



“Responsabilidad con pensamiento positivo”

UNIVERSIDAD TECNOLÓGICA ISRAEL

TRABAJO DE TITULACIÓN

CARRERA: INGENIERÍA EN SISTEMAS INFORMÁTICOS

TEMA: Sistema de Gestión Automatizada de Casos de Pruebas Funcionales

AUTOR: DAVID RUBÉN BALSECA KELAL

TUTOR: Mg. FRANZ DEL POZO

D. M. Quito, Septiembre del 2014

UNIVERSIDAD TECNOLÓGICA ISRAEL

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Titulación certifico:

Que el Trabajo de Titulación “SISTEMA DE GESTIÓN AUTOMATIZADA DE CASOS DE PRUEBAS”, presentado por el Sr. David Rubén Balseca Kelal, estudiante de la Carrera Ingeniería en Sistemas Informáticos, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito D. M., Agosto 2014

TUTOR

Mg. FRANZ DEL POZO

UNIVERSIDAD TECNOLÓGICA ISRAEL

AUTORÍA DE TRABAJO DE TITULACIÓN

El abajo firmante, en calidad de estudiante de la Carrera Ingeniería en Sistemas Informáticos, declaro que los contenidos de este Trabajo de Titulación, requisito previo a la obtención del Grado de Ingeniería en Sistemas Informáticos, son absolutamente originales, auténticos y de exclusiva responsabilidad legal y académica del autor.

Quito D. M., Agosto 2014

David Rubén Balseca Kelal

C.C.: 091085443-9

UNIVERSIDAD TECNOLÓGICA ISRAEL

APROBACIÓN DEL TRIBUNAL DE GRADO

Los miembros del Tribunal de Grado, aprueban el Trabajo de Titulación para la graduación de acuerdo con las disposiciones reglamentarias emitidas por la Universidad Tecnológica Israel para títulos de pregrado.

Quito D. M., Agosto 2014

Para constancia firman:

TRIBUNAL DE GRADO

PRESIDENTE

MIEMBRO 1

MIEMBRO 2

AGRADECIMIENTO

Conocer el grado del amor con el que Jesús Cristo nos ama nos debería hacer quererle de tal forma que quedase reflejado en nuestras actitudes, nuestra conducta y nuestro compromiso. La madurez espiritual se demuestra cuando sabemos activar nuestro conocimiento espiritual. (Edward Bedore).

Agradezco a mis padres, pero sobre todo a mi Madre Norma Kelal Zambrano porque siempre estuvo conmigo, su ayuda y apoyo incondicional para que terminara mis estudios y obtenga mi Título de Ingeniero en Sistemas fue un gran pilar y aliento para poder seguir adelante.

David Rubén Balseca Kelal

DEDICATORIA

Bendito sea Dios. El que nos conforta en toda prueba, para que también nosotros seamos capaces de confortar a los que están en cualquier dificultad (2Cor 1,3-4).

Dedico mi sacrificio a mi familia y en especial a mis hijos por el apoyo constante en el transcurso de mi vida para poder alcanzar mis metas personales y profesionales.

David Rubén Balseca Kelal

UNIVERSIDAD TECNOLÓGICA ISRAEL

PLAN DEL PROYECTO INTEGRADOR DE CARRERA

CARRERA / PROGRAMA:	Ingeniería en Sistemas Informáticos
AUTOR:	David Rubén Balseca Kelal
TEMA DEL TT:	Sistema de Gestión Automatizada de Casos de Pruebas Funcionales
ARTICULACIÓN CON LA LÍNEA DE INVESTIGACIÓN INSTITUCIONAL:	Tecnología Aplicada al sistema de tecnologías de Información.
SUBLÍNEA DE INVESTIGACIÓN INSTITUCIONAL:	Diseño y automatización del proceso de certificación de aplicaciones

ÍNDICE GENERAL

APROBACIÓN DEL TUTOR	II
AUTORÍA DE TRABAJO DE TITULACIÓN	III
APROBACIÓN DEL TRIBUNAL DE GRADO	IV
AGRADECIMIENTO.....	V
DEDICATORIA.....	VI
PLAN DEL PROYECTO INTEGRADOR DE CARRERA	VII
INTRODUCCIÓN	1
Antecedentes	1
Objetivos	2
Objetivo General	2
Objetivos Específicos	2
Idea a defender (Hipótesis)	2
CAPÍTULO 1.....	3
FUNDAMENTACIÓN TEÓRICA.....	3
1.1 Introducción	3
1.2 Marco Conceptual.....	3
1.2.1 Planeación Estratégica	3
1.2.2 Necesidades de Negocios.....	3
1.2.3 Requerimiento Funcional	4
1.2.4 Requerimiento no funcional.....	4
1.2.5 Ingeniero de pruebas / tester	4
1.2.6 Ciclo de vida de los Sistemas	4
1.3 Marco Teórico	4
1.3.1 Creación de ideas	4
1.3.2 Etapas del Plan de Negocio	5
1.3.3 Análisis de factibilidad.....	5
1.3.4 Clasificación de los Requerimientos Funcionales.....	6
1.3.5 Ciclo de vida de los Sistemas	6
1.3.6 Pruebas Funcionales	8

1.3.7	Matriz de Trazabilidad.....	8
1.3.8	Defecto de Software (BUG).....	9
1.3.9	Pruebas de Regresión.	9
1.3.10	Ingeniero de pruebas / tester	10
1.3.11	Indicadores de Gestión.	10
1.3.12	Pruebas de aceptación	11
1.3.13	Control de calidad en proyectos de software.....	11
 CAPÍTULO 2.....		12
DIAGNÓSTICO DEL PROBLEMA.....		12
2.1	Problema Principal.....	12
2.2	Impacto General	12
2.3	Impacto Interno.....	14
2.4	Metodología investigativa.....	16
2.5	Metodología de investigación.....	16
2.6	Métodos de investigación.....	16
2.6.1	Metodología de Campo.....	16
2.6.2	Metodología Analítica.....	16
2.6.3	Metodología Sintética.....	17
 CAPÍTULO 3.....		18
DESARROLLO DE LA PROPUESTA.....		18
3.1	Fundamentación Técnica.....	18
3.1.1	Metodologías utilizadas.....	20
3.1.1.1	Metodología Cascada	20
3.1.1.2	Modelamiento UML.....	21
3.1.1.3	Casos de Usos.....	22
3.1.1.4	Actor	22
3.1.1.5	Base de Datos SQL Server.....	22
3.1.1.6	.Net (C#).....	22
3.1.1.7	PowerDesigner	22
3.2	Desarrollo de la propuesta	23
3.2.1	Alcance funcional del proyecto.....	23

3.2.1.1	Parámetros Generales	23
3.2.1.2	Seguridades.....	24
3.2.1.3	Biblioteca de Casos	24
3.2.1.4	Proyectos.....	25
3.2.1.5	Reportes	26
3.2.1.6	Salir	26
3.2.2	Desarrollo de la propuesta	26
3.2.3	Cronograma.....	26
3.2.4	Requisitos del sistema	27
3.2.5	Modelo de negocio.....	27
3.2.6	Diagrama de actividades.....	28
3.2.6.1	Proceso de creación de proyectos y sus sub-procesos asociados.	28
3.2.6.2	Diagrama de Actividades – Proceso de ejecución de pruebas.	29
3.2.6.3	Diagrama de Actividades – Proceso de flujos de errores.	30
3.2.7	Diagrama de contexto	31
3.2.7.1	Diagrama de contexto principales del Sistema.....	31
3.2.8	Diagrama de casos de usos.....	33
3.2.8.1	Caso de Uso Módulo: Administración del sistema.....	33
3.2.8.2	Caso de Uso Módulo: Seguridades del sistema.....	37
3.2.8.3	Caso de Uso Módulo: Biblioteca de Casos	41
3.2.8.4	Caso de Uso Módulo: Creación de Proyectos.....	43
3.2.8.5	Caso de Uso Módulo: Ejecución	48
3.2.8.6	Caso de Uso Módulo: Reportes	52
3.2.9	Diagrama de secuencias.....	55
3.2.9.1	Diagrama de secuencias – ingreso al sistema	56
3.2.9.2	Diagrama de secuencias – Ejecución de proyectos	57
3.2.10	Diagrama de despliegue	57
3.2.11	Diagrama de Clases.....	59
3.2.11.1	Diagrama de Clases – Negocios.....	59
3.2.11.2	Diagrama de Clases – Conexiones	59
3.2.11.3	Diagrama de Clases – Datos	60
3.2.12	Modelo de la base de datos	61
3.2.13	Diagrama modelo entidad relación.....	62

3.2.14	Interfaces de usuarios.....	63
3.2.14.1	Figura. Interfaz: Pantalla de inicio de sesión	63
3.2.14.2	Figura. Interfaz: Pantalla de menú	63
3.3	Estado de situación actual	64
CONCLUSIONES Y RECOMENDACIONES		66
	Conclusiones	66
	Recomendaciones	67
	Bibliografía	68
ANEXOS		69
	ANEXO – CARTA DE AUSPICIO.....	70
	ANEXO – ENTREVISTA	71
	ANEXO - MANUAL DE USUARIOS	72

ÍNDICE DE TABLAS

Tabla N. 1	Matriz de Trazabilidad	9
Tabla N. 1	Matriz de Trazabilidad	34
Tabla N. 2	Caso de uso – Catálogos	35
Tabla N. 3	Caso de uso – Flujo estados.....	36
Tabla N. 4	Caso de uso – Pantallas	38
Tabla N. 5	Caso de uso – Seguridad usuarios	39
Tabla N. 6	Caso de uso – Seguridad perfiles	40
Tabla N. 7	Caso de uso – Seguridad menú.....	42
Tabla N. 8	Caso de uso – Proyectos	44
Tabla N. 9	Caso de uso – Intervinientes.....	45
Tabla N. 10	Caso de uso – Ciclos casos	46
Tabla N. 11	Caso de uso – Plan de ejecución.....	47
Tabla N. 12	Caso de uso – Aprobación	48
Tabla N. 13	Caso de uso – Ejecución.....	50
Tabla N. 14	Caso de uso – Lista de Tareas	50
Tabla N. 15	Caso de uso – Defectos.....	52
Tabla N. 16	Caso de uso – Consolidado de errores	53
Tabla N. 17	Caso de uso – Detalle de errores.....	54
Tabla N. 18	Caso de uso – Estado del Proyecto	55

ÍNDICE DE GRÁFICOS

Gráfico N. 1	Tablero de control de Proyectos	13
Gráfico N. 2	Distribución de causales de desfases	13
Gráfico N. 3	Esfuerzo en diseño y ejecución de casos	15
Gráfico N. 4	Esfuerzo por fases del ciclo de vida.....	15
Gráfico N. 5	Modelo programación web 3 capas	18
Gráfico N. 6	SQL Server.....	19
Gráfico N. 7	Visual Studio	20
Gráfico N. 8	Ciclo de vida cascada	20
Gráfico N. 9	Cronograma de propuesta.....	27
Gráfico N. 10	Diagrama de actividades Creación de Proyectos.....	28
Gráfico N. 11	Diagrama de actividades Ejecución de pruebas	29
Gráfico N. 12	Diagrama de actividades Flujo de errores.....	30
Gráfico N. 13	Diagrama de contexto 1.....	31
Gráfico N. 14	Diagrama de contexto 2.....	32
Gráfico N. 15	Diagrama de contexto 3.....	32
Gráfico N. 16	Casos de Usos – Administración del Sistema.....	33
Gráfico N. 17	Casos de Usos – Seguridad del Sistema.....	37
Gráfico N. 18	Casos de Usos – Biblioteca de casos	41
Gráfico N. 19	Casos de Usos – Creación de proyectos	43
Gráfico N. 20	Casos de Usos – Ejecución	48
Gráfico N. 21	Casos de Usos – Reportes.....	52
Gráfico N. 22	Diagrama de secuencias – ingreso al sistema.....	56
Gráfico N. 23	Diagrama de secuencias –Ejecución de proyectos.....	57
Gráfico N. 24	Diagrama de despliegue.....	58
Gráfico N. 25	Diagrama de Clases - Negocio	59
Gráfico N. 26	Diagrama de Clases - Conexiones	59
Gráfico N. 27	Diagrama de Clases - Datos.....	60
Gráfico N. 28	Modelo de la base de datos.....	61
Gráfico N. 29	Modelo entidad relación.....	62
Gráfico N. 30	Interfaz de usuario - login	63
Gráfico N. 31	Interfaz de usuario -Menú.....	63
Gráfico N. 32	Tabla comparativa de beneficios en esfuerzos	65

INTRODUCCIÓN

Antecedentes

En la historia muchas empresas han experimentado millonarias pérdidas financieras y de clientes debido a un mal proceso de pruebas e identificación de errores en sus sistemas computacionales.

El no contar en las décadas pasadas con metodologías formales y herramientas automatizadas para el ciclo de pruebas de una aplicación ha generado eventos que en la actualidad siguen siendo referenciados para difundir las metodologías de pruebas funcionales, tales como:

En 1980, la muerte de 25 personas por recibir dosis erradas de radiación por la máquina Therac-25 y la causa fue aludida a un error en el sistema principal del dispositivo.

En 1996, la agencia espacial Europea perdió \$ 1.000,00 millones de dólares en el cohete prototipo Ariane 5, y la causa fue asociada a un error en el programa de la computadora principal del cohete.

El Departamento Nacional de Estándares de Comercio y Tecnología de los Estados Unidos, indicó que los errores o fallas en los sistemas informáticos cuestan mucho dinero a la economía American, y citan como ejemplo que anualmente cuesta un aproximado de 59 millones de dólares. (Mary, 2001)

El termino o palabra “Bug” se traduce como insecto en español y actualmente es una jerga comúnmente usada por personas que trabajan en el ámbito de la computación. (failure, 2010)

Los errores pueden aparecer en cualquier momento en el desarrollo de aplicaciones, por tal motivo, los procesos de certificación deben tomar más fueras y reforzar estas fallas humanas. Dichas fallas aparecen por descuidos o malentendidos en el desarrollo del software.

La cultura del proceso de pruebas funcionales en el país no es acogida totalmente y tiene varios motivadores como son los costos que incurren el adquirir la herramienta y establecer un equipo único dedicado al 100% al proceso de

certificación, por tal motivo, implementan procesos manuales los que incurren en costos adicionales y recurrentes por no tener herramientas adecuadas y posterior a la implementación se presentan errores en los entornos productivos acarreado los problemas e insatisfacción de los clientes externos a las entidades.

El problema que se pretende resolver es la ineficiencia que existe en el proceso de certificación de pruebas funcionales en las aplicaciones desarrolladas, lo que ocasiona un problema de mala calidad en su funcionamiento y de re trabajo por parte de los empleados de la Cooperativa.

Objetivos

Objetivo General

Mejorar la eficiencia en los procesos de pruebas funcionales en el desarrollo de aplicaciones.

Objetivos Específicos

- Analizar los procesos que podrían ser automatizados con el uso de una herramienta.
- Diseñar y desarrollar nuevos procesos que permitan registrar y actualizar en línea los casos de pruebas funcionales.
- Crear una aplicación WEB que permita la gestión de los casos de pruebas funcionales.
- Implementar la aplicación en la cooperativa y a través de un proyecto piloto validar su funcionamiento y aplicabilidad.

Idea a defender (Hipótesis)

Hipótesis Afirmativa: Reducirá el tiempo de ejecución de las pruebas funcionales al menos en un 30%.

Hipótesis Negativa: El tiempo de elaboración de los casos de pruebas no tendrá variación.

CAPÍTULO 1

FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

Para el desarrollo de este trabajo de tesis se ha utilizado varios conceptos teóricos relacionados en los procesos de las necesidades del negocio y del proceso de certificación funcional del desarrollo de aplicaciones.

1.2 Marco Conceptual

1.2.1 Planeación Estratégica

Los planes de negocios son utilizados básicamente para convencer tanto a los administradores con categoría interna y como a los usuarios e interesados sobre la necesidad de una nueva tecnología. Un buen plan de negocios siempre tener una buena estructura, deber ser lógico y además debe persuadir a los interesados. Se mostrará más elementos de los que los asistentes necesitan o puedan comprender.

Es más probable que el trabajo más duro de preparar un plan de negocios es realizado en la fase de la evaluación de las necesidades del negocio. Por lo tanto el plan como la evaluación de necesidades debe estar orientado para convencer, aunque tenga audiencias distintas. Si la evaluación tiene éxito al convencer a los promotores/interesados de la propuesta, todavía toca que recorrer un largo camino antes de plantear el plan de negocios completo para vender por completo la propuesta a todos los interesados en la misma. (Abances, 2010)

1.2.2 Necesidades de Negocios

Las etapas iniciales de una estrategia de negocio para el desarrollo y posible venta de una implantación de un nuevo proyecto siempre será una evaluación de necesidades. Nos debemos preguntar ¿Qué estamos haciendo actualmente? ¿Debemos mejorar? ¿Qué esperan los accionistas? ¿Qué pensará el personal de la presencia de nueva tecnología? En la etapa final del proceso de evaluación, podremos concluir si la tecnología planteada es la que realmente necesitamos.

Cuando se decide incorporar la nueva tecnología o realizar una actualización de la existente, se debe delinear un plan detallado de todas las actividades y de esta

manera asegurar el completo apoyo de todos los interesados y de los fondos requeridos para su implantación. (Abances, 2010)

1.2.3 Requerimiento Funcional

Los requerimientos funcionales de una aplicación detallan las necesidades funcionalidades o los servicios que espera que el sistema provea.

Se lo define en la ingeniería de requisitos, dichos requisitos funcionales describen como debe comportarse el sistema.

1.2.4 Requerimiento no funcional

Detalla claramente aspectos relacionado al sistema que son visibles y tangibles por el usuario pero los cuales no incluyen una relación directa a sus necesidades funcionales.

1.2.5 Ingeniero de pruebas / tester

El ingeniero de pruebas (tester) es un recurso especializado en procesos de verificación y validación de calidad de software.

1.2.6 Ciclo de vida de los Sistemas

El ciclo de vida de los sistemas es el proceso que se sigue para construir, entregar y hacer evolucionar el software, desde la concepción de una idea hasta la entrega y el retiro del sistema. (Sommerville, 2011)

1.3 Marco Teórico

1.3.1 Creación de ideas

Para el desarrollo de este trabajo de tesis se describe conceptos con mayor nivel de detalle que serán utilizados para el desarrollo de la tesis.

Un plan de negocios típico puede contener los siguientes elementos:

- ✓ La idea propuesta.
- ✓ La idea claramente fundamentando la necesidad.
- ✓ Un análisis con otras opciones disponibles y la sustentación del porqué existe inclinación a la propuesta.
- ✓ Mostrar costos y beneficios de la propuesta

- ✓ Análisis de costos y los ahorros monetarios de la propuesta, y debe incluir el costo del mantenimiento a lo largo del ciclo de vida calculado.
- ✓ Un cronograma.
- ✓ Al final una clara recomendación para poder aprobar la propuesta.

1.3.2 Etapas del Plan de Negocio

La evaluación de necesidades normalmente empieza con el análisis que debe ser riguroso sobre los procesos existentes en la empresa. Este proceso permitirá identificar áreas de mejora.

En esta etapa posiblemente permitirá seguir diferentes caminos de consulta simultáneas:

- ✓ Los técnicos podrán determinar si la solución técnica propuesta es adecuada para el propósito requerido.
- ✓ Los asesores financieros estimaran el costo y establecerán si es o no accesible.
- ✓ Los gerentes del proyecto podrán identificar si existe el apoyo en la empresa.
- ✓ Los altos ejecutivos podrán consultar a todos los interesados sobre el cambio propuesto.

Al final de todas estas evaluaciones, el comité que deba, tomará la decisión si es o no, factible la propuesta planteada.

1.3.3 Análisis de factibilidad

Factibilidad describe la disponibilidad de que los recursos que se requieran para alcanzar los objetivos estén disponibles. En muchas ocasiones la factibilidad se la determina comparándola contra un proyecto o estudio.

Realizar el estudio de factibilidad es realizada en etapas tempranas de un desarrollo de un sistema informático.

El estudio incluye aclarar los objetivos del proyecto, qué alcance tendrás e identificar que restricciones existirán en el sistema

Existen varios tipos de factibilidad y básicamente serán:

- ✓ Estudio de factibilidad técnica: que asegura que si existe o se encuentra al alcance de nosotros la tecnología para poder ejecutar o desarrollar el sistema.

- ✓ Estudios de factibilidad económica: demuestra costo beneficio de desarrollar el sistema.
- ✓ Estudio de factibilidad operacional: Demuestra si el sistema puede ser realmente implementado en nuestra organización.

1.3.4 Clasificación de los Requerimientos Funcionales

Los requerimientos funcionales de una aplicación detallan las necesidades funcionalidades o los servicios que espera que el sistema provea. Los mismos pueden ser clasificados de la siguiente manera:

- ✓ Requerimientos de datos o información, requerimientos de negocio.
- ✓ Requerimientos de interfaz de usuario o también conocido como la iteración que existe entre el sistema y el usuario final.
- ✓ Requerimientos de navegación que indican el nivel de complejidad que deberá tener la aplicación para su navegación.

Los requerimientos de carga transaccional o de funciones internas del sistema, recopilan lo que debe hacer el sistema de forma transparente para el usuario.

1.3.5 Ciclo de vida de los Sistemas

El ciclo de vida de desarrollo típicamente tiene las siguientes fases o pasos:

- **Creación e Inicio del Proyecto**

Esta fase involucra las actividades de creación del proyecto.

Esta fase involucra actividades de inicio de proyecto y creación de estructuras para la ejecución del mismo. Se define el Proceso Operativo y se establecen los diferentes planes para gestionar el proyecto. Se identifican los estándares, metodologías, herramientas, personal y otros recursos necesarios para ejecutar el proyecto.

- **Análisis y Determinación de los requisitos del sistema.**

Esta fase cubre actividades enfocadas en la identificación de requerimientos de software. Se busca comprender los requerimientos del usuario a través del estudio de los problemas, los sistemas existentes y discusiones con el usuario.

El análisis permite llegar a una solución de sistema en términos de características vitales, esenciales y deseadas.

- **Diseño alto y bajo nivel del sistema**

Esta fase cubre actividades enfocadas en la identificación de requerimientos de software. Se busca comprender los requerimientos del usuario a través del estudio de los problemas, los sistemas existentes y discusiones con el usuario. El análisis permite llegar a una solución de sistema en términos de características vitales, esenciales y deseadas.

- **Desarrollo de software (diseño físico) y pruebas unitarias**

La Construcción transforma la representación detallada del diseño del producto de software en programa. En esta fase se produce el código fuente, bases de datos, archivos y la documentación que constituye la manifestación física del diseño. Adicionalmente, se integran el código y la base de datos/archivos

- **Pruebas de sistemas**

Esta fase cubre pruebas de sistema para la funcionalidad, seguridad, recuperación, y reinicio, desempeño, interfaces externas, usabilidad e integración de diferentes módulos a nivel de sistema.

Esta fase es recursiva y necesita ser repetida tantas veces como niveles de pruebas existan.

Las combinaciones posibles son:

- ✓ Solo pruebas de sistema
- ✓ Pruebas de sistema y de módulo
- ✓ Pruebas de módulo, sistema y subsistema.

- **Pruebas de Aceptación**

Durante esta fase, el sistema se expone para asegurar que el mismo no tenga fallas y que funcione en base a las necesidades descritas en el documento funcional, es decir, las necesidades del usuario final. Básicamente esta pruebas debe ser realizada por el usuario final y usuario del sistema.

- **Paquete y liberación.**

Esta fase se ejecuta cuando se entrega cualquier entregable al cliente.

Se lleva a cabo la inspección final para asegurar que los productos (programas, información y documentos) estén listos para la entrega. Solo entonces el entregable se empaqueta y se entrega al cliente.

- **Soporte post-implementación**

El soporte post-implementación es el proceso de verificar que la aplicación funciona correctamente. En caso de existir novedades se corrigen los errores.

- **Cierre del Proyecto**

Esta fase involucra las actividades necesarias para cerrar/completar el proyecto. Esta fase incluye actualización de los repositorios, métricas, perfil de proyecto, e información relacionada. Se requiere actualizar el log del proyecto, liberar recursos y entregar registros.

Esta siempre es la fase final de cualquier proyecto. (Tejada, 2007)

1.3.6 Pruebas Funcionales

Las pruebas funcionales tienen como objetivo comprobar que los sistemas desarrollados funcionan de acuerdo a las especificaciones descritas en el documento funcional y que plasman el requerimiento del cliente.

La mayoría de las pruebas funcionales son realizadas de forma manual, pero las mismas pueden ser automatizadas, las cuales, permiten ahorrar tiempo en procedimientos repetitivos.

1.3.7 Matriz de Trazabilidad

En el ciclo de vida de cualquier desarrollo de una aplicación o software en un proyecto se recomienda que exista una trazabilidad entre los requisitos o especificaciones del usuario y el cruce contra lo que se diseñó o construyó en el sistema.

Esta misma matriz sirve para poder realizar el proceso de pruebas funcionales, ya que será una guía de lo que se debe probar o no. Esta matriz será una herramienta que nos permita certificar lo solicitado por el usuario final. Por ejemplo,

Supongamos que estamos en un proyecto con 5 necesidades funcionales (NF1-NF5) y se han creados tres casos para probar la funcionalidad del sistema (P1-P3).

- El script de prueba P1 cubre los requerimientos NF1 y NF4
- El script de prueba P2 cubre los requerimientos NF3 y NF5
- El script de prueba P3 cubre el requerimiento NF3

En este caso la matriz resultante será:

	P1	P2	P3
NF1	X		
NF2			
NF3		X	X
NF4	X		
NF5		X	

Tabla N. 1 Matriz de Trazabilidad
 Autor: David Balseca Fuente: Investigador

Podemos ver claramente dos cosas en la matriz:

1. La necesidad funcional NF3 está probado con 2 scripts de prueba.
2. La necesidad funcional NF2 no será cubierto en la prueba.

1.3.8 Defecto de Software (BUG).

El defecto o fallo de una aplicación o software, resulta de una falla humana durante la creación de los programas informáticos. Este error puede aparecer en cualquiera de las fases de desarrollo normal de una aplicación.

1.3.9 Pruebas de Regresión.

Las pruebas de regresión en cualquier categoría de pruebas de software intentan identificar errores, deficiencia en la funcionalidad implementada, o mal entendidos funcionales por lo requerido por el cliente.

Estos cambios o mal entendidos generan defectos en las aplicaciones y para cada uno de ellos, se necesita realizar modificación o cambios en las aplicaciones.

Como una buena práctica se recomienda probar la corrección o el cambio específico y además, probar el resto de funcionalidad que pudo ser afectada por las modificaciones anteriores, a esto, se llama regresión.

1.3.10 Ingeniero de pruebas / tester

Las macro actividades del ingeniero de pruebas o tester están las siguientes actividades:

- ✓ Elaborar el plan de pruebas para el proyecto.
- ✓ Actualizar de ser necesario el plan de pruebas y realizar las correcciones requeridas.
- ✓ Realizar inspecciones a los productos de software, es decir, validar los documentos de los requerimientos, del diseño y de la arquitectura del sistema.
- ✓ Diseñar las pruebas de catalogadas como caja blanca, caja negra, pruebas de estrés, de integración y las de aceptación del usuario final o cliente.
- ✓ Ejecutar las pruebas planificadas en sus diferentes fases definidas.
- ✓ Generar reporte de resultados de las pruebas y proceso de calidad realizados.
- ✓ Validar que se realicen los cambios requeridos resultantes de los defectos u observaciones encontradas.
- ✓ Documentar todas las actividades del proceso de pruebas realizadas.

1.3.11 Indicadores de Gestión.

Los indicadores de gestión son medidas utilizadas para determinar el éxito de un proyecto o una organización. Los indicadores de gestión suelen establecerse por los líderes del proyecto u organización, y son posteriormente utilizados continuamente a lo largo del ciclo de vida, para evaluar el desempeño y los resultados.

Los indicadores de gestión suelen estar ligados con resultados cuantificables, como ventas anuales o reducción de costos en manufactura.

1.3.12 Pruebas de aceptación

Estas pruebas son realizadas y ejecutadas al 100% por el cliente. Básicamente consiste en pruebas funcionales, sobre todo el sistema completo, y las mismas buscan realizar una validación que tenga una cobertura total validando las especificaciones funcionales. (Hutcheson, 2010)

1.3.13 Control de calidad en proyectos de software.

Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente. (Acuña, 2012)

CAPÍTULO 2

DIAGNÓSTICO DEL PROBLEMA

2.1 Problema Principal

La cooperativa COOPROGESO S.A actualmente presenta inconvenientes importantes en el proceso de certificación funcional de los desarrollos internos y externos para cubrir la demanda funcional de los clientes internos.

Los problemas actuales se ven reflejados en un costo económico por re-trabajo y por el tamaño del esfuerzo dedicado al proceso de diseño de los casos de pruebas y en la certificación funcional, además a estos inconvenientes podemos citar la pérdida de cliente por los problemas ocasionados en producción por la baja calidad de las aplicaciones que liberan a uso de los usuarios internos y externos del banco.

La implementación de un sistema automatizado para el proceso de pruebas funcionales integrado de punta a punta en el proceso de certificación espera cubrir la problemática y reducir el esfuerzo de los empleados y recursos asignados al proceso de pruebas funcionales.

La nueva aplicación de pruebas permitirá administrar de mejor manera el proceso de certificación y generará eficiencia al aplicar el concepto de bibliotecas de casos de pruebas el cual permitirá reutilizar los casos escritos en proceso previos de certificación reduciendo el esfuerzo y las horas hombres de sus empleados de las diferentes áreas de negocios.

2.2 Impacto General

Dentro del portafolio de proyectos actual de la cooperativa COOPROGRESO S.A se puede evidenciar los porcentajes de desviación en los proyectos, para lo se adjunta gráfico de tablero de control de proyectos vigentes al 2014:

DATOS DE PROYECTO EN LA COOPERATIVA			PLANEACION				DATOS ESPERADOS			DATOS A LA FECHA		
Código de proyecto	Tipo	Proyectos	Esfuerzo Total	Recursos	Presupuesto Planeado	Duración planeada (Meses)	Duración Real (Meses)	% de avance esperado	Esfuerzo esperado	Presupuesto Esperado a ejecutar	% REAL AVANCE	% DESVIACION
PRY-14-010	Desarrollo	Proyecto de Mejora en la emisión de cuentas de ahorros	3330	6	73.260,00	3	2,0	67%	2220	48.840,00	50%	17%
PRY-14-013	Infraestructura	Proyecto de apertura de agencia Y	3825	5	84.150,00	4	3,5	88%	3347	73.631,25	82%	6%
PRY-14-020	Desarrollo	Reporte de estructuras ente de control	1800	4	39.600,00	3	2	67%	1200	26.400,00	45%	22%
PRY-14-016	Desarrollo	Emisión de producto cuenta de ahorro con seguro	6200	5	136.400,00	4,5	3,5	78%	4822	106.088,89	44%	34%
PRY-14-022	Desarrollo	Generación de tarjeta de debito para nueva red de ATM	5100	6	112.200,00	6	4,5	75%	3825	84.150,00	65%	10%
PRY-14-018	Infraestructura	Montaje de cableado estructurado en Agencia tumbaco	2656	4	58.432,00	4	3	75%	1992	43.824,00	74%	1%
PRY-13-005	Desarrollo	Corrección de errores varios en el proceso de crédito	16800	7	369.600,00	8	11	138%	23100	508.200,00	50%	88%
PRY-14-024	Infraestructura	Montaje de cableado estructurado en Agencia quito sur	3700	5	81.400,00	4,5	3	67%	2467	54.266,67	65%	2%

Gráfico N. 1 Tablero de control de Proyectos
 Autor: Cooprogreso S.A Fuente: Cooprogreso S.A

En función de las entrevistas con varios de los grupos (stakeholders) que participan en la ejecución de proyectos evidencian varias debilidades en el gerenciamiento y en sus fases del proyecto.

Para lo cual mostramos un gráfico de las causales más recurrentes en la vida de los proyecto. El gráfico esta generado con información de los proyectos históricos del año 2013 y 2014.

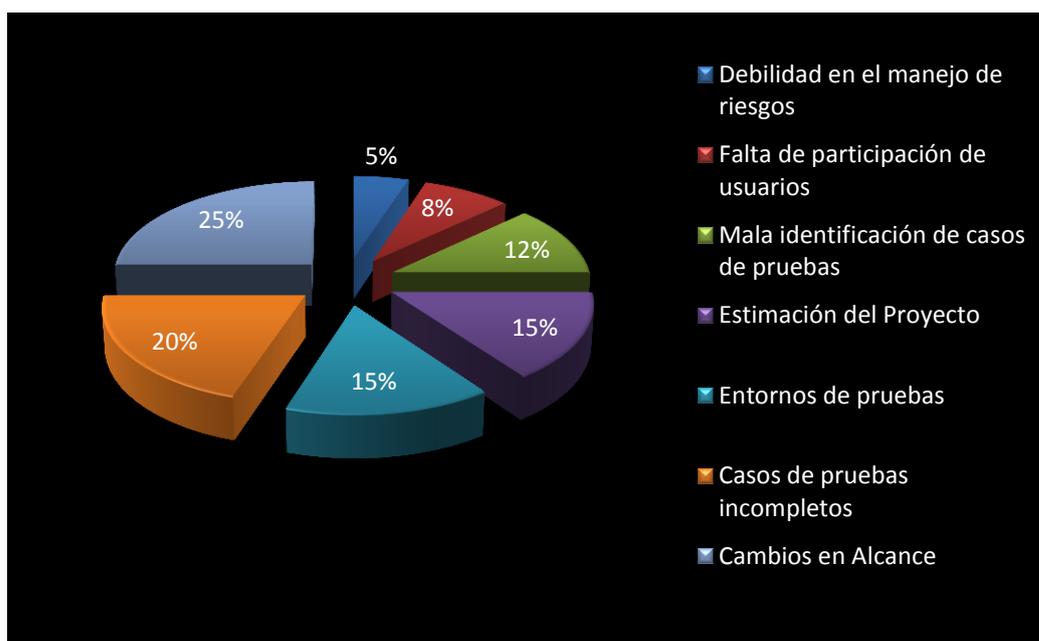


Gráfico N. 2 Distribución de causales de desfases
 Autor: David Balseca Fuente: Investigador

Con los datos adjuntos podemos concluir que de los 7 causales de desviación en los proyectos son:

- ✓ Cambio de Alcance
- ✓ Casos de pruebas incompletos
- ✓ Entorno de Pruebas
- ✓ Estimación inicial de los proyectos
- ✓ Mala identificación de los casos de pruebas
- ✓ Falta de participación de los usuarios
- ✓ Debilidad en el manejo de los riesgos del proyecto

De los cuales los causales 2 y 5 serán cubiertos con esta propuesta de solución automatizada de certificación funcional.

2.3 Impacto Interno

El proceso de recopilación de la información para identificar la problemática de la cooperativa, se basó en entrevistas personales y visita de campo en los proyectos que se encuentra ejecutando la cooperativa la fecha del relevamiento de la información.

Los grupos objetivos que colaboraron en la investigación fueron:

- ✓ Gerencial General
- ✓ Tecnología
- ✓ Grupo de usuarios expertos

Los problemas identificados más comunes asociados a los puntos 2 y 5 (Casos de pruebas incompletos y Mala identificación de casos de pruebas) se pueden resumir en:

- ✓ Casos de pruebas no detallados para la ejecución del caso.
- ✓ Casos de pruebas no identificados
- ✓ Falta de experiencia de los usuarios
- ✓ Demasiado tiempo asignado al proceso de certificación y elaboración de casos de pruebas

- ✓ Re-trabajo constante de escritura de casos de pruebas

Como un dato adicional en las entrevistas realizadas se recopiló información relacionada al esfuerzo que en la actualidad toma las dos principales actividades del proceso de pruebas, los cuales son: Diseño de los casos y ejecución de los casos de pruebas y su resolución en caso de existir defectos en la certificación.

Se adjunta un cuadro del esfuerzo actual en horas de las 2 actividades antes mencionadas.

PROCESO MANUAL (horas)					
	Escritura Casos	Aprobación	Corrección	Aprobación	Horas por Caso
Diseño x caso	3	12	2	6	23
Ejecución x caso	1	0	16	1	18

Gráfico N. 3 Esfuerzo en diseño y ejecución de casos
 Autor: David Balseca Fuente: Investigador

También se desglosó el esfuerzo de los proyectos por cada una de sus fases con la metodología de ejecución de proyectos que utiliza la cooperativa, se tomó como muestra 2 proyectos para poder tener una línea base de medición que será utilizada y comparada con los esfuerzos que arrojará el uso de la nueva aplicación.

Metodología MSF (Microsoft Solution Framework)	Actividades	Esfuerzo por fases	Proyecto: Corrección de errores varios en el proceso de crédito	Emisión de producto cuenta de ahorro con seguro
Vision	Documento de visión del cliente	10%	1680	620
Planificación	Especificaciones funcionales	15%	2520	930
	Diseño de aplicación			
Desarrollo	Diseño de pruebas	40%	6720	2480
	Desarrollo de la aplicación			
	Diseño técnico de desarrollo			
Estabilización	Pruebas Unitarias	30%	5040	1860
	Ejecución de pruebas funcionales			
Implementación	Corrección de errores	5%	840	310
	Liberación a producción			
	Soporte a estabilización			

Gráfico N. 4 Esfuerzo por fases del ciclo de vida
 Autor: David Balseca Fuente: Investigador

2.4 Metodología investigativa

La metodología nos ayuda como guía y nos evitar tomar caminos errados aparente ante los cambios de los fenómenos, y nos provee una guía para poder plantear los problemas de un modo objetivo (p.41). (Hill, 2004)

El método que vamos a utilizar en este proyecto de investigación es el Método deductivo, ya que permitió proveernos de una serie de información teórica que va a fundamentar el proyecto y nos dará una idea más amplia sobre los diversos aspectos a analizar y que formaran parte de la formulación propuesta a fin de que esta pueda generar una posición segura.

2.5 Metodología de investigación

“La investigación de campo que es el diseño de la investigación, es la etapa donde el investigador muestra qué se aplicará para recoger la información, la cual debe estar estrechamente vinculada con los objetivos que se haya planteado”(p.18). (Becerra, 2003)

El método que vamos a utilizar está ubicado dentro de la metodología de investigación de campo, debido a que se realizó en el lugar donde se presenta el problema, estableciendo una interacción entre los objetivos del estudio y la realidad actual de la cooperativa COOPROGRESO S.A.

2.6 Métodos de investigación

Este proyecto aplicará investigación científica a través del uso de varias metodologías como:

2.6.1 Metodología de Campo

Toma toda la información y requerimientos que se encuentran en proceso de pruebas funcionales existentes en la cooperativa COOPROGRESO S.a el cual definirá el campo de acción en el cual se apoyará el desarrollo del aplicativo web.

2.6.2 Metodología Analítica

Genera de manera abstracta los conocimientos que se hayan obtenido los cuales sirven para relacionarlos con el desarrollo de la aplicación web.

2.6.3 Metodología Sintética

Demuestra lo que se ha desarrollado y trata de alcanzar los objetivos planteados para obtener consecutivamente las respectivas conclusiones.

También se utilizó la Investigación Bibliográfica para obtener información valiosa que apoye la elaboración y la sustentación del proyecto.

Para conseguir la correcta realización del sistema de Gestión Automatizada de Casos de Pruebas Funcionales, se basó en la metodología de desarrollo Cascada; que encaminó la ejecución de las etapas del desarrollo de la aplicación (análisis, diseño, desarrollo, pruebas, implementación y cierre) bajo el apoyo de técnicas de notación UML.

Las técnicas de recolección de datos permitieron obtener una gran diversidad de herramientas para el desarrollo de los sistemas de información; por tal motivo tiene la finalidad de buscar mayor información la cual permitirá establecer de mejor manera los parámetros del sistema web y así entender de mejor manera el modelo de negocio actual y que se encuentra funcionando, entre algunas técnicas y herramientas se tienen:

- ✓ Entrevista.
- ✓ Observación.

CAPÍTULO 3

DESARROLLO DE LA PROPUESTA

3.1 Fundamentación Técnica

En función de los beneficios que provee la programación de 3 capas y entre las principales podemos mencionar: la abstracción de cada capa y la simplificación a los cambios que pueda sufrir una aplicación en su ciclo de vida, se seleccionó la misma para la implementación de la aplicación en la cooperativa COOPROGRESO S.A.

El diseño de la arquitectura web con tres capas es reciente y la misma introduce una capa intermedia en el camino o proceso. Cada capa se lo puede ver como procesos independientes y claramente definidos y los mismos tienen independencia de correr o ser ejecutados en plataformas separadas y diferentes. La arquitectura basada en Web convierte la interfaz que permite realizar búsquedas existentes (nos referimos al explorador Web), como la interfaz final del usuario de la aplicación. (México, 2012)

Las capas gráficamente de 3 capas serán las siguientes:



Gráfico N. 5 Modelo programación web 3 capas
Autor: www.jtentor.com.ar Fuente: Internet

- El primer se refiere a la capa de presentación que incluye no únicamente el navegador, sino también al servidor web.
- El segundo nivel se refiere a un programa o script de software.
- El tercer nivel proporciona los datos necesarios al segundo para que se pueda ejecutar. Una aplicación Web típica toma datos del usuario final (nos referimos al primer nivel), este enviará al servidor, el cual ejecutará un programa que exista en el segundo o tercer nivel y el resultado será transcrito y mostrado al usuario a través del navegador (primer nivel).

La plataforma tecnológica actual de la cooperativa está basada tecnología Microsoft por lo cual se seleccionó como motor de base de datos SQL Server 2012 y además por ser uno de los motores de más alto rendimiento, escalabilidad, estabilidad y auto protección.



Gráfico N. 6 SQL Server

Autor: www.microsoft.com Fuente: Internet

Como herramienta de desarrollo se utilizó C# en la plataforma Visual Studio 2010 con Framework 4.5.

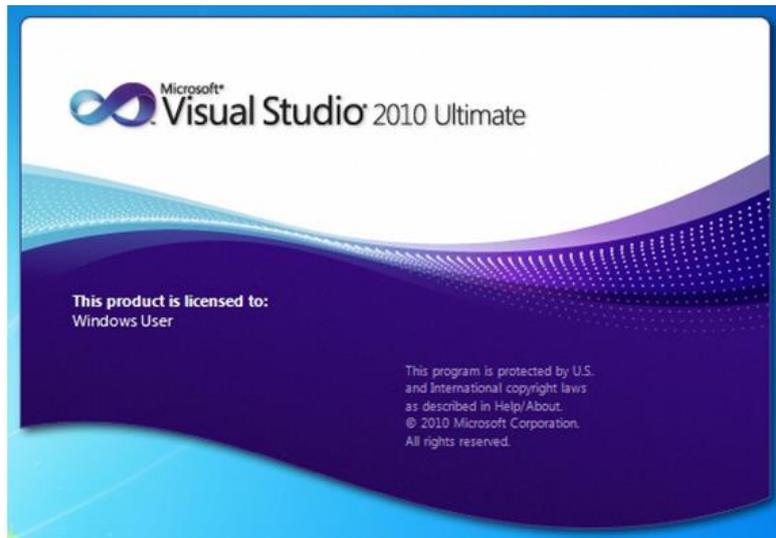


Gráfico N. 7 Visual Studio
Autor: www.microsoft.com Fuente: Internet

3.1.1 Metodologías utilizadas

Para el desarrollo del presente trabajo se utilizó una metodología híbrida entre los modelos cascada y modelamiento UML para los entregables técnicos.

3.1.1.1 Metodología Cascada



Gráfico N. 8 Ciclo de vida cascada
Autor: www.pbworks.com Fuente: Internet

Metodología que claramente describe el inicio del proyecto y en cada fase se genera un entregable que es documentado y validado previo al paso de una siguiente fase dentro del ciclo metodológico.

Con esta técnica de desarrollo de aplicaciones se puede tener control disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo (quién hace qué, cuándo y cómo).

Permite hacer un uso racional de los recursos en cada una de sus fases.

Esta metodología se divide en siete fases:

- ✓ **Inicio** (Autoriza el inicio formal del proyecto)
- ✓ **Análisis** (Se define claramente el alcance con el cliente)
- ✓ **Diseño** (Se realiza el diseño técnico de alto nivel y bajo nivel)
- ✓ **Desarrollo** (Construcción en la herramienta de la solución)
- ✓ **Pruebas** (Proceso de certificación de la aplicación)
- ✓ **Implementación** (Puesta en producción)
- ✓ **Cierre** (Fin formal del proyecto) (Toro, 2013)

3.1.1.2 Modelamiento UML

UML es un lenguaje de modelamiento unificado el cual sirve para esquematizar de manera conceptual y física los artefactos que contempla un sistema, mediante el uso de gráficos simples; lo que se busca es que este esquema permita una fácil interpretación para cualquier persona.

Entre los diagramas más utilizados en UML y reflejados en el desarrollo de la aplicación:

- ✓ **Diagrama de Casos de Uso:** describen la función de cada uno de los actores que interactúan en el sistema.
- ✓ **Diagrama de Contexto:** Representación gráfica de alto nivel que a través de una sola burbuja pretende representar un sistema o los módulos principales del sistema.
- ✓ **Diagrama de Actividades:** sirven para simbolizar el flujo de actividades y la funcionalidad descrita en del sistema.

- ✓ **Diagrama de Clases:** representa el sistema con sus clases, interfaces y sus relaciones.
- ✓ **Diagrama de Secuencia:** muestran la interacción del sistema, siguiendo un flujo transaccional y de negocio.
- ✓ **Diagrama de Despliegue:** extrae los componentes (nodos del sistema) y sus relaciones que conforman el ambiente en donde se ejecutará el sistema. (Larman, 2011)

3.1.1.3 Casos de Usos

El diagrama de casos de uso permite representar como un usuario o actor trabaja con el sistema y explica los procesos en que el acto interactúa.

3.1.1.4 Actor

El Actor es el usuario que trabaja con el sistema.

3.1.1.5 Base de Datos SQL Server

Microsoft SQL Server es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional. Sus lenguajes para consultas son T-SQL y ANSI SQL.

3.1.1.6 .Net (C#).

C# (pronunciado si sharp en inglés) es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270). C# es uno de los lenguajes de programación diseñados para la infraestructura de lenguaje común.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma .NET, similar al de Java, aunque incluye mejoras derivadas de otros lenguajes.

3.1.1.7 PowerDesigner

PowerDesigner, la herramienta de modelamiento número uno de la industria, permite a las empresas, de manera más fácil, visualizar, analizar y manipular metadatos, logrando una efectiva arquitectura empresarial de información.

PowerDesigner para Arquitectura Empresarial también brinda un enfoque basado en modelos, el cual permite alinear al negocio con la tecnología de información, facilitando la implementación de arquitecturas efectivas de información empresarial. Brinda potentes técnicas de análisis, diseño y gestión de metadatos a la empresa.

PowerDesigner combina varias técnicas estándar de modelamiento con herramientas líder de desarrollo, como .NET, Sybase WorkSpace, Sybase Powerbuilder, Java y Eclipse, para darle a las empresas soluciones de análisis de negocio y de diseño formal de base de datos. Además trabaja con más de 60 bases de datos relacionales.

3.2 Desarrollo de la propuesta

Como propuesta de solución a la actual problemática de la cooperativa se plantea el desarrollo de un sistema automatizado con el siguiente alcance funcional.

3.2.1 Alcance funcional del proyecto

Crear una aplicación WEB que permita a la cooperativa COOPROGRESO S.A automatizar los procesos de pruebas funcionales y de esta manera también se refleje una reducción de los defectos encontrados en el proceso de pruebas.

Básicamente, la herramienta permitirá administrar los proyectos, fases o ciclos, casos de pruebas y la gestión de errores encontrados en el proceso de certificación.

Con la herramienta los casos, errores y estado de avance de pruebas se podrán visualizar en línea desde cualquier computador que tenga acceso a la intranet de la empresa.

El sistema, se encuentra conformado por los siguientes módulos:

3.2.1.1 Parámetros Generales

Módulo para la configuración de la aplicación bajo las características de la empresa, el cual consta de los siguientes sub-módulos:

- ✓ **Catalogo**

Permite administrar (crear, modificar, eliminar) los catálogos utilizados por el sistema.

✓ **Flujo Estado**

Permite parametrizar los flujos de aprobación del sistema en función de los perfiles y procesos de funcionamiento.

✓ **Pantallas**

Permite por cada pantalla del sistema parametrizar los combos y controles.

3.2.1.2 Seguridades

Módulo para la administración de usuarios, perfiles y opciones habilitados para el uso de la aplicación y el mismo consta de los siguientes sub-módulos:

✓ **Usuarios**

Permite administrar (crear, modificar, eliminar) los usuarios habilitados para uso del sistema.

✓ **Perfiles**

Permite por perfil administrar (crear, modificar, eliminar) las opciones que los usuarios tendrán acceso a la aplicación, a través de esta misma opción se permitirá habilitar los controles en las pantallas por el perfil y asignar el flujo de aprobación por perfil.

✓ **Menús**

Permite administrar (crear, modificar, eliminar) las opciones existentes en el sistema.

3.2.1.3 Biblioteca de Casos

Sección que permitirá administrar (crear, modificar, eliminar) los planes de pruebas existentes en el sistema.

Los planes de pruebas están conformados por los casos y pasos de pruebas existentes por cada aplicación.

En esta opción se podrá visualizar los planes existentes y lo cual permitirá reutilizar en cada proyecto. (Pello, 2008)

3.2.1.4 Proyectos

Módulo principal del sistema en el cual permite administrar todas las opciones existentes para la creación del proyecto, administración de intervinientes por proyectos, administrar los ciclos y sus respectivos casos por proyectos, administrar el plan de ejecución, aprobar el plan del proyecto, visualizar las actividades pendientes por proyecto y el proceso de ejecución de los casos de pruebas.

✓ Proyectos

Permite administrar (crear, modificar, eliminar) los proyectos y en el cual se incluye información básica del mismo.

✓ Intervinientes

Permite administrar (crear, modificar, eliminar) por perfil los intervinientes que estarán anclados al proyecto creado.

✓ Ciclos Casos

Permite administrar (crear, modificar, eliminar) por proyecto y por ciclo creado los casos de pruebas que se ejecutarán en cada ciclo, en esta opción se visualizarán los planes existentes y aprobados.

✓ Plan de Ejecución

Permite administrar (crear, modificar, eliminar) la ejecución del caso de prueba en el cual se deberá asignar una fecha de inicio y fin del caso con sus respectivo ingeniero de prueba asignado.

✓ Aprobación

Permite la aprobación de los planes creados en cada proyecto y por cada ciclo.

✓ Ejecución

Permite la ejecución de los casos de pruebas creados y aprobados en el plan de ejecución.

✓ Lista de Tareas

Permite visualizar las actividades pendientes para cada usuario, en esta opción se podrá identificar las aprobaciones y defectos pendientes por cada proyecto.

✓ **Defectos**

Permite visualizar el estado de los defectos en el ciclo de ejecución de cada proyecto.

3.2.1.5 Reportes

Sección que permitirá ver el estado de avance y de errores por proyecto y ciclo.

✓ **Consolidado de Errores**

Permitirá visualizar por proyecto y ciclo el número de casos y errores encontrados en la ejecución de los casos de pruebas.

Se mostrará consolidadamente el número de errores por estado del error.

✓ **Detalle de Errores**

Permitirá visualizar por proyecto y ciclo el número de casos y errores a nivel de detalle del error y se mostrará información tales como: descripción, estado, asignado, fecha de reporte, prioridad y tipo de error presentado en el defecto creado.

✓ **Estado del Proyecto**

Permitirá visualizar de una manera consolidada/gerencial el estado de ejecución del proyecto.

Esta opción mostrar información de los casos ejecutados, casos ejecutados fallidos, casos pendientes de ejecución.

3.2.1.6 Salir

Permite abandonar la aplicación.

3.2.2 Desarrollo de la propuesta

3.2.3 Cronograma

Las actividades para el desarrollo del proyecto se realizarán en función de las siguientes fases, actividades y fechas propuestas.

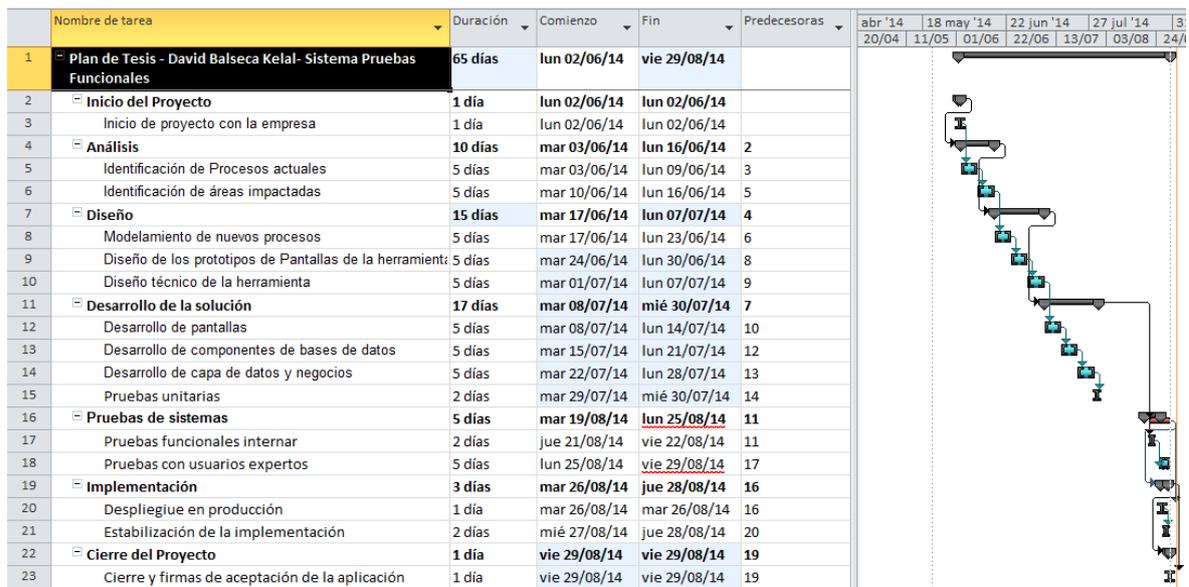


Gráfico N. 9 Cronograma de propuesta
 Autor: David Balseca Fuente: Investigador

3.2.4 Requisitos del sistema

Para el análisis, diseño y desarrollo de la aplicación se utilizan varias herramientas como:

- ✓ Power Designer.- Modelador de Base de datos
- ✓ SQL Server2012 – Manejador de Base de datos
- ✓ Microsoft Visual Studio 2010 – Framework de trabajo
- ✓ Microsoft Visual C# - Lenguaje de Programación
- ✓ Microsoft Office Visio 2010.- Aplicación de dibujos y diagramas

3.2.5 Modelo de negocio

Los modelos de negocios que serán descritos en el presente trabajo básicamente se centraran en 3 procesos principales los cuales son:

- ✓ Proceso de creación de proyectos y sus sub-procesos asociados.
- ✓ Proceso de ejecución de pruebas
- ✓ Proceso de Flujo de errores

3.2.6 Diagrama de actividades

3.2.6.1 Proceso de creación de proyectos y sus sub-procesos asociados.

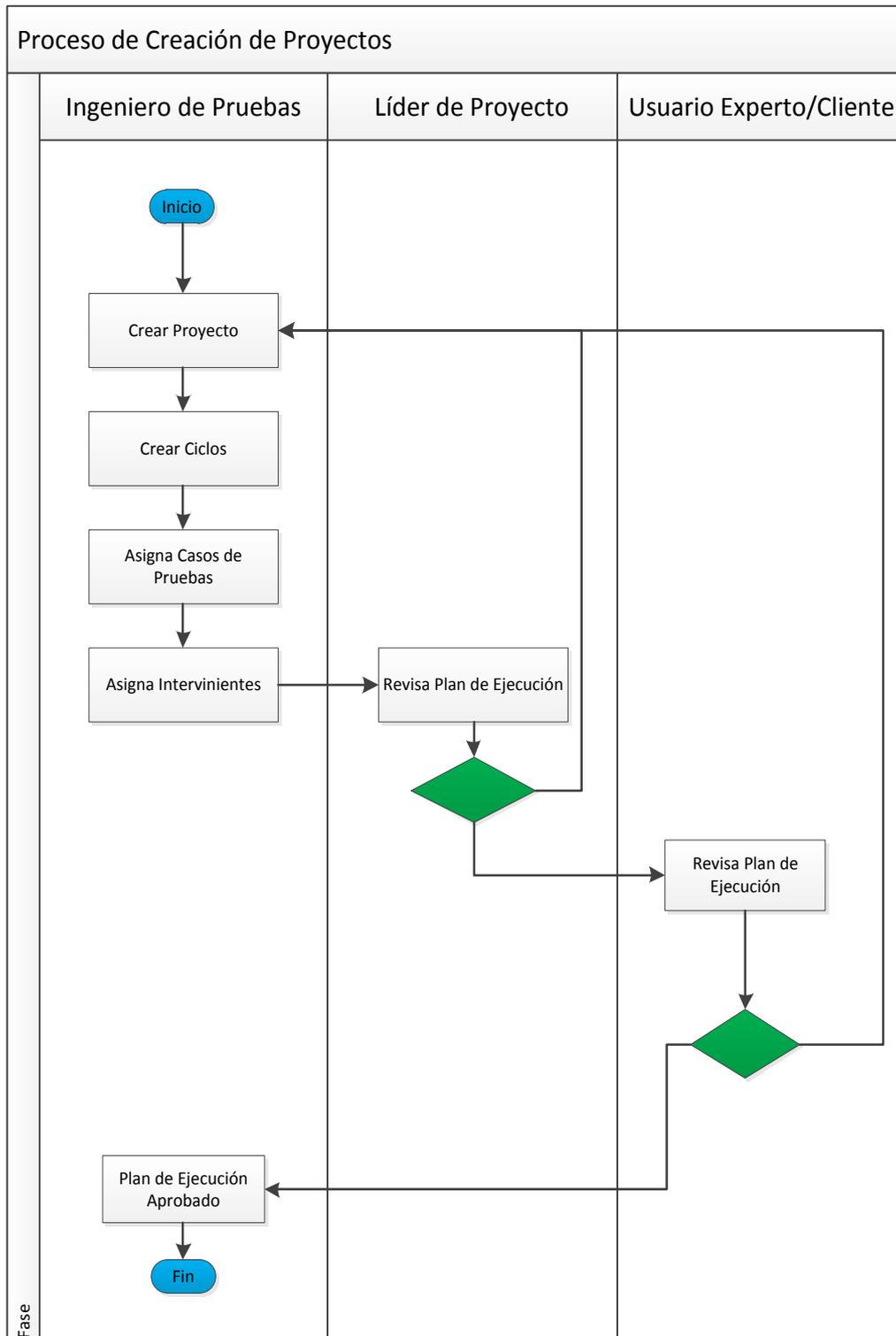


Gráfico N. 10 Diagrama de actividades Creación de Proyectos
 Autor: David Balseca Fuente: Investigador

3.2.6.2 Diagrama de Actividades – Proceso de ejecución de pruebas.

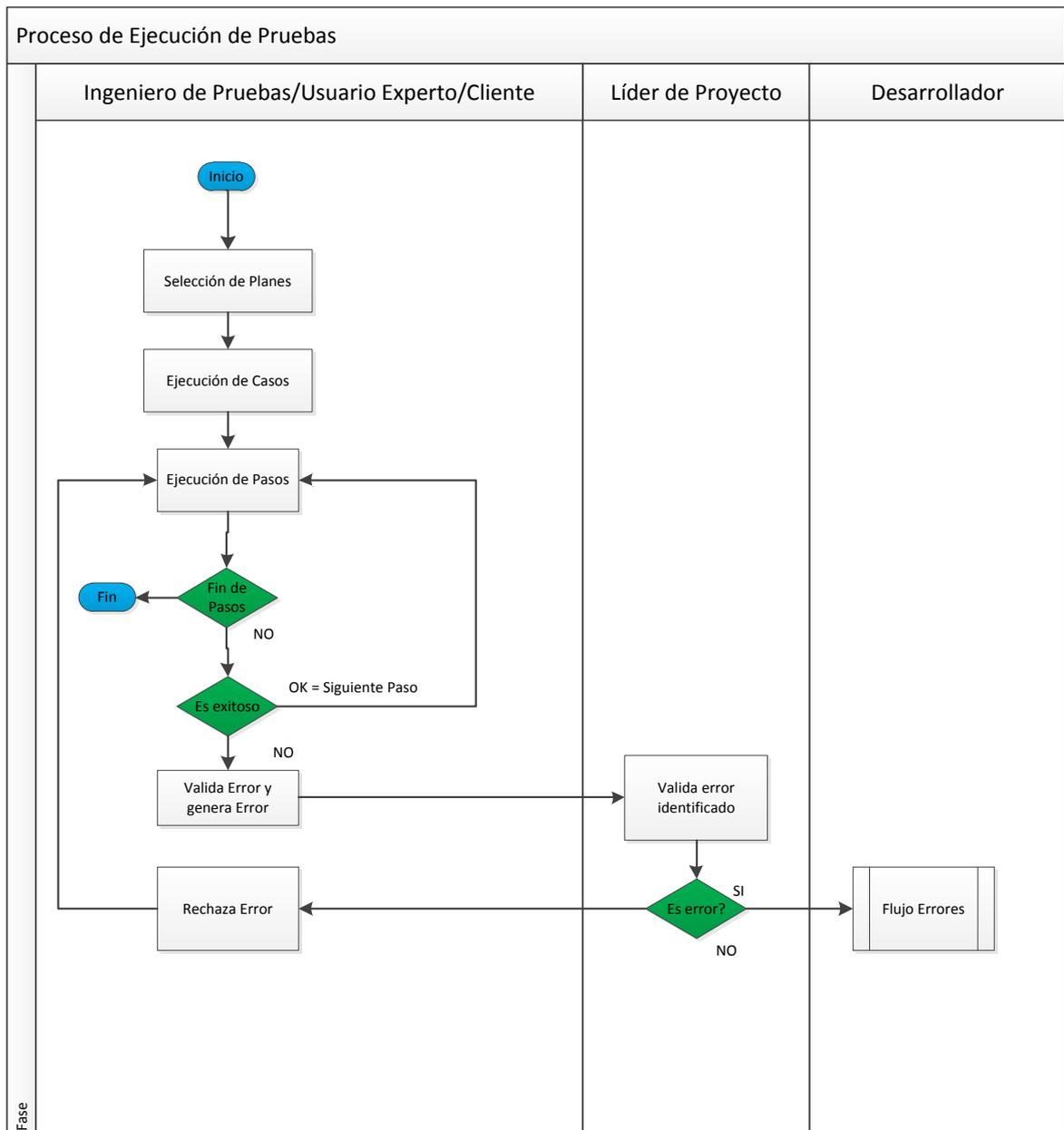


Gráfico N. 11 Diagrama de actividades Ejecución de pruebas
 Autor: David Balseca Fuente: Investigador

3.2.6.3 Diagrama de Actividades – Proceso de flujos de errores.

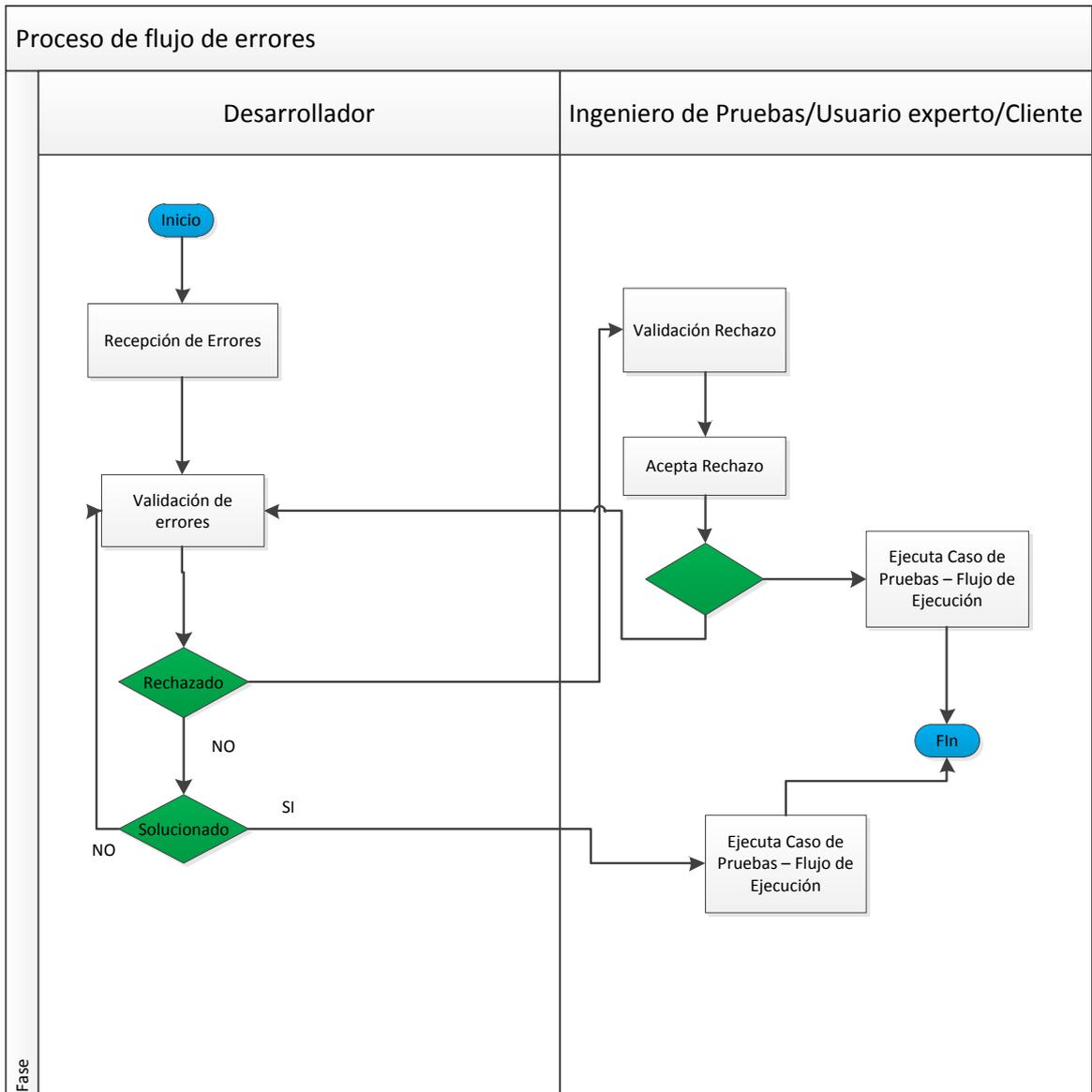


Gráfico N. 12 Diagrama de actividades Flujo de errores
 Autor: David Balseca Fuente: Investigador

3.2.7 Diagrama de contexto

El diagrama de contexto enseña con flujos de datos las relaciones existentes entre los actores y agentes que pueden ser externos y a su vez con la aplicación.

Estos tipos de diagramas muestran a alto nivel las entradas y salidas hacia los actores y/o agentes externos que trabajan en el sistema.

3.2.7.1 Diagrama de contexto principales del Sistema

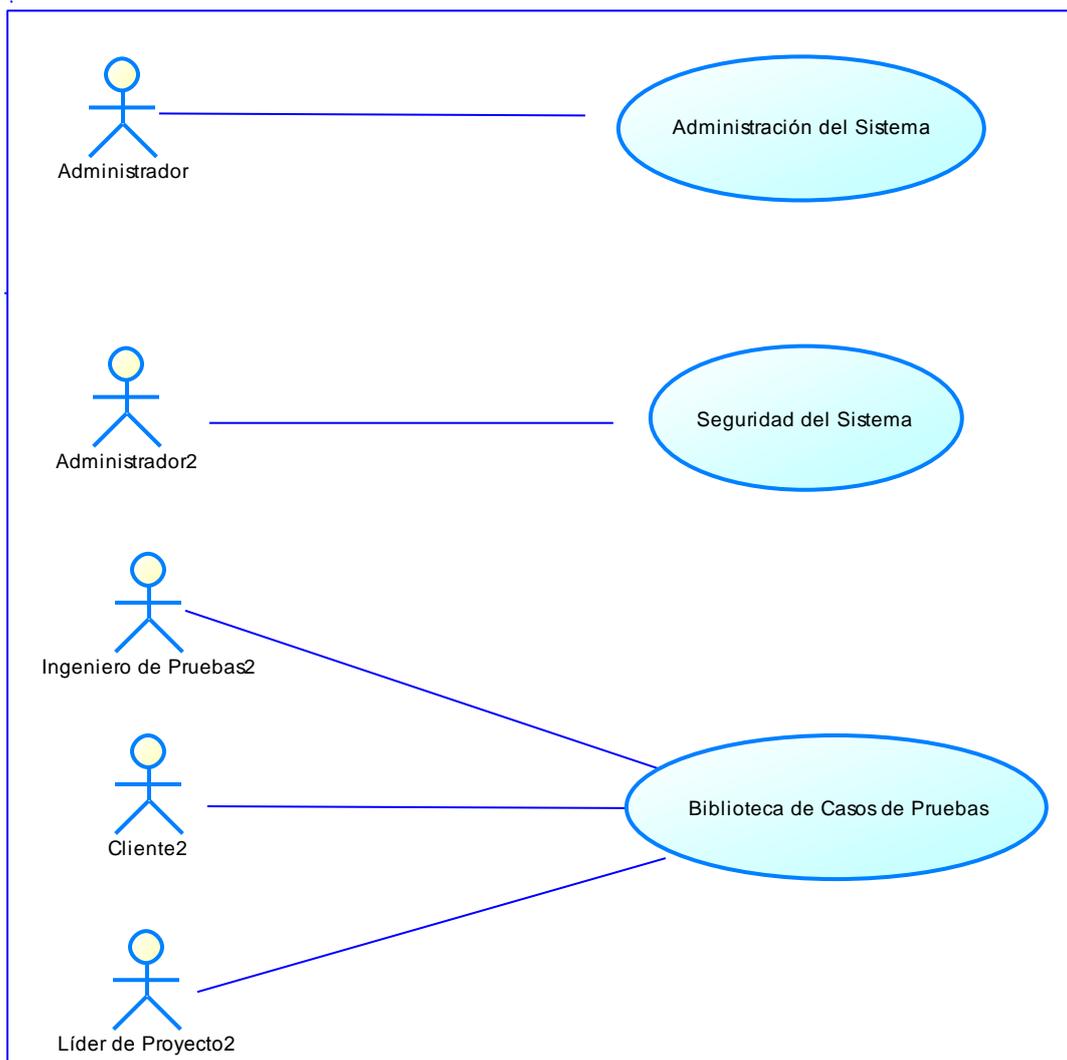


Gráfico N. 13 Diagrama de contexto 1
Autor: David Balseca Fuente: Investigador

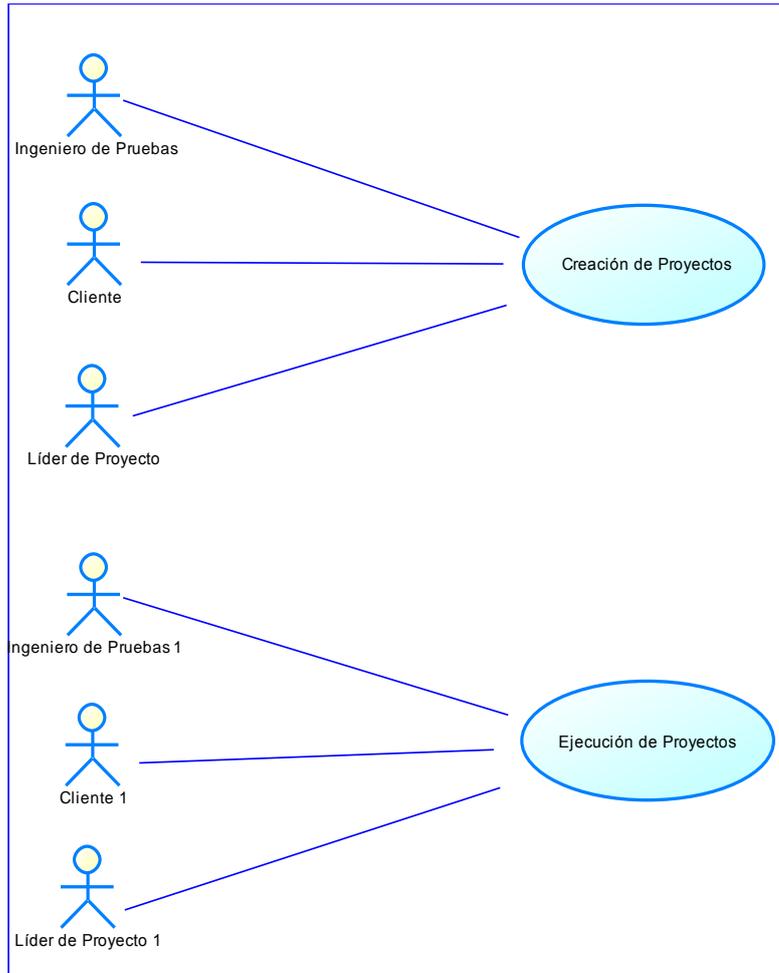


Gráfico N. 14 Diagrama de contexto 2
 Autor: David Balseca Fuente: Investigador

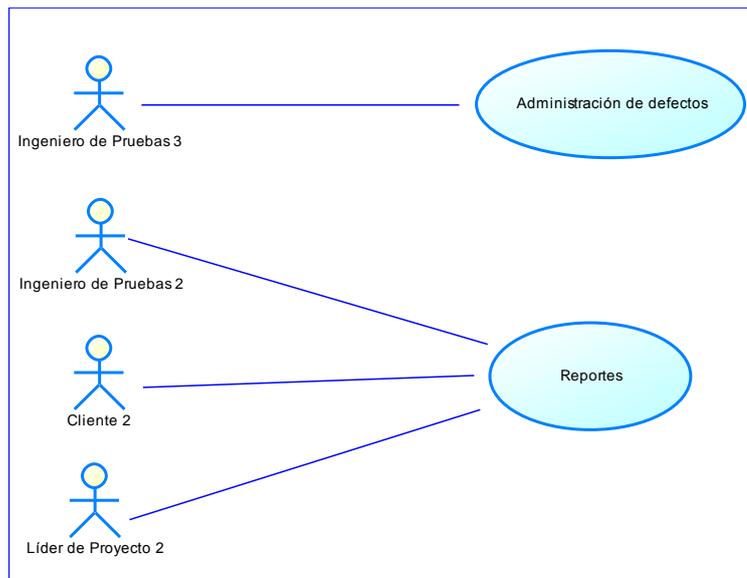


Gráfico N. 15 Diagrama de contexto 3
 Autor: David Balseca Fuente: Investigador

3.2.8 Diagrama de casos de usos

Los diagramas de Casos de Uso registran las diferentes actividades que el actor o usuario realizan en cada uno de los procesos, que se muestran a continuación.

3.2.8.1 Caso de Uso Módulo: Administración del sistema

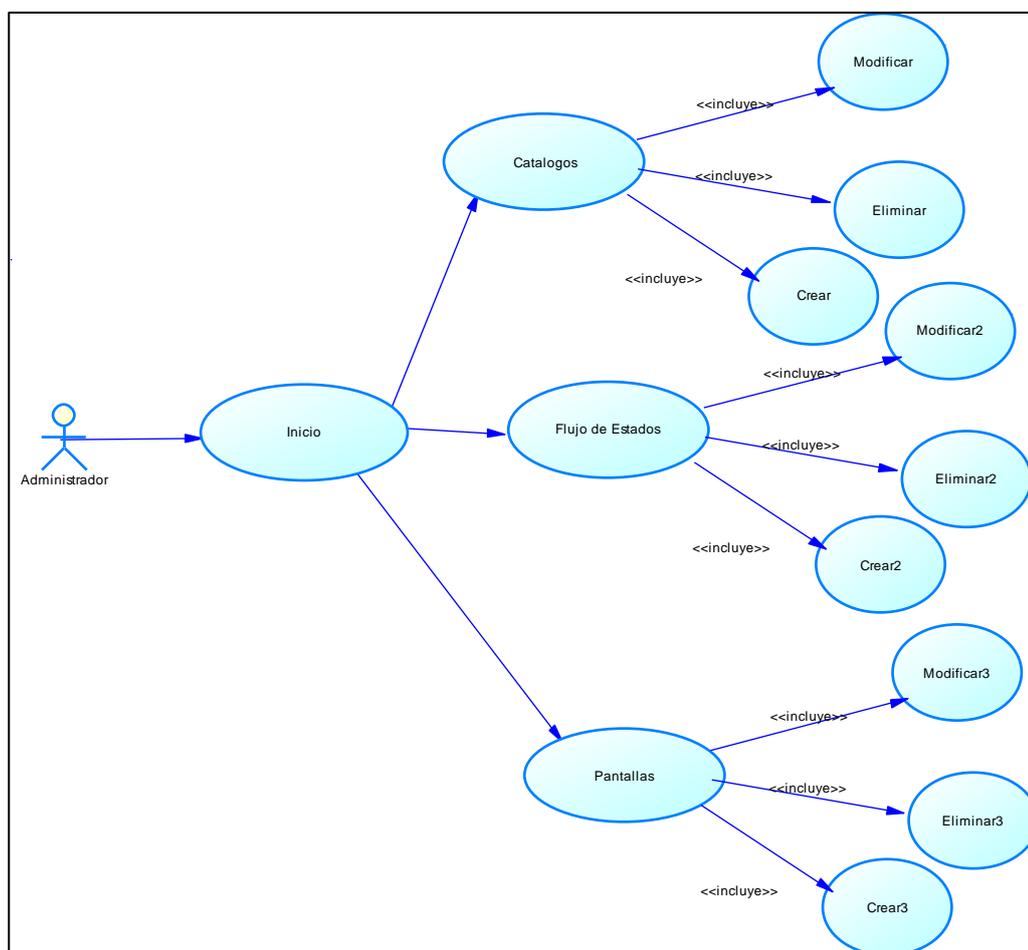


Gráfico N. 16 Casos de Usos – Administración del Sistema
 Autor: David Balseca Fuente: Investigador

Caso de Uso: Administrador del Sistema - Catálogos

Identificador:	CuAd01
Nombre del caso de uso:	Administrador del Sistema - Catálogos
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Administrador del sistema
Descripción:	Permite parametrizar la tabla de catálogos que utilizará el sistema para su correcto funcionamiento
Precondiciones:	El usuario debe estar logeado y con

	autorización al menú Parámetros generales
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario
02	Ingresa clave
03	Selecciona del menú la opción catálogo
04	Seleccionar opción NUEVO
05	Ingresa datos requeridos como son nombre y descripción del catálogo
06	Presiona botón guardar
07	El catálogo se grabó exitosamente
08	Selecciona menú administración del detalle del catálogo
09	Ingresa datos requeridos
10	Selecciona botón guardar
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
05.1	Registro en el catálogo ya existe
05.2	Datos ingresados no validos
06.1	OK
06.2	Cancelar la creación
10.1	OK
10.2	Cancelar la creación

Tabla N. 2 Caso de uso – Catálogos
Autor: David Balseca Fuente: Investigador

Caso de Uso: Administrador del Sistema – Flujo Estado

Identificador:	CuAd02
Nombre del caso de uso:	Administrador del Sistema – Flujo Estado
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Administrador del sistema
Descripción:	Permite parametrizar el nivel de aprobación que utilizará en cada pantalla asociada a un proceso que requiera nivel de

	aprobación superior.
Precondiciones:	El usuario debe estar logeado y con autorización al menú Parámetros generales
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario
02	Ingresa clave
03	Selecciona del menú la opción FLUJO ESTADO
04	Selecciona pantalla del combo pantalla
05	Permite generar uno nuevo
06	Permite cambiar un nivel en el árbol de aprobación
07	Selecciona para ambos casos el proceso, control, detalle y selecciona el nivel a que desea que la aprobación permita seguir, pudiendo ser a: rechazado o aprobado.
08	Selecciona botón modificar
09	Mensaje de grabación
10	El registro se actualizó correctamente
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
09.1	Registro en el catálogo ya existe
09.2	Datos ingresados no válidos

Tabla N. 3 Caso de uso – Flujo estados
Autor: David Balseca Fuente: Investigador

Caso de Uso: Administrador del Sistema – Pantalla

Identificador:	CuAd03
Nombre del caso de uso:	Administrador del Sistema – Pantalla
Creado por:	David Balseca Kelal

Fecha de creación:	12/08/2014
Actores:	Administrador del sistema
Descripción:	Permite parametrizar a nivel de pantalla los controles y combobox de manera automática para cargar los datos a nivel de parámetros
Precondiciones:	El usuario debe estar logeado y con autorización al menú Parámetros generales
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresar usuario
02	Ingresar clave
03	Seleccionar del menú la opción PANTALLA
04	Ingresar nombre de la pantalla a parametrizar
05	Seleccionar botón guardar
06	Mensaje registro se grabó exitosamente
07	Se presiona menú administración del combobox
08	Se ingresan los datos requeridos
09	Se presiona botón guardar
10	Se presiona menú administración del control
11	Se ingresan nombre del control de la pantalla
12	Se presiona botón guardar
13	Mensaje registro se grabó exitosamente
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
05.1	OK
05.2	Cancelar
12.1	OK
12.2	Cancelar

Tabla N. 4 Caso de uso – Pantallas
Autor: David Balseca Fuente: Investigador

3.2.8.2 Caso de Uso Módulo: Seguridad del sistema

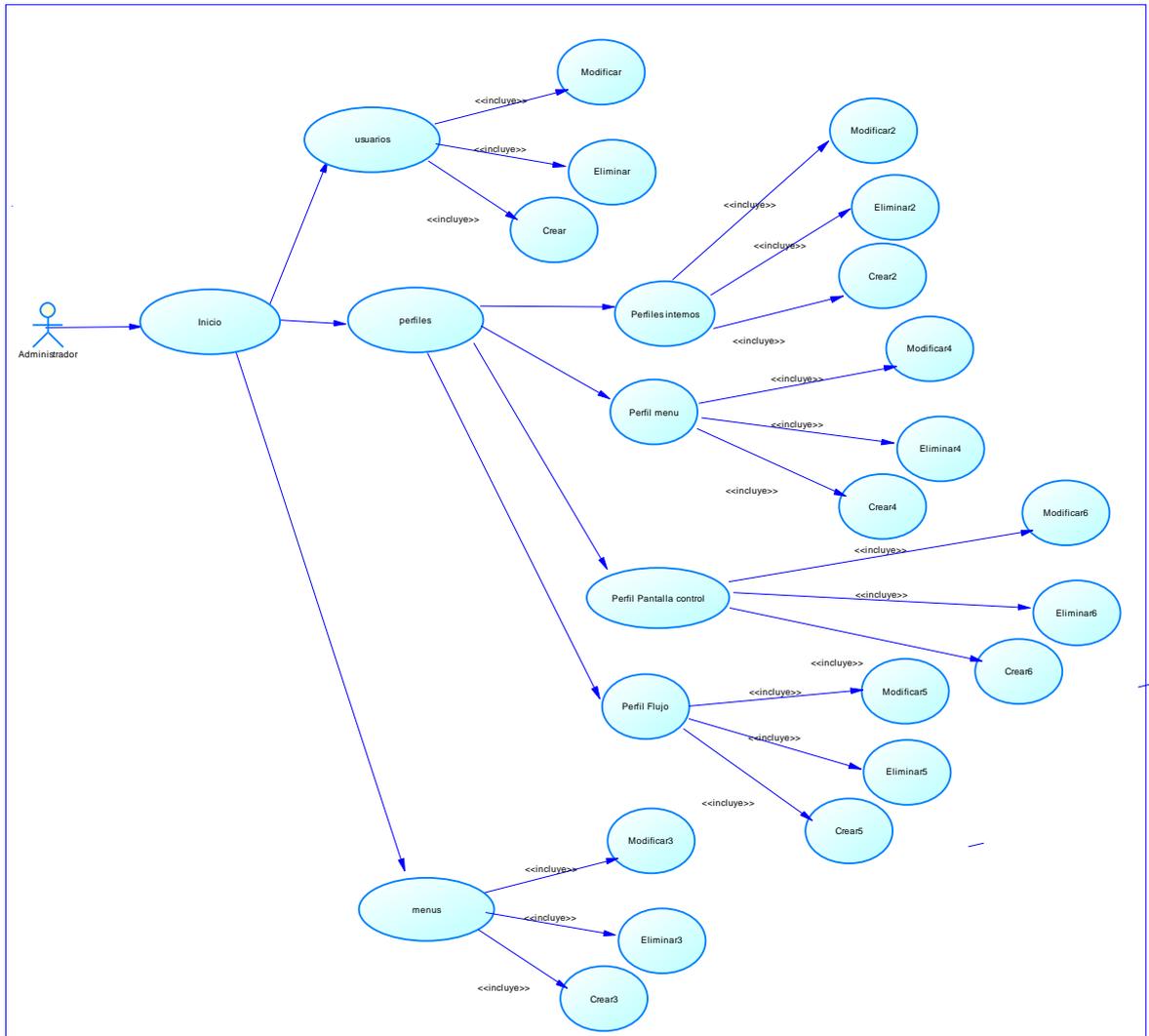


Gráfico N. 17 Casos de Usos – Seguridad del Sistema
 Autor: David Balseca Fuente: Investigador

Caso de Uso: Seguridad – Usuarios

Identificador:	CuAd04
Nombre del caso de uso:	Administrador del Sistema – Usuarios
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Administrador del sistema
Descripción:	Permite parametrizar los usuarios que podrán utilizar el sistema
Precondiciones:	El usuario debe estar logeado y con autorización al menú de seguridad
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	

PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú usuarios
03	Selecciona opción nuevo
04	Ingresa los datos requeridos en la pantalla
05	Se selecciona el perfil del combo perfil
06	Se presiona botón Guardar
07	Mensaje de confirmación de grabar
08	El registro se grabó exitosamente
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para grabar
07.2	Cancela la grabación

Tabla N. 5 Caso de uso – Seguridad usuarios
Autor: David Balseca Fuente: Investigador

Caso de Uso: Seguridades – Perfiles

Identificador:	CuAd05
Nombre del caso de uso:	Administrador del Sistema – Perfiles
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Administrador del sistema
Descripción:	Permite parametrizar los perfiles y las opciones y pantallas que podrá realizar en el sistema
Precondiciones:	El usuario debe estar logeado y con autorización al menú de seguridad
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú perfiles
03	Selecciona opción nuevo
04	Ingresa los datos requeridos en la pantalla
05	Mensaje de confirmación de grabación

06	Registros se grabó exitosamente
07	Se selecciona opción perfil menú
08	Aparecerá el árbol de menús para asociar al perfil
09	Se debe seleccionar el menú a activar dando doble click
10	Se selecciona visible o editable en función de la definición del perfil
11	Se presiona Guardar
12	Mensaje de confirmación
13	Luego se selecciona opción perfil pantalla control
14	Se selecciona la pantalla que ese perfil podrá manipular
15	Aparecerá una lista de todos los controles de la pantalla
16	Se selecciona los controles de la pantalla que ese perfil podrá manipular
17	Se presiona guardar
18	Luego se debe seleccionar opción perfil flujo
19	Se debe seleccionar la pantalla
20	Se mostrará los niveles de aprobación para la pantalla y perfil creado
21	Se selecciona los niveles que podrá realizar el perfil
22	Se presiona botón guardar
23	Mensaje de confirmación
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
05.1	Ok para grabar
05.2	Cancela la grabación
11.1	Ok para grabar
11.2	Cancela la grabación
22.1	Ok para grabar
22.2	Cancela la grabación

Tabla N. 6 Caso de uso – Seguridad perfiles
Autor: David Balseca Fuente: Investigador

Caso de Uso: Seguridades – Menús

Identificador:	CuAd06
Nombre del caso de uso:	Administrador del Sistema – Menús
Creado por:	David Balseca Kelal

Fecha de creación:	12/08/2014
Actores:	Administrador del sistema
Descripción:	Permite administrar los menús del sistema total.
Precondiciones:	El usuario debe estar logeado y con autorización al menú de seguridad
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú "menú"
03	Selecciona opción nuevo
04	Ingresa los datos requeridos en la pantalla
05	Selecciona opciones visible y/o editable
06	Se presiona botón guardar
07	Mensaje de confirmación
08	Registros se grabó exitosamente
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para grabar
07.2	Cancela la grabación

Tabla N. 7 Caso de uso – Seguridad menú
Autor: David Balseca Fuente: Investigador

3.2.8.3 Caso de Uso Módulo: Biblioteca de Casos

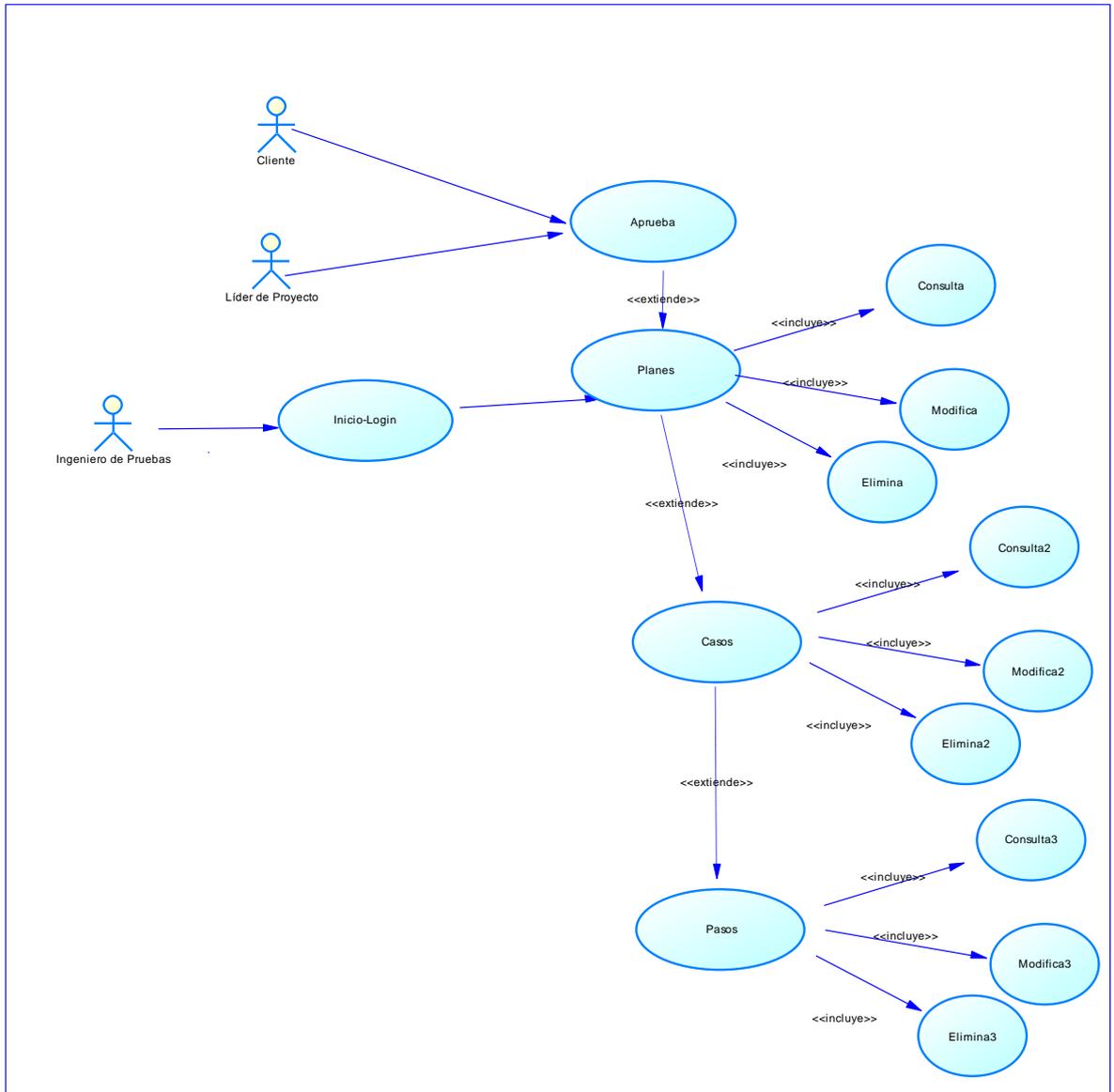


Gráfico N. 18 Casos de Usos – Biblioteca de casos
 Autor: David Balseca Fuente: Investigador

Caso de Uso: Biblioteca de Casos

Identificador:	CuAd07
Nombre del caso de uso:	Biblioteca de Casos
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Cliente Líder de Proyecto Ingeniero de Pruebas
Descripción:	Permite administrar la biblioteca de casos que tendrá en sistema para reutilización
Precondiciones:	El usuario debe estar logeado y con

	autorización al menú de seguridad
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú plan de pruebas
03	Selecciona opción nuevo
04	Ingresa los datos requeridos en la pantalla planes
05	Se selecciona a que aplicación dicho plan será añadido seleccionando del combo de aplicaciones
06	Se presiona botón guardar
07	Mensaje de confirmación
08	Registros se grabó exitosamente
09	Se ingresa los datos de la pantalla casos
10	Se selecciona prioridad, perfil que ejecutará los casos, el nivel de complejidad, tipo de prueba en cada combo existente
11	Se presiona botón guardar
12	Mensaje de confirmación
13	Se selecciona pestaña pasos
14	Se ingresan los datos de cada paso a crear
15	Se presionado botón guardar
16	Mensaje de confirmación
17	En la misma pantalla se presiona botón nuevo para crear otro paso de dicho caso
18	Y este evento se vuelve repetitivo hasta que termine la creación de los pasos
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para grabar
07.2	Cancela la grabación
12.1	Ok para grabar
12.2	Cancela la grabación
16.1	Ok para grabar
16.2	Cancela la grabación

Tabla N. 8 Caso de uso – biblioteca de casos
Autor: David Balseca Fuente: Investigador

3.2.8.4 Caso de Uso Módulo: Creación de Proyectos

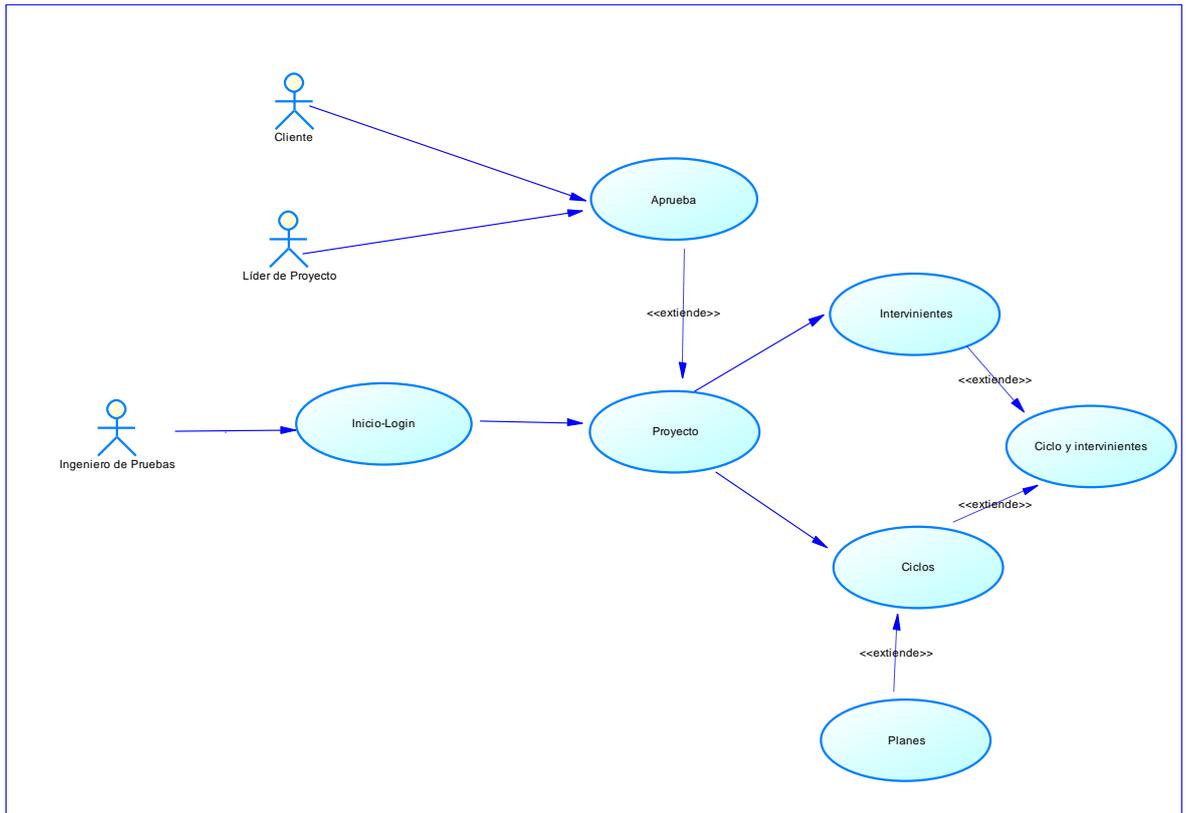


Gráfico N. 19 Casos de Usos – Creación de proyectos
 Autor: David Balseca Fuente: Investigador

Caso de Uso: Proyectos

Identificador:	CuAd08
Nombre del caso de uso:	Proyectos
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de pruebas
Descripción:	Permite crear los proyectos que serán ejecutados con sus respectivos ciclos a realizar en la fase de pruebas
Precondiciones:	El usuario debe estar logeado y con autorización al menú proyectos
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú intervinientes

03	Selecciona opción nuevo
04	Ingresar los datos requeridos en la pantalla planes
05	Se ingresan las fechas de inicio y fin planificadas del proyecto
06	Se presiona botón guardar
07	Mensaje de confirmación
08	Luego se selecciona opción ciclo de proyectos
09	Se ingresan los ciclos requeridos en el proyecto y se indica la fecha de inicio y fin tentativo
10	Se selecciona el estado
11	Se presiona botón guardar
12	Mensaje de confirmación
13	Por cada ciclo que se selecciona las mismas opciones
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para grabar
07.2	Cancela la grabación
12.1	Ok para grabar
12.2	Cancela la grabación

Tabla N. 9 Caso de uso – Proyectos
Autor: David Balseca Fuente: Investigador

Caso de Uso: Intervinientes

Identificador:	CuAd09
Nombre del caso de uso:	Intervinientes
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de pruebas
Descripción:	Permite administrar los intervinientes que se asociarán a un proyecto
Precondiciones:	El usuario debe estar logeado y con autorización al menú proyectos
Post condiciones:	El registro se ha grabado exitosamente

FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresar usuario y clave
02	Selecciona menú intervinientes
03	Se selecciona el proyecto a parametrizar
04	Se selecciona el perfil que desea incorporar al proyecto
05	Del combo intervinientes se seleccionan los usuarios con dicho perfil
06	Se presiona botón guardar
07	Mensaje de confirmación
08	El interviniente se grabó exitosamente
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para grabar
07.2	Cancela la grabación

Tabla N. 10 Caso de uso – Intervinientes
Autor: David Balseca Fuente: Investigador

Caso de Uso: Ciclos Casos

Identificador:	CuAd10
Nombre del caso de uso:	Ciclos Casos
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de pruebas
Descripción:	Permite asociar los casos a cada ciclo creado
Precondiciones:	El usuario debe estar logeado y con autorización al menú proyectos
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresar usuario y clave

02	Selecciona menú ciclo casos
03	Se selecciona el proyecto a parametrizar
04	Se selecciona el ciclo en donde se anclarán los casos
05	Se selecciona el plan para visualizar los planes habilitados para ejecutar
06	Para caso que se desee seleccionar se debe escoger la opción de check ok para asociarlo al plan y ciclo respectivo
07	Si desea eliminar se selecciona el botón eliminar
08	Este proceso se realizará por cada caso
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para eliminar
07.2	Cancela para eliminar la acción

Tabla N. 11 Caso de uso – Ciclos casos
Autor: David Balseca Fuente: Investigador

Caso de Uso: Plan de Ejecución

Identificador:	CuAd11
Nombre del caso de uso:	Plan de Ejecución
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de pruebas
Descripción:	Permite asociar por cada caso de prueba en el proyecto asociar el ingeniero de pruebas que ejecutará cada caso
Precondiciones:	El usuario debe estar logeado y con autorización al menú proyectos
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú plan de ejecución

03	Se selecciona el proyecto a parametrizar
04	Se selecciona el ciclo en donde se anclaran los intervinientes a cada caso
05	Por cada caso que se muestre se ingresa fecha de ingreso, fecha fin y ingeniero que realizará la prueba al caso seleccionado
06	Se presiona botón guardar
07	Mensaje de confirmación
08	Registro se modificó exitosamente
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para eliminar
07.2	Cancela para eliminar la acción

Tabla N. 12 Caso de uso – Plan de ejecución
Autor: David Balseca Fuente: Investigador

Caso de Uso: Aprobación

Identificador:	CuAd12
Nombre del caso de uso:	Aprobación
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Cliente Líder de Proyecto
Descripción:	Permite aprobar los ciclos y casos relacionados al proyecto y sus respectivos ciclos
Precondiciones:	El usuario debe estar logeado y con autorización al menú proyectos
Post condiciones:	El registro se ha grabado exitosamente
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú aprobación
03	Se selecciona el proyecto a parametrizar
04	Se selecciona el ciclo que se desea aprobar

05	Se mostraran los casos anclados a ese ciclo de proyecto para revisión
06	Se debe seleccionar la opción: por aprobar, aprobar, rechazar en función del perfil del usuario
07	Se presiona botón guardar
07	Mensaje de confirmación
08	Registro se modificó exitosamente
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para eliminar
07.2	Cancela para eliminar la acción

Tabla N. 13 Caso de uso – Aprobación
 Autor: David Balseca Fuente: Investigador

3.2.8.5 Caso de Uso Módulo: Ejecución

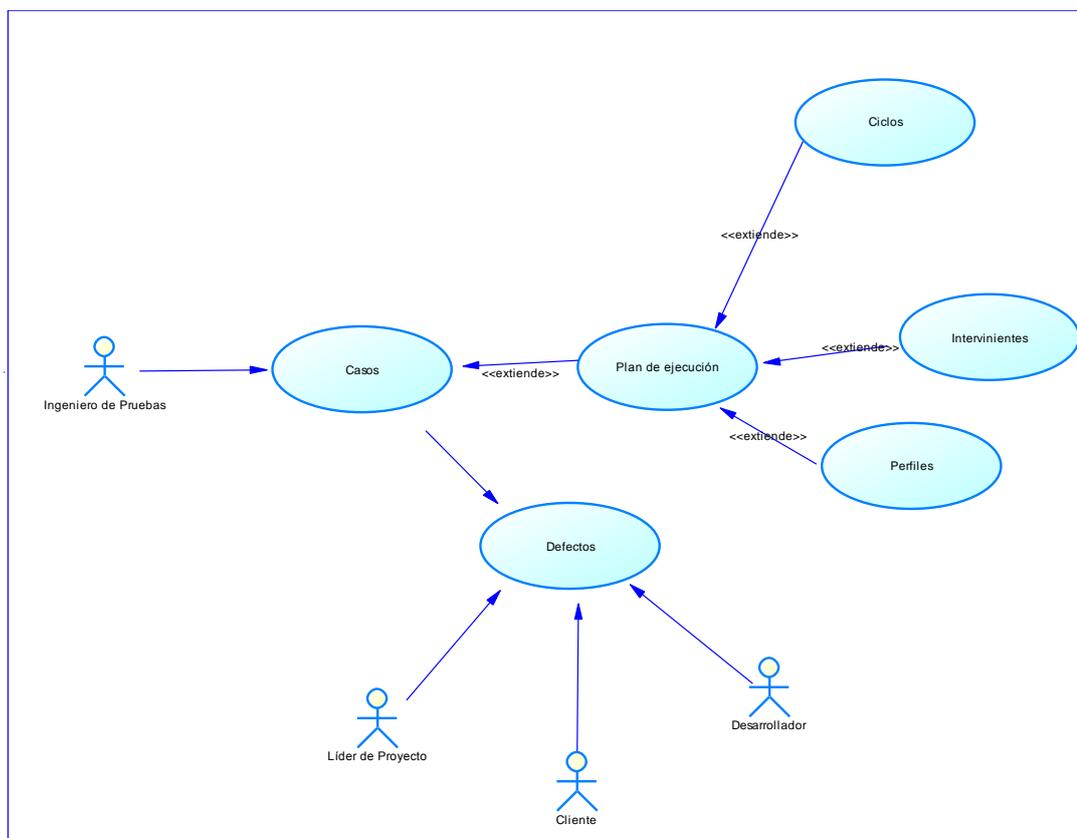


Gráfico N. 20 Casos de Usos – Ejecución
 Autor: David Balseca Fuente: Investigador

Caso de Uso: Ejecución

Identificador:	CuAd13
Nombre del caso de uso:	Ejecución
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de Prueba Cliente Líder de Proyecto
Descripción:	Permite ejecutar los casos asociados a un proyecto
Precondiciones:	El usuario debe estar logeado y con autorización al menú proyectos
Post condiciones:	El caso debe terminar en exitoso o fallido
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú ejecución
03	Se selecciona el proyecto a ejecutar
04	Se mostrará los ciclos creados para ese proyecto
05	Se debe dar click sobre el ciclo que desea visualizar
06	El sistema mostrará los casos asociados al usuario logeado
07	Para ejecutar el caso se da click sobre el ícono ejecución
07	El sistema solicitará confirmación de ejecución
08	Se mostrará una nueva pantalla con el paso a ejecutar
09	Por cada caso se ingresará alguna observación
10	Se debe seleccionar el estado del caso
11	Se debe presionar la flecha próximo para ejecutar el siguiente paso y así hasta el final de los pasos del caso
12	Si desea anclar un defecto se seleccionará la pestaña defectos
13	Se incluye los datos requeridos de la pantalla
14	Se deberá ingresar la descripción del error encontrado
15	Si desea anexar un archivo al defecto se deberá seleccionar la pestaña captura de

	pantalla
16	Se debe presionar el botón seleccionar archivo
17	Luego se debe presionar el botón subir imagen
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para continuar
07.2	Cancelar para la no ejecución

Tabla N. 14 Caso de uso – Ejecución
Autor: David Balseca Fuente: Investigador

Caso de Uso: Lista de tareas

Identificador:	CuAd14
Nombre del caso de uso:	Lista de tareas
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de Prueba Cliente Líder de Proyecto
Descripción:	Permite visualizar las actividades pendientes del usuario logeado
Precondiciones:	El usuario debe estar logeado y con autorización al menú proyectos
Post condiciones:	Vaciar actividades pendientes
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú Lista de tareas
03	El sistema le mostrará las actividades que el usuario tiene pendiente
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente

Tabla N. 15 Caso de uso – Lista de Tareas
Autor: David Balseca Fuente: Investigador

Caso de Uso: Defectos

Identificador:	CuAd15
Nombre del caso de uso:	Defectos
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de Prueba Cliente Líder de Proyecto Desarrollador
Descripción:	Permite visualizar los defectos anclados a los usuarios y cambiar su estado en función del análisis realizado
Precondiciones:	El usuario debe estar logeado y con autorización al menú proyectos
Post condiciones:	El defecto debe quedar cambiado de estado
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú defectos
03	Se selecciona el proyecto que desea visualizar
04	Se mostrará los ciclos creados para ese proyecto
05	Se debe dar click sobre el ciclo que desea visualizar los defectos
06	El sistema mostrará los defectos asociados al usuario logeado
07	Se debe seleccionar click en botón seleccionar para visualizar el detalle del defecto
08	Para ingresar una observación y cambiar de estado se debe presionar la etiqueta observaciones
09	El sistema mostrará una pantalla para cambiar de estado e ingresar una observación
10	Se presiona aceptar para grabar nuevo estado y observación ingresada
11	Registro grabado exitosamente
12	Para visualizar las pantallas anexas al defecto se debe seleccionar la pestaña captura de pantalla
13	Se mostrará los archivos adjuntos

14	Dar click en los archivos y el sistema le solicitará ruta de descarga del archivo seleccionado
15	Si desea visualizar el historial de estados del defecto se debe presionar click en la pestaña historial
16	El sistema mostrará todos los estados
17	Si desea visualizar un historial a detalle deberá dar click en detalle
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente
07.1	Ok para continuar
07.2	Cancelar para la no ejecución

Tabla N. 16 Caso de uso – Defectos
Autor: David Balseca Fuente: Investigador

3.2.8.6 Caso de Uso Módulo: Reportes

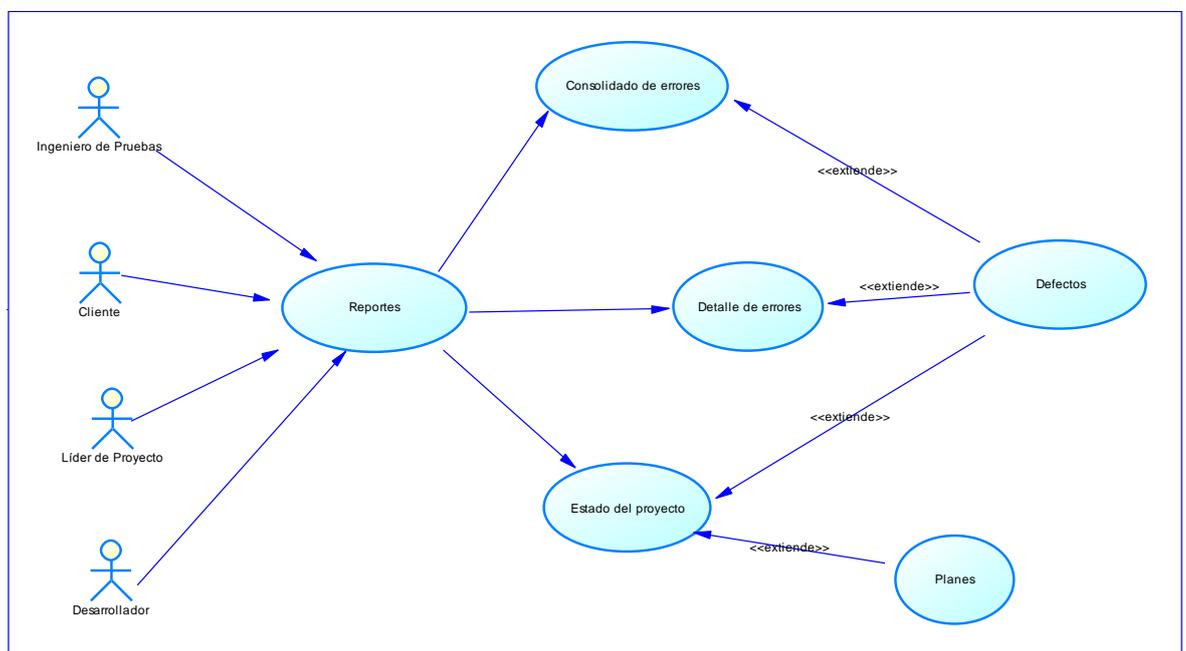


Gráfico N. 21 Casos de Usos – Reportes
Autor: David Balseca Fuente: Investigador

Caso de Uso: Consolidado de errores

Identificador:	CuAd16
Nombre del caso de uso:	Consolidado de errores
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de Prueba Cliente Líder de Proyecto Desarrollador
Descripción:	Permite visualizar los defectos de manera gráfica y consolidada
Precondiciones:	El usuario debe estar logeado y con autorización al menú reportes
Post condiciones:	No aplica
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú consolidado de errores
03	Se selecciona el proyecto que desea visualizar
04	Se selecciona el ciclo que desea seleccionar
05	Mostrará de manera gráfica los errores consolidados
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente

Tabla N. 17 Caso de uso – Consolidado de errores

Autor: David Balseca Fuente: Investigador

Caso de Uso: Detalle de errores

Identificador:	CuAd17
Nombre del caso de uso:	Detalle de errores
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de Prueba Cliente

	Líder de Proyecto Desarrollador
Descripción:	Permite visualizar los defectos de manera detallada
Precondiciones:	El usuario debe estar logeado y con autorización al menú reportes
Post condiciones:	No aplica
FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresa usuario y clave
02	Selecciona menú consolidado de errores
03	Se selecciona el proyecto que desea visualizar
04	Se selecciona el ciclo que desea seleccionar
05	Mostrará una lista con los errores y su información detallada
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente

Tabla N. 18 Caso de uso – Detalle de errores

Autor: David Balseca Fuente: Investigador

Caso de Uso: Estado del proyecto

Identificador:	CuAd18
Nombre del caso de uso:	Estado del proyecto
Creado por:	David Balseca Kelal
Fecha de creación:	12/08/2014
Actores:	Ingeniero de Prueba Cliente Líder de Proyecto Desarrollador
Descripción:	Permite visualizar el estado de avance del proyecto
Precondiciones:	El usuario debe estar logeado y con autorización al menú reportes
Post condiciones:	No aplica

FLUJO NORMAL:	
PASO DEL ACTOR	ACCIÓN
01	Ingresar usuario y clave
02	Selecciona menú consolidado de errores
03	Se selecciona el proyecto que desea visualizar
04	Se selecciona el ciclo que desea seleccionar
05	Mostrará gráficamente el estado de certificación del ciclo y el estado de ejecución del ciclo
06	Además mostrará un desglose de los casos por sus respectivos estados
FLUJO ALTERNATIVOS:	
PASO DEL ACTOR	ACCIÓN
01.1	Clave errada
01.2	Usuario no existente

Tabla N. 19 Caso de uso – Estado del Proyecto
 Autor: David Balseca Fuente: Investigador

3.2.9 Diagrama de secuencias

El diagrama de secuencia es un tipo de diagrama usado para modelar interacción entre objetos en un sistema según UML, muestra cómo interactúan los diferentes componentes del aplicativo web, al ejecutar las peticiones del usuario.

El inicio generalmente se realiza en la interface de Objetos, que es el componente que interactúa con el usuario.

3.2.9.1 Diagrama de secuencias – ingreso al sistema

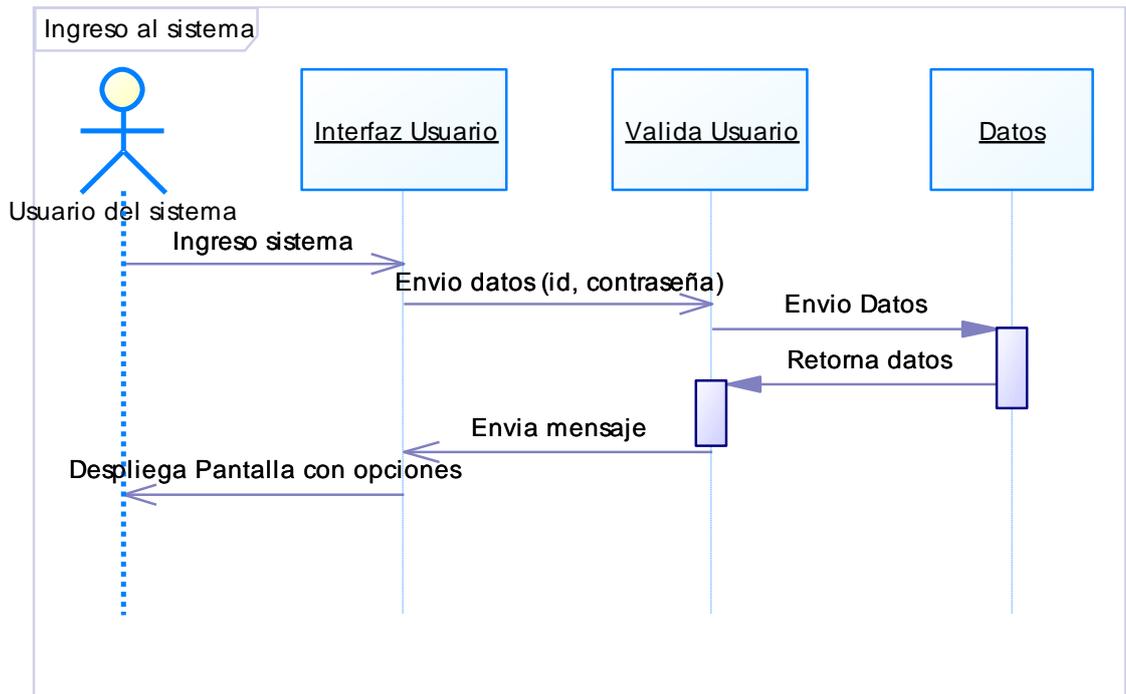


Gráfico N. 22 Diagrama de secuencias – ingreso al sistema

Autor: David Balseca

Fuente: Investigador

3.2.9.2 Diagrama de secuencias – Ejecución de proyectos

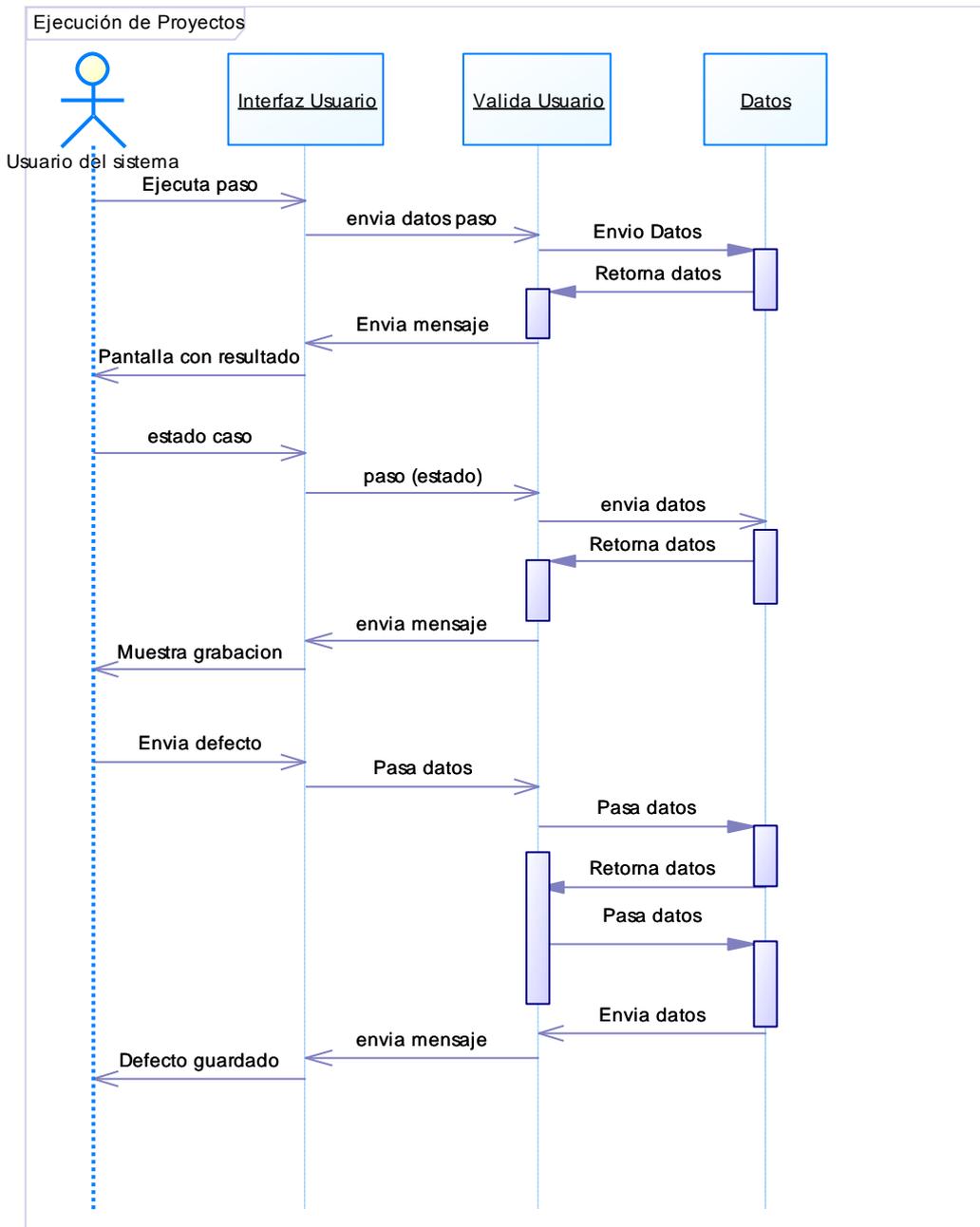


Gráfico N. 23 Diagrama de secuencias –Ejecución de proyectos

Autor: David Balseca Fuente: Investigador

3.2.10 Diagrama de despliegue

El usuario ingresa al aplicativo a través de un explorador web ingresando la dirección URL definida para el sistema. Mediante una pantalla el sistema se solicitará login y validará los datos ingresados, los cuales son gestionados y administrados en el servidor web.

Ingresado en el sistema, el usuario podrá realizar las diferentes actividades en función de sus permisos asignados al perfil.

El usuario del sistema interactúa con el servidor para realizar sus funciones y el servidor interactúa con la base de datos por medio de consultas de datos el cual permitirá obtener el resultado esperado.

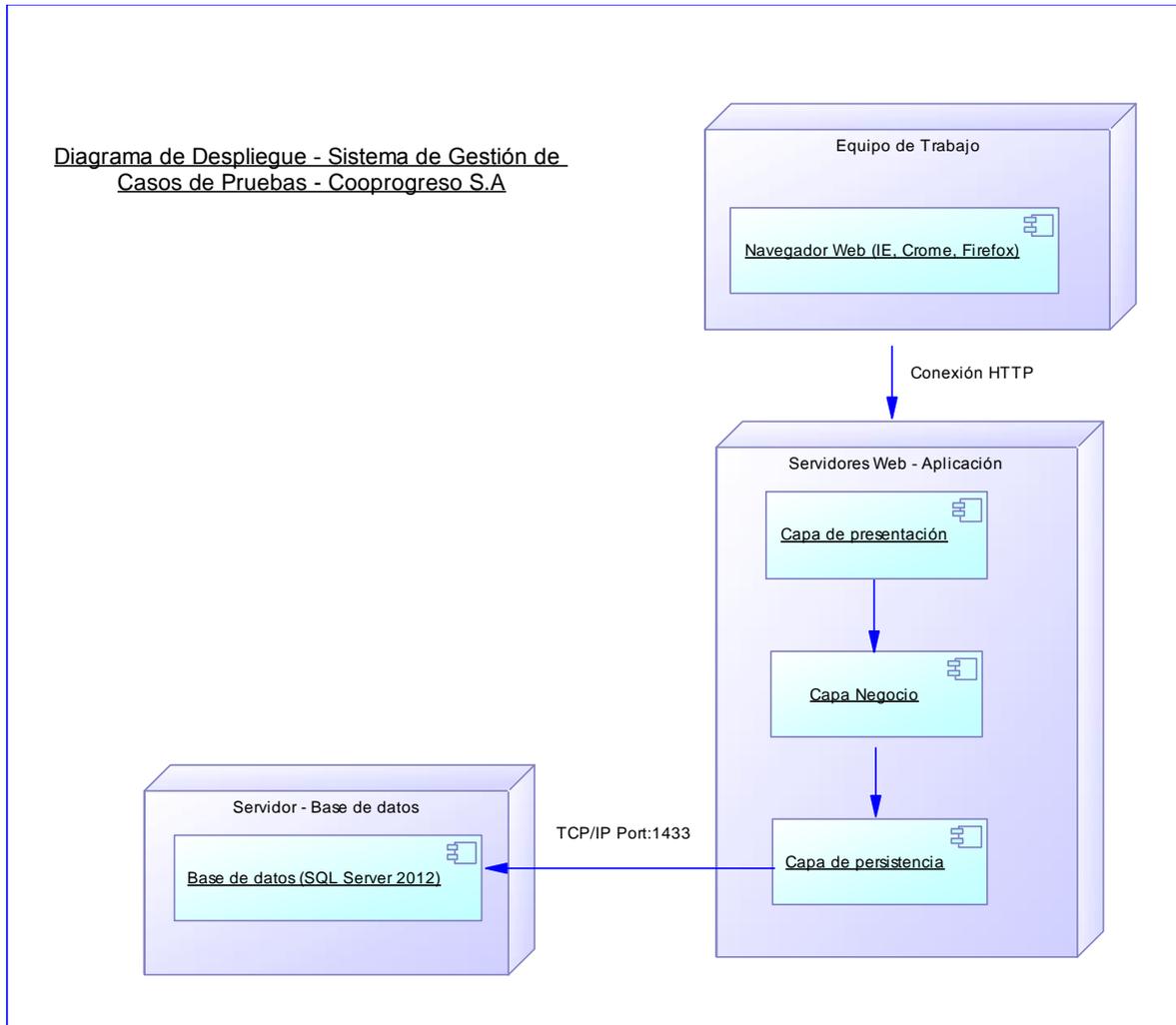


Gráfico N. 24 Diagrama de despliegue
Autor: David Balseca Fuente: Investigador

3.2.11 Diagrama de Clases

Son diagramas estáticos que nos permiten describir las estructuras de nuestro proyecto y aplicación, permitiendo visualizar por cada clase sus atributos, propiedades y su orientación a objetos.

3.2.11.1 Diagrama de Clases – Negocios



Gráfico N. 25 Diagrama de Clases - Negocio
 Autor: David Balseca Fuente: Investigador

3.2.11.2 Diagrama de Clases – Conexiones

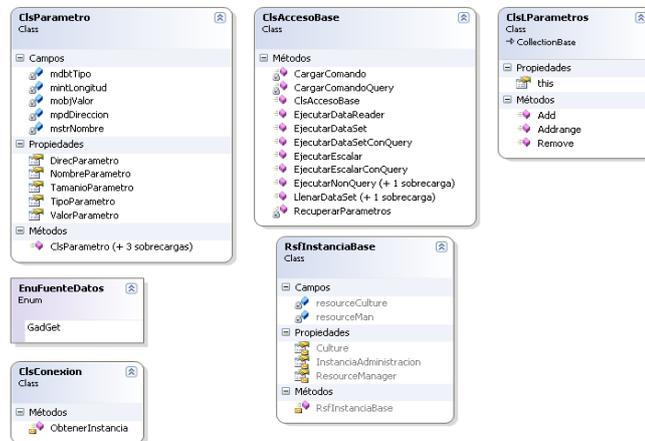


Gráfico N. 26 Diagrama de Clases - Conexiones
 Autor: David Balseca Fuente: Investigador

3.2.12 Modelo de la base de datos

Los modelos de datos describen la base conceptual para el diseño de aplicaciones que hacen uso de datos.

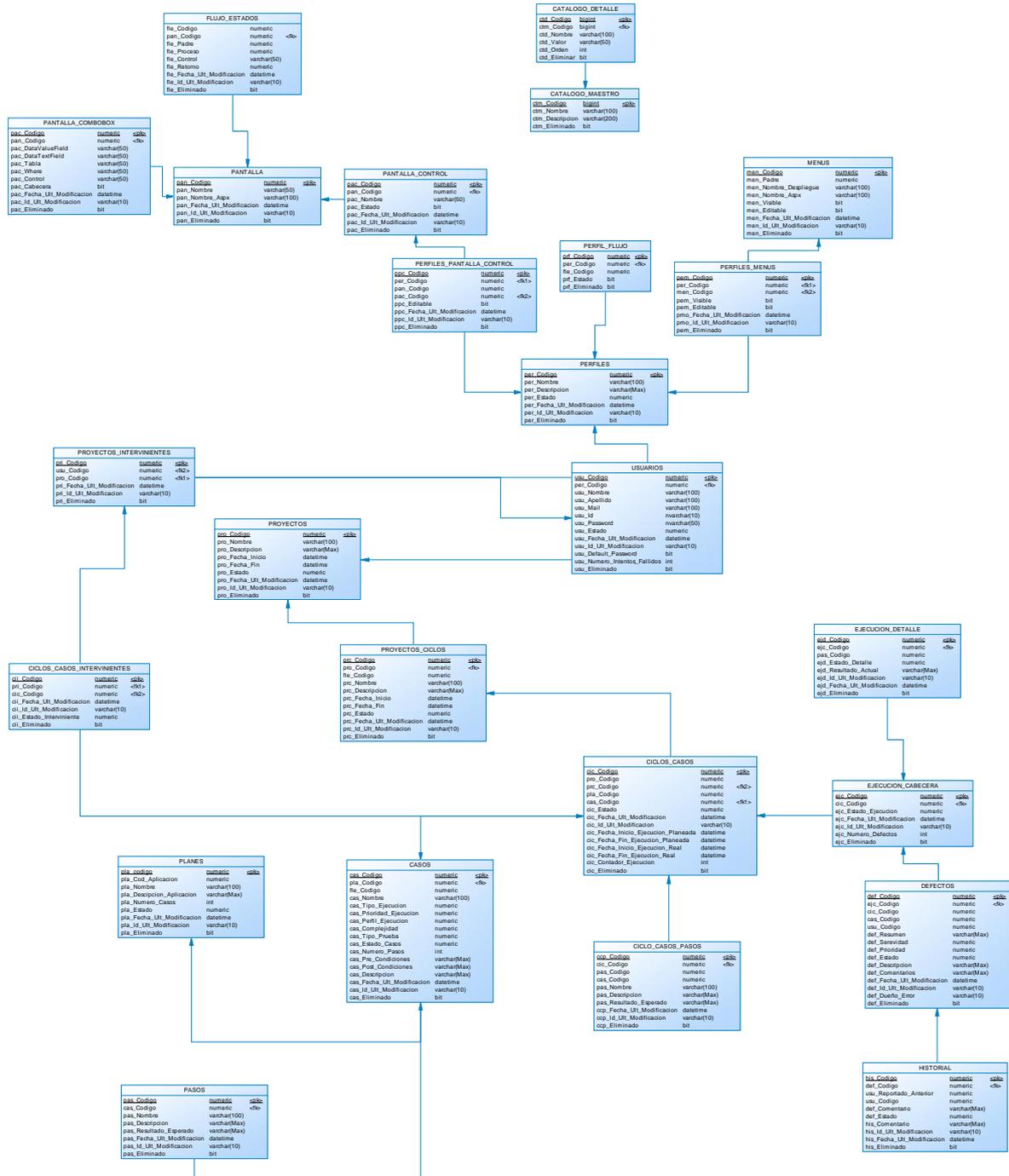


Gráfico N. 28 Modelo de la base de datos
 Autor: David Balseca Fuente: Investigador

3.2.14 Interfaces de usuarios

La pantalla inicial que se muestra (marco) es el formulario para ejecutar el proceso de autenticación (ingreso) del usuario. (Nivel de seguridad).

3.2.14.1 Figura. Interfaz: Pantalla de inicio de sesión



Gráfico N. 30 Interfaz de usuario - login
Autor: David Balseca Fuente: Investigador

Una vez ingresado al sistema con el usuario validado y aceptado se mostrara el menú en función de los permisos asignado al perfil del usuario.

3.2.14.2 Figura. Interfaz: Pantalla de menú

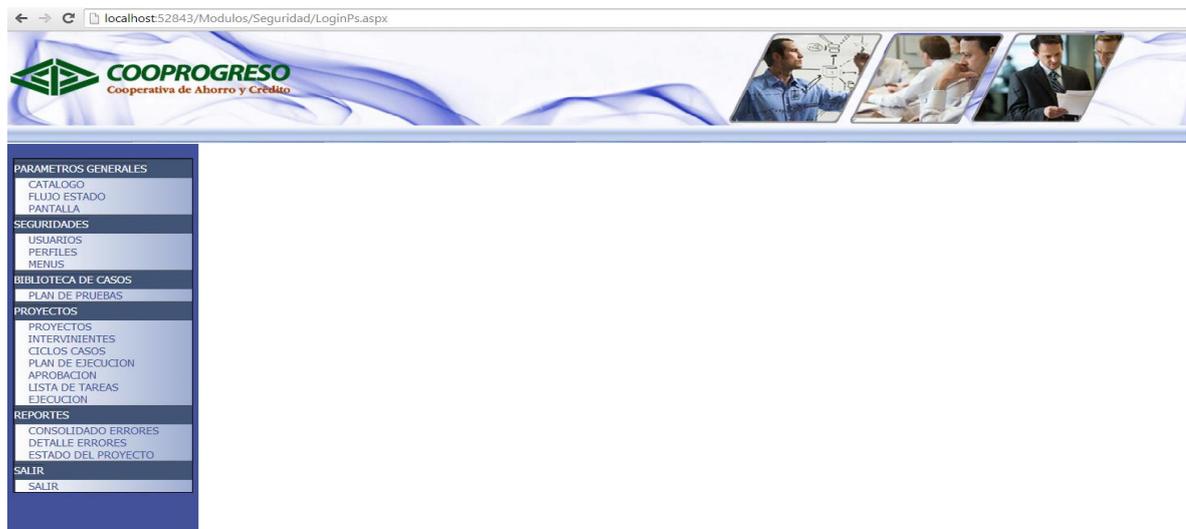


Gráfico N. 31 Interfaz de usuario -Menú
Autor: David Balseca Fuente: Investigador

3.3 Estado de situación actual

La cooperativa COOPROGRESO S.A definió realizar el despliegue de la aplicación de una manera controlada y progresiva, buscando disminuir el impacto al cambio y buscar afinar los procesos que se han definido con el uso de la nueva herramienta.

Como resultado de la implementación se inició un proyecto que se encuentra en la actualidad en la fase de pruebas, y se pudo evidenciar una reducción importante en las siguientes fases del proyecto:

- ✓ En la fase de diseño se evidencia una reducción aproximada de un 15% de esfuerzo en la construcción de casos y flujo de aprobación de los casos.

El cambio que apalanca la reducción se basa que ahora los casos se escriben en el sistema y el proceso de aprobación se hace en línea.

- ✓ En la fase de ejecución se evidencia una reducción aproximada del 25% de esfuerzo en los procesos de ejecución.

Esta reducción se evidencia al realizar las pruebas en una herramienta en línea y no en un archivo Excel como antes lo realizaban.

- ✓ En la fase de ejecución también se evidencia un reducción del 10% de esfuerzo en los procesos de re-prueba. (re-prueba por identificación de errores en las pruebas).

Los programadores reciben los defectos encontrados por los usuarios expertos en línea pudiendo reaccionar con las correcciones respectivas de manera inmediata y ya no al siguiente día como era en el proceso manual.

Metodología MSF (Microsoft Solution Framework)	Actividades	Esfuerzo por fases	Proyecto: Corrección de errores varios en el proceso de crédito	Emisión de producto cuenta de ahorro con seguro	Proyecto Pilito: Cambios en los contratos de apertura de cuenta	Esfuerzo real consumido	% optimizado
Vision	Documento de visión del cliente	10%	1680	620	580	600	3%
Planificación	Especificaciones funcionales	15%	2520	930	870	750	-16%
	Diseño de aplicación						
	Diseño de pruebas						
Desarrollo	Desarrollo de la aplicación	40%	6720	2480	2320	2350	1%
	Diseño técnico de desarrollo						
	Pruebas Unitarias						
Estabilización	Ejecución de pruebas funcionales	30%	5040	1860	1740	1300	-34%
	Corrección de errores						
Implementación	Liberación a producción	5%	840	310	290	100	-190%
	Soporte a estabilización						

Gráfico N. 32 Tabla comparativa de beneficios en esfuerzos

Autor: David Balseca Fuente: Investigador

Otro cambio que se evidenció es el incremento de la productividad de los usuarios expertos porque hacen la revisión en su puesto de trabajo y no abandonan el puesto para revisar todo el set de casos escritos.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- Se implementaron procesos de creación de proyectos, casos y pasos de pruebas para la certificación funcional alineada a la estructura de la cooperativa COOPROGRESO S.A en la que intervienen: líderes de proyectos, ingenieros de pruebas, desarrolladores y usuarios expertos, generando optimización y reducción de esfuerzos en los procesos implementados.
- En las pruebas realizadas en el piloto del proyecto se evidencia una mejora aproximadamente del 45% en reducción de esfuerzo y mejora de eficiencia, lo cual excede lo planificado inicialmente.
- La flexibilidad de la aplicación permitió una adaptación casi transparente al uso de la misma y los usuarios se sintieron motivados ya que la cooperativa está implementando nuevas herramientas que facilitan su gestión.
- La implementación del sistema web se desarrolló con el uso de herramientas existentes en la cooperativa, permitiendo reutilizar su plataforma tecnológica actual reduciendo significativamente el costo del proyecto.
- El uso de la aplicación incrementa la disponibilidad de la información relacionada a proyectos ejecutados y sus casos de pruebas y resultados obtenidos, ya que la información se mantendrá como histórica en la base de datos, en la cual el acceso a la misma es controlado por los perfiles creados en la aplicación.

Recomendaciones

- Resultado de los procesos levantados se recomienda la creación de un área de Calidad con enfoque de pruebas funcionales (Grupo testing) en la estructura orgánica de la cooperativa, y dicha estructura puede ser parte del área de calidad actual.
- El proceso de gestión automatizada de casos de pruebas funcionales el cual está implementado, es una de las fases del ciclo de vida de un proyecto, por lo cual, se recomienda realizar capacitaciones e implementación de otras mejores prácticas o framework de trabajo como lo son el PMI (Mejores prácticas para el manejo de proyectos).
- La cooperativa debe difundir y desplegar a nivel nacional el uso de la aplicación para obtener todas las bondades de una aplicación web.
- Se recomienda realizar proceso de depuración de la base de datos con periodicidad anual y dicha información sea almacenada en cinta con las políticas actuales de la cooperativa.
- Se debe definir en la cooperativa una persona con su backup respectivo que sea el administrador del sistema el cual le permita mantener la continuidad de la aplicación.

Bibliografía

- Abances, I. C. (2010). *CREACION DE EMPRESAS PARA EMPRENDEDORES: GUIA PARA LA ELABORACION DE UN PLAN DE NEGOCIO* . Piramide.
- Acuña, J. A. (2012). *Control de calida. Un enfoque integral y estadístico*. Costa Rica: Instituto Tecnológico de Costa Rica.
- Becerra. (2003). *Técnicas de Investigación científica*.
- failure, D. b. (2010). *stackoverflow*. Obtenido de Difference between bug and failure:
<http://stackoverflow.com/questions/16255039/difference-between-bug-and-failure>
- Hill, M. G. (2004). *Metodología de la Investigación 4ta Edición*. Mc Graw Hill.
- Hutcheson, M. L. (2010). *Software Testing Fundamentals: Methods and Metrics*. Wiley.
- Larman, C. (2011). *UML Y PATRONES*. España: Pearson.
- Mary, P. A. (2001). *Calpoly Edu*. Obtenido de Calpoly Edu:
<http://users.csc.calpoly.edu/~jdalbey/SWE/Papers/THERAC25.html>
- México, I. T. (2012). *www.prograweb.com.mx*. Obtenido de Arquitectura de las aplicaciones Web: <http://www.prograweb.com.mx/pweb/0201arquiAplicaweb.html>
- Pello, J. (2008). *General Software Testing*.
- Sommerville, I. (2011). *Ingeniería de Software-ebook*. México: Pearson.
- Tejada, L. (2007). *Blog*. Obtenido de Ciclo de vida de los sistemas:
www.babotejada.wordpress.com/2007/05/18/ciclo-de-vida-de-los-sistemas/
- Toro, F. (2013). *Administración de proyectos de informática*. Ecoe.

ANEXOS

ANEXO – CARTA DE AUSPICIO

ANEXO – ENTREVISTA

ANEXO - MANUAL DE USUARIOS