

Capítulo 3

Desarrollo del SIGP

3.1 Objetivo del sistema informático para la gestión de proyectos

Como ya se ha mencionado, el SIGP es una aplicación web usada para administrar todos los proyectos del área de Performance Services. Dentro de la empresa este elemento de software es considerado de carácter altamente crítico para sus procesos de mejora continua y ventas, pues en ella los clientes visualizan y manejan información relacionada a las finanzas, seguimiento de partes, gestión de entregables, entre otros. A continuación, se enlista de manera general las áreas que la aplicación comprende:

- Personalización de proyectos
- Almacenamiento de direcciones físicas relacionadas a la entrega del producto
- Calendarización y seguimiento de la facturación de partes
- Gestión de la documentación relacionada a los proyectos
- Registro de los requerimientos comerciales de proyectos de turbinas
- Administración de riesgos
- Manejo de inventarios de componentes físicos
- Registro de base de conocimiento para las mejores prácticas de los proyectos
- Módulo de reportes específicos de cada proyecto
- Sección de reportes del negocio
- Conjunto de reportes generales
- Seguimiento del estado de pérdidas y ganancias de cada proyecto
- Registro de documentos de créditos
- Seguimiento del flujo de aprobaciones para liberación de contrato final del proyecto
- Integración con diversos almacenes de datos y ERPs

Debido a lo anterior, se aprecia que el alcance del SIGP dentro de la empresa es muy amplio. Por otro lado, es de los sistemas más utilizados por los clientes y que agregan de manera importante valor a la compañía.

Con el objetivo de incrementar las ventas y reducir gastos y tiempos de entrega, surge la necesidad de efectuar cambios de relevancia en los procesos internos y externos, así como reestructuraciones organizacionales. Por estas razones, la aplicación necesita adaptarse a los procesos y objetivos de productividad de manera completa.

Otro de los factores claves que dieron pie a la generación de un nuevo producto de software, fue el relacionado a las fallas del sistema. La primera versión del SIGP fue desarrollada en el año 2003. A partir de entonces se actualizó el sistema de manera recurrente mediante mantenimiento adaptativo cada año, hasta el 2006, cuando la versión 4 se hizo disponible. Estas actualizaciones anuales atendían constantemente los requerimientos emergentes del negocio y corregían fallas que el sistema pudiera tener. No así, en el periodo comprendido entre 2006 y 2009, ninguna ventana importante de mantenimiento o mejora fue implementada. El efecto de 3 años sin actualizaciones o correcciones impactó de manera alarmante las operaciones directamente dependientes del funcionamiento apropiado de la aplicación, provocando inconsistencias entre los procesos de negocios en curso y la lógica aplicativa del sistema. Asimismo, la infraestructura mostró evidencia de fallas después del largo tiempo sin el adecuado mantenimiento. En la figura 3.1 se muestra la representación gráfica de los datos históricos relacionados a las fallas generales del sistema, por trimestre en el periodo comprendido entre 2007 y 2009, previas a la instalación de la versión desarrollada en este trabajo de tesis. Posteriormente, se

mostrarán las estadísticas después de la implementación del nuevo sistema para analizar los efectos y resultados del proyecto.

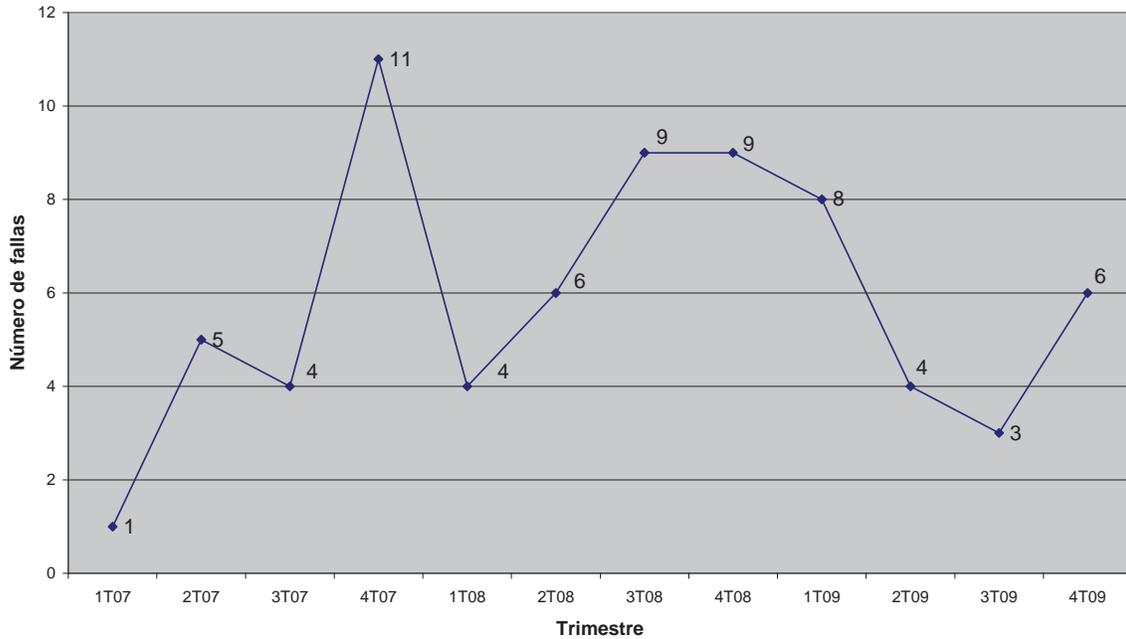


Figura 3.1 Fallas generales trimestrales antes del desarrollo

Se puede apreciar en la figura 3.1 que la disponibilidad del sistema sufrió numerosas interrupciones durante el periodo en el que hubo ausencia de mantenimiento y mejoras. La figura 3.2 representa las fallas por año, antes de que el nuevo sistema fuera liberado. Los niveles que se presentan son muy altos considerando la relevancia de esta aplicación para la compañía. En consecuencia, los usuarios del sistema externaron su alta inconformidad, pues el sistema perdía rápidamente confiabilidad y las ventas de proyectos tendían a verse afectados negativamente.

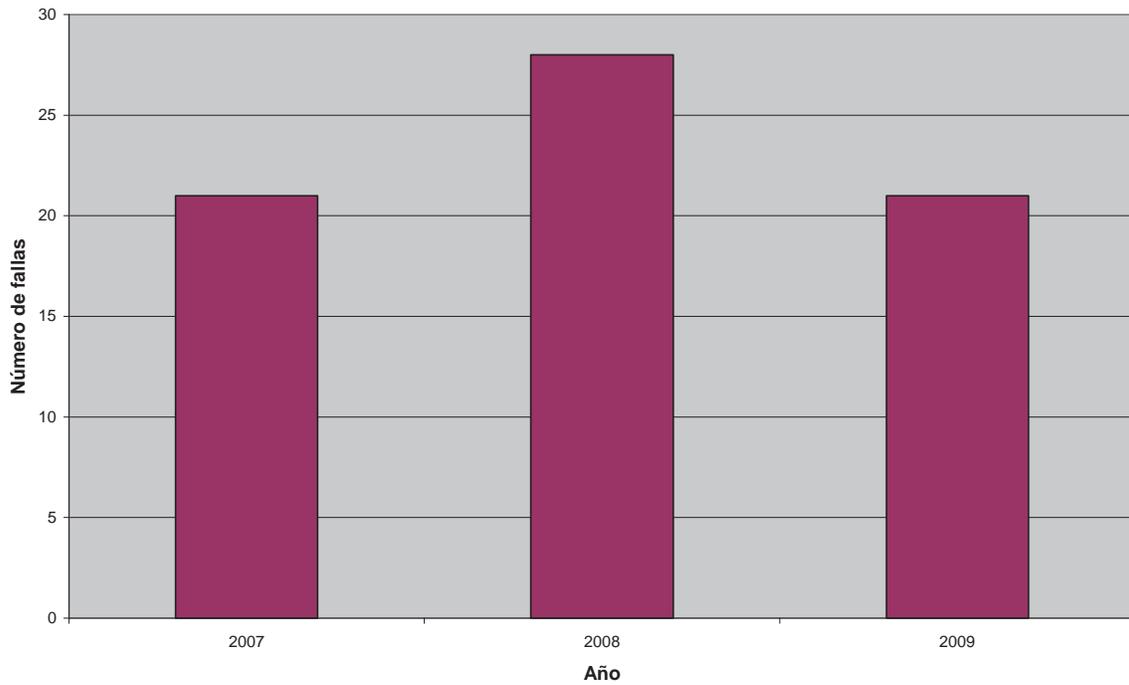


Figura 3.2 Fallas generales por año

Por los factores anteriormente mencionados, el sistema necesita ser actualizado para evitar fallas generales en la aplicación que interrumpan el servicio de la misma. De igual manera el SIGP debe apegarse a los procesos actuales y volver a ser una herramienta de alto nivel de productividad que responda a los requerimientos del cliente. Por lo tanto, para cumplir con estos objetivos, un nuevo grupo de requerimientos, mejoras y correcciones fueron identificados e implementados como parte del proyecto del SIGP versión 5. En resumen, los requerimientos solicitados deben cubrir los siguientes objetivos:

- Mejorar la seguridad de la aplicación
- Asegurar la precisión de los datos en el sistema
- Tener interfaces sencillas para usuarios finales
- Agregar controles apegados a la lógica de negocio para tener un flujo adecuado de los proyectos manipulados en el SIGP

- Sincronizar los datos entre cada fase de los proyectos registrados en el SGIP
- Corregir y mejorar las integraciones que el SIGP tiene con los sistemas externos
- Aumentar la disponibilidad del SIGP

El propósito del proyecto fue desarrollar una nueva versión de la aplicación que atendiera estas necesidades de adaptación a los nuevos procesos de la organización, así como también mejorar el funcionamiento, disponibilidad y desempeño de la misma.

3.2 Expectativas y entregables del proyecto

Como parte de un proyecto de desarrollo como éste, se consideran como entregables los siguientes elementos:

- Diseño de la nueva versión. Una vez recibidos los requerimientos, el equipo de desarrollo discute y propone el diseño para la solución y futura implementación de los requerimientos, generando así la documentación asociada.
- Desarrollo de la nueva versión. Después de contar con un diseño de solución de los requerimientos se codifica su implementación. Los archivos de código generados forman parte de los entregables que serán otorgados al cliente.
- Pruebas unitarias y de aceptación de usuario. Al terminar la codificación se hacen pruebas a cada elemento modificado o creado para hacer las correcciones pertinentes y reducir los defectos. Los resultados de las pruebas deben consolidarse y validarse con el cliente para resguardarlos como evidencia de su ejecución.
- Cambios en configuración. Si así lo amerita, también se considera cualquier cambio de configuración de la aplicación, tales como actualizaciones del servidor de aplicaciones, cambios en el framework de programación, movimientos de la aplicación a otro servidor,

etc. De igual manera debe generarse documentación al respecto para que, de ser necesario, sirva de referencia futura.

- Soporte en la entrega al ambiente productivo. Una vez que la fecha de entrega sea definida y la aplicación esté lista y validada, se procede a reflejar los cambios en contenedor de aplicaciones. Para ello se interactúa con equipos de servidores Unix, Servicios Web, Base de Datos y administración de accesos. El equipo ayuda a que se instale correctamente la aplicación, así como también se involucra en la ejecución de scripts para adaptar la base de datos a los requerimientos nuevos.

- Soporte de post-producción. Para mantener los niveles de calidad, después de la entrega de la aplicación se tiene un periodo de soporte a post-producción o garantía en el que el equipo debe estar pendiente, por cualquier comportamiento inusual en la aplicación, para poder analizarlo al momento y revisar si es consecuencia del desarrollo de la nueva versión o si es un factor ajeno.

- Transferencia de conocimiento al equipo de soporte y mantenimiento del sistema. Parte de los acuerdos generados en el proyecto consiste en transferir el conocimiento técnico y funcional de los cambios implementados para que el equipo encargado de mantener el sistema pueda realizar sus actividades, de acuerdo a la nueva versión y evitar perder tiempo en investigaciones innecesarias.

- Asimismo, se modifica o genera la documentación de la aplicación para que al momento de necesitar alguna referencia, se estén revisando los documentos actualizados acordes con la nueva versión. Se incluyen documentos tales como:

- Documento de especificaciones funcionales
- Documento de interfaz de usuario

- Documento de arquitectura de la aplicación
- Documento de especificaciones técnicas del diseño
- Plan de pruebas
- Documento de los casos de prueba
- Guía de soporte a producción
- Guía de configuración e instalación

3.3 Aplicación del modelo en espiral

A continuación se presenta la aplicación del modelo en espiral al desarrollo del nuevo sistema, organizado por los ciclos o vueltas por los que pasó desde sus etapas iniciales, donde se analizan elementos como la factibilidad del proyecto (considerada como vuelta 0, pues es preliminar al desarrollo del sistema), la generación del plan de trabajo, hasta llegar a la entrega final del producto listo para continuar con su etapa de mantenimiento.

3.3.1 Vuelta 0: Estudio de factibilidad

3.3.1.1 Fase de planificación

Para la primera iteración del modelo se definió como objetivo lo siguiente:

Incrementar la productividad y reducir los problemas recurrentes considerando que el número de usuarios de la aplicación se incrementaría considerablemente.

Las principales restricciones identificadas durante esta fase fueron el elevado costo que involucraba cumplir con dicho objetivo mediante la generación de un nuevo sistema capaz de cubrir las necesidades actuales y nuevas, así como también modificar la infraestructura actual para alojar al nuevo sistema. Por otro lado, de acuerdo a la experiencia del negocio y a la antigüedad tanto del sistema actual como de los usuarios, se recibieron muestras

importantes de resistencia al cambio por medio de comunicados para conservar el sistema actual y no modificar ese esquema.

Considerando estas restricciones se formularon las siguientes alternativas:

- Realizar mejoras menores para las principales necesidades y así evitar un cambio tan grande
- Generar una nueva versión del sistema para lograr un cambio relevante e incrementar la productividad del área de Performance Services de manera substancial.
- No realizar ningún cambio y permanecer con el sistema actual, evitando costos y problemas de resistencia al cambio.

3.3.1.2 Fase de análisis de riesgo

Poniendo en perspectiva el panorama de la situación, se identificaron los siguientes riesgos que podían comprometer el cumplimiento del objetivo:

- Existía una falta de profundidad en los detalles respecto a las necesidades del negocio
- No había un consenso respecto a la definición adecuada de los requerimientos, pues la mayoría de los usuarios principales estipulaban que sus requerimientos eran los más críticos y de alta prioridad
- Permanecía la incertidumbre respecto al tiempo de retorno de la inversión, pues el cliente no estaba seguro si valía la pena tanto esfuerzo, contra el tiempo en el que los beneficios serían visibles y medibles.

3.3.1.3 Fase de ingeniería

Para atender las fuentes de riesgo y la problemática involucrada en el cumplimiento de los objetivos se realizaron las siguientes actividades:

- Encuestas, análisis y reuniones con el equipo funcional (cliente) para determinar de manera concreta las necesidades del negocio, así como también sus quejas y peticiones.
- Se tuvieron reuniones con los equipos de soporte e infraestructura para generar el perfil técnico de la aplicación actual en términos de versión de componentes, tamaño de la base de datos, llamadas de servicio frecuentes, tipos de fallas frecuentes con sus respectivos RCAs (“Root-Cause Analysis”, es decir análisis de la causa raíz). Parte del perfil técnico resultante fue el siguiente:
 - o Servidor de base de datos: Oracle 9.2.0.4
 - o Servidor de aplicaciones: JBoss 3.2.1
 - o Versión de Framework: Casper 3.9
 - o Librerías de terceros: fop205.jar, jdom.jar y jxl.jar
 - o Versión de Java: J2EE 1.4
 - o Desarrollado y contenido en ambiente compartido ubicado en la ciudad de Alpharetta, Georgia.
- Se analizó la posibilidad de que el equipo de soporte pudiera manejar las nuevas modificaciones sin tener que crear un equipo de desarrollo nuevo.

3.3.1.4 Fase de evaluación

Una vez finalizadas las actividades de ingeniería, se obtuvieron los siguientes resultados:

- Después de revisiones con el equipo funcional y usuarios clave se generó el documento llamado Business Requirements Document (BRD), que contenía la lista de requerimientos generados por el cliente y aprobados por los dueños del sistema. Asimismo, estipulaba la visión y justificación de los requerimientos después de un consenso entre el equipo funcional y los usuarios.
- Se llegó a la conclusión de que la versión actual del sistema ya no cumplía con las necesidades del negocio. A su vez, un grupo reducido de requerimientos tampoco podían cubrir la totalidad de las nuevas necesidades y, por tanto, el equipo de soporte no era capaz de ejecutar un proyecto de esta magnitud por su propia cuenta. Como se estipula en los acuerdos de nivel de servicio, el equipo de soporte sólo dispone de 280 horas efectivas para desarrollo de mejoras, mismas que son por mucho insuficientes para el desarrollo de la nueva versión.
- Finalmente, se toma la decisión de crear un nuevo proyecto de desarrollo para producir una nueva versión del sistema actual.

En esta primera vuelta el principal entregable resultó ser el documento BRD y la aprobación del proyecto de desarrollo. Para el cliente el siguiente paso consistió en desempeñar las tareas necesarias para conseguir un proveedor que pudiera desarrollar el proyecto.

3.3.2 Vuelta 1: Propuesta general al cliente

3.3.2.1 Fase de planificación

Tomando como base el documento BRD que el cliente proporcionó, el siguiente objetivo fue proporcionar una respuesta oportuna y apropiada y hacer una propuesta basada en estimaciones de tiempo y costo de los requerimientos a desarrollar. El proceso de selección de proveedor fue definido por el cliente y la idea era ser seleccionados para trabajar en el nuevo sistema.

3.3.2.2 Fase de análisis de riesgo

El mayor riesgo identificado en esta etapa fue la posibilidad de generar una estimación inadecuada de los requerimientos. La causa probable de una mala estimación podría ser debido a preguntas técnicas y funcionales no aclaradas.

Otro riesgo a atender fue el considerar la posibilidad de no contar con los recursos y capacidades necesarias para desarrollar el proyecto, en caso de ser elegidos como el proveedor.

3.3.2.3 Fase de ingeniería

Mediante un análisis detallado de cada requerimiento se generaron los documentos respectivos de estimación en tiempo y costo. Esta tarea la llevó a cabo un equipo conformado por un analista, el equipo de soporte de la aplicación actual, un BRM (Business Relationship Manager), aprovechando un par de sesiones de preguntas y respuestas proporcionadas por el cliente en las que cada posible proveedor tenía oportunidad de aclarar dudas. Los roles involucrados en las sesiones fueron los siguientes:

- BRM
- Advance Solutions Engineer
- Experto de aplicación
- Equipo de soporte
- Dueño de aplicación (conocido como IM Owner)
- Líderes funcionales
- Analista de negocios

Por otro lado, se trabajó con el área de recursos humanos de la compañía para conocer la situación de la misma en cuanto a recursos humanos disponibles.

3.3.2.4 Fase de evaluación

Una vez finalizadas las estimaciones, se generó un documento con la propuesta para el cliente, la cual incluía una versión preliminar del plan del proyecto. Dicho documento fue enviado al área de Performance Services para su evaluación y comparación con las demás propuestas.

En la estimación, se propuso la siguiente estructura en el equipo de desarrollo:

- Un analista certificado Black belt en Six Sigma
- Un BRM
- Un Operational Leader
- Un Project Leader
- Tres desarrolladores
- Tres testers

Respecto a los recursos humanos disponibles, se concluyó que se contaba con la suficiente capacidad para abordar el proyecto. La decisión del cliente fue que la compañía debía hacerse cargo del proyecto.

Los principales entregables de esta etapa fueron la estimación y propuesta al cliente, así como también el plan de trabajo de alto nivel. Los siguientes pasos a trabajar fueron obtener los detalles actuales de la aplicación, tales como documentación, localización de servidores, detalles de acceso, entre otros, así como también programar reuniones para obtener el máximo detalle de la aplicación y sus requerimientos que resultaran en el diseño de la misma.

3.3.3 Vuelta 2: Definición detallada de requerimientos y documentación

3.3.3.1 Fase de planificación

Para esta iteración del modelo en espiral, el objetivo se centra en obtener de cierta manera el primer prototipo del sistema en papel, esto es, mediante la creación de la documentación relacionada con el desarrollo. Las restricciones encontradas radicaron en la falta del nivel de detalle adecuado para generar los documentos de manera precisa. Dado que ya existía la documentación del sistema en uso, se contaba con la alternativa de apoyarse en la misma para generar la nueva documentación o crearla desde cero.

3.3.3.2 Fase de análisis de riesgo

Dependiendo de la alternativa elegida se definían diferentes riesgos. Por un lado, la documentación existente podía servir de apoyo para agilizar el proceso de análisis y escritura de los nuevos documentos, pero se sabía que la misma estaba desactualizada e incompleta. Por el otro lado, generar la documentación desde cero involucraba mayor

esfuerzo y tiempo que podía retrasar los entregables y generar insatisfacción por incumplimiento de las fechas prometidas.

Finalmente, un riesgo que frecuentemente está latente es que el cliente puede cambiar de parecer en cualquier momento, ya sea modificando definiciones de requerimientos ya establecidas o agregando mayor alcance al proyecto de desarrollo con nuevos requerimientos.

3.3.3.3 Fase de ingeniería

Con el apoyo y experiencia del equipo de soporte se pudieron basar los nuevos documentos en los ya existentes, tomando reservas de la existente desactualización de estos últimos. Aún así, la retroalimentación con el equipo de soporte sirvió para identificar los errores y no pasarlos a los nuevos documentos

3.3.3.4 Fase de evaluación

La evaluación de los documentos fue inicialmente manejada internamente en el equipo de desarrollo antes de enviarlos a los dueños de la aplicación. Contamos con el apoyo del Business analyst para hacer una evaluación preliminar y proporcionar retroalimentación respecto a la calidad de los documentos en cuanto a contenido y forma. Algunos errores fueron encontrados y corregidos, antes de enviar finalmente los documentos resultantes listados a continuación:

- Documento de especificación funcional
- Documentos de la interfaz de usuario
- Documentos de requerimientos

3.3.4 Vuelta 3: Diseño del sistema y primera liberación de capa de presentación

3.3.4.1 Fase de planificación

Una vez que los requerimientos han sido aclarados, documentados y validados tanto por el cliente como por el proveedor, la siguiente iteración del modelo se encargó del diseño de solución de los requerimientos. De igual manera, esta vuelta de la espiral fue utilizada para generar una versión de la aplicación, donde se desarrollará la capa de presentación del sistema sin contar con las funciones requeridas pretendiendo mostrar al cliente un demo del sistema.

Las principales restricciones para lograrlo fueron las siguientes:

- Los desarrolladores necesarios para el proyecto aún no estaban listos, de acuerdo al informe del área de recursos humanos.
- El ambiente de pruebas para el nuevo sistema aún no estaba definido, por lo que no había lugar en la infraestructura del cliente para alojar el prototipo de la aplicación.
- Existía una base de datos de desarrollo, sin embargo los datos no eran recientes y no reflejaban la realidad del sistema actual.
- Se carecía de capital de trabajo para comenzar con el desarrollo, pues el cliente no tenía planeado financiar el proyecto hasta después de un mes aproximadamente.

Las alternativas de solución a las anteriores restricciones se muestran a continuación:

- No crear la versión preliminar y esperar a que una versión completamente funcional estuviera lista.
- Crear la versión preliminar enfocando esfuerzo adicional para conseguir un ambiente de pruebas adecuado y conseguir los desarrolladores necesarios a la brevedad.

- Coordinarse con el cliente para obtener el capital de trabajo mínimo para iniciar las operaciones del proyecto.

3.3.4.2 Fase de análisis de riesgo

Por lo anteriormente mencionado, los riesgos radicaban en la posibilidad de retraso en la entrega del proyecto final, que la inversión no proporcionara la relación costo, beneficio y tiempo esperado, o que el cliente confundiera la versión prototipo con una versión funcional sino se aclaraban los detalles del mismo.

3.3.4.3 Fase de ingeniería

En lo que respecta a la ingeniería de este ciclo se realizó una revisión detallada de los documentos de análisis existentes en la versión anterior del sistema. El primer objetivo fue identificar las áreas y funcionalidades existentes relacionadas con los nuevos requerimientos para tener un mejor entendimiento y complementar el análisis del código fuente. De esta manera, se buscó apearse al diseño de la arquitectura general de la aplicación en sus anteriores versiones, dado que el código existente estaba bien organizado y estructurado.

Para el proyecto en cuestión, hubo una reingeniería de procesos del lado del negocio, lo que dio como resultado la generación de nuevas funcionalidades en la aplicación web y la reestructuración y plan de mejora de otras ya existentes. Asimismo, se ejecutó un plan de análisis detallado, tanto en el código fuente como en la base de datos, para entender cada funcionalidad que se requería modificar y obtener una mejor visión del diseño de los requerimientos relacionados a los módulos nuevos.

Un ejemplo del análisis desempeñado se muestra en la figura 3.3, donde se parte de la pantalla mostrada en el browser para conocer los componentes involucrados mediante la dirección URL asociada. Siguiendo el modelo MVC, se puede saber que la interacción puede conformarse tanto por archivos de comandos tipo Java como los relacionados a la presentación del sistema al usuario (archivos JSP, Javascript o HTML). Al conocer el detalle de los comandos asociados se puede obtener el manejador de contexto, cuya función es la de servir como el contenedor de objetos y datos significativos para la interacción. También son identificados los servicios involucrados en la funcionalidad, utilizados para mantener una apropiada organización de las operaciones que definen la lógica del negocio en la aplicación y que son previas a la extracción de datos.



Figura 3.3 Seguimiento de análisis de interacciones del sistema

Posteriormente se ubican objetos de acceso a datos, los cuales se encargan de generar una interfaz apropiada entre la aplicación web y la base de datos mediante la configuración de la conexión a base de datos, la especificación de búsquedas apropiadas y la definición de los parámetros asociados a las consultas. En lo que respecta a la base de datos, se desempeña otro análisis similar para comprender los datos, tablas, triggers, vistas, procedimientos y paquetes que la aplicación necesita para brindar el servicio deseado. Como se mencionó previamente, esta aplicación tiene varias integraciones con otros sistemas, por lo que también necesitamos saber el mapeo entre los objetos fuente y destino la lógica de extracción. Dentro de las tecnologías y lenguajes utilizados en este proyecto se encuentran Java, JSP, XML, HTML, SQL, PL/SQL, entre otros.

Siguiendo la lógica de la interacción se conocen las operaciones, datos y procesos involucrados para reunir la información simplificada y relacionarla con el nuevo requerimiento.

Finalmente, el ejercicio de análisis ayudó a definir el diseño de cada uno de los requerimientos para alinearlos lo mejor posible a la estructura existente. El propósito fue mantener en todo momento la estructura para que el código no se desgastara ni se desorganizara.

3.3.4.4 Fase de evaluación

Al final de la fase de ingeniería se generó el prototipo junto con la siguiente documentación:

- Documento de arquitectura del sistema
- Especificación técnica del diseño del sistema

Cuando los usuarios evaluaron el prototipo generaron una lista de preguntas, errores encontrados, comentarios y sugerencias que fueron atendidos en la siguiente iteración del modelo.

3.3.4 Vuelta 4: Construcción de la parte funcional y pruebas

3.3.4.1 Fase de planificación

Con la retroalimentación del cliente los objetivos de esta vuelta se enfocaron a dar respuesta a las inquietudes y sugerencias del cliente, así como también en corregir los errores encontrados. En términos de reglas de negocio, el esfuerzo también se enfocó a su implementación en el sistema, para así generar una versión completamente funcional y que pudiera ser sometida a un proceso de pruebas.

Asimismo, las restricciones encontradas fueron relacionadas a los errores, dado que podían ser lo suficientemente relevantes como para consumir demasiado tiempo en su corrección. Respecto a los recursos humanos disponibles existía una limitante importante, pues en este punto la rotación del personal dentro del equipo fue grande y un nuevo proceso de adquisición de recursos se inició.

Finalmente, para la aplicación activa, previa a nuestro sistema en desarrollo, se generaron varios requerimientos de mantenimiento correctivo, pues los usuarios reportaron defectos de suma repercusión que no podían esperar a la liberación del nuevo sistema.

Las alternativas de solución a la problemática de la etapa actual fueron las siguientes:

- Debido a la magnitud del proyecto y a la cantidad de requerimientos se contempló la posibilidad de dividir el proyecto en dos liberaciones para atender primero los requerimientos de mayor prioridad, incluyendo aquellos relacionados con los

requerimientos de mantenimiento correctivo solicitados por los usuarios como urgentes.

- Continuar con el proyecto bajo el esquema de una sola liberación del sistema y que el equipo de soporte trabajara a la par para atender los problemas actuales urgentes.
- Atender primero los problemas urgentes para después continuar con el desarrollo de los requerimientos del proyecto.

3.3.4.2 Fase de análisis de riesgo

Durante esta vuelta, a nuestro modelo en espiral se generó una lista de riesgos relacionada a las restricciones y alternativas mencionadas en la fase anterior más las que los que tuvieron que ver con la administración de la infraestructura de base de datos, como se explica a continuación:

- El riesgo principal fue el no contar con los recursos humanos necesarios para continuar con el proyecto, pues podía retrasar gravemente el calendario de entregables.
- Los requerimientos de mantenimiento correctivo eran de tamaño considerable, ya que podían interferir con las actividades de desarrollo, considerando que los ambientes de desarrollo y calidad eran únicos y tanto las pruebas de desarrollo, como las de correcciones se ejecutaban en el mismo ambiente.
- El equipo de infraestructura por parte del cliente notificó que en las siguientes semanas las versiones de base de datos debían ser actualizadas de la versión 9i Oracle a la 10g del mismo proveedor.

3.3.4.3 Fase de ingeniería

Atendiendo a los requerimientos de documentación del sistema se generaron los siguientes entregables:

- Guía de configuración y despliegue del sistema
- Guía de soporte a producción

En cuanto a las actividades necesarias para llevar a cabo las pruebas pertinentes, se inició con la generación del documento de plan de pruebas en el que se definen las actividades requeridas para tener un ciclo de pruebas apropiado. Los documentos generados se listan a continuación:

- Documento de plan de trabajo
- Matriz de pruebas
- Documento de casos de pruebas
- Consolidado de resultados

3.3.4.4 Fase de evaluación

Al concluir la etapa de ingeniería, el analista de negocios procedió a realizar las pruebas correspondientes, de las cuales surgió una larga lista de errores identificados principalmente relacionados con la lógica de las reglas del negocio, así como también con el alcance de los requerimientos. Debido a la gran cantidad de errores identificados por el analista de negocio, se aplicó un proceso integral de validación y verificación, incluyendo al cliente a través de su equipo funcional de aplicación para descartar falsos errores y tener una lista real antes de planear los esfuerzos de la siguiente vuelta.

3.3.5 Vuelta 5: Alcance agregado y transición al nuevo sistema

3.3.5.1 Fase de planificación

Como resultado del ejercicio de validación y verificación en la vuelta anterior, se identificó un número considerable de falsos errores, dado que se trataba de alcance agregado no especificado apropiadamente desde el inicio del proyecto. A causa de que el cliente suponía la implementación de estos requerimientos no contemplados inicialmente, procedió a calificarlos como errores. Después de las aclaraciones correspondientes, se genera un acuerdo para agregar los nuevos requerimientos al proyecto ajustando el plan inicial.

Por estas razones, los objetivos de esta nueva vuelta fueron los siguientes:

- Corregir los errores reales de la liberación anterior para cumplir con lo requerido por el cliente
- Agregar los nuevos requerimientos al proyecto de tal manera que exista el mínimo impacto en el desarrollo.

Las principales restricciones de esta vuelta fueron relacionadas a los recursos humanos disponibles, pues ahora se trataba de más requerimientos a desarrollar en el mismo tiempo establecido. En consecuencia, el presupuesto disponible era limitado para contratar más desarrolladores. Finalmente, el tiempo permanecía como limitante, pues a pesar del alcance agregado la fecha de entrega era inamovible.

3.3.5.2 Fase de análisis de riesgo

El hecho de incorporar nuevos requerimientos a un proyecto en curso genera diversas fuentes de riesgo. Por un lado, puede provocar inconsistencia entre los requerimientos ya existentes, ya que los nuevos requerimientos pueden estar en contra de la lógica de negocio original o no ser compatibles con el sistema. Respecto a la programación de los objetos

existe el riesgo de generar problemas de versiones en los componentes de código, pues es posible que varios requerimientos utilicen las mismas clases y/o métodos, provocando que algunas funcionalidades ya implementadas y probadas sean modificadas negativamente o incluso borradas.

Por otro lado, se presenta el riesgo de no contar con un ambiente de pruebas, pues por política interna del cliente, existe un programa de actualización de servidores durante el transcurso de esta vuelta al modelo en espiral.

3.3.5.3 Fase de ingeniería

Referente a la actualización de los servidores de pruebas, se tuvieron varias reuniones para planear las actividades, de tal manera que no afectaran el flujo del desarrollo del SIGP. Se acordó aplicar la ventana de mantenimiento en fin de semana para no impactar la disponibilidad del ambiente.

Ahora con los nuevos requerimientos se procedió a actualizar la documentación existente para que incluyera el alcance agregado y de manera simultánea se ejecutó una revisión de las especificaciones técnicas y funcionales para evitar incompatibilidades con los requerimientos ya implementados. Afortunadamente la incorporación del nuevo alcance no requirió volver a diseñar o construir lo ya implementado, pues sólo se ampliaban algunas funcionalidades manteniendo su esencia, por lo que la implementación en código fue ejecutada. Respecto a los errores reportados en la vuelta anterior, se trabajó exhaustivamente para eliminarlos, coordinando tanto a los desarrolladores como al equipo de pruebas. De esta manera se aseguró la calidad en la implementación de los requerimientos. A estas alturas del proyecto todos los requerimientos ya estaban implementados e integrados en un ambiente de pruebas adecuado. Debido a esta situación

fue factible ejecutar pruebas de desempeño al sistema, con el propósito de ver su respuesta y hacer las modificaciones pertinentes de ser necesario.

3.3.5.4 Fase de evaluación

Dado que ya se contaba con una versión completamente funcional del SIGP, la fase de evaluación en esta vuelta del modelo en espiral requirió de un equipo robusto de usuarios claves y líderes funcionales por parte del cliente. Así, se busca someter a nuestro sistema a pruebas de flujos de trabajo reales manipulados por las personas que lo utilizarán día a día. Después de dos semanas intensas de pruebas y evaluaciones por parte de un equipo funcional compuesto por 15 personas, se obtuvieron resultados satisfactorios, donde se reportaron sólo algunos defectos mínimos que no representan una amenaza en contra de la liberación de nuestro sistema, siendo de esta manera el punto de partida para el plan de despliegue del SIGP en ambiente de producción.

3.3.6 Vuelta 6: Liberación final

3.3.6.1 Fase de planificación

A pesar de tener el proyecto prácticamente terminado, aún restaba corregir los errores menores y planear las actividades relacionadas al despliegue del SIGP en ambiente de producción. El objetivo principal fue lograr una liberación del sistema de manera correcta sin problemas de configuración y efectuar la transición del sistema previo al nuevo sistema sin impactos en la disponibilidad del servicio.

La principal restricción consistió en efectuar la transición sin impactar a los usuarios en sus operaciones cotidianas. Asimismo, la coordinación con los equipos de infraestructura

representaba una limitante, pues de no tener el apoyo de todos los involucrados el plan de liberación podía verse afectado.

Las alternativas para efectuar el despliegue de nuestro sistema fueron coordinar la liberación en un día hábil, pero en horario no laborable o efectuarla en fin de semana, mediante la negociación con los equipos de infraestructura.

3.3.6.2 Fase de análisis de riesgo

De los principales riesgos en la liberación del SIGP fue la posibilidad de que la nueva configuración, tuviera problemas al momento de desplegar la aplicación en los servidores de producción. Por otro lado, también era latente el riesgo de ejecutar los scripts o actividades en el orden incorrecto causando fallas importantes en el sistema. Asimismo, era probable que el tiempo dedicado al despliegue de la aplicación no fuera suficiente y causara intermitencia en el servicio.

3.3.6.3 Fase de ingeniería

La primera actividad en la fase de ingeniería fue la corrección de los últimos errores encontrados en las pruebas ejecutadas por el equipo funcional. Dado que se trataba de defectos menores, fueron rápidamente atendidos por el equipo de desarrollo y verificados por el cliente con resultados positivos.

En seguida se trabajó en la generación del plan de despliegue del SIGP, para definir cada una de las tareas necesarias y así lograr una transición adecuada. En el plan se definió el orden de las tareas, los scripts que deben ejecutarse, las adaptaciones y cambios en los

archivos de configuración y los detalles de conexión a otros sistemas y a bases de datos con las que nuestro sistema interactúa.

Una vez completado el plan de despliegue, se efectuaron tres simulaciones de la liberación del SIGP en el ambiente de pruebas disponible para corroborar que todas las tareas se definieron correctamente.

Finalmente, se desplegó la aplicación en ambiente de producción en un fin de semana con el propósito de no impactar las operaciones diarias de los usuarios. A parte de un par de errores de configuración, que fueron corregidos al momento, no hubo ningún problema al desplegar nuestro sistema

3.3.6.4 Fase de evaluación

Al tener el SIGP en ambiente de producción, se efectuaron las evaluaciones pertinentes por parte del cliente, con el apoyo de los líderes funcionales y los analistas de negocios para confirmar el adecuado funcionamiento del sistema. A parte de algunas preguntas de los evaluadores del funcionamiento de la aplicación, no se encontraron problemas en la misma y se notificó de la liberación a todos los usuarios para hacerles saber que la nueva versión del sistema había sido entregada y se encontraba disponible para su utilización. De esta manera, el desarrollo de software bajo el modelo en espiral culmina. Lo siguiente por hacer consiste en generar e implementar un plan de mantenimiento para cubrir las necesidades de la aplicación en caso de ocurrir eventualidades o prevenirlas.