

Universidad de las Ciencias Informáticas

FACULTAD 4



Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Título
**Análisis y Diseño del Sistema para la Gestión del
Despliegue de los proyectos productivos de la
Facultad 4.**

Autor: Dagoberto Torres Peña

Tutores: Ing. Alieski Sarmiento Almenares
Ing. Hermes Miguel Velázquez Domínguez

Ciudad de La Habana, Junio del 2009

“Año del 50 Aniversario del Triunfo de la Revolución”

Declaración de Autoría.

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____
del año _____.

Firma del autor

Dagoberto Torres Peña

Firma del tutor

Ing. Alieski Sarmiento Almenares

Firma del tutor

Ing. Hermes Miguel Velázquez Domínguez



“...este país vivirá de la inteligencia y de las producciones intelectuales...”

Fidel Castro Ruz

DATOS DEL CONTACTO

Ing. Alieski Sarmiento Almenares

Profesor Instructor

Graduado en Junio del 2006 con Título de Oro en la Universidad de Holguín. Jefe de Polo Productivo Gestión de Información Biomédica de la Facultad 6.

Ha impartido las asignaturas de Ingeniería de Software 1 y 2 durante tres cursos. Líder del proyecto Prevención del Delito de la República Bolivariana de Venezuela.

E-mail: asarmiento@uci.cu

Ing. Hermes Miguel Velázquez Domínguez

Profesor Adiestrado

Graduado en el 2008 de Ingeniero en Ciencias Informáticas en la Universidad de Ciencias Informáticas (UCI), Título de Oro

Ha impartido asignaturas del departamento Técnicas de Programación, dentro del mismo se desempeñó como Jefe de Asignatura de Prueba de Nivel de Programación durante el curso 2008-2009. Forma parte del equipo de desarrollo del Sistema de Gestión Penitenciaria (SIGEP). Se desempeña como Jefe del Polo Productivo Gestión de Comunidades.

E-mail: hmvelazquez@uci.cu

AGRADECIMIENTOS

A mis padres por su amor infinito, por hacer de mí un hombre de bien, por la educación que en mí inculcaron... por todo.

A mi familia por estar día a día brindándome todo su apoyo.

Agradezco a la Revolución Cubana que me ha dado la oportunidad de convertirme en un buen estudiante y el hombre de ciencia que soy.

A Fidel, máximo guía de esta ardua batalla que lleva nuestro pueblo, forjador de esta universidad en la que he realizado mis estudios.

A la Universidad de las Ciencias Informáticas.

A mis tutores Aliessi y Hermes por tanta dedicación y por tanto apoyo.

A todos los que de una manera u otra contribuyeron al desarrollo de esta investigación lleguen mis agradecimientos más sinceros.

A todos mis sinceros y fieles amigos lo que han estado ahí cuando siempre los necesite.

A todos los que en el transcurso de estos cortos pero fructíferos años han dejado su huella en mi vida.

A todas...

DEDICATORIA

Dedico este trabajo la memoria de mi abuelo Lorenzo Peña Tamayo y mi abuela Gladys Vargas, que dios lo tenga en la gloria.

A mis padres Dagoberto Torres Vargas y Carmen Rosa Peña Rubio cada letra de este Trabajo de Diploma.

RESUMEN

La Universidad de las Ciencias Informáticas es una entidad nueva, llamada a jugar un papel rector en el proceso de informatización y digitalización del país. En la misma se lleva a cabo la tarea de informatizar todos sus procesos, en la que se incluyen los procesos asociados con la gestión del despliegue de los proyectos productivos de la Facultad 4. Dando respuesta a estas necesidades se desarrolló este trabajo con el objetivo de: Realizar el Análisis y el Diseño de una aplicación Web que permita la gestión del despliegue de los proyectos productivos de la Facultad 4.

El trabajo está dividido en tres capítulos, que recogen la Fundamentación Teórica: análisis de aspectos necesarios para comprender los procesos en desarrollo, así como otras aplicaciones similares existentes a nivel internacional, nacional y de la universidad, así como las diferentes metodologías, herramientas y lenguajes de modelado a utilizar en el desarrollo del sistema; Descripción de la Solución, donde se estudia todo el modelado del negocio, modelado del sistema, requerimientos funcionales y no funcionales así como vistas y descripciones de casos de uso; se realiza además un Diseño de la Solución donde queda constituido el mapa de navegación , prototipo de interfaz, arquitectura del sistema, clases del sistema así como la vista del despliegue.

PALABRAS CLAVES

Gestión, Despliegue, Análisis, Diseño.

Summary.

SUMMARY

The Computer Science University is a new entity, known to play a leading role in the process of computerization and digitization of the country. The same is done with the task of computerizing all of its processes, which include the processes associated with managing the deployment of the productive projects of the School 4. In response to these needs this paper was developed with the specific aim of: Model business processes, capture, analyze and manage the requirements of the system to permit the analysis and design system for managing the deployment of projects at the School 4.

The work is divided into three chapters, which represent the theoretical analysis of aspects needed to understand the processes in development, as well as other similar applications exist at international, national and university, as well as various methodologies, tools and languages modeling for use in system development, a description of the solution, which examines the whole business model, system modeling, functional and non functional requirements as well as views and descriptions of use cases, design is also one of the where the solution is formed sitemap prototype interface, system architecture, system classes as well as the view of the deployment.

KEYWORDS

Management, Deployment, Analysis, Design

INDICE DE TABLAS

Tabla 1: Hitos de las fases del RUP.	12
Tabla 2: Comparación de metodologías.	14
Tabla 3: Roles y artefactos que desarrollan los Analistas y Diseñadores del RUP	15
Tabla 4: Patrones de Casos de Uso.	17
Tabla 6: Patrones de diseño GRASP.	18
Tabla 7: Patrones de diseño GoF.	19
Tabla 8: Comparación de herramientas.	23
Tabla 9: Comparación de lenguajes de programación.	27
Tabla 10: Comparación de Sistemas Gestores de bases de datos.	29
Tabla 11: Actores del Negocio.	34
Tabla 12: Trabajadores del Negocio.	35
Tabla 13: Descripción del CUN Elaborar Plan de Despliegue.	36
Tabla 14: Descripción del CUN Elaborar Elementos del Despliegue.	37
Tabla 15: Descripción del CUN Administrar proceso de pruebas de aceptación.	38
Tabla 16: Actores del Sistema.	49
Tabla 17: Descripción del CUS Gestionar Elementos del Despliegue.	50
Tabla 18: Descripción de los métodos de la clase del diseño ProyectoDAO: despliegue de productos de software de la Facultad 4.	73
Tabla 19: Descripción de los métodos de la clase del diseño ProyectoDesplegarFacade: despliegue de productos de software de la Facultad 4.	73
Tabla 20: Descripción de los métodos de la clase del diseño ProyectoDesplegarManager: despliegue de productos de software de la Facultad 4.	74
Tabla 21: Descripción de los métodos de la clase del diseño ElementoDAO: despliegue de productos de software de la Facultad 4.	74
Tabla 22: Descripción de los métodos de la clase del diseño ElementoDespliegueFacade: despliegue de productos de software de la Facultad 4.	75
Tabla 23: Descripción de los métodos de la clase del diseño ElementoDespliegueManager: despliegue de productos de software de la Facultad 4.	75
Tabla 24: Descripción de la tabla Usuario: despliegue de productos de software de la Facultad 4.	79
Tabla 25: Descripción de la tabla Elemento: despliegue de productos de software de la Facultad 4.	79
Tabla 26: Descripción de la tabla Persona: despliegue de productos de software de la Facultad 4.	80
Tabla 27: Descripción de la tabla No Conformidad: despliegue de productos de software de la Facultad 4.	80
Tabla 28: Descripción de la tabla Centro: despliegue de productos de software de la Facultad 4.	80
Tabla 29: Descripción de la tabla Prueba: despliegue de productos de software de la Facultad 4.	80
Tabla 30: Descripción de la tabla Usuario: despliegue de productos de software de la Facultad 4.	81
Tabla 31: Descripción de la tabla Viajes: despliegue de productos de software de la Facultad 4.	81
Tabla 32: Descripción del CUN Administrar proceso de pruebas piloto.	89
Tabla 33: Descripción del CUN Realizar Liberación del Producto.	90
Tabla 34: Descripción del CUN Brindar Soporte al Sistema.	91
Tabla 35: Descripción del CUN Realizar Transferencia Tecnológica.	92
Tabla 36: Descripción del CUS Administrar Proceso de Pruebas de Piloto.	96
Tabla 37: Descripción del CUS Administrar Control de recursos humanos y logística para el despliegue.	100

Índice.

Tabla 38: Descripción del CUS Administrar Control del Sistema.	104
Tabla 39: Descripción del CUS Gestionar Proyecto a Desplegar.	108

INDICE DE FIGURAS

Ilustración 1: Flujos de trabajo y Fases de la metodología RUP. (IVAR JACOBSON, 2000).....	12
Ilustración 2: Flujo de Actividades de la etapa de despliegue. (Rational Unified Process Help, 2006)	32
Ilustración 3: Diagrama de Casos de Uso del Negocio: despliegue de productos de software de la Facultad 4.....	36
Ilustración 4: Diagrama de Actividades del Caso de Uso Elaborar Plan de Despliegue.	40
Ilustración 5: Diagrama de Actividades del Caso de Uso Elaborar Elementos del Despliegue.....	41
Ilustración 6: Diagrama de Actividades del Caso de Uso Administrar proceso de pruebas de aceptación.	42
Ilustración 7: Modelo de Objetos del Negocio: despliegue de productos de software de la Facultad 4.	43
Ilustración 8: Diagrama de Casos de Uso del Sistema: despliegue de productos de software de la Facultad 4.....	50
Ilustración 9: Diagrama de la Vista de los Casos de Uso del Sistema: despliegue de productos de software de la Facultad 4.....	57
Ilustración 10: Mapa de Navegación: despliegue de productos de software de la Facultad 4.	59
Ilustración 11: Prototipo No Funcional Interfaz Usuario Pantalla Autenticación al Sistema: despliegue de productos de software de la Facultad 4.	60
Ilustración 12: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Aceptación Escenario Insertar No Conformidad: despliegue de productos de software de la Facultad 4.....	61
Ilustración 13: Representación básica de la arquitectura del sistema para despliegue de productos de software de la Facultad 4.	62
Ilustración 14: Modelo-Vista-Controlador Web de Spring: despliegue de productos de software de la Facultad 4.....	63
Ilustración 15: Arquitectura básica del Web MVC de Spring: despliegue de productos de software de la Facultad 4.....	63
Ilustración 16: Diagrama de Clases del Diseño del Caso de Uso Gestionar Elementos del Despliegue: despliegue de productos de software de la Facultad 4.....	68
Ilustración 17: Diagrama de Clases del Diseño del Caso de Uso Administrar Proceso Pruebas de Aceptación: despliegue de productos de software de la Facultad 4.	69
Ilustración 18: Diagrama de Secuencia del Caso de Uso Gestionar Elementos del Despliegue escenario Registrar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.....	70
Ilustración 19: Diagrama de Secuencia del Caso de Uso Gestionar Elementos del Despliegue escenario Buscar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.....	71
Ilustración 20: Diagrama de Secuencia del Caso de Uso Administrar Proceso Pruebas de Aceptación escenario Insertar No Conformidad: despliegue de productos de software de la Facultad 4.....	71
Ilustración 21: Diagrama de Secuencia del Caso de Uso Administrar Proceso Pruebas de Aceptación escenario Buscar No Conformidad: despliegue de productos de software de la Facultad 4.....	72
Ilustración 22: Diagrama de Despliegue: despliegue de productos de software de la Facultad 4.	76
Ilustración 23: Diagrama de Clases Persistentes: despliegue de productos de software de la Facultad 4.	77

Índice.

Ilustración 24: Modelo de Datos: despliegue de productos de software de la Facultad 4.	78
Ilustración 25: Diagrama de Actividades del Caso de Uso Administrar proceso de pruebas de piloto.	93
Ilustración 26: Diagrama de Actividades del Caso de Uso Realizar Liberación del Producto.	94
Ilustración 27: Diagrama de Actividades del Caso de Uso Brindar Soporte al Sistema.	95
Ilustración 28: Diagrama de Actividades del Caso Uso Realizar Transferencia Tecnológica.	96
Ilustración 29: Prototipo No Funcional Interfaz Usuario Pantalla Gestionar Elemento del Despliegue Escenario Registrar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.	112
Ilustración 30: Prototipo No Funcional Interfaz Usuario Pantalla Gestionar Elemento del Despliegue Escenario Buscar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.	113
Ilustración 31: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Aceptación Escenario Buscar No Conformidad: despliegue de productos de software de la Facultad 4.	114
Ilustración 32: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Aceptación Escenario Generar Reporte de No Conformidad: despliegue de productos de software de la Facultad 4.	115
Ilustración 33: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Piloto Escenario Registrar Centros de Pilotaje: despliegue de productos de software de la Facultad 4.	116
Ilustración 34: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Piloto Escenario Insertar No Conformidad: despliegue de productos de software de la Facultad 4.	117
Ilustración 35: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Piloto Escenario Buscar No Conformidad: despliegue de productos de software de la Facultad 4.	118
Ilustración 36: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Piloto Escenario Generar reporte de No Conformidades: despliegue de productos de software de la Facultad 4.	119
Ilustración 37: Prototipo No Funcional Interfaz Usuario Pantalla Administración de RR.HH y Logística Escenario Registrar Personal del despliegue: despliegue de productos de software de la Facultad 4.	120
Ilustración 38: Prototipo No Funcional Interfaz Usuario Pantalla Administración de RR.HH y Logística Escenario Buscar Personal del despliegue: despliegue de productos de software de la Facultad 4.	121
Ilustración 39: Prototipo No Funcional Interfaz Usuario Pantalla Administración de RR.HH y Logística Escenario Registrar Viajes a Centros de Pilotaje: despliegue de productos de software de la Facultad 4.	122
Ilustración 40: Prototipo No Funcional Interfaz Usuario Pantalla Administración de RR.HH y Logística Escenario Generar Reporte del Personal del despliegue: despliegue de productos de software de la Facultad 4.	123
Ilustración 41: Prototipo No Funcional Interfaz Usuario Pantalla Administración Control del Sistema Escenario Crear Usuario: despliegue de productos de software de la Facultad 4.	124
Ilustración 42: Prototipo No Funcional Interfaz Usuario Pantalla Administración Control del Sistema Escenario Buscar Usuario: despliegue de productos de software de la Facultad 4.	125

Índice.

Ilustración 43: Prototipo No Funcional Interfaz Usuario Pantalla Gestionar Proyecto a Desplegar Escenario Registrar Proyecto a Desplegar: despliegue de productos de software de la Facultad 4.....	126
Ilustración 44: Prototipo No Funcional Interfaz Usuario Pantalla Gestionar Proyecto a Desplegar Escenario Buscar Proyecto a Desplegar o en Fase de Despliegue: despliegue de productos de software de la Facultad 4.	127
Ilustración 45: Diagrama de Clases del Diseño del Caso de Uso Administrar Proceso de Pruebas Piloto: despliegue de productos de software de la Facultad 4.	128
Ilustración 46: Diagrama de Clases del Diseño del Caso de Uso Administrar Control de Recursos Humanos y Logística: despliegue de productos de software de la Facultad 4.	129
Ilustración 47: Diagrama de Clases del Diseño del Caso de Uso Gestionar Proyecto a Desplegar: despliegue de productos de software de la Facultad 4.	130
Ilustración 48: Diagrama de Clases del Diseño del Caso de Uso Gestionar Control del Sistema: despliegue de productos de software de la Facultad 4.	131
Ilustración 49: Diagrama de Secuencia del Caso de Uso Gestionar Elementos del Despliegue escenario Modificar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.....	132
Ilustración 50: Diagrama de Secuencia del Caso de Uso Administrar prueba de Aceptación escenario Visualizar Estado del Despliegue: despliegue de productos de software de la Facultad 4.....	133
Ilustración 51: Diagrama de Secuencia del Caso de Uso Administrar prueba de Aceptación escenario Generar Reporte de No Conformidades: despliegue de productos de software de la Facultad 4.....	134
Ilustración 52: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Registrar Centros: despliegue de productos de software de la Facultad 4.	135
Ilustración 53: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Registrar No Conformidad: despliegue de productos de software de la Facultad 4.....	136
Ilustración 54: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Buscar No Conformidad: despliegue de productos de software de la Facultad 4.....	137
Ilustración 55: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Generar Reporte No Conformidad: despliegue de productos de software de la Facultad 4.....	138
Ilustración 56: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Visualizar Estado de las Pruebas: despliegue de productos de software de la Facultad 4.....	139
Ilustración 57: Diagrama de Secuencia del Caso de Uso Administrar RR.HH y Logística escenario Registrar Viajes: despliegue de productos de software de la Facultad 4.	140
Ilustración 58: Diagrama de Secuencia del Caso de Uso Administrar RR.HH y Logística escenario Registrar Personal: despliegue de productos de software de la Facultad 4.	141
Ilustración 59: Diagrama de Secuencia del Caso de Uso Administrar RR.HH y Logística escenario Buscar Personal: despliegue de productos de software de la Facultad 4.	142
Ilustración 60: Diagrama de Secuencia del Caso de Uso Administrar RR.HH y Logística escenario Modificar Personal: despliegue de productos de software de la Facultad 4.	143
Ilustración 61: Diagrama de Secuencia del Caso de Uso Gestionar Proyecto a Desplegar escenario Registrar Proyecto: despliegue de productos de software de la Facultad 4.	144

Índice.

Ilustración 62: Diagrama de Secuencia del Caso de Uso Gestionar Proyecto a Desplegar escenario Buscar Proyecto: despliegue de productos de software de la Facultad 4.	145
Ilustración 63: Diagrama de Secuencia del Caso de Uso Gestionar Proyecto a Desplegar escenario Modificar Proyecto: despliegue de productos de software de la Facultad 4.	146
Ilustración 64: Diagrama de Secuencia del Caso de Uso Gestionar Control del Sistema escenario Modificar Usuario: despliegue de productos de software de la Facultad 4.	147
Ilustración 65: Diagrama de Secuencia del Caso de Uso Gestionar Control del Sistema escenario Buscar Usuario: despliegue de productos de software de la Facultad 4. ..	148
Ilustración 66: Diagrama de Secuencia del Caso de Uso Gestionar Control del Sistema escenario Crear Usuario: despliegue de productos de software de la Facultad 4.	149
Ilustración 67: Diagrama de Secuencia del Caso de Uso Gestionar Control del Sistema escenario Eliminar Usuario: despliegue de productos de software de la Facultad 4.	150

INDICE DE CONTENIDO

INTRODUCCION	1
Capítulo 1: Fundamentación Teórica	5
1.1. Introducción.....	5
1.2. Gestión de despliegue.....	5
1.3. Sistemas automatizados existentes.	7
1.3.1. Sistemas existentes a nivel internacional.	7
1.3.2. Análisis crítico sobre la situación de la gestión del despliegue de proyectos informáticos en nuestro país y en la Universidad de las Ciencias Informáticas.	9
1.4. Fundamentación de la metodología a utilizar.....	9
1.4.1. Programación Extrema (XP, <i>eXtreme Programming</i>).	10
1.4.2. Compendio de Soluciones de Microsoft (MSF, <i>Microsoft Solutions Framework</i>).....	10
1.4.3. Proceso Unificado de Desarrollo (RUP, <i>Rational Unified Process</i>).	11
1.5. Patrones.	15
1.5.1. Patrones de casos de uso.....	16
1.5.2. Patrones de arquitectura.....	17
1.5.3. Patrones diseño.	18
1.6. Fundamentación de las herramientas a utilizar para el desarrollo del sistema.	19
1.6.1. Lenguaje Unificado de Modelado 2.0 (UML, <i>Unified Modeling Language</i>).	20
1.6.2. Visual Paradigm for UML 6.1 Enterprise Edition.	21
1.6.3. ArgoUML.....	22
1.6.4. Rational Rose.....	22
1.7. Lenguaje de programación.....	24
1.7.1. Spring Framework.....	27
1.8. Sistema Gestor de Base de Datos.	28
1.9. Conclusiones.....	30
Capítulo 2: Descripción de la Solución.	31
2.1. Introducción.	31
2.2. Modelado del Negocio.	31
2.2.1 Actores del Negocio.....	33
2.2.2 Caso de Uso del Negocio.	34
2.2.3 Trabajadores del Negocio.....	34
2.2.4 Entidades del Negocio.	35
2.3 Diagrama de CUN.	36
2.3.1 Descripciones textuales de los Casos de Uso del Negocio.....	36
2.3.2 Diagramas de actividades los Casos de Uso del Negocio.	40
2.3.3 Modelo de Objetos.....	42
2.4 Reglas del Negocio.....	43
2.5 Modelado del Sistema.	44
2.5.1 Requisitos Funcionales (RF).....	45
2.5.2 Requisitos No Funcionales (RNF).	46
2.5.3 Actor del Sistema.....	48
2.5.4 Casos de Uso del Sistema.....	49
2.5.5 Diagrama de CUS.....	49
2.5.6 Descripciones textuales ampliadas de todos los CUS.....	50

Índice.

2.6. Vista de CU.....	57
2.7. Conclusiones.	57
Capítulo 3: Diseño de la Solución.....	59
3.1 Introducción	59
3.2 Mapa de Navegación.....	59
3.3 Prototipo No Funcional de Interfaz de Usuario.	60
3.4 Arquitectura del sistema.	61
3.4.1 Ambiente de desarrollo	66
3.5 Realización de CU-Diseño.....	67
3.5.1 Diagramas de Interacción.....	70
3.6 Clases del Diseño con las descripciones de sus métodos.	72
3.8 Diseño de la Base de Datos.	76
3.8.1 Diseño de Clases Persistentes	76
3.8.2 Modelo de Datos.....	78
3.8.3 Descripción de las tablas de la Base de Datos.....	79
3.9. Conclusiones.	81
CONCLUSIONES.....	82
RECOMENDACIONES	84
REFERENCIAS BIBLIOGRAFICAS	85
BIBLIOGRAFÍA.....	87
GLOSARIO DE TÉRMINOS.....	88
ANEXOS.....	89
Anexo 1. Descripciones Textuales de los Casos de Uso del Negocio	89
Anexo 2. Diagramas de Actividades	93
Anexo 3.Descripciones Textuales de los Casos de Uso del Sistema.....	96
Anexo 4.Prototipo No Funcional de Interfaz de Usuario.....	112
Anexo 5.Diagramas del Diseño.	128
Anexo 6.Diagramas de Interacción (Diagramas de Secuencia).	132
Anexo 7. Encuesta realizada a estudiantes y profesores que han participado en despliegue de producto de software.	150

INTRODUCCION

La informatización se ha convertido actualmente en una necesidad del desarrollo de la humanidad; esta contribuye a hacer la vida más simple, y ayuda a aumentar la fiabilidad y efectividad de los bienes y servicios. Por su importancia y vigencia, el mundo empresarial no escapa de esta actividad y necesita involucrarse y promover el desarrollo de la informatización y sus actividades relacionadas, para facilitar el intercambio de información y contribuir de manera total a la transferencia de conocimientos y tecnologías.

Cuba está consciente de que una sociedad para ser más eficaz, eficiente y competitiva, debe aplicar la informatización en todas las esferas y procesos. En este sentido, el país ha identificado desde muy temprano la conveniencia y necesidad de dominar e introducir en la práctica social las Tecnologías de la Información y las Comunicaciones (TIC) y lograr una cultura digital como una de las características imprescindibles del hombre nuevo, lo que facilitaría a la sociedad acercarse más hacia el objetivo de un desarrollo sostenible.

Además de que cada vez es más frecuente que el desarrollo de software sea llevado a cabo por personas que no trabajan en un mismo local o que incluso se encuentran dispersas en lugares distantes y que no se relacionan con los clientes o usuarios finales, esto ha traído como consecuencia problemas con el entendimiento de las necesidades del cliente, falta de un acuerdo entre los involucrados en qué el software debe hacer, falta de un mejor entendimiento por parte de los desarrolladores de los requisitos del software o sea no son captadas y analizadas correctamente las necesidades del cliente y no son controlados los cambios realizados. Es por eso que la obtención de un software con calidad implica la utilización de metodologías o procedimientos estándares para la modelación, especificación de requisitos, análisis, diseño, programación y pruebas del mismo que permitan uniformar la filosofía de trabajo.

La Industria Cubana del Software está llamada a convertirse en una potencia generadora de importantes ingresos a la nación, como resultado del correcto aprovechamiento de las ventajas del alto capital humano disponible. La Universidad de las Ciencias Informáticas (UCI) es también una entidad productiva, en la cual se vinculan los procesos docencia-producción-investigación y está llamada a jugar un papel rector en él.

Introducción

Situación problémica:

Actualmente en la Facultad 4 de la Universidad de las Ciencias Informáticas se están desarrollando un conjunto de productos de software, que posterior a su liberación por la Dirección de Calidad, tienen que ser desplegados en los entornos de los clientes para los cuales se han concebido. Durante el proceso de despliegue se emplea mucho tiempo en gestionar y coordinar todas las actividades que necesitan realizarse durante esta etapa, además el jefe de despliegue no tiene un control eficiente de los recursos destinados al mismo ya que todo el proceso se realiza de forma manual lo que conlleva a que el despliegue se retrase en muchos casos. De la experiencia obtenida hasta el momento se concluye que existen problemas durante la ejecución del proceso de despliegue como son:

- No existe un control de los recursos materiales destinados para el despliegue.
- El responsable del despliegue no tiene un reporte actualizado del estado del despliegue.

Los elementos anteriormente mencionados influyen de manera directa en el desarrollo del proceso de despliegue de una solución de software, trayendo consigo un aumento en el tiempo estimado para esta etapa y problemas en la calidad de dicho proceso.

El presente Trabajo de Diploma pretende dar solución a la situación problémica anteriormente expuesta, para lo cual se plantea la siguiente interrogante:

Problema a resolver: ¿Cómo mejorar la gestión del proceso de despliegue de los proyectos productivos de la Facultad 4?

Objeto de estudio: Proceso de gestión de despliegues de productos de software.

Campo de Acción: Gestión del despliegue de los proyectos productivos de la Facultad 4.

Objetivo general

Realizar el Análisis y el Diseño de una aplicación Web que permita la gestión del despliegue de los proyectos productivos de la Facultad 4.

Objetivos específicos

- Analizar los procesos de la gestión del despliegue de los proyectos productivos de la Facultad 4.

Introducción

- Identificar las funcionalidades que tendrá la aplicación.
- Diseñar el sistema analizado.

Tareas de la investigación:

- Estudiar los procesos de la gestión del despliegue de los proyectos productivos de la Facultad 4.
- Estudiar las actividades propuestas en el despliegue por la metodología de desarrollo que se usa en los proyectos de la Facultad 4.
- Estudiar el proceso de negocio actual para la gestión del despliegue de los proyectos productivos de la Facultad 4.
- Identificar los requerimientos funcionales y no funcionales del sistema.
- Definir la arquitectura del sistema.
- Realizar el diseño de las clases candidatas a la implementación.
- Identificar las clases persistentes.
- Realizar el diagrama de despliegue.

Resultados esperados:

El análisis y el diseño del sistema para la gestión del despliegue de los proyectos productivos de la Facultad 4.

Dentro de los métodos de investigación teóricos se emplearán los siguientes:

- Analítico-Sintético porque permite analizar, estudiar en partes e interpretar la teoría con el fin de extraer los elementos más importantes que se relacionan con el objeto.
- Inductivo-Deductivo porque permite a través de un razonamiento llegar a un grupo de conocimientos particulares y generales.
- Histórico-Lógico permite observar la trayectoria de un determinado fenómeno en diferentes períodos de la historia, revelar las etapas principales de su desenvolvimiento y las conexiones históricas fundamentales.
- Método de la Modelación porque permite la creación de modelos, es decir representar lo que se quiere estudiar de forma más simple, explicando lo que pasas de una manera lógica.

Dentro de los métodos de investigación empírico se emplearán:

Introducción

- Método de la Observación porque permite contemplar el desarrollo de la investigación en distintos momentos de esta, conociendo el problema y el objeto de investigación, estudiando su curso natural y recoge la información de cada uno de los conceptos o variables definidas en la hipótesis.
- Método de la Entrevista y Encuesta porque permite obtener la mayor información posible acerca del tema de investigación, además de las experiencias, las ideas y los puntos de vistas de los entrevistados que aportan conocimientos específicos del tema. Ver **Anexo 7**.

El trabajo estará estructurado en los siguientes capítulos:

Capítulo 1. Fundamentación Teórica: se describen los aspectos relacionados con la gestión del despliegue de un producto de software. Se analizan sistemas informáticos a nivel internacional que se centran en la gestión del despliegue de un software. Así como se caracteriza la metodología y las herramientas a utilizar para el desarrollo de la solución. Además se plantea la propuesta de automatización.

Capítulo 2. Descripción de la solución: se desarrollan cada una de las actividades propuestas por el Proceso Unificado de Desarrollo (RUP, por sus siglas en inglés: Rational Unified Process) para la ingeniería de requisitos y se obtienen diferentes artefactos como resultado de estas actividades, modelándose los procesos del negocio y las funcionalidades del sistema.

Capítulo 3. Diseño de la solución: se muestra la arquitectura del sistema, el diseño de la solución, el mapa de navegación, los prototipos de interfaz no funcionales, la realización de los casos de uso del sistema y la vista del despliegue del mismo.

Capítulo 1: Fundamentación Teórica

1.1. Introducción

La informatización de cualquier proceso de una organización lleva un estudio teórico imprescindible para crear una aplicación que responda eficientemente, es por ello que en este capítulo se abordan temas relacionados con la fundamentación teórica de esta investigación. Además se ejemplifican soluciones existentes en el mundo y en nuestro país referente a la gestión del despliegue de software.

Un sistema de gestión no es más que la forma en que una organización dirige y controla sus actividades. El despliegue es una etapa delicada en cualquier proyecto. Esta disciplina describe las actividades asociadas al aseguramiento para que el producto de software esté disponible para sus usuarios finales y es la culminación del esfuerzo de desarrollo de software (JACOBSON, 2000). Su objetivo principal es producir con éxito las distribuciones del producto y distribuirlo a los usuarios. Un despliegue fluido depende de la experiencia en la planificación y gestión, principalmente en proyectos grandes y complejos, ya que esta disciplina se realiza a partir de estos últimos, aplicando la norma mundial de gestión de proyectos que garantiza el éxito de migraciones y la entrega del producto. Ante el colosal reto de las nuevas Tecnologías de la Información y las Comunicaciones (TIC), las organizaciones, empresas y centros de altos estudios han tenido la necesidad de implantar herramientas que informaticen los procesos de manejo de la información, pues los métodos tradicionales manuales no son viables en organizaciones relativamente grandes. La Universidad de las Ciencias Informáticas no ha dejado pasar por alto esta nueva era de apogeo tecnológico, ya que es uno de los motores impulsores de dicha era y en aras de enmarcarse en este evolutivo proceso, hoy la escuela cuenta con varios sistemas de gestión que brindan una mayor eficiencia del trabajo y una mejor terminación del mismo, la Facultad 4 no está ausente a estos sistemas, de ahí el surgimiento de esta investigación.

1.2. Gestión de despliegue.

La gestión del despliegue de un producto se describe de tres modos:

- La instalación del producto.
- El empaquetamiento del producto.

Capítulo 1: Fundamentación Teórica.

- El acceso del software por la Red.

En cada instancia, hay un énfasis en probar el producto en el sitio de desarrollo, seguido por la experimentación beta antes de que el producto se dosifique finalmente al cliente. Aunque las actividades de despliegue alcanzan el máximo en la Fase de Transición, algunas de las actividades ocurren en fases atrás al plan y se preparan para desplegar. Estas actividades son las que rigen todo el flujo de trabajo del proceso de despliegue:

- La planificación del despliegue (Plan de Despliegue) se inicia a primera hora del ciclo de vida del proyecto y se ocupa no sólo de la entrega del software, sino que también del desarrollo de los materiales de enseñanza y soporte del sistema para asegurar que el usuario final puede usar el producto entregado exitosamente. (Rational Unified Process Help, 2006)
- El material del soporte (Material de Soporte) cubre el rango completo de información que se requirió por el usuario final para instalar, dirigir, usar y mantener el sistema entregado. También incluye material de enseñanza para todas las posiciones que estarán obligadas a usar el nuevo sistema. (Rational Unified Process Help, 2006)
- El despliegue coloca un gran énfasis en asegurar que el producto está bien probado antes de su liberación o entrega final al cliente. La actividad de Prueba de Aceptación se refiere a dos clases de ambientes experimentales. En primer lugar la versión actual del software necesita estar suficientemente probada en el ambiente de prueba de desarrollo, y luego re experimentar la prueba en el sitio donde funcionará finalmente el software. El ambiente experimental debería ser una instancia del ambiente final del sistema. (Rational Unified Process Help, 2006)
- Una vez que el producto ha sido probado en el sitio de desarrollo necesita estar preparado para entregarse al cliente. La versión inicial creada para los propósitos de experimentación beta, un despliegue experimental para los usuarios finales, brinda la madurez alcanzada para el producto final. La unidad de despliegue del producto describe la logística de crear una versión del producto que consta del software y los artefactos acompañantes necesarios para instalarlo y usarlo. (Rational Unified Process Help, 2006)

Objetivos de la gestión del despliegue de los proyectos productivos de la Facultad 4:

- Poseer un control de los materiales destinados para el despliegue.
- Realizar reportes actualizados del estado del despliegue para el responsable del mismo.
- Asegurar que el producto final de software esté disponible para los usuarios finales.

Para alcanzar los objetivos anteriormente mencionados, en la Facultad 4 se está desarrollando un sistema para la gestión del despliegue de sus proyectos productivos y como parte de la ejecución del mismo, el presente Trabajo de Diploma recogerá los artefactos de los 3 primeros flujos de trabajo (Negocio, Requisitos y Análisis&Diseño) obtenidos según la metodología Proceso Unificado de Desarrollo (RUP) para dicha aplicación.

1.3. Sistemas automatizados existentes.

Cada vez es más frecuente que productores de software a nivel mundial estén enfrascados en desarrollar sistemas para la gestión de todo tipo de actividad o proceso, permitiendo enfrentar cambios, agregar valor a la información y facilitar el trabajo en equipo para una mejor calidad. En este sentido se analizarán varios sistemas a nivel internacional que abordan áreas relacionadas con la gestión del despliegue de un producto de software.

1.3.1. Sistemas existentes a nivel internacional.

Kaseya

Kaseya es un proveedor global de software de Automatización de Servicios Gestionados (ASG) con sede principal en los Estados Unidos de Norteamérica (EEUU), para proveedores de Soluciones de Tecnologías de la Investigación (TI) y organizaciones de TI Corporativas que se benefician del despliegue de las herramientas de Administración de Sistemas de Kaseya.

Kaseya Enterprise Edition 2009

Kaseya Enterprise Edition 2009 provee despliegue de software rápido y confiable para la totalidad de su infraestructura de computación desde cualquier lugar. Kaseya facilita la tarea de instalar aplicaciones y actualizaciones (KASEYA, 2009).

Capítulo 1: Fundamentación Teórica.

Características principales:

- Despliegue de aplicaciones en base a un simple Wizard.
- Fácil de utilizar.
- Rápido despliegue.

Kaseya, a pesar de ser un consistente sistema para el despliegue, no satisface las necesidades del proceso de gestión de despliegue de software en la Facultad 4 ya que:

- Es un software costoso y que además no brinda soporte técnico a Cuba.
- No gestiona los recursos utilizados para el despliegue.
- No gestiona los posibles errores o problemas ocurridos durante el despliegue de la aplicación ya que sólo se enfoca a desplegar y luego a actualizar.
- No gestiona la planificación para el despliegue ni la capacitación del personal, ya que el mismo es el cliente del sistema que se desplegará.

Blaze Advisor 6.6 2009

Es un sistema de la *Fair Isaac Corporation* situada en Estados Unidos de Norteamérica, proporciona una plataforma con herramientas para la gestión del despliegue, estas herramientas permiten monitorizar, realizar un seguimiento y gestionar el estado de todos sus sistemas, ofreciendo compatibilidad con estándares del sector de forma instantánea. (Isaac, 2009).

Es un sistema altamente costoso, lo cual dificulta su explotación en nuestro país.

AdminStudio

Sistema perteneciente a la compañía Danysoft, compañía situada en Madrid, España, la cual ofrece el resultado de la unión de fuerzas de InstallShield y Macrovision, para ofrecer soluciones que permiten a los creadores de software y a sus clientes aumentar el valor que reciben de sus aplicaciones. Una vez desarrollada la aplicación, los encargados de distribuirla pueden utilizar las soluciones InstallShield para optimizar el despliegue. Con estas soluciones de instalación y gestión de licencias distribuidas por Danysoft, obtendrá la ayuda que necesita para optimizar el despliegue, a la vez que protege sus productos y controla todos los pasos de la distribución. Sus clientes empresariales obtendrán un valor añadido adicional con las herramientas que ponemos a su disposición para re empaquetar aplicaciones, resolver conflictos

Capítulo 1: Fundamentación Teórica.

potenciales, optimizar la adquisición de licencias y gestionar las actualizaciones (Danysoft.AdminStudio, 2009).

Es un sistema donde para adquirirlo hay que hacer pago de licencia, lo cual dificulta su explotación en nuestro país.

1.3.2. Análisis crítico sobre la situación de la gestión del despliegue de proyectos informáticos en nuestro país y en la Universidad de las Ciencias Informáticas.

Hoy Cuba se escala día a día y paso a paso en el inmenso mundo de la informática. El desarrollo y producción de software en nuestro país se encuentra en una fase de madurez y aún tiene mucho camino por recorrer. Las principales empresas líderes en Cuba en la creación de sistemas informáticos carecen de herramientas que gestionen todo el proceso de despliegue del sistema para los clientes finales por lo que es una de las principales dificultades y uno de los peldaños fundamentales para llevar la industria a la mayor eficiencia y madurez posible. La Universidad de las Ciencias Informáticas, forjada en el camino a la excelencia del desarrollo informático en Cuba es hoy uno de los centros de altos estudios y producción de sistemas informáticos en nuestro país. A pesar de que dicho Centro ha realizado numerosos despliegues de sistemas tanto nacionales como de exportación, el trabajo en busca de la eficiencia y el mejor servicio se impone. Aún no se cuenta con un sistema que gestione todo el proceso de despliegue de un software, proceso fundamental que demuestra al cliente la robustez y lo deseado convertido en sistema, luego de un estudio y un análisis crítico llegando a la conclusión de la necesidad de un sistema que gestione todo el proceso de despliegue de un software lo cual llevará a nuestra industria y con ella a nuestro Centro a escalones mayores en la producción de software y a lograr un menor costo a la hora de realizar dicho proceso así como en la satisfacción de nuestros clientes.

1.4. Fundamentación de la metodología a utilizar.

La rama de la metodología, dentro de la ingeniería de software, se encarga de elaborar estrategias de desarrollo de software, centradas en las personas o los equipos, orientadas hacia la funcionalidad y la entrega de comunicación intensiva y que requieren implicación directa del cliente. Su objetivo es elevar la calidad del software (en todas las fases por las que pasa) a través de una mayor transparencia y control sobre el proceso. Se analizarán aquí algunas metodologías de desarrollo de software que existen para luego fundamentar la seleccionada para el desarrollo del sistema. (IVAR JACOBSON, 2000).

1.4.1. Programación Extrema (XP, *eXtreme Programming*).

Es una de las metodologías ligeras de desarrollo de software utilizadas para proyectos de corto plazo y corto equipo. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. Intenta reducir la complejidad del software por medio de un trabajo orientado directamente al objetivo, basado en las relaciones interpersonales y la velocidad de reacción (ESCRIBANO, 2002).

XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. XP es recomendada para aumentar la velocidad de desarrollo de un producto, el cual fundamenta su desarrollo en los casos de prueba y en las historias de usuario (ANDRES, 2004).

Fases de la metodología XP:

- Fase 1 Planificación
- Fase 2 Diseño
- Fase 3 Desarrollo
- Fase 4 Prueba

1.4.2. Compendio de Soluciones de Microsoft (MSF, *Microsoft Solutions Framework*).

Microsoft Solutions Framework (MSF) es una serie de modelos flexibles interrelacionados que guían a una organización sobre cómo ensamblar los recursos, el personal y las técnicas necesarias para asegurar que su infraestructura tecnológica y sus soluciones cumplan los objetivos de negocio (Microsoft, 1997). MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.

MSF se compone de varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente el modelo de Aplicación, proporciona prácticas para planear, diseñar, convertir y desarrollar exitosas soluciones empresariales (SANCHEZ, 2004).

MSF define varias fases para el desarrollo del proyecto:

- Fase 1 - Estrategia y alcance.
- Fase 2 - Planificación y Prueba de Concepto

Capítulo 1: Fundamentación Teórica.

- Fase 3 - Estabilización.
- Fase 4 - Despliegue

1.4.3. Proceso Unificado de Desarrollo (RUP, *Rational Unified Process*).

El Proceso Unificado de Desarrollo es una metodología del Negociador Internacional de Computadoras IBM (*International Business Machines* por sus siglas en inglés) para el desarrollo y construcción de software basado íntegramente en UML (*Unified Modeling Language*) como soporte a la metodología. En RUP un proceso define quién está haciendo qué, cuándo y cómo alcanzar un determinado objetivo (IVAR JACOBSON, 2000).

El Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto (IVAR JACOBSON, 2000).

RUP establece las actividades y los criterios para conducir un sistema desde su máximo nivel de abstracción (la idea del cliente), hasta su nivel más concreto (un programa ejecutándose). Además soporta las técnicas orientadas a objetos.

RUP divide en 4 fases el desarrollo del software:

- Fase de Inicio: tiene como hito fundamental determinar la visión del proyecto.
- Fase de Elaboración: tiene como hito fundamental determinar la arquitectura óptima del sistema.
- Fase de Construcción: tiene como hito fundamental obtener la capacidad operacional inicial de la aplicación.
- Fase de Transición: tiene como hito fundamental obtener el release¹ del proyecto.

Fase del RUP	Hito
Inicio	Objetivos
Elaboración	Arquitectura

¹ release: Versión de un sistema de software que se libera.

Capítulo 1: Fundamentación Teórica.

Construcción	Funcionalidad operativa
Transición	Release ¹ del sistema

Tabla 1: Hitos de las fases del RUP.

En RUP se han agrupado las actividades en grupos lógicos definiéndose nueve flujos de trabajo principales. Los seis primeros son conocidos como flujos de ingeniería y los tres últimos como de apoyo o soporte. **Ilustración 1**

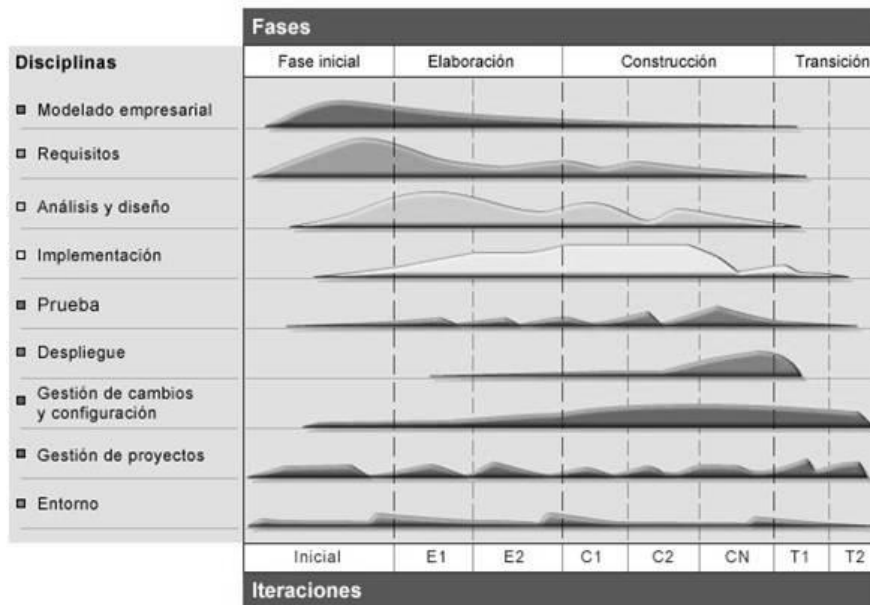


Ilustración 1: Flujos de trabajo y Fases de la metodología RUP. (IVAR JACOBSON, 2000).

Los flujos de trabajo de RUP son:

- Modelamiento del negocio: describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización.
- Requerimientos: define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.
- Análisis y diseño: describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos), por lo que indica con precisión lo que se debe programar.
- Implementación: define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación.
- Prueba (Testeo): Busca los defectos a lo largo del ciclo de vida.

Capítulo 1: Fundamentación Teórica.

- Despliegue: produce release del producto y realiza actividades (empaquete, instalación, asistencia a usuarios, etc.) para entregar el software a los usuarios finales.
- Administración del proyecto: involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes.
- Administración de configuración y cambios: describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a utilización/actualización concurrente de elementos, control de versiones, etc.

Ambiente: contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto; así como el procedimiento para implementar el proceso en una organización. (IVAR JACOBSON, 2000).

RUP	XP	MSF
Nivel de Funcionalidad	Nivel de Funcionalidad	Nivel de Funcionalidad
<p>*Dirigido por los Casos de Uso: los requerimientos funcionales son expresados en la forma de Casos de Uso, que guían la realización de una arquitectura ejecutable de la aplicación.</p> <p>*Centrado en la Arquitectura: el proceso focaliza el esfuerzo del equipo en construir los elementos críticos estructuralmente y del comportamiento (llamados Elementos Arquitecturales) antes de construir elementos menos importantes.</p> <p>*Proceso Iterativo e Incremental: particiona el ciclo de vida en iteraciones que producen versiones incrementales de los</p>	<p>*Pruebas Unitarias: se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándonos en algo hacia el futuro, se pueda hacer pruebas de las fallas que pudieran ocurrir. Es como si nos adelantáramos a obtener los posibles errores.</p> <p>*Re fabricación: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.</p> <p>*Programación en pares: una particularidad de esta metodología es que propone</p>	<p>*Adaptable: su uso es limitado a un lugar específico.</p> <p>*Escalable: puede organizar equipos tan pequeños entre 3 o 4 personas, así como también, proyectos que requieren 50 personas a más.</p> <p>*Flexible: es utilizada en el ambiente de desarrollo de cualquier cliente.</p> <p>*Tecnología Agnóstica: puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología. (SANCHEZ,</p>

Capítulo 1: Fundamentación Teórica.

ejecutables de la aplicación.

*Forma disciplinada de asignar tareas y responsabilidades.

Desarrollo basado en componentes. Utilización de un único lenguaje de modelación.

Proceso Integrado Único. Divide el proceso de desarrollo en ciclos.

(IVAR JACOBSON, 2000).

la programación en pares, la (2004).

cual consiste en que dos desarrolladores participen en un proyecto en una misma

estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está

haciendo en ese momento.

(ANDRES, 2002).

Tabla 2: Comparación de metodologías.

Para guiar la propuesta que presenta este trabajo, se realizó un estudio de las diferentes metodologías de desarrollo de software conocidas mundialmente. En el caso particular de este proyecto, se decidió la utilización de la metodología RUP por poseer un robusto y consistente nivel de funcionalidad, así como adaptabilidad y la Facultad cuenta con un equipo de desarrollo que posee una sólida experiencia en esta metodología.

Con RUP se presentan al cliente los roles y artefactos de cada fase, en este caso las fases a trabajar son Negocio, Requisitos y Análisis&Diseño. Los roles a desempeñar en estas fases y los artefactos a desarrollar para este trabajo son:

Roles	Artefactos
Analista de Procesos del Negocio	Actor del Negocio
Diseñador del Negocio	Caso de Uso del Negocio
	Trabajador del Negocio
	Entidades del Negocio
	Diagrama de CUN
	Descripciones textuales de los CUN
	Trabajador del Negocio
	Entidades del Negocio
	Diagrama de CUN
	Descripciones textuales de los CUN

Capítulo 1: Fundamentación Teórica.

	Diagrama de Actividades Modelo de Objetos Reglas del Negocio Glosario de Términos
Analista del Sistema Especificador de Requerimientos	Requisitos Funcionales (RF) y Requisitos No Funcionales (RNF) Actor del Sistema Casos de Uso del Sistema Diagrama de Casos de Uso del Sistema (DCUS) Descripciones textuales de los CUS Vista de CU
Diseñador Diseñador de Base de Datos Diseñador de Interfaz de Usuario	Mapa de Navegación Prototipos de interfaz de usuario Paquetes del Diseño Subsistemas del Diseño Clases del Diseño Realización de CU-diseño: diagramas de clases del diseño y diagramas de interacción Diagrama de Despliegue Modelo de Datos

Tabla 3: Roles y artefactos que desarrollan los Analistas y Diseñadores del RUP

1.5. Patrones.

El hombre durante su historia ha dominado muchas técnicas. El desarrollo de un software es una tarea complicada, la cual depende en gran medida de la experiencia del equipo de desarrollo. En ocasiones las operaciones se realizan de forma artesanal, los desarrolladores aprenden por un proceso de ensayo y errores y por transmisión de otros desarrolladores, se desarrollan técnicas generalmente aceptadas en el área de trabajo, lográndose un conocimiento común sobre cómo aplicar estas técnicas y además se crea una ciencia alrededor de la tarea; esto ha provocado la evolución de la utilización de estándares de solución a un problema de

Capítulo 1: Fundamentación Teórica.

forma efectiva y reutilizable, es decir la utilización de un patrón que facilite la solución del problema. El patrón describe un problema que ocurre infinidad de veces en nuestro entorno, así como la solución al mismo, de tal modo que se pueda utilizar esta solución un millón de veces más adelante sin tener que volver a pensarla otra vez.

1.5.1. Patrones de casos de uso

Los patrones de Casos de Uso son comportamientos que deben existir en el sistema, ayudan a describir qué es lo que el sistema debe hacer, es decir, describen el uso del sistema y cómo este interactúa con los usuarios. Estos patrones son utilizados generalmente como plantillas que describen cómo deberían ser estructurados y organizados los casos de uso, a continuación se describen algunos de los patrones a utilizar en el desarrollo del sistema (LARMAN, 1999).

No.	Nombre del Patrón	Características
1	Reglas de negocio	Este patrón se aplica a los casos del uso que modelan los servicios que son afectados por las reglas de negocio definidas, con el propósito de capturar lo que está o no está permitido en la organización, las reglas del negocio serán definidas explícitamente de manera que será más fácil cambiar el sistema software o adaptarlo al negocio, por lo que el negocio se hace mucho más ágil (GUNNAR ÖVERGAARD, 2004).
2	Extensión	El patrón de Extensión entre casos de uso consiste en que el comportamiento del caso de uso base se extienda bajo ciertas condiciones por otro caso de uso, especifica cómo el comportamiento definido por el caso de uso de extensión puede insertarse dentro del comportamiento definido por el caso de uso base; se utiliza entre casos de usos para ampliar los servicios que están opcionales en el sistema, permitiendo a los clientes del sistema decidir si hay que incluir un cierto servicio (GUNNAR ÖVERGAARD, 2004).
3	CRUD (<i>Creating, Reading, Updating and Deleting</i>)	CRUD es el patrón que consiste en un caso de uso para administrar la información, modelando todas las diversas operaciones que se puedan realizar de una

Capítulo 1: Fundamentación Teórica.

		parte de la información de cierta clase, tales como crearla, buscarla, modificarla y eliminarla, es utilizado cuando todos los flujos contribuyen al mismo valor de negocio y son todos cortos y simples (GUNNAR ÖVERGAARD, 2004).
4	Múltiples Actores	El patrón de Múltiples Actores cumplimenta las acciones de los actores, ellos representan como el sistema percibe su entorno. Se utiliza cuando dos actores interactúan diferentemente con un caso de uso y de manera alternativa cuando los dos actores desempeñan el mismo papel hacia el caso del uso, este papel es representado por otro actor, heredado por los actores que comparten este rol (GUNNAR ÖVERGAARD, 2004).

Tabla 4: Patrones de Casos de Uso.

1.5.2. Patrones de arquitectura.

Aquellos que expresan un esquema organizativo estructural fundamental para sistemas software (no son otra cosa que los estilos). A continuación se muestran las características del patrón de arquitectura empleado:

- **Modelo:** encapsula los datos y las funcionalidades. El modelo es independiente de cualquier representación de salida y/o comportamiento de entrada. El modelo debe de preservar la integridad de los datos.
- **Vista:** muestra la información al usuario. Pueden existir múltiples vistas del modelo. Cada vista tiene asociado un componente controlador. Interactúa con la interfaz de usuario.
- **Controlador:** reciben las entradas, usualmente como eventos, e interpreta las operaciones del usuario; codificando los movimientos, pulsación de botones del ratón, pulsaciones de teclas, etc. Los eventos son traducidos a solicitudes de servicio ("service requests") para el modelo o la vista. Es el que debe de controlar los eventos.
- Tiene como ventajas que se consiguen múltiples vistas del modelo. Todas las vistas están sincronizadas. No acoplamiento, y facilidad de evolución, para cambiar las vistas y los controladores. La aplicación puede soportar un tipo de interfaz para cada usuario (rol). (LARMAN,1999)

Capítulo 1: Fundamentación Teórica.

1.5.3. Patrones diseño.

Los patrones de diseño proponen una forma reutilizar la experiencia de los desarrolladores, para ello clasifica y describe formas de solucionar problemas que ocurren de forma frecuente en el desarrollo. Por tanto, están basados en la recopilación del conocimiento de los expertos en desarrollo de software (LARMAN, 1999). A continuación se muestran los principales patrones de diseño utilizados en la presente solución:

No.	Nombre del patrón	Características
1	Patrón Experto	Asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para cumplir la responsabilidad (LARMAN, 1999).
2	Patrón Bajo Acoplamiento	Asignar las responsabilidades de forma tal que las clases se comuniquen con el menor número de clases que sea posible (LARMAN, 1999).
3	Patrón Alta Cohesión.	Asignar a las clases responsabilidades que trabajen sobre una misma área de la aplicación y que no tengan mucha complejidad. <ul style="list-style-type: none">• Muy baja cohesión. Una clase es la única responsable de muchas cosas en áreas funcionales muy heterogéneas.• Baja cohesión. Una clase tiene la responsabilidad exclusiva de una tarea compleja dentro de un área funcional.• Alta cohesión. Una clase tiene responsabilidades moderadas en un área funcional y colabora con las otras para llevar a cabo las tareas (LARMAN, 1999).
4	Patrón Controlador.	Asignar la responsabilidad del manejo de mensajes de los eventos del sistema a una clase (LARMAN, 1999).

Tabla 5: Patrones de diseño GRASP.

A continuación se muestran los principales patrones GoF utilizados en la realización del diseño:

Capítulo 1: Fundamentación Teórica.

No.	Nombre del patrón	Características
1	Patrón de Creación, <i>Abstract Factory</i>	Encapsula la creación de objetos con distintas familias de implementaciones. Proporciona un interfaz para crear familias de objetos sin especificar su clase de forma concreta. Encapsula la creación de objetos con distintas familias de implementaciones. Conocido como Fábrica Abstracta, Kit. Proporciona un interfaz común para crear diferentes familias de clases. En muchas ocasiones existen diferentes recursos para realizar lo mismo, e incluso se presume que existirán nuevos recursos y queremos desarrollar sistemas compatibles con todos ellos. Si deseamos que nuestro software funcione sobre distintos recursos debemos abstraernos de qué librerías utilizamos (Saky, 2002).
4	Patrón Estructural, <i>Facade</i> .	Simplifica el acceso a un conjunto de clases o interfaces. Proporcionar una interfaz unificada de un subsistema sin ocultar sus interfaces. Permite acceder a elementos del sistema y realizar operaciones más complejas con ellos de forma transparente. Reduce la dependencia entre clases. Ofrece un punto de acceso al resto de clases, si éstas cambian o se sustituyen por otras, sólo hay que actualizar la clase Facade sin que el cambio afecte a las aplicaciones cliente. No oculta las clases, sino que ofrece una forma más sencilla de acceder a ellas. En los casos en que se requiere se puede acceder directamente a ellas (Saky, 2002).

Tabla 6: Patrones de diseño GoF.

1.6. Fundamentación de las herramientas a utilizar para el desarrollo del sistema.

La demanda del desarrollo de software llevó a la introducción de nuevos conceptos de interacción hombre-máquina, a la multiprogramación y a los sistemas multiusuario. La evolución de los sistemas de computadoras se caracterizó por el procesamiento distribuido (múltiples computadoras, cada una ejecutando funciones concurrentemente y comunicándose con alguna otra), incrementando notablemente

Capítulo 1: Fundamentación Teórica.

la complejidad de los sistemas informáticos. Además de los diversos problemas de software que confrontaban las organizaciones en el momento, por el poco alcance para solucionar dificultades como: la cantidad de software acumulado sin desarrollar, el excesivo mantenimiento de sistemas, la falta de documentación y el envejecimiento de las aplicaciones que enfrentaban muchas empresas. Como un nuevo intento por resolver estos problemas surge una herramienta llamada CASE² es una sigla que corresponde a las iniciales de Computer Aided Software Engineering; y en su traducción al español significa Ingeniería de Software Asistida por Computadora.

¿Qué es CASE?

Las herramientas CASE son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. CASE es también definido como el conjunto de métodos, utilidades y técnicas que facilitan el mejoramiento del ciclo de vida del desarrollo de sistemas de información, completamente o en alguna de sus fases (M.PANIAGUA, 2005).

Dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un software (Investigación Preliminar, Análisis, Diseño, Implementación e Instalación). Proporciona al ingeniero la posibilidad de automatizar actividades manuales y de mejorar su visión general de la ingeniería, las herramientas CASE ayudan a garantizar que la calidad se diseñe antes de llegar a construir el producto (PRESSMAN, 2005).

Para el desarrollo del sistema se analizarán algunas de las herramientas CASE posibles a utilizar en su construcción, justificándose la selección de la misma, teniendo en cuenta los conocimientos, experiencia y habilidades en procesamiento de transacciones que tiene el personal del equipo y la estructura que se desea adoptar en el proyecto.

1.6.1. Lenguaje Unificado de Modelado 2.0 (UML, *Unified Modeling Language*).

El Lenguaje Unificado de Modelado es un lenguaje gráfico para visualizar, especificar, construir, documentar y comunicar los artefactos de un sistema de software (IVAR JACOBSON, 1999). Utilizado para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir con la finalidad de describir modelos del sistema (del mundo real y del mundo del software), basados en los conceptos de objetos.

UML ofrece un estándar para describir un modelo del sistema, incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. El UML especifica varios diagramas, de ellos los diagramas de Casos de Usos y los diagramas de iteración, que son los artefactos concretos a partir de los cuales creamos modelos. Los diagramas UML se visualizan por medio de las vistas (proyecciones visuales del modelo). (LARMAN, 1999).

En lo que corresponde al desarrollo de programas UML, posee elementos gráficos para soportar la captura de requisitos, el análisis, el diseño, la implementación, y las pruebas, se compone de muchos elementos de esquematización que representan las diferentes partes de un sistema de software.

De forma general las principales características son:

Lenguaje unificado para la modelación de sistemas.

- Tecnología orientada a objetos.
- El cliente participa en todas las etapas del proyecto.
- Corrección de errores viables en todas las etapas.
- Aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y cliente/servidor.
- Facilita a los integrantes de un equipo multidisciplinario participar e comunicarse fácilmente. (LARMAN, 1999).

1.6.2. Visual Paradigm for UML 6.1 Enterprise Edition.

Visual Paradigm for UML 6.1 Enterprise Edition, es un software privativo para modelado en UML 2.0. Esta herramienta tiene unas características gráficas muy cómodas, que facilitan la realización de los diagramas de modelado que sigue el estándar de UML, los mismo son: Diagramas de clase, Casos de Uso, Comunicación, Secuencia, Estado, Actividad, Componentes, etc. Se integra con las siguientes herramientas Java: (VISUALPARADIGM, 2009)

- *Eclipse/IBM WebSphere*

² **CASE:** Computer Aided Software Engineering (Ingeniería de Software Asistida por Computadora).

- *JBuilder*
- *NetBeans IDE*
- *Oracle JDeveloper*
- *BEA Weblogic*

1.6.3. ArgoUML.

ArgoUML es una herramienta utilizada en el modelado de sistemas, mediante la cual se realizan diseños en UML llevados a cabo en el análisis y pre-diseño de sistemas de software; incluye el soporte para todos diagramas de UML 1.4 usuales. Funciona en cualquier plataforma de Java y está disponible en diez lenguas. (ARGOUML, 2009).

1.6.4. Rational Rose

Rational Rose es la herramienta Case desarrollada por los creadores de UML que cubren todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes y certificación de las distintas fases (SOFTWARE, 2007). Nos permite una trazabilidad real entre modelo (análisis y diseño) y el código ejecutable, domina el mercado de herramientas para el modelado, análisis, diseño y construcción orientada a objetos, tiene todas las características que los desarrolladores, analistas, y arquitectos exigen, desarrollo basado en componentes con soporte para arquitecturas líderes en la industria y modelos de componentes, facilidad de uso e integración optimizada. Es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros del equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de las grandes ventajas de *Rational Rose* es que utiliza la notación estándar en la arquitectura de software (UML), la cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto.

Además propone la utilización de cuatro tipos de modelo para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico (vista de Casos de Uso, vista Lógica, vista de Componentes y vista de Despliegue). A continuación se muestra una comparación entre las 3 herramientas CASE caracterizadas anteriormente, lo que ayudará a la fundamentación de la seleccionada para el modelado del sistema:

Capítulo 1: Fundamentación Teórica.

Visual Paradigm for UML 6.1 Enterprise Edition.	Rational Rose	ArgoUML
Funcionalidad	Funcionalidad	Funcionalidad
Entorno de creación de diagramas para UML 2.0	Admite como notaciones: UML, COM, OMT y Booch.	Multiplataforma (<i>Java 1.2</i>)
Diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad	Realiza Chequeo semántico de los modelos.	Meta modelo de UML estándar
Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.	Permite generar código a partir de modelos y viceversa.	Soporte para bases de datos
Capacidades de ingeniería directa e inversa.	Desarrollo multiusuario. Integración con modelado de datos. Generación de documentación.	Exporta los diagramas a distintos formatos
Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.	Tiene un lenguaje de script para poder ampliar su funcionalidad.	Generación de código (parcial)
Disponibilidad de múltiples versiones, para cada necesidad.	Disponibles en múltiples plataformas.	Soporte cognitivo: Proactivo (criticas de diseño, listas de cosas por hacer, correcciones automáticas)
Disponibilidad en múltiples plataformas.		Comprensión y solución del problema (perspectivas navegacionales, vistas superpuestas, representaciones alternativas de diseño: gráficos, texto, tablas).
Soporta aplicaciones web.		
Las imágenes y reportes generados, no son de muy buena calidad. Varios idiomas.		

Tabla 7: Comparación de herramientas.

Se utilizará Visual Paradigm for UML 6.1 Enterprise Edition ya que es un software para modelado de UML y además facilita la realización de los diagramas de modelado que

Capítulo 1: Fundamentación Teórica.

sigue el estándar de UML y se integra con herramientas Java, lenguaje de programación que se utilizará para la implementación del sistema.

1.7. Lenguaje de programación.

Java

Fue diseñado por la compañía *Sun Microsystems Inc.*, con el propósito de crear un lenguaje que pudiera funcionar en redes computacionales heterogéneas (redes de computadoras formadas por más de un tipo de computadora, ya sean PC, MAC's, estaciones de trabajo, etc.), y que fuera independiente de la plataforma en la que se vaya a ejecutar. Esto significa que un programa de Java puede ejecutarse en cualquier máquina o plataforma. La programación en Java, permite el desarrollo de aplicaciones bajo el esquema de Cliente Servidor, como de aplicaciones distribuidas, lo que lo hace capaz de conectar dos o más computadoras u ordenadores, ejecutando tareas simultáneamente, y de esta forma logra distribuir el trabajo a realizar (MENDEZ, 2009). Java como lenguaje de programación ofrece diversas características como ser fácil de usar y tomar lo mejor de otros lenguajes orientados a objetos para el desarrollo de aplicaciones. Es un lenguaje distribuido, proporcionando una colección de clases para su uso en aplicaciones de red, que permiten abrir sockets y establecer y aceptar conexiones con servidores o clientes remotos. Java es robusto ya que fue diseñado para crear software altamente fiable. Para ello proporciona numerosas comprobaciones en compilación y en tiempo de ejecución. Siendo un lenguaje seguro presenta barreras de seguridad en el lenguaje y en el sistema de ejecución en tiempo real.

Java es compilado en la medida en que su código fuente se transforma en una especie de código máquina. Por otra parte, es interpretado, ya que se puede ejecutar directamente sobre cualquier máquina a la cual se hayan portado el intérprete y el sistema de ejecución en tiempo real. Independiente de la arquitectura de hardware: Java está diseñado para soportar aplicaciones que serán ejecutadas en los más variados entornos de red, desde Unix a Windows NT, pasando por Mac y estaciones de trabajo, sobre arquitecturas distintas y con sistemas operativos diversos. (MENDEZ, 2009).

PHP

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor pero actualmente puede ser utilizado desde una interfaz de línea de

Capítulo 1: Fundamentación Teórica.

comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica (PHP, 2009).

Python

Python se utiliza como lenguaje de programación interpretado, lo que ahorra un tiempo considerable en el desarrollo del programa, pues no es necesario compilar ni enlazar. El intérprete se puede utilizar de modo interactivo, lo que facilita experimentar con características del lenguaje, escribir programas desechables o probar funciones durante el desarrollo del programa. El principal objetivo que persigue este lenguaje es la facilidad, tanto de lectura, como de diseño. Python permite dividir el programa en módulos reutilizables desde otros programas Python. Viene con una gran colección de módulos estándar que se pueden utilizar como base de los programas (Python, 2009).

Java JDK v6.0	PHP v5.0.3	Python
Diseño y Funcionalidad	Diseño y Funcionalidad	Diseño y Funcionalidad
<p>*Simple. Elimina la complejidad de los lenguajes como "C" y da paso al contexto de los lenguajes modernos orientados a objetos.</p> <p>*Orientado a Objetos. La filosofía de programación orientada a objetos es diferente a la programación convencional.</p> <p>*Familiar. C o en C++, la sintaxis de Java es muy similar al de estos.</p> <p>*Robusto. El sistema de Java maneja la memoria de la computadora por ti. No te tienes que preocupar por apuntadores, memoria que</p>	<p>*Es un lenguaje multiplataforma.</p> <p>*Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.</p> <p>*Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).</p> <p>*Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en</p>	<p>*Es un lenguaje de programación multiparadigma. Esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación estructurada y programación funcional.</p> <p>*Usa tipo de dato dinámico y reference counting para el manejo de memoria.</p> <p>*La resolución dinámica de nombres, o ligadura dinámica de métodos.</p> <p>*Puede utilizarse como un lenguaje de extensión para módulos y aplicaciones que</p>

Capítulo 1: Fundamentación Teórica.

no se esté utilizando, etc.	un único archivo de ayuda.	necesitan de una interfaz programable (Python, 2009).
*Seguro. El sistema de Java tiene ciertas políticas que evitan se puedan codificar virus con este lenguaje.	*Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.	
*Portable. Como el código es interpretado, un programa compilado de Java puede ser utilizado por cualquier computadora que tenga implementado el interprete de Java .	*Permite las técnicas de Programación Orientada a Objetos.	
	*Biblioteca nativa de funciones sumamente amplia e incluida.	
	*No requiere definición de tipos de variables.	
*Independiente a la arquitectura. Al compilar un programa este código es interpretado por diferentes computadoras de igual manera, solamente hay que implementar un intérprete para cada plataforma. De esa manera Java logra ser un lenguaje que no depende de una arquitectura computacional definida.	*Tiene manejo de excepciones (desde PHP5). (PHP, 2009)	
*Multithreaded. Un lenguaje que puede ejecutar diferentes líneas de código al mismo tiempo.		
*Interpretado. Java corre en máquina virtual, por lo tanto es interpretado.		
*Dinámico. Java no requiere que compile todas las clases de un programa para		

que este funcione. Si realizas una modificación a una clase Java se encarga de realizar un Dynamic Bynding o un Dynamic Loading para encontrar las clases (MENDEZ, 2009).

Tabla 8: Comparación de lenguajes de programación.

Se utilizará como lenguaje de programación para el desarrollo de la aplicación Java JDK 6.0 por su robustez, fiabilidad y seguridad.

1.7.1. Spring Framework.

Es un framework de código abierto orientado al desarrollo de aplicaciones para la plataforma Java. Fue creado por Rod Johnson, quien lo describió por primera vez en su libro “Expert One-on-One Java EE Design and Development”. Es el más popular y el más ambicioso de todos los framework de peso ligero. Es el único framework que interviene en todas las capas arquitectónicas de una aplicación JEE. Además está diseñado para facilitar una flexibilidad arquitectónica. (Johnson, 2005).

Los principales valores de Spring, según Rod Johnson, se pueden resumir en: no es agresivo, provee un modelo consistente de programación, ayuda a promover la reusabilidad de código, facilita el diseño Orientado a Objetos en aplicaciones JEE, permite la extracción de valores de configuración desde el código java a archivos XML³ o archivos de propiedades, está diseñado a fin de que las aplicaciones lo usen para que las pruebas sean lo más fácil posible. Spring hace de soluciones existentes un uso más fácil, dentro de una arquitectura consistente. (Areces Gonzalez, 2008).

Además de esto, Spring provee en su paquete Spring Web MVC una implementación del patrón MVC⁴, el cual brinda una limpia y clara separación entre las 3 capas arquitectónicas: Acceso a Datos, Negocio y Presentación. Esta implementación facilita una serie de clases controladoras, que manejan la lógica de la navegación e interactúa con la capa de Negocio de la aplicación web. (Areces Gonzalez, 2008).

3

³Extensible Markup Language, lenguaje de marcas ampliable, es un metalenguaje extensible de etiquetas. Permite definir la gramática de lenguajes específicos.

⁴Modelo Vista Controlador, es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

1.8. Sistema Gestor de Base de Datos.

PostgreSQL v8.3

Es un sistema diseñado para administrar grandes cantidades de datos, que tiene la fama de ser la base de datos de código abierto (*Open Source*) más avanzada del mundo. El PostgreSQL se ejecuta en la mayoría de los Sistemas Operativos más utilizados en el mundo incluyendo Linux, varias versiones de UNIX y por supuesto Windows. Se ha preocupado por ser una solución real a los complejos problemas del mundo empresarial y a la vez mantener la eficiencia al consultar los datos.

Con ese fin, se han desarrollado y añadido al PostgreSQL las más interesantes y útiles características que antes sólo podían hallarse en sistemas manejadores de bases de datos comerciales como Oracle, DB2 o Sybase, lo cual lo coloca, como su lema indica, como "el manejador de bases de datos de código abierto más avanzado del mundo". (PostgreSQL, 2007).

MySQL v5.0.3

Es un sistema de gestión de bases de datos relacionales, multihilo y multiusuario, es muy utilizado en aplicaciones web implementadas en Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. Es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones (MySQL, 2009).

PostgreSQL v8.3	MySQL v5.0.3
Diseño y Funcionalidad	Diseño y Funcionalidad

Capítulo 1: Fundamentación Teórica.

DBMS Objeto-Relacional.	Es un sistema de administración relacional de bases de datos
PostgreSQL aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas.	Es software de fuente abierta
Altamente Extensible.	Disponibilidad en gran cantidad de plataformas y sistemas.
PostgreSQL soporta operadores, funciones métodos de acceso y tipos de datos definidos por el usuario.	Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
Integridad Referencial.	Conectividad segura.
PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.	Replicación.
API Flexible.	Búsqueda e indexación de campos de texto.
La flexibilidad del API de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para el RDBMS PostgreSQL. Estas interfaces incluyen Object Pascal, Python, Perl, PHP, ODBC, Java/JDBC, Ruby, TCL, C/C.	Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
Cliente/Servidor.	Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor (MySQL, 2009).
PostgreSQL usa una arquitectura proceso-por-usuario cliente/servidor . Hay un proceso maestro que se ramifica para proporcionar conexiones adicionales para cada cliente que intente conectar a PostgreSQL (González, 2009).	

Tabla 9: Comparación de Sistemas Gestores de bases de datos.

Se utilizará como gestor de Base Datos el PostgreSQL v8.3 por su alta extensibilidad, integridad y robustez.

Capítulo 1: Fundamentación Teórica.

Como servidor web se seleccionó el Apache Tomcat v6.0.18. Es un servidor con soporte de servlets y JSPs. Incluye el compilador Jasper, que compila JSPs convirtiéndolas en servlets. Tomcat puede funcionar como servidor web por sí mismo. Es usado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad. Dado que Tomcat fue escrito en Java, funciona en cualquier sistema operativo que disponga de la máquina virtual Java (Apache-Tomcat, 2009).

1.9. Conclusiones

Luego de un estudio exhaustivo de herramientas, metodologías, tecnologías y de los procesos relacionados con la gestión del despliegue de un proyecto de software se arribaron a las siguientes conclusiones:

- La informatización del proceso de gestión de despliegue de los proyectos productivos de la Facultad 4 constituye una necesidad para el funcionamiento eficiente del Vicedecanato de Producción e Investigación de la Facultad.
- La gestión del despliegue de soluciones de software son, dentro del proceso completo de desarrollo los más costosos y están entre los más complejos, debido a la carga de personal necesario para realizar la gestión, al numeroso grupo de actividades que conlleva, al soporte que se mantendrá al software, y a que este es el proceso que se realiza con la participación directa de los usuarios finales del software. La capacitación de los usuarios finales del software implica la utilización adicional de recursos materiales y humanos, además de la necesidad de realizar esta tarea en el tiempo más corto posible, lo que la convierte en la tarea más compleja del proceso de despliegue.
- En los análisis realizados a sistemas similares existentes a escala nacional e internacional se concluyó que ninguno se ajusta al proceso de gestión de despliegue de los proyectos productivos de la Facultad 4, por lo que es necesario realizar un sistema que satisfaga las necesidades de la Facultad.

Capítulo 2: Descripción de la Solución.

2.1. Introducción.

Las necesidades reales de los usuarios son desafíos comunes en el proceso de desarrollo de un software de calidad, en este sentido las buenas prácticas en la ingeniería de software componen la base para la calidad de los proyectos. Como se había mencionado en epígrafes anteriores el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que define roles, fases para todo el ciclo de vida del software, se crean artefactos como resultado de cada iteración, describe cómo obtener los requerimientos, cómo organizarlos, cómo documentar los requerimientos de funcionalidad y las restricciones, cómo rastrear y documentar decisiones y cómo captar y comunicar requerimientos del sistema, de estos aspectos se hace referencia en este Capítulo.

En el presente Capítulo se realiza el modelado del negocio según el Proceso Unificado, así como se identifican los actores, casos de uso y trabajadores del negocio y se describen todos los requerimientos que tendrá el sistema. Se muestran los diagramas de actividades y las descripciones de los casos de uso así como las reglas del negocio actual; además de los requisitos funcionales y no funcionales que debe tener el sistema que se propone y su modelado.

2.2. Modelado del Negocio.

El propósito del Modelado del Negocio es entender la estructura y dinámica de la organización, entender los problemas actuales e identificar las mejoras potenciales que se pueden lograr, asegurarse que los clientes, usuarios finales y desarrolladores tengan una misma idea de la organización. Además de entender el contexto en el cual el problema se va a enmarcar. Es decir, lograr un dominio global del problema objeto de estudio y delimitar las fronteras del mismo e identificar los procesos involucrados, donde los procesos se hallan sujetos a un conjunto de reglas de negocio, que determinan las políticas y la estructura de la información.

Actualmente la gestión del despliegue de los proyectos productivos de la facultad 4 no se realiza de forma automatizada, los proyectos que hasta el momento han realizado el despliegue de su sistema lo han desarrollado enfocándose al usuario, es decir a lo que el usuario final necesita, RUP en una de sus fases, la de transición, enmarca algunos de los procesos a llevar a cabo a la hora de desplegar un software, en la Facultad 4 han utilizado para el desarrollo de despliegues de varios proyectos

Capítulo 2: Descripción de la Solución.

productivos los elementos definidos por la metodología RUP como apoyo de tan importante proceso. La falta de un proceso de gestión más robusto y orientado a un buen producto al usuario final es el obstáculo que actualmente en la Facultad 4 se presenta a la hora de realizar un despliegue de proyecto.

El negocio que se estudia en el presente trabajo de diploma consiste en cada proyecto de la Facultad 4 que se esté desarrollando y entre en la etapa del despliegue. A continuación se presenta el flujo de actividades que propone el Proceso Unificado para la ejecución del proceso de despliegue de un producto informático. **Ilustración 2.**

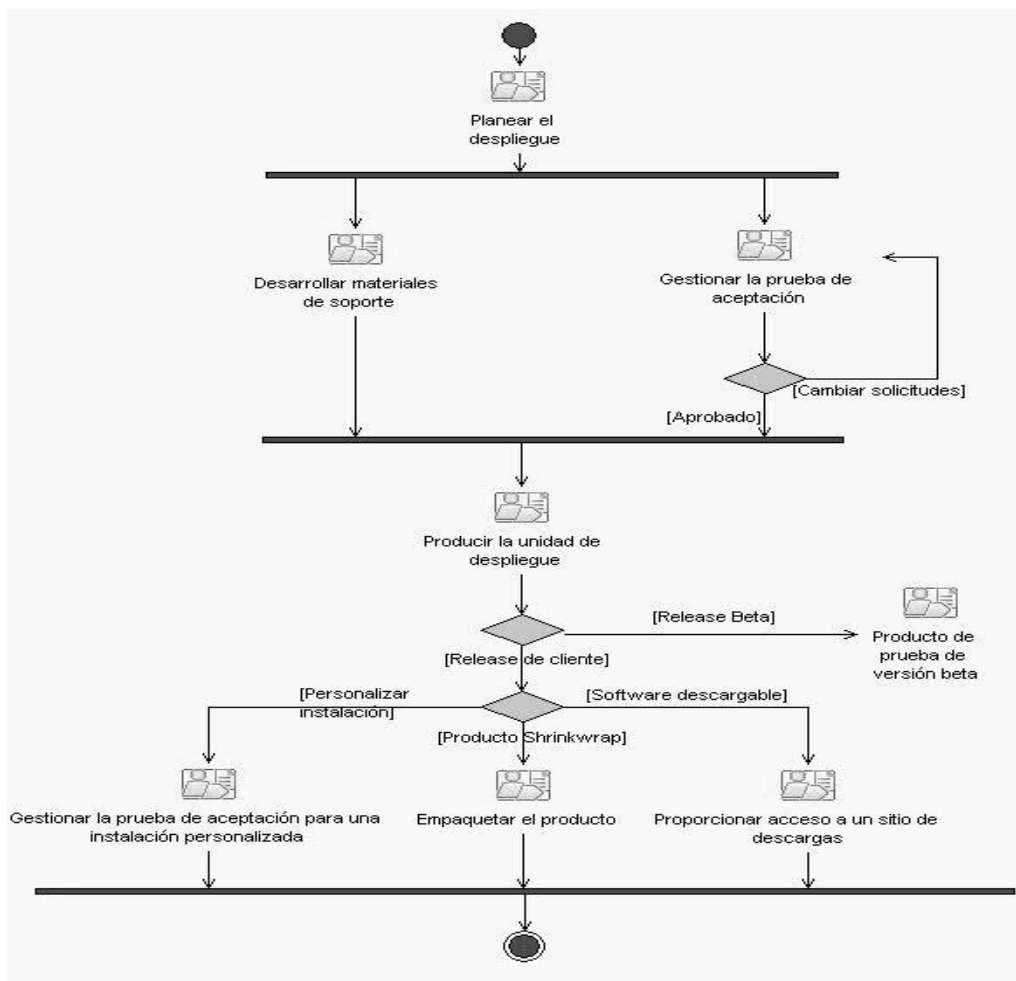


Ilustración 2: Flujo de Actividades de la etapa de despliegue. (Rational Unified Process Help, 2006)

Procesos que actualmente se llevan a cabo para desplegar un sistema informático tanto nacional como de exportación en la Facultad 4.

- Elaboración de la planificación de despliegue donde se definen todas las tareas de despliegue así como fecha de su desarrollo, personal que estará en función

Capítulo 2: Descripción de la Solución.

de desplegar el sistema, materiales y recursos necesarios para el desarrollo del mismo así como la participación directa del usuario final.

- Elaboración de los elementos para la configuración del sistema, instalación y sistema operativo con el cual funcionará.
- Ejecutar la Prueba de Aceptación la cual se realiza con el fin de probar el sistema y verificar que cumple con lo deseado y requerido por el cliente.
- Redactar el Acta de Liberación del producto luego de realizar la prueba de aceptación el producto es revisado por el Centro de Calidad de la universidad este en caso necesario emite notas de liberación que no son más que nuevas características y cambios ocurridos durante la liberación y finalmente se aprueba y se firma la prueba de aceptación.
- Ejecutar la Prueba Inicial o Piloto luego de aprobarse la prueba de aceptación se procede a desarrollar una prueba inicial o prueba beta al producto para el cliente final con el objetivo de probar la eficiencia y solidez del sistema y el cumplimiento de lo requerido por el cliente.
- Brindar el Soporte del Sistema es decir los materiales de apoyo, manuales de usuarios, además de brindar también actualización al sistema.
- Brindar Materiales de Entrenamiento es decir cursos, manuales para el usuario final con el objetivo de adiestrarlo en el sistema que utilizarán.

2.2.1 Actores del Negocio.

Un candidato a actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos, que interactúa con el negocio (IVAR JACOBSON, 2000). El término actor significa el rol que algo o alguien juega cuando interactúa con el negocio, para beneficiarse de sus resultados.

Actores del Negocio	Descripción
Vicedecano(a) de Producción	Es la persona que está al mando de todo lo correspondiente a la producción de la Facultad desde todos los proyectos que se encuentran en la misma y la situación de los mismos si se encuentran en desarrollo si están desplegando o demás.

Capítulo 2: Descripción de la Solución.

Ciente	Es la persona que se beneficiará de los resultados del sistema.
Personal de Calidad de software	Son las personas encargadas de realizar pruebas de calidad al sistema para consolidar el producto final, si luego de realizar todas las pruebas de calidad es aprobado, entonces el personal de calidad emite la liberación del producto en un acta de aprobación y firmada.

Tabla 10: Actores del Negocio.

2.2.2 Caso de Uso del Negocio.

Los casos del uso del negocio representan un proceso dentro del negocio que se estudia, por lo que se corresponde con una secuencia de acciones con un orden lógico y que producen un resultado observable para ciertos actores del negocio (IVAR JACOBSON, 2000). Se utiliza por los analistas del proceso negocio y los diseñadores del negocio para entender y mejorar la manera que los clientes obran con su ambiente de negocio, además para proporcionar el contexto para el desarrollo del software.

- Elaborar Plan de Despliegue.
- Identificar Elementos del Despliegue.
- Realizar Liberación del Producto.
- Administrar proceso de pruebas de aceptación.
- Administrar proceso de pruebas piloto.
- Realizar Transferencia Tecnológica
- Brindar Soporte al Sistema.

2.2.3 Trabajadores del Negocio.

Un trabajador del negocio es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado (software); que actúa en el negocio realizando una o varias actividades que están comprendidas dentro de un caso de uso, interactuando con otros trabajadores del negocio y manipulando entidades del negocio (IVAR JACOBSON, 2000). Representa un rol y desarrollan las acciones que posibilitan que se cumpla la función de un proceso de negocio.

Capítulo 2: Descripción de la Solución.

Trabajador del Negocio	Descripción
Líder del Proyecto	Es la persona que está a cargo del proyecto desde su inicio hasta su fin, lidera, controla y gestiona toda la información y procesos del producto a desarrollar.
Administrador del Despliegue	Es la persona encargada como su rol lo indica a llevar a cabo todo el despliegue del software, controla y elabora los principales artefactos que se desarrollan en esta fase, gestiona toda la información referente al despliegue así como el plan de despliegue etc.
Implementador	Es el encargado de desarrollar todos los artefactos para la instalación del sistema.
Cliente	Es la persona que se beneficiará de los resultados del sistema.
Personal de Calidad	Son las personas encargadas de realizar pruebas de calidad al sistema para consolidar el producto final, si luego de realizar todas las pruebas de calidad es aprobado, entonces el personal de calidad emite la liberación del producto en un acta de aprobación y firmada.

Tabla 11: Trabajadores del Negocio

2.2.4 Entidades del Negocio.

Las entidades del negocio no son más que contenedores de información, es decir planillas, documentos, que brindan información sólida del negocio.

- Acta de Liberación del Producto.
- Acta de Aceptación del producto.
- Manual de Instalación.

Capítulo 2: Descripción de la Solución.

- Planilla de No Conformidades.
- Acta de Aceptación del Pilotaje del Producto.

2.3 Diagrama de CUN.

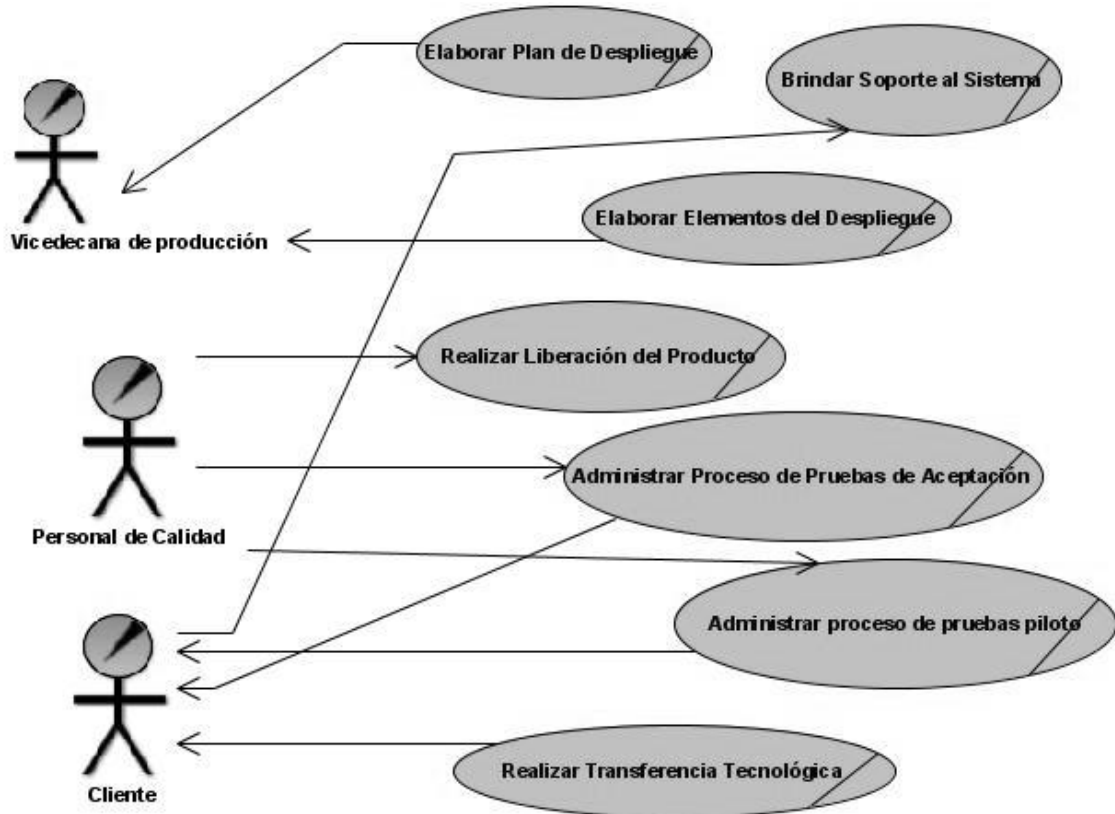


Ilustración 3: Diagrama de Casos de Uso del Negocio: despliegue de productos de software de la Facultad 4.

2.3.1 Descripciones textuales de los Casos de Uso del Negocio.

Las descripciones textuales de los casos de uso permiten detallar cada una de las actividades que se definen en el proceso del negocio, con carácter narrativo, se utiliza para especificar la secuencia de actividades que se desarrolla entre los participantes del proceso.

Tabla 12: Descripción del CUN Elaborar Plan de Despliegue.

Caso de Uso del Negocio	Elaborar Plan de Despliegue
Actores	Vicedecana de Producción, Cliente
Trabajador	Administrador de Despliegue
Propósito	El propósito de este caso de uso es obtener el Plan de Despliegue de cada proyecto de la Facultad que entre en esta fase de su etapa de desarrollo.

Capítulo 2: Descripción de la Solución.

Resumen:	
El caso de uso se inicia cuando el Administrador de Despliegue elabora el Plan de Despliegue y se lo envía a la Vicedecana de Producción para ser aprobado, si esta lo aprueba entonces se envía al Cliente, el caso de uso termina cuando es aprobado, por el Cliente el Plan de despliegue.	
Curso Normal de los Eventos	
Acción del actor	Respuesta del proceso de negocio
	1- El Administrador de Despliegue elabora el Nombre del proyecto desplegar, Alcance, Definiciones, Acrónimos, y Abreviaturas, Referencias, Planificación del Despliegue, Responsabilidades, Recursos, Facilidades, Hardware, Unidad de Despliegue, Soporte del Software, Documentación del Soporte, Personal para el Soporte, Adiestramiento de los Usuarios, elementos que conforman el Plan de Despliegue. 2- El Administrador de Despliegue entrega el Plan de Despliegue a la Vicedecana de Producción para que sea aprobado.
3- La Vicedecana de Producción recibe y aprueba el Plan del Despliegue.	
	5- El Administrador de Despliegue envía el Plan de Despliegue al Cliente para que este de su visión sobre el mismo.
6- El Cliente recibe el Plan de Despliegue, lo consulta y lo aprueba.	
Prioridad	Crítico
Flujos Alternos	
Acción del actor	Respuesta del proceso de negocio
	3.1 Si detecta algún error en el Plan de despliegue se le informa al Administrador de Despliegue para que sea corregido.
	6.1 Si el cliente desea añadir o reestructurar algún elemento del plan lo informa al Administrador de Despliegue.
Mejoras	Se informatizará la planilla “Plan de Despliegue”.

Tabla 13: Descripción del CUN Elaborar Elementos del Despliegue.

Caso de Uso del Negocio	Elaborar Elementos del Despliegue
--------------------------------	-----------------------------------

Capítulo 2: Descripción de la Solución.

Actores	Vicedecana de Producción
Trabajador	Implementador
Propósito	El propósito de este caso de uso es crear una unidad de despliegue que conste del software a desplegar, así como de los artefactos acompañantes necesarios para realizar eficazmente la instalación del sistema, dígame instrucciones de instalación, archivos del sistema.
Resumen: El caso de uso se inicia cuando el Implementador elabora los elementos para el desarrollo del despliegue se los muestra a la Vicedecana de Producción para ser consultados y aprobados y termina cuando la Vicedecana de Producción aprueba los elementos para el despliegue.	
Curso Normal de los Eventos	
Acción del actor	Respuesta del proceso de negocio
	1-El Implementador elabora la Instalación de secuencias de comandos, Configuración de Archivos, Instrucciones de instalación, Archivos del Sistema, los elementos para el desarrollo del despliegue. 2-El Implementador entrega los elementos para el proceso de despliegue a la Vicedecana de Producción para ser consultados y aprobados.
3.- La Vicedecana de Producción recibe los elementos para el desarrollo del despliegue, los revisa y los aprueba.	
Flujos Alternos	
Acción del actor	Respuesta del proceso de negocio
	3.1 Si detecta algún error en los elementos de despliegue se le informa al Implementador para que sea corregido.
Mejoras	Se informatizará el registro y aprobación de los elementos de despliegue

Tabla 14: Descripción del CUN Administrar proceso de pruebas de aceptación.

Caso de Uso del Negocio	Administrar proceso de pruebas de aceptación
Actores	Personal de Calidad(inicia), Cliente
Trabajador	Líder del Proyecto, Administrador de Despliegue, Implementador
Propósito	Controlar el proceso de las pruebas de aceptación que ejecuta el Cliente, para corroborar que el sistema desplegado cumple con todos los requisitos para el cual se concibió.

Capítulo 2: Descripción de la Solución.

Resumen:	
El caso uso se inicia cuando el Personal de Calidad procederá con la administración del proceso de pruebas de aceptación conjuntamente con el personal de desarrollo del sistema: el líder del proyecto y el administrador del proyecto además del Cliente, el caso de uso termina cuando es finalizado el proceso de pruebas y es aceptado por el cliente el producto.	
Curso Normal de los Eventos	
Acción del actor	Respuesta del proceso de negocio
1.- El Personal de Calidad según el calendario del Plan de Despliegue y del Plan de Pruebas de Aceptación (que es obtenido en el FT Pruebas), comienzan el proceso de pruebas de aceptación con el Cliente.	
2- El cliente ejecuta las pruebas de aceptación. Si detecta alguna No Conformidad se pone en la Planilla de No Conformidades y al concluir las pruebas le envía al Implementador dicha planilla.	
	3- El Implementador revisa la Planilla de No Conformidades y procede a resolver las No Conformidades
5- El Personal de Calidad elabora el Acta de Aceptación del Producto.	
6- El Cliente firma el Acta de Aceptación del Producto	
	7 -El Administrador de Despliegue recoge el Acta de Aceptación del Producto y se continúa con el proceso del despliegue.
Prioridad	Crítico
Flujos Alternos	
Acción del actor	Respuesta del proceso de negocio
	2.1 Si el cliente no encuentra alguna No Conformidad entonces finalizan las pruebas se elabora el acta y se aprueba.
Mejoras	Se informatizará en la administración del proceso de pruebas de aceptación, las No Conformidades, el estado de las pruebas y el Acta de Aceptación del Producto

Ver **Anexo 2** para las demás descripciones de Caso de Uso del Negocio.

Capítulo 2: Descripción de la Solución.

2.3.2 Diagramas de actividades los Casos de Uso del Negocio.

Un diagrama de actividad describe un proceso que explora el orden de las tareas o actividades que logran los objetivos del negocio, establece cuándo inicia, cómo se ejecuta y cuándo termina. Además es posible determinar cuáles acciones ocurren en paralelo o cuál secuencia de actividades o subactividades siguen luego de determinar el valor de una condición. Permite la administración adecuada y una visión simplificada de la complejidad del proceso que se está modelando como descripción de la secuencia de actividades que se ejecutan.

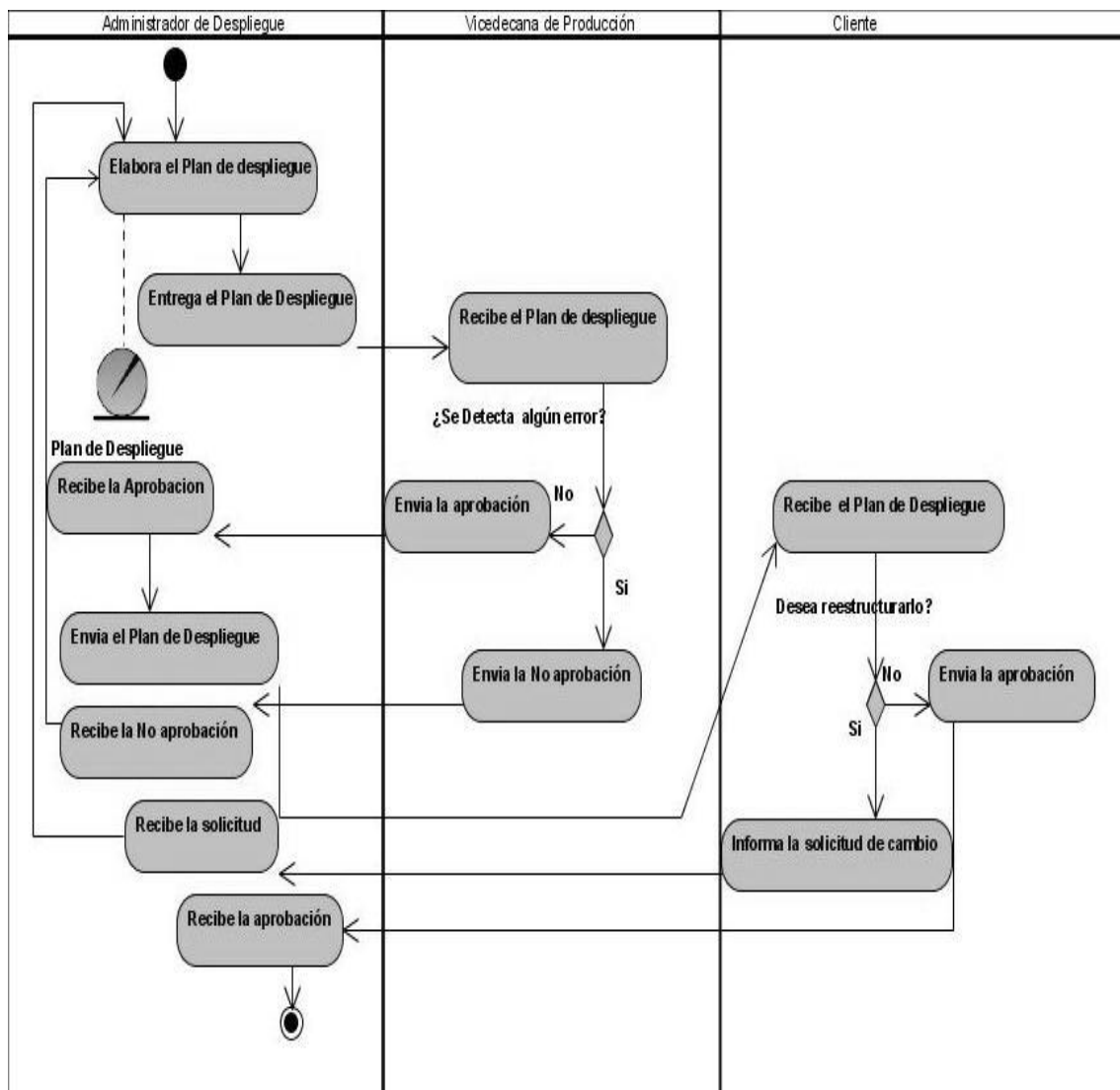


Ilustración 4: Diagrama de Actividades del Caso de Uso Elaborar Plan de Despliegue.

Capítulo 2: Descripción de la Solución.

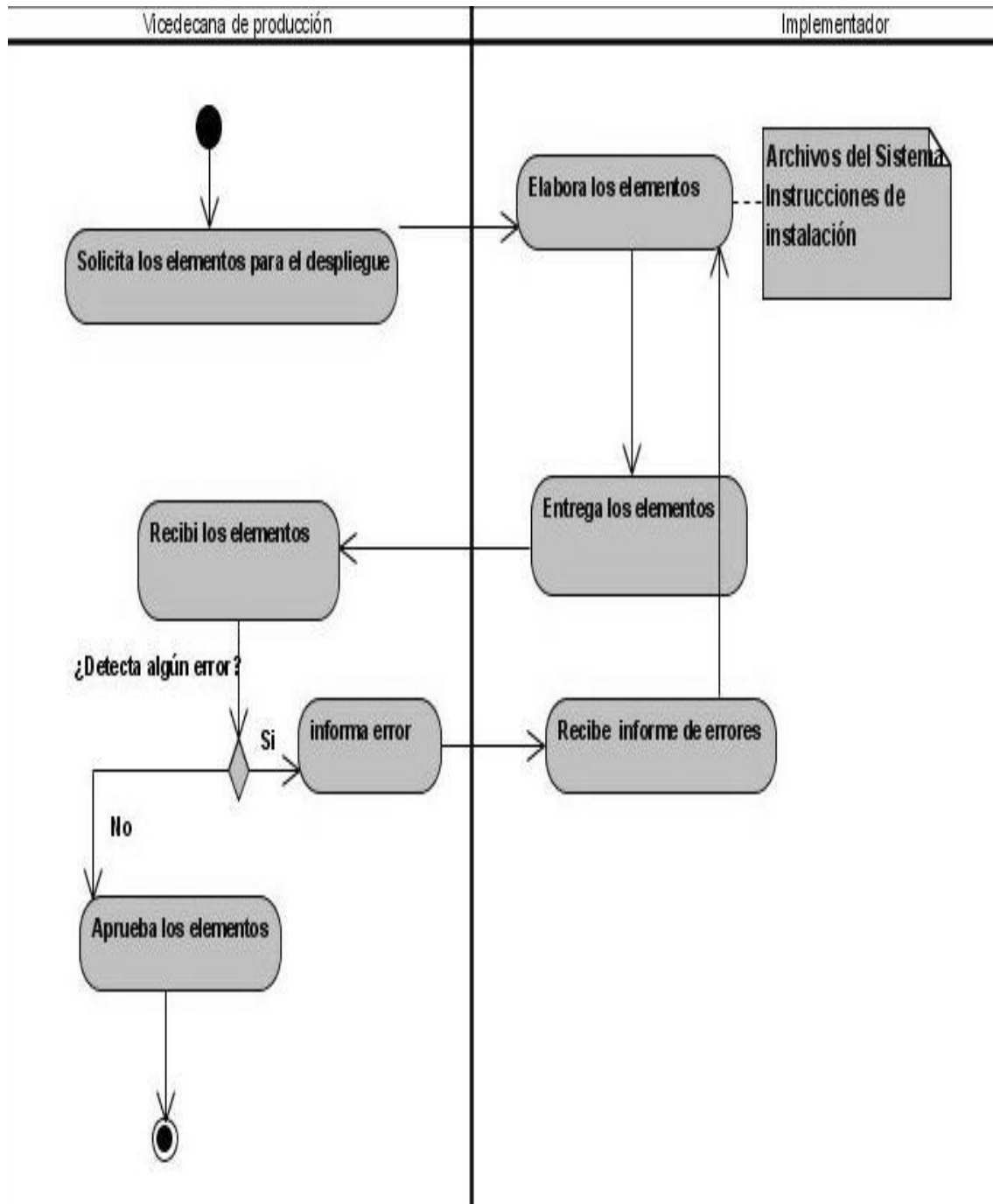


Ilustración 5: Diagrama de Actividades del Caso de Uso Elaborar Elementos del Despliegue.

Capítulo 2: Descripción de la Solución.

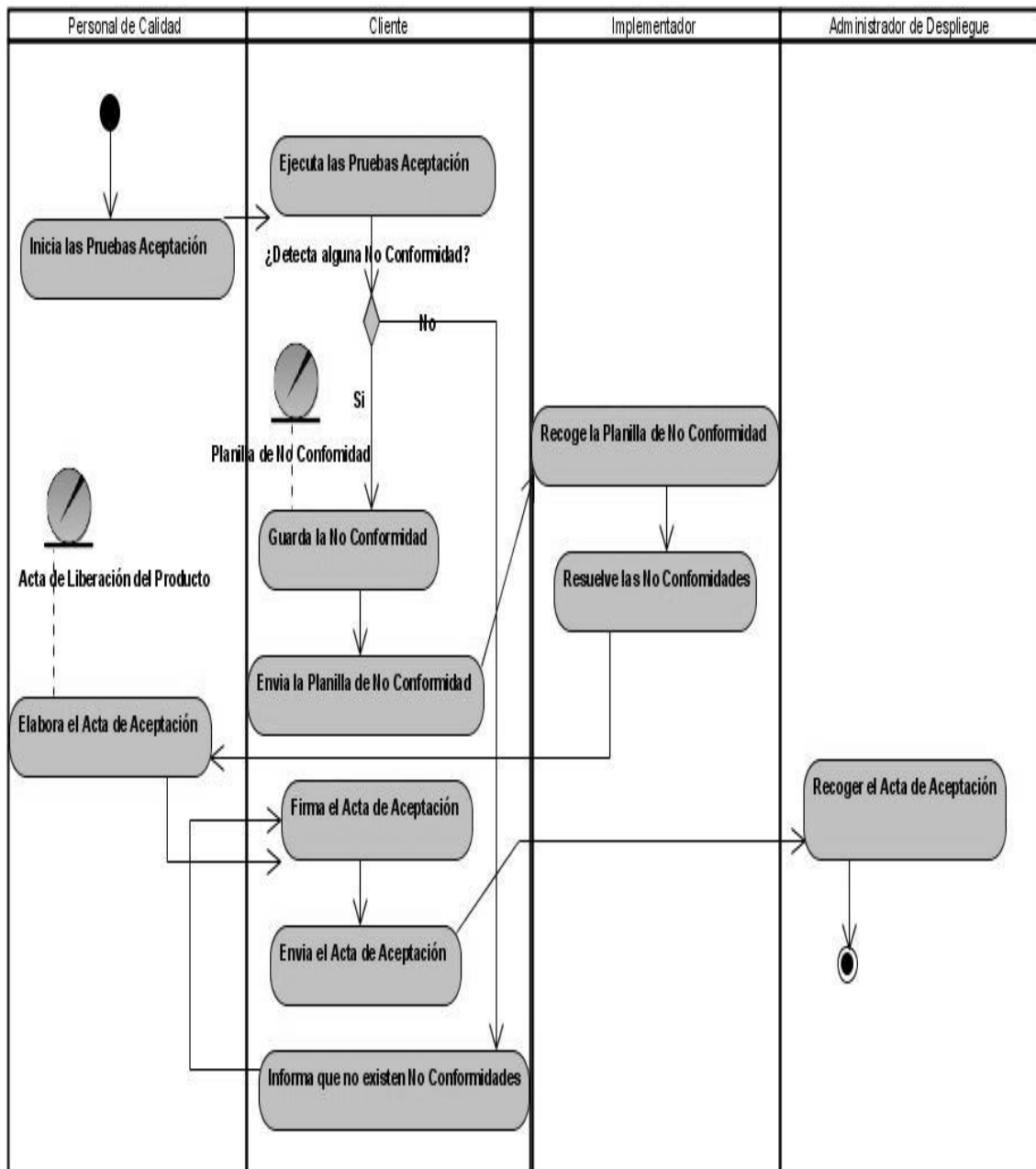


Ilustración 6: Diagrama de Actividades del Caso de Uso Administrar proceso de pruebas de aceptación.

2.3.3 Modelo de Objetos.

El modelo de Objetos del Negocio describe la realización de cada caso de uso del negocio, estableciendo los trabajadores y la información que en términos generales manipulan, asociados al caso de uso del negocio. Es decir, sirve como abstracción de cómo los trabajadores del negocio y las entidades de negocio necesitan ser relacionados y colaborar para realizar el negocio.

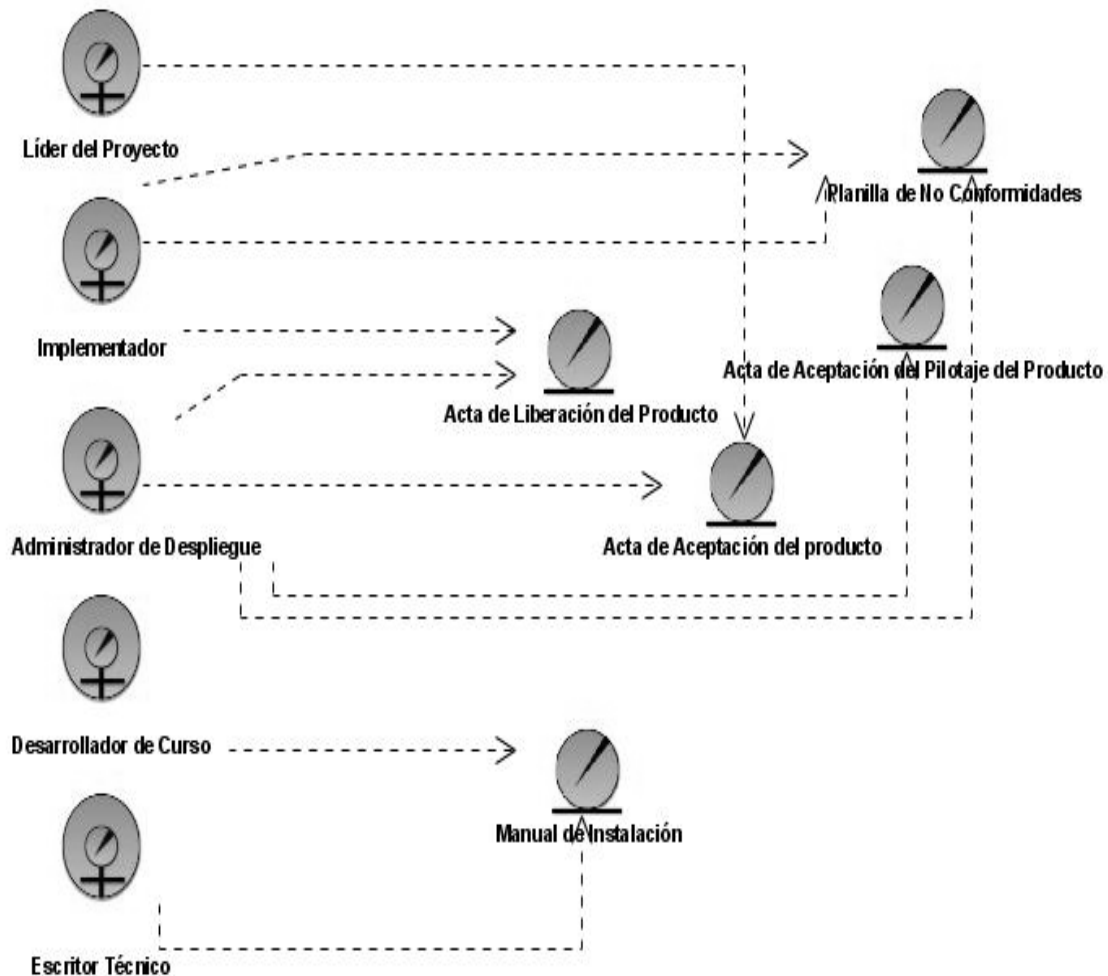


Ilustración 7: Modelo de Objetos del Negocio: despliegue de productos de software de la Facultad 4.

2.4 Reglas del Negocio.

Describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio.

Hay que identificarlas dentro del negocio, evaluar si son relevantes dentro del campo de acción que se está modelando e “implementarlas” en la propuesta de solución.

Reglas del Negocio a considerar.

- Un producto para ser desplegado tiene que haber sido liberado anteriormente por el Grupo de Calidad de la y Facultad y haber pasado las pruebas de aceptación con el cliente.
- El Arquitecto y los principales programadores del equipo de desarrollo participan en las respuestas a las No Conformidades que se detecten durante las pruebas de liberación y las pruebas de aceptación del cliente.

Capítulo 2: Descripción de la Solución.

- El Arquitecto y el Líder del proyecto son los responsables principales del proceso de pilotaje y despliegue de la aplicación en el entorno del cliente.
- La Vicedecana de Producción es la que decide cuándo un software entra en Pruebas de Liberación y cuándo se puede proceder a desplegar el mismo, teniendo el visto bueno del Grupo de Calidad de la Facultad y la opinión del líder de proyecto.
- Cuando se lleva a cabo una prueba al software y se encuentra una nueva no conformidad, tiene que agregarse a la Planilla de No Conformidades (planilla que emite el Personal de Calidad) correspondiente a la prueba que se este realizando en ese momento.
- Si una prueba, después de todas las repeticiones posibles, da definitivamente No Conforme, este resultado se informa al Equipo de Desarrollo de software con la Planilla de No Conformidades.

2.5 Modelado del Sistema.

Quedando definidas las funcionalidades del sistema, documentadas con un nivel apropiado y haciendo uso de los patrones de caso de uso mencionados en el capítulo anterior, se comienza a modelar el sistema que será construido por los desarrolladores; para ello se identifican los requisitos funcionales y no funcionales, se representan mediante el diagrama de caso de uso, es decir la interacción entre el sistema y los actores, y finalmente se elaboran las descripciones textuales de cada caso de uso y los prototipos de interfaz de usuario. Para definir los requisitos se utilizó una estructura predefinida que el equipo de desarrollo fue cumplimentando usando para ello el lenguaje del usuario y el patrón CRUD de manera que se eliminen la ambigüedades y que cada requisito quede claramente definido como un caso de uso

El flujo de trabajo de Requerimientos, de la metodología RUP, tiene como propósitos generales los siguientes:

- Establecer y mantener el acuerdo con los clientes y demás interesados en la aplicación, en lo que el sistema debe hacer.
- Proporcionar a los desarrolladores del sistema una mejor comprensión de los requisitos de sistema.
- Definir las fronteras del sistema.
- Proporcionar una base para planificar el contenido técnico de las iteraciones.

Capítulo 2: Descripción de la Solución.

- Proporcionar una base para estimar el costo y el tiempo de desarrollo del sistema.
- Definir una interfaz para el usuario del sistema, enfocado en las necesidades y metas de los mismos.

2.5.1 Requisitos Funcionales (RF).

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir.

RF 1. Gestionar Elementos del Despliegue.

RF 1.1 Registrar elemento del despliegue.

RF 1.2 Modificar elemento del despliegue.

RF 1.3 Buscar elemento del despliegue.

RF 1.4 Visualizar elemento del despliegue.

RF 1.5 Subir documento Word "Manual de Instalación".

RF 1.6 Visualizar documento Word "Manual de Instalación".

RF 2. Administrar proceso de Pruebas de Aceptación.

RF 2.1 Insertar No Conformidad.

RF 2.2 Buscar No Conformidad.

RF 2.3 Visualizar No Conformidad.

RF 2.3 Visualizar estado de las pruebas.

RF 2.4 Generar Planilla de No Conformidades.

RF 2.4.1 Imprimir planilla.

RF 3. Administrar proceso de Pruebas de Piloto.

RF 3.1 Registrar en cuantos centros o instituciones se realizará el pilotaje.

RF 3.2 Insertar No Conformidad.

RF 3.3 Buscar No Conformidad.

RF 3.4 Visualizar No Conformidad.

RF 3.5 Visualizar estado de las pruebas.

RF 3.6 Generar Planilla de No Conformidades.

RF 3.6.1 Imprimir planilla.

RF 4 Administrar Control de recursos humanos y logística para el despliegue.

RF 4.1 Registrar personal que participará en el despliegue.

RF 4.2 Buscar personal del despliegue.

RF 4.3 Visualizar personal del despliegue.

RF 4.4 Modificar personal del despliegue.

RF 4.6 Registrar viajes de ida y regreso a los centro clientes.

Capítulo 2: Descripción de la Solución.

RF 5 Administrar Control del Sistema.

RF 5.1 Crear Usuario.

RF 5.2 Buscar Usuario.

RF 5.3 Visualizar Usuario.

RF 5.4 Modificar Datos de Usuario.

RF 5.5 Eliminar Usuario.

RF 6. Gestionar Proyecto a Desplegar.

RF 6.1 Registrar Proyecto a Desplegar.

RF 6.2 Modificar Registro del Proyecto.

RF 6.3 Buscar Proyecto Desplegado o en Fase de Despliegue.

RF 6.4 Visualizar Estado del Despliegue de un Proyecto.

Los requisitos del sistema fueron validados a través de la aplicación de encuestas a estudiantes y profesores que han trabajado directamente en el proceso de despliegue de productos de software en la Universidad, así como también a través de la realización de los prototipos de interfaz no funcionales que se mostrarán más adelante.

2.5.2 Requisitos No Funcionales (RNF).

Los requerimientos no funcionales presentados a continuación constituyen propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

RNF 1 Requerimientos de Software.

RNF 1.1 Se utilizará un Servidor Web Apache de soporte al servidor Tomcat.

RNF 1.2 Se utilizará un servidor de Base de Datos con software libre (PostgreSQL v8.3).

RNF 1.3 Para que el sistema corra es necesario computadoras donde el sistema operativo sea de la familia Windows 95, 98, NT, XP o Linux Debian o Ubuntu.

RNF 1.4 En las terminales clientes para la utilización del sistema es necesario disponer de cualquier navegador Web.

RNF 2 Requerimientos de Hardware.

RNF 2.1 Red de área local.

Capítulo 2: Descripción de la Solución.

RNF 2.2 Máquina con procesador Pentium IV o superior para el servidor y los clientes.

RNF 2.3 Se requiere de una impresora.

RNF 2.4 Memoria RAM 256 Mb en clientes y 512 Mb como mínima en servidores.

RNF 3 Requerimientos de Apariencia o interfaz externa.

RNF 3.1 Cada página no debe exceder 100 Kb en las imágenes.

RNF 3.2 Interfaz amigable, fácil de usar, sencilla, interactiva y debe mantener el mismo formato en todas las páginas.

RNF 3.3 Se utilizará java script y CSS 2.0 o superior como hoja de estilo en cascada.

RNF 4 Requerimientos de Seguridad.

RNF 4.1 Para la seguridad de la información que se maneja en el sistema se definirán grupos de usuarios, los cuales tendrán asignados permisos de acción sobre cada información manejada por el sistema, para lo cual se requiere la autenticación del usuario.

RNF 4.2 El sistema debe comunicarse usando un protocolo seguro (https), durante la autenticación del usuario, para evitar que las contraseñas viajen por la red en texto plano.

RNF 4.3 Confidencialidad: Cada grupo de usuarios accederá a la información correspondiente a cada uno.

RNF 4.4 Integridad: En la base de dato se garantizará la integridad mediante checks y triggers que se ejecutarán a la hora de efectuar las operaciones de inserción, actualización y eliminación de la información.

RNF 4.5 Disponibilidad: Se mantendrá el servidor Web funcionando las 24 horas y la aplicación implementará mecanismos para recuperarse ante los posibles fallos que puedan ocurrir.

RNF 5 Requerimientos de Usabilidad.

RNF 5.1 El sistema no será concebido sólo para usuarios de la Universidad, sino también para los Clientes de los productos que estemos desarrollando y desplegando en este caso, por lo que es necesario que cuente con un diseño de interfaz de fácil uso.

RNF 5.2 Los grupos de usuarios se diferenciarán en las opciones que el sitio les posibilite.

RNF 6 Requerimientos de Restricciones en el diseño y la implementación.

Capítulo 2: Descripción de la Solución.

RNF 6.1 El sistema se desarrollará en Java JDK v6.0 como lenguaje de programación.

RNF 6.2 La comunicación que habrá entre la base de datos y la aplicación, será por medio de módulos Java de acceso a datos.

RNF 6.3 El gestor de Base de datos a utilizar será PostgreSQL.

RNF 7 Requerimientos de Soporte.

RNF 7.1 El sistema llevará incluido un video que servirá de ayuda para los usuarios.

RNF 7.2 El sistema constará de una fase de prueba con los clientes donde podrán ser encontrados posibles errores que se puedan presentar.

RNF 8 Rendimiento.

RNF 8.1 La búsqueda de cualquier documento y su visualización en pantalla debe ser como mínimo de 2 segundos y no debe exceder los 10 segundos.

RNF 8.2 La aplicación debe estar concebida para el consumo mínimo de recursos.

2.5.3 Actor del Sistema.

Los actores del sistema son personas y/o otros sistemas externos que interactúan con el sistema, ven la funcionalidad del sistema y describen cómo será usado. Cada actor define un papel cohesivo y una clasificación independiente de los otros, puede participar en varios casos de uso y un caso de uso puede interactuar con varios actores.

Actor del Sistema	Descripción
Líder del Proyecto	Es la persona que está a cargo del proyecto desde su inicio hasta su fin, lidera, controla y gestiona toda la información y procesos del producto a desarrollar además realiza la función de administrador del sistema.
Implementador	Es el encargado de desarrollar todos los artefactos para la instalación del sistema.
Cliente	Es la persona que se beneficiará de los resultados del sistema.

Capítulo 2: Descripción de la Solución.

Personal de Calidad de software	Son las personas encargadas de realizar pruebas de calidad al sistema para consolidar el producto final, si luego de realizar todas las pruebas de calidad es aprobado, entonces el personal de calidad emite la liberación del producto en un acta de aprobación y firmada.
Administrador del Sistema	Es la persona que está al mando de todo lo correspondiente a la administración del sistema, desde el control de acceso hasta la gestión de los proyectos a desplegar.
Usuario	El Usuario esta compuesto por el Implementador, Personal de calidad y Cliente, los cuales interactúan a través de el con los casos de usos pertinentes.

Tabla 15: Actores del Sistema.

2.5.4 Casos de Uso del Sistema.

Los casos de uso constituyen la técnica básica y más aceptada de definición de requisitos, pues son el resultado del análisis de las necesidades de los usuarios. Permiten mostrar el contorno (actores) y el alcance (requisitos funcionales expresados como casos de uso) de un sistema. Los casos de uso describen la secuencia de interacciones que se producen entre el sistema y los actores del mismo para realizar una determinada función. A continuación se muestra el diagrama de Casos de Uso del sistema.

- Gestionar Elementos del Despliegue.
- Administrar proceso de Pruebas de Aceptación.
- Administrar proceso de Pruebas Piloto.
- Administrar Control de Recursos Humanos y Logística para el Despliegue.
- Administrar Control del Sistema.
- Gestionar Proyecto a Desplegar.

2.5.5 Diagrama de CUS.

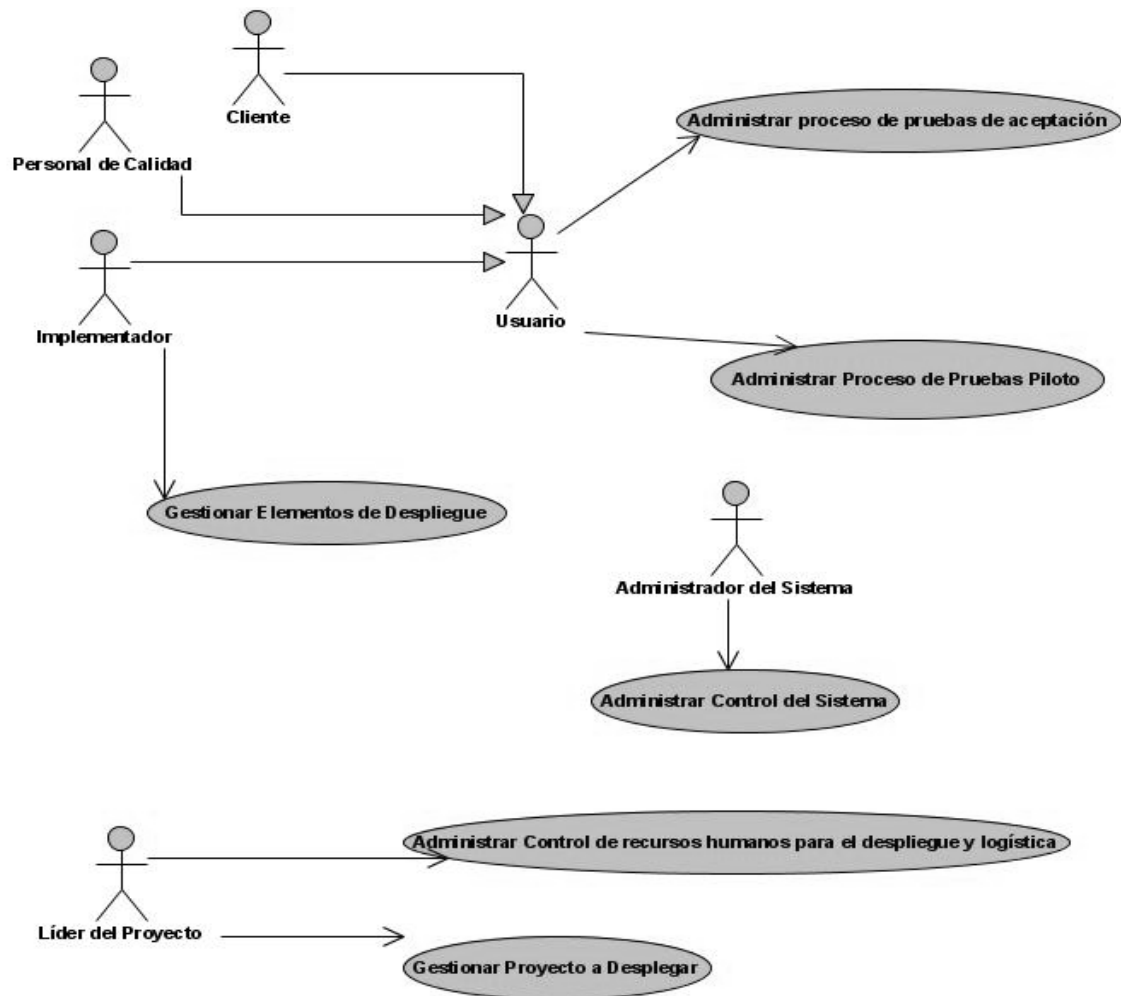


Ilustración 8: Diagrama de Casos de Uso del Sistema: despliegue de productos de software de la Facultad 4.

2.5.6 Descripciones textuales ampliadas de todos los CUS.

Tabla 16: Descripción del CUS Gestionar Elementos del Despliegue.

Nombre del Caso de Uso	Gestionar Elementos del Despliegue
Actores	Implementador(inicia)
Propósito	Crear Elementos del Despliegue, buscar y visualizar los elementos en el sistema, modificar datos de elementos creados para el despliegue.
Resumen	<p>El caso de uso se inicia cuando el Implementador va a realizar alguna de las siguiente operaciones relacionadas con los Elementos del Despliegue:</p> <ul style="list-style-type: none"> • Registrar Elementos del Despliegue. • Modificar Datos Elementos del Despliegue. • Buscar Elementos del Despliegue.

Capítulo 2: Descripción de la Solución.

	<ul style="list-style-type: none"> • Visualizar Elementos del Despliegue. • Subir documento Word “Manual de Instalación”. • Visualizar documento Word “Manual de Instalación”. <p>El sistema le muestra la interfaz correspondiente según su solicitud y ejecuta las acciones necesarias. El caso de uso finaliza cuando se emite el resultado de la operación solicitada.</p>
Precondiciones	Que el Implementador se haya autenticado en la aplicación.
Referencia	R1.1, R1.2, R1.3, R1.3, R1.4 , R1.5, R1.6
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
<p>1. El Implementador, quiere realizar una de las siguientes operaciones:</p> <ul style="list-style-type: none"> • Registrar Elementos del Despliegue. • Modificar datos Elementos de Despliegue. • Buscar Elementos del Despliegue. • Visualizar Elementos del Despliegue. • Subir documento Word “Manual de Instalación”. • Visualizar documento Word “Manual de Instalación”. 	<p>2. El sistema, en dependencia de la operación que solicita realizar, hace lo siguiente:</p> <ul style="list-style-type: none"> • Si decide Elementos del Despliegue, ir a Sección “Registrar Elementos del Despliegue”. • Si decide modificar datos de Elementos del Despliegue, ir a Sección “Modificar datos de Elementos de Despliegue”. • Si decide Buscar Elementos del Despliegue y visualizarlos, ir a Sección “Buscar Elementos del Despliegue”. • Si decide Subir y Visualizar documento Word “Manual de Instalación”, ir a Sección “Subir documento Word “Manual de Instalación”.
Sección “Registrar Elementos de Despliegue”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente para el registro de elementos del despliegue.
2. El Implementador provee los datos necesarios para registrar elementos del despliegue: Archivos del Sistema, dígame el o los Instaladores especificando el nombre del fichero y la versión los cuales se registrarán en la base de dato.	3. Verifica que los datos que introdujo fueron los necesarios para proceder a registrar elementos de despliegue: Archivos del Sistema

Capítulo 2: Descripción de la Solución.

	4. Registra elementos del despliegue y emite un mensaje de operación satisfactoria.
Flujos Alternos	
	3.1 Si falta algún dato necesario para la registro del elemento, emite un mensaje de error y no crea el elemento.
Sección “Modificar datos Elementos del Despliegue”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente para la modificación de los datos de elementos de despliegue.
	2. Ir a la sección “Buscar Elementos de Despliegue”.
	3. Localiza el elemento de despliegue y da la posibilidad de modificar los datos. Los campos que podrán ser modificados son: versión (Archivos del Sistema). El resto de los campos aparecerán deshabilitados, o sea, sin posibilidad de modificar los datos.
4. El Implementador actualiza los datos de elementos del despliegue.	5. Verifica que no se haya quedado ningún campo de información obligatoria vacío (los indicados en el registro).
	6. Registra la actualización en la base de datos y emite un mensaje de operación satisfactoria.
Flujos Alternos	
	3.1 Si los datos proveídos no son suficientes para la búsqueda o el elemento no existe en el sistema, emite un mensaje de error o no existencia del elemento.
	5.1 Si quedó algún campo de información necesaria vacío, emite un mensaje de error de completitud de datos.
Sección “Buscar Elementos de Despliegue”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente para la búsqueda de elementos del despliegue.
2. El Implementador provee los datos necesarios para localizar los elementos del despliegue que se quieren visualizar: <ul style="list-style-type: none"> • Nombre del archivo. 	3. Muestra el resultado de la búsqueda.
4. Selecciona el elemento del despliegue el cual quiere visualizar.	5. Visualiza el elemento del despliegue.

Capítulo 2: Descripción de la Solución.

	6. Da la posibilidad de buscar otro elemento.
Flujos Alternos	
	3.1 Si el resultado de la búsqueda es vacío, mostrar un mensaje "No está registrado el elemento de despliegue en la Base de Datos".
Sección "Subir Elemento de Despliegue"	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a subir documento Word "Manual de Instalación" como elemento del despliegue Instrucciones de Instalación.
2. El Implementador sube al sistema el documento Word "Manual de Instalación" como elemento del despliegue Instrucciones de Instalación.	3. El sistema carga el documento y muestra un mensaje de "Operación satisfactoria".
4. Selecciona el documento el cual quiere visualizar.	5. Visualiza el documento.
Flujos Alternos	
	2.1 Si al subir el documento ocurre algún fallo, mostrar un mensaje "Operación Fallo vuelva a intentarlo".
Requerimientos especiales	-
Prioridad	Crítico
Postcondiciones	El sistema registra Elementos del Despliegue correspondiente al software en despliegue en el sistema, busca y visualiza los elementos del despliegue almacenados, modifica los datos que sean necesarios de los elementos buscados, visualizados y creados.

Tabla 22: Descripción del CUS Administrar Proceso de Pruebas de Aceptación.

Nombre del Caso de Uso	Administrar Proceso de Pruebas de Aceptación
Actores	Usuario(Personal Calidad, Cliente, Implementador)
Propósito	Administrar el proceso de pruebas de Aceptación insertando No Conformidades, buscar y visualizar las No Conformidades, visualizar el estado de las pruebas, generar la Plantilla de No Conformidades e imprimirla, además generar el Acta de Aceptación del Producto e imprimirla.
Resumen	El caso de uso se inicia cuando el Personal de Calidad va a realizar alguna de las siguientes operaciones relacionadas con el proceso de pruebas de aceptación: <ul style="list-style-type: none"> • Insertar No Conformidad. • Buscar No Conformidad.

Capítulo 2: Descripción de la Solución.

	<ul style="list-style-type: none"> • Visualizar No Conformidad. • Visualizar estado de las pruebas. • Generar Reporte de No Conformidades. • Imprimir planilla. <p>El sistema le muestra la interfaz correspondiente según su solicitud y ejecuta las acciones necesarias. El caso de uso finaliza cuando se emite el resultado de la operación solicitada.</p>
Precondiciones	Que el Usuario se haya autenticado en la aplicación.
Referencia	R2.1, R2.2, R2.3, R2.4, R2.5, R2.5.1
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
<p>1. El Usuario, quiere realizar una de las siguientes operaciones:</p> <ul style="list-style-type: none"> • Insertar No Conformidad. • Buscar No Conformidad. • Visualizar No Conformidad. • Visualizar estado de las pruebas. • Generar Planilla de No Conformidades. • Imprimir planilla. 	<p>2. El sistema, en dependencia de la operación que solicita realizar, hace lo siguiente:</p> <ul style="list-style-type: none"> • Si decide Buscar No Conformidad, ir a Sección “Buscar No Conformidad”. • Si decide visualizar No Conformidad, ir a Sección “Visualizar No Conformidad”. • Si decide visualizar el estado de las pruebas, ir a la Sección “Visualizar estado de las pruebas”. • Si decide generar Reporte de No Conformidades e imprimirlo, ir a la Sección “Generar Planilla de No Conformidades”.
Sección “Insertar No Conformidad”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a insertar no conformidades.
<p>2. El Personal de Calidad selecciona el proyecto sobre el cual se están desarrollando las pruebas e introduce los parámetros para la inserción de la(s) no conformidades.</p> <ul style="list-style-type: none"> • Tipo (Alta , media y baja) • Descripción de la No Conformidad. • Fecha(se registra la fecha del momento que se inserta la no 	3. Inserta la No conformidad y muestra un mensaje de operación satisfactoria.

Capítulo 2: Descripción de la Solución.

conformidad) <ul style="list-style-type: none"> • Persona que la inserta. 	
Flujos Alternos	
	3.1 Si quedó algún campo de información necesaria vacío, emite un mensaje de error de completitud de datos.
Sección “Buscar No Conformidad”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a la búsqueda de no conformidades.
3. El Personal de Calidad selecciona el proyecto sobre el cual se están desarrollando las pruebas e introduce los parámetros por los cuales desea realizar la búsqueda de la(s) no conformidades. <ul style="list-style-type: none"> • Fecha (la fecha de la no conformidad estará en el rango de fecha de duración de las pruebas de aceptación) • Tipo (Alta , media y baja) 	
	3. Muestra el resultado de la búsqueda, según parámetros seleccionados anteriormente por el usuario.
4. El Personal de Calidad selecciona la no conformidad que desea visualizar.	5. Visualiza los datos de la no conformidad seleccionada.
	6. Da la posibilidad de realizar otra búsqueda.
Flujos Alternos	
2.1 Si solicita realizar la búsqueda y no seleccionó ningún parámetro, emite un mensaje de error: “Debe seleccionar al menos un parámetro para realizar la búsqueda”.	
4.1 Si no desea visualizar los datos de alguna de las no conformidades del resultado de la búsqueda, sale de la sección o puede realizar otra búsqueda.	
Sección “Visualizar estado de las pruebas”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a visualizar el estado de las pruebas.
2. El Usuario selecciona el proyecto del cual desea ver el estado de sus pruebas.	3. Visualiza el estado de la prueba seleccionada. <ul style="list-style-type: none"> • En Tiempo. • Atrasadas. • Actualmente realizándose.
Flujos Alternos	

Capítulo 2: Descripción de la Solución.

2.1 Si no desea visualizar el estado de las pruebas del resultado de la búsqueda, sale de la sección o puede realizar otra búsqueda.	
Sección “Generar Reporte de No Conformidad”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a la generación de la Reporte de No Conformidad.
2. El Personal de Calidad toma de la base de datos del reporte de No Conformidades: <ul style="list-style-type: none"> • Nombre del producto. • Fecha (fecha inicio y fecha fin) • Observaciones 	3. Muestra el reporte y guarda la fecha del reporte que se genera y da la posibilidad de imprimir dicho reporte.
4. El personal de Calidad indica imprimir el reporte.	
	5. Imprime el reporte.
	6. Da la posibilidad de generar otro reporte.
Flujos Alternos	
	2.1 Si quedó algún campo de información necesaria vacío, emite un mensaje de error de completitud de datos.
	4.1 Si no desea imprimir el reporte, sale de la sección o puede generar otro reporte.
Requerimientos especiales	-
Prioridad	Crítico
Postcondiciones	El sistema inserta las No Conformidades correspondientes al software en despliegue en el sistema, busca y visualiza las No Conformidades almacenadas, genera e imprime la Planilla de No Conformidades y el Acta de Aceptación del Producto.

2.6. Vista de CU.

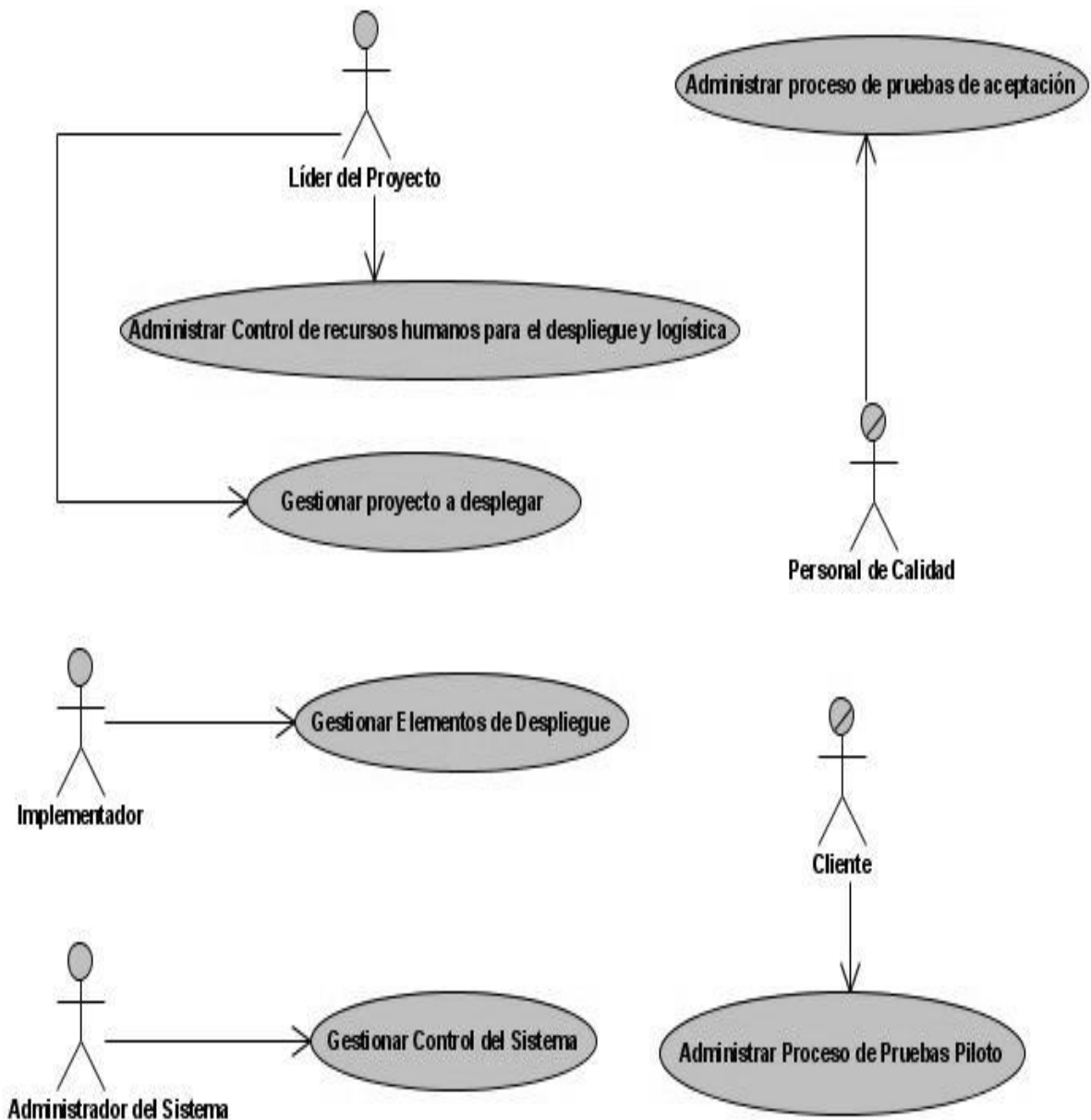


Ilustración 9: Diagrama de la Vista de los Casos de Uso del Sistema: despliegue de productos de software de la Facultad 4.

2.7. Conclusiones.

Con el uso de RUP quedó definido:

El proceso del negocio: la identificación de los actores, trabajadores y los casos de uso correspondientes.

El límite de las expectativas del cliente: características que se incluyeron en el producto y el trabajo realizado para obtenerlo.

Capítulo 2: Descripción de la Solución.

La comprensión clara de las necesidades de los usuarios y los cambios pertinentes permitieron definir los requisitos funcionales y no funcionales de la aplicación que se va a desarrollar, presentando los casos de uso y sus relaciones con los actores. La validación de los requisitos, demostró que los requerimientos definidos eran los óptimos y necesarios para el desarrollo de la aplicación.

Capítulo 3: Diseño de la Solución.

3.1 Introducción

El presente capítulo se encarga de definir cómo el sistema cumplirá con los requisitos establecidos, para así satisfacer las necesidades del cliente. El mismo está constituido por el diseño, el cual se encarga de definir como interactúan las clases y por la arquitectura llevará cabo dicho sistema.

3.2 Mapa de Navegación.



Ilustración 10: Mapa de Navegación: despliegue de productos de software de la Facultad 4.

3.3 Prototipo No Funcional de Interfaz de Usuario.



Ilustración 11: Prototipo No Funcional Interfaz Usuario Pantalla Autenticación al Sistema: despliegue de productos de software de la Facultad 4.

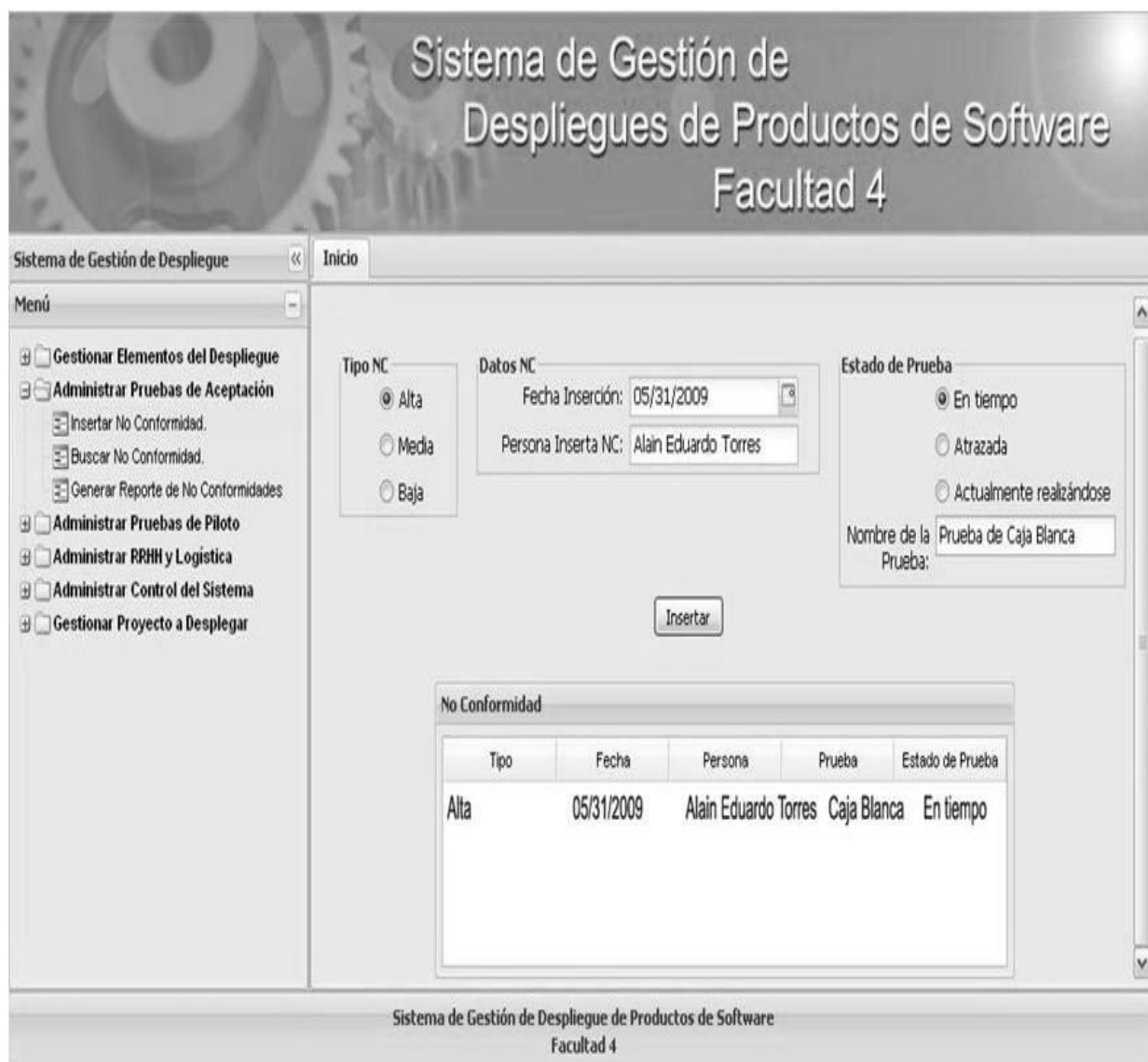


Ilustración 12: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Aceptación Escenario Insertar No Conformidad: despliegue de productos de software de la Facultad 4.

Ver **Anexo 4** para observar las demás pantallas del prototipo no funcional de interfaz de usuario.

3.4 Arquitectura del sistema.

La arquitectura definida para el desarrollo del sistema para la gestión de los proyectos productivos de la Facultad 4 es orientada al framework Spring para Java, está basada en una arquitectura de tres capas lógicas fundamentales: Capa de Presentación, Capa de Lógica de Negocio, y Capa de Acceso a Datos **Ilustración 13**. Dichas capas están bien delimitadas una de la otra, una capa superior interactúa con la inferior mediante interfaces que definen las funcionalidades que la misma debe brindar.

Capítulo 3: Diseño de la Solución.

En la arquitectura del sistema para la gestión de los proyectos productivos de la Facultad 4 se define una fachada que representa la interacción entre las Capas de Presentación y Lógica de negocio, la utilización de esta fachada cumple con el patrón de diseño Facade.



Ilustración 13: Representación básica de la arquitectura del sistema para despliegue de productos de software de la Facultad 4.

Capa de Presentación

En esta capa como su nombre lo indica es donde se maneja todo lo referente a la representación del sistema al usuario y de la interacción de este con dicho sistema, para esto se utilizó en esta capa el *Spring Web MVC* el mismo se basa fundamentalmente en separar el modelo de datos de la aplicación de su representación de cara al usuario y de la interacción de éste con la aplicación mediante la división en tres partes fundamentales: el modelo, que contiene la lógica de negocio de la aplicación; la vista, que muestra al usuario la información que éste necesita; el controlador, que recibe e interpreta la interacción del usuario, actuando sobre modelo y vista de manera adecuada para provocar cambios de estado en la representación interna de los datos, así como en su visualización. **Ilustración 14.** Es bastante flexible y altamente configurable, pero esa flexibilidad no le quita sencillez, ya que se pueden desarrollar aplicaciones sencillas sin tener que configurar muchas opciones.

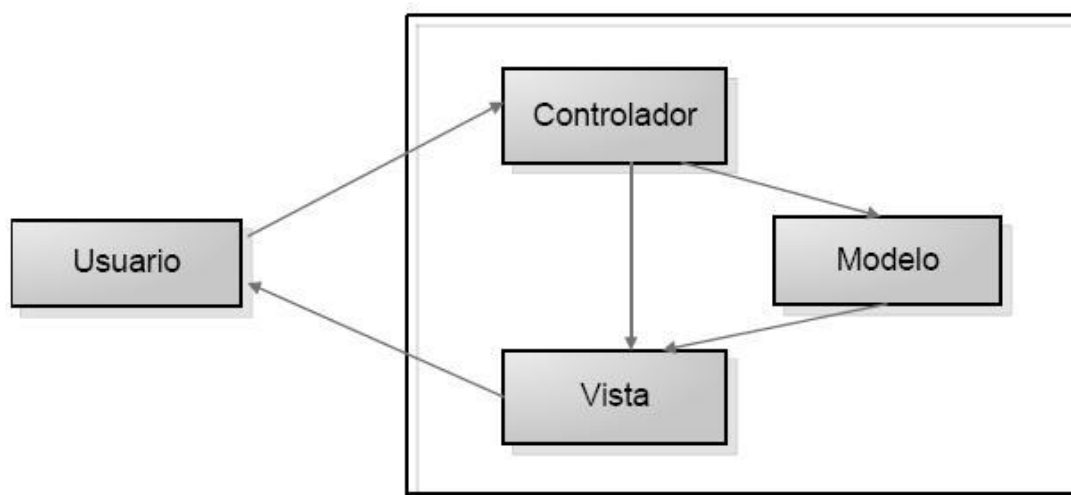


Ilustración 14: Modelo-Vista-Controlador Web de Spring: despliegue de productos de software de la Facultad 4.

En la **Ilustración 15**: se muestra la arquitectura básica de *Web MVC de Spring* la cual es utilizada en la capa de presentación para el desarrollo de la arquitectura del sistema para despliegues de productos de software de la Facultad 4.

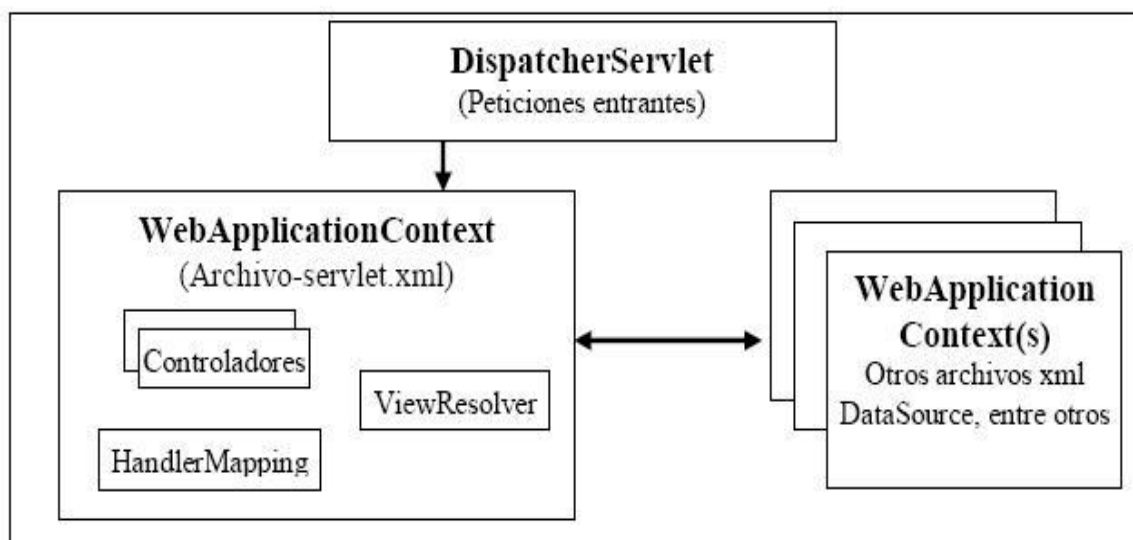


Ilustración 15: Arquitectura básica del Web MVC de Spring: despliegue de productos de software de la Facultad 4.

Los principales *Beans* a utilizar serán los *ViewResolver*, los controladores y el *HandlerMapping*.

El *HandlerMapping* es el encargado de manejar el mapeo para esto se utilizará el *BeanNameUrlHandlerMapping*, el cual mapea el Url hacia el controlador en base al nombre del bean del controlador.

Los *Views* o *Vistas* es un bean que transforma los resultados para que sean visibles al usuario y los pueda interpretar de una mejor forma es decir es quien implementa la

Capítulo 3: Diseño de la Solución.

interfaz, además se encarga de resolver el nombre lógico que regresa un controlador en un objeto *ModelAndView*, el que se utiliza es el *InternalResourceViewResolver* el cual resuelve los nombres lógicos en un archivo tipo View utilizando una plantilla de archivo jsp.

El controlador a utilizar es el controlador para formularios donde se rellena el archivo .jsp con *tags* que provee la librería *spring.tld*, luego se configura el *ApplicationContext*, para el control de la forma se crea un objeto que tenga atributos, cada uno de ellos con sus respectivos métodos *set* y *get*, para que a través del patrón de Inyección de Dependencia se pueda poner la información dentro de ese objeto y sea mas fácil manipular y almacenar, este objeto es el *command object*.

Se utilizará además el *Validator*, bean de referencia de la clase que extiende a la interfaz, el cual se encarga de después que se hizo la vinculación de validar que todos los campos cumplan con todos los requisitos y regresará los errores y las violaciones y en que campo fue donde se cometieron así como el mensaje de error que se debe desplegar

También se hará uso de los *ApplicationContext* es una subinterfaz del *BeanFactory* la cual realiza todo lo que la *BeanFactory* puede realizar es decir agrega información de la aplicación que puede ser utilizada por todos los componentes, además carga múltiples contextos, accesos a recursos etc. Y este nace del *Spring Context* que no es mas que un archivo de configuración que provee de información contextual al framework y el cual esta ligado al *Spring Core* quien es utilizado en esta arquitectura y esta compuesto por el *BeanFactory* y utiliza el patrón de Inversión de Control y configura los objetos a través de Inyección de Dependencia. El *Beanfactory* crea muchos tipos diferentes de beans los cuales pueden ser llamados por nombre y se encargan de manejar las relaciones entre objetos, además implementa la interfaz *BeanFactory* con instancias que pueden ser accedidas a través de esta interfaz y soportan objetos de dos modos diferentes *Singleton* y *Prototype*.

Capa de Servicios de Negocio

En esta capa radican los objetos de negocio o entidades del dominio y es donde se realiza todo el negocio de la aplicación.

Estas son las funcionalidades fundamentales que presenta esta capa:

- **Lógica de negocio específica de procesos de negocios:** A veces es más oportuno para los objetos de dominio contener lógica de negocio aplicable a muchos casos de uso específicos. Sin embargo, existen funcionalidades que

Capítulo 3: Diseño de la Solución.

son realizadas en la capa de servicios de negocios. Pero en esta definición de arquitectura los objetos de dominio no presentan ningún tipo de lógica de negocio, sino que esta responsabilidad recae sobre los objetos de negocio, permitiendo usar a los objetos de dominio como objetos de transferencia que se mueven entre las capas arquitectónicas de la aplicación.

- **Puntos de entrada muy bien definidos para las operaciones de negocio implementadas:** Los objetos de negocio brindan las interfaces usadas por la capa de presentación.
- **Control de transacciones:** Las políticas de transacciones de la aplicación serán planteadas sobre los objetos de negocio.
- **Ejecución de restricciones de seguridad:** Las restricciones de seguridad en esta capa se realizarán en los puntos de entradas a la capa media, es decir en los objetos de negocio.
- **Ejecución de la auditoría:** Sobre esta capa se llevará a cabo la auditoría sobre los métodos de negocio.

Por cada módulo se definirá una o más fachadas en caso de que se requiera que agrupen los métodos de negocio implementados en los manejadores o Managers que serán explicados más adelante. La fachada de un módulo se basa en el patrón Facade para permitir una clara división entre las capas arquitectónicas. Los Managers son las clases que se especializan en un conjunto de funcionalidades que representan el negocio sobre una o varias entidades. Los Managers son las únicas clases en la aplicación que tendrán lógica de negocio mientras que las fachadas se limitarán solamente a agrupar las funcionalidades para ser expuestas a capas superiores.

Capa de Acceso a Datos

La interacción del negocio con la capa de acceso a datos se realizará mediante el uso de interfaces. El término de Objeto de Acceso a Datos o en inglés, *Data Access Object* (DAO) es ampliamente usado en el desarrollo de software.

Los DAO encapsulan el manejo de acceso a datos de los objetos de dominios, proveen la persistencia de los objetos transitorios y las actualizaciones de los objetos existentes en la base de datos. Las implementaciones de los DAO estarán disponibles para los objetos (típicamente para los objetos de negocio) haciendo uso de la inyección de dependencias con los objetos de negocio y las instancias de los DAO, configurada en el contenedor de inversión de control de Spring Framework.

Capítulo 3: Diseño de la Solución.

Las interfaces de los DAO contienen básicamente los siguientes tipos de métodos:

- **Métodos para descubrir:** Estos localizan los objetos almacenados para ser usados por la capa de servicios de negocio.
- **Métodos para persistir o salvar:** Estos hacen persistentes a los objetos transitorios.
- **Métodos para eliminar:** Estos eliminan a los objetos guardados en el medio de almacenamiento (generalmente una base de datos).
- **Métodos para conteos y otras funciones agregadas:** Estos devuelven los resultados de operaciones que son más eficientes implementarlas usando funcionalidades de la base de datos que iterar sobre los mismos objetos.

3.4.1 Ambiente de desarrollo

El ambiente de desarrollo (Development Environment) es algo imprescindible en la producción de software. Aquí se definen el conjunto de herramientas y tecnologías (frameworks), versiones a usar y su integración, que intervienen en un proceso de desarrollo de software.

Ambiente de desarrollo integrado

Un ambiente de desarrollo integrado (Integrated Development Environment (IDE)) es un programa compuesto por un conjunto de herramientas para un programador. El mismo puede dedicarse a un solo lenguaje de programación o a varios.

Herramientas utilizadas en nuestro ambiente de desarrollo:

Tecnología JEE v 5.0

La tecnología Java Enterprise Edition (JEE) proporciona una completa y potente plataforma orientada al desarrollo de aplicaciones corporativas distribuidas y a los servicios web. La misma integra un conjunto de APIs, frameworks y patrones de programación que permiten responder de una forma robusta y flexible a todas las demandas de este tipo de aplicaciones.

Spring Framework v 2.0

Spring Framework es un framework de aplicación de código abierto que ayuda a hacer el desarrollo en JEE mucho más fácil. Ayuda a estructurar aplicaciones completas en una manera consistente y productiva para crear arquitecturas coherentes.

Presenta un diseño que brinda una gran flexibilidad arquitectónica e interviene en todas las capas de una aplicación.

Capítulo 3: Diseño de la Solución.

El mismo presenta varios módulos de los cuales los más importantes son:

- Contenedor de Inversión de Control.
- Un framework para la Programación Orientada a Aspectos (AOP).
- Una abstracción de acceso a datos.
- Simplificación de JDBC.
- Administración de transacciones.
- Framework Web MVC.
- Simplificación para trabajar con JNDI, JTA y otros APIs de JEE.
- Comunicación remota de peso ligero (Lightweight remoting).
- Soporte para Servicio de Mensajería de Java (Java Message Service, JMS).
- Soporte para Java Management Extensión (JMX).
- Soporte para comprensivas estrategias de pruebas para desarrolladores de aplicación.

Hibernate v 3.0

Hibernate es un potente framework de mapeo objeto/relacional y servicio de consultas para Java (y disponible también para .Net con el nombre de NHibernate). Permite diseñar objetos persistentes que podrán incluir polimorfismo, relaciones, colecciones, y un gran número de tipos de datos. Soporta diferentes entornos como: Oracle, DB2, MySQL, etcétera.

Spring web MVC 2.0

Flexible y altamente configurable, hace una clara visión entre controladores, modelo JavaBeans y vistas, es basado en interfaces, provee interceptores al igual que controladores, no obliga a utilizar jsp como única tecnología view también se puede utilizar otras.

3.5 Realización de CU-Diseño.

En este tipo de diagrama de clases es en el que se modelan las relaciones que pueden existir entre las páginas Servidoras, las páginas Cliente y los Formularios. Las relaciones pueden ser de distintos tipos, entre las cuales se pueden mencionar *build*, *submit* y *link*. La figura que se muestran a continuación representa el diagrama de diseño del cada caso de uso del sistema Administrar Proceso de Pruebas de

Capítulo 3: Diseño de la Solución.

Aceptación, para ver los restantes diagramas del diseño de los demás casos de uso del sistema ver Anexos.

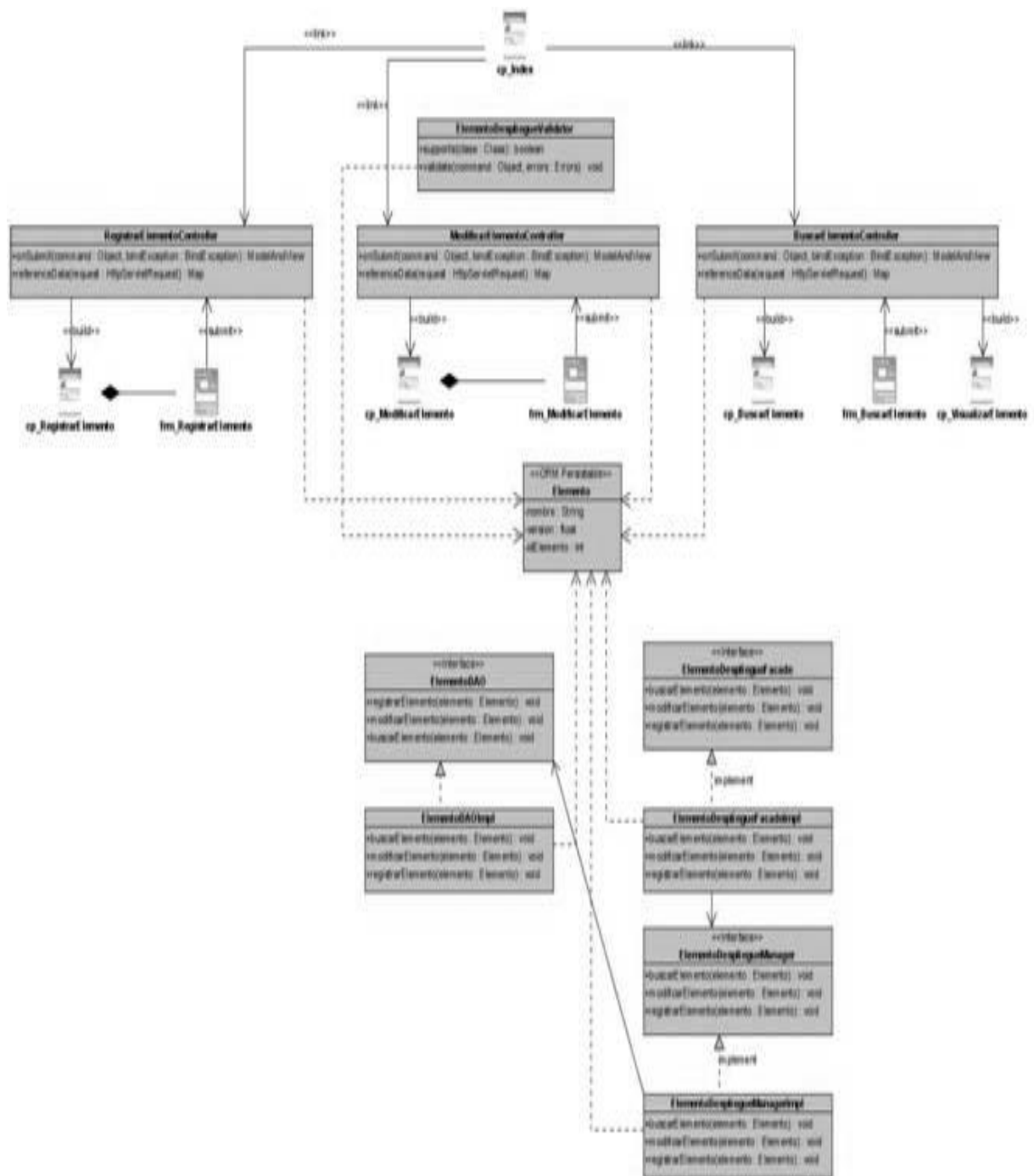


Ilustración 16: Diagrama de Clases del Diseño del Caso de Uso Gestionar Elementos del Despliegue: despliegue de productos de software de la Facultad 4.

Capítulo 3: Diseño de la Solución.

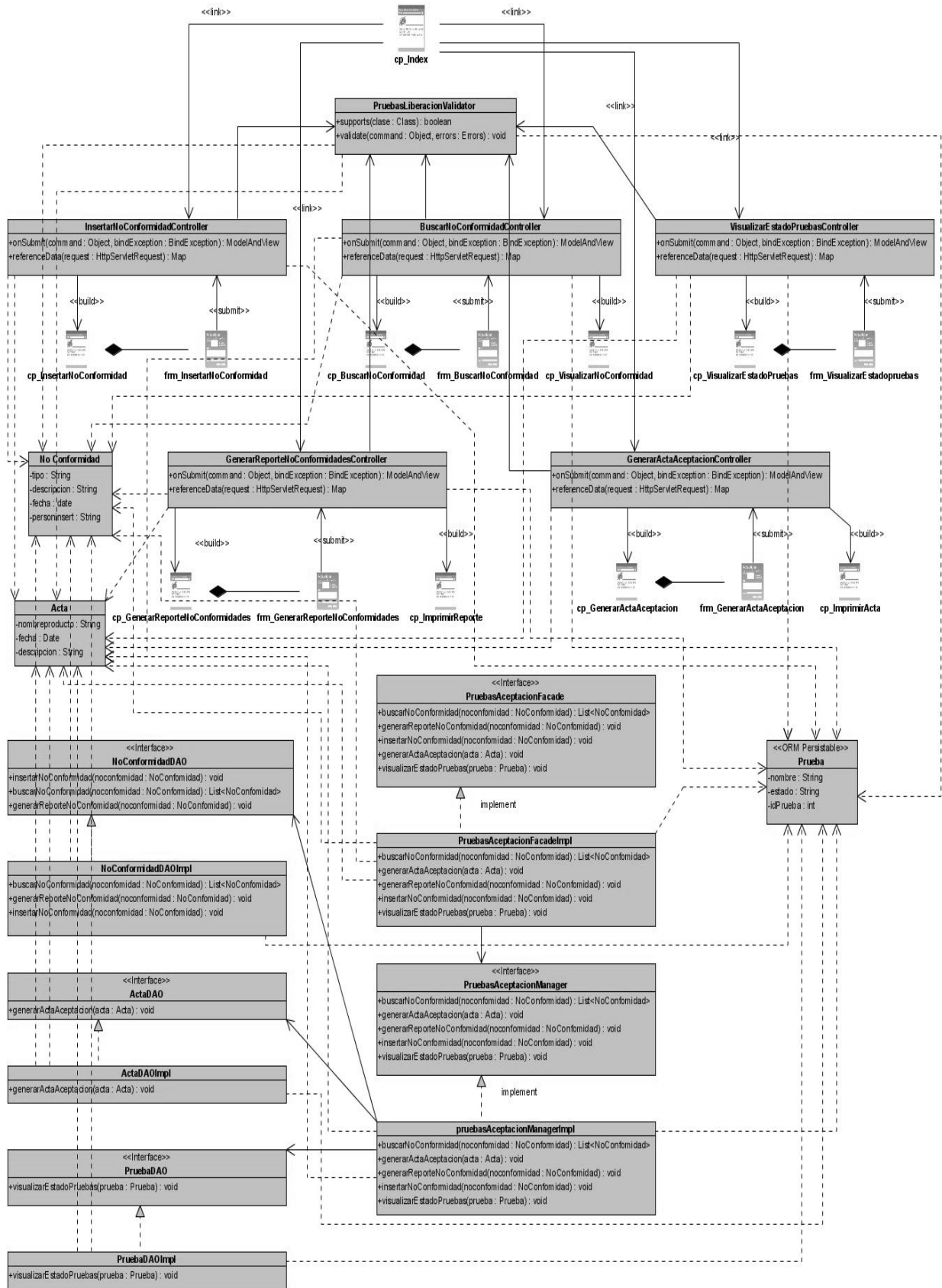


Ilustración 17: Diagrama de Clases del Diseño del Caso de Uso Administrar Proceso Pruebas de Aceptación: despliegue de productos de software de la Facultad 4.

3.5.1 Diagramas de Interacción.

Los diagramas de interacción muestran cómo se comunican los objetos en una interacción. Existen dos tipos de diagramas de interacción:

- Diagramas de Colaboración: Resaltan la organización de los objetos que participan en una interacción.
- Diagramas de Secuencia: Resaltan el orden temporal de los mensajes.

Para modelar los diagramas de interacción de los casos de uso para despliegues de productos de software de la Facultad 4, se escoge el tipo de diagrama secuencia debido a que los diagramas de secuencia son más adecuados para observar la perspectiva cronológica de las interacciones.

Las figuras que se muestran a continuación representan uno de los escenarios principales de los diagramas de secuencia de tres de los principales cada caso de uso del sistema, para ver los restantes diagramas de secuencia de los demás escenarios de estos casos de uso y los diagramas de secuencias de los demás casos de uso del sistema y sus escenarios ver **Anexos 6**.

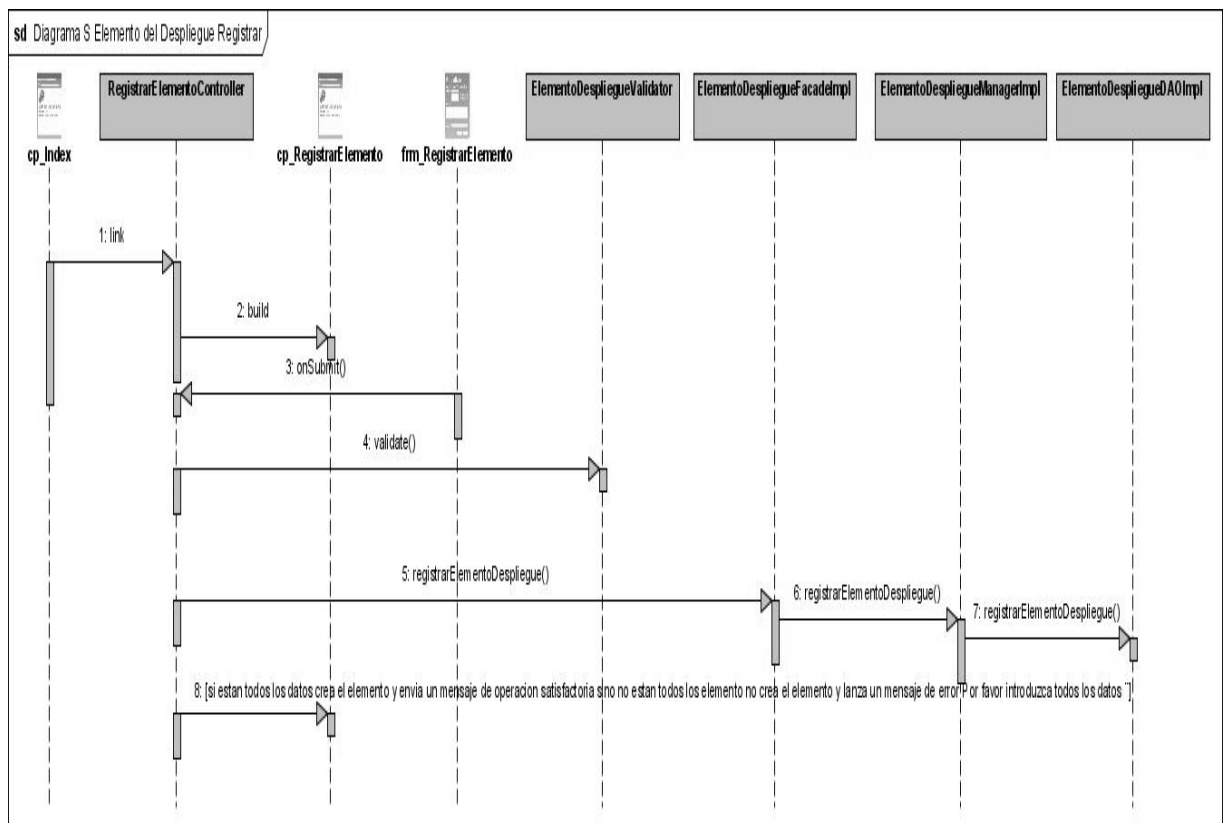


Ilustración 18: Diagrama de Secuencia del Caso de Uso Gestionar Elementos del Despliegue escenario Registrar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.

Capítulo 3: Diseño de la Solución.

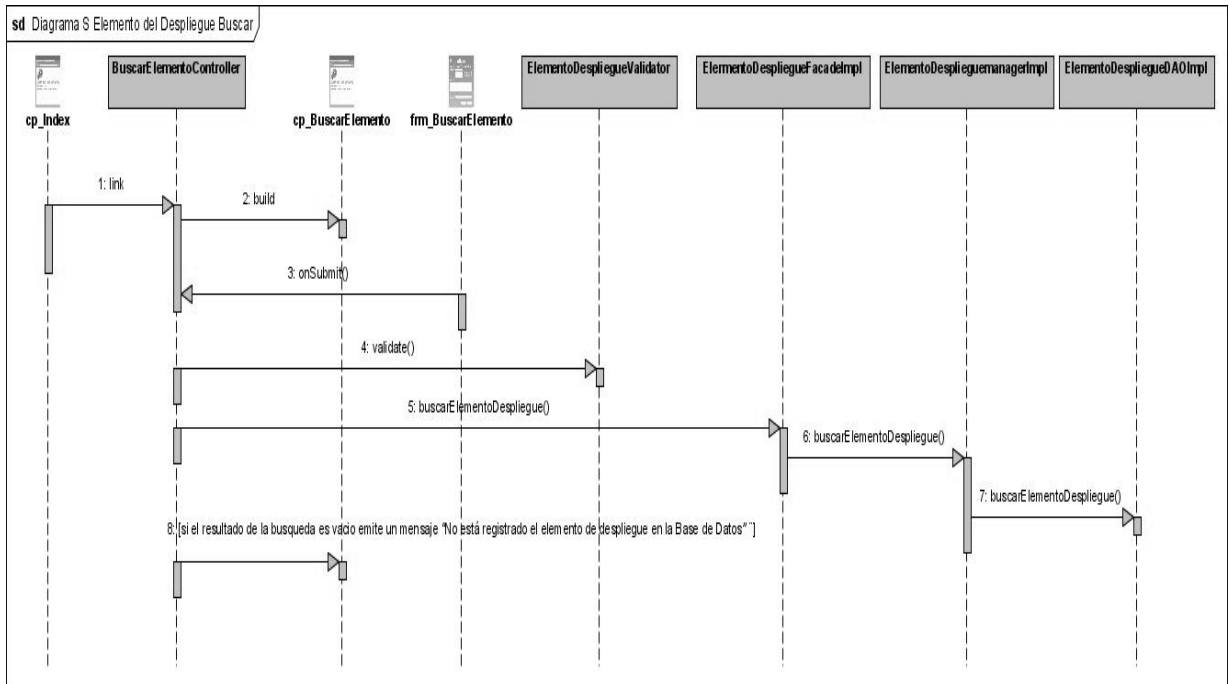


Ilustración 19: Diagrama de Secuencia del Caso de Uso Gestionar Elementos del Despliegue escenario Buscar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.

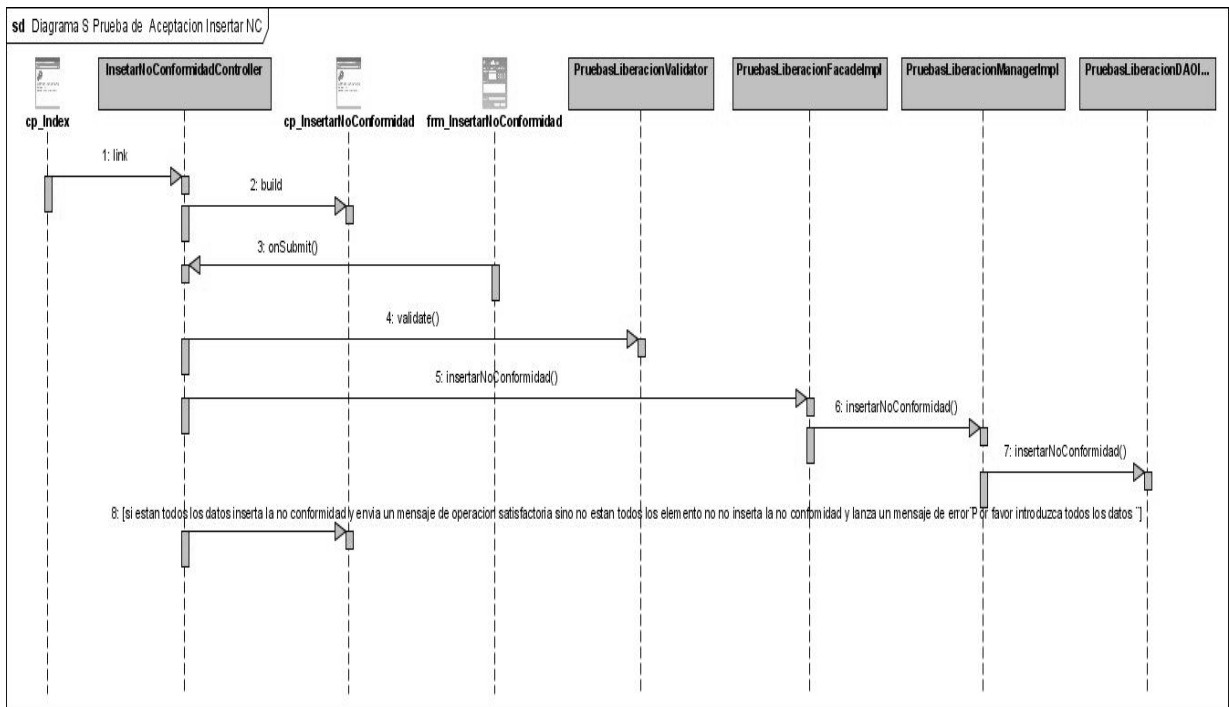


Ilustración 20: Diagrama de Secuencia del Caso de Uso Administrar Proceso Pruebas de Aceptación escenario Insertar No Conformidad: despliegue de productos de software de la Facultad 4.

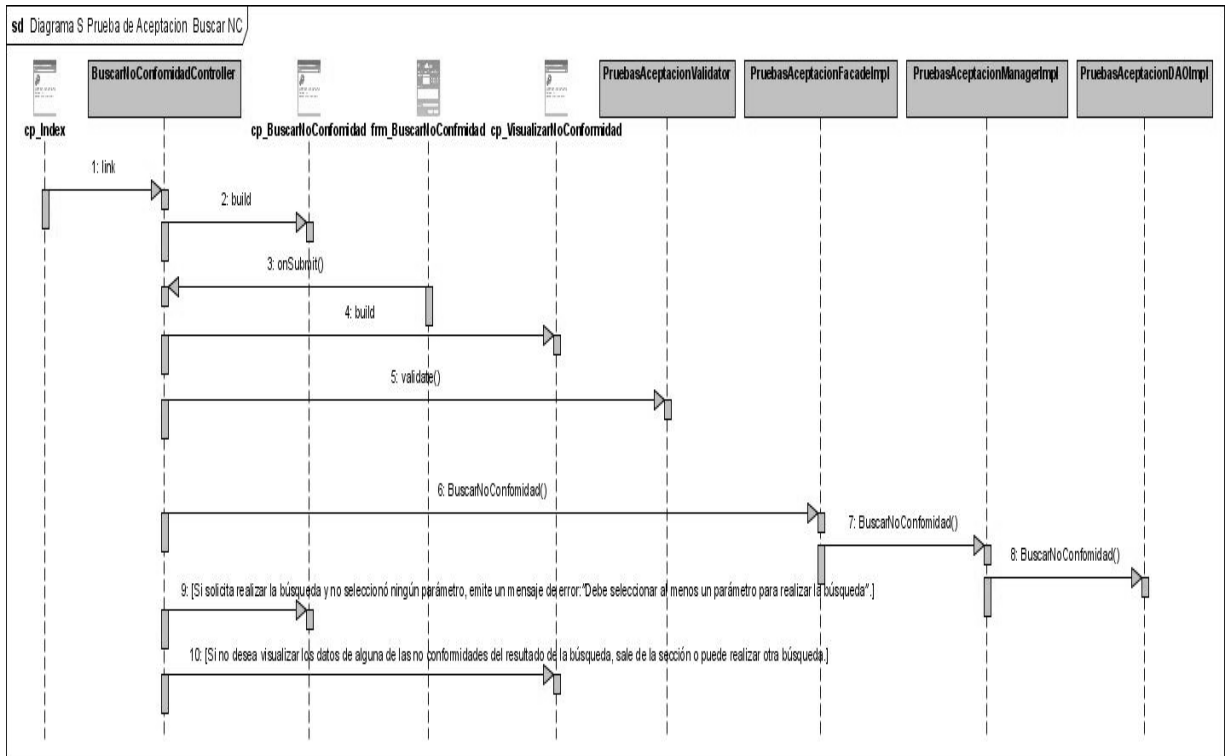


Ilustración 21: Diagrama de Secuencia del Caso de Uso Administrar Proceso Pruebas de Aceptación escenario Buscar No Conformidad: despliegue de productos de software de la Facultad 4.

3.6 Clases del Diseño con las descripciones de sus métodos.

A continuación se describe las clases principales del diseño de dos de los casos de uso del sistema.

Caso de Uso Gestión de Proyecto a Desplegar

Nombre de la Clase: ProyectoDAO	
Descripción: Contiene los datos de la clase proyecto en la base de datos, pertenece a la clase del diseño del caso de uso Gestión de Proyecto a Desplegar.	
Método	Descripción
registrarProyectoDesplegar(proyecto : Proyecto) : void	A través de este método se registra un proyecto a desplegar pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente el proyecto registrado.
modificarProyectoDesplegar(proyecto : Proyecto) : void	A través de este método se modifica un proyecto en despliegue pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente el proyecto modificado.
buscarProyectoDesplegar(proyecto :	A través de este método se busca un proyecto

Capítulo 3: Diseño de la Solución.

Proyecto) : void	en despliegue pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente la búsqueda.
------------------	---

Tabla 17: Descripción de los métodos de la clase del diseño ProyectoDAO: despliegue de productos de software de la Facultad 4.

Nombre de la Clase: ProyectoDesplegarFacade	
Descripción: La clase ProyectoDesplegarFacade es la aplicación del patrón estructural Facade que provee de una interfaz sencilla y unificada k actúa como intermediaria entre la ProyectoDesplegarManager, pertenece a la clase del diseño del caso de uso Gestión de Proyecto a Desplegar.	
Método	Descripción
registrarProyectoDesplegar(proyecto : Proyecto) : void	A través de este método se registra un proyecto a desplegar pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente el proyecto registrado.
modificarProyectoDesplegar(proyecto : Proyecto) : void	A través de este método se modifica un proyecto en despliegue pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente el proyecto modificado.
buscarProyectoDesplegar(proyecto : Proyecto) : void	A través de este método se busca un proyecto en despliegue pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente la búsqueda.

Tabla 18: Descripción de los métodos de la clase del diseño ProyectoDesplegarFacade: despliegue de productos de software de la Facultad 4.

Nombre de la Clase: ProyectoDesplegarManager	
Descripción: La clase ProyectoDesplegarManager es la que se encarga del negocio en la aplicación, pertenece a la clase del diseño del caso de uso Gestión de Proyecto a Desplegar.	
Método	Descripción
registrarProyectoDesplegar(proyecto : Proyecto) : void	A través de este método se registra un proyecto a desplegar pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente el proyecto registrado.
modificarProyectoDesplegar(proyecto	A través de este método se modifica un

Capítulo 3: Diseño de la Solución.

: Proyecto) : void	proyecto en despliegue pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente el proyecto modificado.
buscarProyectoDesplegar(proyecto : Proyecto) : void	A través de este método se busca un proyecto en despliegue pasándole como parámetro un proyecto de tipo proyecto devolviendo finalmente la búsqueda.

Tabla 19: Descripción de los métodos de la clase del diseño ProyectoDesplegarManager: despliegue de productos de software de la Facultad 4.

Caso de Uso Gestión de los Elementos del Despliegue.

Nombre de la Clase: ElementoDAO	
Descripción: Contiene los datos de la clase elemento en la base de datos, pertenece a la clase del diseño del caso de uso Gestión de Elementos del Despliegue.	
Método	Descripción
registrarElemento(elemento : Elemento) : void	A través de este método se registra un elemento del despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente el elemento registrado.
modificarElemento(elemento : Elemento) : void	A través de este método se modifica un elemento del despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente el elemento modificado.
buscarElemento(elemento : Elemento) : void	A través de este método se busca un elemento de despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente la búsqueda.

Tabla 20: Descripción de los métodos de la clase del diseño ElementoDAO: despliegue de productos de software de la Facultad 4.

Nombre de la Clase: ElementoDespliegueFacade	
Descripción: La clase ElementoDespliegueFacade es la aplicación del patrón estructural Facade que provee de una interfaz sencilla y unificada k actúa como intermediaria entre la ElementoDespliegueFacade, pertenece a la clase del diseño del caso de uso Gestión de Elementos del Despliegue.	
Método	Descripción
registrarElemento(elemento : Elemento) : void	A través de este método se registra un elemento

Capítulo 3: Diseño de la Solución.

Elemento) : void	del despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente el elemento registrado.
modificarElemento(elemento : Elemento) : void	A través de este método se modifica un elemento del despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente el elemento modificado.
buscarElemento(elemento : Elemento) : void	A través de este método se busca un elemento de despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente la búsqueda.

Tabla 21: Descripción de los métodos de la clase del diseño ElementoDespliegueFacade: despliegue de productos de software de la Facultad 4.

Nombre de la Clase: ElementoDespliegueManager	
Descripción: La clase ElementoDespliegueFacade es la que se encarga del negocio en la aplicación, pertenece a la clase del diseño del caso de uso Gestión de los Elementos del Despliegue.	
Método	Descripción
registrarElemento(elemento : Elemento) : void	A través de este método se registra un elemento del despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente el elemento registrado.
modificarElemento(elemento : Elemento) : void	A través de este método se modifica un elemento del despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente el elemento modificado.
buscarElemento(elemento : Elemento) : void	A través de este método se busca un elemento de despliegue pasándole como parámetro un elemento de tipo elemento devolviendo finalmente la búsqueda.

Tabla 22: Descripción de los métodos de la clase del diseño ElementoDespliegueManager: despliegue de productos de software de la Facultad 4.

3.7 Vista de Despliegue: diagrama de despliegue.

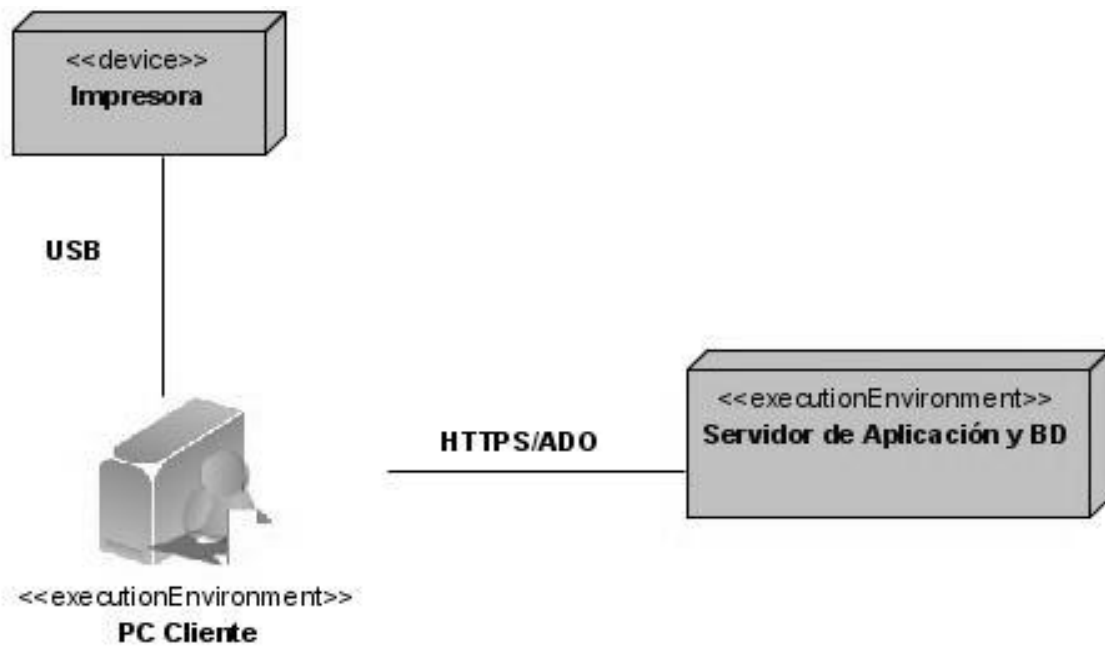


Ilustración 22: Diagrama de Despliegue: despliegue de productos de software de la Facultad 4.

3.8 Diseño de la Base de Datos.

El diseño de la base de datos es un aspecto importante, teniendo en cuenta que debe brindar la persistencia al modelo que se ha descrito con anterioridad en los epígrafes. A continuación se muestra el diagrama de clases persistentes, además se muestra también el modelo de datos que se usó.

3.8.1 Diseño de Clases Persistentes

Todas las clases que se identifican en el dominio del análisis no son persistentes. La persistencia es la capacidad de un objeto de mantener su valor en el espacio y en el tiempo. Es importante tener en cuenta que es responsabilidad del diseñador definir cuáles son las que deben ser clases persistentes.

La figura que se muestra a continuación presenta el Diagrama de Clases Persistentes.

Capítulo 3: Diseño de la Solución.

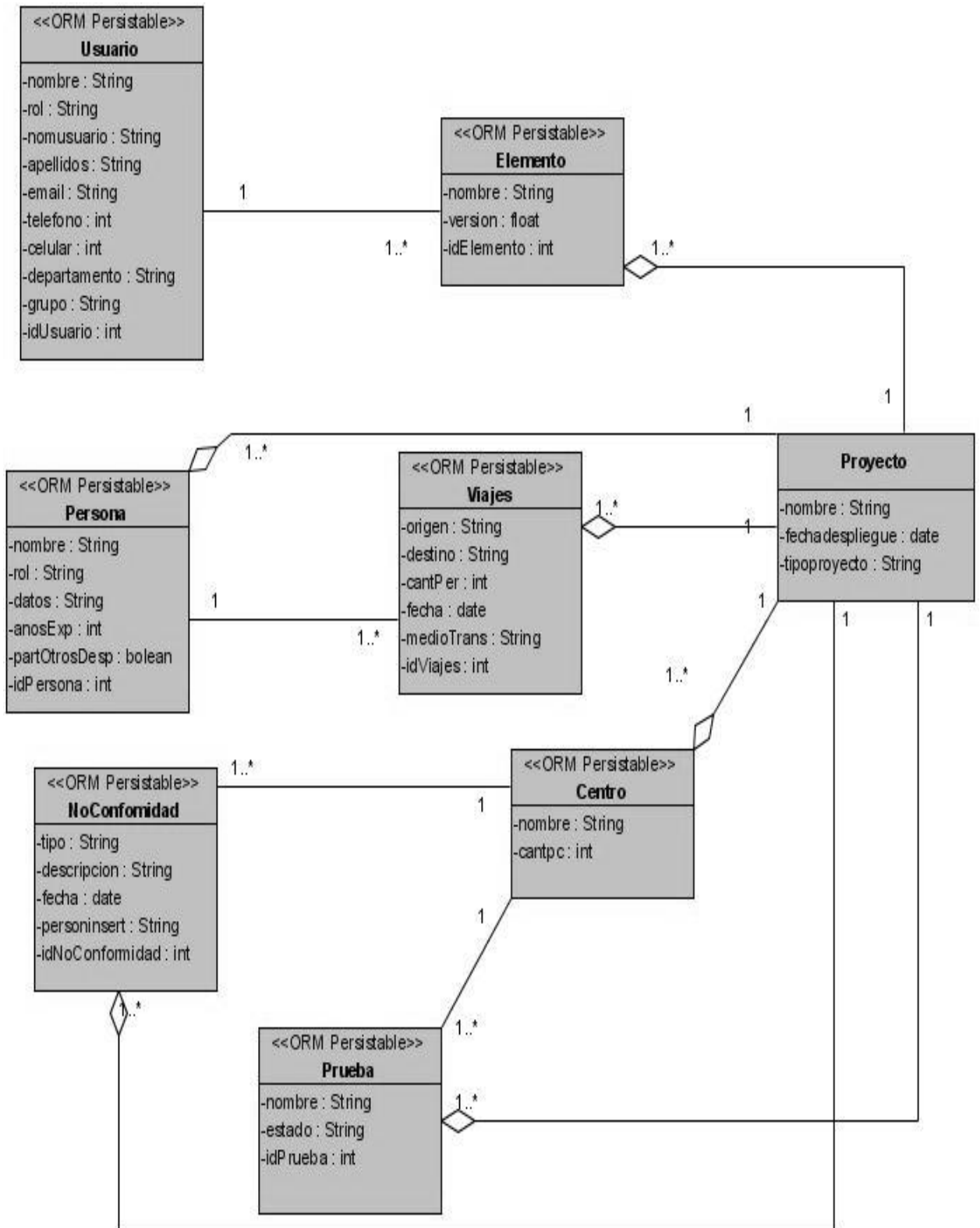


Ilustración 23: Diagrama de Clases Persistentes: despliegue de productos de software de la Facultad 4.

3.8.2 Modelo de Datos

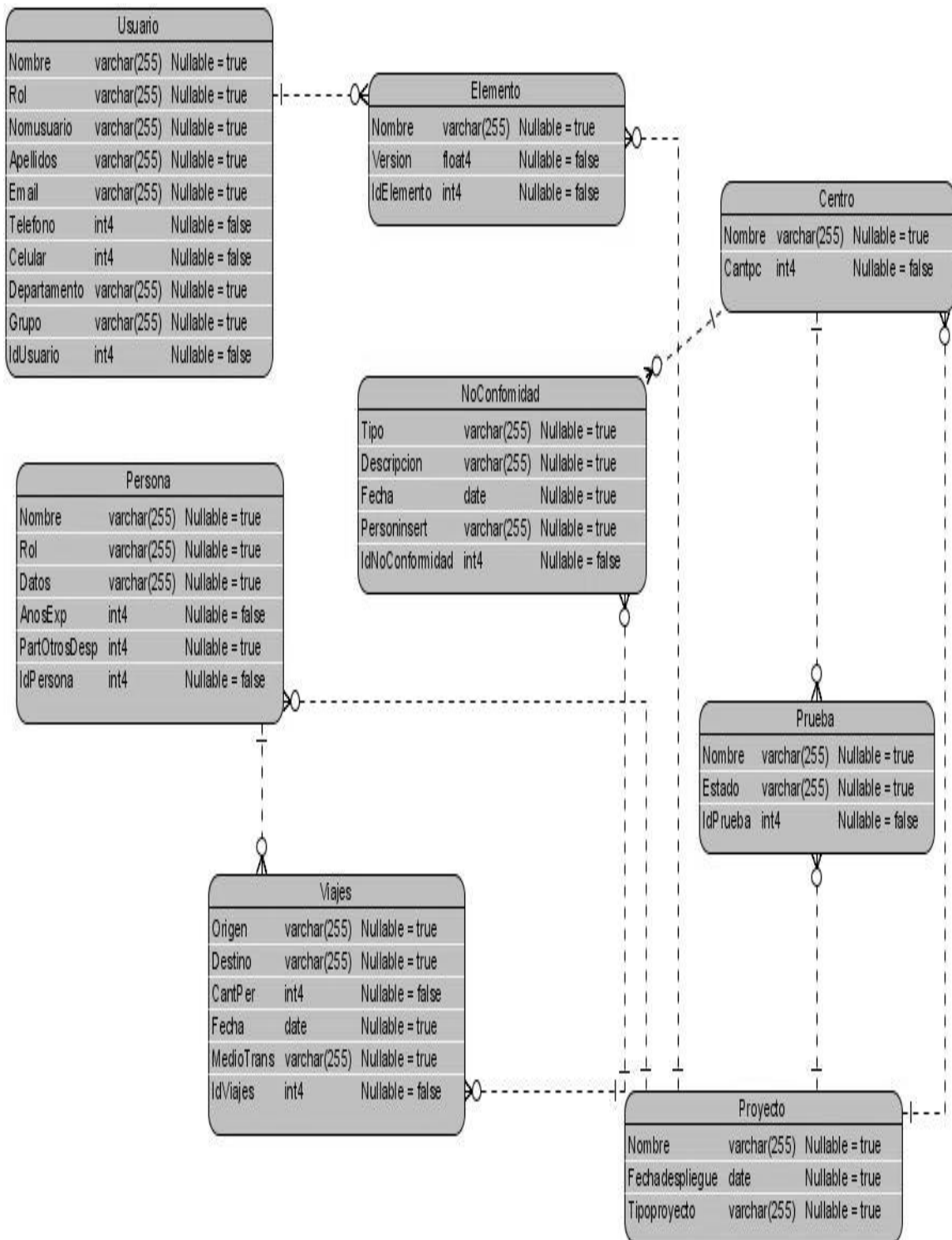


Ilustración 24: Modelo de Datos: despliegue de productos de software de la Facultad 4.

Capítulo 3: Diseño de la Solución.

3.8.3 Descripción de las tablas de la Base de Datos

Nombre: Usuario		
Descripción: Contiene todos los datos de los usuarios.		
Atributo	Tipo	Descripción
nombre	varchar	Contiene el nombre de la persona
rol	varchar	Rol del usuario
nombrequisario	varchar	Nombre del usuario
apellidos	varchar	Apellidos de la persona
email	varchar	Correo electrónico para contactar o localizar.
teléfono	int	Número de teléfono para contactar o localizar
celular	int	Número del celular para contactar o localizar
departamento	varchar	Indica el departamento al que pertenece la persona en caso ser profesor
grupo	varchar	Indica el grupo al que pertenece la persona en caso ser estudiante.
idusuario	int	Identificador de la tabla usuario

Tabla 23: Descripción de la tabla Usuario: despliegue de productos de software de la Facultad 4.

Nombre: Elemento		
Descripción: Contiene todos los datos de un elemento.		
Atributo	Tipo	Descripción
nombre	varchar	Contiene el nombre del elemento.
version	float	Contiene la versión del elemento.
idelemento	int	Identificador de la tabla elemento

Tabla 24: Descripción de la tabla Elemento: despliegue de productos de software de la Facultad 4.

Nombre: Persona		
Descripción: Contiene los datos de una persona.		
Atributo	Tipo	Descripción
nombre	varchar	Contiene el nombre de la persona
rol	varchar	Contiene el rol de la persona
datos	varchar	Contiene los datos de la persona
anosExp	int	Años de experiencia de la persona en despliegues
partOtrosDesp	int	Si ha participado en otros despliegues.

Capítulo 3: Diseño de la Solución.

idpersona	int	Identificador de la tabla persona
------------------	-----	-----------------------------------

Tabla 25: Descripción de la tabla Persona: despliegue de productos de software de la Facultad 4.

Nombre: No Confomidad		
Descripción: Contiene los datos de la no conformidad.		
Atributo	Tipo	Descripción
tipo	varchar	Contiene el tipo de no conformidad: Alta, Media, Baja
descripcion	varchar	Contiene la descripción de la no conformidad.
fecha	date	Contiene la fecha en que se inserta la no conformidad.
personinsert	varchar	Contiene la persona que inserta la no conformidad.
idNoConfomidad	int	Identificador de la tabla no conformidad

Tabla 26: Descripción de la tabla No Conformidad: despliegue de productos de software de la Facultad 4.

Nombre: Centro		
Descripción: Contiene los datos de un centro.		
Atributo	Tipo	Descripción
nombre	varchar	Contiene el nombre del centro o institución
cantpc	int	Contiene la cantidad de computadoras del centro
idcentro	Int	Identificador de la tabla centro

Tabla 27: Descripción de la tabla Centro: despliegue de productos de software de la Facultad 4.

Nombre: Prueba		
Descripción: Contiene los datos de las pruebas.		
Atributo	Tipo	Descripción
nombre	varchar	Contiene el nombre de la prueba a realizar
estado	varchar	Contiene el estado de las pruebas: En tiempo, Atrasadas, Actualmente realizándose.
idprueba	int	Identificador de la tabla prueba.

Tabla 28: Descripción de la tabla Prueba: despliegue de productos de software de la Facultad 4.

Capítulo 3: Diseño de la Solución.

Nombre: Proyecto		
Descripción: Contiene los datos de un proyecto.		
Atributo	Tipo	Descripción
nombre	varchar	Contiene el nombre del proyecto
fechadespliegue	date	Contiene la fecha de inicio y la fecha de fin del despliegue
tipoproyecto	varchar	Contiene el tipo de proyecto: Exportación o Nacional
idproyecto	int	Identificador de la tabla proyecto.

Tabla 29: Descripción de la tabla Usuario: despliegue de productos de software de la Facultad 4.

Nombre: Viajes		
Descripción: Contiene todos los datos de los viajes		
Atributo	Tipo	Descripción
origen	varchar	Contiene el origen del viaje
destino	varchar	Contiene el destino del viaje
cantper	int	Contiene la cantidad de personas en el viaje
fecha	date	Fecha que se realiza el viaje
mediotrans	varchar	Contiene el tipo de medio de transporte con que se realiza el viaje: Ómnibus o Microbús
idviaje	int	Identificador de la tabla viaje.

Tabla 30: Descripción de la tabla Viajes: despliegue de productos de software de la Facultad 4.

3.9. Conclusiones.

En este capítulo quedo definido el diseño del sistema y sus relaciones describiendo las mismas en términos de diagramas de clases. También se representó el diseño de la arquitectura usando patrones tanto de arquitectura como de diseño.

CONCLUSIONES

En el presente trabajo se Modelaron los procesos del negocio, capturaron, analizaron, especificaron y validaron los requerimientos del sistema para la gestión del despliegue de los proyectos de la Facultad 4 dando cumplimiento al objetivo principal del trabajo.

Se cumplió con el desarrollo claro de las tareas definidas, planteándose las conclusiones siguientes:

- En el presente trabajo se modelaron los procesos del negocio, capturaron, analizaron, especificaron y validaron los requerimientos del sistema para la gestión del despliegue de los proyectos de la Facultad 4 dando cumplimiento al objetivo principal del trabajo.
- La informatización del proceso de gestión del despliegue de los proyectos productivos de la Facultad 4 constituye una necesidad para el funcionamiento eficiente de los proyectos productivos que en la Facultad 4 se desarrollan.
- Los sistemas similares existentes analizados no se ajustan a las necesidades que han surgido en la Facultad 4 de la Universidad de las Ciencias Informáticas (UCI) por lo que se decidió desarrollar un nuevo sistema, que recoge lo mejor de los sistemas analizados.
- La metodología, las herramientas y los patrones utilizados permitieron que el trabajo se desarrollara con mayor eficiencia.
- Con los requisitos funcionales quedaron claras todas las funcionalidades que debe soportar el sistema, para dar cumplimiento a las necesidades y expectativas del cliente.
- La definición de las Reglas del Negocio como precondiciones a cumplir en el modelado del sistema se garantizó el cumplimiento las políticas que rigen el proceso de gestión del despliegue.
- La descripción de los casos de usos constituyen una mejor manera de comprender las funcionalidades del sistema.
- La definición de la arquitectura así como la realización de las clases del diseño candidatas a la implementación y la identificación de las clases persistentes brindaron una mayor consistencia del sistema.

Conclusiones.

- Durante el desarrollo del trabajo todos los artefactos generados permitieron un mejor entendimiento para el desarrollo del trabajo, dando cumplimiento así al objetivo propuesto.

RECOMENDACIONES

Los objetivos generales de este trabajo han sido logrados, pero a lo largo de su desarrollo, han ido surgiendo ideas que podrían considerarse en un futuro, para lo cual se recomienda:

- Implementar la aplicación.
- Refinar el modelado del negocio a partir de nuevos procesos que se manejen en la Facultad 4 de la UCI y automatizar dichos procesos.
- Aplicar la solución propuesta para el resto de las facultades de la UCI.
- Hacer de esta investigación un material de consulta del personal de los proyectos productivos de las demás Facultades.
- El estudio realizado ha confirmado que es verdaderamente necesario realizar un análisis profundo de la gestión del despliegue de los proyectos productivos de la universidad de modo que permita entender su funcionamiento para lograr una automatización mas avanzada.

REFERENCIAS BIBLIOGRAFICAS

- ANGEL A. M. PANIAGUA, A. R. T. Herramientas CASE 2005. [Disponible en: <http://www.monografias.com/trabajos24/herramientas-case/herramientas-case.shtml>]
- ANDRES, K. B. Y. C. Extreme Programming Explained: Embrace Change (Paperback). 2da edición 2004. P
- Apache-Tomcat,2009 [Disponible en: <http://es.wikipedia.org/wiki/Apache-Tomcat>]
- CALABRIA, L. Metodología FDD, 2003. [Disponible en: <http://www.javahispano.org/articles.article.action?id=76>]
- Carlos D. González, 2009. [Disponible en: <http://www.usabilidadweb.com.ar/postgre.php>]
- Danysoft. AdminStudio 2009. [Disponible en: <http://www.danysoft.com/bol/adminstudio.htm>]
- ESCRIBANO, G. F. Introducción a Extreme Programming, 2002.
- Fair Isaac Corporation. Blaze Advisor 6.6 2009. [Disponible en: http://www.financialtech-mag.com/000_estructura/index.php?id=24&idb=169&ntt=10775&sec=12&vn=1]
- KASEYA, A. S. G. Kaseya Enterprise Edition 2009. [Disponible en: <http://www.kaseya.com.mx/products/remote-software-deployment.php>]
- GUNNAR ÖVERGAARD, K. P. Use Cases Patterns and Blueprints, Addison Wesley Professional, 2004.
- IVAR JACOBSON, G. B., JAMES RUMBAUGH. El Proceso Unificado de Desarrollo de Software. . Madrid, 2000. 16 p.
- IVAR JACOBSON, G. B., JAMES RUMBAUGH. El Lenguaje Unificado de Modelado. UML 1999. p.
- JUSTO MENDEZ. LENGUAJE DE PROGRAMACION 2009. [Disponible en: <http://www.monografias.com/trabajos/lengprog/lengprog.shtml>]

Referencias Bibliográficas.

- Jesús María Ramos Saky, Patrones de Diseño, Apuntes MDS Curso 2002/2003. [Disponible en: <http://webs.teleprogramadores.com/patrones/http://home.earthlink.net/~huston2/dp/patterns.html>]
- LARMAN, C. UML Y Patrones. Introducción al análisis y diseño orientado a objeto. México 1999. p.
- IBM Rational Software Corporation, Rational Unified Process Help 1987, 2006.
- MICROSOFT. Microsoft Solutions Framework 1.0. Microsoft Corporation. USA. , 1997. [Disponible en: <http://www.rational.com/uml>]
- ARGOUML, 2009. [Disponible en: <http://es.wikipedia.org/wiki/ArgoUML>]
- VISUAL, PARADIGM, 2009. [Disponible en: <http://www.visual-paradigm.com/product/vpum/>]
- MySQL,2009 [Disponible en: <http://es.wikipedia.org/wiki/MySQL>]
- PostgreSQL, 2007. [Disponible en:<http://www.http-peru.com/postgresql.php>]
- PHP. 2009 [Disponible en: <http://es.wikipedia.org/wiki/.php>]
- Python,2009 [Disponible en: <http://es.wikipedia.org/wiki/Python>]
- PRESSMAN, R. S. Ingeniería de Software. Un enfoque Práctico, 2005. 560.
- SANCHEZ, M. A. M. Metodologías De Desarrollo De Software rup, xp, msf, 2004. [Disponible en: <http://www.informatizate.net>]
- SOFTWARE, I. R. rational Software, 2007. [Disponible en: www.rational.com]
- D. Delgado and E. Domínguez, "Propuesta de Arquitectura para JIMMSIM 2.0" Facultad 6.Universidad de las Ciencias Informáticas, Cuba, 2008]

BIBLIOGRAFÍA

- GUNNAR ÖVERGAARD, K. P. Use Cases Patterns and Blueprints, Addison Wesley Professional, 2004.
- IVAR JACOBSON, G. B., JAMES RUMBAUGH. . El Proceso Unificado de Desarrollo de Software. . Madrid, 2000. 16 p.
- IVAR JACOBSON, G. B., JAMES RUMBAUGH. El Lenguaje Unificado de Modelado. UML 1999. p.
- LARMAN, C. UML Y Patrones. Introducción al análisis y diseño orientado a objeto. México 1999. p.
- LARMAN, C. UML Y Patrones. Introducción al análisis y diseño orientado a objeto. México 1999. p.
- MICROSOFT. Microsoft Solutions Framework 1.0. Microsoft Corporation. USA. , 1997. [Disponible en: <http://www.rational.com/uml>]
- JOHNSON, R. Professional Java Development with the Spring Framework. 2005. p. 0764574833
- PIMENTEL, L. A. and J. E. MARTÍNEZ. Documento de Arquitectura de Software. Cuba, 2008. p.
- PIMENTEL, L. A. and I. P. RIVERO. ArBaWeb: ARQUITECTURA BASE SOBRE LA WEB. La Habana, Universidad de las Ciencias Informáticas, 2007. p.

GLOSARIO DE TÉRMINOS

- CASE: Computer Aided Software Engineering (Ingeniería de Software Asistida por Computación).
- IDE: Integrated Development Environment (Ambiente de desarrollo integrado).
- Interbase: Es un Sistema de Administración de Base de Datos Relacionales (RDBMS).
- Navegador: Programa utilizado para ubicar y ver páginas Web. Por ejemplo Netscape,
- Mosaic, Microsoft Internet Explorer, FireFox, Opera y otros.
- PHP: Es un acrónimo recursivo que significa "Hypertext Pre-processor". Es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas Web dinámicas.
- Release: Versión de un sistema de software que se libera.
- RUP: Rational Unified Process (Proceso Unificado de Desarrollo de Software).
- Script: Segmento de instrucciones que realizan operaciones sencillas dentro del código de una aplicación.
- TI: Tecnologías de la Investigación (TI).
- UCI: Universidad de las Ciencias Informáticas.
- UML: Lenguaje Unificado de Modelado.
- Web browser: Navegador.

ANEXOS

Anexo 1. Descripciones Textuales de los Casos de Uso del Negocio

Tabla 31: Descripción del CUN Administrar proceso de pruebas piloto.

Caso de Uso del Negocio	Administrar proceso de pruebas de piloto	
Actores	Personal de Calidad (inicia), cliente.	
Trabajadores	Líder de proyecto, Administrador de Despliegue, Implementador.	
Propósito	El propósito de este caso de uso es realizar las pruebas pilotos al producto desplegado en los lugares escogidos por el Cliente previo acuerdo con el Líder del Proyecto y aprobado por la Vicedecana de Producción de la Facultad.	
Resumen: El caso de uso se inicia cuando el Personal de Calidad comienza las pruebas pilotos y a su vez administra las mismas, el Cliente es quien lleva a cabo las pruebas pilotos del producto en un ambiente similar al ambiente donde va a terminar funcionando finalmente el producto, el caso de uso termina cuando es probado el producto durante el pilotaje por el Cliente.		
Curso Normal de los Eventos		
Acción del actor	Respuesta del proceso de negocio	
1- El Personal de Calidad procede a administrar el proceso de pruebas piloto.		
2- El Cliente ejecuta las pruebas de aceptación. Si detecta alguna No Conformidad se pone en la Planilla de No Conformidades y al concluir las pruebas se le envía al Implementador dicha planilla.		
	3- El Implementador revisa la Planilla de No Conformidades y procede a resolver las No Conformidades	
5- El Personal de Calidad elabora el Acta de Aceptación del Pilotaje del Producto.		
6- El Cliente firma el Acta de Aceptación del Producto		
	6- El Administrador de Despliegue recoge el Acta de Aceptación del Producto y se continúa con el proceso del despliegue.	
Prioridad	Opcional	
Flujos Alternos		
Acción del actor	Respuesta del proceso de negocio	
	2.1 Si el cliente no encuentra alguna No Conformidad entonces finalizan las pruebas se elabora el acta y se	

Anexos.

	aprueba y se continua con el proceso de despliegue.
Mejoras	Se informatizará en la administración del proceso de pruebas pilotos, las No Conformidades, el estado de las pruebas.

Tabla 32: Descripción del CUN Realizar Liberación del Producto.

Caso de Uso del Negocio	Realizar Liberación del Producto	
Actores	Personal de Calidad	
Trabajador	Líder del proyecto, Administrador de Despliegue.	
Propósito	El propósito de la liberación del producto es realizar las pruebas de liberación del producto para que se emita el Acta de Liberación que permitirá entrarlo en Pruebas de Aceptación.	
Resumen: El caso uso se inicia cuando el Personal de Calidad realiza pruebas de calidad con el objetivo de liberar el producto para ser aceptado y piloteado por el cliente, el caso de uso termina cuando luego de terminar las pruebas de calidad es liberado el producto emitiendo el Acta de Liberación.		
Curso Normal de los Eventos		
	Acción del actor	Respuesta del proceso de negocio
	1-El personal de calidad inicia las pruebas de calidad de software: Pruebas de Caja Blanca y Caja Negra. 2- Si encuentra algún error en la realización de las pruebas es informado al personal de desarrollo del software dígame: Líder del proyecto, Administrador de Despliegue, Implementador etc.	
		3- El Líder del Proyecto recibe el informe de errores durante la realización de las pruebas.
	4- El personal de calidad concluye las pruebas de software. 5- El Personal de Calidad emite el Acta de Liberación del Producto.	
		7- El Líder del Proyecto recibe el Acta de Liberación del Producto firmada por el personal de Calidad.
Prioridad		Crítico
Flujos Alternos		
	Acción del actor	Respuesta del proceso de negocio

Anexos.

	4.1 Si se hay alguna No Conformidad, no se libera el producto.
Mejoras	

Tabla 33: Descripción del CUN Brindar Soporte al Sistema.

Caso de Uso del Negocio	Brindar Soporte al Sistema	
Actores	Cliente (inicia)	
Trabajador	Implementador, Escritor Técnico.	
Propósito	El propósito de este caso de uso es responder a las solicitudes de soporte realizadas por los clientes de los sistemas desplegados por la Facultad.	
Resumen: El caso uso se inicia cuando el Cliente solicita que se le brinde soporte al Sistema desplegado y finaliza cuando finaliza el soporte al sistema.		
Curso Normal de los Eventos		
Acción del actor	Respuesta del proceso de negocio	
1.- El Cliente solicita al escritor técnico el soporte del sistema.	2-El Escritor Técnico en conjunto con el Implementador, preparan el soporte al sistema el cual es: Actualización de la Base de Dato, Actualización de nueva versión del sistema. 3-El escritor técnico y el implementador, entregan finalmente realizado el soporte del sistema.	
4-El cliente recoge el soporte del sistema.		
	5- El escritor técnico recibe la No conformidad en algún elemento del Soporte al Sistema y realiza el cambio pertinente.	
6-El cliente emite la aceptación del soporte al sistema.		
	6- El escritor técnico recibe la aceptación del cliente del soporte del sistema.	
Prioridad	Crítico	
Flujos Alternos		
Acción del actor	Respuesta del proceso de negocio	
	5.1 Si encuentra algún elemento con el cual no esté conforme, se lo envía al escritor técnico para ser cambiado.	
Mejoras		

Anexos.

Tabla 34: Descripción del CUN Realizar Transferencia Tecnológica.

Caso de Uso del Negocio	Realizar Transferencia Tecnológica	
Actores	Cliente	
Trabajador	Desarrollador de Curso.	
Propósito	El propósito del caso de uso es brindar todos los materiales de enseñanza para la utilización del producto del software.	
<p>Resumen: El caso uso se inicia cuando el desarrollador de curso se prepara para realizar la entrega al cliente de toda la documentación de aprendizaje del sistema dígame manuales, guías, libros, cursos y finaliza cuando el cliente se encuentra en condiciones para manipular el sistema.</p>		
Curso Normal de los Eventos		
Acción del actor	Respuesta del proceso de negocio	
	1-El desarrollador de curso prepara la información necesaria para capacitar a los usuarios del producto. 2-El desarrollador de curso elabora los materiales para la capacitación dígame Curso para aprender a utilizar las herramientas del sistema, cursos para darle mantenimiento al sistema. 3-El desarrollador de curso entrega los materiales de capacitaciones a los usuarios finales del producto.	
4- El cliente se dispone a recibir la documentación que le facilita el aprendizaje del sistema a utilizar.		
Prioridad	Crítico	
Flujos Alternos		
Acción del actor	Respuesta del proceso de negocio	
	4.1 Si el cliente encuentra algún material de capacitación con elementos engorrosos o con duda se dispone a los cambios pertinentes en los materiales de capacitación.	
Mejoras		

Anexo 2. Diagramas de Actividades

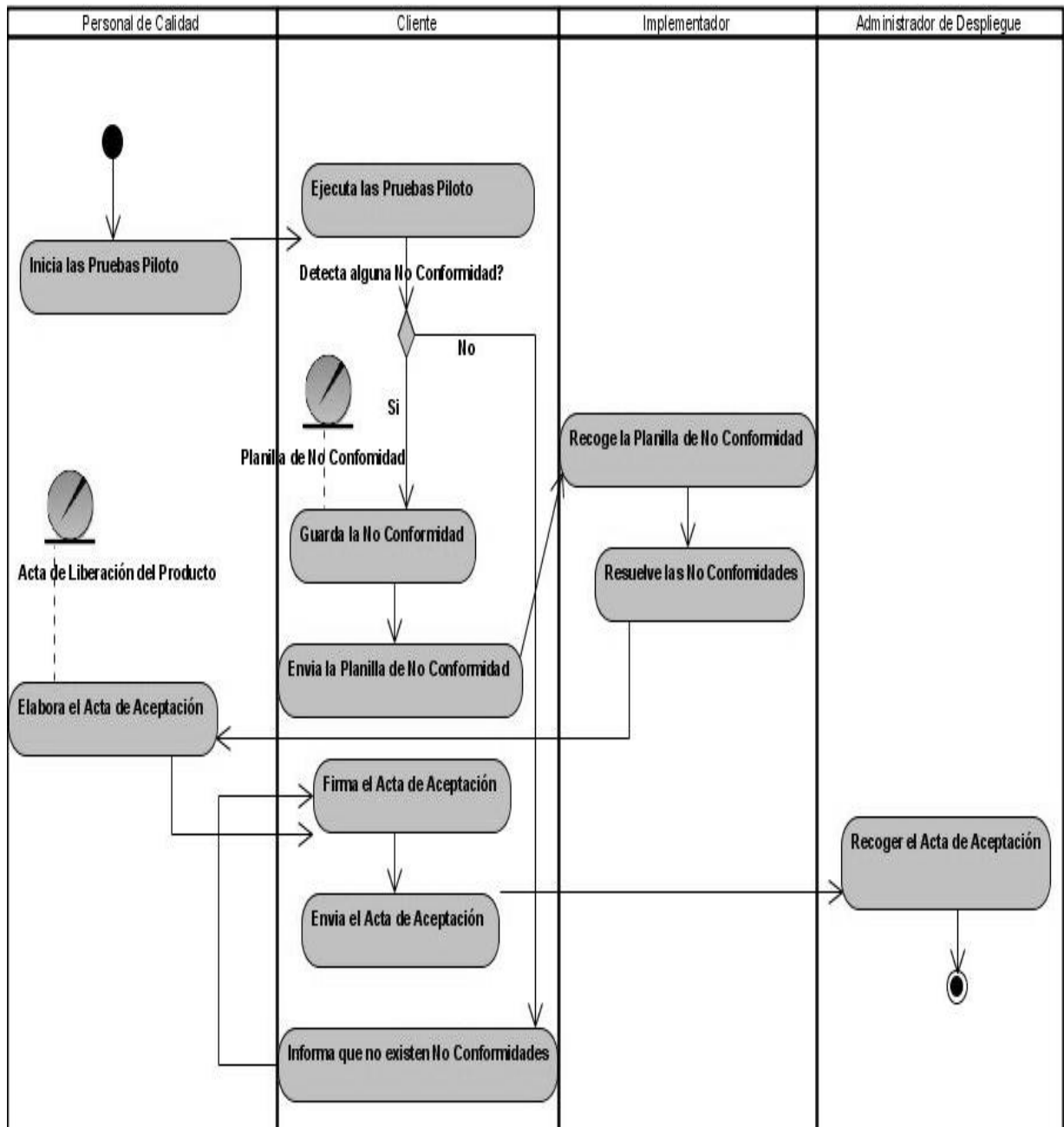


Ilustración 25: Diagrama de Actividades del Caso de Uso Administrar proceso de pruebas de piloto.

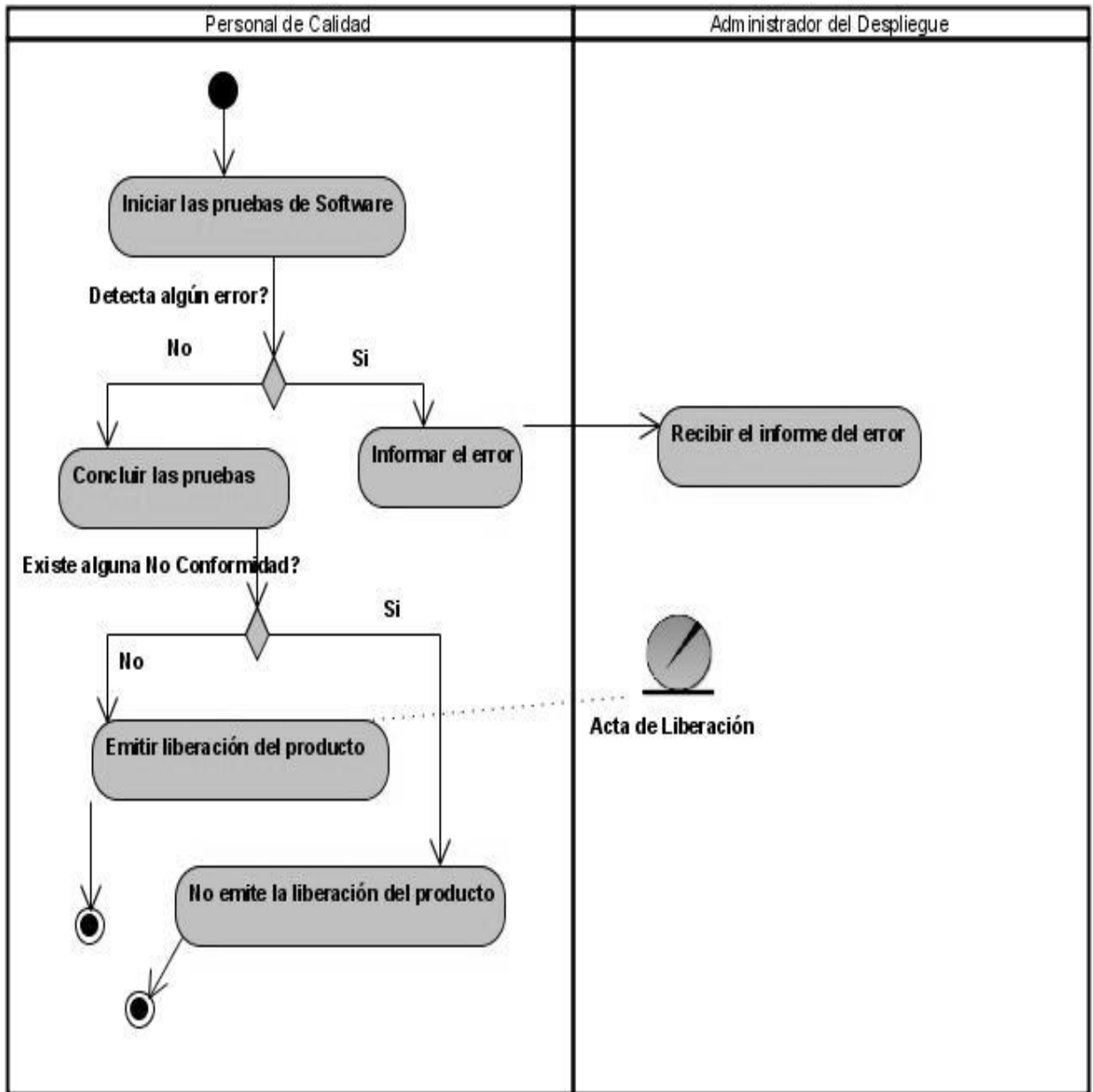


Ilustración 26: Diagrama de Actividades del Caso de Uso Realizar Liberación del Producto.

Anexos.

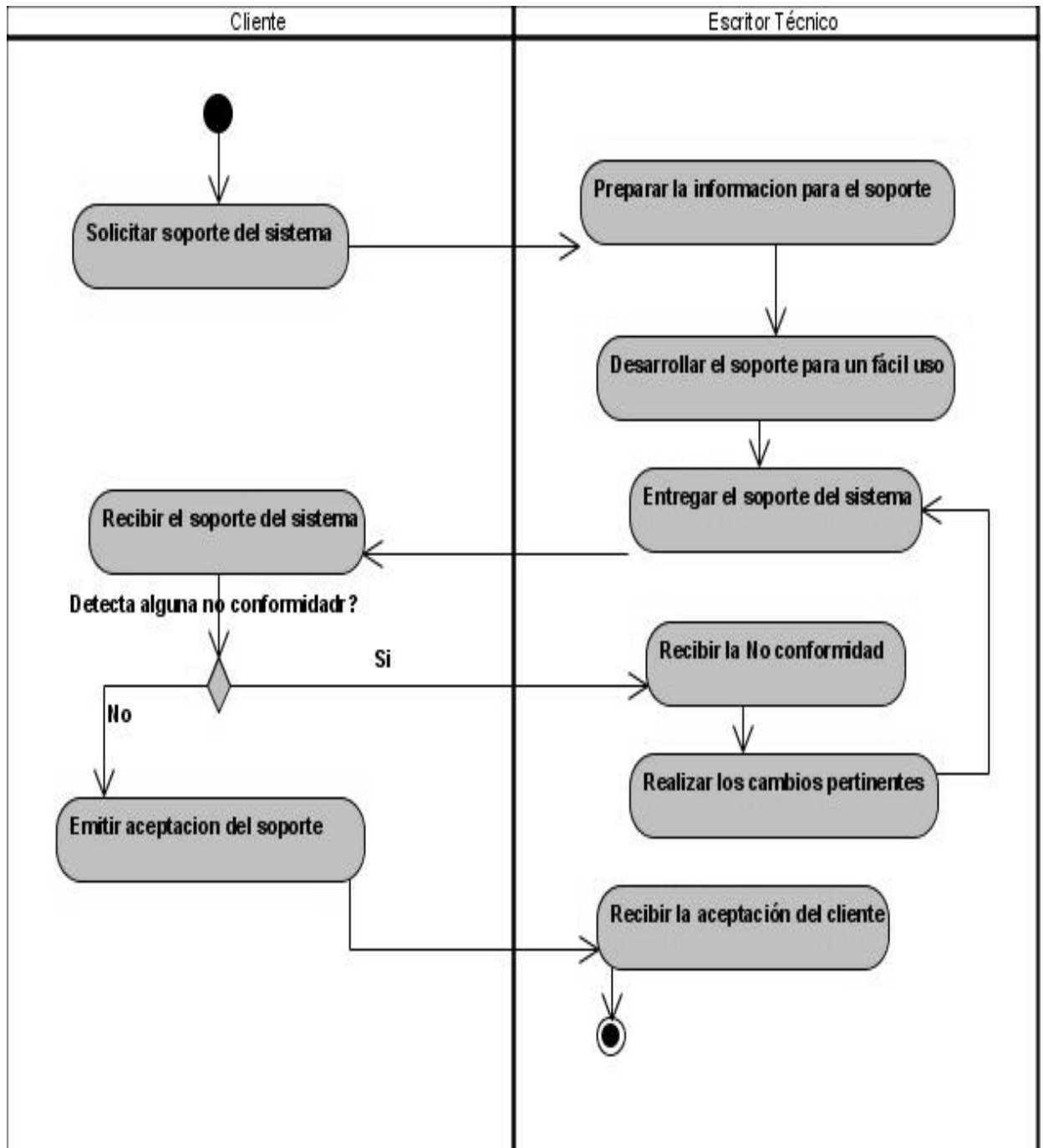


Ilustración 27: Diagrama de Actividades del Caso de Uso Brindar Soporte al Sistema.

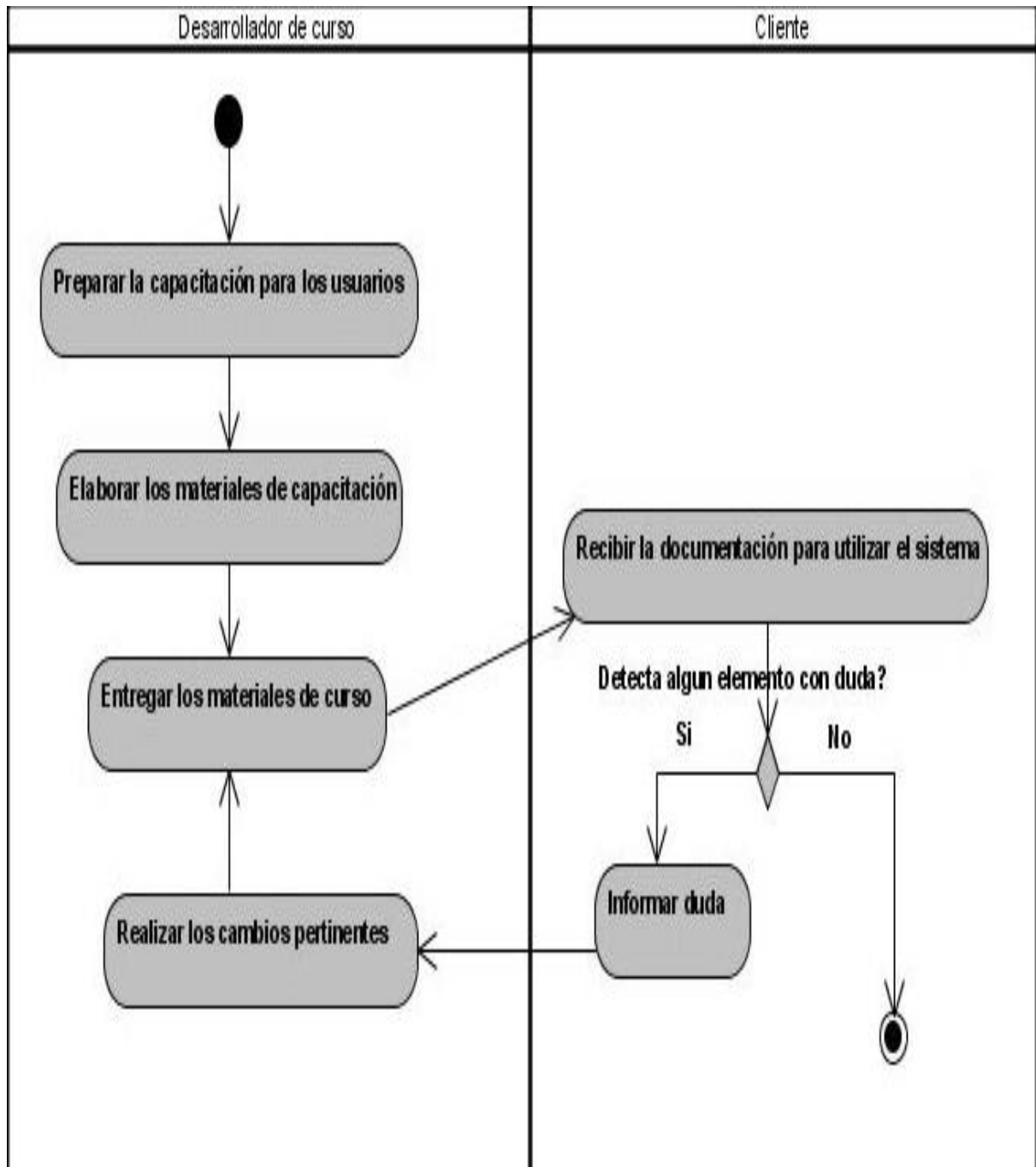


Ilustración 28: Diagrama de Actividades del Caso Uso Realizar Transferencia Tecnológica.

Anexo 3.Descripciones Textuales de los Casos de Uso del Sistema

Tabla 35: Descripción del CUS Administrar Proceso de Pruebas de Piloto.

Nombre del Caso de Uso	Administrar Proceso de Pruebas de Piloto
Actores	Usuario(Personal Calidad, Cliente, Implementador)
Propósito	Administrar el proceso de pruebas piloto registrando en cuantos centros o instituciones se realizará el pilotaje, insertando No Conformidades, buscar y visualizar las No Conformidad, visualizar el estado de las pruebas, generar la Plantilla de No Conformidades e

Anexos.

	imprimirla, además generar el Acta de Aceptación del Producto e imprimirla.
Resumen	<p>El caso de uso se inicia cuando el Personal de Calidad va a realizar alguna de las siguiente operaciones relacionadas con el proceso de pruebas de aceptación:</p> <ul style="list-style-type: none"> • Registrar en cuantos centros o instituciones se realizará el pilotaje • Insertar No Conformidad. • Buscar No Conformidad. • Visualizar No Conformidad. • Visualizar estado de las pruebas. • Generar Planilla de No Conformidades. • Imprimir planilla. <p>El sistema le muestra la interfaz correspondiente según su solicitud y ejecuta las acciones necesarias. El caso de uso finaliza cuando se emite el resultado de la operación solicitada.</p>
Precondiciones	Que el Usuario se haya autenticado en la aplicación.
Referencia	R4.1, R4.2, R4.3, R4.4, R4.5, R4.6, R4.6.1, R4.7, R4.7.1
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
<p>2. El Personal de Calidad, quiere realizar una de las siguientes operaciones:</p> <ul style="list-style-type: none"> • Registrar en cuantos centros o instituciones se realizará el pilotaje. • Insertar No Conformidad. • Buscar No Conformidad. • Visualizar No Conformidad. • Visualizar estado de las pruebas. • Generar Planilla de No Conformidades. • Imprimir planilla. 	<p>2. El sistema, en dependencia de la operación que solicita realizar, hace lo siguiente:</p> <ul style="list-style-type: none"> • Si decide Registrar en cuantos centros o instituciones se realizará el pilotaje, ir a Sección “Registrar en cuantos centros o instituciones se realizará el pilotaje”. • Si decide Insertar No Conformidad, ir a Sección “Insertar No Conformidad”. • Si decide Buscar No Conformidad y visualizarla, ir a Sección “Buscar No Conformidad”. • Si decide visualizar el estado de las pruebas, ir a la Sección “Visualizar estado de las pruebas”.

Anexos.

	<ul style="list-style-type: none"> Si decide generar Reporte de No Conformidades e imprimirlo, ir a la Sección “Generar Planilla de No Conformidades”.
Sección “Registrar en cuantos centros o instituciones se realizará el pilotaje”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente al registro de los centros o instituciones donde se realizará el pilotaje.
2. El Cliente provee los datos necesarios para el registro de los centros o instituciones donde se realizará el pilotaje: <ul style="list-style-type: none"> Nombre del centro o institución. Cantidad de PC del centro. 	3. Verifica que introdujo los <i>datos obligatorios</i> para poder proceder al registro de los centros o instituciones donde se realizará el pilotaje: <ul style="list-style-type: none"> Nombre del centro o institución. Cantidad de PC del centro.
	4. Registra el centro o institución y emite un mensaje de registro satisfactorio.
	5. Da la posibilidad de registrar otro centro o institución.
Flujos Alternos	
	8.1 Si falta algún dato necesario para el registro del centro o institución donde se realizará el pilotaje, emite un mensaje de error y no procede al registro.
Sección “Insertar No Conformidad”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a insertar no conformidades.
4. El Cliente selecciona el proyecto sobre el cual se están desarrollando las pruebas e introduce los parámetros para la inserción de la(s) no conformidades. <ul style="list-style-type: none"> Tipo (Alta , media y baja) Descripción de la No Conformidad. Fecha(se registra la fecha del momento que se inserta la no conformidad) Persona que la inserta. 	3. Inserta la No conformidad y muestra un mensaje de operación satisfactoria.
4. El Cliente selecciona la no conformidad que desea visualizar.	5. Visualiza los datos de la no conformidad seleccionada.
	6. Da la posibilidad de realizar otra búsqueda.
Flujos Alternos	
	3.1 Si quedó algún campo de información necesaria vacío, emite un

Anexos.

	mensaje de error de completitud de datos.
Sección “Buscar No Conformidad”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a la búsqueda de no conformidades.
2. El Usuario Introduce los parámetros por los cuales desea realizar la búsqueda de la(s) no conformidades. <ul style="list-style-type: none"> • Fecha • Tipo 	3. Muestra el resultado de la búsqueda, según parámetros seleccionados anteriormente por el usuario.
4. El Usuario selecciona la no conformidad que desea visualizar.	5. Visualiza los datos de la no conformidad seleccionada.
	6. Da la posibilidad de realizar otra búsqueda.
Flujos Alternos	
2.1 Si solicita realizar la búsqueda y no seleccionó ningún parámetro, emite un mensaje de error: “Debe seleccionar al menos un parámetro para realizar la búsqueda”.	
4.1 Si no desea visualizar los datos de alguna de las no conformidades del resultado de la búsqueda, sale de la sección o puede realizar otra búsqueda.	
Sección “Visualizar estado de las pruebas”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a visualizar el estado de las pruebas.
2. El Usuario selecciona la prueba que desea visualizar su estado.	3. Visualiza el estado de la prueba seleccionada.
Flujos Alternos	
2.1 Si no desea visualizar el estado de las pruebas del resultado de la búsqueda, sale de la sección o puede realizar otra búsqueda.	
Sección “Generar Reporte de No Conformidad”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a la generación de la Reporte de No Conformidad.
2. El Cliente toma de la base de datos del reporte de No Conformidades: <ul style="list-style-type: none"> • Nombre del producto. • Fecha (fecha de inicio y fecha de fin) • Observaciones. 	3. Muestra el reporte y guarda la fecha del reporte que se genera y da la posibilidad de imprimir dicho reporte.
4. El Cliente indica imprimir el reporte.	

Anexos.

	5. Imprime el reporte.
	6. Da la posibilidad de generar otro reporte.
Flujos Alternos	
	2.1 Si quedó algún campo de información necesaria vacío, emite un mensaje de error de completitud de datos.
	4.1 Si no desea imprimir el reporte, sale de la sección o puede generar otro reporte.
Requerimientos especiales	-
Prioridad	Crítico
Postcondiciones	El sistema registra en cuantos centros o instituciones se realizará el pilotaje, inserta las No Conformidades correspondiente al software en despliegue en el sistema, busca y visualiza el las No Conformidades almacenadas, genera e imprime la Planilla de No Conformidades y el Acta de Aceptación del Producto.

Tabla 36: Descripción del CUS Administrar Control de recursos humanos y logística para el despliegue.

Nombre del Caso de Uso	Administrar Control de recursos humanos y logística para el despliegue.
Actores	Líder de Proyecto (inicia)
Propósito	Registrar el personal que participará en el despliegue, buscar y visualizar el personal del despliegue del proyecto en despliegue, modificar datos del personal del despliegue registrado en el sistema, generar reporte del personal del despliegue e imprimir dicho reporte, además de Registrar viajes de ida y regreso a los centro clientes.
Resumen	<p>El caso de uso se inicia cuando el Líder va a realizar alguna de las siguiente operaciones relacionadas con el Control de recursos humanos para el despliegue:</p> <ul style="list-style-type: none"> • Registrar personal que participará en el despliegue. • Buscar personal del despliegue. • Visualizar personal del despliegue. • Modificar personal del despliegue. • Generar reporte de personal del despliegue. • Imprimir reporte.

Anexos.

	<ul style="list-style-type: none"> • Registrar viajes de ida y regreso a los centro clientes. <p>El sistema le muestra la interfaz correspondiente según su solicitud y ejecuta las acciones necesarias. El caso de uso finaliza cuando se emite el resultado de la operación solicitada.</p>
Precondiciones	Que el Líder se haya autenticado en la aplicación y que haya escogido un proyecto sobre el cual va hacer el control de recursos humanos y logística.
Referencia	R5.1, R5.2, R5.3, R5.4, R5.5, R5.5.1
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
<p>1. El Líder del Proyecto, quiere realizar una de las siguientes operaciones:</p> <ul style="list-style-type: none"> • Registrar personal que participará en el despliegue. • Buscar personal del despliegue. • Visualizar personal del despliegue. • Modificar personal del despliegue. • Generar reporte de personal del despliegue. • Imprimir reporte. • Registrar viajes de ida y regreso a los centro clientes. 	<p>2. El sistema, en dependencia de la operación que solicita realizar, hace lo siguiente:</p> <ul style="list-style-type: none"> • Si decide Registrar personal que participará en el despliegue, ir a Sección “Registrar personal que participará en el despliegue”. • Si decide buscar personal del despliegue, ir a Sección “Buscar personal del despliegue”. • Si decide visualizar personal del despliegue, ir a Sección “Visualizar personal del despliegue”. • Si decide modificar personal del despliegue, ir a Sección “Modificar personal del despliegue”. • Si decide Registrar viajes de ida y regreso a los centro clientes, ir a Sección “Registrar viajes de ida y regreso a los centro clientes”.
Sección “Registrar personal que participará en el despliegue”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente al registro del personal que participará en el despliegue.

Anexos.

<p>2. Provee los datos necesarios para el registro del personal que participará en el despliegue (el sistema solo registrará una persona por vez):</p> <ul style="list-style-type: none"> • Nombre de la persona. • Rol en el proyecto. • Datos. (teléfono, email, dirección particular) • Años de experiencia en despliegue. • Participación en otros despliegues. 	<p>3. Verifica que introdujo los <i>datos obligatorios</i> para poder proceder al registro del personal que participará en el despliegue:</p> <ul style="list-style-type: none"> • Nombre de la persona. • Rol en el proyecto. • Años de experiencia en despliegue. • Participación en otros despliegues.
	<p>4. Registra el personal que participará en el despliegue (el sistema solo registrará una persona por vez) y emite un mensaje de registro satisfactorio.</p>
	<p>5. Da la posibilidad de registrar más personal que participará en el despliegue.</p>
Flujos Alternos	
	<p>8.1 Si falta algún dato necesario para el registro del personal que participará en el despliegue, emite un mensaje de error y no procede al registro.</p>
Sección “Buscar personal del despliegue”	
Acciones del Actor	Respuesta del Sistema
	<p>1. Muestra la interfaz correspondiente a la búsqueda del personal del despliegue.</p>
<p>2. Introduce los parámetros por los cuales desea realizar la búsqueda del personal del despliegue:</p> <ul style="list-style-type: none"> • Nombre de la persona. • Rol en el proyecto. 	<p>3. Muestra el resultado de la búsqueda, según parámetros seleccionados anteriormente por el usuario.</p>
<p>3. Selecciona la persona del despliegue que desea visualizar sus datos.</p>	<p>4. Visualiza los datos del personal del despliegue seleccionado.</p>
	<p>5. Da la posibilidad de realizar otra búsqueda.</p>
Flujos Alternos	
<p>2.1 Si solicita realizar la búsqueda y no seleccionó ningún parámetro, emite un mensaje de error: "Debe seleccionar al menos un parámetro para realizar la búsqueda".</p>	
<p>3.1 Si no desea visualizar los datos de algún personal del despliegue del resultado de la búsqueda, sale de la sección o puede realizar otra búsqueda.</p>	
Sección “Modificar personal del despliegue”	
Acciones del Actor	Respuesta del Sistema
	<p>1. Muestra la interfaz correspondiente para la modificación de los datos del</p>

Anexos.

	personal de despliegue.
	2. Ir a la sección "Buscar personal del despliegue".
	3. Ir a la sección "Visualizar personal del despliegue".
	4. Localiza el personal del despliegue y da la posibilidad de modificar los datos. Los campos que podrán ser modificados son: Rol en el proyecto, Datos (Teléfono, email). El resto de los campos aparecerán deshabilitados, o sea, sin posibilidad de modificar los datos.
5. Actualiza los datos del personal del despliegue.	6. Verifica que no se haya quedado ningún campo de información obligatoria vacío (los indicados en el registro).
	7. Registra la actualización en la base de datos y emite un mensaje de operación satisfactoria.
	8. Permite buscar otra persona del despliegue para hacer modificación en sus datos.
Flujos Alternos	
	4.1 Si los datos proveídos no son suficientes para la búsqueda o el personal del despliegue no existe en el sistema, emite un mensaje de error o no existencia de la persona del despliegue.
	6.1 Si quedó algún campo de información necesaria vacío, emite un mensaje de error de completitud de datos.
Sección "Generar Reporte de Personal de Despliegue"	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a la generación de la Reporte del Personal de Despliegue
2. El Líder toma de la base de datos el reporte del Personal del Despliegue el cual se carga en una tabla los datos del personal y genera el reporte: <ul style="list-style-type: none"> • Nombre de la persona. • Rol en el proyecto. • Datos. (teléfono, email, dirección particular) • Años de experiencia en despliegue. • Participación en otros despliegues. 	3. Muestra el reporte con todos los datos de todo el Personal del Despliegue y da la posibilidad de imprimir dicho reporte.
4. El Cliente indica imprimir el reporte.	

Anexos.

	5. Imprime el reporte.
Flujos Alternos	
	2.1 Si quedó algún campo de información necesaria vacío, emite un mensaje de error de completitud de datos.
	4.1 Si no desea imprimir el reporte, sale de la sección o puede generar otro reporte.
Sección “Registrar viajes de ida y regreso a los centro clientes”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente al registro de los viajes de ida y regreso a los centros clientes.
2. Provee los datos necesarios para el registro de los viajes de ida y regreso a los centros clientes (el sistema solo registrará una un viaje por vez): <ul style="list-style-type: none"> • Origen. • Destino. • Cantidad de Personas. • Fecha. • Medio de Transporte (Microbús, Ómnibus). 	3. Verifica que introdujo los <i>datos obligatorios</i> para poder proceder al registro de los viajes de ida y regreso a los centros clientes: <ul style="list-style-type: none"> • Origen. • Destino. • Cantidad de Personas. • Fecha. • Medio de Transporte (Microbús, Ómnibus).
	4. Registra el viaje de ida y de regreso al centro cliente (el sistema solo registrará una viaje por vez) y emite un mensaje de registro satisfactorio.
	5. Da la posibilidad de registrar más viajes de ida y regreso a los centros clientes.
Flujos Alternos	
	8.1 Si falta algún dato necesario para el registro de los viajes de ida y regreso a los centros clientes, emite un mensaje de error y no procede al registro.
Requerimientos especiales	-
Prioridad	Crítico
Postcondiciones	El sistema registra el personal que participará en el despliegue, busca y visualiza el personal del despliegue del proyecto en despliegue, modifica datos del personal del despliegue registrado en el sistema, genera el reporte del personal del despliegue e imprime dicho reporte.

Tabla 37: Descripción del CUS Administrar Control del Sistema.

Nombre del Caso de Uso	Administrar Control del Sistema
-------------------------------	---------------------------------

Anexos.

Actores	Administrador del Sistema(inicia)
Propósito	Crear un usuario en el sistema, buscar y visualizar el usuario creado en el sistema, modificar datos permisos del usuario creado, eliminar usuario del sistema.
Resumen	<p>El caso de uso se inicia cuando el Administrador del sistema va a realizar alguna de las siguiente operaciones relacionadas con el Control del Sistema:</p> <ul style="list-style-type: none"> • Autenticar Usuario. • Crear Usuario. • Buscar Usuario. • Visualizar Usuario. • Modificar Permiso de Usuario. • Eliminar Usuario. <p>El sistema le muestra la interfaz correspondiente según su solicitud y ejecuta las acciones necesarias. El caso de uso finaliza cuando se emite el resultado de la operación solicitada.</p>
Precondiciones	Que el Administrador del Sistema se haya autenticado en la aplicación.
Referencia	R6.1, R6.2, R6.3, R6.4, R6.5
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
<p>1. El Administrador del Sistema, quiere realizar una de las siguientes operaciones:</p> <ul style="list-style-type: none"> • Autenticar Usuario. • Crear Usuario. • Buscar Usuario. • Visualizar Usuario. • Modificar Permiso de Usuario. • Eliminar Usuario. 	<p>2. El sistema, en dependencia de la operación que solicita realizar, hace lo siguiente:</p> <ul style="list-style-type: none"> • Si decide entrar al sistema, ir a la Sección de entrada del sistema “Autenticar usuario”. • Si decide Crear un usuario, ir a Sección “Crear usuario”. • Si decide Buscar usuario y visualizarlo, ir a Sección “Buscar usuario”. • Si decide modificar permiso de usuario, ir a Sección “Modificar permiso de usuario”. • Si decide eliminar un usuario, ir a la Sección “Eliminar usuario”.
Sección “Autenticar Usuario”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a la autenticación de un usuario.

Anexos.

2. Provee los datos necesarios para la autenticación de un usuario: <ul style="list-style-type: none"> Nombre de usuario. Contraseña. 	2. Verifica que introdujo los <i>datos obligatorios</i> para poder proceder a la autenticación de un usuario: <ul style="list-style-type: none"> Nombre de usuario. Contraseña.
	3. Autentica el usuario y emite un mensaje de registro satisfactorio.
Flujos Alternos	
	3.1 Si falta algún dato necesario para la autenticación del usuario o los datos no son correctos, emite un mensaje de error y no procede a la autenticación.
Sección “Crear Usuario”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente creación de un usuario.
2. Provee los datos necesarios para la creación de un usuario: <ul style="list-style-type: none"> Nombre de usuario. Rol. Id (Estudiante o Profesor). Nombre 1er Apellido 2do Apellido E-mail Teléfono Celular Departamento (Profesor) Grupo (Estudiante) 	3. Verifica que introdujo los <i>datos obligatorios</i> para poder proceder a la creación de un usuario: <ul style="list-style-type: none"> Nombre de usuario Rol. Id (Estudiante o Profesor). 1er Apellido 2do Apellido E-mail Teléfono Celular Departamento (Profesor) Grupo (Estudiante)
	4. Crea el usuario y emite un mensaje de registro satisfactorio.
	5. Da la posibilidad de crear otro usuario.
Flujos Alternos	
	3.1 Si falta algún dato necesario para la creación del usuario, emite un mensaje de error y no procede a la creación.
Sección “Buscar Usuario”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a la búsqueda de un usuario.
2. Introduce los parámetros por los cuales desea realizar la búsqueda del usuario. <ul style="list-style-type: none"> Nombre de usuario. Rol. Id (Estudiante o Profesor). 	3. Muestra el resultado de la búsqueda, según parámetros seleccionados anteriormente por el usuario.
4. Selecciona el usuario que desea visualizar sus datos.	5. Visualiza los datos del usuario seleccionado.
	4. Da la posibilidad de realizar otra búsqueda.

Anexos.

Flujos Alternos	
2.1 Si solicita realizar la búsqueda y no seleccionó ningún parámetro, emite un mensaje de error: "Debe seleccionar al menos un parámetro para realizar la búsqueda".	
4.1 Si no desea visualizar los datos de algún usuario del resultado de la búsqueda, sale de la sección o puede realizar otra búsqueda.	
Sección "Modificar Datos de usuario"	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente para la modificación de los datos del usuario.
	2. Ir a la sección "Buscar usuario".
	3. Ir a la sección "Visualizar usuario".
	4. Localiza el usuario y da la posibilidad de modificar sus datos. Los campos que podrán ser modificados son: <ul style="list-style-type: none"> • Nombre de usuario. • Rol. • Id (Estudiante o Profesor). • E-mail • Teléfono • Celular • Departamento (Profesor) • Grupo (Estudiante) • Privilegio o Tipo Usuario. El resto de los campos aparecerán deshabilitados, o sea, sin posibilidad de modificar los datos.
5. Actualiza los datos del usuario. <ul style="list-style-type: none"> • Nombre de usuario. • Rol. • Id (Estudiante o Profesor). • E-mail • Teléfono • Celular • Departamento (Profesor) • Grupo (Estudiante) 	6. Verifica que no se haya quedado ningún campo de información obligatoria vacío (los indicados en creación).
	7. Registra la actualización en la base de datos y emite un mensaje de operación satisfactoria.
	8. Permite buscar otro usuario para hacer modificación en su permiso.
Flujos Alternos	

Anexos.

Acción del actor	2.1 Si los datos proveídos no son suficientes para la búsqueda o el usuario no existe en el sistema, emite un mensaje de error o no existencia de la muestra.
	6.1 Si quedó algún campo de información necesaria vacío, emite un mensaje de error de completitud de datos.
Sección “Eliminar usuario”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente a eliminar un usuario.
2. Selecciona el usuario que desea eliminar del sistema y confirma la solicitud de eliminación (Aceptar o Cancelar).	3. Elimina el usuario seleccionado.
Flujos Alternos	
2.1 Si no desea eliminar ningún usuario del sistema, sale de la sección.	
Requerimientos especiales	-
Prioridad	Critico
Postcondiciones	El sistema crear un usuario correspondiente, busca y visualiza el usuario creado en el sistema, modifica datos permisos del usuario creado, eliminar usuario del sistema.

Tabla 38: Descripción del CUS Gestionar Proyecto a Desplegar.

Nombre del Caso de Uso	Gestionar proyecto a Desplegar
Actores	Líder de Proyecto (inicia)
Propósito	Registrar el proyecto a desplegar, modificar el registro del proyecto en despliegue, buscar el proyecto desplegado o en fase a desplegar, visualizar el estado del despliegue de un proyecto.
Resumen	<p>El caso de uso se inicia cuando el Líder va a realizar alguna de las siguiente operaciones relacionadas con la gestión del proyecto a desplegar:</p> <ul style="list-style-type: none"> • Registrar Proyecto a Desplegar. • Modificar Registro del Proyecto. • Buscar Proyecto Desplegado o en Fase de Despliegue. • Visualizar Estado del Despliegue de un Proyecto. <p>El sistema le muestra la interfaz correspondiente según su solicitud y ejecuta las acciones necesarias. El caso de uso finaliza cuando se emite el resultado de la operación solicitada.</p>
Precondiciones	Que el Líder se haya autenticado en la aplicación.
Referencia	R7.1, R7.2, R7.3, R7.4
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema

Anexos.

<p>1. El administrador del Sistema, quiere realizar una de las siguientes operaciones:</p> <ul style="list-style-type: none"> • Registrar Proyecto a Desplegar. • Modificar Registro del Proyecto. • Buscar Proyecto Desplegado o en Fase de Despliegue. 	<p>2. El sistema, en dependencia de la operación que solicita realizar, hace lo siguiente:</p> <ul style="list-style-type: none"> • Si decide Registrar proyecto a desplegar, ir a Sección “Registrar proyecto a Desplegar”. • Si decide modificar registro del proyecto, ir a Sección “Modificar registro del proyecto”. • Si decide Buscar proyecto desplegado o en fase de despliegue, ir a Sección “Buscar Proyecto Desplegado o en Fase de Despliegue”. • Si decide visualizar estado del despliegue, ir a la Sección “Visualizar Estado del Despliegue de un Proyecto”.
Sección “Registrar proyecto a Desplegar”	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente al registro del proyecto a desplegar.
<p>2. Provee los datos necesarios para el registro del proyecto a desplegar:</p> <ul style="list-style-type: none"> • Nombre del proyecto a desplegar. • Fecha del Despliegue (Inicio-Fin) • Tipos de proyecto (Exportación o Nacional) 	<p>3. Verifica que introdujo los <i>datos obligatorios</i> para poder proceder al registro del proyecto a desplegar:</p> <ul style="list-style-type: none"> • Nombre del proyecto a desplegar. • Fecha del Despliegue (Inicio-Fin) • Tipos de proyecto (Exportación o Nacional)
	4. Registra el nuevo proyecto a desplegar y emite un mensaje de registro satisfactorio.
	5. Da la posibilidad de registrar otro proyecto a desplegar.
Flujos Alternos	
	3.1 Si falta algún dato necesario para el registro del proyecto a desplegar, emite un mensaje de error y no procede al registro.
Sección “Modificar registro del proyecto”	

Anexos.

Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente para la modificación del registro del proyecto
	2. Ir a la sección "Buscar proyecto desplegado o en fase de despliegue".
	3. Ir a la sección "Visualizar proyecto desplegado o en fase de despliegue".
	4. Localiza el proyecto y da la posibilidad de modificar su registro. Los campos que podrán ser modificados son: <ul style="list-style-type: none"> • Nombre del proyecto a desplegar. • Fecha del Despliegue (Inicio-Fin) El resto de los campos aparecerán deshabilitados, o sea, sin posibilidad de modificar los datos.
5. Actualiza los datos del registro. <ul style="list-style-type: none"> • Nombre del proyecto a desplegar. • Fecha del Despliegue (Inicio-Fin) 	6. Verifica que no se haya quedado ningún campo de información obligatoria vacío (los indicados en el registro).
	7. Registra la actualización en la base de datos y emite un mensaje de operación satisfactoria.
	8. Permite buscar otro proyecto o fase para hacer modificación en su registro.
Flujos Alternos	
	2.1 Si los datos proveídos no son suficientes para la búsqueda o el proyecto desplegado o fase en despliegue no se encuentran en el sistema, emite un mensaje de error o no existencia.
	6.1 Si quedó algún campo de información necesaria vacío, emite un mensaje de error de completitud de datos.
Sección "Buscar Proyecto Desplegado o en Fase de Despliegue"	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente para la búsqueda de proyecto desplegado o en fase de despliegue.
2. Provee los datos necesarios para localizar el proyecto desplegado o en fase de despliegue: <ul style="list-style-type: none"> • Nombre del proyecto a desplegar. 	3. Muestra el resultado de la búsqueda.
	4. Da la posibilidad de buscar proyecto desplegado o en fase de despliegue.
Flujos Alternos	

Anexos.

	3.1 Si el resultado de la búsqueda es vacío, mostrar un mensaje "No está registrado el proyecto desplegado o la fase en despliegue en la Base de Datos".
Sección "Visualizar Estado del Despliegue de un Proyecto"	
Acciones del Actor	Respuesta del Sistema
	1. Muestra la interfaz correspondiente para la visualización del proyecto desplegado o en fase de despliegue.
2. Selecciona el proyecto desplegado o en fase de despliegue del cual quiere visualizar el estado, luego de buscarlo por su nombre de proyecto.	3. Visualiza el estado del proyecto desplegado o en fase de despliegue: <ul style="list-style-type: none"> • Estado del proyecto: Desplegado o en fase de despliegue (Cual fase esta en ese momento).
	4. Da la posibilidad de otro proyecto o fase en despliegue para visualizar su estado.
Flujos Alternos	
2.1 Si no desea visualizar el estado de ningún proyecto desplegado o en qué fase se encuentra, sale de la sección.	
Requerimientos especiales	-
Prioridad	Critico
Postcondiciones	El sistema registra el proyecto a desplegar, modifica el registro del proyecto en despliegue, busca el proyecto desplegado o en fase a desplegar, visualiza el estado del despliegue de un proyecto.

Anexo 4. Prototipo No Funcional de Interfaz de Usuario.

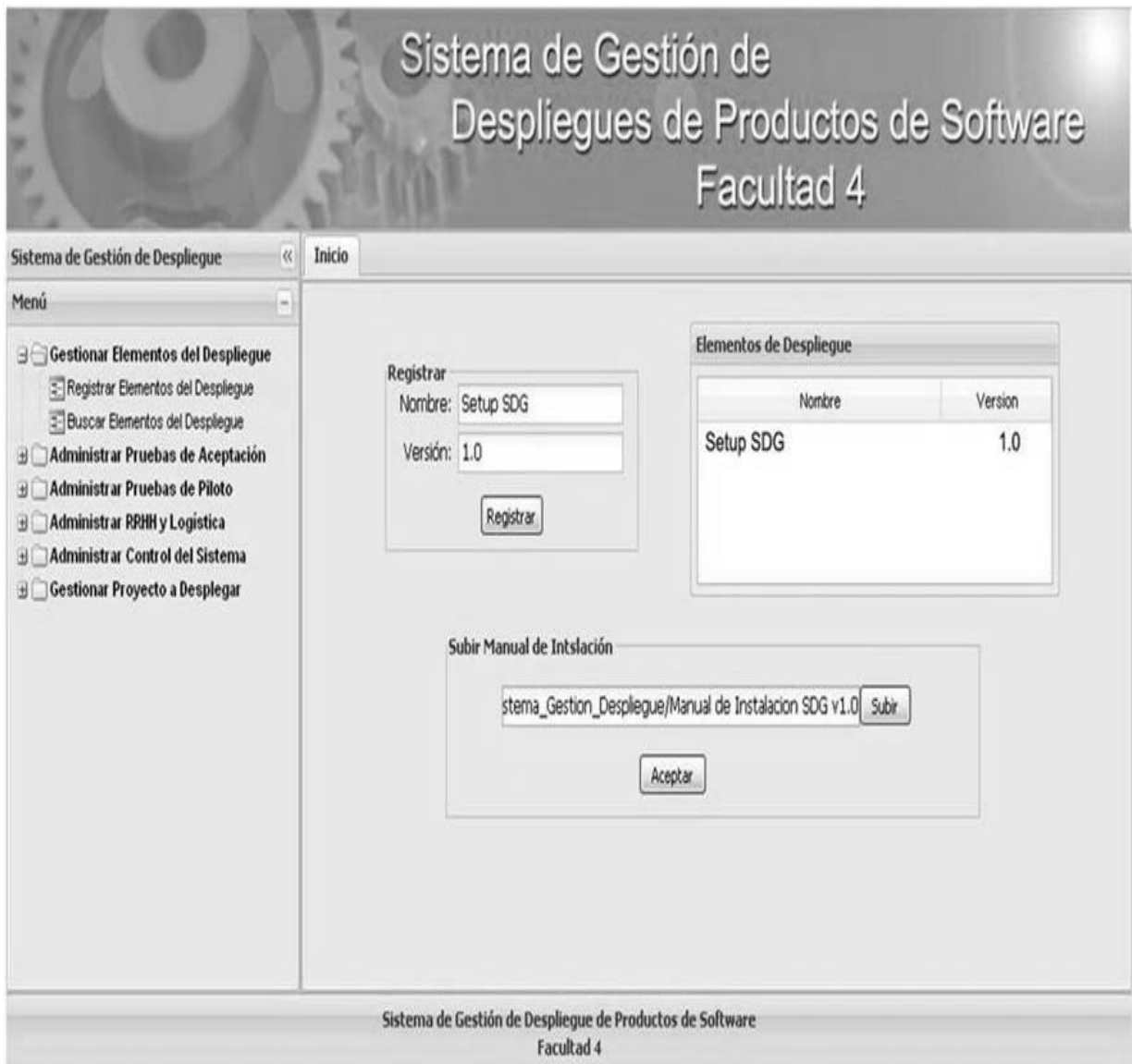


Ilustración 29: Prototipo No Funcional Interfaz Usuario Pantalla Gestionar Elemento del Despliegue Escenario Registrar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.

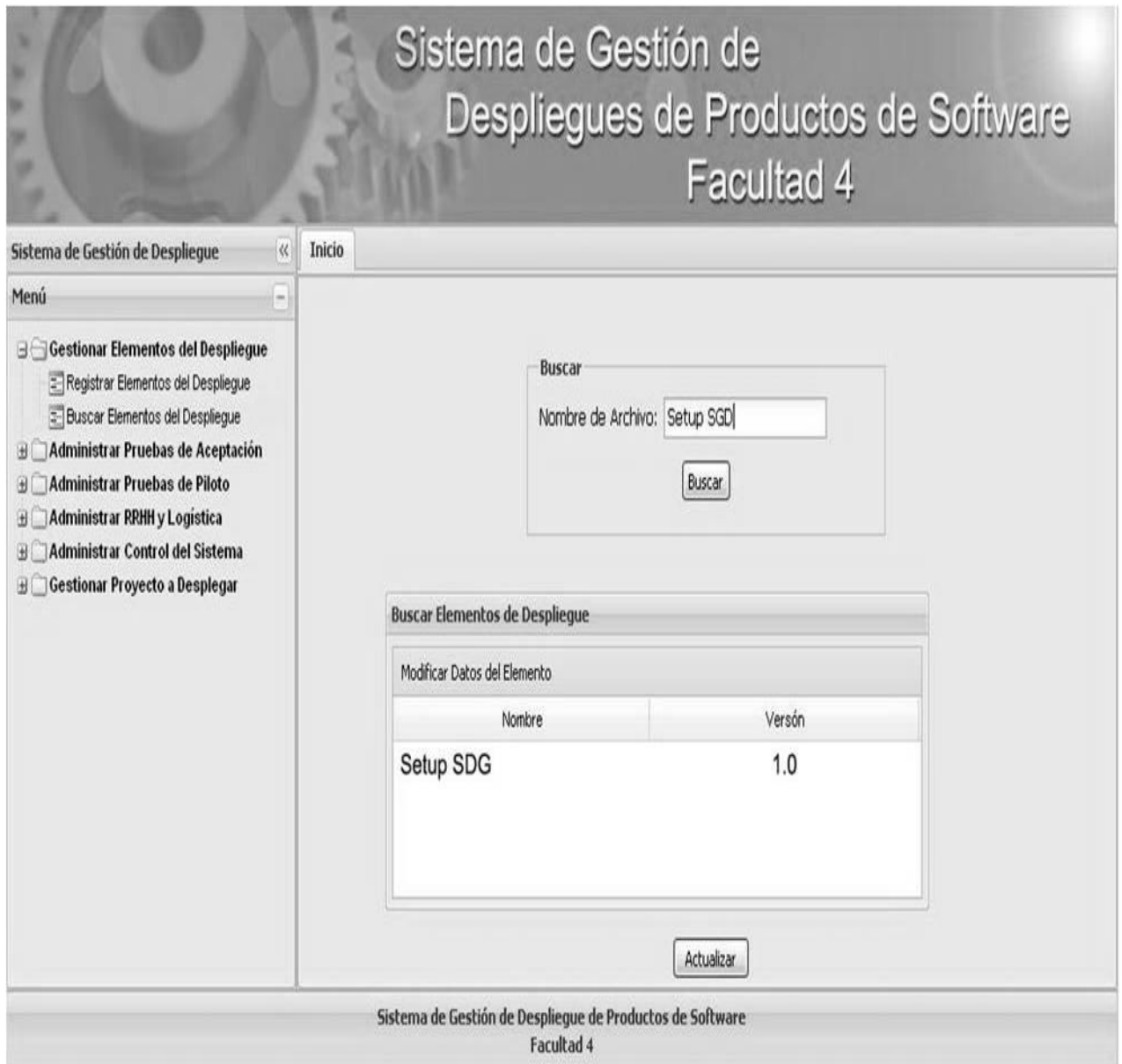


Ilustración 30: Prototipo No Funcional Interfaz Usuario Pantalla Gestionar Elemento del Despliegue Escenario Buscar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.

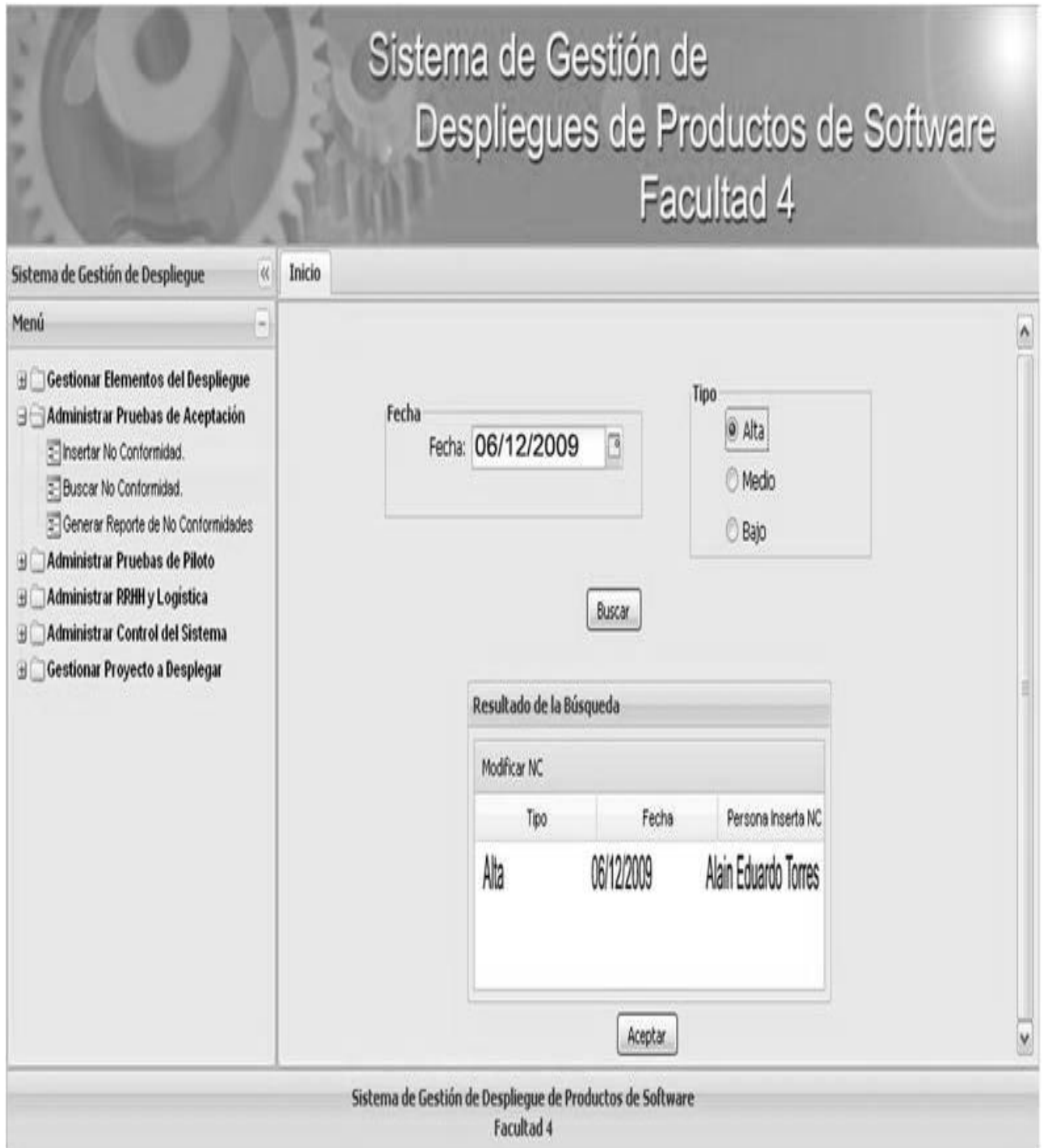


Ilustración 31: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Aceptación Escenario Buscar No Conformidad: despliegue de productos de software de la Facultad 4.

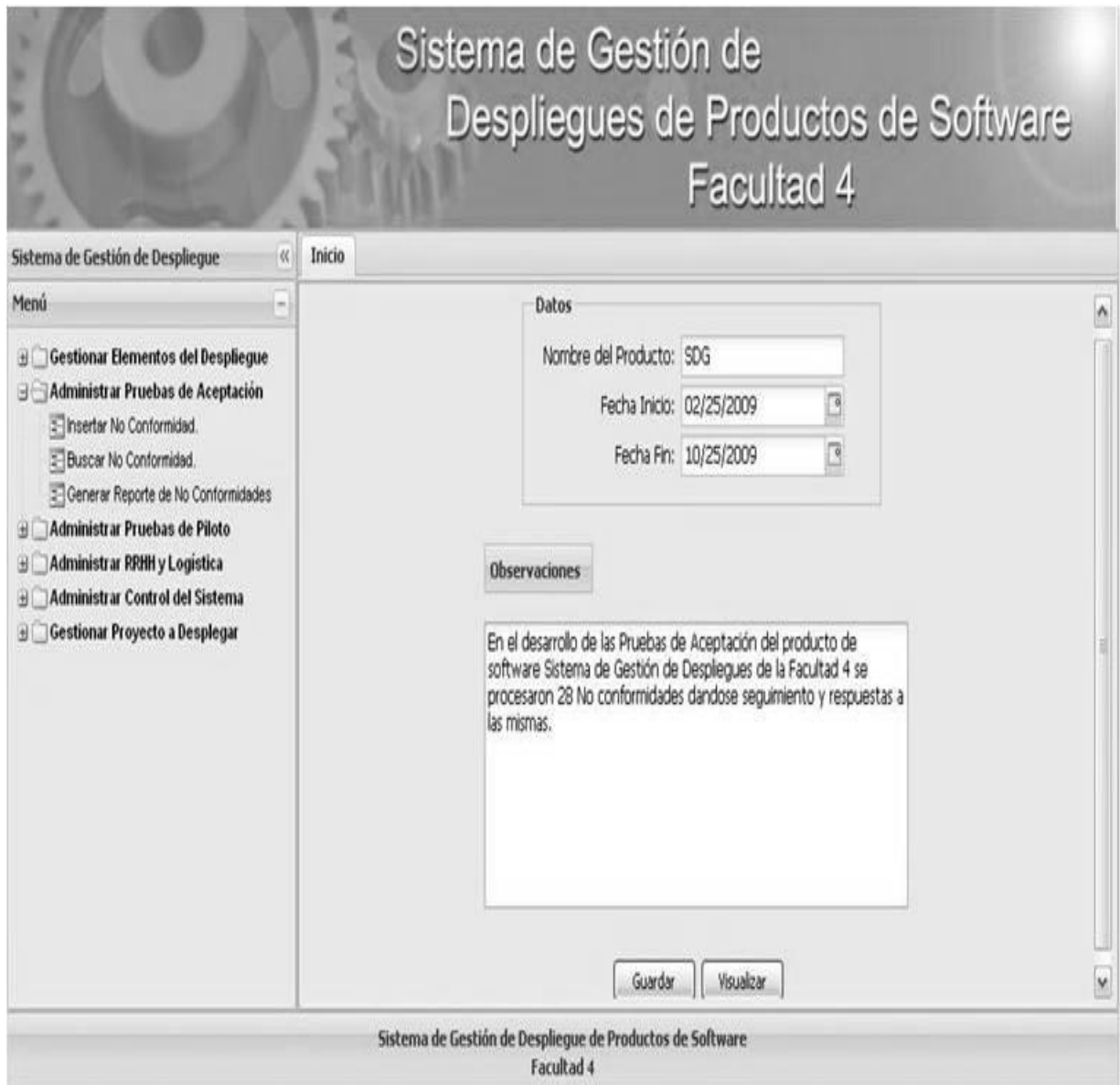


Ilustración 32: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Aceptación Escenario Generar Reporte de No Conformidad: despliegue de productos de software de la Facultad 4.

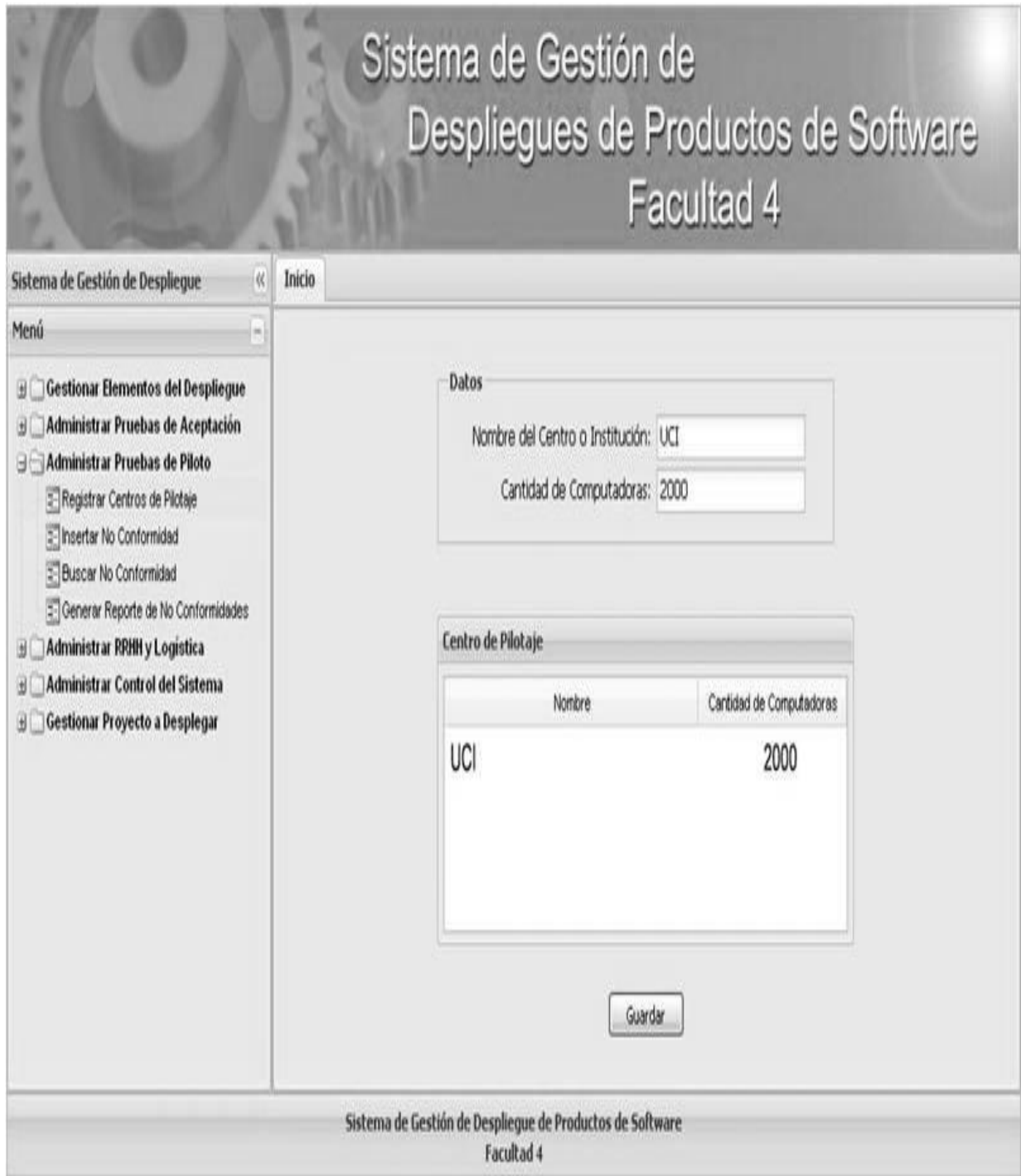


Ilustración 33: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Piloto Escenario Registrar Centros de Pilotaje: despliegue de productos de software de la Facultad 4.



Ilustración 34: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Piloto Escenario Insertar No Conformidad: despliegue de productos de software de la Facultad 4.



Ilustración 35: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Piloto Escenario Buscar No Conformidad: despliegue de productos de software de la Facultad 4.



Ilustración 36: Prototipo No Funcional Interfaz Usuario Pantalla Administración de Pruebas de Piloto Escenario Generar reporte de No Conformidades: despliegue de productos de software de la Facultad 4.



Ilustración 37: Prototipo No Funcional Interfaz Usuario Pantalla Administración de RR.HH y Logística Escenario Registrar Personal del despliegue: despliegue de productos de software de la Facultad 4.

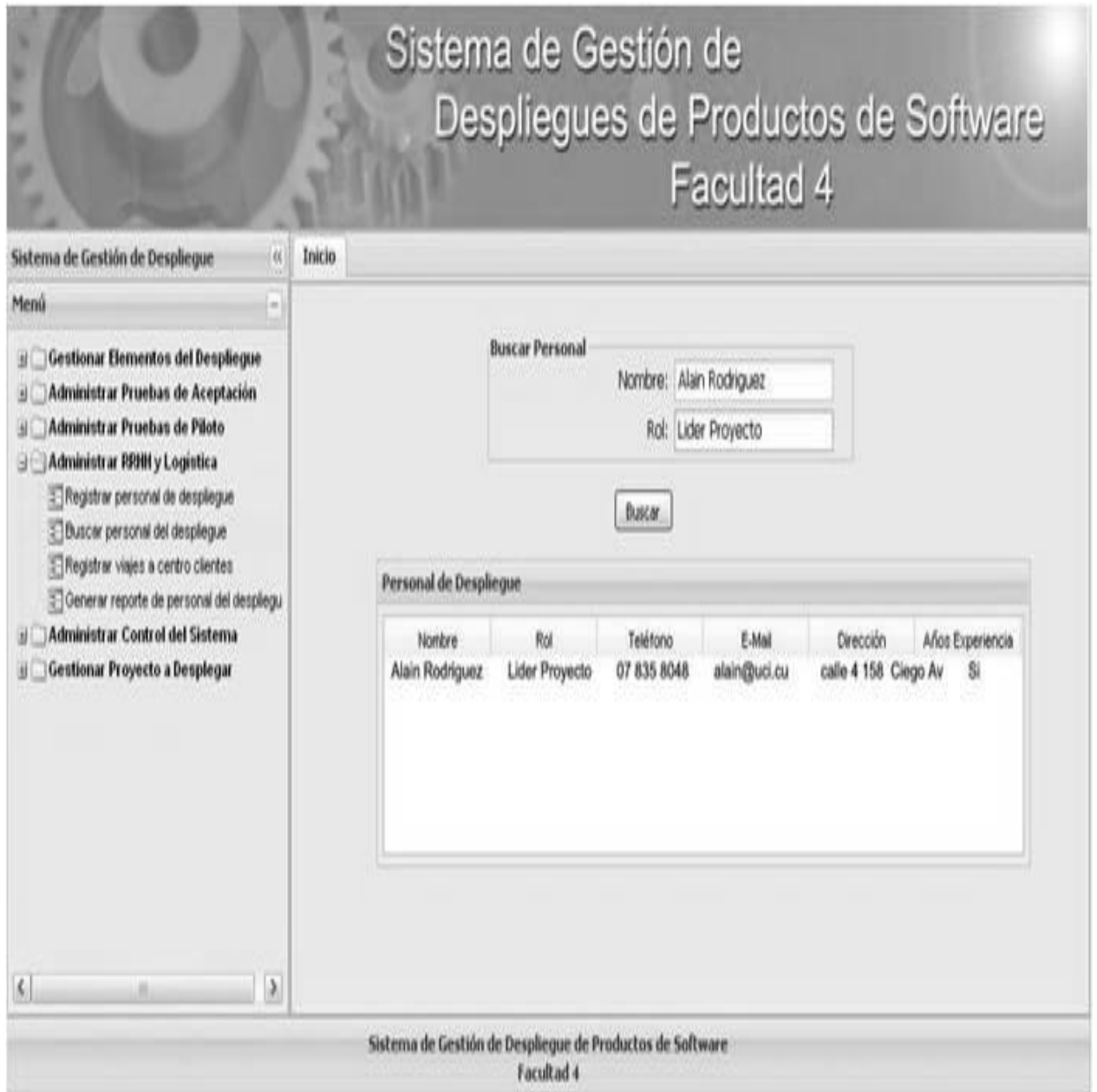


Ilustración 38: Prototipo No Funcional Interfaz Usuario Pantalla Administración de RR.HH y Logística Escenario Buscar Personal del despliegue: despliegue de productos de software de la Facultad 4.



Ilustración 39: Prototipo No Funcional Interfaz Usuario Pantalla Administración de RR.HH y Logística Escenario Registrar Viajes a Centros de Pilotaje: despliegue de productos de software de la Facultad 4.



Ilustración 40: Prototipo No Funcional Interfaz Usuario Pantalla Administración de RR.HH y Logística Escenario Generar Reporte del Personal del despliegue: despliegue de productos de software de la Facultad 4.

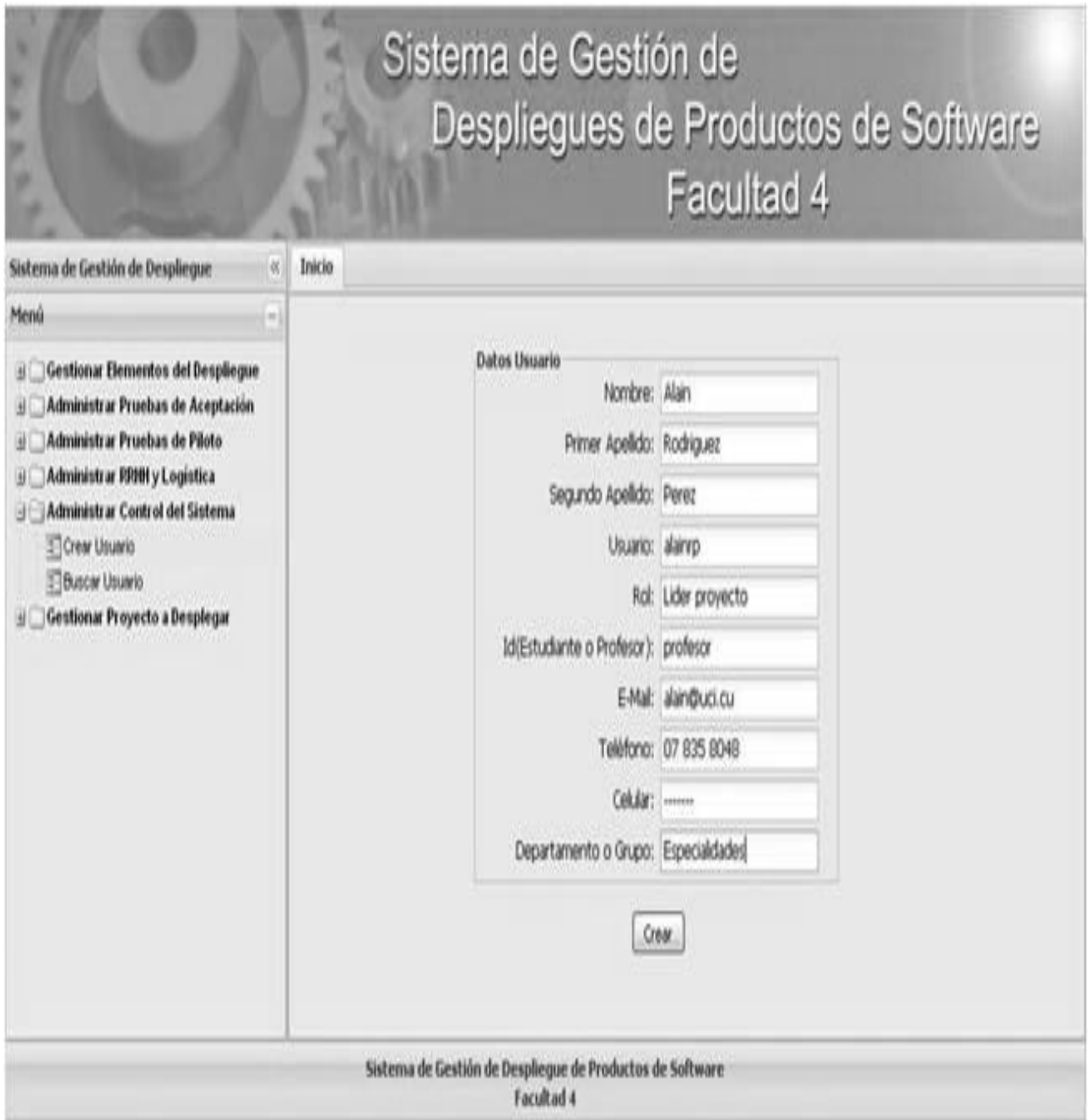


Ilustración 41: Prototipo No Funcional Interfaz Usuario Pantalla Administración Control del Sistema Escenario Crear Usuario: despliegue de productos de software de la Facultad 4.



Ilustración 42: Prototipo No Funcional Interfaz Usuario Pantalla Administración Control del Sistema Escenario Buscar Usuario: despliegue de productos de software de la Facultad 4.



Ilustración 43: Prototipo No Funcional Interfaz Usuario Pantalla Gestionar Proyecto a Desplegar Escenario Registrar Proyecto a Desplegar: despliegue de productos de software de la Facultad 4.

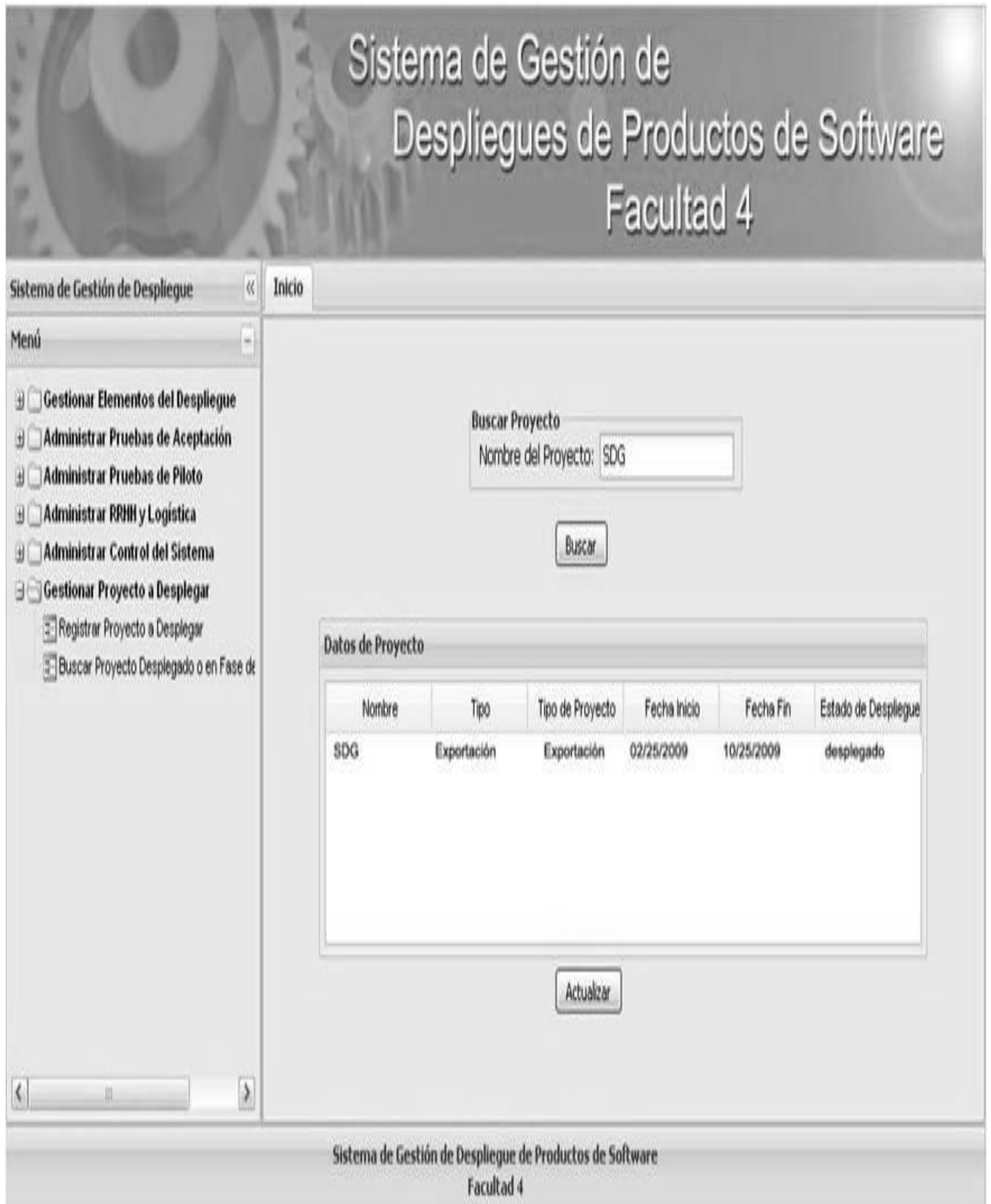


Ilustración 44: Prototipo No Funcional Interfaz Usuario Pantalla Gestionar Proyecto a Desplegar Escenario Buscar Proyecto a Desplegar o en Fase de Despliegue: despliegue de productos de software de la Facultad 4.

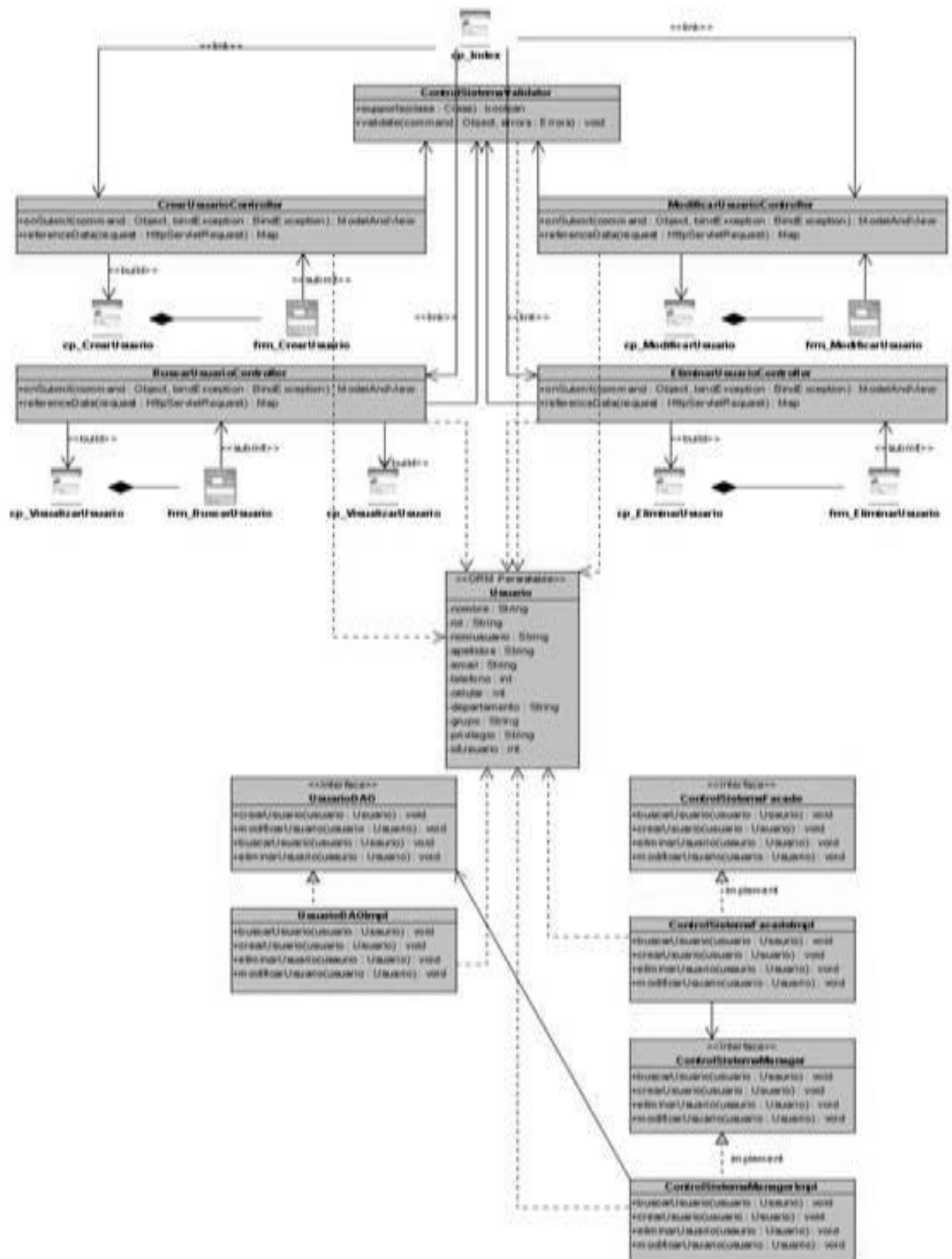


Ilustración 48: Diagrama de Clases del Diseño del Caso de Uso Gestionar Control del Sistema: despliegue de productos de software de la Facultad 4.

Anexo 6. Diagramas de Interacción (Diagramas de Secuencia).

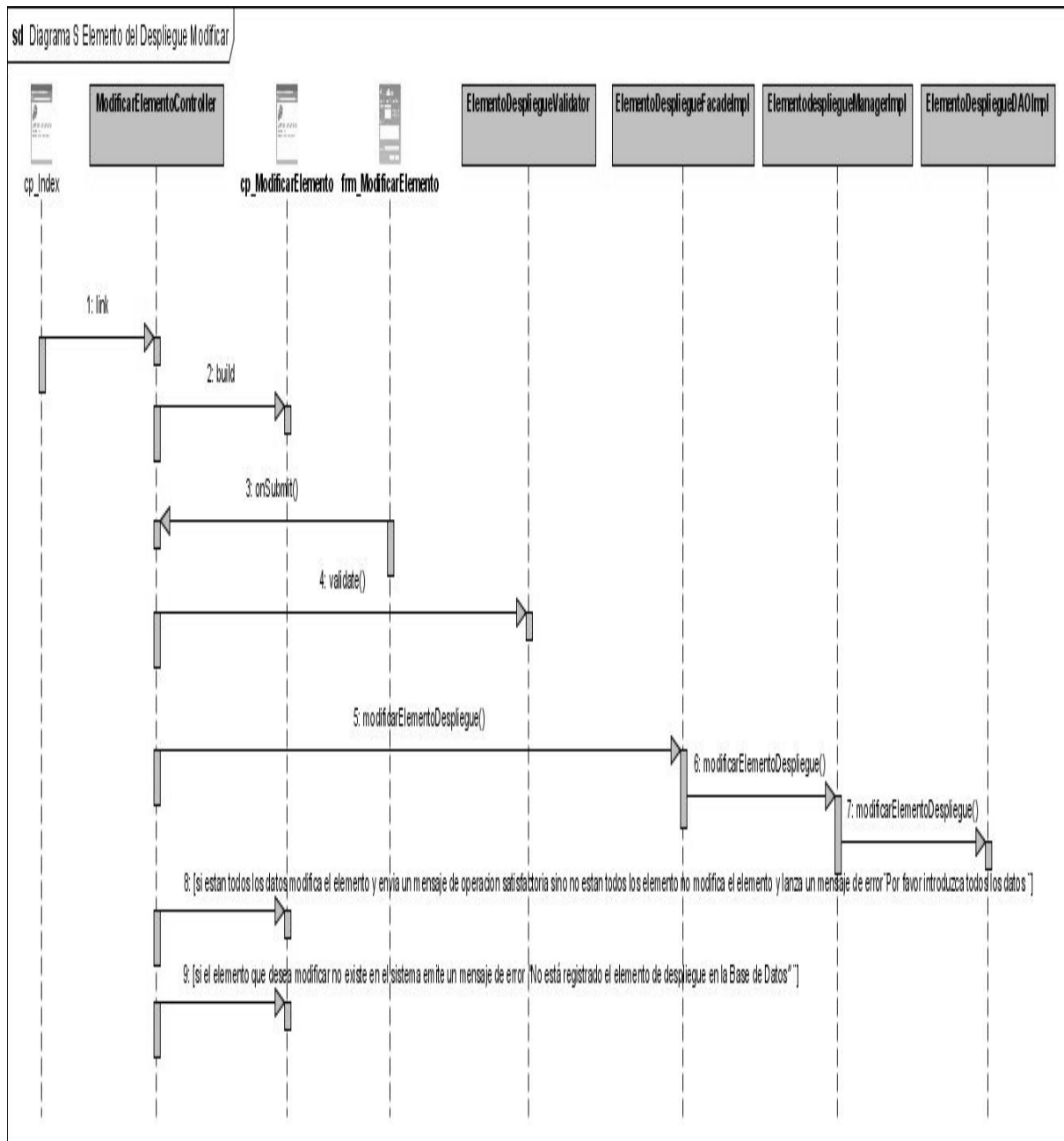


Ilustración 49: Diagrama de Secuencia del Caso de Uso Gestionar Elementos del Despliegue escenario Modificar Elemento del Despliegue: despliegue de productos de software de la Facultad 4.

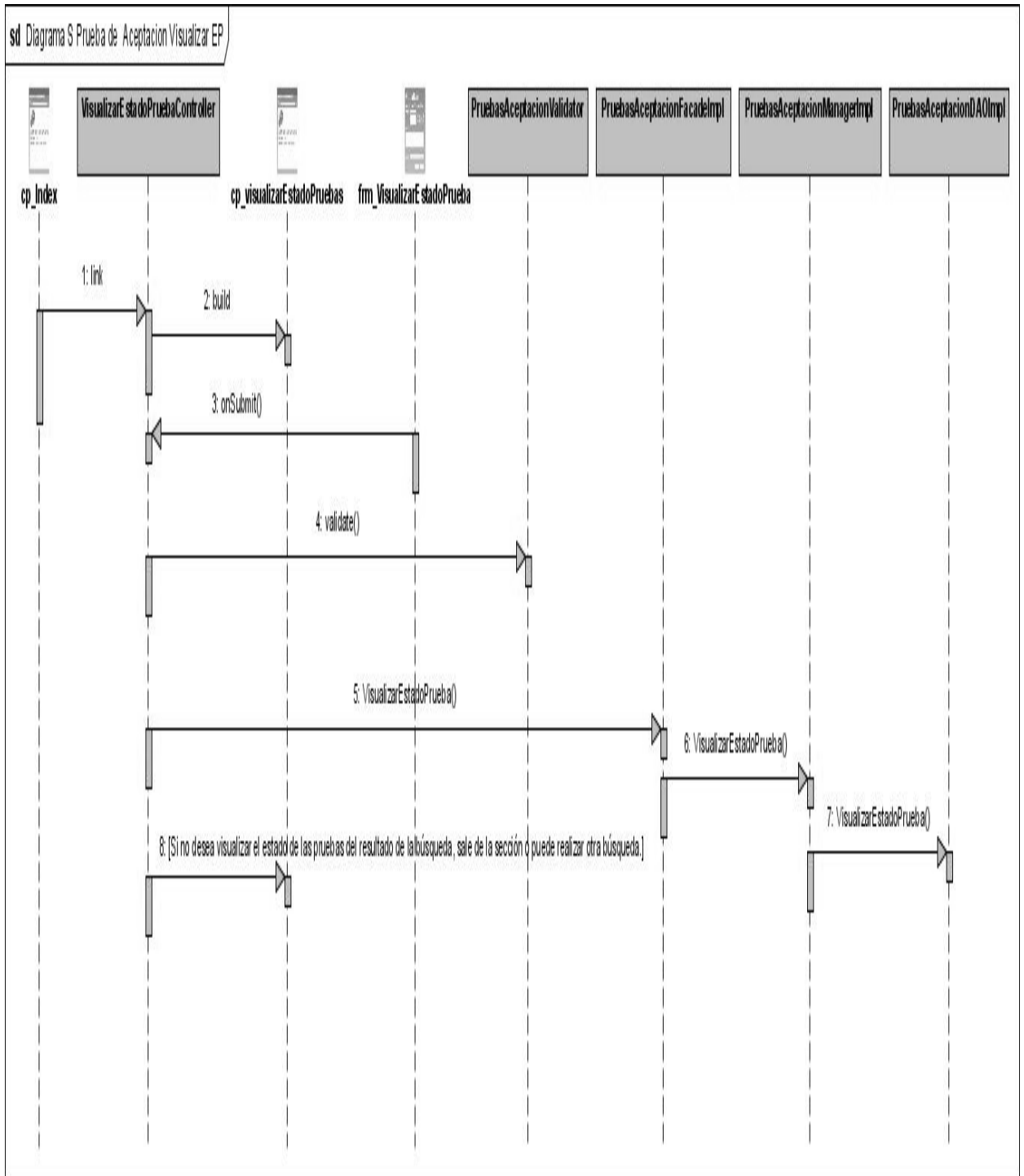


Ilustración 50: Diagrama de Secuencia del Caso de Uso Administrar prueba de Aceptación escenario Visualizar Estado del Despliegue: despliegue de productos de software de la Facultad 4.

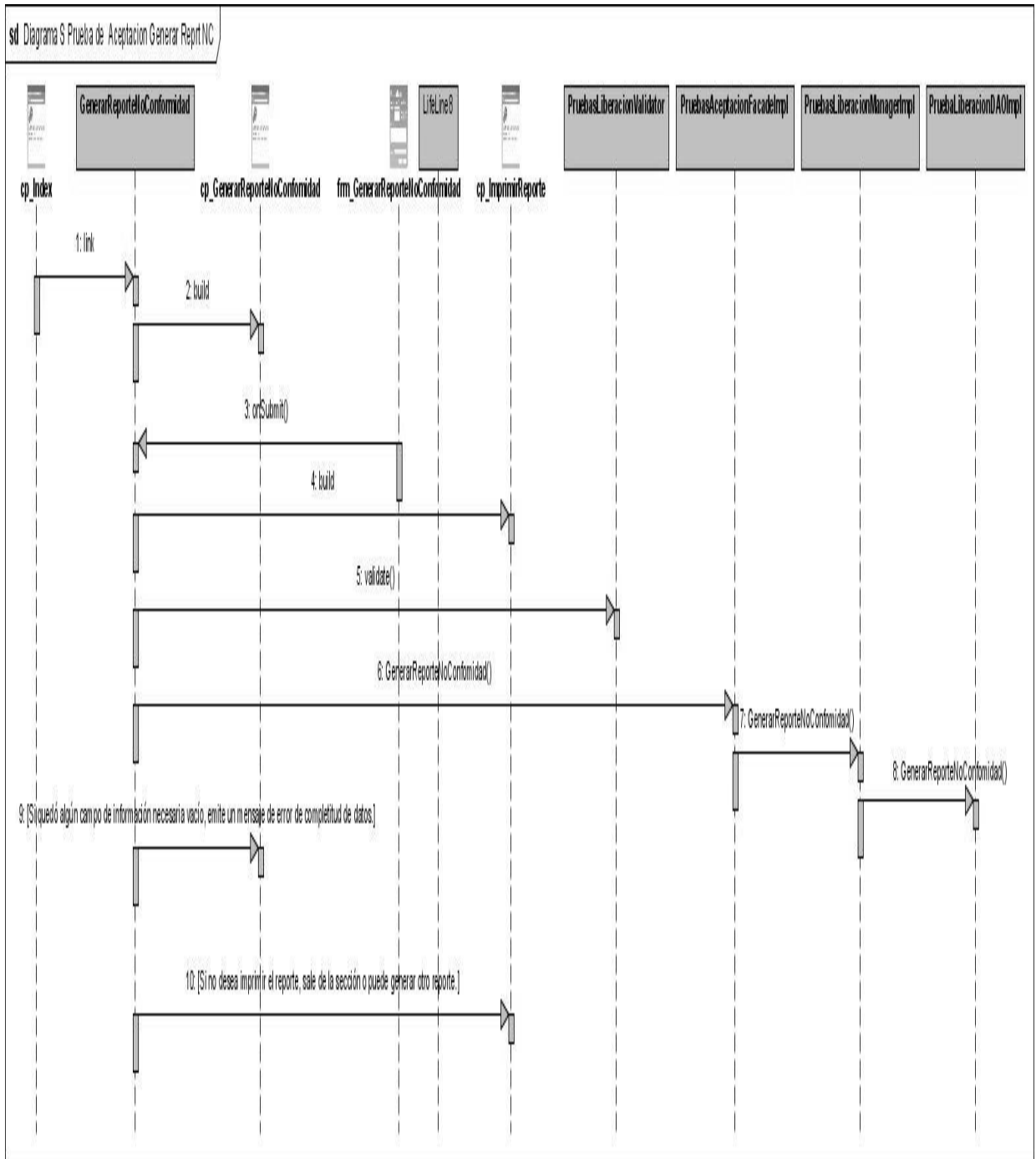


Ilustración 51: Diagrama de Secuencia del Caso de Uso Administrar prueba de Aceptación escenario Generar Reporte de No Conformidades: despliegue de productos de software de la Facultad 4.

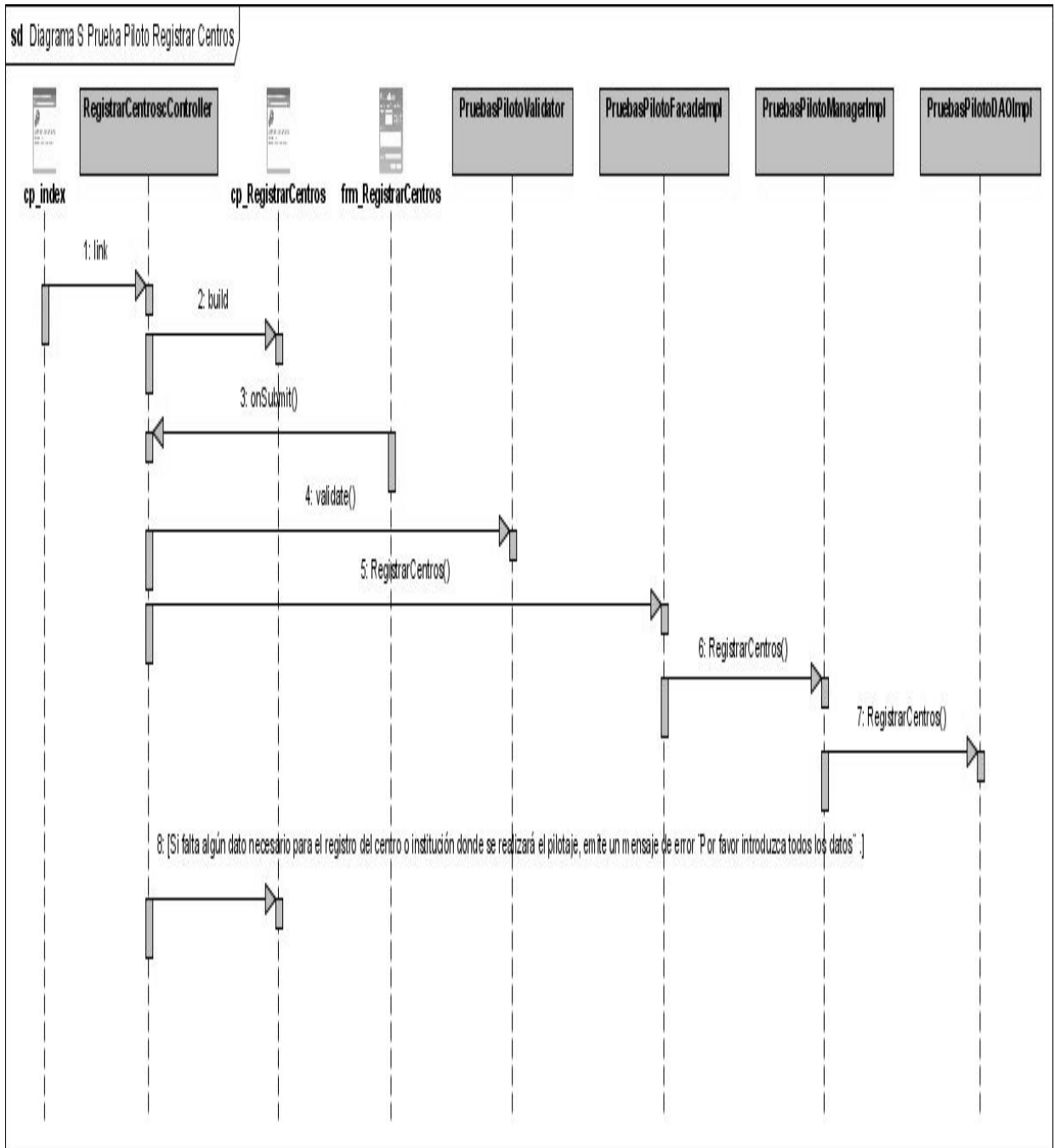


Ilustración 52: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Registrar Centros: despliegue de productos de software de la Facultad 4.

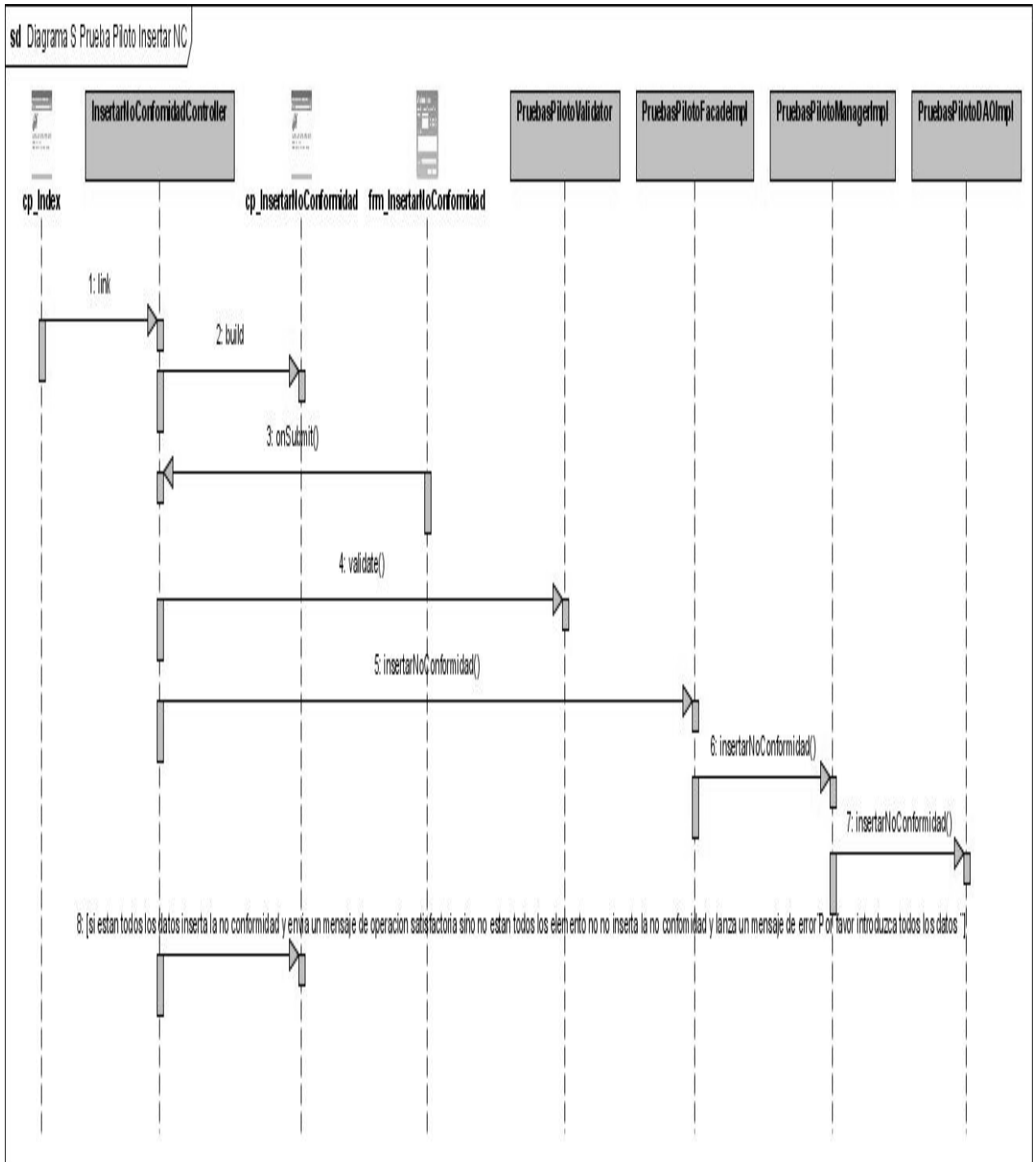


Ilustración 53: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Registrar No Conformidad: despliegue de productos de software de la Facultad 4.

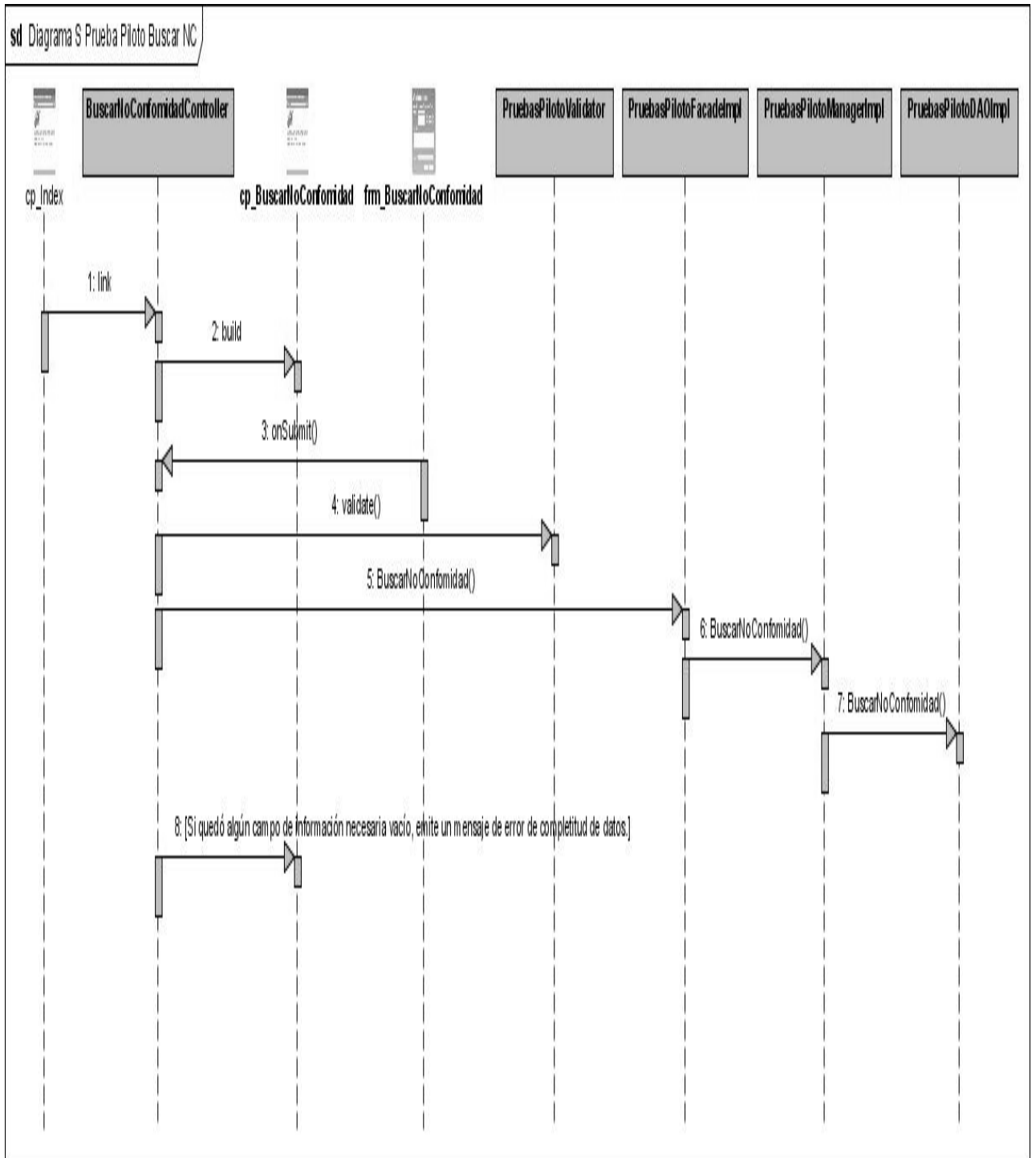


Ilustración 54: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Buscar No Conformidad: despliegue de productos de software de la Facultad 4.

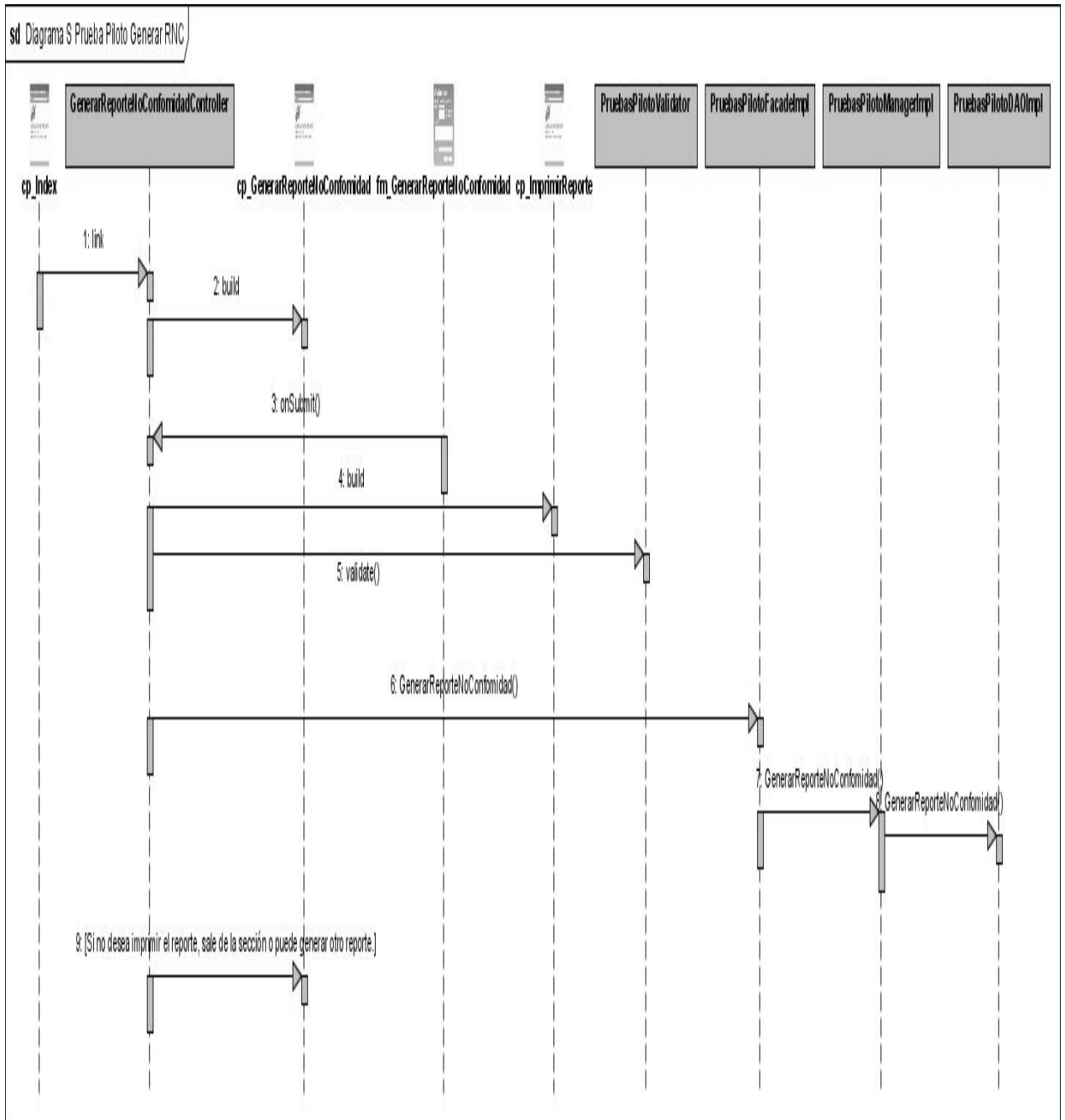


Ilustración 55: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Generar Reporte No Conformidad: despliegue de productos de software de la Facultad 4.

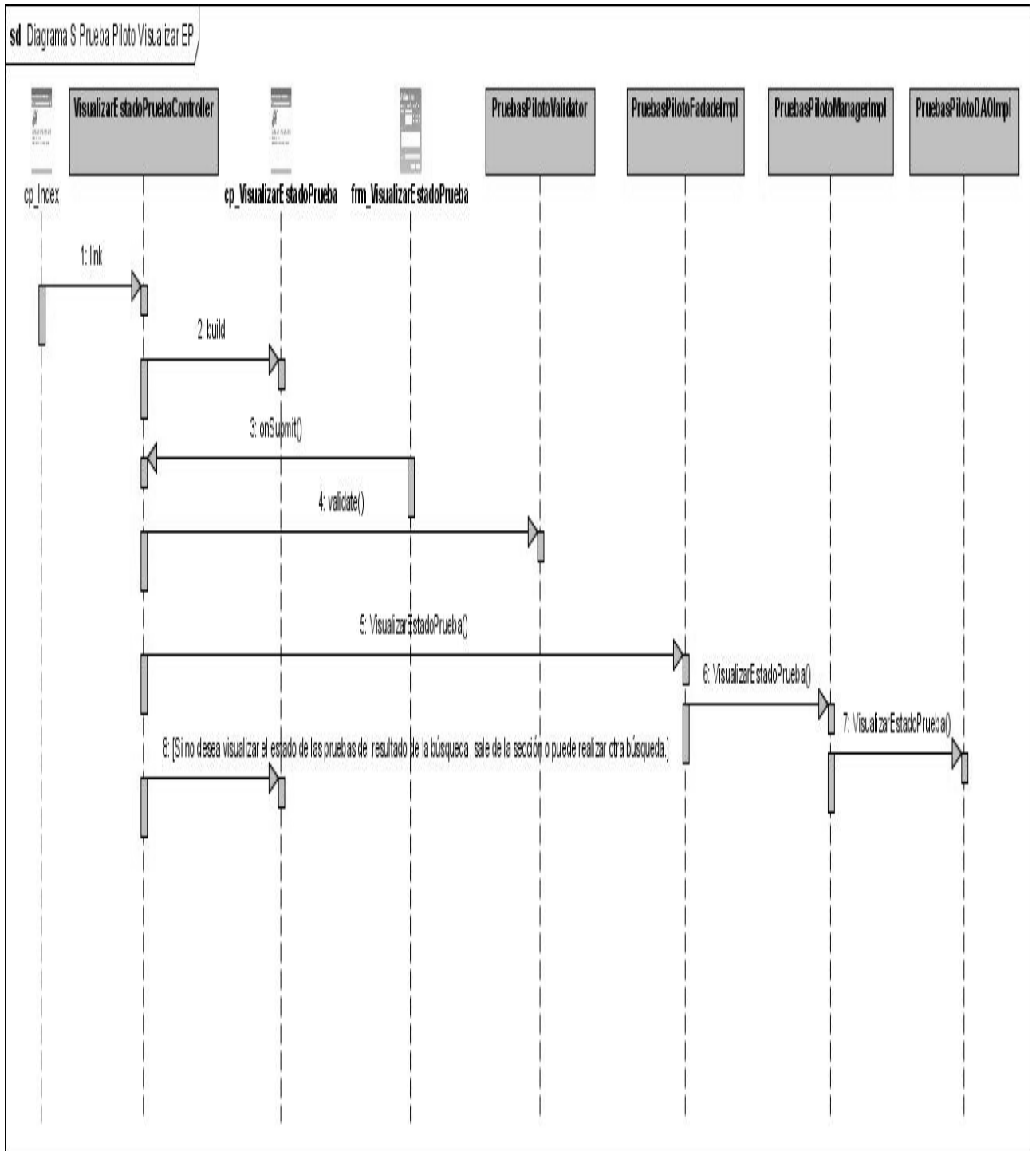


Ilustración 56: Diagrama de Secuencia del Caso de Uso Administrar Proceso de Pruebas Piloto escenario Visualizar Estado de las Pruebas: despliegue de productos de software de la Facultad 4.

Anexos.

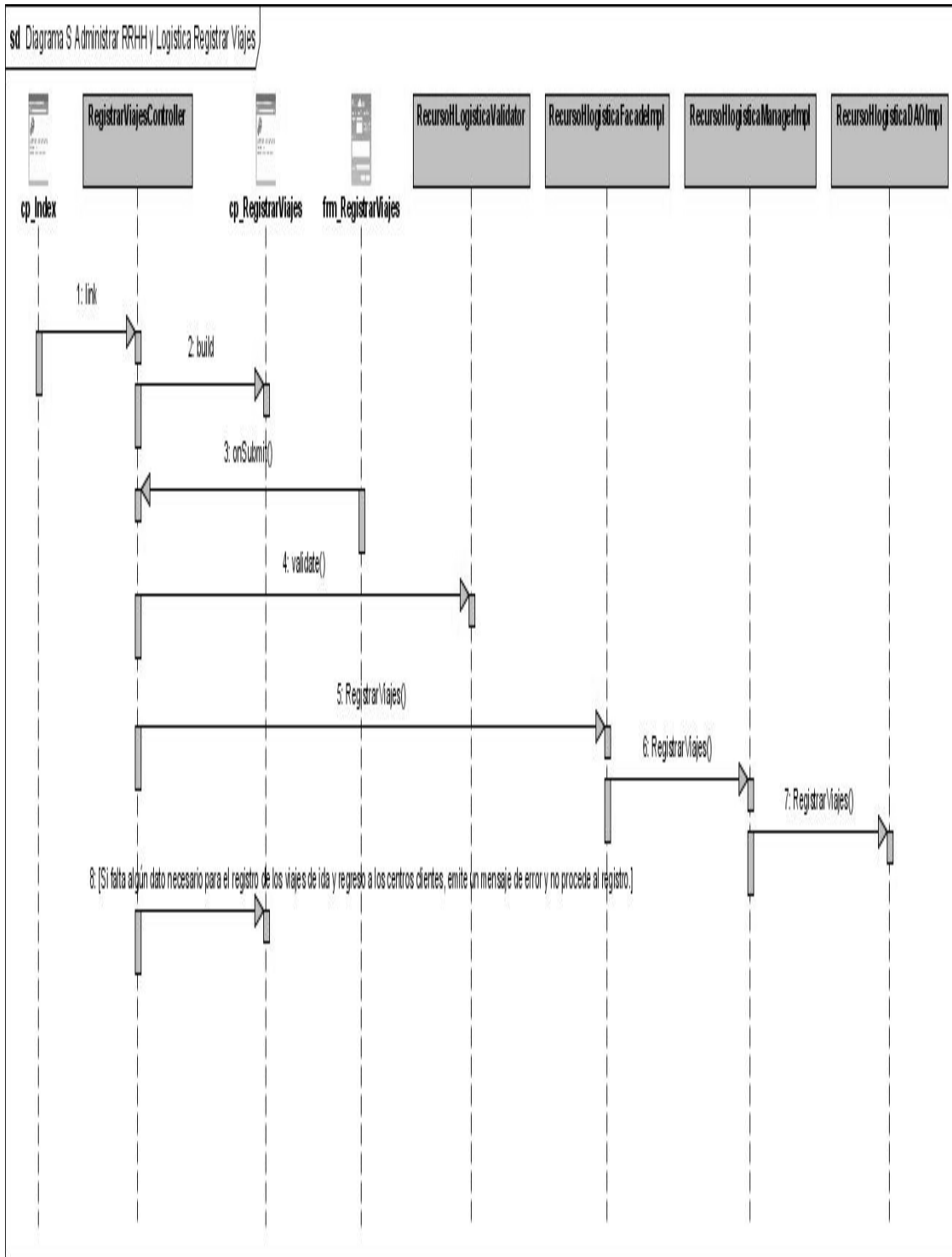


Ilustración 57: Diagrama de Secuencia del Caso de Uso Administrar RR.HH y Logística escenario Registrar Viajes: despliegue de productos de software de la Facultad 4.

Anexos.

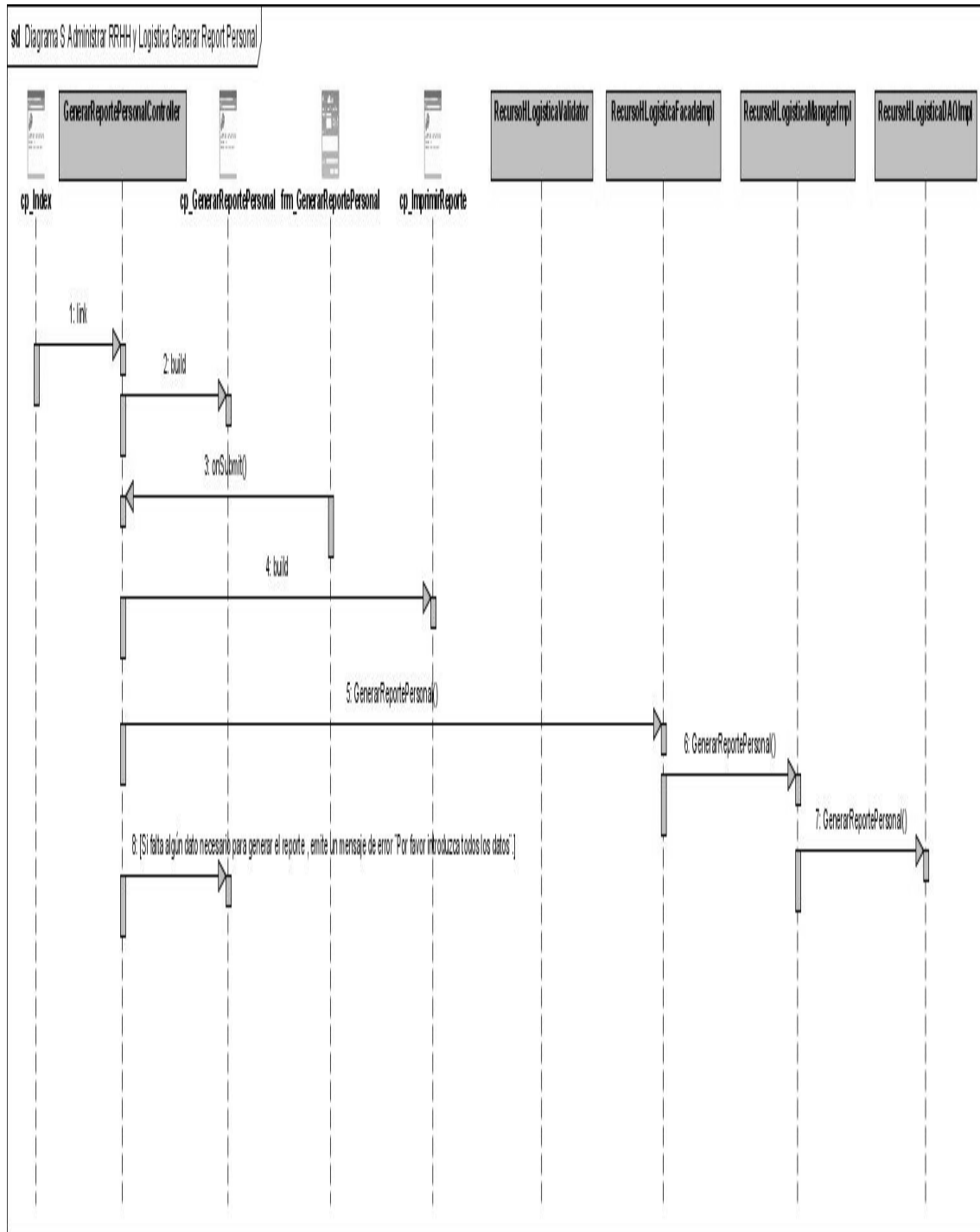


Ilustración 58: Diagrama de Secuencia del Caso de Uso Administrar RR.HH y Logística escenario Registrar Personal: despliegue de productos de software de la Facultad 4.

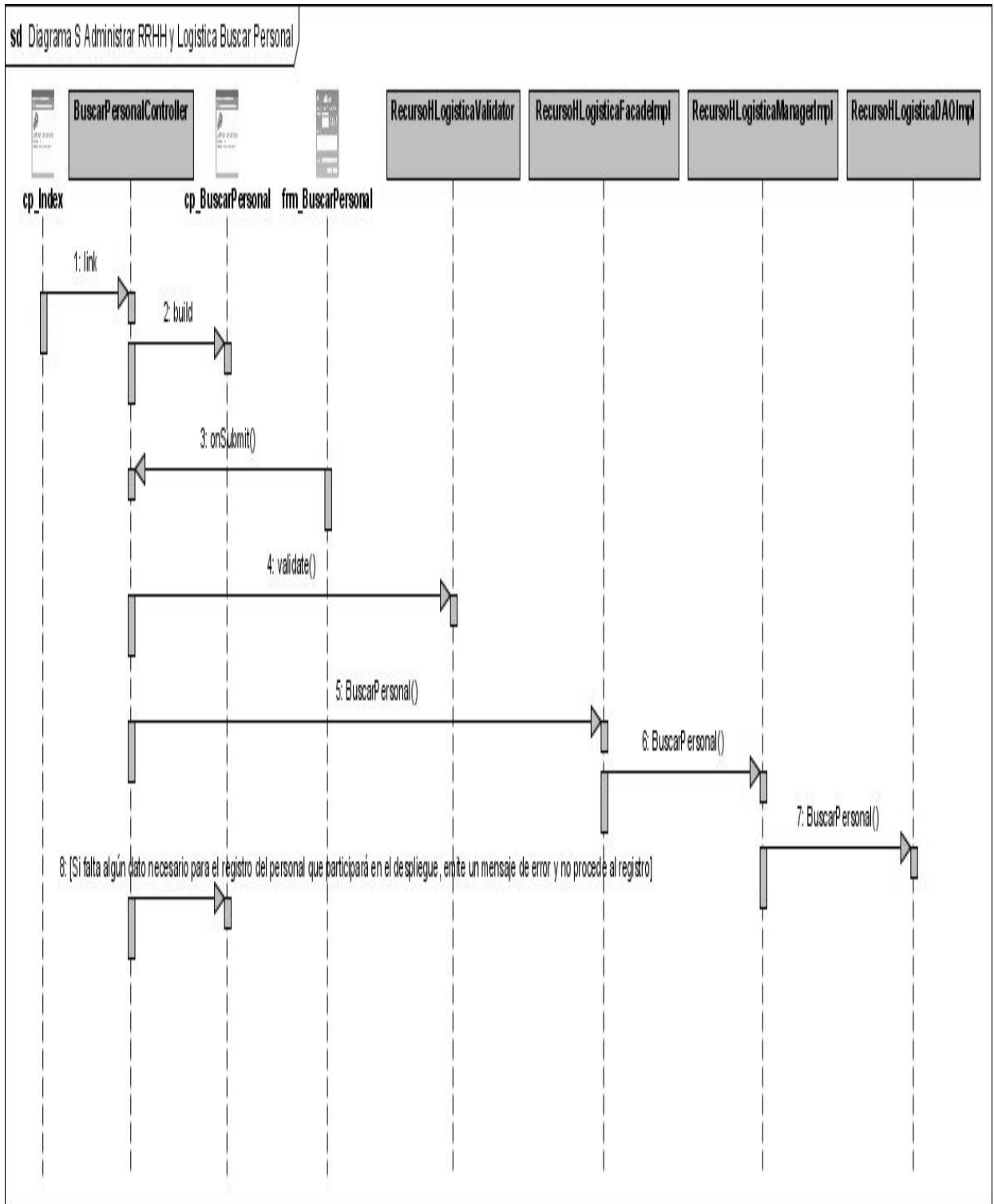


Ilustración 59: Diagrama de Secuencia del Caso de Uso Administrar RR.HH y Logística escenario Buscar Personal: despliegue de productos de software de la Facultad 4.

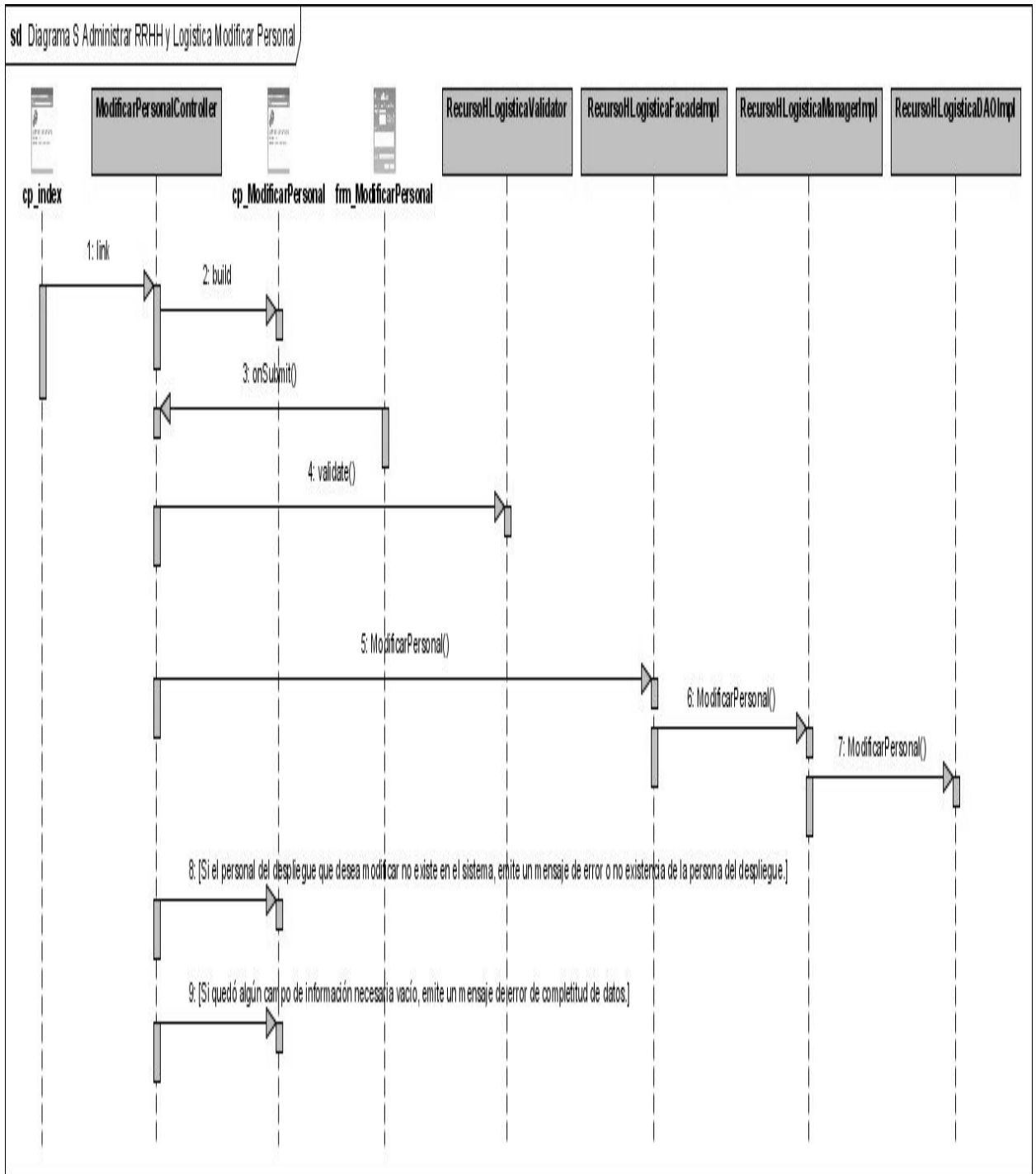


Ilustración 60: Diagrama de Secuencia del Caso de Uso Administrar RR.HH y Logística escenario Modificar Personal: despliegue de productos de software de la Facultad 4.

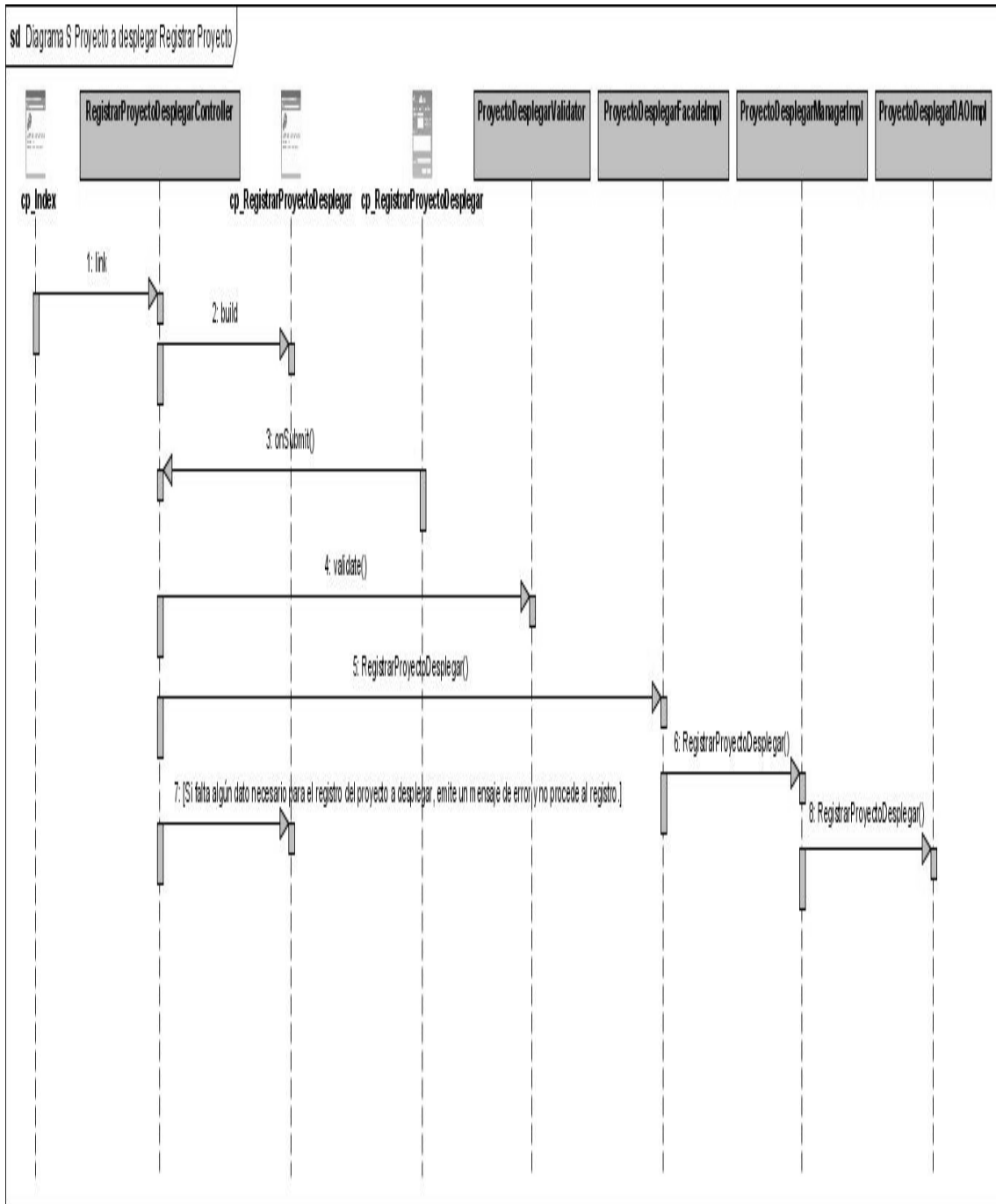


Ilustración 61: Diagrama de Secuencia del Caso de Uso Gestionar Proyecto a Desplegar escenario Registrar Proyecto: despliegue de productos de software de la Facultad 4.

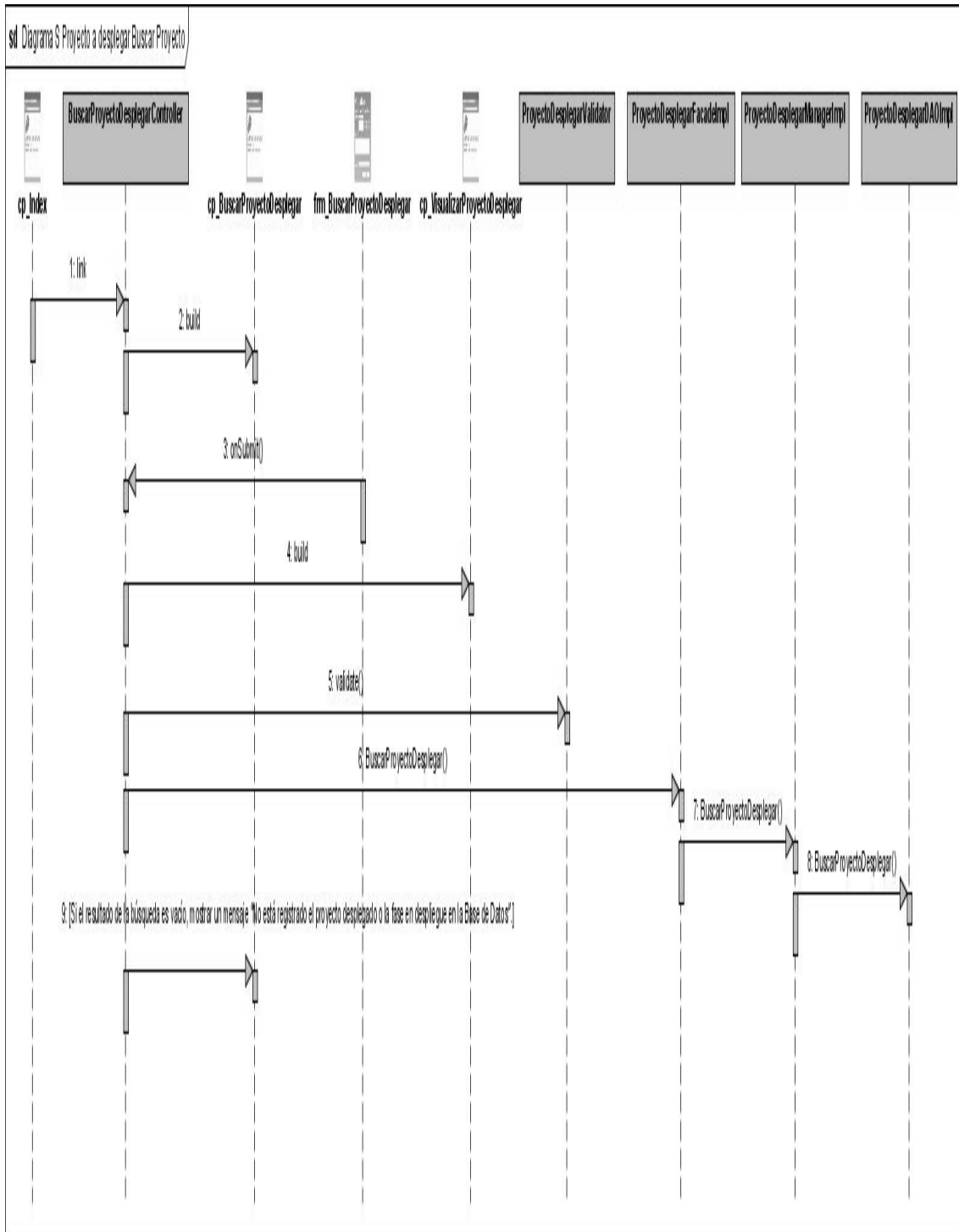


Ilustración 62: Diagrama de Secuencia del Caso de Uso Gestionar Proyecto a Desplegar escenario Buscar Proyecto: despliegue de productos de software de la Facultad 4.

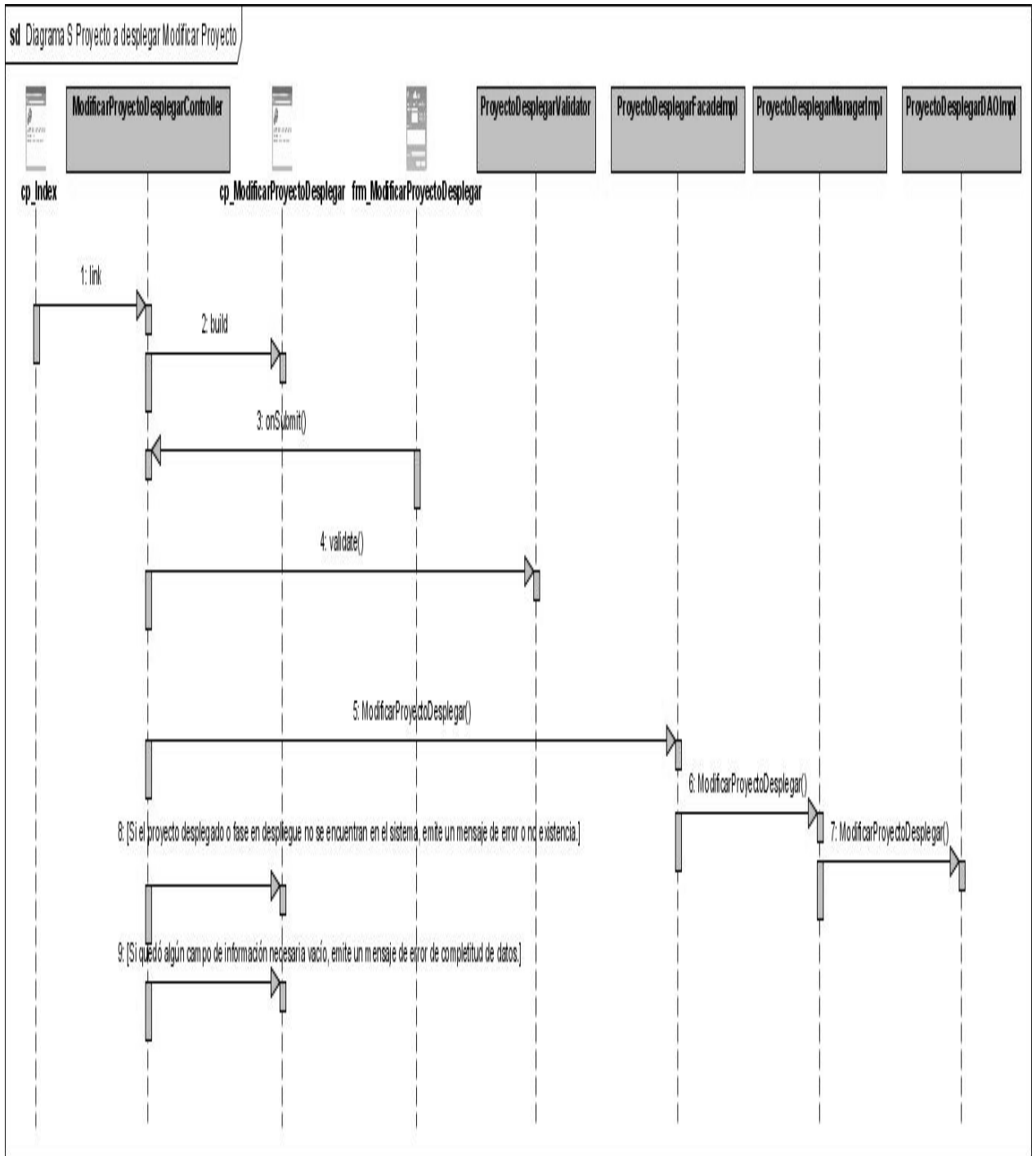


Ilustración 63: Diagrama de Secuencia del Caso de Uso Gestionar Proyecto a Desplegar escenario Modificar Proyecto: despliegue de productos de software de la Facultad 4.

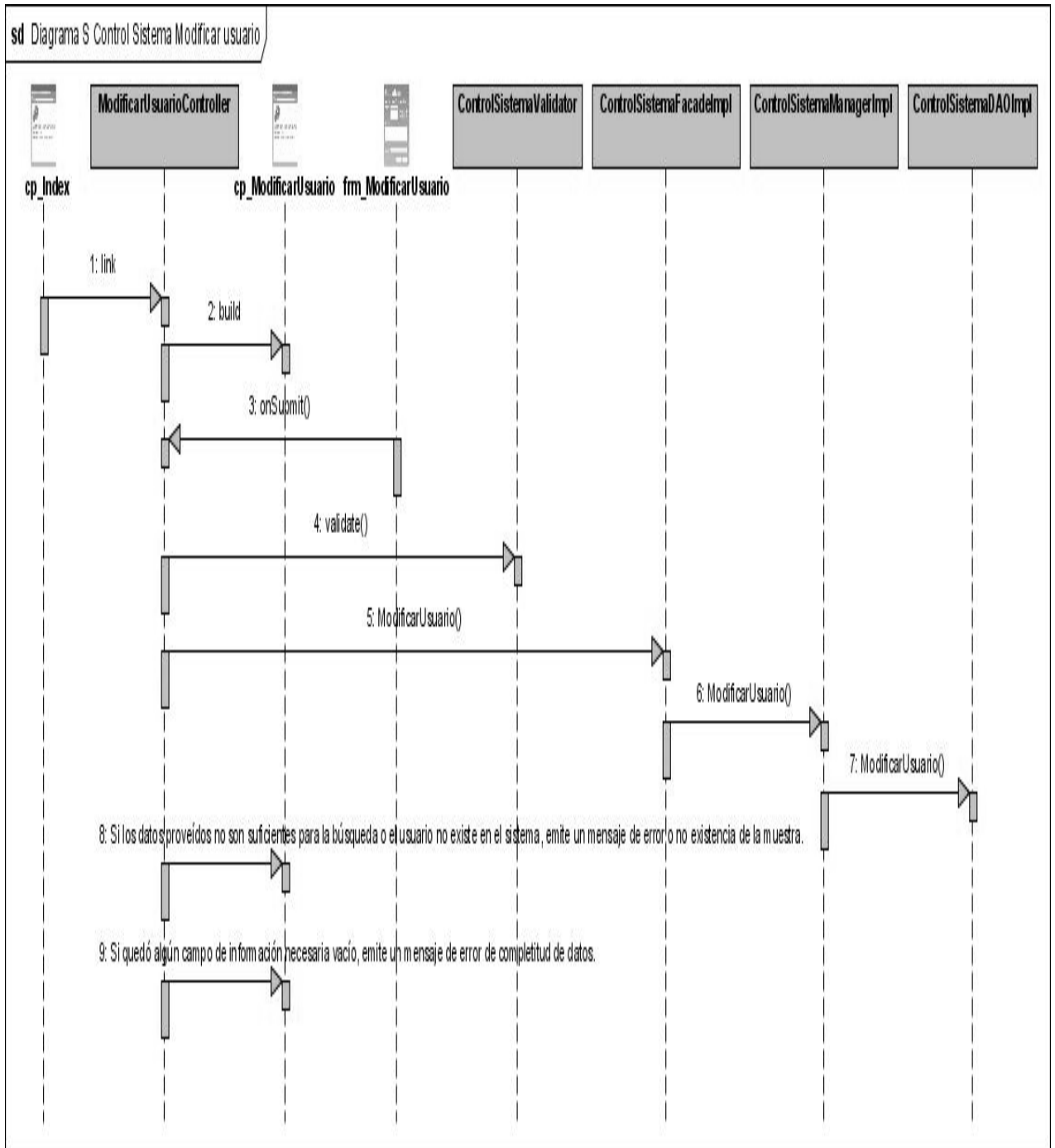


Ilustración 64: Diagrama de Secuencia del Caso de Uso Gestionar Control del Sistema escenario Modificar Usuario: despliegue de productos de software de la Facultad 4.

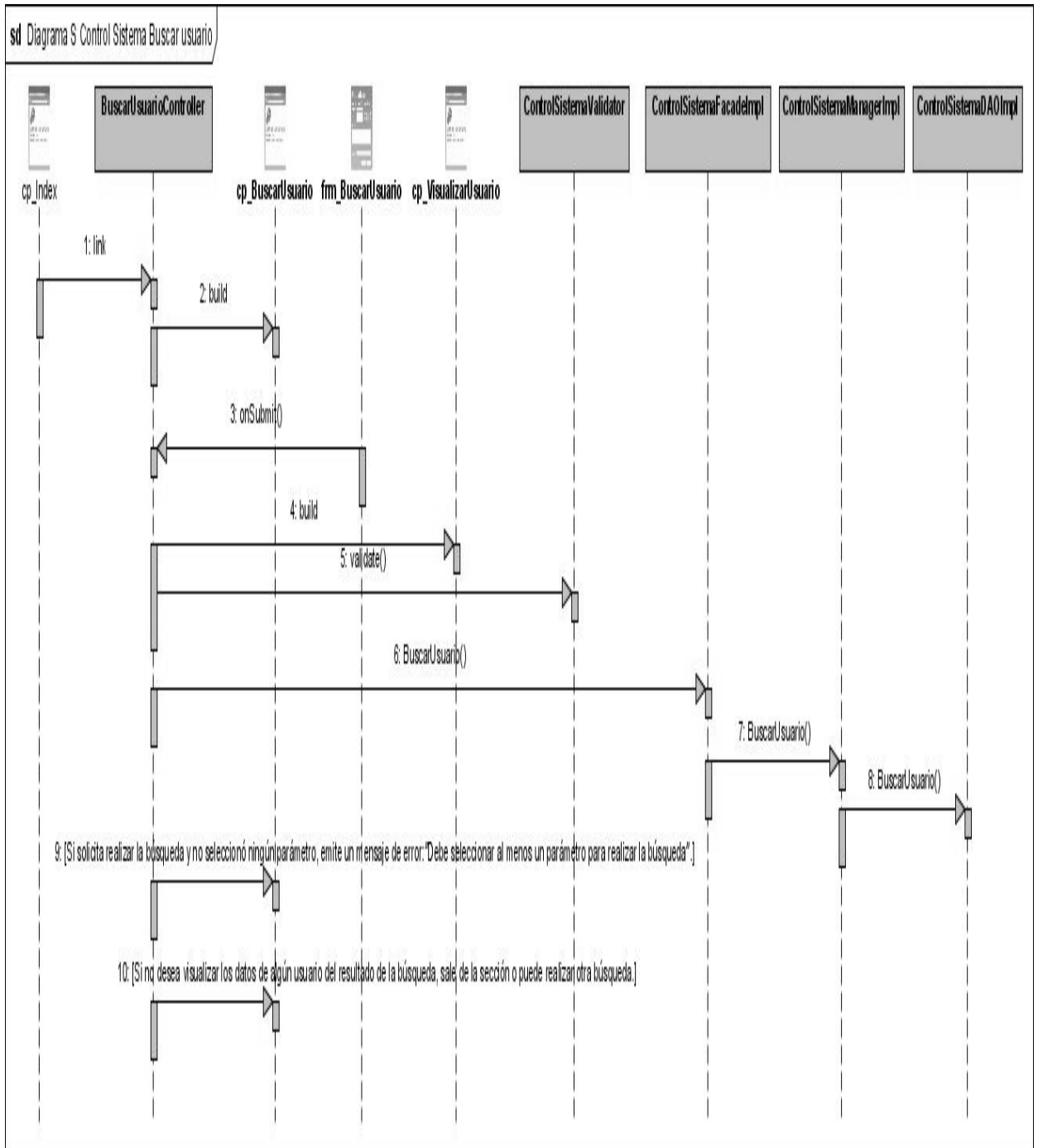


Ilustración 65: Diagrama de Secuencia del Caso de Uso Gestionar Control del Sistema escenario Buscar Usuario: despliegue de productos de software de la Facultad 4.

Anexos.

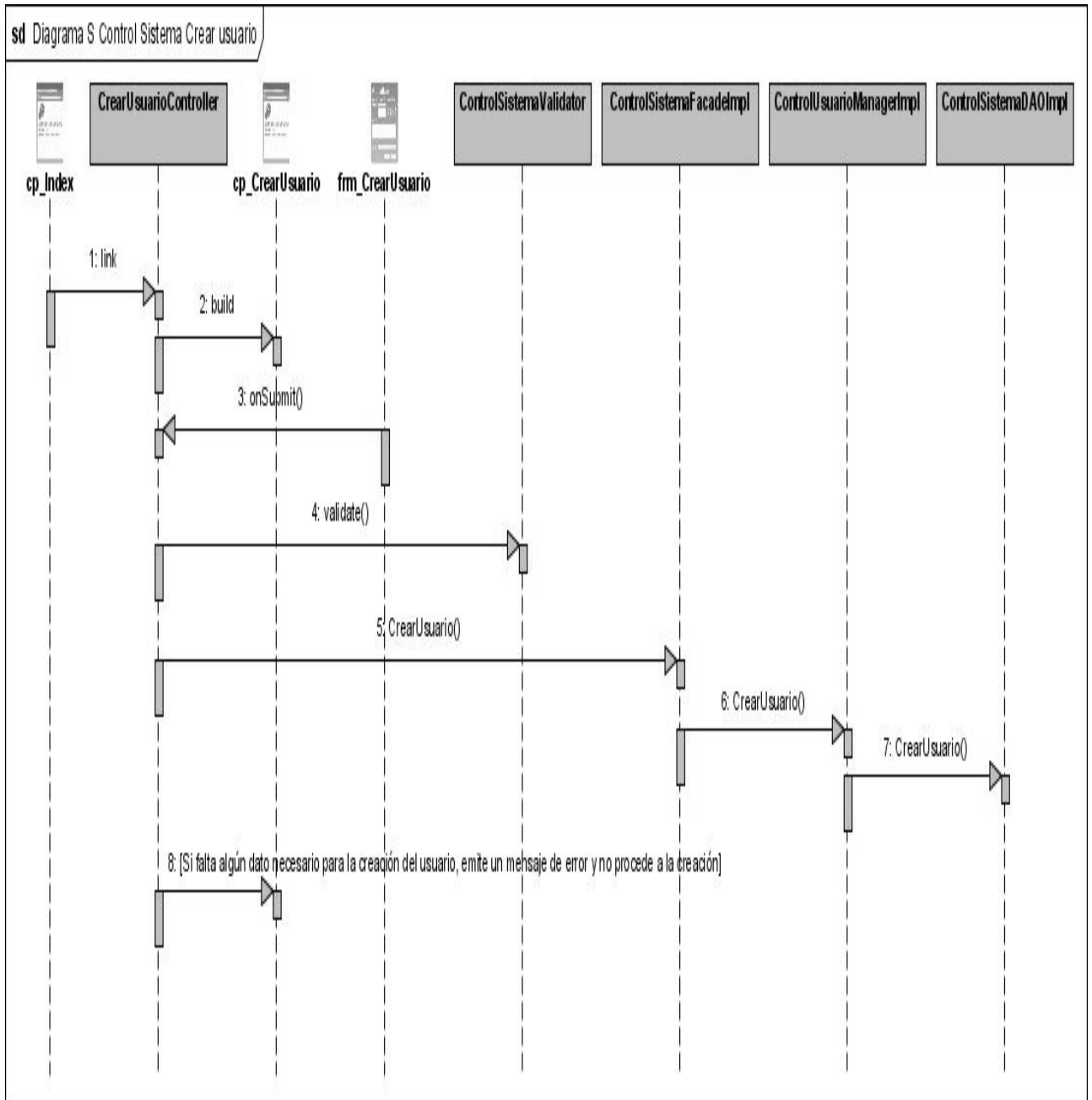


Ilustración 66: Diagrama de Secuencia del Caso de Uso Gestionar Control del Sistema escenario Crear Usuario: despliegue de productos de software de la Facultad 4.

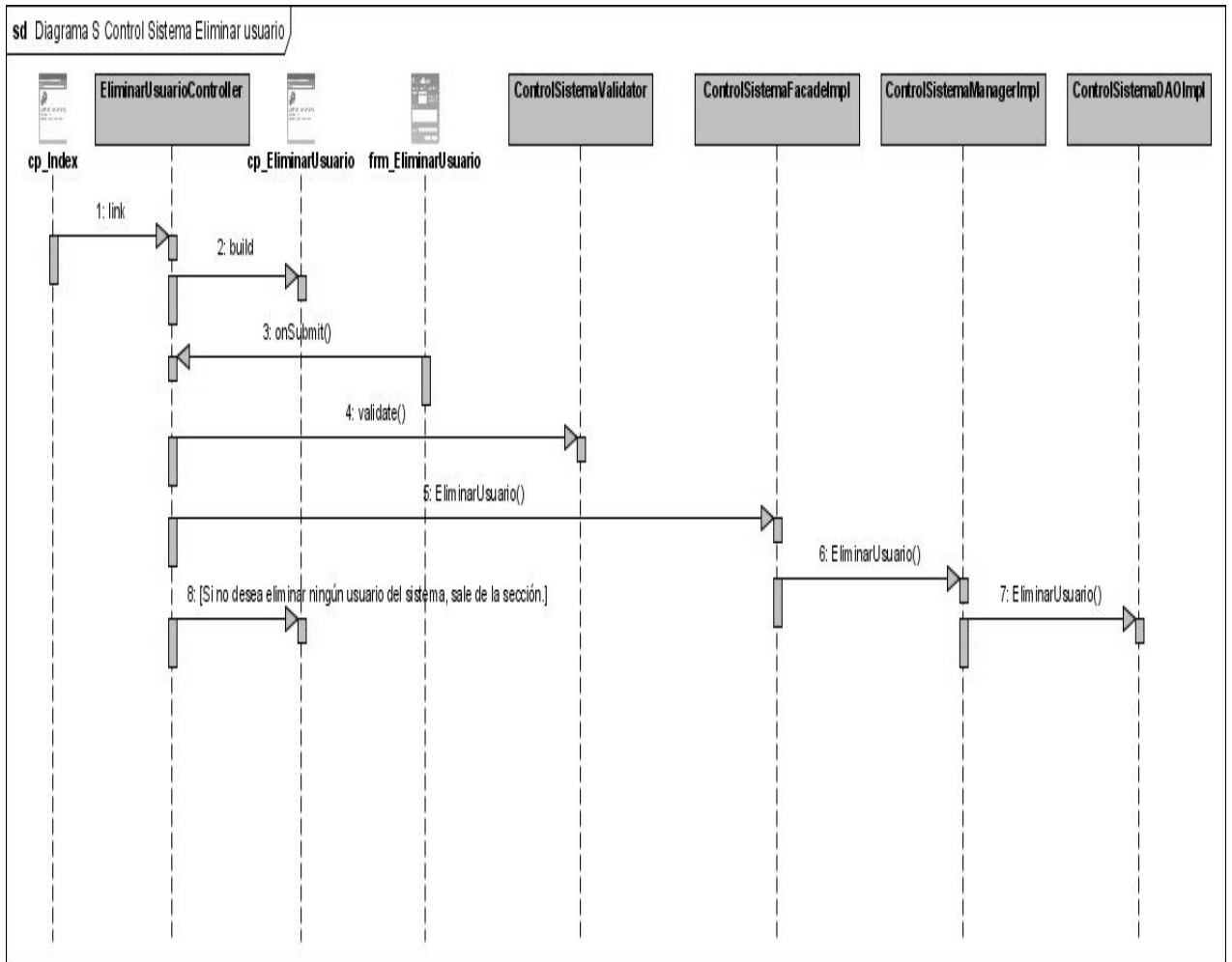


Ilustración 67: Diagrama de Secuencia del Caso de Uso Gestionar Control del Sistema escenario Eliminar Usuario: despliegue de productos de software de la Facultad 4.

Anexo 7. Encuesta realizada a estudiantes y profesores que han participado en despliegue de producto de software.

ENCUESTA

Estimado(a) compañero(a):

La dirección de producción de la Facultad 4 está desarrollando un sistema informático para la gestión del despliegue de sus productos de software (nacionales y de exportación) y como parte del proceso de desarrollo de dicho sistema queremos obtener la experiencia suya en esta área para enriquecer nuestra investigación y poder desarrollar un sistema que no sólo sirva a nuestra Facultad, sino que lo desarrollemos con perspectivas de ser usado por cualquier productor de software de nuestra Universidad.

Anexos.

Preguntas:

1. ¿Qué tipo de producto desplegó? Marque con una X

Producto de software___ Software a la medida para un cliente___
Componente de software___

2. ¿Qué alcance tiene el proyecto que Ud. ha desplegado (o participado en el proceso de despliegue indirectamente)? Marque con una X

Nacional___ Exportación___

3. Nos podría decir cuáles elementos controló durante dicho despliegue:

Anexos.

- a. _____
- b. _____
- c. _____
- d. _____

3. ¿Qué información considera Ud. no debe faltar en un sistema de gestión de despliegue de proyectos informáticos?

- a. _____
- b. _____
- c. _____
- d. _____

4. ¿La gestión del despliegue de los proyectos productivos en su área (facultad, polo, proyecto) se guía por algún estándar o metodología? Marque con una X

SI__ NO__

En caso positivo, podría decirnos cuál (metodología/estándar)
