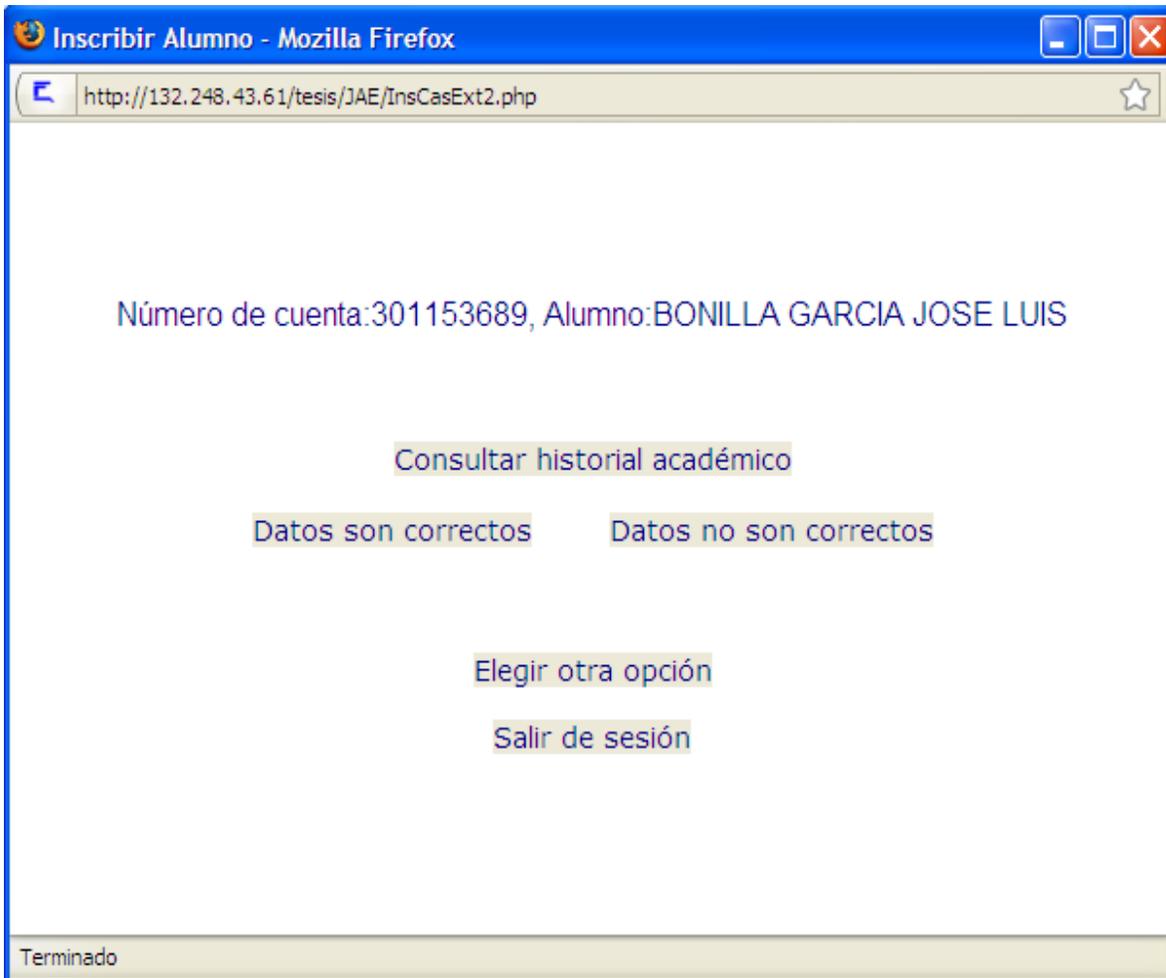
Salir de sesión

Terminado

Campos:

Campo	Representación
Número de cuenta	Text Field
Ingresar	Button
Volver al menú de opciones	Link
Salir de sesión	Link

Nombre: **InsCasExt2.php**  
 Página Anterior: InsCasExt.php  
 Página Siguiente: InsCasExt3.php  
 ConHisAcaAlu.php  
 opciones.php  
 salir.php  
 Descripción: Página en donde se muestran los datos del Alumno.  
 Pantalla:



Campos:

Campo	Representación
Consultar Historial Académico	Link
Datos son correctos	Link
Datos no son correctos	Link
Elegir otra opción	Link
Salir de sesión	Link

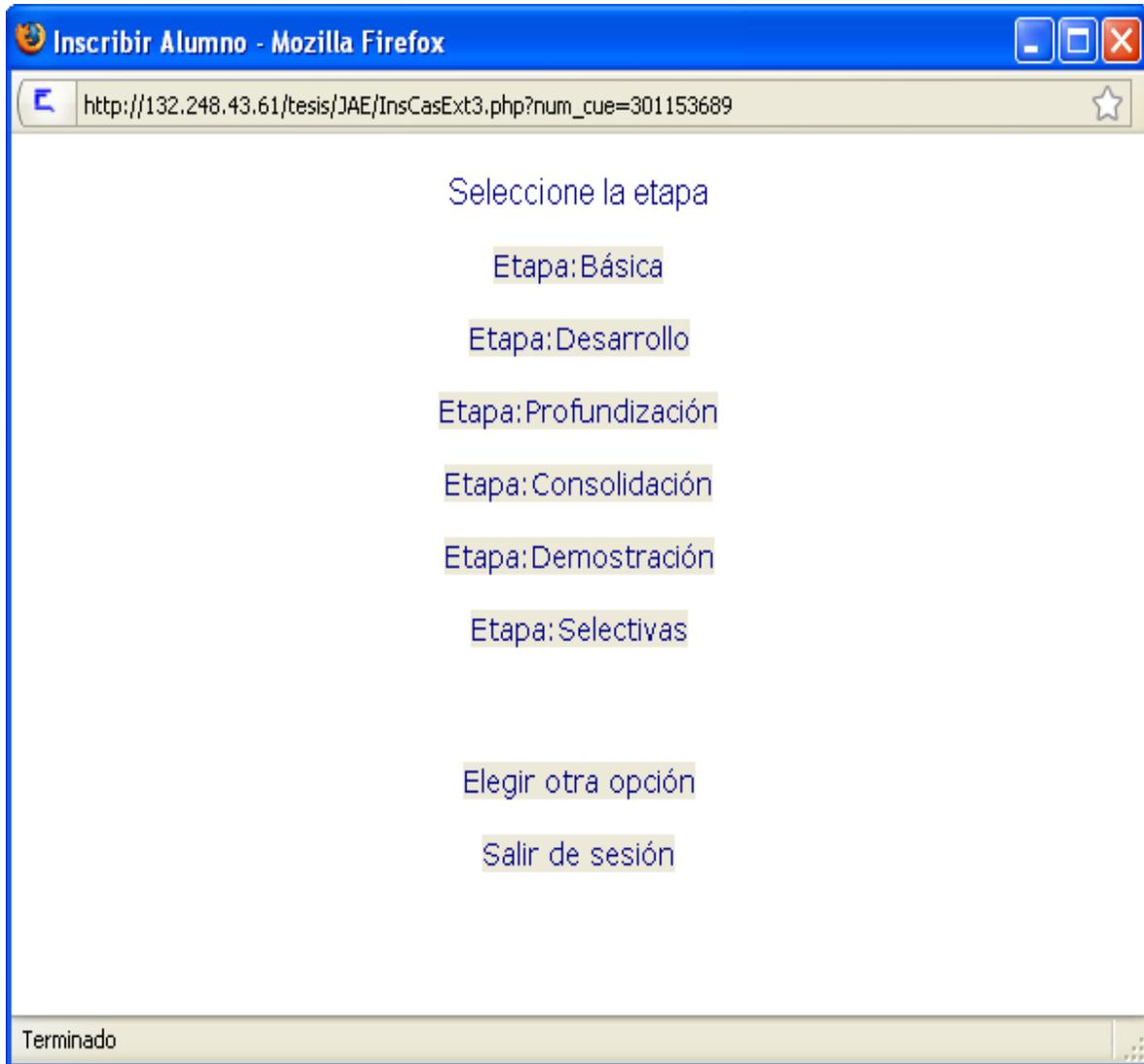
Nombre: **InsCasExt3.php**

Página Anterior: InsCasExt2.php

Página Siguiente: InsCasExt4.php  
opciones.php  
salir.php

Descripción: Página en donde se despliegan las etapas de la carrera de Arquitectura incluyendo los cursos selectivos internos.

Pantalla:



Campos:

Campo	Representación
Etapa: Básica	Link
Etapa: Desarrollo	Link
Etapa: Profundización	Link
Etapa: Consolidación	Link
Etapa: Demostración	Link
Etapa: Selectivas	Link
Elegir otra opción	Link
Salir de sesión	Link

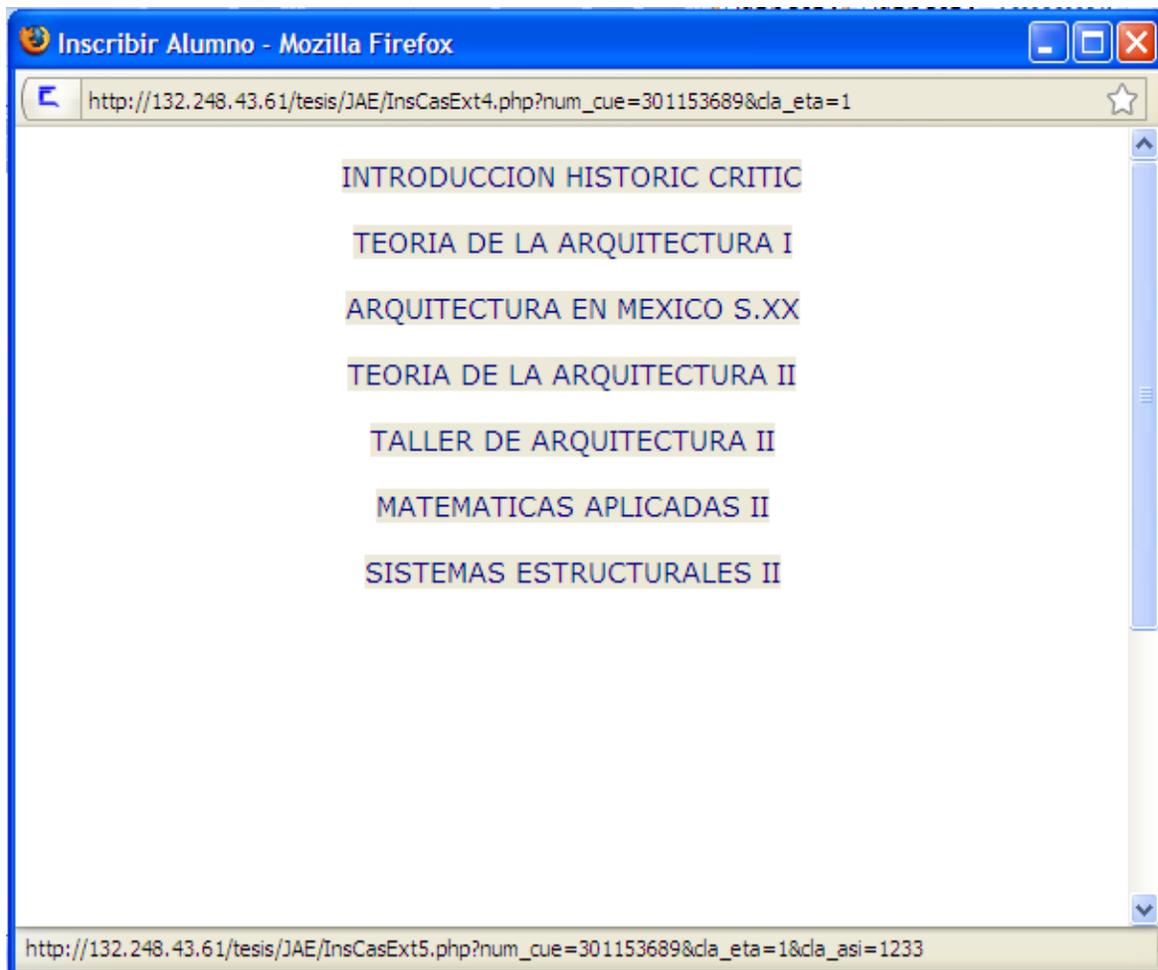
Nombre: **InsCasExt4.php**

Página Anterior: InsCasExt3.php

Página Siguiente: InsCasExt5.php  
opciones.php  
salir.php

Descripción: Página en donde se muestran las asignaturas correspondientes a la etapa seleccionada.

Pantalla:



Campos:

Campo	Representación
INTRODUCCION HISTORIC CRITIC	<a href="#">Link</a>
TEORIA DE LA ARQUITECTURA I	<a href="#">Link</a>
ARQUITECTURA EN MEXICO S. XX	<a href="#">Link</a>
TEORIA DE LA ARQUITECTURA II	<a href="#">Link</a>
TALLER DE ARQUITECTURA II	<a href="#">Link</a>
MATEMATICAS APLICADAS II	<a href="#">Link</a>
SISTEMAS ESTRUCTURALES II	<a href="#">Link</a>
Elegir otra opción	<a href="#">Link</a>
Salir de sesión	<a href="#">Link</a>

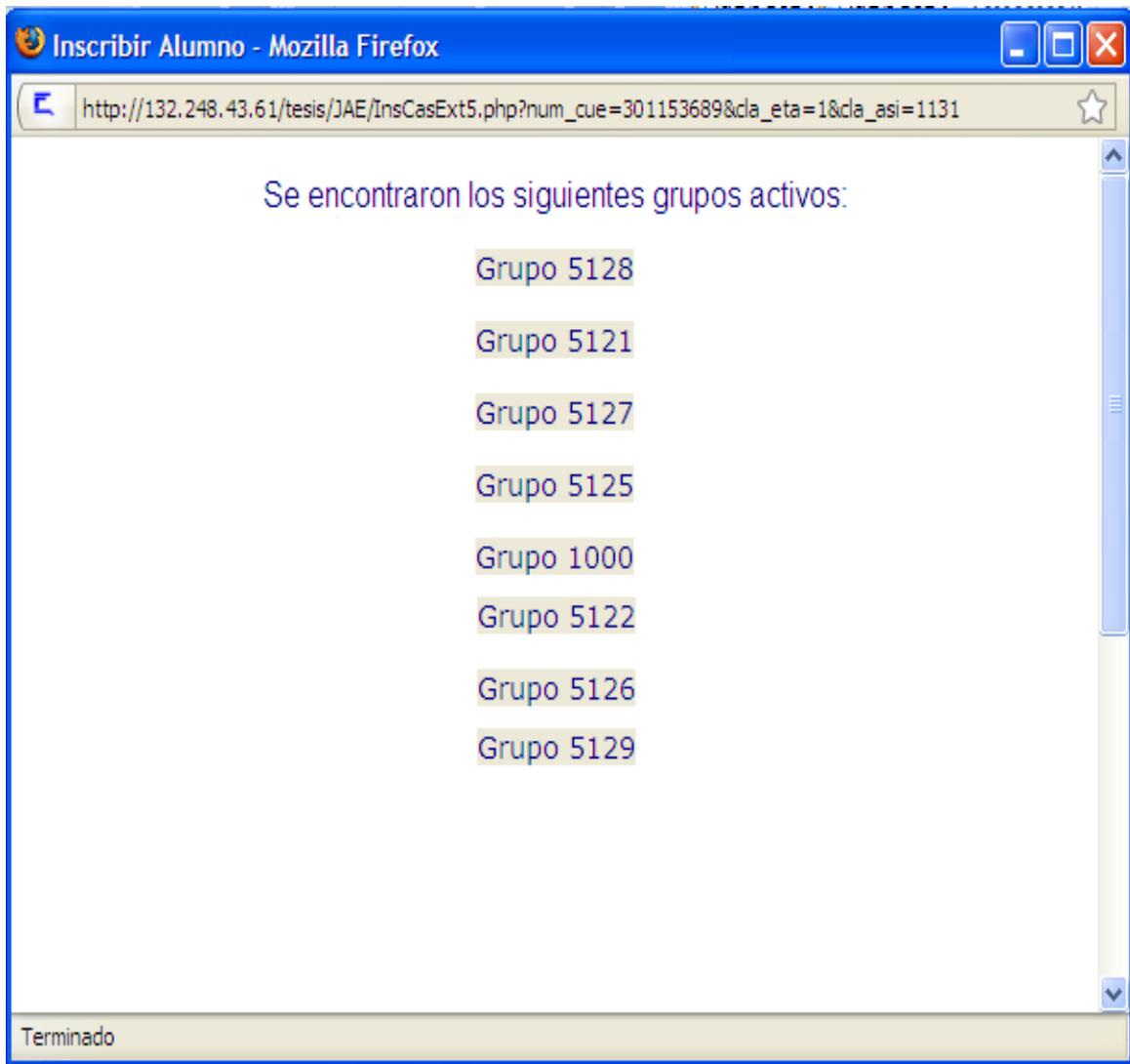
Nombre: **InsCasExt5.php**

Página Anterior: InsCasExt4.php

Página Siguiente: InsCasExt6.php  
opciones.php  
salir.php

Descripción: Página en donde se muestran los grupos correspondientes a la asignatura elegida.

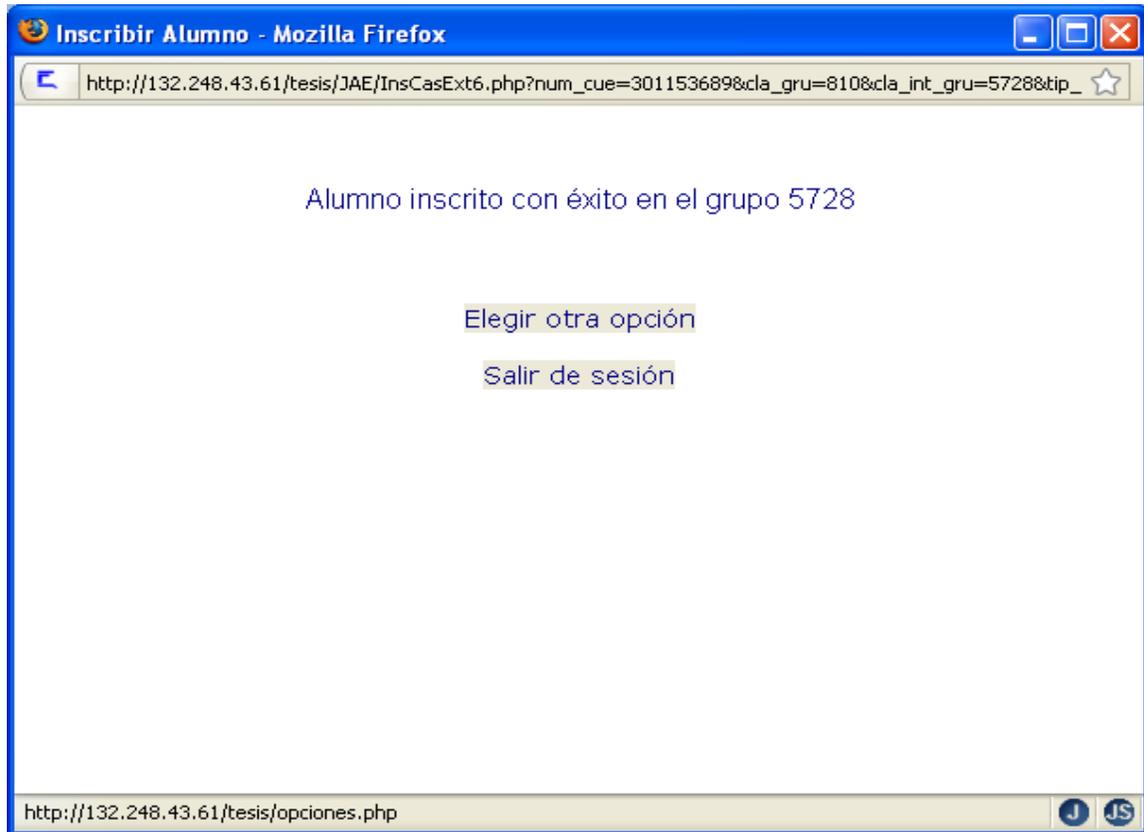
Pantalla:



Campos:

Campo	Representación
Grupo 5128	Link
Grupo 5121	Link
Grupo 5127	Link
Grupo 5125	Link
Grupo 1000	Link
Grupo 5122	Link
Grupo 5126	Link
Grupo 5129	Link
Elegir otra opción	Link
Salir de sesión	Link

Nombre: **InsCasExt6.php**  
Página Anterior: InsCasExt5.php  
Página Siguiente: opciones.php  
salir.php  
Descripción: Página en donde se muestra que el Alumno ha sido inscrito al grupo de la asignatura seleccionada.  
Pantalla:



Campos:

Campo	Representación
Elegir otra opción	Link
Salir de sesión	Link

## Consultar historial académico

Nombre: **ConHisAcaAlu.php**  
Página Anterior: **InsCasExt2.php**  
Página Siguiente: **Ninguna**  
Descripción: **Página en donde se despliegan la Historia Académica del Alumno: asignaturas con calificación, clave, periodo así como tipo de examen.**  
Pantalla:



Campos:

Campo	Representación
Cerrar ventana	Link

# Anexo E

## Diccionario de Datos

### Tabla actores\_asuntos\_escolares

Nombre: **CLA\_AAE**

Tipo de dato: character varying

Longitud: 4

Tipo de llave: primaria

Valor nulo: no aplica

Significado: La clave que identifica a cada uno de los Actores de Asuntos. Escolares:

Mantenimiento.

Jefe de Asuntos Escolares.

Coordinador de Carrera.

Coordinadores de Talleres.

Valores posibles:

MA01	Mantenimiento
CT01	Coordinador del Taller Arquitecto Luís Barragán
CT02	Coordinador del Taller Arquitecto Max Cetto
CT03	Coordinador del Taller Ehécatl XXI
CT04	Coordinador del Taller Arquitecto Juan García Gayoú
CT05	Coordinador del Taller Arquitecto Domingo García ramos
CT06	Coordinador del Taller Arquitecto Jorge González Reyna
CT07	Coordinador del Taller Arquitecto Carlos Lazo Barreiro
CT08	Coordinador del Taller Arquitecto Carlos Leduc Montaña
CT09	Coordinador del Taller Arquitecto Ramón Marcos Noriega
CT10	Coordinador del Taller Arquitecto Federico Mariscal y Piña
CT11	Coordinador del Taller Arquitecto Hannes Meyer
CT12	Coordinador del Taller Arquitecto Juan O’Gorman

CT13 Coordinador del Taller José Revueltas  
CT14 Coordinador del Taller Tres  
CT15 Coordinador del Taller Uno  
CT16 Coordinador del Taller Arquitecto José Villagrán García  
CC01 Coordinador de Carrera.  
JAE1 Jefe de Asuntos Escolares

Nombre: **LOG\_AAE**

Tipo de dato: character varying

Longitud: 15

Valor nulo: no aplica

Significado: Login, nombre del usuario que se utilizará para conectarse al servidor.

Valores posibles:

MA01 Mantenimiento  
CT01 Coordinador del Taller Arquitecto Luís Barragán  
CT02 Coordinador del Taller Arquitecto Max Cetto  
CT03 Coordinador del Taller Ehécatl XXI  
CT04 Coordinador del Taller Arquitecto Juan García Gayoú  
CT05 Coordinador del Taller Arquitecto Domingo García ramos  
CT06 Coordinador del Taller Arquitecto Jorge González Reyna  
CT07 Coordinador del Taller Arquitecto Carlos Lazo Barreiro  
CT08 Coordinador del Taller Arquitecto Carlos Leduc Montaña  
CT09 Coordinador del Taller Arquitecto Ramón Marcos Noriega  
CT10 Coordinador del Taller Arquitecto Federico Mariscal y Piña  
CT11 Coordinador del Taller Arquitecto Hannes Meyer  
CT12 Coordinador del Taller Arquitecto Juan O’Gorman  
CT13 Coordinador del Taller José Revueltas  
CT14 Coordinador del Taller Tres  
CT15 Coordinador del Taller Uno  
CT16 Coordinador del Taller Arquitecto José Villagrán García  
CC01 Coordinador de Carrera.

JAE1 Jefe de Asuntos Escolares

Nombre: **PAS\_AAE**

Tipo de dato: character varying

Longitud: 15

Valor nulo: no aplica

Significado: Password, contraseña del usuario.

Valores posibles:

Mazeus	Mantenimiento
Ctarqluibar	Coordinador del Taller Arquitecto Luís Barragán
Ctarqmaxcet	Coordinador del Taller Arquitecto Max Cetto
Ctehe21	Coordinador del Taller Ehécatl XXI
Ctarqjuaagargay	Coordinador del Taller Arquitecto Juan García Gayó
Ctarqdomgarram	Coordinador del Taller Arquitecto Domingo García ramos
Ctarqjorgonrey	Coordinador del Taller Arquitecto Jorge González Reyna
Ctarqcarlazbar	Coordinador del Taller Arquitecto Carlos Lazo Barreiro
ctarqcarledmon	Coordinador del Taller Arquitecto Carlos Leduc Montaña
Ctarqrammarnor	Coordinador del Taller Arquitecto Ramón Marcos Noriega
Ctarqfedmarpiñ	Coordinador del Taller Arquitecto Federico Mariscal y Piña
ctarqhanmey	Coordinador del Taller Arquitecto Hannes Meyer
Ctarqjuaogo	Coordinador del Taller Arquitecto Juan O’Gorman
Ctjosrev	Coordinador del Taller José Revueltas
Cttres	Coordinador del Taller Tres
Ctuno	Coordinador del Taller Uno
Ctarqjosvilgar	Coordinador del Taller Arquitecto José Villagrán García
Ccfacarq	Coordinador de Carrera.
Jaefacarq	Jefe de Asuntos Escolares

Nombre: **NOM\_AAE**

Tipo de dato: character varying

Longitud: 45

Valor nulo: no

Significado: Nombre de cada uno de los Actores de Asuntos Escolares.

Valores posibles: Los valores ingresados al momento de la captura.

Nombre: **CLA\_CAR**

Tipo de dato: character varying

Longitud: 4

Tipo de llave: foránea

Valor nulo: si

Significado: Clave de la carrera de cada Coordinador de Carrera y Coordinador de Taller.

Valores posibles: 102 para los involucrados en la carrera de Arquitectura.

Nombre: **CLA\_TAL**

Tipo de dato: character varying

Longitud: 2

Tipo de llave: Foránea

Valor nulo: si

Significado: Clave que identifica a cada Taller de la carrera de Arquitectura.

Valores posibles:

0	Sin Taller
1	Taller Arquitecto Luís Barragán
2	Taller Arquitecto Max Cetto
3	Taller Ehécatl XXI
4	Taller Arquitecto Juan García Gayoú
5	Taller Arquitecto Domingo García ramos
6	Taller Arquitecto Jorge González Reyna
7	Taller Arquitecto Carlos Lazo Barreiro
8	Taller Arquitecto Carlos Leduc Montaña
9	Taller Arquitecto Ramón Marcos Noriega
10	Taller Arquitecto Federico Mariscal y Piña
11	Taller Arquitecto Hannes Meyer
12	Taller Arquitecto Juan O’Gorman

- 13 Taller José Revueltas
- 14 Taller Tres
- 15 Taller Uno
- 16 Taller Arquitecto José Villagrán García

Nombre: **CLA\_TIP**

Tipo de dato: character

Longitud: 1

Tipo de llave: foránea

Valor nulo: no aplica

Significado: Tipo de Actor de Asuntos Escolares.

Valores posibles:

- 1 Mantenimiento
- 2 Coordinador de Taller
- 3 Coordinador de Carrera
- 4 Jefe de Asuntos Escolares

**Script para generar la tabla actores\_asuntos\_escolares:**

```
CREATE TABLE actores_asuntos_escolares (
    cla_aae character varying(4) NOT NULL,
    log_aae character varying(15) NOT NULL,
    pas_aae character varying(15) NOT NULL,
    nom_aae character varying(45) NOT NULL,
    cla_car character varying(4),
    cla_tal character varying(2),
    cla_tip character(1) NOT NULL,
    PRIMARY KEY (cla_aae)
);
```

**Comando SQL para insertar datos en la tabla actores\_asuntos\_escolares:**

```
INSERT INTO actores_asuntos_escolares VALUES ('CC01', 'CC01', 'ccfacarq',
'SN', '102', '0', '3');
INSERT INTO actores_asuntos_escolares VALUES ('CT01', 'CT01',
'ctarquibar', 'fulano1', '102', '1', '2');
```

## Tabla alumnos

Nombre: **NUM\_CUE**

Tipo de dato: integer

Tipo de llave: primaria

Valor nulo: no

Significado: el número de cuenta de cada Alumno.

Valores posibles: números de cuenta.

Nombre: **CLA\_TAL**

Tipo de dato: character varying

Longitud: 2

Tipo de llave: foránea

Valor nulo: no

Significado: Clave del Taller de la carrera de Arquitectura al que pertenece el Alumno.

Valores posibles:

0	Sin taller
1	Taller Arquitecto Luís Barragán
2	Taller Arquitecto Max Cetto
3	Taller Ehécatl XXI
4	Taller Arquitecto Juan García Gayoú
5	Taller Arquitecto Domingo García ramos
6	Taller Arquitecto Jorge González Reyna
7	Taller Arquitecto Carlos Lazo Barreiro
8	Taller Arquitecto Carlos Leduc Montaña
9	Taller Arquitecto Ramón Marcos Noriega
10	Taller Arquitecto Federico Mariscal y Piña
11	Taller Arquitecto Hannes Meyer
12	Taller Arquitecto Juan O’Gorman
13	Taller José Revueltas

- 14 Taller Tres
- 15 Taller Uno
- 16 Taller Arquitecto José Villagrán García

Nombre: **CLA\_SIT**

Tipo de dato: character varying

Longitud: 3

Tipo de llave: foránea

Valor nulo: sí

Significado: la situación académica en que se encuentra el Alumno.

Valores posibles:

- Act Activo
- SuT Suspensión Temporal
- A22 Artículo 22

Nombre: **CLA\_TIP**

Tipo de dato: character

Longitud: 1

Tipo de llave: foránea

Valor nulo: sí

Significado: el tipo de Alumno para fines de registro e inscripción.

Valores posibles:

- 1 Taller
- 2 Regular
- 3 Irregular
- 4 Extraordinario

Nombre: **PAS\_ALU**

Tipo de dato: character varying

Longitud: 15

Valor nulo: no

Significado: la contraseña para autentificar al Alumno en el sistema.

Valores posibles: combinación de caracteres.

Nombre: **NOM\_ALU**

Tipo de dato: character varying

Longitud: 45

Tipo de llave: no aplica

Valor nulo: no

Significado: el nombre propio de cada Alumno.

Valores posibles: nombres y apellidos de los Alumnos.

Nombre: **ETA\_MIN**

Tipo de dato: integer

Valor nulo: no

Significado: la etapa de avance cubierto íntegramente en la carrera por el Alumno.

Valores posibles:

- 0 Alumno que aún no concluye totalmente la etapa Básica.
- 1 Alumno que ha concluido la etapa Básica.
- 2 Alumno que ha concluido la etapa Desarrollo.
- 3 Alumno que ha concluido la etapa Profundización.
- 4 Alumno que ha concluido la etapa Consolidación.
- 5 Alumno que ha concluido la etapa Demostración.

Nombre: **CALLE**

Tipo de dato: character varying

Longitud: 60

Valor nulo: sí

Significado: la calle donde reside el Alumno.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **NUM\_EXT**

Tipo de dato: character varying

Longitud: 4

Valor nulo: no

Significado: el número exterior del domicilio del Alumno.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **NUM\_INT**

Tipo de dato: character varying

Longitud: 4

Valor nulo: sí

Significado: el número interior del domicilio del Alumno.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **COL**

Tipo de dato: character varying

Longitud: 50

Valor nulo: sí

Significado: la colonia donde se ubica el domicilio.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **DEL**

Tipo de dato: character varying

Longitud: 35

Valor nulo: sí

Significado: la delegación donde se ubica el domicilio.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **COD\_POS**

Tipo de dato: character varying

Longitud: 5

Valor nulo: sí

Significado: el código postal del domicilio.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **EST**

Tipo de dato: character varying

Longitud: 21

Valor nulo: sí

Significado: la entidad federativa donde reside el Alumno.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **TEL\_ALU**

Tipo de dato: character varying

Longitud: 8

Valor nulo: sí

Significado: teléfono particular del Alumno.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **TEL\_CEL**

Tipo de dato: character varying

Longitud: 10

Valor nulo: sí

Significado: teléfono celular del Alumno.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **COR\_ELE**

Tipo de dato: character varying

Longitud: 40

Valor nulo: sí

Significado: correo electrónico del Alumno.

Valores posibles: valores recibidos durante la captura de datos.

Nombre: **FEC\_REG**

Tipo de dato: date

Valor nulo: sí

Significado: la fecha a partir de la cual el Alumno puede comenzar su registro de asignaturas para reinscripción.

Valores posibles: valores de fecha en el formato YY-MM-DD

Nombre: **CLA\_ABC**

Tipo de dato: integer

Tipo de llave: foránea

Valor nulo: no

Significado: la hora a partir de la cual el Alumno puede comenzar su periodo para altas, bajas y/o cambios de asignaturas.

Valores posibles: valores de horas en el formato HH:MM:SS

Nombre: **CUR\_COM**

Tipo de dato: character

Longitud: 1

Valor nulo: sí.

Significado: indica si el Alumno ha cumplido o no el requisito de Curso de Cómputo.

Valores posibles:

- 0 No necesita curso de cómputo.
- 1 Necesita curso de cómputo.

Nombre: **CLA\_TAL\_SOL**

Tipo de dato: character varying

Longitud: 2

Valor nulo: sí.

Significado: clave del Taller al que se ha inscrito el Alumno.

Valores posibles:

- 0 Sin Taller
- 1 Taller Arquitecto Luís Barragán

- 2 Taller Arquitecto Max Cetto
- 3 Taller Ehécatl XXI
- 4 Taller Arquitecto Juan García Gayoú
- 5 Taller Arquitecto Domingo García ramos
- 6 Taller Arquitecto Jorge González Reyna
- 7 Taller Arquitecto Carlos Lazo Barreiro
- 8 Taller Arquitecto Carlos Leduc Montaña
- 9 Taller Arquitecto Ramón Marcos Noriega
- 10 Taller Arquitecto Federico Mariscal y Piña
- 11 Taller Arquitecto Hannes Meyer
- 12 Taller Arquitecto Juan O’Gorman
- 13 Taller José Revueltas
- 14 Taller Tres
- 15 Taller Uno
- 16 Taller Arquitecto José Villagrán García

Nombre: **ACT\_DAT**

Tipo de dato: character

Longitud: 1

Valor nulo: sí

Significado: indica si los datos del Alumno ya han sido actualizados.

Valores posibles:

0 Datos no actualizados

1 Datos actualizados

Nombre: **PRO**

Tipo de dato: character varying

Longitud: 5

Valor nulo: no

Significado: promedio de las asignaturas del Alumno.

Valores posibles: valores numéricos con una mantisa de 2 dígitos en su parte fraccionaria.

### Script para generar la tabla alumnos:

```
CREATE TABLE alumnos (  
  
    num_cue integer NOT NULL,  
    cla_tal character varying(2),  
    cla_sit character varying(3),  
    cla_tip character(1),  
    pas_alu character varying(15) NOT NULL,  
    nom_alu character varying(45) NOT NULL,  
    eta_min integer NOT NULL,  
    calle character varying(60) NOT NULL,  
    num_ext character varying(4) NOT NULL,  
    num_int character varying(4),  
    col character varying(50) NOT NULL,  
    del character varying(35) NOT NULL,  
    cod_pos character varying(5) NOT NULL,  
    est character varying(21) NOT NULL,  
    tel_alu character varying(8),  
    tel_cel character varying(10),  
    cor_ele character varying(40),  
    fec_reg date,  
    cur_com character(1) DEFAULT '0'::bpchar NOT NULL,  
    cla_tal_sol character varying(2),  
    act_dat character(1) DEFAULT '0'::bpchar NOT NULL,  
    cla_abc integer,  
    pro character varying(5) NOT NULL,  
  
    PRIMARY KEY (num_cue),  
    FOREIGN KEY (cla_abc) REFERENCES horarios_abc(cla_abc),  
    FOREIGN KEY (cla_sit) REFERENCES situaciones_academicas(cla_sit),  
    FOREIGN KEY (cla_tal) REFERENCES talleres(cla_tal),  
    FOREIGN KEY (cla_tip) REFERENCES tipos_alumnos(cla_tip)  
  
);
```

### Comando SQL para insertar datos en la tabla alumnos:

```
INSERT INTO alumnos VALUES (301162403, '1', 'Act', '1', '28111983', 'CRUZ  
MENDOZA TOMAS', 1, ' TLÁLOC', '54', '2', 'COLONIAL', 'ECATEPEC', '69330',  
'ESTADO DE MÉXICO', '55443322', '5544332211', 'cruz_tomas@mail.com',  
NULL, ' ', '2', '1', 48, '7.00');
```

## Tabla alumnos\_carreras

Nombre: **NUM\_CUE**

Tipo de dato: integer

Tipo de llave: primaria, foránea

Valor nulo: no

Significado: el número de cuenta de cada Alumno.

Valores posibles: números de cuenta.

Nombre: **CLA\_CAR**

Tipo de dato: character

Longitud: 4

Tipo de llave: primaria, foránea

Valor nulo: no

Significado: Clave de la carrera de cada Alumno

Valores posibles: 102 para los que cursan la carrera de Arquitectura.

### Script para generar la tabla alumnos\_carreras:

```
CREATE TABLE alumnos_carreras (  
    cla_car character(4) NOT NULL,  
    num_cue integer NOT NULL,  
  
    PRIMARY KEY (cla_car, num_cue),  
    FOREIGN KEY (num_cue) REFERENCES alumnos(num_cue)  
);
```

### Comando SQL para insertar datos en la tabla alumnos\_carreras:

```
INSERT INTO alumnos_carreras VALUES ('102 ', 301168041);
```

## Tabla alumnos\_errores

Nombre: **NUM\_CUE**

Tipo de dato: integer

Tipo de llave: primaria, foránea

Valor nulo: no

Significado: el número de cuenta de cada Alumno.

Valores posibles: números de cuenta.

Nombre: **CLA\_ERR**

Tipo de dato: integer

Tipo de llave: primaria, foránea

Valor nulo: no

Significado: la clave correspondiente a los distintos mensajes de error de cada etapa.

Valores posibles:

- 1 Asignatura ya ha sido acreditada.
- 2 Se excede el número máximo de Exámenes Extraordinarios Largos permitidos por semestre.
- 3 No hay lugares disponibles para el grupo registrado.
- 4 El total de créditos para el semestre excede el máximo permitido de 51.
- 5 No cumple requisitos para inscribir Práctica Profesional Supervisada.
- 6 No cumple con requisitos para inscribir Curso Selectivo.
- 7 Número total de cursos selectivos sobrepasa el máximo permitido de 12.
- 8 No cumple con requisitos para inscribir curso selectivo.
- 9 No cumple con seriación por asignatura.
- 10 No cumple seriación por etapa.

Nombre: **CLA\_ASI**

Tipo de dato: integer

Tipo de llave: primaria

valor nulo: no

Significado: clave de cada asignatura de la carrera de Arquitectura.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

**Script para generar la tabla alumnos\_errores:**

```
CREATE TABLE alumnos_errores (  
    num_cue integer NOT NULL,  
    cla_err integer NOT NULL,  
    cla_así integer NOT NULL  
  
    PRIMARY KEY (num_cue, cla_err, cla_así),  
    FOREIGN KEY (cla_así) REFERENCES asignaturas(cla_así),  
    FOREIGN KEY (cla_err) REFERENCES errores(cla_err),  
    FOREIGN KEY (num_cue) REFERENCES alumnos(num_cue)  
  
);
```

**Comando SQL para insertar datos en la tabla alumnos\_errores:**

```
INSERT INTO alumnos_errores VALUES (301153689, 3, 1130);
```

## Tabla asignaturas

Nombre: **CLA\_ASI**

Tipo de dato: integer

Tipo de llave: primaria

valor nulo: no

Significado: clave de cada asignatura de la carrera de Arquitectura.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **CLA\_ETA**

Tipo de dato: integer

Tipo de llave: foránea

Valor nulo: no.

Significado: Clave de las etapas de la carrera de Arquitectura.

Valores posibles:

- |   |                |
|---|----------------|
| 1 | Básica         |
| 2 | Desarrollo     |
| 3 | Profundización |
| 4 | Consolidación  |
| 5 | Demostración   |

Nombre: **TIP\_ASI**

Tipo de dato: character

Longitud: 1

Valor nulo: no

Significado: Tipo de asignatura.

Valores posibles:

- |   |                        |
|---|------------------------|
| O | Asignatura Obligatoria |
| S | Asignatura Selectiva   |

Nombre: **NUM\_SEM**

Tipo de dato: character varying

Longitud: 2

Valor nulo: no

Significado: Número de semestre en que se imparte la asignatura.

Valores posibles: Los especificados en cada plan de estudios.

Nombre: **CRE**

Tipo de dato: integer

Valor nulo: no

Significado: Los créditos de la asignatura.

Valores posibles: los asignados para cada asignatura.

Nombre: **NOM\_ASI**

Tipo de dato: character varying

Longitud: 60

Valor nulo: no

Significado: el nombre de cada asignatura.

Valores posibles: los asignados para cada asignatura.

Nombre: **CUR\_COM**

Tipo de dato: character

Longitud: 1

Valor nulo: sí

Significado: indica si la asignatura necesita curso de cómputo para inscribirla.

Valores posibles:

0 No necesita curso de cómputo.

1 Necesita curso de cómputo.

**Script para generar la tabla asignaturas:**

```
CREATE TABLE asignaturas (  
  
    cla_asi integer NOT NULL,  
    cla_eta integer NOT NULL,  
    tip_asi character(1) NOT NULL,  
    num_sem character varying(2) NOT NULL,  
    cre integer NOT NULL,  
    nom_asi character varying(60) NOT NULL,  
    cur_com character(1),  
  
    PRIMARY KEY (cla_asi),  
    FOREIGN KEY (cla_eta) REFERENCES etapas(cla_eta)  
  
);
```

**Comando SQL para insertar datos en la tabla asignaturas:**

```
INSERT INTO asignaturas VALUES (1432, 2, 'O', '4', 4, 'TEORIA DE LA  
ARQUITECTURA IV', '0');
```

## Tabla asignaturas\_seriaciones

Nombre: **CLA\_ASI**

Tipo de dato: integer

Tipo de llave: primaria, foránea

Valor nulo: no

Significado: clave de cada asignatura de la carrera de Arquitectura.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **SER\_CON**

Tipo de dato: integer

Tipo de llave: primaria, foránea

Valor nulo: no

Significado: clave de la asignatura con la que hay seriación.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

### Script para generar la tabla asignaturas\_seriaciones:

```
CREATE TABLE asignaturas_seriaciones (  
    cla_asi integer NOT NULL,  
    ser_con integer NOT NULL,  
  
    PRIMARY KEY (cla_asi, ser_con)  
);
```

### Comando SQL para insertar datos en la tabla asignaturas\_seriaciones:

```
INSERT INTO asignaturas_seriaciones VALUES (324, 1036);  
INSERT INTO asignaturas_seriaciones VALUES (350, 1037);
```

## Tabla aulas

Nombre: **CLA\_AUL**

Tipo de dato: character varying

Longitud: 4

Tipo de llave: primaria.

Valor nulo: no.

Significado: la clave que identifica a cada aula de la Facultad de Arquitectura.

Valores posibles: las claves capturadas.

Nombre: **NOM\_AUL**

Tipo de dato: character varying

Longitud: 30

Valor nulo: no

Significado: El nombre de cada aula.

Valores posibles: Los asignados a cada aula.

### Script para generar la tabla aulas:

```
CREATE TABLE aulas (  
    cla_aul character varying(4) NOT NULL,  
    nom_aul character varying(30),  
    PRIMARY KEY (cla_aul)  
);
```

### Comando SQL para insertar datos en la tabla aulas:

```
INSERT INTO aulas VALUES ('5P', 'ENRIQUE DEL MORAL');
```

## Tabla carreras

Nombre: **CLA\_CAR**

Tipo de dato: character varying

Longitud: 4

Tipo de llave: primaria.

Valor nulo: si

Significado: Clave de la carrera de cada Coordinador de Carrera y Coordinador de Taller.

Valores posibles: 102 para los involucrados en la carrera de Arquitectura.

Nombre: **NOM\_CAR**

Tipo de dato: character varying

Longitud: 24

Valor nulo: no

Significado: Nombre de cada una de las carreras de la Facultad de Arquitectura.

Valores posibles: Arquitectura.

Nombre: **NUM\_ASI**

Tipo de dato: integer

Valor nulo: no

Significado: número de asignaturas de las carreras de la Facultad de Arquitectura.

Valores posibles: los especificados para cada carrera.

Nombre: **CRE\_OBL**

Tipo de dato: integer

Valor nulo: no

Significado: número total de créditos obligatorios para el plan vigente en las carreras de la Facultad de Arquitectura.

Valores posibles: los especificados para cada carrera.

344 para Arquitectura en el plan vigente.

Nombre: **CRE\_OPT**

Tipo de dato: integer

Valor nulo: no

Significado: número total de créditos optativos para el plan vigente en las carreras de la Facultad de Arquitectura.

Valores posibles: los especificados para cada carrera.

48 para Arquitectura en el plan vigente.

Nombre: **CRE\_TOT**

Tipo de dato: integer

Valor nulo: no

Significado: número total de créditos totales para el plan vigente en las carreras de la Facultad de Arquitectura.

Valores posibles: los especificados para cada carrera.

392 para Arquitectura en el plan vigente.

#### **Script para generar la tabla carreras:**

```
CREATE TABLE carreras (  
    cla_car character varying(4) NOT NULL,  
    nom_car character varying(24) NOT NULL,  
    num_asi integer NOT NULL,  
    cre_obl integer NOT NULL,  
    cre_opt integer NOT NULL,  
    cre_tot integer NOT NULL,  
  
    PRIMARY KEY (cla_car)  
);
```

#### **Comando SQL para insertar datos en la tabla carreras:**

```
INSERT INTO carreras VALUES ('102', 'Arquitectura', 54, 344, 48, 392);
```

## Tabla errores

Nombre: **CLA\_ERR**

Tipo de dato: integer

Tipo de llave: primaria, foránea

Valor nulo: no

Significado: la clave correspondiente a los distintos mensajes de error de cada etapa.

Valores posibles:

- 1 Asignatura ya ha sido acreditada.
- 2 Se excede el número máximo de Exámenes Extraordinarios Largos permitidos por semestre.
- 3 No hay lugares disponibles para el grupo registrado.
- 4 El total de créditos para el semestre excede el máximo permitido de 51.
- 5 No cumple requisitos para inscribir Práctica Profesional Supervisada.
- 6 No cumple con requisitos para inscribir Curso Selectivo.
- 7 Número total de cursos selectivos sobrepasa el máximo permitido de 12.
- 8 No cumple con requisitos para inscribir curso selectivo.
- 9 No cumple con seriación por asignatura.
- 10 No cumple seriación por etapa.

Nombre: **DES**

Tipo de dato: character

Longitud: 100

Valor nulo: si

Significado: Descripción de cada clave de error

Valores posibles:

- 1 Asignatura ya ha sido acreditada.
- 2 Se excede el número máximo de Exámenes Extraordinarios Largos permitidos por semestre.
- 3 No hay lugares disponibles para el grupo registrado.
- 4 El total de créditos para el semestre excede el máximo permitido de 51.
- 5 No cumple requisitos para inscribir Práctica Profesional Supervisada.
- 6 No cumple con requisitos para inscribir Curso Selectivo.
- 7 Número total de cursos selectivos sobrepasa el máximo permitido de 12.
- 8 No cumple con requisitos para inscribir curso selectivo.
- 9 No cumple con seriación por asignatura.
- 10 No cumple seriación por etapa.

**Script para generar la tabla errores:**

```
CREATE TABLE errores (  
    cla_err integer NOT NULL,  
    des character(100),  
  
    PRIMARY KEY (cla_err)  
  
);
```

**Comando SQL para insertar datos en la tabla errores:**

```
INSERT INTO errores VALUES (2, 'Se excede el número máximo de Exámenes  
Extraordinarios Largos permitidos por semestre.');
```

## Tabla etapas

Nombre: **CLA\_ETA**

Tipo de dato: integer

Tipo de llave: primaria.

Valor nulo: no

Significado: Clave de las etapas de la carrera de Arquitectura.

Valores posibles:

1 Básica

2 Desarrollo

3 Profundización

4 Consolidación

5 Demostración

Nombre: **CLA\_CAR**

Tipo de dato: character varying

Longitud: 4

Tipo de llave: foránea.

Valor nulo: si

Significado: Clave de la carrera de cada Etapa de la carrera de Arquitectura.

Valores posibles: 102 para las etapas de la carrera de Arquitectura.

Nombre: **NUM\_ASI**

Tipo de dato: integer

Valor nulo: no

Significado: Número de asignaturas correspondientes a cada etapa.

Valores posibles: los especificados para cada etapa.

Nombre: **NOM\_ETA**

Tipo de dato: character varying

Longitud: 25

Valor nulo: no

Significado: Nombre de cada etapa.

Valores posibles:

Básica

Desarrollo

Profundización

Consolidación

Demostración

#### Script para generar la tabla etapas:

```
CREATE TABLE etapas (  
    cla_eta integer NOT NULL,  
    cla_car character varying(4) NOT NULL,  
    num_asi integer NOT NULL,  
    nom_eta character varying(20) NOT NULL,  
    KEY (cla_eta),  
    FOREIGN KEY (cla_car) REFERENCES carreras(cla_car)  
);
```

#### Comando SQL para insertar datos en la tabla etapas:

```
INSERT INTO etapas VALUES (1, '102', 10, 'Básica');
```

## Tabla fechas

Nombre: **CLA\_FEC**

Tipo de dato: integer

Tipo de llave: primaria.

Valor nulo: no

Significado: Clave correspondiente a cada etapa.

Valores posibles: fechas con formato YYYY-MM-DD

Nombre: **FEC\_INI**

Tipo de dato: date

Valor nulo: no

Significado: Fecha de inicio de cada etapa.

Valores posibles: fechas con formato YYYY-MM-DD

Nombre: **FEC\_FIN**

Tipo de dato: date

Valor nulo: no

Significado: Fecha de fin de cada etapa.

Valores posibles: fechas con formato YYYY-MM-DD

Nombre: **DES**

Tipo de dato: character varying

Longitud: 40

Valor nulo: no

Significado: descripción del periodo de cada etapa.

Valores posibles:

- 1 Periodo de Registro
- 2 Periodo de Asignacion
- 3 Periodo de Altas, Bajas y Cambios
- 4 Periodo Previo ABC

**Script para generar la tabla fechas:**

```
CREATE TABLE fechas (  
    cla_fec integer NOT NULL,  
    fec_ini date NOT NULL,  
    fec_fin date NOT NULL,  
    des character varying(40) NOT NULL,  
  
    PRIMARY KEY (cla_fec)  
);
```

**Comando SQL para insertar datos en la tabla fechas:**

```
INSERT INTO fechas VALUES (1, '2009-03-12', '2009-03-13', 'Periodo de  
Registro');
```

## Tabla grupos

Nombre: **CLA\_GRU**

Tipo de dato: integer

Tipo de llave: primaria.

Valor nulo: no

Significado: Clave que identifica a cada grupo de las asignaturas.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **CLA\_ASI**

Tipo de dato: integer

Tipo de llave: foránea.

Valor nulo: no

Significado: clave de cada asignatura de la carrera de Arquitectura.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **CLA\_TAL**

Tipo de dato: character varying

Longitud: 2

Tipo de llave: foránea.

Valor nulo: no

Significado: Clave que identifica a cada Taller de la carrera de Arquitectura.

Valores posibles:

0	Sin taller
1	Taller Arquitecto Luís Barragán
2	Taller Arquitecto Max Cetto
3	Taller Ehécatl XXI
4	Taller Arquitecto Juan García Gayoú

- 5 Taller Arquitecto Domingo García ramos
- 6 Taller Arquitecto Jorge González Reyna
- 7 Taller Arquitecto Carlos Lazo Barreiro
- 8 Taller Arquitecto Carlos Leduc Montaña
- 9 Taller Arquitecto Ramón Marcos Noriega
- 10 Taller Arquitecto Federico Mariscal y Piña
- 11 Taller Arquitecto Hannes Meyer
- 12 Taller Arquitecto Juan O’Gorman
- 13 Taller José Revueltas
- 14 Taller Tres
- 15 Taller Uno
- 16 Taller Arquitecto José Villagrán García

Nombre: **CLA\_AUL**

Tipo de dato: character varying

Longitud: 4

Tipo de llave: foránea.

Valor nulo: no

Significado: la clave que identifica a cada aula de la Facultad de Arquitectura.

Valores posibles: las claves capturadas.

Nombre: **CLA\_INT\_GRU**

Tipo de dato: character varying

Longitud: 4

Valor nulo: no

Significado: la clave de grupo ordinario de cada asignatura manejada para fines de inscripciones.

Valores posibles: la clave asignada por las coordinaciones.

Nombre: **CLA\_EA**

Tipo de dato: character varying

Longitud: 4

Valor nulo: no

Significado: la clave de Examen Extraordinario Largo de cada grupo.

Valores posibles: Se compone de las letras EA seguidas de los 2 últimos dígitos de la clave de grupo ordinario (ver columna anterior).

Nombre: **CUP**

Tipo de dato: integer

Valor nulo: no

Significado: el cupo de cada grupo.

Valores posibles: desde 0 hasta el total de lugares asignados (por lo general de 60).

Nombre: **DEM**

Tipo de dato: integer

Valor nulo: no

Significado: la demanda de cada grupo (peticiones hechas durante la etapa de registro).

Valores posibles: desde 0 hasta un límite máximo de 2147483647 (el límite de los números enteros en postgresql).

Nombre: **INS**

Tipo de dato: integer

Valor nulo: no

Significado: el número de alumnos inscritos en el grupo.

Valores posibles: desde 0 hasta los que las reglas del sistema lo permitan.

Nombre: **MOD\_GRU**

Tipo de dato: character varying

Longitud: 4

Valor nulo: sí

Significado: La clave que identifica al actor de Asuntos Escolares que haya realizado la última modificación al grupo.

Valores posibles:

MA01 Mantenimiento

CT01 Coordinador del Taller Arquitecto Luís Barragán  
 CT02 Coordinador del Taller Arquitecto Max Cetto  
 CT03 Coordinador del Taller Ehécatl XXI  
 CT04 Coordinador del Taller Arquitecto Juan García Gayoú  
 CT05 Coordinador del Taller Arquitecto Domingo García ramos  
 CT06 Coordinador del Taller Arquitecto Jorge González Reyna  
 CT07 Coordinador del Taller Arquitecto Carlos Lazo Barreiro  
 CT08 Coordinador del Taller Arquitecto Carlos Leduc Montaña  
 CT09 Coordinador del Taller Arquitecto Ramón Marcos Noriega  
 CT10 Coordinador del Taller Arquitecto Federico Mariscal y Piña  
 CT11 Coordinador del Taller Arquitecto Hannes Meyer  
 CT12 Coordinador del Taller Arquitecto Juan O’Gorman  
 CT13 Coordinador del Taller José Revueltas  
 CT14 Coordinador del Taller Tres  
 CT15 Coordinador del Taller Uno  
 CT16 Coordinador del Taller Arquitecto José Villagrán García  
 CC01 Coordinador de Carrera.  
 JAE1 Jefe de Asuntos Escolares

Nombre: **EST**

Tipo de dato: character

Longitud: 1

Valor nulo: no

Significado: El estado en que el grupo se encuentra: cancelado o no.

Valores posibles:

0 Grupo Cancelado.

1 Grupo Activo.

Nombre: **FEC\_MOD**

Tipo de dato: date

Valor nulo: sí

Significado: guarda la fecha de la última modificación realizada al grupo.

Valores posibles: fechas con formato YY-MM-DD.

Nombre: **HOR\_MOD**

Tipo de dato: time without time zone

Valor nulo: sí

Significado: la hora de la última modificación realizada al grupo.

Valores posibles: horas con el formatos HH:MM:SS

#### Script para generar la tabla grupos:

```
CREATE TABLE grupos (  
    cla_gru integer DEFAULT nextval('gru_cla_gru_seq'::regclass) NOT  
NULL,  
    cla_asi integer NOT NULL,  
    cla_tal character varying(2) NOT NULL,  
    cla_aul character varying(4) NOT NULL,  
    cla_int_gru character varying(4),  
    cla_ea character varying(4),  
    cup integer NOT NULL,  
    dem integer NOT NULL,  
    ins integer NOT NULL,  
    mod_gru character varying(4),  
    est character(1) NOT NULL,  
    fec_mod date,  
    hor_mod time without time zone,  
  
PRIMARY KEY (cla_gru),  
FOREIGN KEY (cla_aul) REFERENCES aulas(cla_aul)  
);
```

#### Comando SQL para insertar datos en la tabla grupos:

```
INSERT INTO grupos VALUES (277, 1232, '2', 'A201', '5202', 'EA02', 50, 0,  
0, 'CT02', '1', '2008-12-22', '21:19:05');
```

## Tabla historias\_academicas\_completas

Nombre: **CLA\_HIS\_ACA**

Tipo de dato: integer

Tipo de llave: primaria.

Valor nulo: no

Significado: la clave que identifica a cada registro de cada historia académica.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **CLA\_ASI**

Tipo de dato: integer

Tipo de llave: foránea.

Valor nulo: no

Significado: clave de cada asignatura de la carrera de Arquitectura.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **NUM\_CUE**

Tipo de dato: integer

Tipo de llave: foránea.

Valor nulo: no

Significado: el número de cuenta de cada Alumno de Arquitectura.

Valores posibles: números de cuenta.

Nombre: **PER**

Tipo de dato: character varying

Longitud: 6

Valor nulo: no

Significado: el período que aparece en el registro de la historia académica.

Valores posibles: semestres.

Nombre: **GRU**

Tipo de dato: character varying

Longitud: 4

Valor nulo: no

Significado: el grupo en que el Alumno cursó la asignatura.

Valores posibles: el grupo que aparece en el registro de la historia académica.

Nombre: **CAL**

Tipo de dato: character varying

Longitud: 2

Valor nulo: no

Significado: la calificación que aparece en el registro de la historia académica.

Valores posibles: calificaciones.

Nombre: **TIP\_EXA**

Tipo de dato: character

Longitud: 1

Valor nulo: no

Significado: tipo de examen que aparece en la historia académica.

Valores posibles:

ORD	Ordinario
EXT	Extraordinario.

Nombre: **FOL\_ACT**

Tipo de dato: character varying

Longitud: 7

Valor nulo: no

Significado: el folio que aparece en la historia académica.

Valores posibles: folios.

Nombre: **MOD\_CAL**

Tipo de dato: character varying

Longitud: 4

Tipo de llave: foránea.

Valor nulo: sí

Significado: La clave que identifica al actor de Asuntos Escolares que haya realizado la última modificación a la historia académica.

Valores posibles:

MA01 Mantenimiento

JAE1 Jefe de Asuntos Escolares

Nombre: **FEC\_MOD**

Tipo de dato: date

Valor nulo: sí

Significado: guarda la fecha de la última modificación realizada a la historia académica.

Valores posibles: fechas con formato YYYY-MM-DD.

Nombre: **HOR\_MOD**

Tipo de dato: time without time zone

Valor nulo: sí

Significado: la hora de la última modificación realizada a la historia académica.

Valores posibles: horas con el formato HH:MM:SS

#### **Script para generar la tabla historias\_academicas\_completas:**

```
CREATE TABLE historias_academicas_completas (  
    cla_his_aca integer DEFAULT  
nextval('his_aca_com_cla_his_aca'::regclass) NOT NULL,  
    cla_asi integer NOT NULL,  
    num_cue integer NOT NULL,  
    per character varying(6) NOT NULL,  
    gru character varying(4) NOT NULL,  
    cal character varying(2) NOT NULL,  
    tip_exa character(1) NOT NULL,  
    fol_act character varying(7) NOT NULL,  
    fec_mod date,
```

```
hor_mod time without time zone,  
mod_cal character varying(4),  
  
PRIMARY KEY (cla_his_aca),  
FOREIGN KEY (cla_asi) REFERENCES asignaturas(cla_asi),  
FOREIGN KEY (mod_cal) REFERENCES actores_asuntos_escolares(cla_aae),  
FOREIGN KEY (num_cue) REFERENCES alumnos(num_cue)  
  
);
```

**Comando SQL para insertar datos en la tabla historias\_academicas\_completas:**

```
INSERT INTO historias_academicas_completas VALUES (68, 1130, 301152280,  
'20061', '5101', '9', '0', '0024040', '2009-02-03', '20:41:27', 'MA01');
```

## Tabla horarios\_abc

Nombre: **CLA\_ABC**

Tipo de dato: integer

Tipo de llave: primaria.

Valor nulo: no

Significado: La clave que identifica a cada turno para el periodo de Altas, Bajas y Cambios.

Valores posibles: Valores enteros resultado del proceso de Asignación de horarios para Altas, Bajas y Cambios correspondiente al usuario Mantenimiento.

Nombre: **FEC\_ABC**

Tipo de dato: date

Valor nulo: no

Significado: Fecha correspondiente a cada turno para el periodo de Altas, Bajas y Cambios.

Valores posibles: Fechas con formato YYYY-MM-DD.

Nombre: **HOR\_ABC**

Tipo de dato: character

Longitud: 5

Valor nulo: no

Significado: Hora correspondiente a cada turno para el periodo de Altas, Bajas y Cambios.

Valores posibles: Horas con formato HH:MM.

**Script para generar la tabla horarios\_abc:**

```
CREATE TABLE horarios_abc (  
    cla_abc integer DEFAULT nextval('hor_abc_cla_abc_seq'::regclass) NOT  
NULL,  
    fec_abc date NOT NULL,  
    hor_abc character(5) NOT NULL,  
  
PRIMARY KEY (cla_abc)  
);
```

**Comando SQL para insertar datos en la tabla horarios\_abc:**

```
INSERT INTO horarios_abc VALUES (1, '2009-02-08', '08:00');
```

## Tabla horarios\_grupos

Nombre: **CLA\_HOR\_GRU**

Tipo de dato: integer

Tipo de llave: primaria.

Valor nulo: no

Significado: es la clave para cada horario de cada grupo debido a que cada grupo puede o no tener varios horarios en distintos días a distintas horas.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647

Nombre: **CLA\_GRU**

Tipo de dato: integer

Tipo de llave: foránea

Valor nulo: no

Significado: es la clave que identifica a cada grupo.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647

Nombre: **DIA**

Tipo de dato: character

Longitud: 1

Valor nulo: no

Significado: Día de la semana en que se imparte la clase de la asignatura.

Valores posibles:

- 1.- L Lunes
- 2.- M Martes
- 3.- I Miércoles
- 4.- J Jueves
- 5.- V Viernes

## 6.-S Sábado

Nombre: **HOR\_INI**

Tipo de dato: character

Longitud: 5

Significado: Hora en que da inicio la clase de la asignatura.

Valores posibles: valores de horas desde las 07:00 hasta las 20:00

Nombre: **HOR\_FIN**

Tipo de dato: character

Longitud: 5

Tipo de dato: CHAR

Longitud: 4

Valor nulo: no

Significado: Hora en que finaliza la clase de la asignatura.

Valores posibles: valores de horas desde las 08:00 hasta las 22:00

### Script para generar la tabla horarios\_grupos:

```
CREATE TABLE horarios_grupos (  
    cla_hor_gru integer DEFAULT  
nextval('hor_gru_cla_hor_gru_seq'::regclass) NOT NULL,  
    cla_gru integer,  
    dia character(1),  
    hor_ini character(5),  
    hor_fin character(5),  
  
PRIMARY KEY (cla_hor_gru)  
FOREIGN KEY (cla_gru) REFERENCES grupos(cla_gru)  
);
```

### Comando SQL para insertar datos en la tabla horarios\_grupos:

```
INSERT INTO horarios_grupos VALUES (1, 1, 'M', '15:30', '17:00');
```

## Tabla inscripciones

Nombre: **NUM\_CUE**

Tipo de dato: integer

Tipo de llave: primaria, foránea.

Valor nulo: no

Significado: el número de cuenta de cada Alumno.

Valores posibles: números de cuenta.

Nombre: **CLA\_ASI**

Tipo de dato: integer

Tipo de llave: primaria, foránea.

Valor nulo: no

Significado: clave de cada asignatura de la carrera de Arquitectura.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **CLA\_GRU**

Tipo de dato: integer

Tipo de llave: foránea.

Valor nulo: no

Significado: es la clave del grupo en que el Alumno se ha inscrito.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647

Nombre: **REA\_POR**

Tipo de dato: character

Longitud: 4

Tipo de llave: foránea.

Valor nulo: no

Significado: La clave que identifica al actor que haya realizado la inscripción al grupo.

Valores posibles:

MA01 Mantenimiento  
CT01 Coordinador del Taller Arquitecto Luís Barragán  
CT02 Coordinador del Taller Arquitecto Max Cetto  
CT03 Coordinador del Taller Ehécatl XXI  
CT04 Coordinador del Taller Arquitecto Juan García Gayoú  
CT05 Coordinador del Taller Arquitecto Domingo García ramos  
CT06 Coordinador del Taller Arquitecto Jorge González Reyna  
CT07 Coordinador del Taller Arquitecto Carlos Lazo Barreiro  
CT08 Coordinador del Taller Arquitecto Carlos Leduc Montañó  
CT09 Coordinador del Taller Arquitecto Ramón Marcos Noriega  
CT10 Coordinador del Taller Arquitecto Federico Mariscal y Piña  
CT11 Coordinador del Taller Arquitecto Hannes Meyer  
CT12 Coordinador del Taller Arquitecto Juan O’Gorman  
CT13 Coordinador del Taller José Revueltas  
CT14 Coordinador del Taller Tres  
CT15 Coordinador del Taller Uno  
CT16 Coordinador del Taller Arquitecto José Villagrán García  
CC01 Coordinador de Carrera  
JAE1 Jefe de Asuntos Escolares  
Números de cuenta de alumnos.

Nombre: **FEC**

Tipo de dato: date

Valor nulo: no

Significado: la fecha en que se realizó la inscripción al grupo.

Valores posibles: fechas con formato DD-MM-YYYY.

Nombre: **HOR**

Tipo de dato: character

Longitud: 8

Valor nulo: no

Significado: la hora en que se realizó la inscripción al grupo.

Valores posibles: horas con el formato HH:MM:SS

Nombre: **BAJ**

Tipo de dato: character

Longitud: 1

Valor nulo: sí

Significado: indica si la inscripción se ha dado o no de baja.

Valores posibles:

1 Registro activo.

0 Registro dado de baja.

Nombre: **TIP\_GRU**

Tipo de dato: character

Longitud: 1

Valor nulo: sí

Significado: indica si la inscripción se ha dado o no de baja.

Valores posibles:

1 Grupo ordinario.

0 Grupo extraordinario

**Script para generar la tabla inscripciones:**

```
CREATE TABLE inscripciones (  
  
    num_cue integer NOT NULL,  
    cla_asi integer NOT NULL,  
    cla_gru integer NOT NULL,  
    rea_por character(4) NOT NULL,  
    fec date,  
    hor character(8),  
    baj character(1),  
    tip_gru character(1),  
  
    PRIMARY KEY (num_cue, cla_asi),  
    FOREIGN KEY (cla_asi) REFERENCES asignaturas(cla_asi),  
    FOREIGN KEY (cla_gru) REFERENCES grupos(cla_gru),  
    FOREIGN KEY (num_cue) REFERENCES alumnos(num_cue)  
  
);
```

**Comando SQL para insertar datos en la tabla inscripciones:**

```
INSERT INTO inscripciones VALUES (301163383, 1536, 685, 'MA01', '2009-02-  
23', '12:54:05', '1', '1');
```

## Tabla planes

Nombre: **CLA\_PLA**

Tipo de dato: character varying

Longitud: 4

Tipo de llave: primaria.

Valor nulo: no.

Significado: la clave del plan de estudio.

Valores posibles: los especificados al momento de realizar la captura.

Nombre: **NOM\_PLA**

Tipo de dato: character varying

Longitud: 45

Valor nulo: no.

Significado: el nombre del plan de estudios.

Valores posibles: los especificados al momento de realizar la captura.

### Script para generar la tabla planes:

```
CREATE TABLE planes (  
    cla_pla character varying(4) DEFAULT ''::character varying NOT NULL,  
    nom_pla character varying(45) DEFAULT ''::character varying NOT NULL,  
  
    PRIMARY KEY (cla_pla)  
);
```

### Comando SQL para insertar datos en la tabla planes:

```
INSERT INTO planes VALUES ('0842', 'Arquitecto');
```

## Tabla profesores

Nombre: **CLA\_PRO**

Tipo de dato: character varying

Longitud: 18

Tipo de llave: primaria.

Valor nulo: no

Significado: Clave que identifica a cada profesor.

Valores posibles: varios, pudiendo ser CURP, RFC, número de cuenta o alguna asignada al carecer de los datos anteriores.

Nombre: **NOM\_PRO**

Tipo de dato: character varying

Longitud: 45

Valor nulo: no

Significado: el nombre de cada profesor.

Valores posibles: los nombres capturados.

### Script para generar la tabla profesores:

```
CREATE TABLE profesores (  
    cla_pro character varying(18) NOT NULL,  
    nom_pro character varying(45),  
    PRIMARY KEY (cla_pro)  
);
```

### Comando SQL para insertar datos en la tabla profesores:

```
INSERT INTO profesores VALUES ('AAAE5007151NA', 'DE ANDA ALANIS ENRIQUE  
JAVIER, ARQ.');
```

## Tabla profesores\_grupos

Nombre: **CLA\_PRO\_GRU**

Tipo de dato: integer

Tipo de llave: primaria, foránea.

Valor nulo: no

Significado: Clave que identifica a cada grupo correspondiente a un profesor.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **CLA\_PRO**

Tipo de dato: character varying

Longitud: 18

Tipo de llave: primaria, foránea.

Valor nulo: no

Significado: Clave que identifica a cada profesor.

Valores posibles: varios, pudiendo ser CURP, RFC, número de cuenta o alguna asignada al carecer de los datos anteriores.

Nombre: **CLA\_GRU**

Tipo de dato: integer

Tipo de llave: foránea.

Valor nulo: no

Significado: Clave que identifica a cada grupo de las asignaturas.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

**Script para generar la tabla profesores\_grupos:**

```
CREATE TABLE profesores_grupos (  
    cla_pro_gru integer DEFAULT  
nextval('pro_gru_cla_pro_gru_seq'::regclass) NOT NULL,  
    cla_pro character varying(18),  
    cla_gru integer,  
  
PRIMARY KEY (cla_pro_gru),  
FOREIGN KEY (cla_gru) REFERENCES grupos(cla_gru),  
FOREIGN KEY (cla_pro) REFERENCES profesores(cla_pro)  
  
);
```

**Comando SQL para insertar datos en la tabla profesores\_grupos:**

```
INSERT INTO profesores_grupos VALUES (1, 'GASS4502229Z2', 139);
```

## Tabla registros

Nombre: **CLA\_ASI**

Tipo de dato: integer

Tipo de llave: primaria, foránea.

Valor nulo: no

Significado: clave de cada asignatura de la carrera de Arquitectura.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647.

Nombre: **NUM\_CUE**

Tipo de dato: integer

Tipo de llave: primaria, foránea.

Valor nulo: no

Significado: el número de cuenta de cada Alumno de Arquitectura.

Valores posibles: números de cuenta.

Nombre: **CLA\_GRU**

Tipo de dato: integer

Tipo de llave: primaria, foránea.

Valor nulo: no

Significado: es la clave del grupo en que el alumno se ha registrado.

Valores posibles: una sucesión numérica entera comenzando desde 1 hasta un límite máximo de 2147483647

Nombre: **BAJ**

Tipo de dato: character

Longitud: 1

Valor nulo: sí

Significado: indica si el registro se ha dado o no de baja.

Valores posibles:

- 1 Registro activo.
- 0 Registro dado de baja.

**Script para generar la tabla inscripciones:**

```
CREATE TABLE registros (  
    cla_asi integer NOT NULL,  
    num_cue integer NOT NULL,  
    cla_gru integer NOT NULL,  
    baj character(1),  
  
    PRIMARY KEY (cla_asi, num_cue),  
    FOREIGN KEY (cla_asi) REFERENCES asignaturas(cla_asi),  
    FOREIGN KEY (cla_gru) REFERENCES grupos(cla_gru),  
    FOREIGN KEY (num_cue) REFERENCES alumnos(num_cue)  
);
```

**Comando SQL para insertar datos en la tabla inscripciones:**

```
INSERT INTO registros VALUES (1230, 301168041, 221, '1');
```

## Tabla situaciones\_academicas

Nombre: **CLA\_SIT**

Tipo de dato: character varying

Longitud: 3

Tipo de llave: primaria.

Valor nulo: no

Significado: la situación académica en que se encuentra el Alumno.

Valores posibles:

Act	Activo
SuT	Suspensión Temporal
A22	Artículo 22

Nombre: **DES**

Tipo de dato: character varying

Longitud: 28

Valor nulo: no

Significado: descripción de la situación en que se encuentra el Alumno.

Valores posibles:

Activo
Suspensión Temporal
Artículo 22

**Script para generar la tabla situaciones\_academicas:**

```
CREATE TABLE situaciones_academicas (  
    cla_sit character varying(3) NOT NULL,  
    des character varying(28) NOT NULL,  
  
    PRIMARY KEY (cla_sit)  
);
```

**Comando SQL para insertar datos en la tabla situaciones\_academicas:**

```
INSERT INTO situaciones_academicas VALUES ('A22', 'Artículo 22');
```

## Tabla talleres

Nombre: **CLA\_TAL**

Tipo de dato: character varying

Longitud: 2

Tipo de llave: primaria.

Valor nulo: no

Significado: Clave que identifica a cada Taller de la carrera de Arquitectura.

Valores posibles:

0	Sin Taller
1	Taller Arquitecto Luís Barragán
2	Taller Arquitecto Max Cetto
3	Taller Ehécatl XXI
4	Taller Arquitecto Juan García Gayoú
5	Taller Arquitecto Domingo García Ramos
6	Taller Arquitecto Jorge González Reyna
7	Taller Arquitecto Carlos Lazo Barreiro
8	Taller Arquitecto Carlos Leduc Montañaño
9	Taller Arquitecto Ramón Marcos Noriega
10	Taller Arquitecto Federico Mariscal y Piña
11	Taller Arquitecto Hannes Meyer
12	Taller Arquitecto Juan O’Gorman
13	Taller José Revueltas
14	Taller Tres
15	Taller Uno
16	Taller Arquitecto José Villagrán García

Nombre: **NOM\_TAL**

Tipo de dato: character varying

Longitud: 42

Valor nulo: no

Significado: nombre de cada uno de los talleres de la carrera de Arquitectura.

Valores posibles:

Taller Arquitecto Luís Barragán

Taller Arquitecto Max Cetto

Taller Ehécatl XXI

Taller Arquitecto Juan García Gayoú

Taller Arquitecto Domingo García ramos

Taller Arquitecto Jorge González Reyna

Taller Arquitecto Carlos Lazo Barreiro

Taller Arquitecto Carlos Leduc Montaña

Taller Arquitecto Ramón Marcos Noriega

Taller Arquitecto Federico Mariscal y Piña

Taller Arquitecto Hannes Meyer

Taller Arquitecto Juan O'Gorman

Taller José Revueltas

Taller Tres

Taller Uno

Taller Arquitecto José Villagrán García

Nombre: **TUR\_TAL**

Tipo de dato: character

Longitud: 1

Valor nulo: no

Significado: el turno de cada taller.

Valores posibles:

M Matutino

V Vespertino

**Script para generar la tabla talleres:**

```
CREATE TABLE talleres (  
    cla_tal character varying(2) NOT NULL,  
    nom_tal character varying(42) NOT NULL,  
    tur_tal character(1) NOT NULL,  
  
    PRIMARY KEY (cla_tal)  
);
```

**Comando SQL para insertar datos en la tabla talleres:**

```
INSERT INTO talleres VALUES ('1', 'Taller Arquitecto Luis Barrag&acute;  
n', ' ');
```

## Tabla tipos\_actores\_asuntos\_escolares

Nombre: **CLA\_TIP**

Tipo de dato: character

Longitud: 1

Tipo de llave: primaria.

Valor nulo: no

Significado: Tipo de Actor de Asuntos Escolares.

Valores posibles:

- 1 Mantenimiento
- 2 Coordinador de Taller
- 3 Coordinador de Carrera
- 4 Jefe de Asuntos Escolares

Nombre: **CLA\_DES**

Tipo de dato: character varying

Longitud: 25

Valor nulo: no

Significado: Descripción de cada tipo de Actor de Asuntos Escolares.

Valores posibles:

- Mantenimiento
- Coordinador de Taller
- Coordinador de Carrera
- Jefe de Asuntos Escolares

**Script para generar la tabla tipos\_actores\_asuntos\_escolares:**

```
CREATE TABLE tipos_actores_asuntos_escolares (  
    cla_tip character(1) NOT NULL,  
    des character varying(25) NOT NULL,  
  
    PRIMARY KEY (cla_tip)  
);
```

**Comando SQL para insertar datos en la tabla tipos\_actores\_asuntos\_escolares:**

```
INSERT INTO tipos_actores_asuntos_escolares VALUES ('1',  
'Mantenimiento');
```

## Tabla tipos\_alumnos

Nombre: **CLA\_TIP**

Tipo de dato: character

Longitud: 1

Tipo de llave: primaria.

Valor nulo: no

Significado: el tipo de Alumno para fines de registro e inscripción.

Valores posibles:

- |   |                |
|---|----------------|
| 1 | Taller         |
| 2 | Regular        |
| 3 | Irregular      |
| 4 | Extraordinario |

Nombre: **DES**

Tipo de dato: character varying

Longitud: 40

Valor nulo: no

Significado: descripción del tipo de Alumno para fines de registro e inscripción.

Valores posibles:

- |                |
|----------------|
| Taller         |
| Regular        |
| Irregular      |
| Extraordinario |

**Script para generar la tabla tipos\_alumnos:**

```
CREATE TABLE tipos_alumnos (  
    cla_tip character(1) NOT NULL,  
    des character varying(40) NOT NULL,  
  
    PRIMARY KEY (cla_tip)  
);
```

**Comando SQL para insertar datos en la tabla tipos\_alumnos:**

```
INSERT INTO tipos_alumnos VALUES ('1', 'Taller');
```

# Glosario

**Alumnos Regulares:** Se denomina de esta manera a los alumnos que llevan las materias que corresponden a su semestre, es decir, que no se han rezagado en el avance de sus asignaturas. Si el alumno no acredita Taller o no lo inscribe ya no es considerado alumno regular.

**Alumnos Irregulares:** Se denomina de esta manera a los alumnos que se han rezagado en el avance de sus asignaturas sin que entren en la aplicación de los artículos 22 y 33 del Reglamento General de Inscripciones. Es decir, que estén rezagados, pero que aún puedan cursar todas sus materias de forma ordinaria.

**Apache:** El servidor HTTP Apache es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh entre otras.

**API:** Interfaz de programación de aplicaciones, es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

**Artículo 22:** Se aplica a aquellos alumnos que no han concluido sus estudios en el período máximo indicado por el Reglamento General de Inscripciones (15 semestres contabilizados a partir del ingreso a la licenciatura): “Para concluir sus estudios estos alumnos sólo podrán acreditar las asignaturas faltantes a través de exámenes extraordinarios”.

En la facultad de Arquitectura, si un Alumno se encuentra en estado de Artículo 22, solo puede inscribir EA, así como EB (no manejado por este sistema) y cursos de regularización (no manejado tampoco por este sistema), siendo éstos cursos en total un máximo de 6 (el sistema solo validaría los EA). Debe de respetar la seriación tanto de etapas como de asignaturas.

**Artículo 33:** Se aplica a aquellos alumnos que han estado inscritos de forma ordinaria dos veces en una misma asignatura y no la han acreditado: “Estos alumnos únicamente podrán acreditar la asignatura mediante examen extraordinario”.

**Asignatura compatible al Taller:** Es una asignatura que pertenece únicamente a un Taller.

**ASP.NET:** Es una tecnología libre que permite a los programadores crear aplicaciones web dinámicas.

**BSD:** Son las iniciales de *Berkeley Software Distribution* (Distribución de Software Berkeley) y se utiliza para identificar un sistema operativo derivado del sistema Unix nacido a partir de los aportes realizados a ese sistema.

**C:** Es un lenguaje de programación de alto nivel desarrollado en el año 1972 por Dennis Ritchie en AT&T Bell Labs. La legibilidad, facilidad de mantenimiento y la portabilidad son algunas de las ventajas de este lenguaje, además que permite descender a nivel de hardware.

**C++:** Es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue el extender al exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido.

**Cliente:** El cliente web es el encargado de ejecutar aplicaciones en la máquina del usuario. Son las aplicaciones tipo Java o Javascript: el servidor proporciona el código de las aplicaciones al cliente y éste, mediante el navegador, las ejecuta. Es necesario, por tanto, que el cliente disponga de un navegador con capacidad para ejecutar aplicaciones (también llamadas scripts).

**Código:** Término utilizado para describir instrucciones con la finalidad de ser usados por una computadora.

**Código abierto (Open source):** Es un método de desarrollo de software que es distribuido libremente. La finalidad del código abierto es crear software de mejor calidad, mayor fiabilidad, una mayor flexibilidad y a un menor costo.

**Curso Obligatorio:** Son cursos complementarios a los talleres, mismos que pertenecen a la carrera en que el alumno está inscrito.

**Curso de Regularización:** Son cursos que el alumno puede tomar en caso de no haber aprobado alguna materia en el semestre. Estos se toman al finalizar el semestre y cada curso de regularización cuenta como un examen extraordinario.

**Curso Selectivo:** Los cursos selectivos se refieren a asignaturas optativas y de otras carreras que pueden llevarse.

Los cursos selectivos impartidos dentro de la facultad de Arquitectura se manejan en el sistema (se les denomina cursos selectivos internos.)

Los cursos selectivos que pertenecen a otras facultades deben inscribirse llenando un formato llamado F306, el cual puede descargarse en la misma página de servicios escolares de la Facultad de Arquitectura y en caso de ser aprobada la solicitud, ésta se hace llegar al Ing. Rodrigo Rosas para que registre a los alumnos (no es en línea); esto no se maneja por el sistema desarrollado.

**Cygwin:** Es un entorno Linux para Windows de fácil instalación y gran flexibilidad. Esta compuesto por dos partes: Un DLL que funciona como un emulador API de Linux y una colección de herramientas que ofrece Linux para un mejor medio de trabajo.

**CSS:** Es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos.

**DBMS:** Los sistemas de gestión de base de datos (SGBD) consiste en una colección de datos interrelacionados, denominada base de datos, y un conjunto de programas para acceder a estos. El objetivo del SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera práctica y eficiente.

**Diccionario de datos:** Es donde se guarda la información relativa a la estructura de la BD. Es una base de datos que sirve para documentar los datos.

**Extraordinario Corto:** Es un tipo examen extraordinario como los que existen en otras facultades de la UNAM, con la diferencia de que pueden presentarse aún sin haber cursado la asignatura correspondiente. El alumno únicamente debe respetar la seriación del plan de estudios vigente para cada carrera de la Facultad.

**Extraordinario Largo:** Equivalente a ser Oyente (alumno sin derecho a reinscripción por los artículos 22 y 33 del Reglamento General de Inscripciones). El alumno realiza este

procedimiento de inscripción durante la etapa de registro de asignaturas, lo cual significa que debe de tomar la asignatura como un alumno regular; pero en su historial académico quedará asentada como un examen extraordinario.

**Folio de registro:** Anteriormente Formato F99. Es el registro que realiza el Coordinador de cada Taller a los alumnos para ser inscritos en el mismo. Esto es, se lleva a cabo una entrevista coordinador-estudiante, en la cual el coordinador decide si el alumno es admitido; si esto es así se realiza la inscripción del alumno en ese taller.

**Formato 99A:** Es un folio de registro de asignaturas, que se utiliza en caso de que el alumno no haya inscrito un taller. Este folio es asignado por el sistema de reinscripción en la etapa de registro de asignaturas.

**Formato F306:** Es un formato que sirve para registrar asignaturas de otros planteles y/o de otras carreras de la misma facultad de Arquitectura. La información debe ser llenada por el alumno en la misma página de servicios escolares de la Facultad de Arquitectura y posteriormente debe de ser impresa (no forma parte del sistema de reinscripciones). Una vez impreso, el alumno debe de recabar las firmas de el Coordinador académico de su carrera, Secretaria de Asuntos Escolares de la Facultad de Arquitectura, coordinador académico de la carrera destino, Secretario de Asuntos Escolares del Plantel Destino. Finalmente se entregan 2 copias en la Secretaría de Asuntos Escolares.

**FTP:** Protocolo de Transferencia de Archivos, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP, basado en la arquitectura cliente-servidor.

**Java:** Es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems. Permite desarrollar software en una plataforma y ejecutarlo en prácticamente cualquier otra plataforma, crear programas para que funcionen en un navegador web y en servicios web, desarrollar aplicaciones para servidores como foros en línea, tiendas, encuestas, procesamiento de formularios HTML, entre otras aplicaciones.

**Javascript:** Es un lenguaje interpretado basado en guiones que son integrados directamente en el código HTML. El código es transferido al cliente para que este lo interprete al cargar la página.

**HTML:** Lenguaje de Marcado de Hipertexto, compuesto por una serie de tags que permiten definir la estructura lógica de un documento web y establecer los atributos del mismo.

**HTTP:** El protocolo de transferencia de hipertexto, usado en cada transacción de la Web (WWW). Especifica un conjunto de métodos y encabezados que permiten a los clientes y a los servidores interactuar e intercambiar información de una manera uniforme.

**Inscripción de manera extemporánea:** Es una inscripción de un Alumno a una asignatura, fuera del periodo de inscripciones regulares, realizada por el Jefe de Asuntos Escolares.

**Link:** Es el texto o gráfico en un documento HTML que solicita a un programa web browser cargar otro documento o gráfico.

**LDD:** Lenguaje de Definición de Datos, permite a los usuarios definir la estructura y organización de los datos almacenados así como sus relaciones entre los ítems de los datos ya almacenados. Por sus siglas en inglés suele denominarse DDL también.

**LMD:** Es el lenguaje de manipulación de datos; permite realizar la recuperación y actualización (inserción, actualización y borrado) de los datos almacenados en la base de datos. Por sus siglas en inglés suele denominarse DML también.

**Linux:** Es un sistema operativo, compatible Unix. Pertenece a la corriente open source y viene acompañado del código fuente. El sistema lo forman el núcleo del sistema (kernel) más un gran número de programas / bibliotecas que hacen posible su utilización. Muchos de estos programas y bibliotecas han sido posibles gracias al proyecto GNU, por esto mismo, muchos llaman a Linux, GNU/Linux.

**Microsoft Access:** Es un programa Sistema de gestión de base de datos relacional creado y modificado por Microsoft. Gracias a la interfaz de usuario de Microsoft Office Fluent y sus capacidades de diseño, Microsoft Office Access facilita el seguimiento y la creación de informes. Puede recopilar información mediante formularios por correo electrónico o importe datos de aplicaciones externas y compartir información moviendo los archivos de Office Access a un sitio Web, donde puede realizar una auditoría del historial de revisiones, recuperar la información eliminada, establecer permisos de acceso a los datos y crear copias de seguridad de la información a intervalos regulares.

**OS X:** Es una línea de sistemas operativos computacionales desarrollada, comercializada y vendida por Apple Inc. Se basa en Unix y usa una interfaz gráfica desarrollada por Apple llamada Aqua, que se inspira libremente en la interfaz de Mac OS Classic.

**ODBC:** Es un estándar de acceso a Bases de datos desarrollado por Microsoft Corporation, el objetivo de *ODBC* es hacer posible el acceder a cualquier dato desde cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos almacene los datos

**PDF:** Es un formato de documento portátil que permite obtener y visualizar información desde cualquier aplicación y en cualquier sistema informático, así como compartirla con prácticamente cualquier persona en cualquier sitio.

**Perl:** Es un lenguaje de programación diseñado por Larry Wall en 1987. Perl toma características del lenguaje C, del lenguaje interpretado shell, AWK, sed, Lisp y, en un grado inferior, de muchos otros lenguajes de programación. Estructuralmente, Perl está basado en un estilo de bloques como los del C o AWK, y fue ampliamente adoptado por su destreza en el procesado de texto y no tener ninguna de las limitaciones de los otros lenguajes de script.

**PHP:** Acrónimo de "PHP: Hypertext Preprocessor, es un lenguaje de código abierto interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor.

**Prototipo:** Es una representación del diseño de un sistema que permite a las partes responsables de su desarrollo evaluarlo, detallarlo y probarlo en situaciones reales.

**Python:** Es un lenguaje de programación dinámico orientado a objetos que es utilizado para el desarrollo de muchos tipos de software. Ofrece un fuerte soporte para su integración con otros lenguajes y herramientas, contiene extensas librerías estándar impulsando con ello el desarrollo de software de mayor calidad y facilidad de mantenimiento en el código.

**Sesión:** Es una conexión entre dos máquinas en una red o sobre Internet.

**Script:** Es un programa que puede acompañar a un documento HTML o que puede estar incluido en él. El programa se ejecuta en la máquina del cliente cuando se carga el documento, o en algún otro instante, como por ejemplo cuando se activa un vínculo. El soporte de scripts de HTML es independiente del lenguaje de scripts.

**Servidor:** En internet, los servidores son los proveedores de todos sus servicios, incluyendo la WWW (las páginas web), el FTP, el correo electrónico, los grupos de noticias, entre otras.

**Sistema operativo:** Un sistema operativo optimiza todos los recursos del sistema para soportar los requerimientos. Controla todas las partes del sistema y presenta al usuario una interfaz o máquina virtual. Se ejecuta en modo central o modo de supervisión, con máxima prioridad y generalmente con protección por hardware.

**SQL:** Es una abreviatura de *Structured Query Language* (lenguaje de consulta estructurado). Es una herramienta para organizar, manejar y consultar información almacenada en una base de datos.

**Suspensión Temporal:** Es manejada por dos sistemas independientes al desarrollado:

- Primero el alumno debe de registrar al menos una asignatura en el sistema de reinscripciones en la etapa de registro; ya que para ser alumno debe de estar inscrito en alguna asignatura.
- En el segundo sistema (localizado en el menú de trámites de la página de Asuntos Escolares) el alumno llena una solicitud de suspensión temporal la cual es entregada en la Secretaría de Asuntos Escolares.

El tiempo en el cual el alumno está dentro de suspensión temporal no se contabiliza dentro de los 15 semestres que tiene para terminar sus créditos.

**Taller de Arquitectura:** Se le considera como la parte central de la carrera de Arquitectura. Éste es el espacio académico donde confluyen los conocimientos teóricos, tecnológicos y de diseño, aplicados en la solución de problemas arquitectónicos en forma de ejercicios. De esta manera se adquieren los conocimientos, habilidades y actitudes necesarias en cada nivel de formación.

A lo largo de la licenciatura los Alumnos pueden cursar distintos talleres, pero solo un taller a la vez en cada etapa del mapa curricular.

**Tcl:** Lenguaje de herramientas de comando, es un potente lenguaje dinámico de programación apropiado para una gran variedad de usos, incluyendo aplicaciones web y de

escritorio, networking, administración y testing entre otros. Perteneciente a la corriente open, source Tcl es un lenguaje multiplataforma, fácil de utilizar y extensible.

**UNIX:** Es un sistema operativo portable, multitarea y gestiona simultáneamente múltiples usuarios.

**URL:** Localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.

**W3C:** El Consorcio World Wide Web es un consorcio internacional donde las organizaciones miembro, personal y el público en general, trabajan conjuntamente para desarrollar estándares Web. La misión del W3C es guiar la Web hacia su máximo potencial a través del desarrollo de protocolos y pautas que aseguren el crecimiento futuro de la Web.

**XML:** Es un Lenguaje de Etiquetado Extensible, estricto que juega un papel fundamental en el intercambio de una gran variedad de datos. Las tecnologías XML son un conjunto de módulos que ofrecen servicios útiles a las demandas más frecuentes por parte de los usuarios. XML sirve para estructurar, almacenar e intercambiar información. Es un formato que permite la lectura de datos a través de diferentes aplicaciones.

# Referencia

Aplique sql, Groff, James R., Weinberg Paul N., Osborne McGraw-Hill, 1991.

Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process, Larman Craig, Second Edition, Prentice Hall, Pub, 2001.

Beginning PHP and PostgreSQL e-commerce : from novice to professional, Apress, Cristian Darie, Emilian Balanescu, Mihai Bucica. c2006

Diseño de páginas Web interactivas con JavaScript y CSS, Oros Juan Carlos, Alfaomega Ra-Ma, 2004.

Fundamentos de bases de datos, Silberschatz Abraham, F. Korth Henry, Sudarshan S., McGraw-Hill Interamericana, 5a ed., 2006.

Hackers de sitios Web, Scambray Joel, Shema Mike, McGraw-Hill Interamericana, 2003

HTML4.1, Galeano Gil Germán, Sánchez Alonso José Carlos, Díaz Márquez Pablo, ANAYA Multimedia, 2005.

Navegar en Internet Diseño de páginas Web interactivas con Javascript y CSS. 3a. edición ampliada y actualizada a la versión JavaScript 1.5, Orós Cabello Juan Carlos, editorial alfaomega.

Object-Oriented Design with UML and Java, Barclay K. & Savage J., editorial Elsevier Butterworth Heinemann, 2004.

PHP and PostgreSQL: Advanced Web Programming, Geschwinde Ewald, Hans-Jürgen Schöning, Editorial Sams.

PHP y MySQL Tecnologías para el desarrollo de aplicaciones WEB, Cobo Ángel, Gómez Patricia, Pérez Daniel, Rocha Rocío, Díaz de Santos, 2005.

SQL: the complete reference, Groff, James R., Weinberg Paul N., Osborne McGraw-Hill, 1999.

UML by example, Ghinwa Jalloul, Cambridge University, 2004

<http://proquest.safaribooksonline.com/0130925691>, 2007, Lenguaje Unificado de Modelado.

<http://www.iesluisbraille.es/archivos/departamentos/informatica/SGBD/Tema%2002.doc>, 2007, Teoría de los Sistemas Gestores de Bases de Datos

<http://www.php.net/manual/en/>, 2008, Documentación y referencia en línea del lenguaje PHP

<http://www.postgresql.org/docs/8.0/static/index.html>, 2008, Manejador de Bases de Datos POSTGRESQL

<http://proquest.safaribooksonline.com/0672323826>, 2009, Teoría del lenguaje PHP y PostgreSQL

<http://www.arq.unam.mx>, 2007, Información general de la Facultad de Arquitectura

<http://html.conclase.net/w3c/html401-es/intro/intro.html>, 2008, Estándares del W3C.

<http://www.bidi.unam.mx>, 2007, Biblioteca en línea

[http://sisbib.unmsm.edu.pe/bibVirtual/Publicaciones/indata/v04\\_n1/lenguajes.htm#CONCEPTOS](http://sisbib.unmsm.edu.pe/bibVirtual/Publicaciones/indata/v04_n1/lenguajes.htm#CONCEPTOS), 2009, Teoría de lenguajes de programación: conceptos y paradigmas

<http://www.w3.org/MarkUp/>, 2009, Especificaciones del W3C.

<http://site.unam.mx/servidores/docs/bd/ANSI%20SQL.pdf>, 2009, Teoría de servidores.

<http://cygwin.com/>, 2009, Instalación de un entorno Linux bajo Windows.

[http://www.linux-es.org/sobre\\_linux](http://www.linux-es.org/sobre_linux), 2009, Sistema operativo Linux.

<http://www.python.org/>, 2009, Lenguaje Python.

<http://www.adobe.com/es/products/acrobat/adobepdf.html>, 2009, Documentos PDF (Formato de Documento Portátil).

<http://www.opensource.org/>, 2009, Corriente open source.

<http://www.tcl.tk/>, 2009, Teoría del lenguaje Tcl.

<http://office.microsoft.com/es-es/access/HA101656303082.aspx>, 2009, Microsoft Access.

<http://www.java.com/es/about/>, 2009, Lenguaje java.

<http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>, 2009, Teoría de Hojas de Estilo CSS.

<http://www.alegsa.com.ar/>, 2009, Diccionario Informático.

<http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/SOF.htm>, 2009, Teoría de los Sistemas Operativos.

[www.dsic.upv.es/users/ia/lstarin/t1\\_SGBD.doc](http://www.dsic.upv.es/users/ia/lstarin/t1_SGBD.doc), 2009, Teoría de Base de Datos