



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
INGENIERÍA DE SISTEMAS – PLANEACIÓN

ADMINISTRACIÓN DE PROYECTOS:
UNA APLICACIÓN WEB PARA LA COORDINACIÓN Y CONTROL

TESIS
QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN INGENIERÍA

PRESENTA:
DAFNE CITLALLI ABAD MARTÍNEZ

TUTOR PRINCIPAL
M.I. ARTURO FUENTES ZENÓN, FACULTAD DE INGENIERÍA

MÉXOCO, D. F. NOVIEMBRE DE 2015

JURADO ASIGNADO:

Presidente: Dr. Gabriel Sánchez Guerrero

Secretario: Dr. Ricardo Aceves García

Vocal: M. I. Arturo Fuentes Zenón

1^{er}. Suplente: Dr. Mariano Antonio García Martínez

2^{do}. Suplente: M. I. José Antonio Rivera Colmenero

Lugar o lugares donde se realizó la tesis: México, Distrito Federal

TUTOR DE TESIS:

M. I. Arturo Fuentes Zenón

FIRMA

AGRADECIMIENTOS

A mi familia: Arturo, Alejandra y Tonatiuh. Por todo el apoyo y la confianza que día a día depositan en mí. Por el esfuerzo que nos representó este trabajo. Gracias por siempre estar, los amo con todo mi corazón.

Al Mtro. Arturo Fuentes toda mi gratitud por el apoyo y la dirección para realizar esta tesis.

Al Ing. Ángel Sánchez, por el apoyo técnico pero principalmente por su amistad.

A la Mtra. Abigail Serralde y al Mat. Alejandro Ramírez Nieto, gracias por impulsarme y concederme el tiempo para estudiar y realizar este trabajo.

A mis profesores, que con su paciencia y dedicación supieron compartir sus conocimientos y experiencia para permitirme crecer tanto profesional como personalmente.

A todos mis amigos, del posgrado y del trabajo, sin su apoyo en cada una de las etapas de este camino no hubiera sido posible llegar a la meta.

ÍNDICE

Glosario.....	1
1 Introducción	5
2 Conceptos y técnicas básicas de la administración de proyectos.....	9
2.1 Conceptos.....	9
2.1.1 Proyecto.....	9
2.1.2 Ciclo de vida de un proyecto	9
2.1.3 Tipos de proyectos.....	11
2.1.4 Administración de proyectos.....	12
2.2 Técnicas para la administración de proyectos.....	13
2.2.1 Estructura de descomposición del trabajo	13
2.2.2 Diagramas de red.....	16
2.2.3 Diagrama de Gantt.....	18
2.2.4 Ruta crítica	20
2.2.5 Kanban dirigido a la administración de proyectos	22
3 TIC y gestión de proyectos	25
3.1.1 Tecnologías de la Información y la Comunicación, TIC	25
3.1.2 Almacenamiento y procesamiento remoto, nube	26
3.1.3 TIC tradicionales para la gestión de proyectos.....	29
4 Aplicaciones Kanban para la gestión de proyectos.....	31
4.1 Una mirada al mercado de las aplicaciones.....	31
4.2 Análisis comparativo de las aplicaciones seleccionadas.....	36
4.3 LeanKit.....	38
4.3.1 Características.....	38
4.3.2 Ventajas, desventajas y potencialidades	45
5 Desarrollo de un puente entre <i>Ms Project</i> y <i>LeanKit</i>	49
5.1 Descripción del funcionamiento	50
5.2 Uso del programa.....	51
5.3 Condiciones y limitaciones del programa	54
6 Caso de estudio	55
6.1 Breve descripción del IIBI.....	55
6.1.1 Departamento de cómputo	55
6.2 Descripción del proyecto	56
6.2.1 ¿Qué es el SiGeDAA?	56
6.2.2 Ventajas del proyecto	56
6.2.3 Beneficiarios	56

6.3	Planeación y seguimiento.....	57
6.4	Impresiones sobre el uso <i>LeanKit</i>	67
6.4.1	Situaciones para el Aprovechamiento del software	67
6.4.2	<i>LeanKit</i> en el desarrollo de sistemas.....	68
6.4.3	Mejoras obtenidas	68
6.4.4	Lecciones aprendidas	69
	Conclusiones	71
	Anexos	73
A.	Estadísticas y reportes de <i>LeanKit</i>	73
B.	Diccionario de Clases del bridge ProyectoToLeankit	75
C.	Estructura de desglose de trabajo para el SIGEDAA	78
	Bibliografía	79

GLOSARIO

Actividad. Una fracción de trabajo del proyecto que comprende varias tareas, cada una de las cuales pueden llevarse a cabo por personas distintas.

API. La interfaz de programación de aplicaciones o API (del inglés *Application Programming Interface*) es el conjunto de subrutinas, funciones y procedimientos que ofrece cierto software para ser utilizado por otro.

Aplicación informática. Es un tipo de programa informático diseñado para permitir a un usuario realizar diversos tipos de trabajos; a diferencia de los sistemas operativos (que hacen funcionar la computadora), los utilitarios (que realizan tareas de mantenimiento), y los lenguajes de programación (para crear programas informáticos). Algunos ejemplos de programas de aplicación son los procesadores de textos, hojas de cálculo y base de datos.

Aplicación móvil. Aplicación informática diseñada para ser ejecutada en teléfonos inteligentes, tabletas y otros dispositivos móviles. En general se encuentran disponibles a través de plataformas de distribución, operadas por las compañías propietarias de los sistemas operativos móviles como *Android* e *iOS*, entre otros.

Aplicación o herramienta web. Se denomina así a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor *web* a través de Internet o de una intranet mediante un navegador.

Blog. Un blog (del inglés *web log*) o bitácora web es un sitio web que incluye, a modo de diario personal de su autor o autores, contenidos de su interés, actualizados con frecuencia y a menudo comentados por los lectores.

Bug tracker. Aplicación informática diseñada para el seguimiento de incidentes, su función es ayudar a asegurar la calidad del software y asistir a los desarrolladores y a los usuarios de los sistemas informáticos en el seguimiento de defectos del software.

Clases. En informática, modelo que representa objetos o conceptos donde se define un conjunto de estados y comportamientos asociados a dicho objeto.

FAQ. Lista de preguntas y respuestas (del inglés *Frequently Asked Questions*) que surgen frecuentemente dentro de un determinado contexto y para un tema en particular.

GITHUB. Plataforma de desarrollo colaborativo para alojar proyectos utilizando el sistema de control de versiones *Gi* donde el código se almacena de forma pública en su mayoría.

Hardware. Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.

IDE. Entorno de desarrollo integrado (del inglés *Integrated development environment*) es una aplicación que facilita a los programadores de computadoras el desarrollo de software. Normalmente, un IDE consiste de un editor de código fuente, herramientas de construcción automáticas y un depurador.



Informática. Conjunto de conocimientos científicos y técnicas que hacen posible el tratamiento automático de la información por medio de computadoras. También llamada computación, es una ciencia que estudia métodos, procesos, técnicas, con el fin de almacenar, procesar y transmitir información y datos en formato digital.

Interfaz. En informática, se utiliza para nombrar a la conexión física y funcional entre dos sistemas o dispositivos de cualquier tipo dando una comunicación entre distintos niveles.

Java. Es un lenguaje de programación orientado a objetos diseñado para tener la mínima dependencia de implementación; permite que los desarrolladores escriban el programa una vez y lo ejecuten en cualquier dispositivo.

Nativo digital. Persona que nació entre las décadas de los años 1980 y 1990, cuando ya existía una tecnología digital desarrollada al alcance de muchos. Por otra parte, el término *inmigrante digital* se refiere los nacidos entre 1940 y 1980 y se considera que han sido espectadores del proceso de cambio tecnológico.

.NET. Es un marco de desarrollo de *Microsoft* que hace énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones. Dentro de este mercado, es la respuesta a la competencia de Java y PHP.

NUBE. También conocida como computación, servicios, informática en la nube o nube de cómputo, es un paradigma que permite ofrecer servicios de computación a través de una red, usualmente Internet.

PMI. Instituto de Administración de Proyectos (del inglés *Project Management Institute*) es una organización internacional sin fines de lucro que asocia a profesionales relacionados con la Gestión de Proyectos.

Portafolio. En administración de proyectos, un portafolio se refiere a un conjunto de proyectos o programas, no necesariamente directamente relacionados, que se agrupan para facilitar su dirección eficaz para cumplir con los objetivos estratégicos del negocio.

RSS. Siglas de *Really Simple Syndication*, es un formato para difundir información actualizada frecuentemente a usuarios suscritos a la fuente de contenidos.

SCRUM. Es el nombre con el que se denomina a los marcos de desarrollo ágiles caracterizados por adoptar estrategias de programación incremental; basa la calidad de los resultados en el conocimiento tácito de equipos auto organizados.

Sistema de Información. En el contexto de dirección de proyectos, es un sistema compuesto por herramientas y técnicas utilizadas para recopilar, integrar y difundir las salidas de los procesos resultados de la dirección de proyectos. Se utiliza para respaldar todos los aspectos del proyecto desde el inicio hasta el cierre, y puede incluir tanto sistemas manuales como automatizados.

Sistemático. Adjetivo que describe al objeto que sigue o se ajusta a un sistema o conjunto de elementos ordenados. Que actúa con constancia y bajo un método determinado.

Software. Conjunto de programas y rutinas que permiten a una computadora realizar determinadas tareas.

Software libre. Conjunto de software que por elección manifiesta de su autor, puede ser copiado, estudiado, modificado, utilizado libremente con cualquier fin y redistribuido con o sin cambios o mejoras.

Software propietario. Concepto que se aplica a cualquier programa informático que no es libre o que lo es parcialmente ya sea porque su uso, redistribución o modificación están prohibida, o porque requiere permiso expreso del titular del software.

Skype. Software que permite comunicaciones de texto, voz y vídeo sobre Internet.

Tarea. Trabajo relativamente pequeño realizado por una persona.

Telemática. También conocida como teleinformática es una disciplina originada por la convergencia entre las tecnologías de las telecomunicaciones y las de la informática. Entre sus aplicaciones se encuentran las comunicaciones a través de internet, el uso de las mensajerías instantáneas y los sistemas GPS.

Telecomunicaciones. Disciplina que estudia, diseña, desarrolla y explota aquellos sistemas que permiten la comunicación.

TFS. *Visual Studio Team Foundation Server* (TFS) es la plataforma de colaboración de la solución de administración del ciclo de vida de las aplicaciones de *Microsoft*. Admite prácticas ágiles de desarrollo, varios IDE de manera local o en la nube.

Usuario final. En informática, designa a la persona que manipulará de manera directa un producto de software, sin ser necesariamente el cliente o comprador. El concepto clave es la interacción directa con el programa, no la propiedad.

Valor agregado. En administración de proyectos, refiere a la característica de las actividades que producen una transformación física y/o química del producto, por las cuales el cliente está dispuesto a pagar.

Webinar. Término para denominar al servicio de reuniones en línea o conferencias *web* donde se permite a los asistentes interactuar entre sí.

Wiki. Nombre que recibe un sitio *web* cuyas páginas pueden ser editadas directamente desde el navegador, donde los usuarios crean, modifican o eliminan contenidos que, generalmente comparten.

ZAPIER. Servicio de conexión y automatización de tareas entre aplicaciones *web*.

1 INTRODUCCIÓN

La administración de proyectos es una disciplina ampliamente estudiada y ejecutada que se encarga de planear, organizar, dirigir y controlar actividades encaminadas a lograr un objetivo específico optimizando el uso de sus recursos disponibles.

Con el propósito de lograr resultados cada vez más efectivos, se manifiesta la necesidad de adaptar y crear nuevas técnicas utilizadas en esta materia; parte de esta evolución ha llevado a los administradores a integrar el uso tecnologías de información a su profesión.

La oferta de programas de cómputo para la gestión de proyectos crece con respecto a las condiciones planteadas por un mundo que exige herramientas con cualidades como mayor exactitud y tiempos más rápidos de respuesta, por nombrar algunas.

El presente trabajo nace del interés de incorporar alguna de estas nuevas tecnologías de información en beneficio de la gestión de proyectos.

Con la intención de sumar un instrumento que enriquezca el conjunto de herramientas básicas de la administración de proyectos y que además explote las tendencias de colaboración y del procesamiento en la nube, se indagó en el vasto universo de las TIC hallando la aplicación *LeanKit*, una implementación *web* de un tablero *Kanban* orientado a la planeación y seguimiento de los proyectos.

Para comprobar la conveniencia de su uso e identificar sus ventajas, se incorporó como instrumento de monitoreo del proyecto *Sistema de Gestión de Documentos Académicos y Administrativos, SIGEDAA*, en el Instituto de Investigaciones Bibliotecológicas y de la Información de la UNAM.

De ello, sobresalen las mejoras relacionadas en su seguimiento pues permite supervisar y controlar puntualmente el proyecto mediante la visualización del flujo de trabajo, observar los cuellos de botella y tomar acciones sobre las tareas que lo requieran, por lo que auxilia a la dirección del proyecto para equilibrar el trabajo asignado.

LeanKit brinda información actualizada pues refleja de manera inmediata los cambios hechos por algún colaborador, mantiene abiertos los canales de comunicación al interior del equipo de trabajo, reduciendo el riesgo de malos entendidos u olvidos.

Otros puntos a favor son su portabilidad entre sistemas operativos y dispositivos móviles; y que puede ajustarse a la forma de trabajo del director del proyecto.

El enfoque atendido para la realización de esta tesis fue partir de un conocimiento previo sobre temas de dirección de proyectos así como la revisión de fuentes bibliográficas en el campo, se indagó entre las TIC alguna que pudiera apoyar la administración, resultado de esta búsqueda se encontraron varios desarrollos, entre las que destaca la mencionada *LeanKit*.

6

Con el desarrollo de este documento se encontró una utilidad mayor en el seguimiento que para la planeación, asimismo propicia la interacción de los involucrados mediante el acceso a la nube, por lo que se aprecia benéfica su incorporación. Todo ello sin dejar de considerar que la investigación fue delimitada por una búsqueda parcial (no exhaustiva) dada la gran oferta de software por lo que se presupone que es de las mejores herramientas que actualmente existe en el mercado.

Ya que este trabajo ambiciona sumar el uso de una TIC basada en la nube, encuentra originalidad intrínseca en los temas que aborda.

Las implicaciones prácticas de la adición de *LeanKit* en la planeación y seguimiento del *SIGEDAA* contemplan la falta de características puntuales como tableros de indicadores, asignación de costos y gestión de riesgos, sin embargo, no hay que perder de vista que ésta es una herramienta complementaria al uso de otras tantas.

Además, se hace presente el factor humano, por lo que es importante cuidar la forma de implementación de ésta y cualquier otra herramienta, considerando la dinámica actual de trabajo, la resistencia al cambio y el manejo de nuevas tecnologías.

Finalmente, podemos mencionar que independientemente de las características de una herramienta, un sistema que no se utiliza y no se actualiza con información, puede terminar siendo obsoleto e incluso volverse una carga más para el administrador y para el grupo de trabajo.

Como guía para la lectura y contenido se ofrece al lector la estructura que sigue este documento:



Figura 1. Estructura del trabajo de tesis.

2 CONCEPTOS Y TÉCNICAS BÁSICAS DE LA ADMINISTRACIÓN DE PROYECTOS

2.1 CONCEPTOS

2.1.1 PROYECTO

El *Project Management Institute*, PMI, define un proyecto como un conjunto de actividades temporales orientadas a producir un producto, servicio o resultado único con recursos limitados y alcances definidos¹.



Figura 2. Descripción de proyecto.

Por lo tanto cada proyecto genera un resultado único, ya sea tangible o intangible y, si bien en el desarrollo de las actividades para lograr el objetivo pueden existir elementos repetitivos en entregables o actividades, esta repetición no cambia la característica de unicidad del resultado.

2.1.2 CICLO DE VIDA DE UN PROYECTO

Se define como ciclo de vida de un proyecto al conjunto de etapas que lo conforman, desde su inicio hasta su cierre. Estas etapas generalmente son secuenciales y algunas veces se traslapan; se determinan con base en la naturaleza, tamaño o área de aplicación del proyecto, así como en las necesidades del administrador y la organización que lo ejecuta.

Estas fases se acotan determinando un tiempo de ejecución o un punto de control y para precisarlas se pueden dividir por objetivos, resultados intermedios o disponibilidad financiera. Así, su ciclo de vida proporciona el contexto básico de administración del proyecto, sin importar el trabajo específico que se esté desarrollando.

¹ A Guide to the Project Management Body of Knowledge (5th Ed.). Project Management Institute. P. 3.

De acuerdo con el PMI, el ciclo de vida de cualquier proyecto, sin importar su naturaleza y complejidad, se puede configurar en una estructura general con las siguientes fases²:

- Inicio
- Organización y preparación
- Ejecución del trabajo
- Cierre del proyecto

10

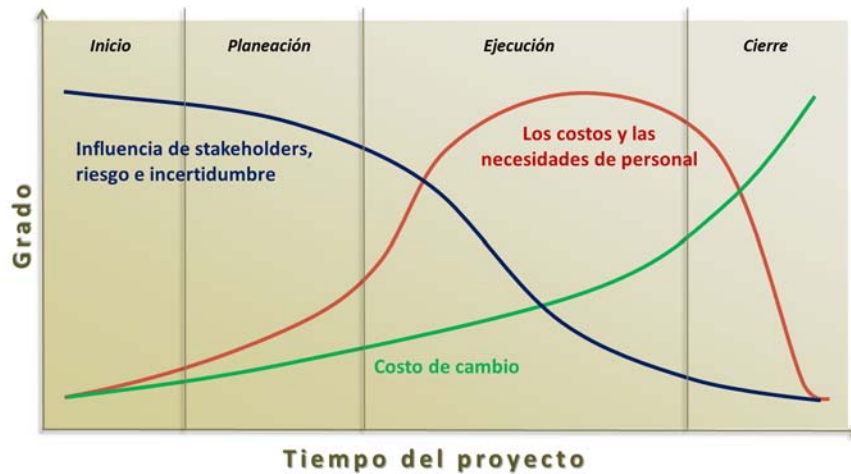


Figura 3. Cambios en costos y riesgos a través del ciclo de vida del proyecto. PMBoK.

A lo largo del proyecto las variables involucradas como costos, alcances o riesgos cambian el grado en el que lo afectan. Podemos dar un vistazo a la figura 3 para darnos una idea de cuánto impactan estas variables y llegar a las siguientes conclusiones:

- Los costos y las necesidades de personal son bajos al inicio del proyecto, se incrementan cuando se ejecuta y bajan en su cierre.
- La incertidumbre y los riesgos son mayores al inicio del proyecto, conforme se avanza en él decrecen.
- El costo por hacer un cambio en alguna de las dimensiones del proyecto (costo, tiempo, alcance y calidad) se hace mayor conforme se acerca su final.

² Ibídem. p. 38.

2.1.3 TIPOS DE PROYECTOS

Una vez entendido el concepto de proyecto es claro que su implementación es tan variada como las necesidades de las organizaciones que lo ejecuta, por lo que, para clasificarlos se pueden tomar un sinnúmero de características, por ejemplo su complejidad, disciplinas involucradas, objetivos, tiempo de ejecución, con base en el origen de sus recursos, su impacto, etcétera.

11

Para clasificarlos en este documento, se consideran algunas características de la naturaleza de la planeación³ y la tipificación de los proyectos de Kerzner⁴; resultado de lo anterior se obtiene la siguiente categorización:

- Proyectos pequeños
 - o De duración entre tres meses hasta un año.
 - o Comunicación continua entre todos los miembros del equipo.
 - o Administración manual en mayor medida que computarizada.
 - o La estructura de desglose de trabajo no supera los tres niveles.
 - o Una organización es capaz de manejar todas las actividades para desarrollar el proyecto.
- Mega proyectos
 - o Procedimientos claramente definidos, comunicación efectiva en todos los niveles y formación en administración de proyectos son factores críticos para su éxito.
 - o Involucran a varios contratistas y proveedores.
 - o Suelen enfrentarse a la falta de recursos específicos (materiales y humanos).
 - o Son requeridos grandes cantidades de personas, en general por cortos periodos.
 - o La estructura organizacional varía de acuerdo con la naturaleza del proyecto.

³ Ackoff, Russell L. (1980). Un concepto de planeación de empresas. México: Limusa.

⁴ Kerzner, Harold. (2009). Project Management: A systems approach to planning, Scheduling, and Controlling. EUA: Wiley. pp. 336, 338-339.

2.1.4 ADMINISTRACIÓN DE PROYECTOS

Se puede definir como la aplicación de conocimientos, habilidades, técnicas y herramientas aplicadas en las actividades de un proyecto para satisfacer los requerimientos establecidos. Con base en lo anterior, dirigir un proyecto implica, entre otras cosas, identificar requisitos, planificar y ejecutar el proyecto, gestionar la comunicación y equilibrar sus restricciones (alcance, calidad, tiempo, presupuestos, riesgos, etcétera).

La administración de proyectos ofrece pautas para la organización, ayuda a aprovechar de la mejor forma los recursos, brinda alternativas de control y seguimiento; además se apoya del conocimiento previo para producir estimaciones más cercanas.

Se debe entender que la dirección es una actividad iterativa y progresiva que se mejora de forma continua ya que, conforme se avanza en el proyecto, se cuenta con información más detallada y estimaciones más precisas. Para esta gran tarea, la quinta edición de la *Guía de fundamentos para la dirección de proyectos* del PMI, *PMBok* por sus siglas en inglés, contiene 47 procesos categorizados en 5 grupos de procesos: Inicio, planificación, ejecución, monitoreo y control, y cierre⁵.



Figura 4. Grupos de procesos según el PMI.

⁵ (2013). A Guide to the Project Management Body of Knowledge (5th Ed.). Project Management Institute. p. 5.

2.2 TÉCNICAS PARA LA ADMINISTRACIÓN DE PROYECTOS

A lo largo de los años, los profesionales de la administración de proyectos han incluido herramientas para ayudarles en su labor. Las siguientes son una revisión rápida a algunas de estas herramientas, si el lector desea conocer más acerca de estos temas se le invita a consultar la bibliografía referida.

13

2.2.1 ESTRUCTURA DE DESCOMPOSICIÓN DEL TRABAJO

DESCRIPCIÓN

El *Work Breakdown Structure*, *WBS*, o la estructura de descomposición del trabajo, EDT, es el desglose jerárquico de un proyecto en tareas más pequeñas y fáciles de manejar, permitiendo organizar y tener una visión estructurada del alcance del proyecto.

Gráficamente, esta descomposición suele tener la estructura de un organigrama tal como se muestra en la figura 5; en la parte más alta se representa el proyecto en sí, en el segundo nivel se representan categorías o fases y la planificación de las actividades se exponen en los niveles más bajos. A estas últimas unidades se les conoce como paquetes de trabajo.

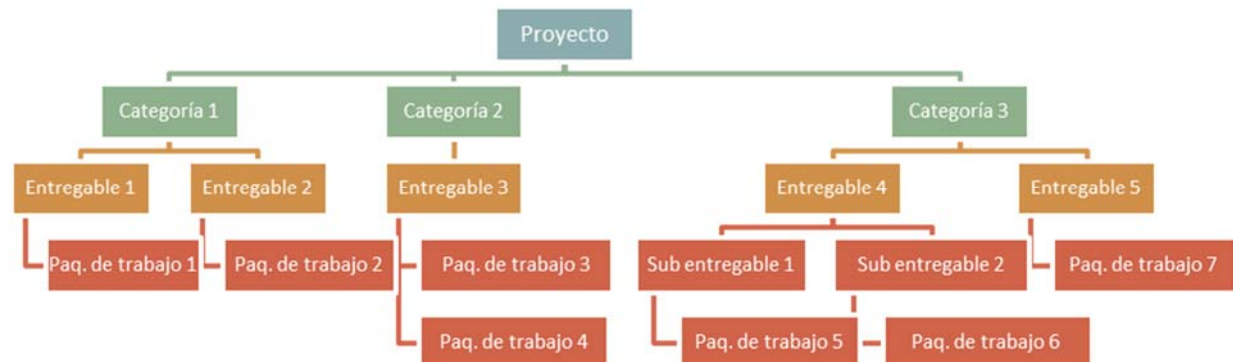


Figura 5. Representación general de un WBS.

Al final de la descomposición quedan los paquetes de trabajo que son unidades individualmente más claras que el proyecto en sí. De esta forma, el total del trabajo situado en los niveles inferiores corresponderá al acumulado para los niveles superiores, asegurando que no se omite o se realicen tareas de más.

No existe una forma única ni método preciso para hacer un WBS y su nivel de detalle corresponderá a las necesidades del administrador del proyecto. En la práctica generalmente se realiza bajo diversos enfoques y se deben utilizar los objetivos del proyecto y sus requerimientos, así como la experiencia obtenida en el desarrollo de WBS anteriores.

PROCEDIMIENTO

1. Identificar los entregables del proyecto, es decir lo considerado dentro de los alcances del proyecto.
2. Elegir el grado y la característica por la cual se organizará el WBS, por ejemplo, las fases del ciclo del proyecto o los entregables principales.
3. Identificar las actividades que se deben ejecutar para lograr los objetivos particulares.
4. El procedimiento concluye cuando cada actividad se divide en tareas claramente definidas y que, generalmente, implican entre ellas el mismo esfuerzo para ser realizadas.

Importante:

- La EDT o WBS no muestra dependencias entre actividades.
- No tiene ni está basada en una escala de tiempo.
- La subdivisión de tareas termina cuando se requiere una cantidad de tiempo igual a la unidad más pequeña programada para realizar un paquete de trabajo.
- Cuanto más complejo es el proyecto, más niveles se requerirán.

EJEMPLO⁶

ABC Auto Company piensa introducir un nuevo coche de pasajeros. Se trata de un proyecto muy grande. En el nivel más alto, los empleados se enfrentan a cuatro tareas:

1. Determinar los requisitos de los clientes.
2. Diseñar un vehículo que cumpla con estos requisitos.
3. Construir el vehículo.
4. Hacer pruebas al vehículo.

15

Como indica la figura 6, cada una de esas tareas de máximo nivel se puede desglosar en un conjunto de subtarear. Y cada una de las subtarear que aparecen en la figura puede, a su vez, desglosarse aún más. Por ejemplo, esquemas del motor, bajo la tarear de diseño del vehículo, se puede descomponer en docenas de subtarear, como el diseño del sistema de transmisión o el diseño del equipo de refrigeración.



Figura 6. WBS para el ejemplo del fabricante de autos.

⁶ (2004). Gestión de proyectos. Serie Harvard Business Essentials. Ed. Deusto. p.76.

2.2.2 DIAGRAMAS DE RED

DESCRIPCIÓN

Un diagrama de red, figura 7, es la representación gráfica de las relaciones lógicas que hay entre los paquetes de trabajo que integran un proyecto.

Se forma mediante círculos o nodos que representan actividades y flechas que ejemplifican la secuencia que guardan, esto implica que no se puede iniciar con un nuevo paquete mientras no se finalicen todas las actividades del paquete que le antecede.

Los nodos pueden representar tanto paquetes de trabajo como actividades generales y este diagrama no muestra una escala de tiempo, por lo que se debe entender a cada nodo como el instante en el que inicia o termina la actividad que simboliza.

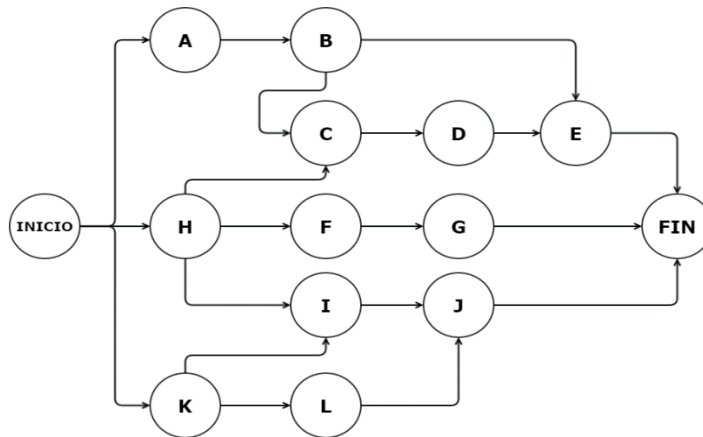


Figura 7. Diagrama de red.

PROCEDIMIENTO

1. Definir las actividades que se representarán gráficamente así como su secuencia. Generalmente este paso se realiza en una tabla con la siguiente configuración:

Actividad	Predecesora inmediata
1.	
2.	

Tabla 1. Configuración de una tabla de actividades secuenciadas.

2. De acuerdo con la tabla de actividades se dibuja el primer nodo correspondiente a la primera actividad, a veces sólo llamada INICIO.
3. Continúe trabajando de izquierda a derecha hasta finalizar con la tabla o lista de actividades, dibujando las líneas que unirán una actividad con sus actividades siguientes inmediatas.

Importante:

- El tiempo de ejecución va de izquierda a derecha. Las flechas que unen a los nodos siempre apuntarán hacia adelante en el tiempo, es decir, hacia la derecha.
- El tamaño de las flechas que unen a los nodos no es proporcional al tiempo que tarda en ejecutarse la actividad.
- Se pueden validar las dependencias lógicas recorriendo el diagrama del final al inicio.
- Este diagrama siempre es continuo, es decir, un nodo siempre tiene una flecha que entra y una que sale, exceptuando el INICIO y el FIN.

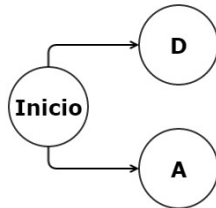
17

EJEMPLO

1. Tabla de actividades secuenciadas.

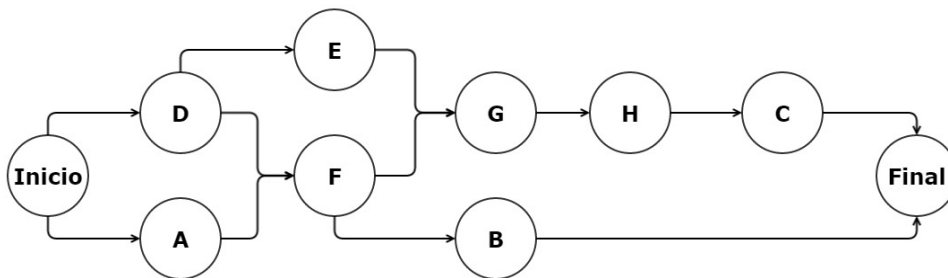
Actividad	Inicio	D	A	F	E	G	B	H	C	Final
Predecesora inmediata	----	Inicio	Inicio	D ; A	D	F ; E	F	G	H	C ; B

2. Comenzar con el primer nodo.



Ya que cada nodo representa una actividad finalizada, este diagrama se puede interpretar como “Inicia el proyecto y después se concluyen las actividades D y A respectivamente”.

3. Dibujar de izquierda a derecha una a una las actividades secuenciadas.



2.2.3 DIAGRAMA DE GANTT

DESCRIPCIÓN

También conocido como diagrama de barras, ayuda a representar gráficamente la duración de cada uno de las actividades de un proyecto y en algunos casos puede mostrar las dependencias que guardan.

18

En él, la lista de actividades se encuentra en el eje vertical; las unidades de tiempo en el eje horizontal superior; y la duración de cada actividad es representada mediante barras cuyo tamaño se establece en función de las fechas de inicio y de término, por lo que, para elaborarlo es necesario conocer la relación entre actividades así como la estimación de su duración. En la figura 8 se puede observar un ejemplo general de un diagrama de barras.

PROCEDIMIENTO

1. Para comenzar el diagrama de barras es necesario contar con la lista secuenciada de las actividades y con dependencias de tareas o fases que se representarán, además de las fechas estimadas de inicio y término, por lo que es recomendable preparar con antelación el diagrama de red.
2. Cada elemento de la lista de tareas se escribe en un renglón de la primera columna del diagrama.
3. De acuerdo con la fecha de inicio de la primera actividad y la fecha de término de la última se crea la línea de tiempo del diagrama, ésta se localiza en el eje horizontal superior.
4. Para cada actividad o fase se dibuja la barra de tiempo en función de su duración y fecha de inicio estimada.

Importante:

- Cuidar que las actividades guarden su secuencia para que el diagrama luzca ordenado y cumpla su propósito. Además, se puede dibujar la dependencia entre actividades con una flecha que enlace el final de una actividad con el principio de otra.
- Se debe utilizar las mismas unidades de tiempo para todas las actividades del proyecto.

EJEMPLO

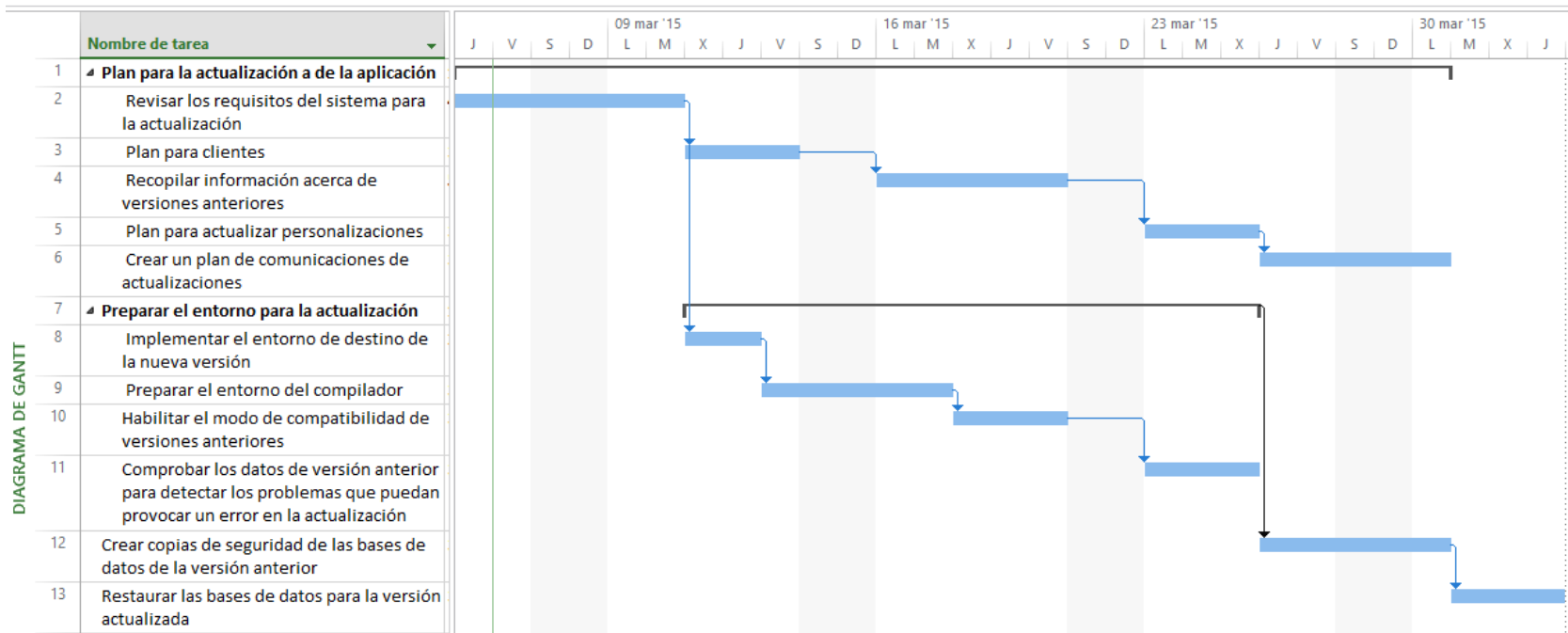


Figura 8. Ejemplo de diagrama de Gantt.

2.2.4 RUTA CRÍTICA

DESCRIPCIÓN

La ruta crítica o CPM por sus siglas en inglés (*critical path method*), es un algoritmo que se utiliza para estimar la duración mínima de un proyecto.

Conociendo la duración y secuencia de cada paquete de trabajo, se calcula la ruta crítica para todas las actividades planeadas bajo un escenario ideal, es decir, sin más restricciones que las calculadas desde el inicio del proyecto.

La ruta crítica es la secuencia de actividades más larga y determina la menor duración posible del proyecto, y en la cual se debe poner especial atención si se desea cumplir estrictamente con lo programado.

Ya que los proyectos pueden llegar a tener un número importante de paquetes de trabajo, la realización de esta tarea se puede simplificar utilizando software especializado.

PROCEDIMIENTO

1. Realizar el diagrama de red del proyecto o considerar la tabla de actividades equivalente.
2. Recorrer la tabla de actividades o el diagrama de red por cada uno de los posibles caminos a seguir entre el inicio y el final *I-F*, sumando las duraciones de cada paquete de trabajo.
3. Hallar la red de actividades más larga y de mayor duración. Esta será ruta crítica y determinará la duración mínima del proyecto y suele resaltarse en el diagrama.

Importante:

- Dentro de las redes de un cronograma puede haber más de una ruta crítica.
- La ruta crítica además se utiliza para determinar la holgura en la planeación de las actividades dentro de un cronograma, es decir, que tanto se puede retrasar o extender una actividad respecto de las fechas programadas sin alterar la fecha final del proyecto.

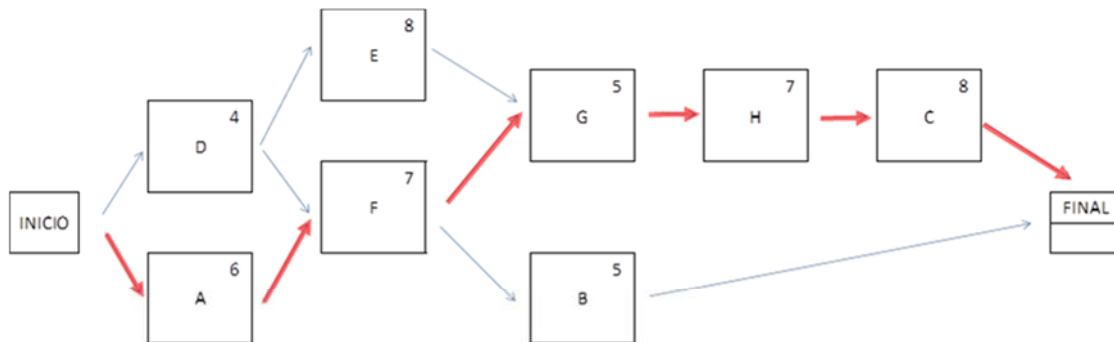
EJEMPLO

1. Considerando la tabla de actividades siguiente:

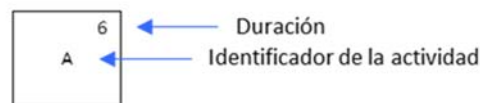
Identificador de la actividad	Predecesora inmediata	Tiempo de ejecución
Inicio	----	0 meses
D	Inicio	4 meses
A	Inicio	6 meses
F	D ; A	7 meses
E	D	8 meses
G	F ; E	5 meses
B	F	5 meses
H	G	7 meses
C	H	8 meses
Final	C ; B	0 meses

21

Se realiza el diagrama de red con la duración de cada actividad.



Donde:



2. Determinación de las diferentes rutas para concluir el proyecto, sumando las duraciones de las actividades.

- | | |
|---|------------------|
| 1. Inicio – D(4) – E(8) – G(5) – H(7) – C (8) – Final | Total = 32 meses |
| 2. Inicio – D(4) – F(7) – G(5) – H(7) – C (8) – Final | Total = 31 meses |
| 3. Inicio – D(4) – F(7) – B(5) – Final | Total = 16 meses |
| 4. Inicio – A(6) – F(7) – G(5) – H(7) – C (8) – Final | Total = 33 meses |
| 5. Inicio – A(6) – F(7) – B(5) – Final | Total = 18 meses |

3. Determinación de la ruta crítica

La ruta crítica es la de mayor duración, es decir el camino 4) con 33 meses de duración.

2.2.5 KANBAN DIRIGIDO A LA ADMINISTRACIÓN DE PROYECTOS

Además de las herramientas clásicas, actualmente se han llevado al campo de la administración de proyectos otras técnicas que nacieron inicialmente para mejorar los procesos de manufactura, específicamente el *kanban*.

Este contenido se brinda al lector como una introducción a la incorporación del *kanban* en la administración de proyectos, misma que se abordará en el capítulo 4.

DESCRIPCIÓN

En japonés *kanban* significa "señal visual" y es un mecanismo de control para el seguimiento de actividades conforme éstas recorren un flujo de trabajo. Pertenece a un cúmulo de técnicas que buscan mejorar los procesos productivos a través del uso de recursos estrictamente necesarios, sin embargo cerca de 2006 comenzó su aplicación en la gestión del trabajo intelectual.

Este sistema es altamente visual y permite una comunicación sencilla para que cada trabajador sepa fácilmente el qué y cuándo de sus tareas asignadas⁷.

El método se basa en limitar el trabajo en curso y comenzar con algo nuevo sólo hasta que el bloque de trabajo anterior haya sido entregado a la siguiente fase en la cadena de producción. Para el seguimiento del flujo de trabajo se usa típicamente un panel o pizarrón dividido en columnas donde se adhieren notas que indican el estado de cada actividad.

PROCEDIMIENTO

1. Dividir, preferentemente, un muro o pizarrón en columnas que correspondan a las etapas consideradas en el desarrollo del trabajo programado, es decir que describan el flujo de trabajo.
2. Limitar el trabajo en curso máximo para cada carril o columna.
3. Desagregar el proyecto en tareas específicas, escribir cada elemento en una tarjeta y colocarlo en la primera columna del pizarrón. Para este paso se puede utilizar un WBS.
4. Asignar a algún recurso (grupo de trabajo o individuo) la realización de cada tarea.
5. Cuando la tarea es completada y el recurso asignado cambie la tarjeta de carril, el siguiente grupo de trabajo puede comenzar con las actividades necesarias para culminar la siguiente fase.

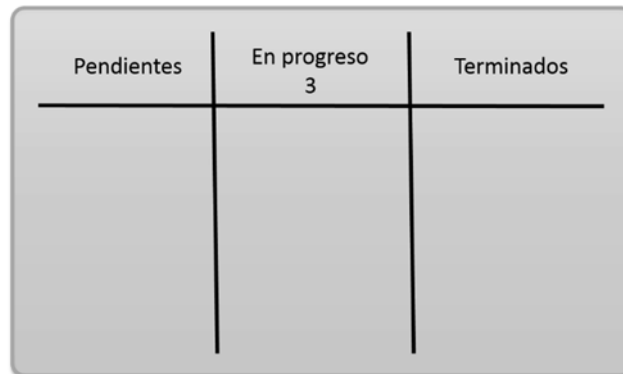
⁷ Se invita al lector interesado en el detalle de este método a consultar la bibliografía referida.

Importante:

- Cuidar que el trabajo en progreso no supere el máximo establecido. Se puede ajustar cada que sea conveniente para el proyecto con el objetivo de mejorar la administración de los recursos.
- Se han identifican cinco prácticas para su correcta implementación:
 1. Hacer visible el flujo de trabajo para comprender cómo avanza.
 2. Limitar el trabajo en curso.
 3. Dirigir y gestionar el flujo de trabajo.
 4. Determinar las reglas y directrices que norman el avance de las tarjetas.
 5. Proponer acciones de mejora basado en los resultados obtenidos.

EJEMPLO

1. Dividir un pizarrón en las columnas para el seguimiento del flujo de trabajo y determinar el límite de trabajo en progreso.

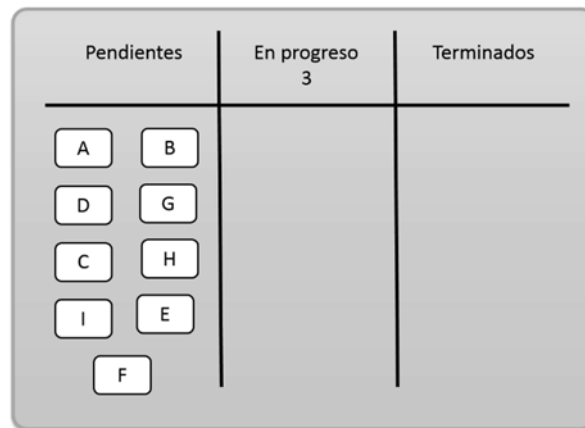


2. Escribir lista de tareas y poner cada una en una tarjeta.

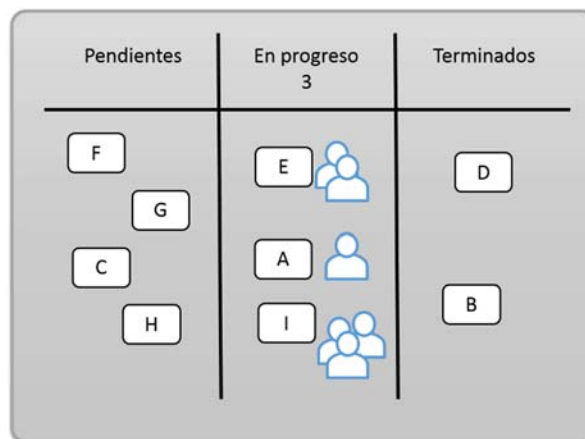
Considerando la siguiente tabla de actividades:

Actividad	Predecesora inmediata	Actividad	Predecesora inmediata
D	Inicio	E	D
A	Inicio	G	E
B	Inicio	H	F
F	D ; A	I	B
E	D	Final	G H I

Al inicio el tablero lucirá de la siguiente forma:



3. Con la asignación de recursos y dando seguimiento al proyecto, en algún momento el tablero se puede ver así:



4. Si es necesario se debe ajustar el límite del trabajo en progreso para mejorar la distribución del trabajo entre los recursos humanos disponibles.

3 TIC Y GESTIÓN DE PROYECTOS

La penetración de la tecnología en casi todos los aspectos de la vida cotidiana es indiscutible. Actualmente estos avances nos proporcionan una caja de herramientas para ayudarnos en nuestras labores sin importar la disciplina en la que nos desenvolvamos, al mismo tiempo que podemos accederla desde virtualmente cualquier lugar (incluso *en el camino*) mediante algún dispositivo electrónico móvil, viéndonos limitados principalmente por nuestra destreza, la conectividad y el acceso a la energía.

Dado que la administración de proyectos es una tarea constante que ajusta lo planeado conforme las variantes de alcance-tiempo-costos-calidad, con proyectos cada vez más grandes y complejos, se hace necesario el uso de tecnologías que permitan tomar decisiones con base en la mejor información disponible en el momento.

Con objeto de complementar el contexto relacionado con el uso de aplicaciones móviles para la administración de proyectos, se proporcionan de manera general los conceptos de tecnologías de información y comunicación TIC, así como los modelos de almacenamiento y procesamiento en la nube.

3.1.1 TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN, TIC

Una definición amplia de las TIC es la concepción de un universo de dos conjuntos, representados por las tradicionales tecnologías de la comunicación y por las tecnologías de la información.

Las tecnologías de información están relacionadas principalmente con el procesamiento, almacenamiento y transmisión de la información, abarca la informática, electrónica y telecomunicaciones. Por otro lado a las tecnologías de la comunicación les conciernen las teorías y aparatos que posibilitan el desarrollo de prácticas comunicativas.

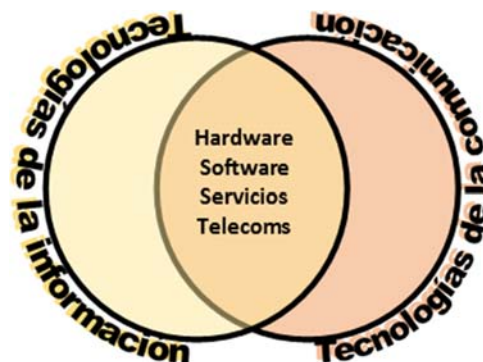


Figura 9. TIC como unión de dos conjuntos.

Así para mencionar ejemplos de tecnologías de información están las computadoras, el software como hojas de cálculo y bases de datos, y los dispositivos de almacenamiento; en complemento, el teléfono, la radio o los satélites son ejemplos de tecnologías de comunicación.

Las Tic, son entonces el resultado de la relación tecnológica entre las telecomunicaciones, las ciencias de la computación y la administración y el manejo de la información. Se consideran como sus componentes el hardware, el software, los servicios y las telecomunicaciones.⁸

De este modo es posible referirse al conjunto de los dispositivos y de los conocimientos que permiten el procesamiento, la transmisión y el almacenamiento de datos y que favorecen que las personas desarrollen comunicaciones

Ya que el término TIC puede ser ambiguo de acuerdo al ambiente y grado de profundidad donde se ocupe, es necesario definirlo en el contexto del presente trabajo. Por lo que definiremos a las TIC son todos aquellos recursos, herramientas y programas que se utilizan para procesar, administrar y compartir información mediante diversas plataformas tecnológicas como computadoras, dispositivos móviles, reproductores portátiles de audio, etc.

3.1.2 ALMACENAMIENTO Y PROCESAMIENTO REMOTO, NUBE⁹

La computación en la nube también conocida como *cloud computing* es un paradigma de servicios informáticos que se ofrecen a través de Internet.

Es un modelo de servicios que permite el acceso ubicuo bajo demanda para acceder a un conjunto de recursos computacionales compartidos (redes, servidores, almacenamiento, aplicaciones y servicios) que pueden ser rápidamente provistos y liberados con el mínimo esfuerzo de administración o interacción con el proveedor de los servicios. Es decir,

Según el Instituto Nacional de Estándares y Tecnología, NIST por sus siglas en inglés, el modelo de nube define cinco características esenciales, tres modelos de servicios y cuatro modelos de implementación¹⁰.

⁸ Rodríguez K. Las TIC como apoyo al proceso de enseñanza-aprendizaje en Bibliotecología. Escuela de Bibliotecología, Documentación e Información. Universidad Nacional de Costa Rica.

⁹ Borges H. (2015). Tecnologías de la información para la administración de proyectos en una empresa consultora. Tesis del Posgrado de Ingeniería. UNAM.

¹⁰ Mell P. Grance T. (2011). The NIST Definition of Cloud Computing. National Institute of Standards and Technology.

Las características esenciales mencionadas son autoservicio bajo demanda, amplio acceso a la red, conjunto público de recursos, flexibilidad y supervisión al servicio.

Por otro lado, los modelos de servicios de la computación en la nube son:

- Software como servicio.

Son aplicaciones dirigidas al usuario final obtenidas directamente del proveedor. Principalmente basado en *web*, por lo que no requiere instalación y su mantenimiento es automático. Los consumidores no tienen el control de la infraestructura. *Ejemplo: Correo electrónico basado en web.*

- Plataforma como servicio.

Son plataformas de desarrollo en internet dirigidas a programadores de software. Aún cuando los consumidores no tienen control de la infraestructura, si lo tienen en las aplicaciones de desarrollo así como la configuración de sus ambientes. *Ejemplo: Google App Engine.*

- Infraestructura como servicio.

El procesamiento, el almacenamiento, las redes y otros recursos, son puestos a disposición del usuario para hacer uso de software, incluyendo sistemas operativos y aplicaciones, sobre los cuales tiene completo control, pero el mantenimiento lo realiza el proveedor.

Ejemplo: Virtualización de servidores.

Finalmente, los cuatro tipos de formas fundamentales de prestación de los servicios en la nube en función del control y gestión de los entornos informáticos.

- Nube Privada

Se puede utilizar cuando los datos de una organización son demasiado sensibles o valiosos, pues es operada exclusivamente para ella. Un ejemplo de infraestructura privada como servicio es la virtualización de servidores que la Facultad de Ingeniería ofrece a sus divisiones a través de la Unidad de Servicios de Cómputo Académico, UNICA, de la Secretaría General.

- Nube comunitaria

Es una nube privada compartida por varias organizaciones y da soporte a una comunidad con requisitos comunes como seguridad, políticas y aspectos de cumplimiento normativo. El servicio de publicación de páginas académicas que la UNAM ofrece es un ejemplo de esta implementación.

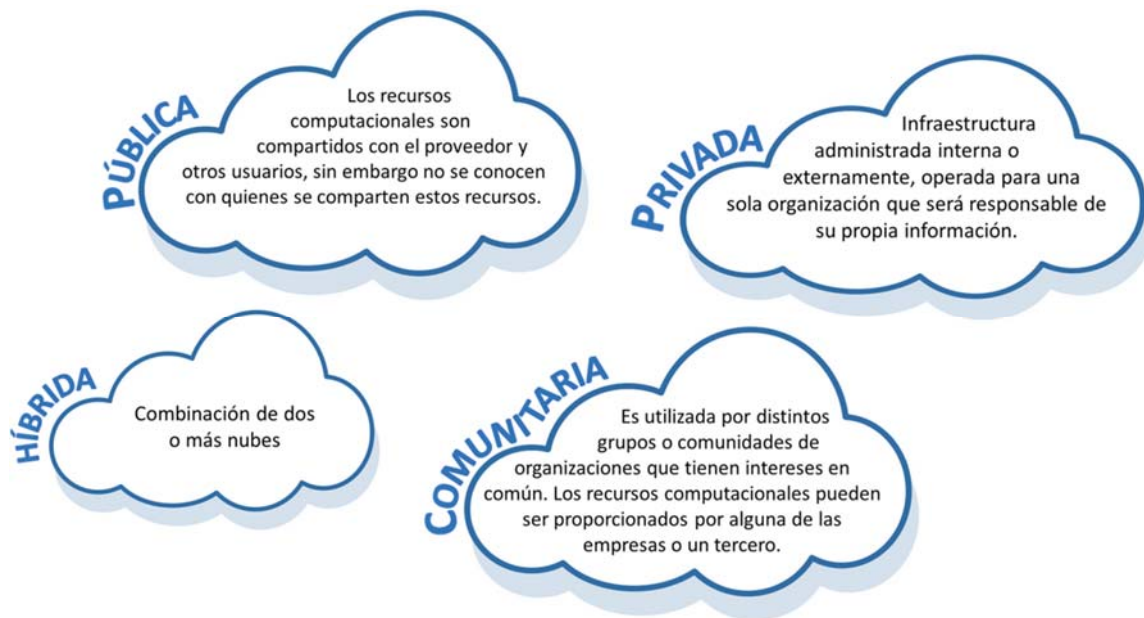


Figura 10. Modelos de implementación de computación en la nube.

- **Nube pública**
Se encuentra a disposición del público en general por lo que puede generar riesgos en términos de seguridad, privacidad o rendimiento. La oferta de servicios de Google (Drive, Gmail, Maps, etc.) es ejemplo de una nube de software de servicios públicos.
- **Nube híbrida**
La infraestructura en la nube se compone de dos o más nubes que están vinculadas por tecnología estandarizada que permite la portabilidad de datos y aplicaciones. La cuenta de correo de comunidad UNAM en conjunto con los servicios que ofrece Office 365 son un buen ejemplo de nube híbrida.

3.1.3 TIC TRADICIONALES PARA LA GESTIÓN DE PROYECTOS

Hoy en día, el mercado ofrece un sinfín de herramientas informáticas que implementan las técnicas clásicas para la administración de proyectos y que podemos encontrarlas en diferentes formatos, como software libre o propietario, aplicaciones *web*, móviles o de escritorio, de almacenamiento en la nube o en nuestros propios recursos.

Cada herramienta tiene su propia complejidad y alcances, algunas permiten ver de forma gráfica el progreso de ejecución, holguras en la programación de tareas, utilización de los recursos y otro tipo de información. Para citar dos ejemplos, en la figura 11 se pueden observar algunas de las características de dos de los programas más robustos y populares para la administración de proyectos, *Microsoft Project* y *Oracle Primavera*.

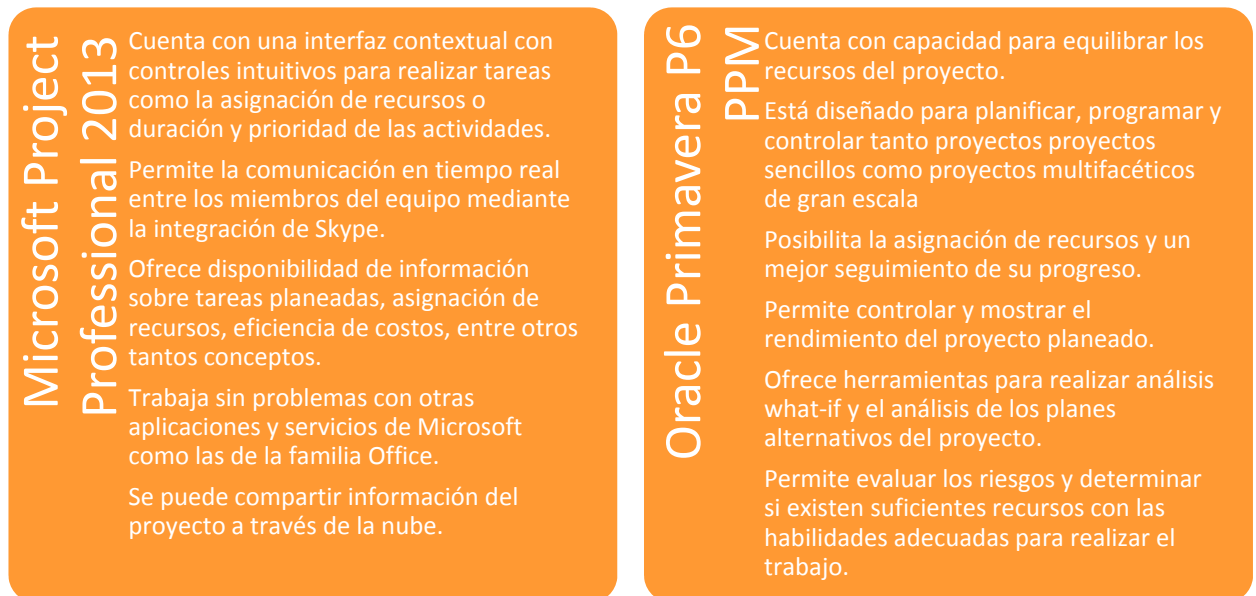


Figura 11. Características de software robusto para la administración de proyectos.

4 APLICACIONES KANBAN PARA LA GESTIÓN DE PROYECTOS

4.1 UNA MIRADA AL MERCADO DE LAS APLICACIONES

La tecnología juega un papel influyente entre los directores ya que pone a su disposición armas que les permite realizar de mejor forma su labor. Por tal razón es común la búsqueda continua de soluciones innovadoras que les facilite estar a la altura de los retos y demandas con los que se enfrentan.

Para dirigir y gestionar el trabajo del proyecto, los administradores deben apoyarse en sistemas que proporcionan acceso a la información como cronogramas, autorización de órdenes de trabajo, indicadores clave de desempeño, etc.

En el mercado existen diversos programas y aplicaciones para apoyar el seguimiento de un proyecto. Conforme a la propuesta de esta tesis, se buscaron herramientas colaborativas que apoyan la gestión de proyectos e implementan una vista del flujo de trabajo además de otras funcionalidades, dando como resultado, de entre muchas, las siguientes aplicaciones:

- Jira
- Teampulse
- Axosoft ontime SCRUM
- LeanKit
- Trello

A continuación se mencionan algunas de sus características:

JIRA

Desarrollada por *Atlassian*, *Jira* es una plataforma integral de gestión de portafolios y proyectos y cuenta con diferentes herramientas complementarias como tableros para controlar los flujos de trabajo y desarrollos bajo la metodología SCRUM.

Escalable desde 10 hasta más de 10 mil usuarios¹¹ y con la posibilidad de alojarse tanto en la nube como en el hardware del usuario, la herramienta para la gestión de proyectos cuenta con flujos de trabajo personalizables, interfaz móvil, gestión de requisitos, generador de reportes e importación de datos a otras soluciones como *GitHub* y *Microsoft Excel*.

¹¹ En su versión Servidor.

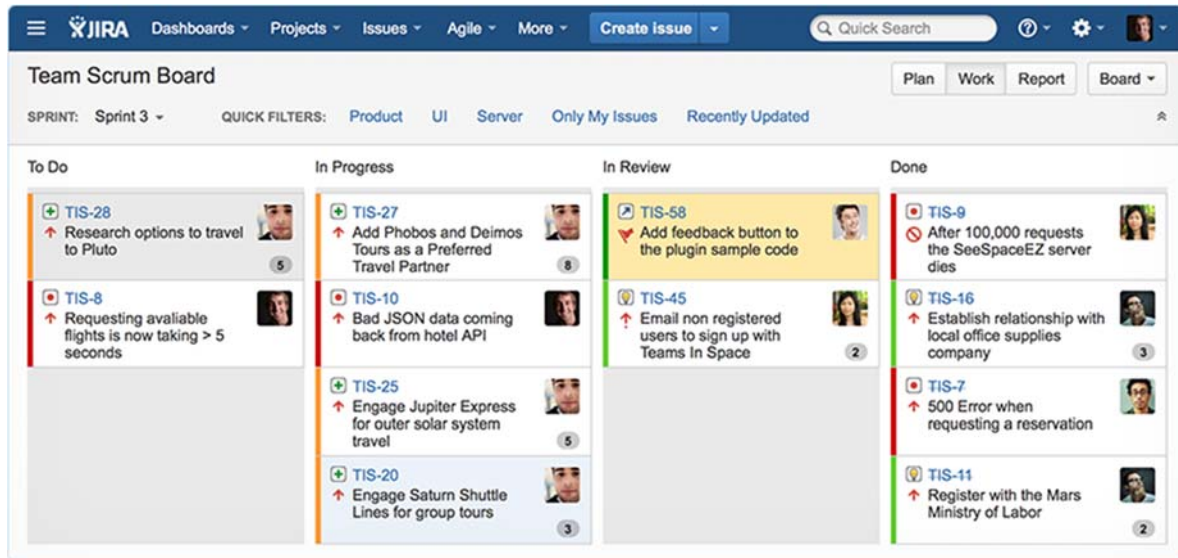


Figura 12. Ejemplo de tablero SCRUM del software Jira.

TEAMPULSE

Desarrollada por *Telerik*, empresa dedicada a construir aplicaciones para soluciones de TI, es una herramienta *web* que ayuda en la gestión de requisitos y recursos de cualquier tipo de proyecto, pues soporta modelos de desarrollo como SCRUM, kanban y otros modelos personalizados.

Cuenta con un tablero en donde, de acuerdo al rol de cada miembro del equipo, se colocan tareas asignadas del proyecto. Para roles de administración, se pueden obtener gráficas del estado del proyecto, por eficiencia de recurso y por iteración del mismo.

Posee elementos para asignar tiempos estimados a las actividades, priorizaciones de tareas, vistas de portafolios, un blog de realimentación entre los clientes y el equipo o el administrador del proyecto, y mecanismos de envío de mensajes y correos electrónicos. En su página *web* se puede obtener una prueba por 30 días y la versión mínima de paga soporta 5 usuarios.

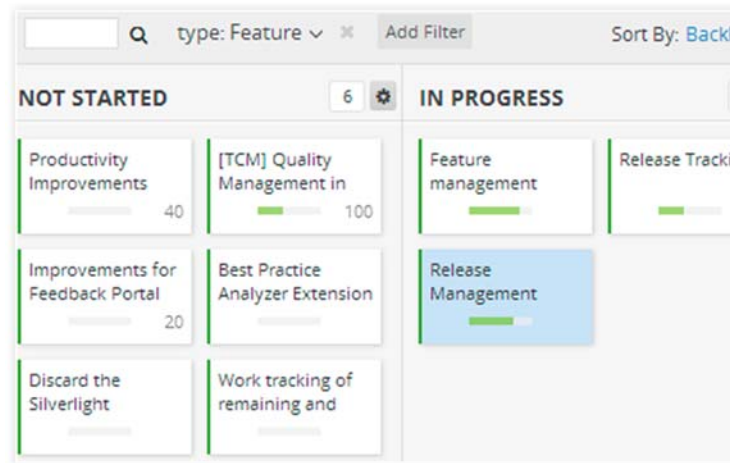


Figura 13. Visualización del tablero SCRUM de TeamPulse.

AXOSOFT ONTIME SCRUM

Axosoft ayuda a los administradores de proyectos y a los equipos a gestionar requisitos, comunicaciones e iteraciones en el desarrollo del proyecto. Se puede conectar con otras plataformas de desarrollo y cooperación. Cuenta con herramientas para mantener la comunicación entre los involucrados en el proyecto como un wiki y *bug tracker*.

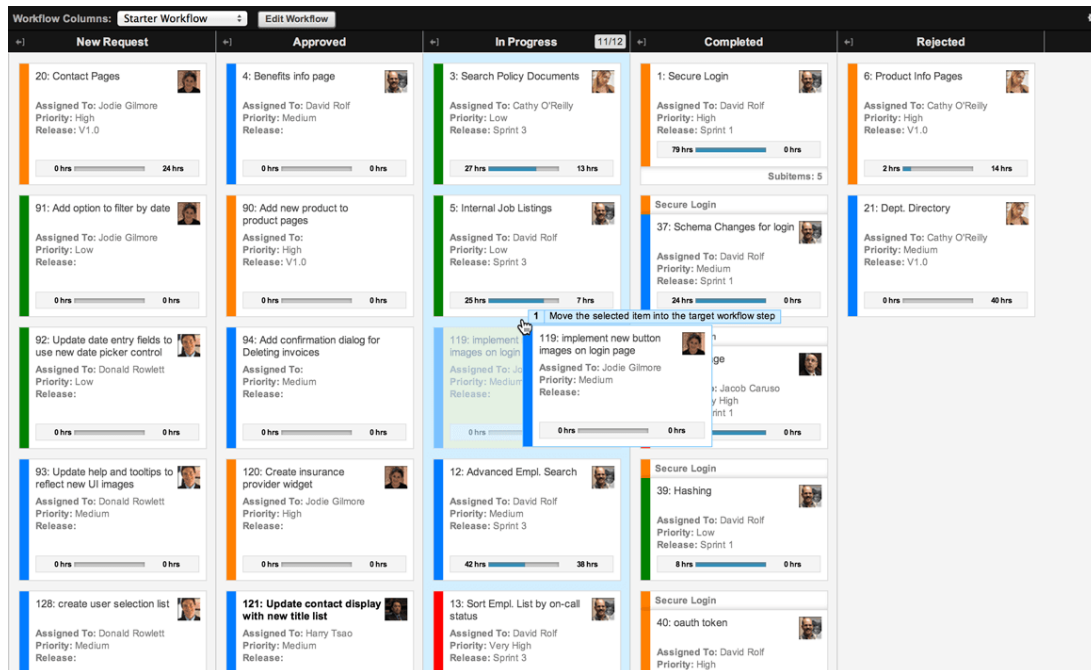


Figura 14. Vista del tablero de Axosoft¹².

¹² www.axosoft.com/features

Se adapta a métodos de administración como SCRUM, kanban y metodologías personalizadas pues cuenta con flujos de trabajo predefinidos y para personalizar. Con pantallas para el administrador del proyecto, se puede obtener reportes de progreso y trabajo actual por recurso, ayuda a una buena comunicación con notificaciones vía correo electrónico. En su página *web* se puede obtener una versión de prueba por 15 días.

34

LEANKIT

Herramienta informática con interfaz mediante navegador *web* y aplicación para dispositivos móviles, destaca el poder visualizar el proyecto con una imagen completa, evolucionada hacia las tecnologías actuales, de acceso y procesamiento en la nube y con interacción con otras plataformas o desarrollos.

Es la virtualización de un *kanban*, los usuarios pueden cambiar de estado sus tarjetas de acuerdo al avance del proyecto. Los tableros se pueden personalizar acorde con las necesidades del proyecto o industria, lo que la hace una herramienta versátil. Cuenta con métodos de comunicación para facilitar el trabajo colaborativo y seguimiento de cada tarjeta.

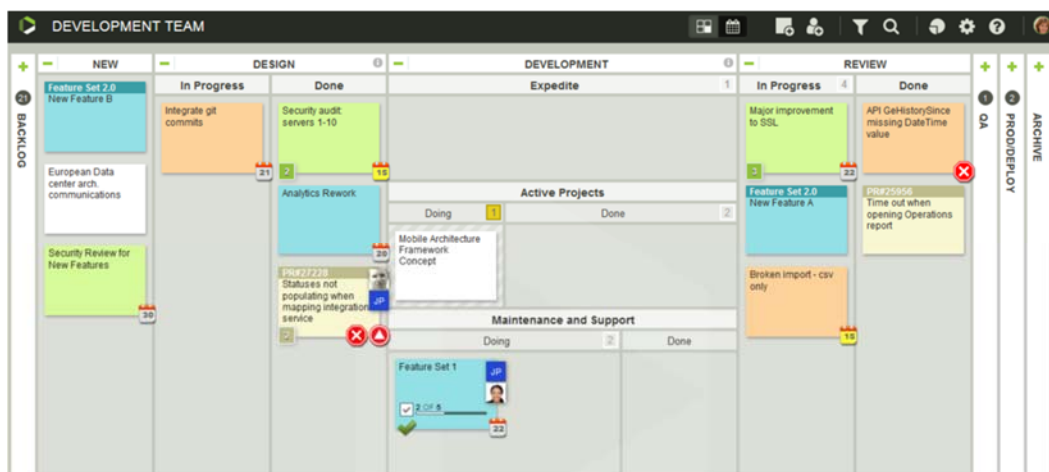


Figura 15. Vista del tablero de LeanKit¹³.

¹³ Leankit.com

TRELLO

Aplicación para dispositivos móviles desarrollada por *Trello Enterprise*, se autodefine como una lista de listas pero con mayores posibilidades, con una vista de portafolio puede organizar varios proyectos de cualquier tamaño y cualquier disciplina.

Para conocer el estado del proyecto se realizan listas que contienen actividades o más listas, se agregan miembros a cada actividad, y en cada una de sus tarjetas se pueden agregar comentarios, adjuntar archivos, crear listas de control, etiquetas y fechas de vencimiento. Al mover tarjetas entre las diferentes listas se puede implementar un tablero *kanban*.

Apoya el trabajo colaborativo entre los miembros del grupo ya que se notifica al recurso vía correo electrónico o aplicación con sólo citarlo. Es importante mencionar que todo el equipo ve el mismo tablero. En su página *web* se puede descargar la versión gratuita y adquirir la membresía *oro*.

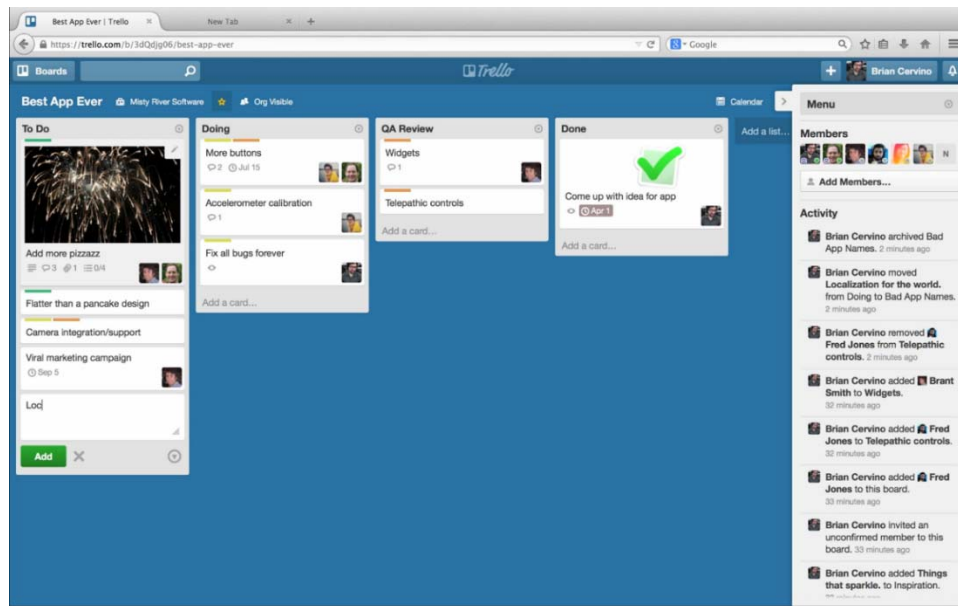


Figura 16. Tablero de la aplicación móvil Trello¹⁴.

¹⁴ help.trello.com/article/899-getting-started-video-demo

4.2 ANÁLISIS COMPARATIVO DE LAS APLICACIONES SELECCIONADAS

Como ya se ha mencionado, existen un gran número de herramientas colaborativas para el seguimiento de proyectos, todas ellas variables en su forma de procesamiento, almacenamiento y capacidades. Entre las características que comparten las aplicaciones descritas anteriormente se encuentran que son sistemas con arquitectura basada en *web*, cuentan procedimientos para ser utilizado por otro software (API), tienen capacidades multiusuarios con roles y notificaciones por correo electrónico y para realizar una distinción entre ellas, se construyó el siguiente cuadro comparativo¹⁵.

Nombre	Axosoft	JIRA	LeanKit	TeamPulse	Trello
Propiedades de gestión de proyectos					
Tareas jerárquicas	Sí	sí	sí	sí	no
Seguimiento de hitos	Sí	no	sí	no	no
Dependencias de tareas	No	sí	sí	sí	no
Gestión de ruta crítica	No	no	no	no	no
Diagramas de Gantt	No	sí	sí	sí	no
Diagramas Pert	No	no	no	no	no
Manejo de líneas de base	No	no	no	no	no
Administración de recursos	No	no	sí	sí	no
Nivelación de recursos	no	no	no	no	no
Seguimiento del tiempo	sí	sí	sí	sí	no
Seguimiento de costos	no	sí	no	no	no
Gestión de riesgos	no	sí	no	sí	no
Apoyo para SCRUM	sí	sí	sí	sí	no
Gestión de la cartera de proyectos	no	sí	sí	no	no
Flujos de trabajo personalizados	sí	sí	sí	sí	-
Propiedades de colaboración del proyecto					
Wiki de proyectos	sí	no	no	-	no
Foro de discusión	no	-	no	-	sí
Calendario compartido	no	sí	sí	-	sí
Comunicación entre involucrados	sí	-	sí	-	no
Intercambio de documentos	no	sí	sí	-	sí

Tabla 2. Comparación entre características de aplicaciones web para la gestión de proyectos.

¹⁵ Creación propia apoyada en <http://project-management.zone>. Project Management Zone es una iniciativa que recoge y presentar información sobre los sistemas de gestión de proyectos, muestra sus propiedades más importantes y se pueden comparar diferentes sistemas lado al lado.

Además, en la tabla 3 se observa el posicionamiento de herramientas de planificación de proyectos con base en sus menciones por profesionales en el área y en sitios *web*, ofertas de trabajo, los mejores clasificados y popularidad en redes sociales.

Nombre	Clasificación	Aplicación móvil	Disponibilidad y versiones
Axosoft	37	-	Prueba gratis por 14 días. Versión completa ¹⁶ \$25 USD usuario/mes.
JIRA	3	Android, iOS	Prueba gratis por 7 días. Versión completa ¹⁸ \$10 usuarios/mes.
LeanKit	29	Android, iOS	Versión lite gratis. Prueba completa gratis por 30 días. Versión estándar \$12 USD usuarios/mes.
TeamPulse	89	-	Prueba gratis por 30 días.
Trello	5	Android, iOS	Versión lite gratis. Versión negocios \$8.33USD usuario /mes.

Tabla 3. Características adicionales de las aplicaciones web para la administración de proyectos¹⁷.

Con base en estas dos tablas, las herramientas que incluyen un mayor número de propiedades para la gestión de proyectos son *Jira* y *LeanKit*, sin embargo, esta última cuenta con versiones de uso gratis lo que apoyaría su utilización en organizaciones que requieran optimizar sus costos.

Es por ello que, como complemento a las técnicas usuales de seguimiento y control de proyectos, se plantea el uso de programas informáticos y de las herramientas mencionadas para apoyar la gestión de proyectos mediante la visualización del flujo de trabajo, la que mejor las desarrolla, con base en el cuadro comparativo, es *LeanKit*.

Dada la similitud cuantitativa de características, no se descarta la posibilidad de utilizar *Jira* en futuros desarrollos con objeto de comprobar y comparar los resultados logrados.

¹⁶ Versión con almacenamiento en la nube. Mínimo de usuarios.

¹⁷ *Ibíd.*

4.3 LEANKIT

Después de varios años de trabajar en diferentes organizaciones y enfrentarse con los problemas típicos de la administración de proyectos, Jon Terry, Chris Hefley, Stephen Franklin y Daniel Norton desarrollaron *LeanKit* en respuesta a su propia necesidad de disponer de una herramienta con mayor dinamismo que las existentes.

LeanKit es una herramienta *web* que permite organizar de manera visual las actividades que conforman un proyecto mediante la distribución de tarjetas en un pizarrón; se puede describir fácilmente como un tablero *kanban* virtualizado que permite distinguir características de las tareas, flujos de trabajo y cuellos de botella dentro del desarrollo de uno o más proyectos.

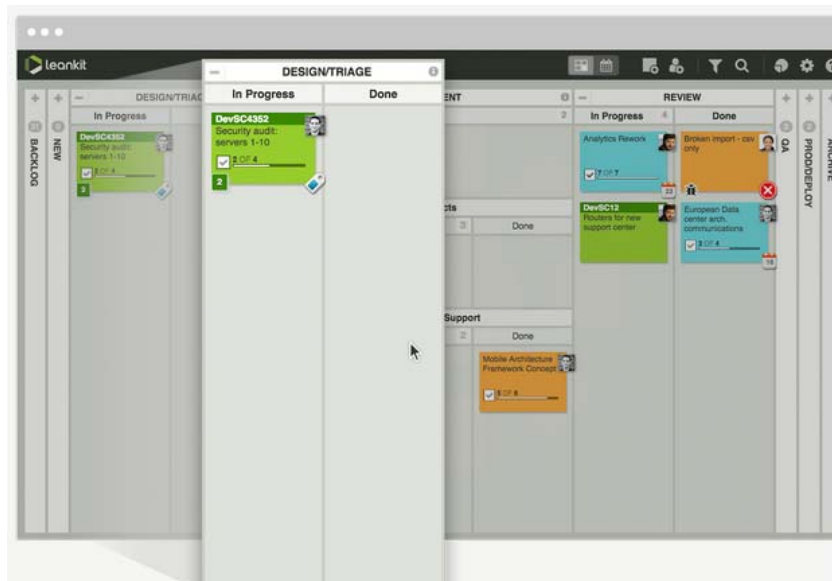


Figura 17. Vista de un tablero en LeanKit.

Es una manera de visualizar el trabajo y su flujo, reflejando tanto el estado actual del proyecto como el siguiente paso a dar. Opera bajo la filosofía de gestión a través de sistemas visuales, donde la visualización del trabajo advierte futuros problemas, considerando que cada actividad requiere una serie de pasos para alcanzar su cumplimiento.

4.3.1 CARACTERÍSTICAS

Tal como en un *kanban* convencional sus elementos esenciales son los tableros con carriles que describen cada etapa del proyecto; las tarjetas que representan actividades, y que se distribuyen y reasignan a lo largo del tablero conforme se realiza el trabajo; y los recursos o equipos asignados a cada tarea.

TABLERO

LeanKit permite que los carriles de los tableros se puedan adaptar al contexto y necesidades de cada equipo, así la personalización es al gusto del administrador pues es quien define las acciones y estados para controlar el avance de su proyecto.

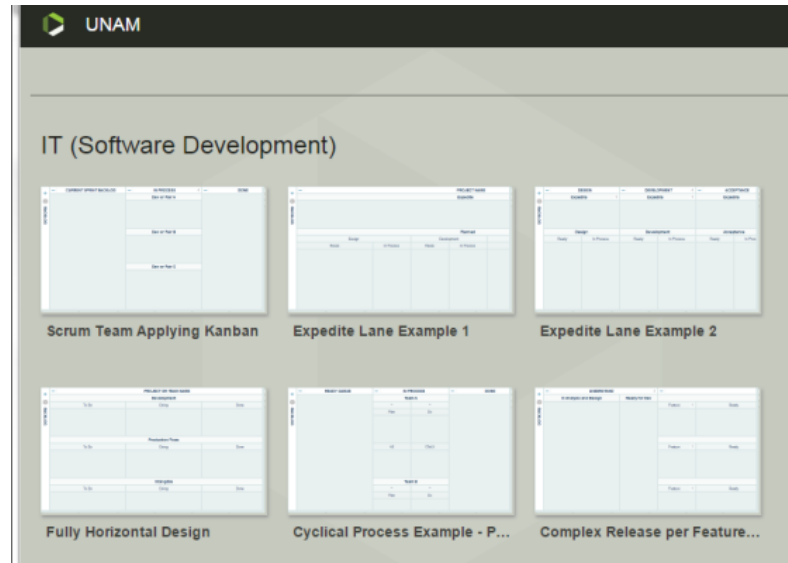


Figura 18. Templates o plantillas disponibles en LeanKit.

De acuerdo con la licencia elegida, se puede tener desde tres hasta un número ilimitado de tableros, para administrar programas o portafolio, e incluso es posible vincularlos entre ellos delegando una actividad a otro tablero y disgregar el trabajo en él.

Cada uno de los tableros en *LeanKit* tiene su propia URL o dirección de acceso, se pueden crear a partir más de 20 plantillas predefinidas y permite su personalización, incluso después de iniciado el trabajo; además se pueden guardar plantillas propias a partir de tableros ya creados para facilitar los trabajos futuros. Esta flexibilidad de configuración la hace una solución para diferentes tipos de proyectos e industrias.

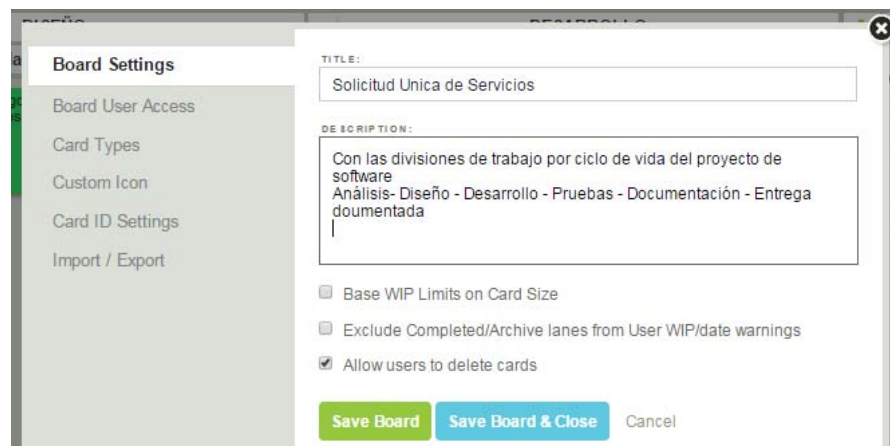


Figura 19. Configuraciones para los pizarrones.

TARJETAS

Los elementos de trabajo se representan visualmente mediante tarjetas que se mueven de acuerdo a su estado al carril que mejor representa su situación actual, haciendo más sencillo para el equipo comprender rápidamente el progreso y la prioridad de las actividades.

40

The screenshot shows the 'Details' view of a task card. On the left is a sidebar with navigation links: Tasks (0), Connections (0), Assigned Users (0), Comments (0), Attachments (0), and History (0). Below these is a 'PARENT CARD' section showing a green card 'Cancelación de folio' and a yellow card 'Tarjeta ligada desde otro tablero'. The main area contains a form for the task 'Tarjeta ligada desde otro tablero'. Fields include: TITLE (pre-filled), DESCRIPTION (with a rich text editor), PLANNED START (calendar icon), PLANNED FINISH (calendar icon), PRIORITY (Normal), UNBLOCKED (button), LANE (Deseos), PARENT CARD (Cancelación de folio), ADD EXTERNAL LINK (with examples like http://www.leankit.com and LeanKit), CARD TYPE (Task), SIZE, and TAGS. At the bottom are 'Save Changes' and 'Cancel' buttons. A footer bar includes a checkbox for 'Subscribe to Card Events by Email' and a URL.

Figura 20. Detalles de una tarjeta de actividad.

La información que se puede registrar en una tarjeta es extensa y variada como recursos asignados, tipo, prioridad, fechas de ejecución programada, hipervínculos y adjuntar documentos.

The screenshot shows a Kanban board with cards like 'Auditor', 'Manual Usuario', and 'Tablas de tiempos'. A context menu is open over a card, listing actions: Assign to, Unassign from, Subscribe, Open, Copy, Quick Edit, and Edit Dates. In the foreground, a 'CURRENT TYPES' dialog box is open, displaying a table of card types and task types.

CURRENT TYPES	CARD TYPE	TASK TYPE
Arquitectura	☒	☒
Búsqueda y act	☒	☒
Capacitación de usuarios	☒	☒
Carga de info	☒	☒

Below the table is an 'Add Card Type' button. At the bottom, there are dropdowns for 'DEFAULT CARD TYPE:' (set to Error) and 'DEFAULT TASK TYPE:' (set to Error).

Figura 21. Accesos rápidos y atributos de tarjetas.

LeanKit permite añadir tarjetas con un clic y modificar sus características mediante accesos rápidos, lo que la hace una herramienta accesible de manejar. Además se pueden crear tarjetas vía correo electrónico o aplicación móvil.

Otras características que tienen las tarjetas virtuales de *LeanKit* son:

- Incorpora un historial por elemento en el tablero.
- Aun cuando cada tarjeta representa una tarea, pueden tener “pesos” diferentes para balancear el trabajo de mejor forma.
- Cada actividad se puede extender en subactividades, incluso tener su propio tablero, haciendo este diagrama más legible y manejable.

ADMINISTRACIÓN DE USUARIOS

Para que el trabajo tenga una coordinación adecuada, se dispone de diferentes perfiles de acceso para los miembros del equipo de cada proyecto. De acuerdo con la licencia adquirida es el número de usuarios que se pueden unir, desde tres para la versión gratuita hasta un número ilimitado.

El administrador invita mediante un correo electrónico a cada usuario del tablero y maneja sus permisos acorde con su grado de involucramiento, garantizando que cada usuario tenga el nivel correcto de acceso.

Dependiendo del tipo de usuario, los permisos que se pueden otorgar son:

- | | |
|--------------------------------------|---|
| - Suscribirse a las actualizaciones. | - Modificar la configuración del tablero. |
| - Crear y actualizar tarjetas. | - Reasignar usuarios a las tarjetas. |
| - Importar y exportar informes. | - Agregar nuevos usuarios. |

VISUALIZACIÓN DEL TRABAJO

Ésta es la parte esencial de *LeanKit* pues aquí se observa lo que se está trabajando. Si se agregan fechas de inicio y final, también se puede tener una vista interactiva tipo calendario del proyecto. Una ventaja es que arrastrando las tarjetas en esta vista se añaden automáticamente fechas de inicio y final.

Ambas vistas cuentan con herramientas de búsqueda y filtros para agilizar la localización de tareas específicas y facilitar su utilización, así como la descarga de una imagen completa del tablero.

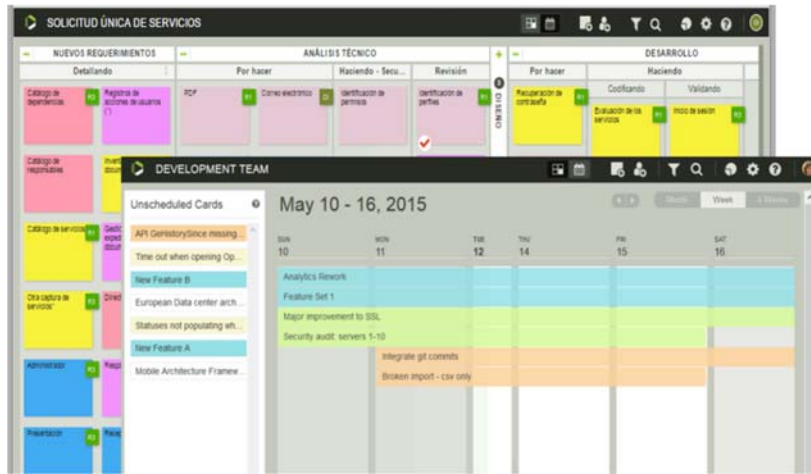


Figura 22. Vistas de flujo de trabajo y calendario.

ESTADÍSTICAS Y REPORTES

LeanKit ofrece métricas y reportes analíticos que permite conocer y diferenciar detalles del flujo de trabajo, haciendo posible un seguimiento puntual a la asignación y cumplimiento de tareas. Los reportes que se pueden hacer son los de flujo de trabajo acumulado, tiempo de ejecución y distribución de tarjetas (anexo A).

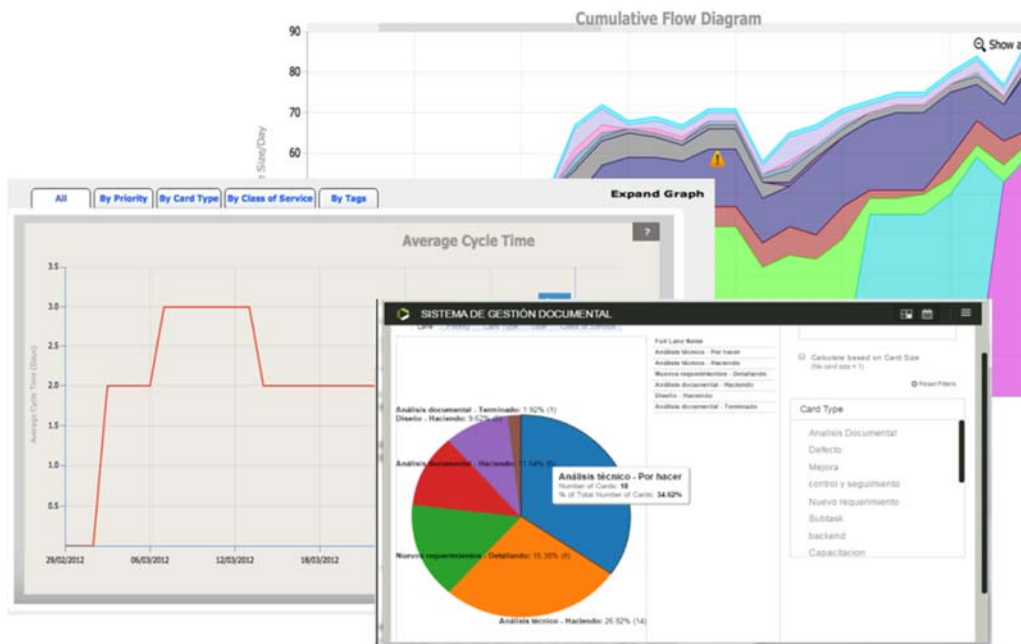


Figura 23. Estadísticas y reportes.

TRABAJO COLABORATIVO

Dadas las características de los ambientes *web*, este software permite que el trabajo colaborativo sea más efectivo, ya que cada involucrado, de acuerdo a su jerarquía, puede mover las actividades, asignar recursos, anotar comentarios o visualizar el estado del proyecto.



Figura 24. Seguimiento de actividad por usuario.

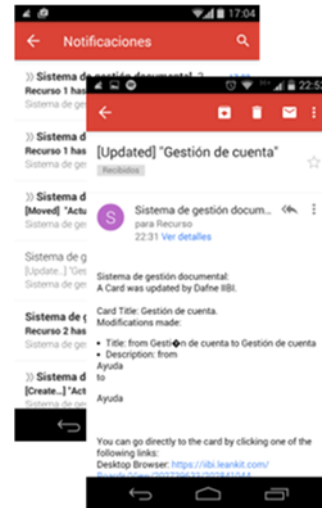


Figura 25. Alertas vía correo electrónico.

Las notificaciones sobre los cambios de estado se realizan justo en el momento que suceden los cambios y se envían de manera concreta a quienes corresponda y estén suscritos, ayudando así a solucionar problemas de comunicación.

Otra característica de trabajo colaborativo son las menciones tipo *twitter* en los comentarios (*@usuario*), que alertan al recurso nombrado en alguna tarjeta, aun cuando no se haya suscrito a la misma. Además, para ampliar el contenido de alguna actividad, se pueden adjuntar archivos PDF, imágenes u otros documentos de apoyo.

MULTIPLATAFORMA, PROCESAMIENTO Y ALMACENAMIENTO EN LA NUBE

LeanKit puede catalogarse como una herramienta informática móvil, pues además de poder acceder mediante un navegador *web* en una computadora personal, también es accesible desde *smartphones* o *tablets*. La interfaz en estos dispositivos es práctica y de acuerdo al sistema operativo tiene funcionalidad completa o solamente muestra las tarjetas y características previa suscripción.

44

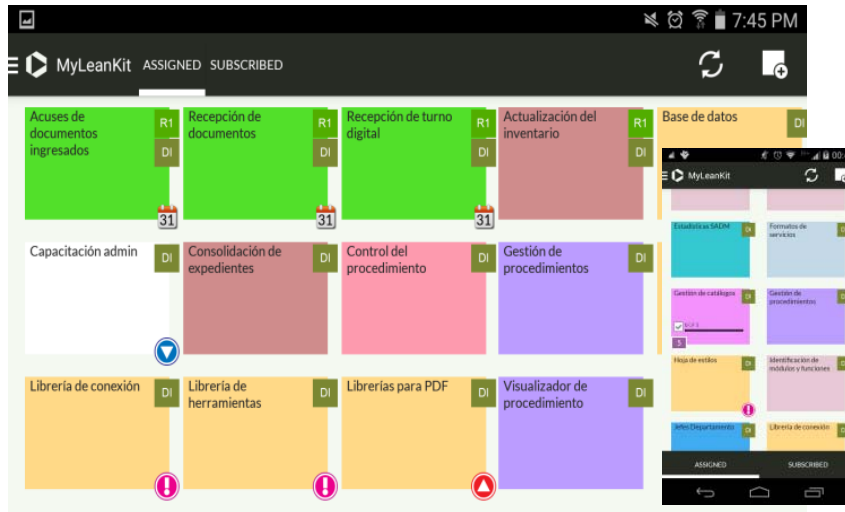


Figura 26. Vista de la aplicación móvil.

Respecto al procesamiento en la nube, esta solución es distribuida a través de un entorno descentralizado. Cuenta con equipo dedicado, bases de datos separadas de las organizacionales, filtros de acceso vía direcciones IP, encriptación de datos, respaldos programados y otros mecanismos para salvaguardar la información de sus usuarios y garantizar su disponibilidad e integridad¹⁸.

¹⁸ Dependiendo del esquema convenido.

CONECTIVIDAD E INTEGRACIÓN

LeanKit está incorporada en el grupo de aplicaciones que se pueden interconectar mediante la plataforma *Zapier* (www.zapier.com), lo que le permite vincularse con diferentes aplicaciones *web* tales como *Twitter*, *Facebook*, *Dropbox* y las herramientas de *Google*, entre muchos otros. Esta capacidad aun cuando no es estrictamente un servicio de *LeanKit* permite agregar las tarjetas mediante otros sistemas.

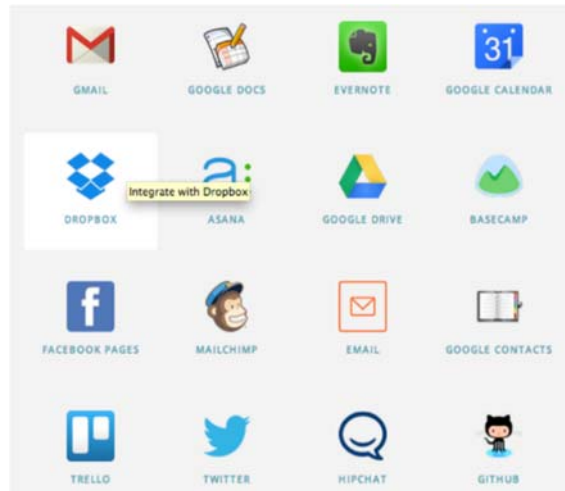


Figura 27. Aplicaciones en la plataforma.

Además, *LeanKit* provee una plataforma para desarrolladores interesados en crear aplicaciones propias para la administración de su herramienta (API).

4.3.2 VENTAJAS, DESVENTAJAS Y POTENCIALIDADES

De acuerdo con las características descritas, se encuentran las siguientes ventajas, desventajas y potencialidades en el uso de este software.

VENTAJAS

1. Planeación.
 - a. Favorece la planeación participativa al proyectar el tablero en una pantalla externa permitiendo la intervención del grupo en la planificación y seguimiento del trabajo.
 - b. La conectividad de los tableros ayuda a la administración de fases de proyectos o gestión de programas.

2. Comunicación.

- a. Ayuda a mantener la comunicación entre el equipo de trabajo, ya que provee de alarmas vía correo electrónico, dando así seguimiento puntual a actividades específicas.
- b. *LeanKit* puede ayudar a reducir el riesgo de malos entendidos, olvidos y retrasos con sus funciones de colaboración como la posibilidad de incorporar preguntas y comentarios por tarea.

3. Monitoreo y control.

- a. Por medio de las estadísticas que proporciona se puede medir el rendimiento y el tiempo de entrega del proyecto, ayudando así en la toma de decisiones.
- b. Se puede balancear el trabajo entre el grupo, asignando un peso a cada tarjeta dependiendo del esfuerzo necesario para terminarla.
- c. Permite observar la aparición de cuellos de botella y delegar el trabajo entre equipos o descomponer las tareas en un nuevo tablero para tener un mejor control sobre los recursos utilizados.
- d. Se pueden marcar las actividades que estén bloqueando el flujo de trabajo.
- e. Los cambios que realice algún colaborador se verán reflejados en el momento en el que suceden, teniendo así información actualizada del estado del proyecto.

4. Versatilidad

- a. Ofrece multiplataforma y conectividad mediante dispositivos móviles lo que permite conocer el estado del proyecto y mantener conectado al equipo sin importar su distribución geográfica.
- b. La exportación de bitácoras de uso, reportes y estado del tablero ayudan a llevar el trabajo a otros ambientes, tanto digitales como reales.
- c. La personalización de los tableros hace que el trabajo sea más sencillo ya que se ajusta a la forma de gestión del director del proyecto o de la organización misma.
- d. La importación de tarjetas mediante archivos CSV (delimitados por comas) es de gran ayuda si se tiene el detalle del proyecto en este formato.

5. Documentación.

- a. Los desarrolladores y el grupo que mantienen este software tienen una base de conocimientos con lista de FAQs, tutoriales, *webinars*, y foros en donde se pueden despejar dudas respecto de su uso y apoyan con artículos para su implementación.

DESVENTAJAS

Las desventajas que se encontraron en la aplicación son:

1. La interfaz en dispositivos móviles es limitada (para dispositivos *Android*) pues no es posible realizar cambios al tablero, sino sólo a las tarjetas que se controlan.
2. Falta la opción de exportar la configuración del tablero por lo que se tiene que hacer siempre "a mano".
3. No cuenta con dependencia obligatoria entre actividades, ni considera la ruta crítica del proyecto.

47

Entre los factores no imputables al sistema es importante mencionar:

- Para poder mapear un proyecto al tablero es necesario realizar previamente un esfuerzo en la planeación de sus actividades.
- La resistencia al cambio y un nivel limitado el manejo de nuevas tecnologías de los participantes del proyecto.
- Un sistema que no se utiliza y no se alimenta con información, al final termina siendo una herramienta más.
- La realización del proyecto no se lleva a cabo sólo con una buena documentación del mismo, hacen falta llevar a cabo las actividades que se programaron.
- El uso de alguna TIC para el manejo de un proyecto no garantiza su éxito.

POTENCIALIDADES

Entre los aspectos que considero mejorarían el funcionamiento de esta herramienta están:

1. Capacidad de asignar un carril completo a un usuario o grupo de trabajo.
2. Asignar de forma automática a un usuario las tarjetas de un carril determinado.
3. Automatizar la creación de tarjetas por medio de otro software.
4. Contar con un foro de discusión entre el cliente, el grupo de trabajo y el administrador.
5. Exportar la configuración del tablero y el tipo de actividades mediante otro software.
6. Actualmente se puede integrar a *Microsoft Project* sólo a través de *Team Foundation Software*. Sería de mayor utilidad una integración o exportación directa desde *MS Project* y otros paquetes.

5 DESARROLLO DE UN PUENTE ENTRE *MS PROJECT* Y *LEANKIT*

Ya se ha dicho que en el mercado existen gran cantidad de aplicaciones ayudan en la administración de proyectos, sin embargo, se puede obtener un mayor provecho explotando la viabilidad de interacción entre ellas, facilitando aún más el trabajo del líder de proyecto.

En el capítulo anterior se mencionaron a grandes rasgos las características de *LeanKit*, y entre ellas se halló la posibilidad de interconectarla a otras aplicaciones mediante la plataforma para desarrolladores que provee¹⁹.

Por otro lado, dos de las potencialidades señaladas consideran una mejora con respecto a la integración de esta aplicación y el popular *MS Project* y la automatización de la creación de elementos en el tablero.²⁰

Para sacar ventaja de la combinación de estos dos hallazgos y con el objetivo de propiciar la aceptación del uso de esta plataforma por parte de los administradores de proyectos, se planteó y desarrolló un programa que se comunica con uno de los programas de cómputo más populares y la aplicación *web* propuesta.

Así, se creó un programa puente que por un lado lee directamente el archivo de *Project* para obtener el listado y jerarquía de las tareas, y por el otro adquiere la configuración de los tableros de *LeanKit* y finalmente agrega las tareas de manera automática en forma de tarjetas en los diferentes carriles del tablero seleccionado.

Para el desarrollo de este puente se aprovechó la documentación y APIs de ambos programas: *Microsoft Project Exchange* (MPXJ), desarrollada en 2003 y que facilita manipular la información contenida en archivos mediante *.Net* y *Java*; y *LeanKit API*, provista por *LeanKit Inc.* que permite la manipulación de tarjetas contenidas en un tablero de dicha aplicación.

Asimismo, aprovechando los métodos integrados en las interfaces de programación, se codificaron *clases* (anexo B) para obtener los datos mínimos del tablero objetivo y del archivo *Ms Project*, así como para procesar la información en ambos programas.

¹⁹ Capítulo 4, Características. Pág. 45.

²⁰ Capítulo 4, Potencialidades. Pág. 47.

5.1 DESCRIPCIÓN DEL FUNCIONAMIENTO

Se considera que se comenzará a monitorear un proyecto desde su inicio, es decir que todas las tareas están pendientes o "*por hacer*". Las tarjetas que suba el programa a *LeanKit* se verán reflejadas en los carriles que correspondan a las tareas pendientes. Estos carriles suelen llamarse "To Do", "Por hacer" o simplemente "Pendientes".

El administrador utilizará el archivo en *Project* para subir las tarjetas a un tablero previamente configurado.

Para facilitar el uso de este programa se pueden colocar todas las tareas del proyecto en un solo carril, evitando la pre-configuración del tablero *LeanKit*. Por otro lado, si el proyecto (en el archivo *Project*) está dividido en etapas, el puente las reconocerá e indicará el número de etapas pre-configuradas en el tablero *LeanKit* y en el archivo *Ms Project* a fin de determinar la compatibilidad de las estructuras para poder asignar las tarjetas.

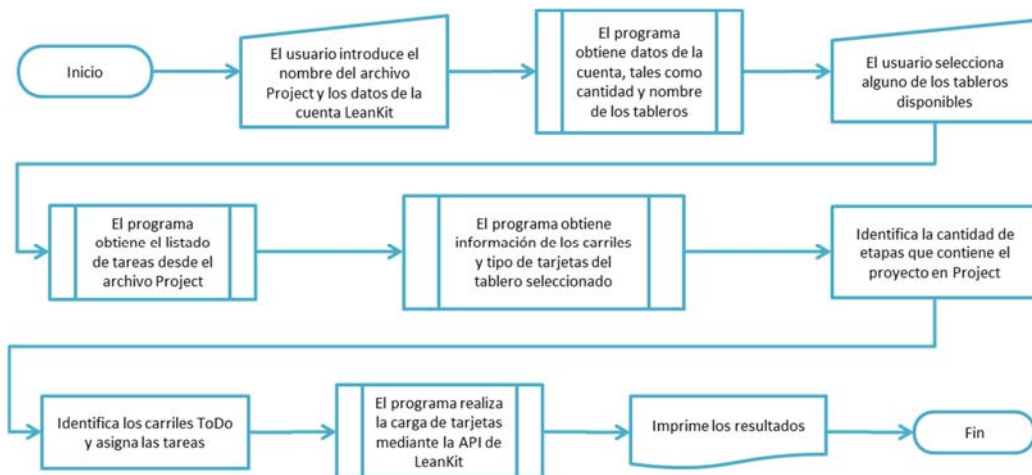


Figura 28. Diagrama del flujo que sigue el bridge para realizar su labor.

Las tareas que sean agregadas en forma de tarjetas a *LeanKit* conservarán algunos atributos provenientes del archivo *Ms Project*, tales como la fecha de inicio, fecha de término y prioridad.

Cabe destacar que el programa que se codificó sólo es para agregar las tareas en el tablero de *LeanKit*, los cambios que se realicen en el archivo de *Project* después de la carga no se verán reflejados en el tablero.

5.2 USO DEL PROGRAMA

La ejecución del programa se realiza de la siguiente manera:

El programa se distribuye en un archivo ZIP.

Nombre	Fecha de modifica...	Tipo	Tamaño
 ProjectToLeanKit	23/09/2015 03:07 ...	Archivo WinRAR ZIP	4,960 KB

51

Figura 29. Presentación del empaquetado del programa puente.

Al descomprimir el archivo se genera una carpeta llamada *ProjectToLeanKit* que contiene un archivo formato *JAR* y una carpeta llamada *lib* que contiene archivos necesarios para el funcionamiento del programa, figura 30.

Nombre	Fecha de modifica...	Tipo	Tamaño
 lib	07/09/2015 11:27 a...	Carpeta de archivos	
 ProjectToLeanKit	07/09/2015 11:26 a...	Executable Jar File	591 KB

Figura 30. Muestra del contenido del empaquetado de distribución del puente.

Para que el programa pueda leer el archivo de *Ms Project*, el usuario deberá copiarlo en la carpeta *ProjectToLeanKit*. Posteriormente, estando dentro de ésta carpeta se debe abrir una ventana de comandos, esto se puede realizar al mantener pulsada la tecla *Shift* y dar clic secundario en alguna parte vacía de la ventana.

Nombre	Fecha de modifica...	Tipo	Tamaño
 lib	24/09/2015 01:17 a...	Carpeta de archivos	
 ProjectToLeanKit	24/09/2015 01:13 a...	Executable Jar File	589 KB
 sigedda	24/09/2015 01:14 a...	Documento de Mi...	688 KB

Figura 31. Localización del archivo de *Ms Project* para exportar.

Aparecerá una ventana lista para recibir instrucciones. En la ventana debe escribir:

```
java -jar ProjectToLeanKit.jar <fileName.extension>
```

reemplazando <fileName.extension> con el nombre de su archivo y la extensión del mismo. Por ejemplo:

```
java -jar ProjectToLeanKit.jar sigedda.mpp
```

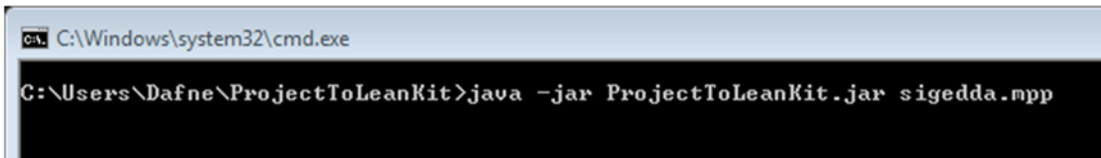


Figura 32. Ejemplo de instrucción para ejecutar el puente.

El programa solicitará tres datos esenciales de la aplicación *LeanKit*:

- o El dominio del tablero: se trata de una parte de la dirección URL que aparece al ingresar a *LeanKit* y se encuentra después de la doble diagonal y antes de “.leankit.com”



Figura 33. Dominio de la aplicación en un navegador web.

- o El correo electrónico asociado a la cuenta: Al crear una cuenta en *LeanKit* los usuarios ingresan un correo electrónico.
- o La contraseña de la cuenta: proporcionada al momento de crear la cuenta en *LeanKit*.

Al pulsar la tecla *enter* el programa se conectará a la cuenta de *LeanKit* y mostrará en pantalla el nombre de todos los tableros disponibles, permitiendo seleccionar el destino ingresando el número que le corresponda.

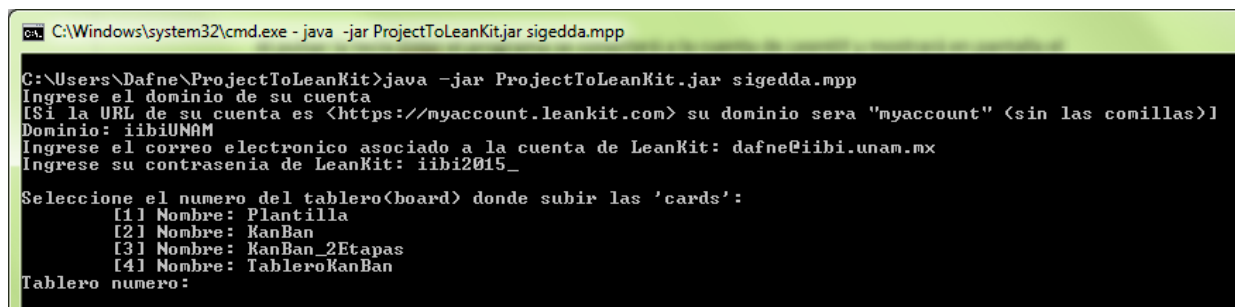


Figura 34. El programa muestra los tableros disponibles en la cuenta.

El programa identificará si el proyecto (en formato *mmp*) cuenta con más de una etapa definida. Posteriormente le permitirá indicar el número de etapas pre-configuradas el tablero *LeanKit* para poder asignar las tarjetas.

	 Nombre de tarea	Duración	Comienzo	Fin	Pr
1	▲ Sistema de control de documentos	363 días	mar 06/01/15	jue 26/05/16	
2	▷ Análisis documentales	79 días	mar 06/01/15	vie 24/04/15	
55	▷ Sistema de gestión de documentos	363 días	mar 06/01/15	jue 26/05/16	

El proyecto parece tener 2 etapas principales

1. Análisis documentales
2. Sistema de gestión de documentos

Que acción desea tomar?:

[1] Subir las tarjetas a la aplicación LeanKit respetando las dos etapas

[2] Subir las tarjetas a la aplicación LeanKit utilizando un tablero KanBan simple

Respuesta:

Figura 35. Etapas en el archivo de Microsoft Project y correspondencia en tablero kanban.

O bien, si el usuario lo deseara puede colocar todas las tareas del proyecto en un solo carril, facilitando el uso del programa y evitando la pre-configuración del tablero en la aplicación *LeanKit*. En ambos casos se deberá indicar el nombre de los carriles que representen la sección *To Do* o *Pendientes*. Enseguida se cargarán las tareas a *LeanKit* en forma de tarjetas y el programa habrá terminado.

Que acción desea tomar?:

[1] Subir las tarjetas a la aplicación LeanKit respetando las dos etapas

[2] Subir las tarjetas a la aplicación LeanKit utilizando un tablero KanBan simple

Respuesta: 1

Ingrese el nombre del carril destinado a las tareas pendientes: ToDo

KANBAN

+	-	TODO	-	DOING

leankit KANBAN_2ETAPAS

+	-	ETAPA1			-	ETAPA2	
		ToDo	Doing	Done		ToDo	Doing

Figura 36. Ejemplos de tableros kanban de una y dos etapas.

Al final de la ejecución del programa, el tablero kanban en *LeanKit* deber presentar las tareas en forma de tarjetas:

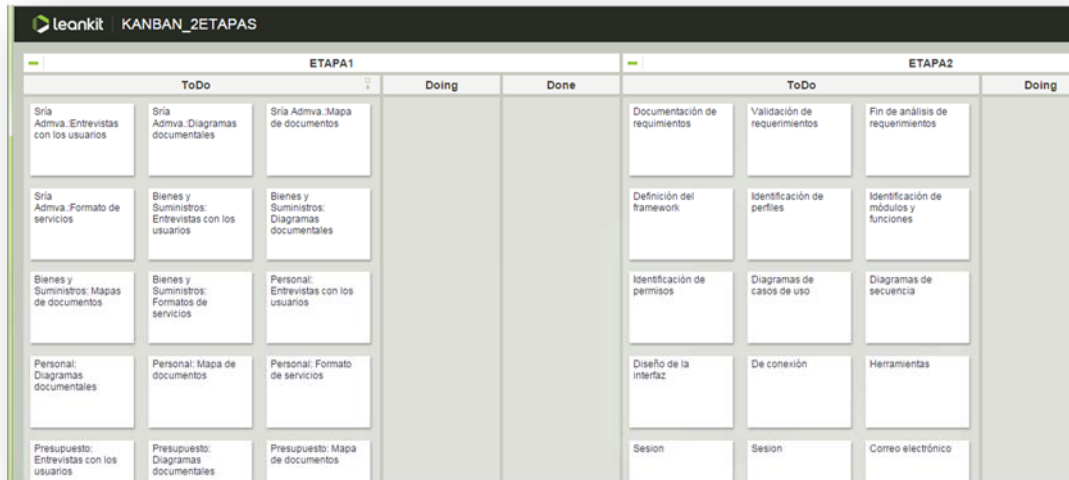


Figura 37. Resultado final de la ejecución del programa desarrollado.

5.3 CONDICIONES Y LIMITACIONES DEL PROGRAMA

CONDICIONES PREVIAS

Para que el programa tenga un adecuado funcionamiento, el usuario debe considerar lo siguiente:

1. Se debe contar con una cuenta activa de *LeanKit* y conocer sus credenciales y detalles de los tableros.
2. El tablero al cual se agregarán las tarjetas debe estar configurado de acuerdo al flujo de trabajo que el administrador monitoreará.
3. La computadora en la que se correrá el puente debe tener instalada la máquina virtual de java, disponible en <https://java.com/es/download>

LIMITACIONES

El programa cuenta con limitaciones en cuanto a funcionalidad y la estructura del tablero final, a saber:

1. Debido a que la API de *LeanKit* no permite agregar ni eliminar *carriles* dentro de un tablero, la estructura de los carriles debe estar previamente configurada de manera manual por el usuario y realizar la carga de tarjetas mediante este programa.
2. Dadas las características de la API, actualmente es necesario trabajar con la versión 2007 de *Microsoft Project*. Para hacerlo con herramientas más recientes es necesario esperar la liberación de la actualización de la API por parte de sus propietarios.

6 CASO DE ESTUDIO

Para ejemplificar el uso y ventajas de la herramienta seleccionada se presenta el proyecto *Sistema de Gestión de Documentos Académicos y Administrativos, SiGeDAA*, que actualmente se desarrolla en el departamento de cómputo del Instituto Investigaciones Bibliotecológicas y de la Información, IIBI, de la UNAM.

6.1 BREVE DESCRIPCIÓN DEL IIBI

Este Instituto tiene como misión generar investigaciones relevantes en el campo de la bibliotecología y de la información, apoyar la formación de recursos humanos de competencia internacional, así como difundir la disciplina.

Referente de primer orden en Latinoamérica por sus contribuciones teóricas y de investigación aplicada, es responsable del Posgrado en Bibliotecología y Estudios de la Información, incluido en el Padrón del Programa Nacional de Posgrados de Calidad del CONACYT.

Su comunidad se conforma por más de un centenar de personas entre investigadores, académicos y administrativos donde cada uno, en un ambiente participativo y de colaboración coadyuva en los logros del Instituto. Por citar ejemplos, en su informe 2013 - 2014, se enumeran la impartición de 46 cursos y seminarios, la dirección de 79 tesis, el desarrollo de 30 proyectos de investigación y la realización de 238 actividades académicas²¹.

6.1.1 DEPARTAMENTO DE CÓMPUTO

Apoya las actividades académicas del Instituto desarrollando, administrando y controlando su infraestructura tecnológica y de telecomunicación, tales como plataformas educativas, aulas virtuales y otros sistemas de apoyo a los investigadores; así como la facilitación de diversos servicios relacionados con el área.

²¹ Ríos J. (2014). 2do Informe de actividades. Instituto de Investigaciones Bibliotecológicas y de la Información, UNAM.

6.2 DESCRIPCIÓN DEL PROYECTO

6.2.1 ¿QUÉ ES EL SIGEDAA?

Es un sistema que facilitará el registro, control, recuperación y disposición de los documentos generados a partir del quehacer académico y administrativo del Instituto. Con él se podrá registrar la documentación y dar seguimiento de manera automática a las solicitudes de trámites y servicios que se realicen en las secretarías y unidades académicas y administrativas.

Asimismo, será un respaldo eficiente de los documentos derivados de sus actividades y funciones, garantizando la evidencia y rendición de cuentas, y cumplir con la normatividad requerida.

6.2.2 VENTAJAS DEL PROYECTO

- Brinda al usuario información adecuada y vigente para agilizar trámites o servicios, así como del estado de su solicitud.
- Reduce el uso de papel al implementar formatos y documentos electrónicos.
- Agiliza las búsquedas de expedientes y documentos.
- Apoya la generación de los indicadores para el Sistema de Gestión de la Calidad de la UNAM.
- Contribuye al control documental de la UNAM.

6.2.3 BENEFICIARIOS

- Comunidad del Instituto.
- Institutos, centros y programas del Subsistema de Humanidades.
- Personal de confianza o funcionarios que presten servicios administrativos.
- Entidades de la UNAM involucradas en el intercambio documental.

6.3 PLANEACIÓN Y SEGUIMIENTO

Ya que el objetivo de esta tesis va más allá de la utilización de las herramientas clásicas para la administración de proyectos, de primera instancia se ejemplificará la planeación del *Sistema de Gestión de Documentos Académicos y Administrativos* con un WBS abreviado con actividades relevantes del proyecto, figura 38, lo que dará como resultado una lectura sencilla de los diagramas de red, Gantt y ruta crítica derivados.

En la segunda parte, donde se desarrolla el seguimiento del proyecto con *LeanKit*, se utilizará el WBS completo para relacionarlo con las actividades del tablero.

Primera parte: Aplicación de las herramientas clásicas de la administración de proyectos.

Se realizó una lista, tabla 4, con las principales actividades a realizar para lograr el sistema, de ella se derivarán los diagramas subsecuentes.

Clave	Concepto	Antecedente	Duración (días)
INICIO	Principio del proyecto	-	0
A1	Análisis de gestión documental	Inicio	79
A2	Análisis de procedimientos	Inicio	79
B	Análisis de requerimientos	Inicio	90
C1	Definición de arquitectura de sistema	B	5
C2	Diseño de perfiles, módulos y funciones	A1,A2,B,C1	5
C3	Elaboración de diagramas	C2	30
D1	Codificación del <i>framework</i>	C3	22
D2	Desarrollo de módulos funcionales	D1	91
D3	Desarrollo de reportes	D1,E	16
E	Población de la base de datos	D1	3
F1	Pruebas por función del módulo	D2	15
F2	Pruebas de integración	D3,F2	15
F3	Pruebas de aceptación	F2	5
G1	Generación de manuales de usuario	F3	5
G2	Generación del manual técnico	F3	1
H	Migración al servidor final	F3	5
I1	Entrega	H,G1,G2	1
I2	Capacitación	I1	1
FIN	Cierre del proyecto	I2	0

Tabla 4. Relación de actividades principales para la realización del SiGeDAA.

Con base en la tabla anterior se realiza la estructura desglosada de trabajo.

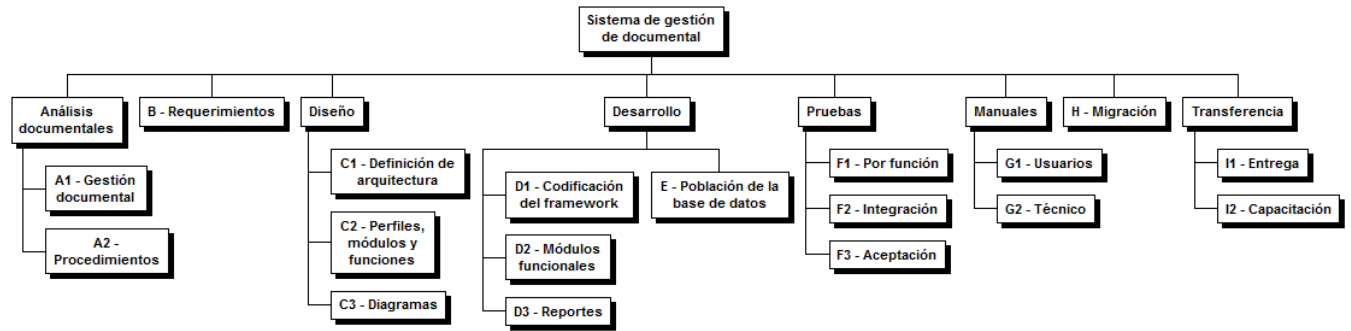


Figura 38. Estructura desglosada de trabajo abreviada para el SiGeDAA.

Es relevante comentar que esta EDT no es precisa siguiendo las recomendaciones para su elaboración sugeridas en el apartado de *Técnicas clásicas para la administración de proyectos*, por ejemplo los últimos niveles no representan el mismo esfuerzo de trabajo entre sí.

Considerando la tabla cuatro y analizando las dependencias entre tareas se realizó el siguiente diagrama de red.

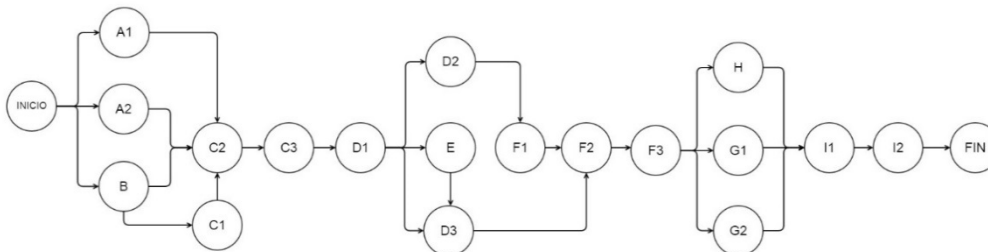


Figura 39. Diagrama de red de alto nivel para el Sistema de Gestión Documental.

Con este diagrama y agregando el tiempo necesario para realizar las actividades, es decir, calendarizándolas, se obtiene el diagrama de Gantt, en este caso utilizando *Microsoft Project*.

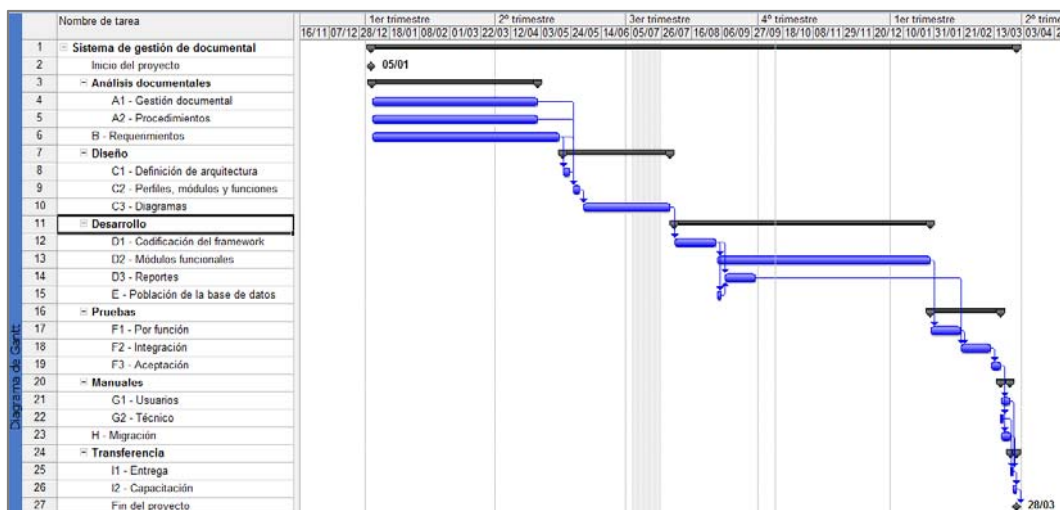


Figura 40. Diagrama de Gantt para el SiGeDAA.

Utilizando la función que provee *Microsoft Project* se calcula de manera inmediata la ruta crítica del proyecto:

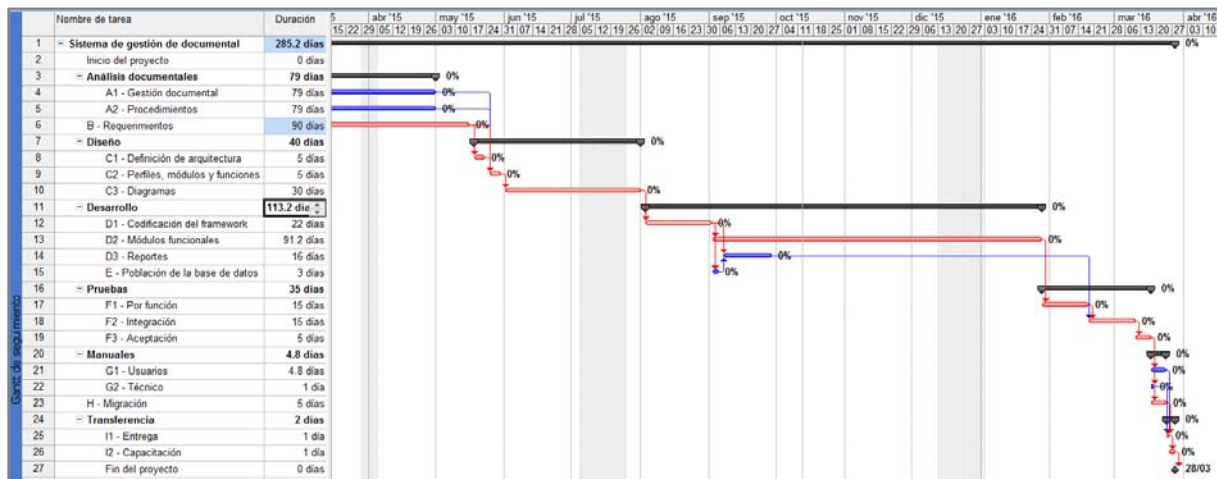


Figura 41. Diagrama de Gantt con ruta crítica en rojo del SiGeDAA.

La ruta crítica encontrada corresponde a la secuencia de actividades figura 42, con una duración de 285 días que es el total del proyecto.

Inicio - B - C1 - C2 - C3 - D1 - D2 - F1 - F2 - F3 - H - I1 - I2 - Fin

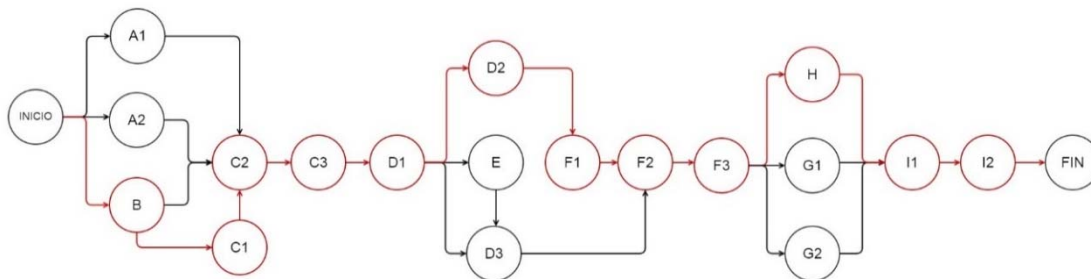


Figura 42. Diagrama de red con ruta crítica del SiGeDAA.

Con la realización de este ejercicio se puede observar que la ruta crítica contiene la mayoría de las actividades del proyecto, debido a que:

- El equipo de trabajo es limitado y en este momento no se planea hacer tareas en paralelo.
- Las actividades que se eligieron para este ejercicio son las más representativas, por lo que era de esperarse que todas estuvieran dentro de esta ruta crítica.

Segunda parte: Seguimiento del proyecto con LeanKit.

Para ilustrar el seguimiento del proyecto se utilizó la lista completa de actividades contempladas en para la realización del *SiGeDAA*.

Ya se ha mencionado que el resultado del uso de las herramientas para administración de proyectos es dinámico, es decir, cambian conforme avanza el proyecto; con base en ello, el WBS que se presenta en la figura 43 es el obtenido después del levantamiento de requerimientos, por lo que los módulos a desarrollar ya habían sido identificados a detalle.

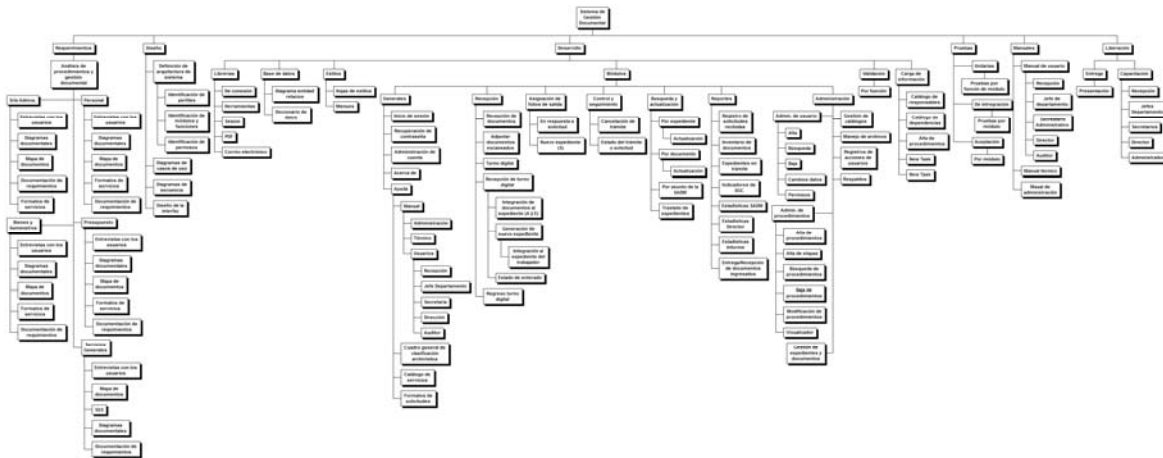


Figura 43. Estructura desglosada de trabajo del *SiGeDAA*²².

En primera instancia las tarjetas se agregaron de forma convencional, es decir con los mecanismos propios de la aplicación, sin embargo, en pruebas subsecuentes la carga de las 193 tareas resultantes se realizó con el programa puente desarrollado²³.

²² Se puede observar una figura de mayor tamaño en el anexo C.

²³ El detalle del funcionamiento de este desarrollo se presenta en el capítulo 5 de esta tesis.

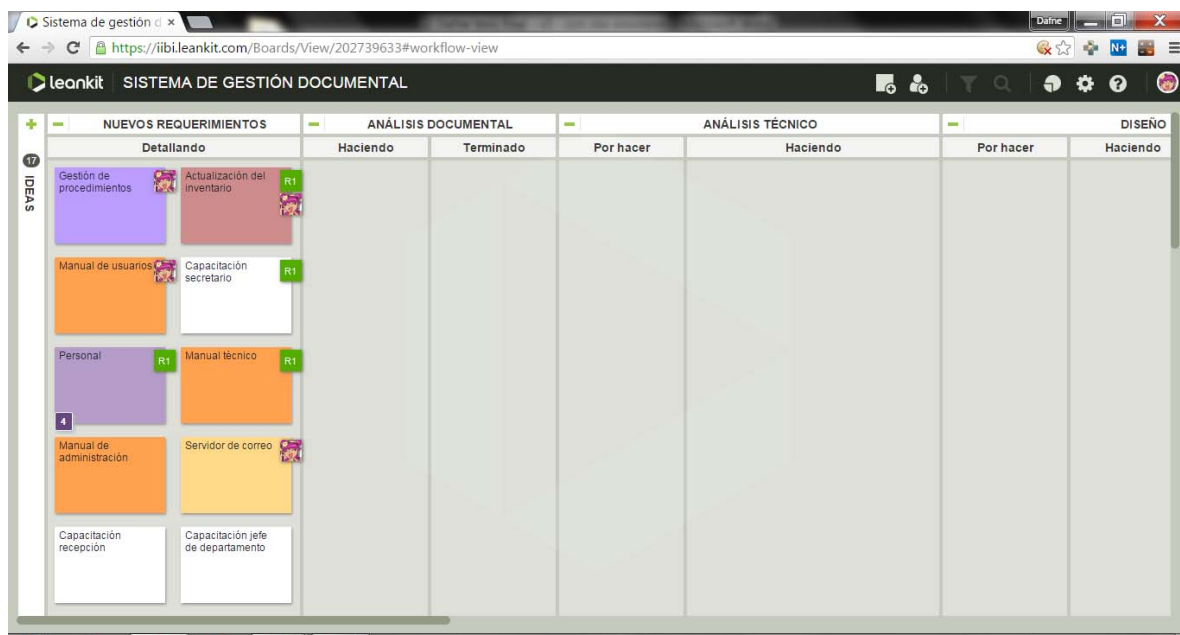


Figura 44. Tablero kanban para el seguimiento del SiGeDAA.

Algunos de los ajustes que se realizaron en la dinámica del grupo de trabajo para implantar el *kanban* fueron:

- La fuerte relación que tiene el proyecto con la teoría archivística, lo encamina no sólo a la sistematización del control y recuperación de documentos de trámite, sino a delimitar todos los flujos documentales, conocer a fondo la generación y manejo de estos documentos.

Esta característica impactó en el planteamiento y seguimiento del proyecto, derivando en la generación un pizarrón propio para ambas visiones, lo que permitió aprovechar la característica de tableros interconectados de *LeanKit*.

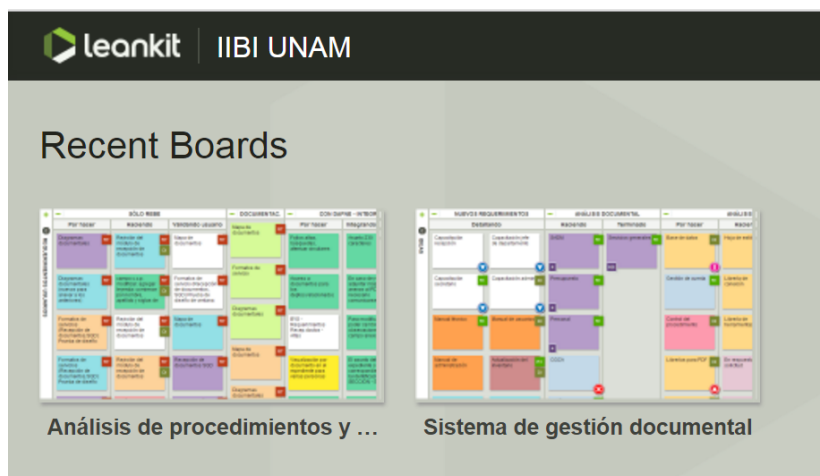


Figura 45. Imagen del home de la cuenta IIBI.

- El tablero para el seguimiento del análisis de procedimientos y gestión documental se alineó con base en los actores principales y su nivel de actuación en flujo de trabajo.

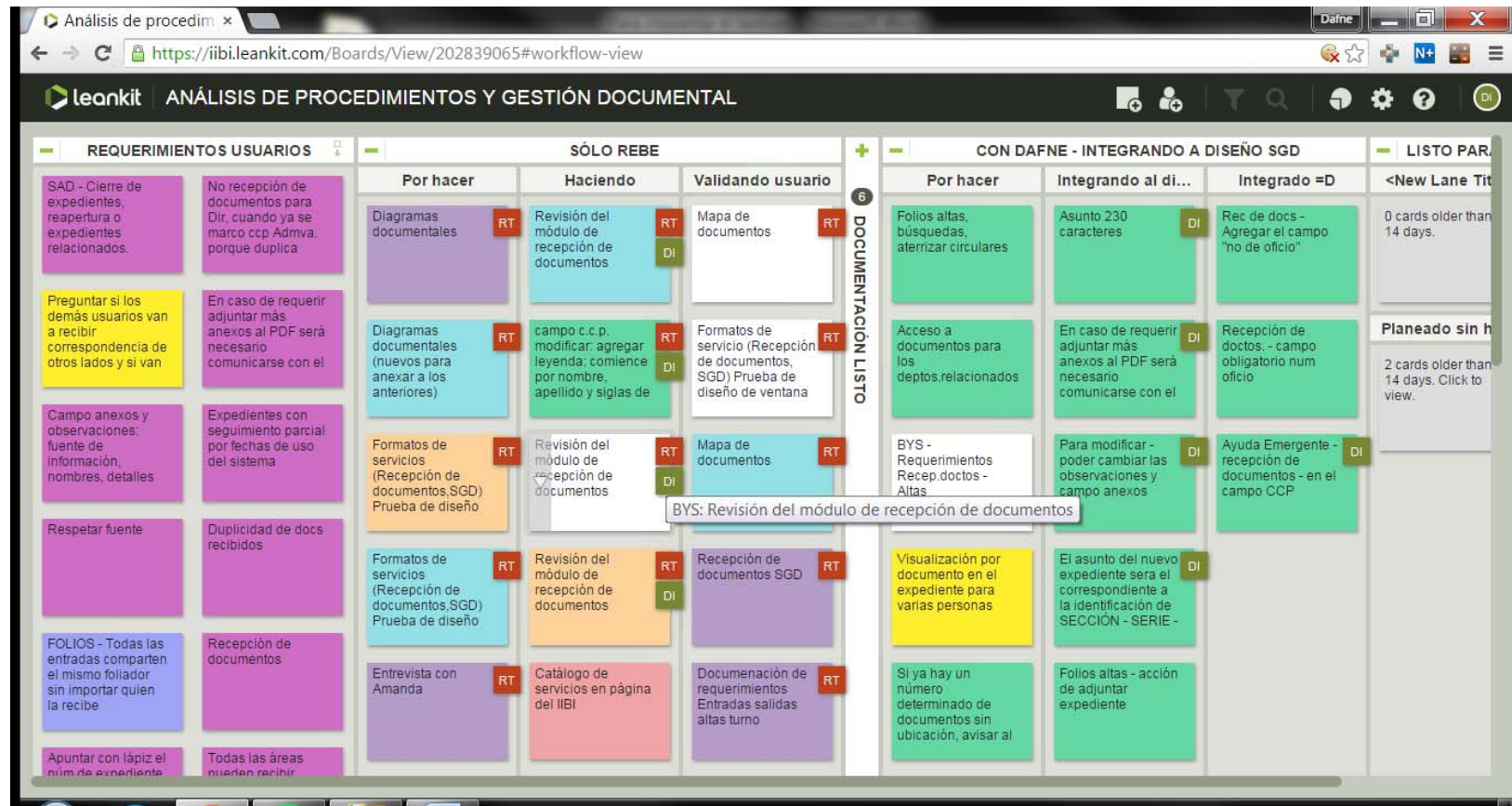


Figura 46. Tablero para el análisis documental.

- Por otro lado, el tablero para el monitoreo del desarrollo del sistema se dispuso acorde con el ciclo básico de programación es decir análisis, diseño, desarrollo, pruebas e implementación, cada uno de ellos con sus propios carriles *Por hacer*, *Haciendo* y *Revisando*. En la siguiente figura, además se puede observar una vista del modo de configuración del tablero.

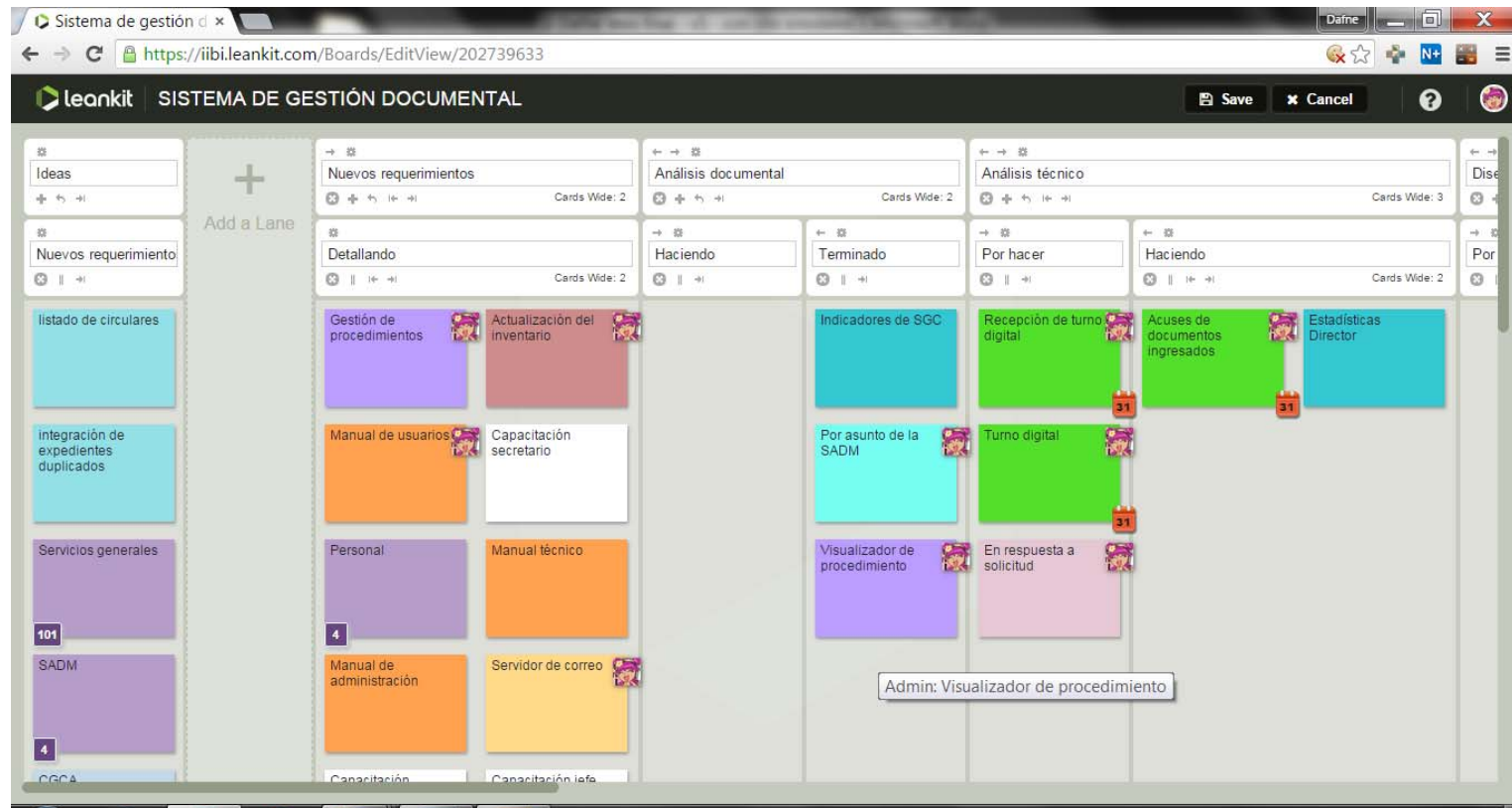


Figura 47. Disposición de los carriles para el tablero de SiGeDAA.

Debido a que el resultado del análisis documental descrito en el pizarrón anterior debe reflejarse en el diseño y funcionamiento del sistema, se dispuso una columna específica para tal efecto, aprovechando la característica que tiene *LeanKit* de ligar dos tableros mediante una tarjeta.



Figura 48. Ejemplo tarjeta ligada con otro tablero.

- Para contribuir con una buena comunicación, se crearon convenciones tanto para los indicadores de las tarjetas (color, símbolos, prioridades); cómo para los comentarios a las actividades.

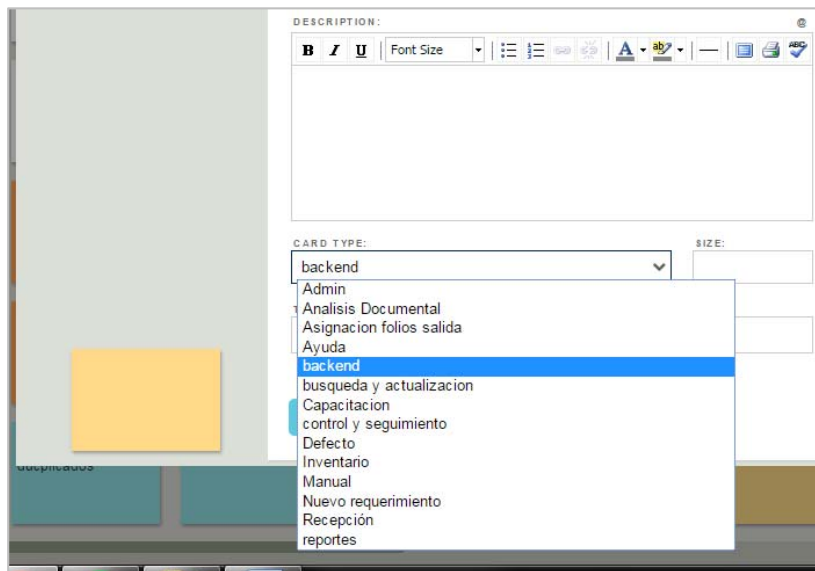


Figura 49. Lista de tipos de tarjetas disponibles para el tablero SiGeDAA.

Por citar un ejemplo, las políticas que se detectó deben implementarse en el Instituto para el correcto funcionamiento del sistema documental (procedimientos y herramienta *web*) son tarjetas color magenta, dispuestas en la primer columna del tablero correspondiente al Análisis de procedimientos y gestión documental, Requerimientos usuario.

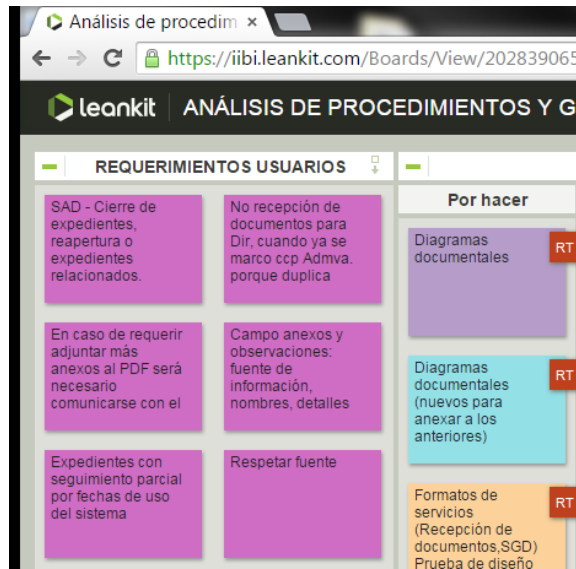


Figura 50. Ejemplo de convención para las tarjetas relacionadas con políticas a implementar.

- Se convino que en el orden del día de las reuniones se revisara el avance del proyecto con base en el estado del tablero.
- Este instrumento se implementó sólo para el grupo de desarrollo de este proyecto, el desarrollo de los demás sistemas y la jefatura del departamento siguen trabajando de manera tradicional.

Para las dos ópticas, se mapearon los paquetes de trabajo en cada tablero para dar seguimiento al desarrollo del proyecto, configurando según las necesidades, los carriles y tipos de tarjetas.

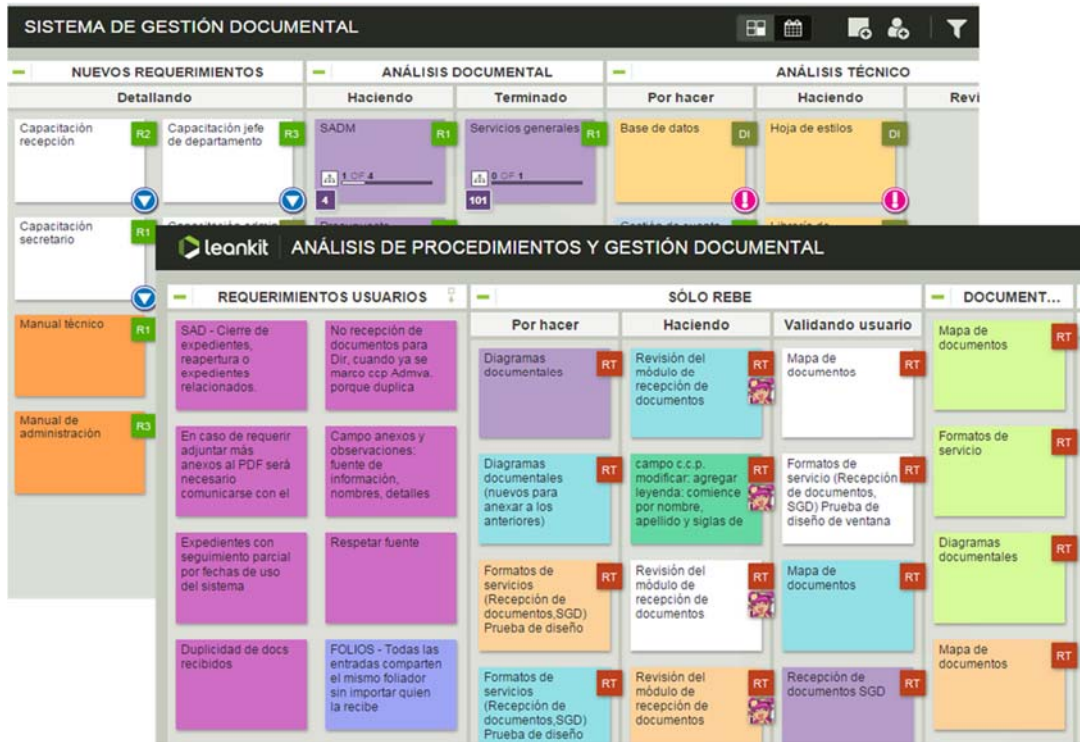


Figura 51. Mapeo de actividades principales por subproyecto del SGD.

En cada tablero se han asignado tareas al grupo de desarrollo, quienes se encargan de actualizar sus estados y contribuyen a la buena comunicación con una participación activa.

Los cuellos de botella se pueden observar cuando, por un lado un número importante de tarjetas se agrupa en un carril (ya el WIP no se restringió) y por el otro, las tarjetas llevan “cierto tiempo sin moverse”.

Este proyecto actualmente está en desarrollo y contempla un tiempo de ejecución de año y medio comenzando en enero de 2015. La implantación de *LeanKit* para su monitoreo se realizó a lo largo del mes de mayo. Al implementar este sistema se tendrá un impacto significativo para la comunidad del Instituto y en su propia labor con respecto a la gestión documental y la calidad en los servicios que ofrece su Secretaría Administrativa.

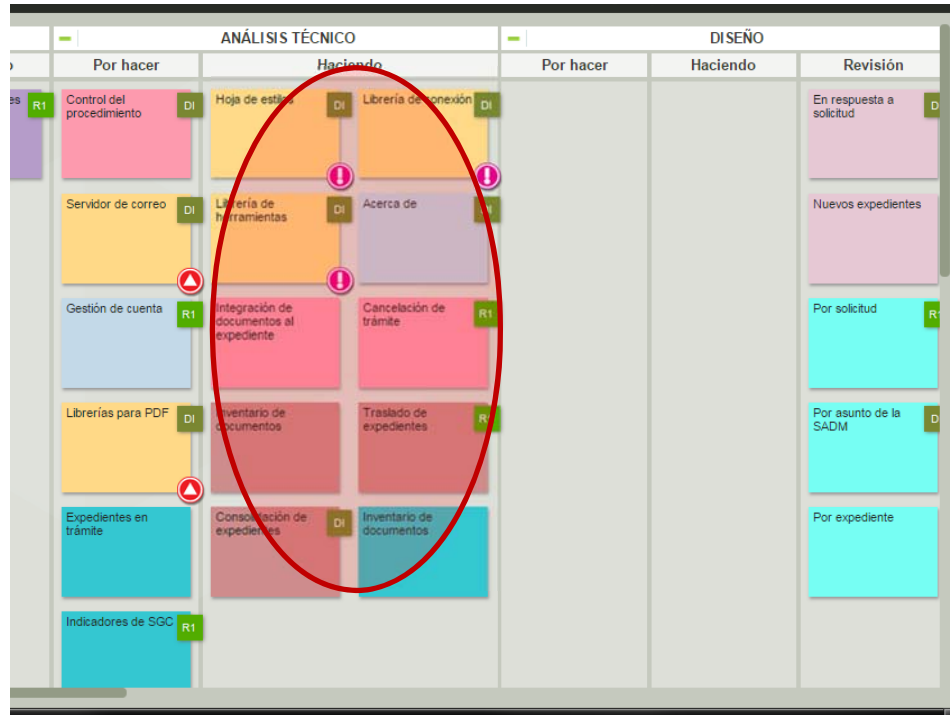


Figura 52. Cuello de botella en análisis técnico del SiGeDAA.

Para ejemplificar el uso de la aplicación propuesta se dispusieron impresiones de pantalla tomadas durante el desarrollo del sistema, sin embargo, considero más importante destacar las ventajas y lecciones aprendidas ganadas.

6.4 IMPRESIONES SOBRE EL USO LEANKIT

6.4.1 SITUACIONES PARA EL APROVECHAMIENTO DEL SOFTWARE

Las principales circunstancias que presentaba el desarrollo del proyecto y sobre las cuáles se aprovechó esta herramienta fueron:

- La comunicación de cambios y nuevos requerimientos se realizaban principalmente de forma verbal o mediante cadenas de correo electrónico, lo que dificulta el seguimiento de acuerdos.
- Las juntas de trabajo solían ser largas y sin un orden de atención por la gran cantidad de asuntos por atender y los diferentes estilos de trabajo.
- La limitante en los recursos económicos disponibles estimuló experimentar otras formas de realizar las actividades encomendadas.
- La información del estado del proyecto no estaba en una sola fuente de información.

6.4.2 LEANKIT EN EL DESARROLLO DE SISTEMAS

El equipo de desarrollo de software del IIBI se caracteriza por buscar nuevas formas de realizar las tareas cotidianas como la integración de diferentes herramientas de TI para la administración de sus proyectos.

68

Actualmente, se está utilizando *LeanKit* en un desarrollo de gran impacto proyectado a largo plazo. Con la incorporación de esta herramienta, se busca mejorar la coordinación de grupos multidisciplinarios, donde la comunicación efectiva es base para lograr los objetivos establecidos.

Cabe mencionar que no todo el grupo de desarrolladores en el IIBI utiliza *LeanKit* para administrar sus proyectos, sin embargo, después del plazo de evaluación y aprendizaje, se pretende incorporarla en tanto en otros proyectos del departamento de cómputo como en otros departamentos de apoyo técnico.

6.4.3 MEJORAS OBTENIDAS

La más importante es conocer el estado actual del proyecto, pues a primera vista se pueden observar las tareas que requieren atención, las prioritarias y donde se están generando cuellos de botella. Ahora todos los involucrados tenemos acceso al instante a la misma información precisa y actualizada.

Otra ha sido la forma en que nos comunicamos y colaboramos, pues se tienen expectativas claras sobre lo que se necesita realizar de forma inmediata y proporciona un recordatorio visual de nuestros compromisos.

Con la adición *LeanKit*, las reuniones de trabajo se atienden con más orden, en conocimiento preciso de los objetivos alcanzados y los no logrados.

El uso de esta aplicación mejoró el monitoreo, haciéndolo también sencillo y amigable aun cuando algunos de los usuarios no fuesen "nativos digitales".

Finalmente, los informes obtenidos mediante *LeanKit* se pueden incorporar en la documentación del proyecto.

6.4.4 LECCIONES APRENDIDAS

- El número de actividades/tarjetas a gestionar rebasó la capacidad de administración complicando la gestión en un inicio. Para mejorar esto se redujo su número, en algunos casos, sólo a tareas clave.
- Es importante elegir el nivel de profundidad de las actividades se van a gestionar con *LeanKit* ya que sus limitantes actuales no permiten dependencia obligatoria ni gestión simultanea de actividades, por lo que se puede aumentar el tiempo invertido en la administración.
- Ya que en un principio los colaboradores dejaron de lado la actualización del tablero, se revela la importancia de realizar la implantación del *kanban* de forma incluyente, haciendo del conocimiento de todos el impacto que puede ocasionar su uso e involucrándolos en la configuración del tablero; lo anterior con el fin de estimular la actualización del estado del proyecto a sus tareas cotidianas.

Para el analista documental fue difícil enrolarse en una dinámica donde ningún trabajo debe ser hecho fuera de la secuencia acordada, sin embargo, su utilización fue bien aceptada después de resaltar las ventajas más un periodo de ajuste.

- Falta establecer políticas sobre cómo y cuándo debe actualizarse el tablero, así como limitar el “trabajo en curso” de cada carril y delimitar el esfuerzo preciso para cada tarjeta. Todo ello es necesario para la mejora continua del monitoreo del trabajo.

Con base en lo anterior se puede decir que la incorporación de herramientas visuales ayuda a crear transparencia sobre el trabajo en curso y genera un cambio en la perspectiva del avance del proyecto. Con este conocimiento es más sencillo hacer cambios en la planeación del trabajo para usar de forma eficiente los recursos; además que ayuda a los miembros del equipo enfocarse en su trabajo y ser más productivos para cumplir sus metas.

CONCLUSIONES

Esta tesis tiene como objetivo mostrar la conveniencia de incorporar *LeanKit* a las herramientas básicas de la administración de proyectos e identificar las ventajas su utilización mediante la presentación de un caso práctico.

La forma en la que accedemos y ocupamos recursos de información ha cambiado, por ejemplo, hoy en día la computación en la nube ofrece servicios y funcionalidades procesados en una ubicación ajena a nosotros. El uso estas aplicaciones es ventajosa para las empresas que no cuentan con las máquinas apropiadas para procesar información, además elimina costos de mantenimiento, sin embargo, se generan costos por la renta de dichos servicios.

Existe una vasta oferta de aplicaciones *web* para la administración y seguimiento de proyectos que además seguirá incrementándose por la naturaleza misma del desarrollo de software. Entre sus características se encuentran las tareas jerárquicas, los diagramas de Gantt, la administración de recursos, el seguimiento del tiempo y flujos de trabajo personalizados.

Sobre la incorporación de *LeanKit* al seguimiento del proyecto *Sistema de Gestión de Documentos Académicos y Administrativos* se observa que esta herramienta:

- Permite conocer de un vistazo el estado actual del proyecto, los involucrados tienen acceso instantáneo a la misma información precisa y actualizada.
- Ayuda a balancear el trabajo y funciona como recordatorio visual de los compromisos que tiene cada integrante del grupo de desarrollo.
- Facilita el seguimiento y gestión del proyecto al responsabilizar a cada persona la actualización del estado de su trabajo.
- Ofrece conectividad mediante dispositivos móviles y cuenta con recursos de aprendizaje como bases de conocimientos, lista de FAQs, tutoriales, *webinars* y foros de discusión, lo que es de gran ayuda para su uso e implementación.

Al mismo tiempo, en apoyo a la aceptación del uso de *LeanKit* en el seguimiento de proyectos, se desarrolló un programa puente entre ésta aplicación *web* y el popular *Microsoft Project* aprovechando que ambos programas cuentan con procedimientos compatibles que permiten su interoperatividad.

El programa puente busca las actividades en un archivo de *Project* (formato *mmp*) y coloca las actividades en forma de tarjetas en alguno de los tableros de *LeanKit* previamente verificados, automatizando así uno de los primeros pasos para monitorear un proyecto con la herramienta propuesta.

Finalmente, para poder dar una justa valoración sobre la incorporación del *LeanKit* para el seguimiento de proyectos es necesario realizar una exploración más profunda sobre las demás herramientas de su ramo, así como su incorporación en proyectos similares y comparar resultados respecto de las mejoras en la administración.

Comentario final

Con base en mi experiencia profesional así como con el desarrollo de este trabajo, sobresale la importancia del seguimiento de un proyecto para que se lleve a buen término, por lo que incluir herramientas que ayuden en este sentido puede marcar la diferencia en el logro de este cometido.

El objetivo de este trabajo no es hacer que los directores de proyectos cambien al uso exclusivo de esta herramienta, sino que la consideren como una más dentro de los recursos que pueden apoyar su actividad.

Más allá de la elección y utilización de software para la gestión de proyectos, lo importante es conocer los objetivos que se persiguen con su empleo para que impacten de forma positiva en la ejecución del proyecto.

Es importante mencionar que la integración de estas herramientas no asegura el éxito del proyecto, pero si representa una ventaja en su administración, por lo que, saber elegirirlas y combinarlas es una habilidad que los administradores de proyectos deben cultivar.

ANEXOS

A. ESTADÍSTICAS Y REPORTES DE LEANKIT

A continuación se describen algunos de los reportes analíticos que ofrece LeanKit:

Diagramas de distribución de tarjetas. Muestran la distribución actual de las tarjetas, ya sea por tipo, por recurso asignado o prioridad, esto ayuda a notar que recurso está más cargado de actividades, cual es el tipo predominante tipo de actividades en el proyecto.

73

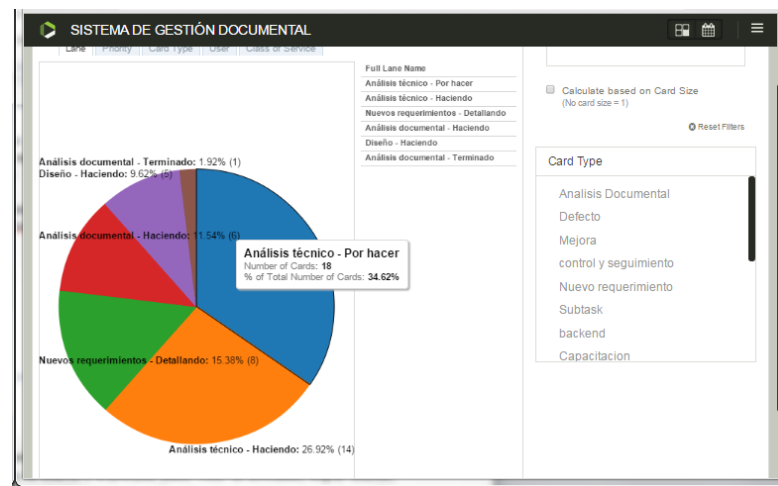


Figura 53. Distribución de actividades por recursos.

Diagrama de flujo acumulado. Registra la cantidad de trabajo en progreso con respecto del tiempo para cada carril del tablero. Con este diagrama se pueden conocer los carriles saturados y cuando se comenzó a retrasar el trabajo en ellos.

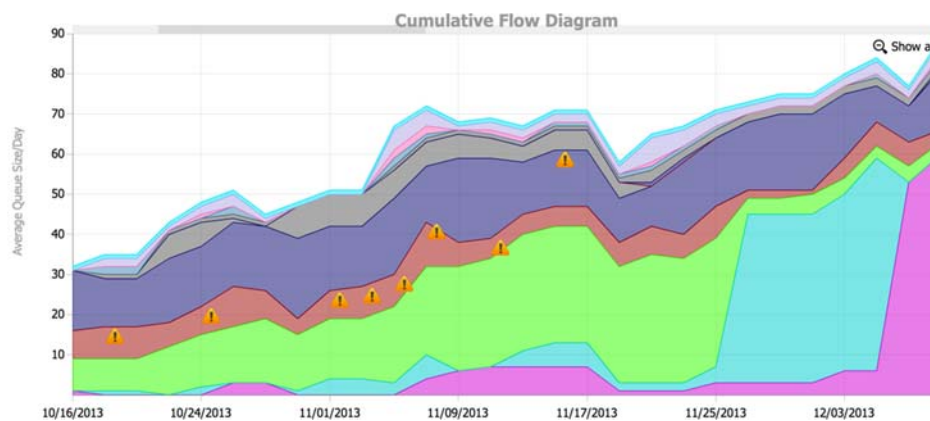


Figura 54. Diagrama de flujo acumulado²⁴.

²⁴ <https://support.leankit.com>

Diagramas de tiempo de ejecución. Muestra el tiempo que se ocupa en cada tarjeta para cambiar su estado, o cual es la etapa o carril con más tiempo de ejecución, además se puede obtener el tiempo de entrega especificando un carril de inicio y otro de término para poder calcular el tiempo. Estos reportes se pueden dar por prioridad, tipo de tarjeta, etiqueta o todo el contenido del tablero.

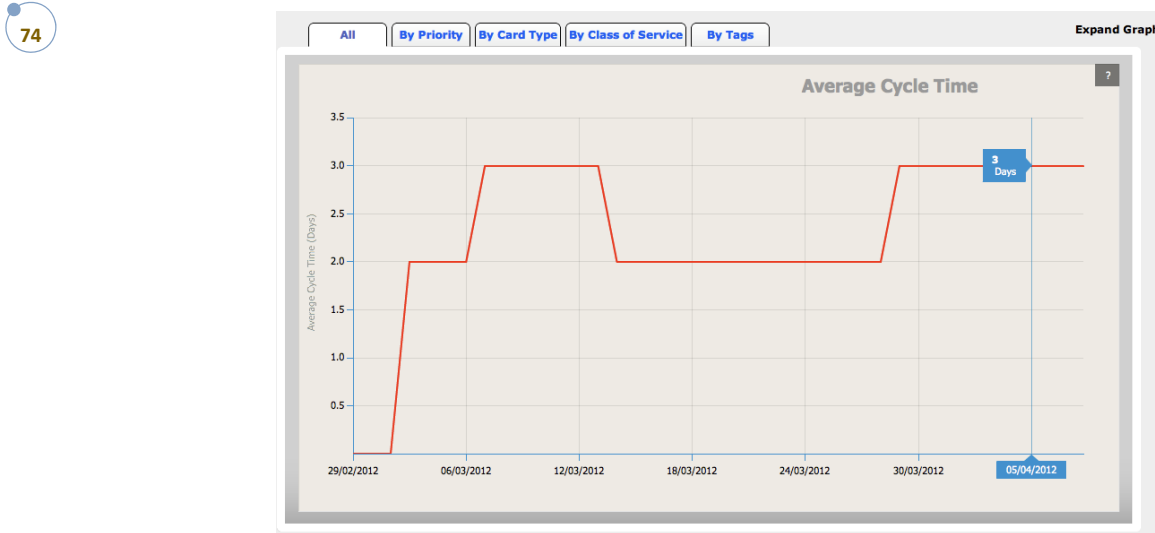


Figura 55. Ciclo de tiempo promedio²⁵.

²⁵ <https://support.leankit.com>

B. DICCIONARIO DE CLASES DEL BRIDGE PROJECTOTOLEANKIT

Para interpretar las descripciones de las clases es necesario considerar las equivalencias de los términos en español e inglés (carril por lane, tarjeta o carta por card y tablero por board).

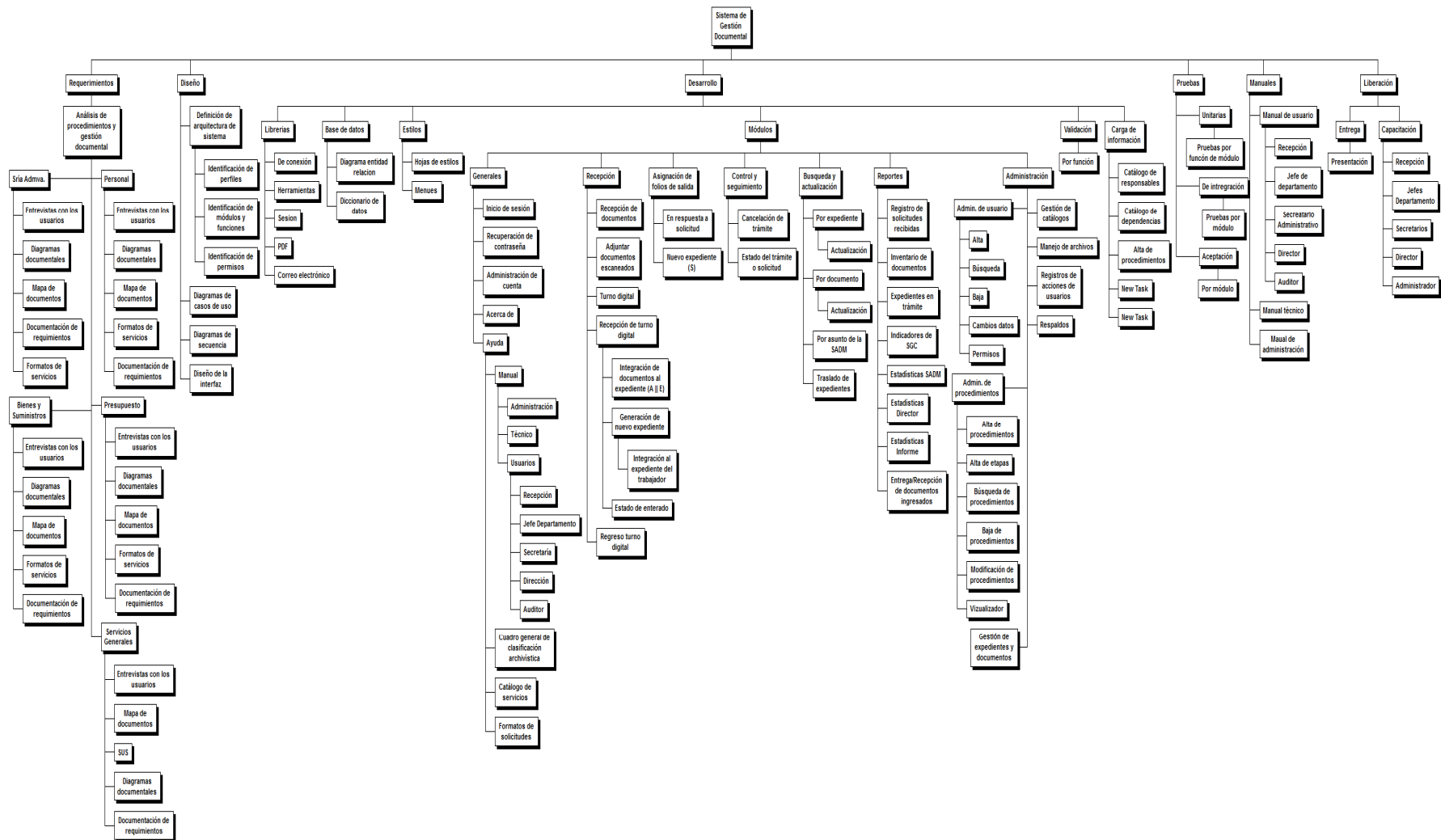
Clase: ProjectToLeanKit		
Atributos		
Métodos	main(String [] args)	Método principal del programa

Clase: JsonManager		
Métodos	JSONPrettyPrint(String JSONString) : String	El método recibe un objeto JSON en forma de cadena y lo devuelve de una forma legible mediante identaciones. Se utiliza básicamente para depuración.
	getLanes(JSONObject boardIdentifiersInJSONFormat) : Map	El método recibe los identificadores de un determinado tablero en formato JSONObject y obtiene los carriles válidos en un objeto Map<String NombreLane, Lane lane>. Los nombres de los carriles o lanes están conformados por la concatenación del nombre del Lane y todos su Lanes padres separados por dos puntos, por ejemplo "persona:profesor:profesorEmerito".
	getBoardsId(JSONObject boardInJSONFormat)	El método recibe la información de los boards válidos para una cuenta determinada en formato JSONObject y obtiene los boards válidos en un objeto Map<String NombreBoard,Map board>
	getCardTypes(JSONObject boardIdentifiersInJSONFormat)	El método recibe los identificadores de un determinado board en formato JSONObject y obtiene los CardTypes válidos en un objeto Map<String NombreCardType,Map cardType>
	getLanes(JSONObject boardIdentifiersInJSONFormat)	El método recibe los identificadores de un determinado board en formato JSONObject y obtiene los Lanes válidos en un objeto Map<String NombreLane,Lane lane>. Los nombres de los Lanes están conformados por la concatenación del nombre del Lane y todos su Lanes padres separados por dos puntos, por ejemplo "persona:profesor".
	getPriorities(JSONObject boardIdentifiersInJSONFormat)	El método recibe los identificadores de un determinado board en formato JSONObject y obtiene las prioridades válidas en un objeto Map<String NombrePrioridad,Map prioridad>

Clase: HttpRequest		
Atributos	String user	Cadena que almacena el usuario utilizado en la aplicación <i>LeanKit</i> .
	String password	Cadena que almacena la contraseña utilizada en la aplicación <i>LeanKit</i> .
Métodos	addCards(String domain,String boardID, JSONArray cardsInJSONArrayFormat)	Recibe el dominio, el ID del tablero a donde se desean subir las cards y las cards en un formato JSONArray. El método determina cuantas peticiones post se debe realizar, ya que la aplicación permite agregar no más de 100 cartas por request. Por cada petición imprime en consola la respuesta del servidor <i>LeanKit</i> en formato JSONObject.
	getBoards(String domain) : JSONObject	El método realiza una petición GET al servidor de <i>LeanKit</i> que devuelve información en formato JSONObject de los distintos tableros asociados a la cuenta, tal como nombre y su ID.
	sendGet(String domain, String boardID, String action) : JSONObject	El método realiza una petición GET al servidor de <i>LeanKit</i> que devuelve información en formato JSONObject. La información recibida dependerá del parámetro action, por ejemplo “getCards”, “getBoardsIdentifiers”, “getCards”. Se trata de un constructor genérico de peticiones GET.
	sendPost(String domain, String boardID, String action, JSONArray jsonArray) : JSONObject	El método realiza una petición POST al servidor de <i>LeanKit</i> que devuelve información en formato JSONObject. La información recibida dependerá del parámetro action, por ejemplo “addCards” o “deleteCards”. A diferencia del método sendGet, sendPost requiere de información adicional adjunta a la petición POST, que en este caso se proporciona en formato JSONArray. Se trata de un constructor genérico de peticiones POST.
	sendPost(String domain, String boardID, String action, JSONObject jsonObject) : JSONObject	Se trata de una sobrecarga del método anterior. En este caso el método recibe información en formato JSONObject.
Constructor	HttpRequest(String user, String password)	Constructor de la clase que configura los parámetros user y password.

Clase: MppH		
Atributos	MultiValueMap(String etapa, List(tasks)) tareasHijas	MultiValueMap utilizado de manera global. Al final de la ejecución el método readMPP este objeto contendrá las tareas contenidas en el archivo mpp identificadas por la etapa a la que pertenecen. Se trata de un MultiValueMap porque para distintas una misma etapa consta de varias Tasks.
Métodos	ListHierarchy(ProjectFile file, int level, String trace)	Este método recibe un objeto ProjectFile (definido en la API MPXJ) un entero que define el nivel inicial de las tareas (normalmente 0) y una cadena inicial que representa la etapa de cada tarea (normalmente se proporciona una cadena vacía). El método itera en cada una de las tareas del primer nivel, y por cada tarea obtiene sus tareas hijas y manda a llamar al método sobrecargado ListHierarchy(Task task, String ident, int level String trace)
	ListHierarchy(Task task, String ident, int level String trace)	Este método es una sobrecarga del método anterior. Itera por todas las tasks hijas de la task que recibe y por cada una se llama recursivamente. Durante su ejecución, si encuentra task que no tengan task hijas, las almacena en el objeto tareasHijas
	ReadMPP(String path, String fileName) : MultiValueMap	Este método recibe la ruta y el nombre del archivo mpp a utilizar y crea un objeto ProjectFile que contiene toda la información del archivo mpp. Posteriormente llama al método ListHierarchy que recorrerá todas las tareas recursivamente en busca de task que no tengan task hijas.

C. ESTRUCTURA DE DESGLOSE DE TRABAJO PARA EL SIGEDAA



BIBLIOGRAFÍA

(2013). A Guide to the Project Management Body of Knowledge (5th Ed.). Project Management Institute.

Ackoff, Russell L. (1980). Un concepto de planeación de empresas. México: Limusa.

Amaru C. (2009). Fundamentos de la administración. Teoría general y procesos administrativos. Pearson Prentice Hall.

Borges H. (2015). Tecnologías de la información para la administración de proyectos en una empresa consultora. Tesis del Posgrado de Ingeniería. UNAM.

Capps M. (2011, Sept). A-round ahead? Leankit's Hefley visualizes future for local software venture. Venture Nashville Connections. Recuperado el 23 de septiembre de 2015, de <http://venturenashville.com/a-round-ahead-leankits-hefley-visualizes-future-for-local-software-venture-cms-678>

Estrada M. (2013). Adaptación del PMI para proyectos de pequeña y mediana escala. Tesis del Posgrado de Ingeniería. UNAM.

(2004). Gestión de proyectos. Serie Harvard Business Essentials. Ed. Deusto.

Herrera H., Muzquiz C. Venegas L. (2014) Apuntes para el Diplomado administración de proyectos, 8va. generación. DGTIC, UNAM.

Kanban roadmap: how to get started in 5 steps. LeanKit. Recuperado 25 de septiembre de 2015, de <http://info.leankit.com/kanban-roadmap>.

Kerzner, Harold. (2009). Project Management: A systems approach to planning, Scheduling, and Controlling (10th Ed.). EUA: Wiley.

Kniberg H., Skarin M. (2010). Kanban and Scrum: making the most of both, de la serie InfoQ Enterprise Software Development. USA: C4Media Inc. [Versión electrónica]. Recuperado el 25 de septiembre de 2015, de www.agileinnovation.eu/wordpress/wp-content/uploads/2010/09/KanbanAndScrum_MakingTheMostOfBoth.pdf

Martínez, B. (2003). Estructuración de un plan de acción. Tesis del Posgrado de Ingeniería. UNAM.

Mell P. Grance T. (2011). The NIST Definition of Cloud Computing. National Institute of Standards and Technology. Recuperado el 23 de septiembre de 2015, de <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>.

Peter. (2011). Behind LeanKitKanban – Jon Terry. Agile Scout. Recuperado el 23 de septiembre de 2015, de <http://agilescout.com/behind-leankitkanban-jon-terry>.

Ríos J. (2014). 2do Informe de actividades. Instituto de Investigaciones Bibliotecológicas y de la Información, UNAM.

Ríos J. (2012). Plan de desarrollo 2012-2016. Instituto de Investigaciones Bibliotecológicas y de la Información, UNAM.

Rodrigues A., Bowers J. (1996, Summer). System dynamics in projects management: a comparative analysis with traditional methods. System Dynamics Review. Recuperado el 23 de septiembre de 2015, de [http://onlinelibrary.wiley.com/doi/10.1002/\(SICI\)1099-1727\(199622\)12:2<121::AID-SDR99>3.0.CO;2-X/epdf](http://onlinelibrary.wiley.com/doi/10.1002/(SICI)1099-1727(199622)12:2<121::AID-SDR99>3.0.CO;2-X/epdf).

Rodríguez K. Las TIC como apoyo al proceso de enseñanza-aprendizaje en Bibliotecología. Escuela de Bibliotecología, Documentación e Información. Universidad Nacional de Costa Rica. Recuperado el 23 de septiembre de 2015, de http://iibi.unam.mx/publicaciones/21/calidad_educacion_superior_al_Karla_Rodriguez_Salas.html.

Yung L. (2001). Gestione bien sus proyectos. Gedisa.

SITIOS WEB CONSULTADOS

- www.axosoft.com/product
- <http://leankit.com>
- <https://products.office.com/es-mx/project/project-and-portfolio-management-software>
- <http://project-management.zone>
- www.oracle.com/applications/primavera/solutions/products.html
- www.telerik.com
- <http://trello.com>

PARA EL DESARROLLO DEL BRIDGE

- <https://support.leankit.com/hc/en-us/sections/200668393-LeanKit-API-Application-Programming-Interface>
- <http://mpxj.sourceforge.net/howto-start.html>
- <http://programmers.stackexchange.com>
- <http://sourceforge.net/projects/mppviewer>
- <http://commons.apache.org/proper/commons-collections/javadocs/api-3.2.1/org/apache/commons/collections/map/MultiValueMap.html>
- <https://github.com/google/gson>

PARA LA REALIZACIÓN DEL GLOSARIO

- <http://www.rae.es>
- <https://es.wikipedia.org>