

SISTEMA DE INFORMACIÓN PARA LA GESTIÓN DE PROYECTOS DE
GRADO DEL PROGRAMA INGENIERÍA INDUSTRIAL DE LA UNIVERSIDAD
CATÓLICA DE PEREIRA

PRESENTADO POR
DIEGO MAURICIO HENAO ZULETA

UNIVERSIDAD CATÓLICA DE PEREIRA
PROGRAMA INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES
FACULTAD DE CIENCIAS BÁSICAS E INGENIERÍA
2019

SISTEMA DE INFORMACIÓN PARA LA GESTIÓN DE PROYECTOS DE
GRADO DEL PROGRAMA INGENIERÍA INDUSTRIAL DE LA UNIVERSIDAD
CATÓLICA DE PEREIRA

DIEGO MAURICIO HENAO ZULETA

PROYECTO DE GRADO

DIRECTOR DEL PROYECTO
JUAN CARLOS BLANDON ANDRADE, M.Sc., Ph.D.

UNIVERSIDAD CATÓLICA DE PEREIRA
PROGRAMA DE INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES
FACULTAD DE CIENCIAS BASICAS E INGENIERÍA
2019

DECLARACIÓN DE DERECHOS DE AUTOR

Como estudiante de la Universidad Católica de Pereira del programa de Ingeniería de Sistemas y Telecomunicaciones, declaro que este proyecto fue realización propia, resultado de un estudio hecho a conciencia, en el cual se vio la necesidad de desarrollar un Sistema de Información para la Gestión de Proyectos de grado de la carrera de Ingeniería Industrial de la Universidad Católica de Pereira.

AGRADECIMIENTOS

Ante todo quiero dar las gracias a mi madre por su ayuda y comprensión, también quiero agradecer a mis compañeros por aportar un poco cada día de su tiempo y palabras que ayudaron a mi carrera como profesional y persona. A la universidad Católica de Pereira por ser el alma mater de mis padres y ahora como alumno y a puertas de ser un profesional.

DEDICATORIA

Dedico este proyecto a mi madre (María Teresa Zuleta García) quien me brindó su apoyo para la realización de este proceso, ya que de una u otra manera colaboró con su paciencia y amor a la culminación de mi proyecto de grado, pero quien me inspiró y me motivo para la consecución de mi carrera como Ingeniero de Sistemas y Telecomunicaciones.

RESUMEN

El programa de Ingeniería Industrial de la Universidad Católica de Pereira, tiene como opción de grado la modalidad proyecto de grado, de la cual se administra gran cantidad de información, que corresponde a los trabajos de grado de los estudiantes, los profesores que los dirigen y la trazabilidad del proyecto. La literatura encontrada destaca la importancia del desarrollo de este tipo de sistemas, porque ayuda a mejorar el proceso de organización de los proyectos. Dado que la información se maneja actualmente de forma manual, es necesaria una mejor gestión del proceso del proyecto de grado. Se propone crear un sistema de información en un entorno web, utilizando el modelo de desarrollo en cascada y el IDE Visual Studio con lenguaje de programación C#. Se pretende mejorar en gran medida la gestión de los procesos, para poder generar informes pertinentes en corto tiempo.

PALABRAS CLAVE: Aplicación Web, Herramienta de Programación, Ingeniería de Software, Sistema de Información, Proyectos de Grado.

ABSTRACT

The program of Industrial Engineering of the Catholic University of Pereira, has as option of degree the modality project of degree, of which a great amount of information is administered, that correspond to the works of degree of the students, the professors who direct them and the traceability of the project. The literature found highlights the importance of the development of this type of systems, because it helps to improve the process of project organization. Given that information is currently handled manually, it is proposed to create an information system in a web environment, using the cascade development model and the Visual Studio IDE with C# programming language. It is intended to greatly improve the management of processes, in order to generate relevant reports in a short time.

KEYWORDS: Application Web, Information System, Programming Tool, Degree Project, Software Engineering.

TABLA DE CONTENIDO

TABLA DE CONTENIDO	7
LISTA DE FIGURAS	10
LISTA DE TABLAS.	13
INTRODUCCION	15
1.1. DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA.....	16
1.1.1. Situación Problemática.....	16
1.2. OBJETIVOS.....	16
1.2.1. Objetivo General.....	16
1.2.2. Objetivos Específicos	16
1.3. JUSTIFICACIÓN.....	16
1.4. CRONOGRAMA	17
2. MARCO CONTEXTUAL.....	18
3. MARCO TEORICO	19
3.1. Antecedentes	19
3.1.1. Sistema de Información para la gestión de proyectos de grado para la Universidad Minuto de Dios (Siap).	19
3.1.2. Sistema de Información para la Gestión de los Trabajos de Grado (Universidad de Cartagena).	19
3.1.3. Gestión de Proyectos de Final de Grado (Universidad de Barcelona).	19
3.1.4. Prototipo de una Aplicación Web para la Gestión en el Proceso de Trabajos de Grado de la Universidad Ean.....	20
3.2. Antecedentes Teóricos	20
3.2.1. Modelo en Cascada.....	20
3.2.2 Ciclo de Vida V	22
3.2.3. Modelo en Espiral.....	23
3.2.4. Modelo de desarrollo Evolutivo o Prototipo	24
3.2.5. Metodología de Desarrollo Rápido de Aplicaciones (RAD)	26
3.2.6. Metodologías Agiles.....	27
3.2.6.1. Metodología SCRUM.....	27

3.2.6.2. Metodología de Programación Extrema	28
3.2.7. Arquitectura MVC.....	30
3.2.7.1. ¿Qué es Asp.Net?	31
3.2.7.2. ¿Qué es Entity Framework?	31
3.2.7.3. ¿Qué es Visual Studio?	32
3.2.8. ¿Qué son Lenguajes de Programación?.....	32
3.2.8.1. ¿Qué es Java?	32
3.2.8.2. ¿Qué es PHP?	32
3.2.8.3. ¿Qué es JavaScript?	33
4. MODELO TEORICO	34
4.1. APORTE TEORICO Y PRÁCTICO	34
4.1.1. Proceso de Ingeniería de Software	35
4.1.2. Requisitos.....	36
4.1.3. Diseño	38
4.1.4. Módulos y Funcionalidades del Sistema de Información para la Gestión de Proyectos de Grado	38
4.1.5. Requisitos Específicos.	42
4.2. Listado de Requerimientos	44
4.3. Requisitos No Funcionales	51
4.3.1. Análisis	52
4.4. Diagrama de Flujo.....	52
4.4.1. Descripción del Diagrama de Flujo.....	52
4.5. Casos de Uso	54
4.5.1. Descripción de Casos de Uso.	60
4.6. Diseño.....	85
4.6.1. Modelo Relacional	85
4.6.2. Base de Datos de SQL Server Management.	86
4.6.3. Diagrama de Secuencias	87
4.6.4. Arquitectura	98
4.6.5. Diagrama de Clases.	99
4.6.6. Interfaz Gráfica.....	100
4.7. CONCRECIÓN DEL MODELO	105

4.7.1. IMPLEMENTACIÓN.....	105
4.7.2. PRUEBAS.....	111
4.7.2.1. ¿Qué son Pruebas unitarias?.....	112
4.7.2.2. Prueba de Integración Descendente.....	125
4.7.2.3. Diseño de Casos de Prueba del Sistema.....	126
4.7.2.4. Bugs del Sistema.....	127
4.7.3. MANTENIMIENTO.....	129
5. CONCLUSIONES.....	130
6. RECOMENDACIONES.....	131
7. REFERENCIA.....	132
8. ANEXOS.....	134
8.1. Manual del Usuario.....	134

LISTA DE FIGURAS

Figura 1: Modelo en Cascada.....	20
Figura 2: Ciclo de Vida en V.....	22
Figura 3. Modelo en Espiral.	23
Figura 4. Modelo de desarrollo Evolutivo.....	25
Figura 5: Metodología RAD.....	26
Figura 6. Metodología SCRUM.....	28
Figura 7: Metodología XP.....	29
Figura 8: MVC.....	30
Figura 9. Entity Framework	31
Figura 10: Diagrama de Flujo del Sistema.....	53
Figura 11: Caso de Uso General.	54
Figura 12: Caso de Uso Registrar Usuario.....	55
Figura 13: Caso de Uso Gestionar Proyecto.....	55
Figura 14: Caso de Uso Gestionar Actas.....	56
Figura 15: Caso de Uso de Gestionar Actas.....	56
Figura 16: Caso De Uso Gestionar Modalidad.....	57
Figura 17: Caso de Uso Gestionar Línea de Investigación.....	57
Figura 18: Caso de Uso Gestionar Permisos del Sistema.....	58
Figura 19: Caso de Uso Gestionar Cronograma.....	58
Figura 20: Caso de Uso Gestionar Nota Proyecto.....	59
Figura 21: Caso de Uso Gestionar Observaciones.....	59
Figura 22: Modelo Relacional.....	85
Figura 23: Base de Datos SQL Server.....	86
Figura 24: Diagrama de Secuencia Registro de Usuarios.....	87
Figura 25: Diagrama de Secuencia Logueo Usuarios.....	87
Figura 26: Diagrama de secuencia Crear Proyecto.....	88
Figura 27: Diagrama de Secuencia Actualizar Proyecto.....	88
Figura 28: Diagrama de Secuencia Buscar Proyecto.....	89
Figura 29: Diagrama de Secuencia Borrar Proyecto.....	89
Figura 30: Diagrama de Secuencia Crear Acta del Proyecto.....	90
Figura 31: Diagrama de Secuencia Actualizar acta.....	90
Figura 32: Diagrama de Secuencia Buscar Acta.....	91
Figura 33: Diagrama de Secuencia Borrar Acta.....	91
Figura 34: Diagrama de Secuencia Crear Informe Ejecutivo.....	92
Figura 35: Diagrama de Secuencia Actualizar Informe Ejecutivo.....	92
Figura 36: Diagrama de Secuencia Buscar Informe Ejecutivo.....	93
Figura 37: Diagrama de Secuencia Borrar Informe Ejecutivo.....	93
Figura 38: Diagrama de Secuencia Gestionar Permisos del Sistema.....	94
Figura 39: Diagrama de Secuencia Crear Observaciones del Proyecto.....	94
Figura 40: Diagrama de Secuencia Actualizar Observaciones del Proyecto.....	95
Figura 41: Diagrama de Secuencia Buscar Observaciones.....	95

Figura 42: Diagrama de Secuencia Borrar Observaciones del Proyecto.....	96
Figura 43: Diagrama de Secuencia Nota del Proyecto.....	96
Figura 44: Diagrama de Secuencia Crear Cronograma del Proyecto.....	97
Figura 45: Diagrama de Secuencia Actualizar Cronograma del Proyecto Propia.....	97
Figura 46: Diagrama de Secuencia Buscar Cronograma del Proyecto.....	98
Figura 47: Diagrama de Secuencia Borrar Cronograma del Proyecto.....	98
Figura 48: Diagrama de Clases.....	99
Figura 49: Pantalla Principal del Sistema.....	100
Figura 50: Pantalla de Registrar Usuarios.....	100
Figura 51: Pantalla de Iniciar Sesión.....	101
Figura 52: Pantalla de Crear Proyectos.....	101
Figura 53: Pantalla de Crear Actas del Proyecto.....	102
Figura 54: Pantalla de Crear Cronogramas.....	102
Figura 55: Pantalla de Crear Observaciones del Proyecto.....	103
Figura 56: Pantalla de gestionar Permisos del Sistema.....	103
Figura 57: Pantalla de Crear Modalidad del Proyecto.....	104
Figura 58: Pantalla de Crear Línea De Investigación del Proyecto.....	104
Figura 59: Implementación Pantalla principal.....	105
Figura 60: implementación registro de Usuarios.....	106
Figura 61: Implementación de Crear Proyectos.....	106
Figura 62: implementación Crear Actas del Proyecto.....	107
Figura 63: Implementación Crear Cronogramas del Proyecto.....	107
Figura 64: Implementación Crear Observaciones del Proyecto.....	108
Figura 65: Implementación Gestionar permisos del Sistema.....	109
Figura 66: Implementación Crear modalidad del Proyecto.....	109
Figura 67: Implementación Crear línea de Investigación.....	110
Figura 68: Implementación Crear Nota del Proyecto.....	111
Figura 69: Pruebas Unitarias del Sistema.....	111
Figura 70: Pruebas Unitarias: Seleccionar Proyecto para Pruebas.....	112
Figura 71: Pruebas Unitarias: test del Sistema.....	112
Figura 72: Pruebas Unitarias: Guardado del Test.....	113
Figura 73: Pruebas Unitarias: Código del Sistema.....	113
Figura 74: Pruebas Unitarias: Pruebas del Test.....	114
Figura 75: Pruebas Unitarias: Pruebas exitosas.....	114
Figura 76: Prueba de Registro del Sistema.....	115
Figura 77: Prueba de acceso a estudiantes.....	115
Figura 78: Prueba de Acceso Tutor.....	116
Figura 79: Pruebas de Logueo para Tutor.....	116
Figura 80: Pruebas de Acceso al Administrador.....	117
Figura 81: Pruebas de Acceso al Administrador.....	117
Figura 82: Pruebas de Crear un Proyecto.....	118
Figura 83: Pruebas de Acceso de Crear un Proyecto.....	118
Figura 84: Prueba de Crear un Cronograma.....	119

Figura 85: Prueba de Crear un Cronograma.....	119
Figura 86: Prueba de Crear Observaciones del Proyecto.....	120
Figura 87: Prueba de Crear Observaciones del Proyecto.....	120
Figura 88: Pruebas de Gestionar Permisos del Sistema.....	121
Figura 89: Pruebas de Gestionar Permisos del Sistema.....	121
Figura 90: Prueba de Crear Modalidad del Proyecto.....	122
Figura 91: Prueba para Crear Modalidad del Proyecto.	122
Figura 92: Prueba de Crear Línea de Investigación del Proyecto.	123
Figura 93: Prueba de Crear Línea de Investigación del Proyecto.	123
Figura 94: Prueba de Crear Actas del Proyecto.....	124
Figura 95: Prueba de Crear Acta del Proyecto.....	124
Figura 96: Prueba de Crear nota del Proyecto.....	125
Figura 97: Prueba de Integración Descendente.....	125
Figura 98: Diseños de Casos de Prueba.	127
Figura 99: Bugs del Sistema.	127
Figura 100: Bugs del Sistema.	128
Figura 101: Bugs del Sistema.	128
Figura 102: Manual del Usuario: Pantalla Principal del Sistema.	134
Figura 103: Manual del Usuario: Registro en el sistema.	135
Figura 104: Manual del Usuario: Pantalla de Logueo en el Sistema.....	135
Figura 105: Manual del Usuario: Ingreso Pantalla Principal.	136
Figura 106: Manual del Usuario: Creación de proyectos.....	136
Figura 107: Manual del Usuario: Creación del Proyecto.	137
Figura 108: Manual del Usuario: Creación Modalidad del Proyecto.....	137
Figura 109: Manual del Usuario: Creación Modalidad del Proyecto.....	138
Figura 110: Manual del Usuario: Creación Línea de Investigación del Proyecto.	138
Figura 111: Manual del Usuario: Creación Línea de Investigación del Proyecto.	139
Figura 112: Manual del Usuario: Creación del Cronograma del Proyecto.....	139
Figura 113: Manual del Usuario: Creación Cronogramas del Proyecto.	140
Figura 114: Manual del Usuario: Creación Observaciones del Proyecto.....	140
Figura 115: Manual del Usuario: Creación Observaciones del Proyecto.....	141
Figura 116: Manual del Usuario: Creación Permisos Administrativos.	141
Figura 117: Manual del Usuario: Creación Permisos Administrativos.	142
Figura 118: Manual del Usuario: Creación Actas del Proyecto.....	142
Figura 119: Manual del Usuario: Creación de Actas del Proyecto.	143

LISTA DE TABLAS.

Tabla 1: Cronograma del Proyecto.	17
Tabla 2: Proceso de Ingeniería del Software.	35
Tabla 3: Personal Involucrado: Administrador del Sistema.....	36
Tabla 4: Personal Involucrado: Director del Proyecto.	37
Tabla 5: Personal Involucrado: Desarrollador del Proyecto.	37
Tabla 6: Personal Involucrado: Usuario.	37
Tabla 7: Personal Involucrado: Evaluadores.....	37
Tabla 8: Personal Involucrado: Comité Curricular.....	38
Tabla 9: Características de los Usuarios.....	41
Tabla 10: Características del Usuario.	42
Tabla 11: Características de los Usuarios.....	42
Tabla 12: Requisito 1. Registro en el Sistema.	45
Tabla 13: Requisito 2. Logueo en el Sistema.....	46
Tabla 14: Requisito 3. Gestión de Tutores.....	46
Tabla 15: requisito 4. Seguimientos a Los Proyectos.	47
Tabla 16: Requisito 5. Permisos del Sistema para Tutor.	47
Tabla 17: Requisito 6. Registro de Fechas.	48
Tabla 18: Requisito 7. Registro de Nota para Tutor.	48
Tabla 19: Requisito 8. 1º Informe Ejecutivo.	49
Tabla 20: Requisito 9. 2º Informe Ejecutivo.	49
Tabla 21: Requisito 10. 3º Informe Ejecutivo.	50
Tabla 22: Requisito 11. Asignación de Jurado.....	50
Tabla 23: Descripción Caso de Uso Registro.	60
Tabla 24: Descripción Caso de uso de logueo.....	61
Tabla 25: Descripción Caso de Uso Crear Proyecto.....	61
Tabla 26: Descripción Caso de Uso Actualizar Proyecto.....	63
Tabla 27: Descripción de Caso de Uso Buscar Proyecto.	64
Tabla 28: Descripción Caso de Uso Borrar Proyecto.....	65
Tabla 29: Descripción Caso de Uso Crear Acta del Proyecto.....	66
Tabla 30: Descripción Caso de Uso Actualizar Acta del Proyecto.....	68
Tabla 31: Descripción Caso de Uso Buscar Acta del Proyecto.....	69
Tabla 32: Descripción Caso de Uso Borrar Acta del Proyecto.....	69
Tabla 33: Descripción Caso de Uso Crear Informe Ejecutivo.	70
Tabla 34: Descripción Caso de Uso Actualizar Informe Ejecutivo.	72
Tabla 35: Descripción Caso de Uso Buscar Informe Ejecutivo.	73
Tabla 36: Descripción Caso de Uso Borrar Informe Ejecutivo.	74
Tabla 37: Descripción Caso de Uso Gestionar Permisos del Sistema.....	75
Tabla 38: Descripción Caso de Uso Crear Observaciones del Proyecto.	76
Tabla 39: Descripción Caso de Uso Actualizar Observaciones del Proyecto. ...	77

Tabla 40: Descripción Caso de Uso Buscar Observaciones del Proyecto.	78
Tabla 41. Descripción Caso de Uso Borrar Observaciones del Proyecto.	79
Tabla 42. Descripción Caso de Uso Crear Nota para Proyectos.	79
Tabla 43: Descripción Caso de Uso Crear Cronograma del Proyecto.	81
Tabla 44: Descripción Caso de Uso Actualizar Cronograma del Proyecto.	82
Tabla 45: Descripción Caso de Uso Buscar Cronograma del Proyecto.	83
Tabla 46: Descripción Caso de Uso Borrar Cronograma del Proyecto.	84
Tabla 47: Diseño de Casos de Prueba.	127

INTRODUCCION

La Universidad Católica de Pereira, entre sus diferentes modalidades de grado existe una llamada proyecto de grado, en ella se maneja gran cantidad de información referente a estudiantes profesores y procesos de los trabajos. El programa de Ingeniería Industrial queriendo entrar en la era digital, requiere de un sistema que administre información de los proyectos y que pueda dar un manejo sistematizado dando la posibilidad que tanto el tutor, estudiante y director de programa puedan tener acceso a ella.

Existen algunos trabajos relacionados para la gestión de proyectos, por ejemplo, La Universidad Minuto de Dios tiene una aplicación que se denomina sistema de información para la gestión de proyectos de grado (SIAP)[1], donde los estudiantes gestionan sus proyectos de grado, por medio de la virtualidad a través de los correos electrónicos asignados por la universidad; así mismo la Universidad de Cartagena tiene una aplicación llamada sistema de información para la gestión de trabajos de grado, unificando manuales de usuario, sistema, informes de investigación, logrando de esta manera ahorrar tiempo y dinero en la gestión de la información.

Los trabajos que se mencionan anteriormente, no cuentan con un sistema que ayude a gestionar la información entre tutor y estudiante. Permitir que los estudiantes puedan hacer una buena gestión con la información de los proyectos, en tiempo real, evidenciando los logros y alcances del estudiante.

La propuesta consiste en desarrollar una aplicación en ambiente web, que permita a estudiantes y tutores contar con un sistema que los ayude en el proyecto de grado. Para el desarrollo del sistema de información, se utiliza el modelo en cascada, que facilita el desarrollo en fases. Esto será posible con la ayuda de Visual Studio utilizando el lenguaje programación C#.

El trabajo se organiza así: en la primera parte se presenta la formulación del proyecto, en la segunda parte, el marco contextual, la tercera parte contiene el marco teórico, la cuarta parte el modelo teórico y la quinta parte se presenta la concreción del modelo.

1. FORMULACIÓN DEL PROYECTO

1.1. DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA

1.1.1. Situación Problemática

Desde la Facultad de Ciencias Básicas e Ingeniería de la Universidad Católica de Pereira, en el programa de Ingeniería Industrial, se observa la necesidad de gestionar la información que se obtiene de los proyectos de grado, debido a que no se cuenta con un software que gestione la información, que además contribuya al avance del proyecto de los estudiantes del programa Ingeniería Industrial.

1.2. OBJETIVOS

1.2.1. Objetivo General

Desarrollar un sistema web para la gestión de la información de los proyectos de grado del programa de Ingeniería Industrial, Facultad de Ciencias Básicas e Ingeniería de la Universidad Católica de Pereira.

1.2.2. Objetivos Específicos

- Identificar de manera detallada los requisitos necesarios para el desarrollo de un sistema de información para la gestión de proyectos de grado del programa Ingeniería Industrial de la Universidad Católica de Pereira.
- Realizar el diseño del sistema de información para la gestión de proyectos de grado, aplicando buenas prácticas de Ingeniería del Software.
- Implementar el sistema de información para la gestión de proyectos de grado.
- Realizar las pruebas de software para verificar el correcto funcionamiento del sistema.

1.3. JUSTIFICACIÓN

Para mejorar los procesos del manejo de la información de los trabajos de grado, del programa Ingeniería Industrial, se observa la necesidad de desarrollar un sistema de información, para que tutores y estudiantes tengan una buena coordinación de las tareas que se presentan durante el desarrollo del proyecto

La creación de este sistema tiene una relevancia significativa para el programa de Ingeniería Industrial, en la que se demuestra la necesidad de tener un sistema de gestión de la información, este sistema de información ayudará al docente y estudiante a tener un mejor seguimiento al proyecto, dado que se estará contribuyendo a que los proyectos del programa puedan tener un seguimiento.

1.4. CRONOGRAMA

Dado el proyecto que se propone, las actividades del mismo se regirán de acuerdo a las fechas establecidas en el cronograma mostrado en la Tabla 1.

Tabla 1: Cronograma del Proyecto.

Año	2019								
Actividad	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic
Requisitos									
Diseño									
Implementación									
Pruebas									

Fuente: Elaboración Propia.

- Requisitos: Esta actividad se desarrollará en un tiempo estimado de 3 meses.
- Diseño: Esta actividad se implantará en un tiempo de 3 meses para gestionar el proyecto.
- Implementación: Esta actividad será desarrollada en un lapso máximo de 3 meses.
- Pruebas: Para las pruebas respectivas puede tomar un mes y se realizarán pruebas unitarias.

2. MARCO CONTEXTUAL.

Desde el programa de Ingeniería Industrial de la facultad de Ciencias Básicas e Ingeniería de la Universidad Católica de Pereira, se viene presentando un problema con los proyectos de grado, dado que no poseen ningún sistema que ayude a la gestión de la información de los proyectos, en la actualidad los alumnos presentan su proyecto de grado de manera manual, dando lugar a que los usuarios se retrasen o pierdan la información.

En la actualidad no existen sistemas que ayuden a gestionar proyectos de grado en la carrera de Ingeniería Industrial de la Universidad Católica de Pereira, dando la posibilidad de crear un sistema de información que ayude en gran medida a resolver el problema.

Las características que presenta este sistema de información para la gestión de proyectos son:

- Ambiente de usuario sencillo.
- Ingreso ágil al sistema.
- Ingreso de información fácil y rápida.
- Fácil manejo.

3. MARCO TEORICO

3.1. Antecedentes

3.1.1. Sistema de Información para la gestión de proyectos de grado para la Universidad Minuto de Dios (Siap).

Se busca crear un Sistema de Información transaccional, el cual permita conocer al detalle cada avance de un proyecto de grado mediante casos generados, asesorías brindadas, estimación de tiempos, actualizaciones de estado y notificaciones personales por medio de correo a estudiantes, docentes y directivos, usando para ello la metodología RAD (Rapid Application Development – Desarrollo Rápido de Aplicaciones), la cual permite un rápido desarrollo y entrega de una alta calidad, en este proyecto de bajo coste de inversión [1] de acuerdo a la cita mencionada, se hace evidente la necesidad que ésta universidad observa en la misma línea de trabajo, alrededor de los proyectos de grado, sin embargo, no se describe de una manera detallada como pretende este sistema, que se desarrolla para el programa de Ingeniería Industrial.

3.1.2. Sistema de Información para la Gestión de los Trabajos de Grado (Universidad de Cartagena).

El desarrollo del trabajo se fundamenta teóricamente a partir de estudios realizados sobre: sistemas de información, metodología de desarrollo RUP, procedimientos administrativos y trabajos de grado. Como resultado del trabajo se obtuvo un sistema de información para la gestión de los Trabajos de grado en la Universidad de Cartagena, junto con los manuales de usuario y del sistema, además del informe de investigación; y logrando concluir con un sistema que mejora la gestión de los trabajos de grado se observó además un sistema meramente básico en el que el tutor y estudiante interactúan de manera virtual sin mayor consecuencia que un proyecto unificado sin mayor pérdida de información [2].

3.1.3. Gestión de Proyectos de Final de Grado (Universidad de Barcelona).

La Universidad de Barcelona ha desarrollado durante los últimos años una aplicación integrada en el Racó (Intranet de la facultad) para la gestión de los proyectos finales de carrera de la FIB que agiliza todos los trámites que hasta ahora se tenían que hacer presencialmente en secretaría.

La nueva aplicación de gestión de trabajos finales de grado amplía esta aplicación para apoyar la evaluación de los trabajos en base a rúbricas y a la orientación a competencias. Además este proyecto está hecho con los programas Tomcat, Oracle, Java, HTML5, CSS, JSP, AJAX. También cuenta con áreas de especialización aplicadas al proyecto en entornos y servicios TIC de soporte al aprendizaje [3]

3.1.4. Prototipo de una Aplicación Web para la Gestión en el Proceso de Trabajos de Grado de la Universidad Ean.

La Universidad EAN, ubicada al norte la ciudad de Bogotá, es una institución educativa de prestigio dado a sus excelentes programas académicos tanto de Pregrado, Posgrado y a distancia, como en el buen impacto emprendedor que genera en las empresas. En cuanto a las carreras de Pregrado, unos de los requisitos fundamentales para la titulación, es la presentación de un Trabajo de grado, donde se evidencie todo el conocimiento adquirido a lo largo de la carrera. Se realiza mediante un prototipo web donde se incorporan módulos de autenticación de usuarios, creación de proyectos, jurados y los reportes de cada uno de los proyectos de grado [4].

3.2. Antecedentes Teóricos

3.2.1. Modelo en Cascada

Es llamado también secuencial o ciclo de vida, este modelo ordena muy rigurosamente las etapas del proceso para el proyecto de software, como se muestra en la Figura 1, de esta forma el inicio de cada etapa debe esperar a que finalice la anterior, para poder continuar con el desarrollo del sistema [5].

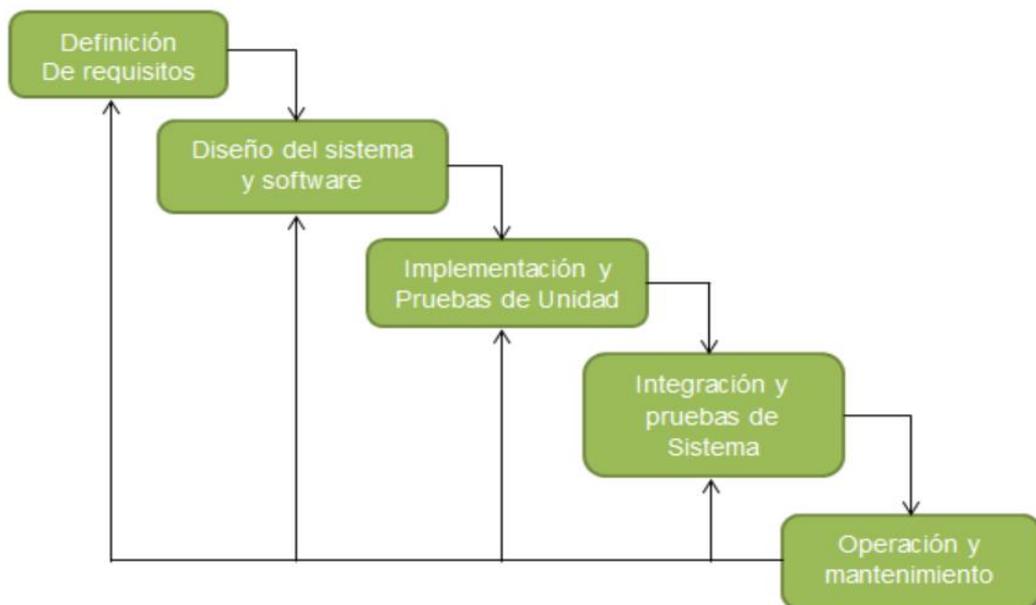


Figura 1: Modelo en Cascada.
Fuente: Elaboración propia basado en [5]

- Análisis: En esta fase comienza la obtención de requisitos u objetivos para determinar si es posible realizar el software; también se tiene que tener en cuenta los casos de uso, prototipos de pantallas e informes.
- Diseño: En esta etapa del sistema es donde se adapta ese análisis al desarrollo del software, teniendo en cuenta la arquitectura general del sistema de información, división de subsistemas, modelo de datos lógicos, modelo de clases, y por último en caso de desarrollar en objetos el plan detallado de pruebas.
- Codificación: En esta fase se realiza la construcción del sistema y las pruebas, como pruebas unitarias, integración y del sistema. Este proceso tiene otras etapas como la realización de carga inicial de datos o la construcción del procedimiento de migración.
- Pruebas: En esta etapa es donde se hacen las pruebas pertinentes a la implantación (donde se verifican aspectos no funcionales) donde se espera que los usuarios aprueben cómo funciona el sistema. Por último se realiza la implantación del sistema entorno a la producción.
- Mantenimiento: Una vez que se ha desarrollado el sistema, se clasifican los mantenimientos en correctivos, evolutivos, adaptativos y periféricos [5].

Ahora se mostrarán las ventajas y desventajas del modelo en cascada [6]:

✓ Ventajas:

- Se tiene un orden establecido y nunca se mezclan las fases.
- Se puede planificar de una manera fácil y sencilla.
- La calidad final del software es de alta calidad.
- Se puede empezar bastante rápido el software.
- Puede estimar el presupuesto con una mayor precisión.

✓ Desventajas:

- Se tarda bastante tiempo en pasar los ciclos.
- Es difícil la incorporación de nuevas actualizaciones para el software.
- Las iteraciones son un poco costosas.
- Alterar en diseño del proyecto en cualquiera de una de sus fases es demasiado complicado.
- Cuando se termina una fase es muy difícil volver a realizar cambios al desarrollo.

3.2.2 Ciclo de Vida V

El modelo de ciclo en v, muestra cómo se relacionan las actividades de pruebas de análisis y diseño, este hace que las iteraciones y repeticiones que están ocultas en este modelo v, como se muestra en la figura 2, además este modelo se centra en las actividades y corrección [5].

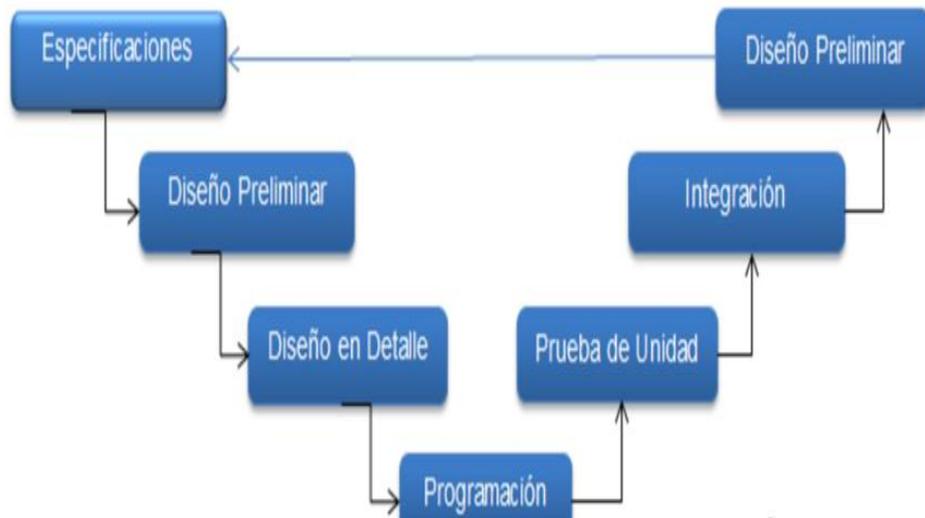


Figura 2: Ciclo de Vida en V.
Fuente: Elaboración propia basado en [5].

Ahora se mostrarán las ventajas y desventajas del modelo en V [7]:

✓ Ventajas:

- La relación entre desarrollo y tipos de pruebas facilitan la corrección de fallos.
- Este modelo es fácil de aprender.
- Está bien Explicada la parte de iteración y trabajo a revisar.
- Involucra al usuario en el sistema.
- Las pruebas que se hacen en cada fase ayudan a corregir posibles errores que se presenten sin que finalice cada fase del modelo.

✓ Desventajas:

- Es difícil que el cliente exponga explícitamente todos los requisitos.
- El producto final obtenido puede que no refleje todos los requisitos del usuario.
- Al encontrarse errores luego de haber realizado las pruebas se pierde demasiado tiempo y dinero.

- El modelo no permite la posibilidad de retornar a fases anteriores, pero esto suele suceder.

3.2.3. Modelo en Espiral

Es un modelo de software Evolutivo propuesto por Boehm, el cual integra la metodología repetitiva para la construcción de prototipos con aspectos sistemáticos y controlados del modelo lineal secuencial o cascada como en la Figura 3. Este modelo tiene un sin número de iteraciones, y es el equipo de gestión de proyecto el que decide qué iteraciones llevará el proyecto [5].

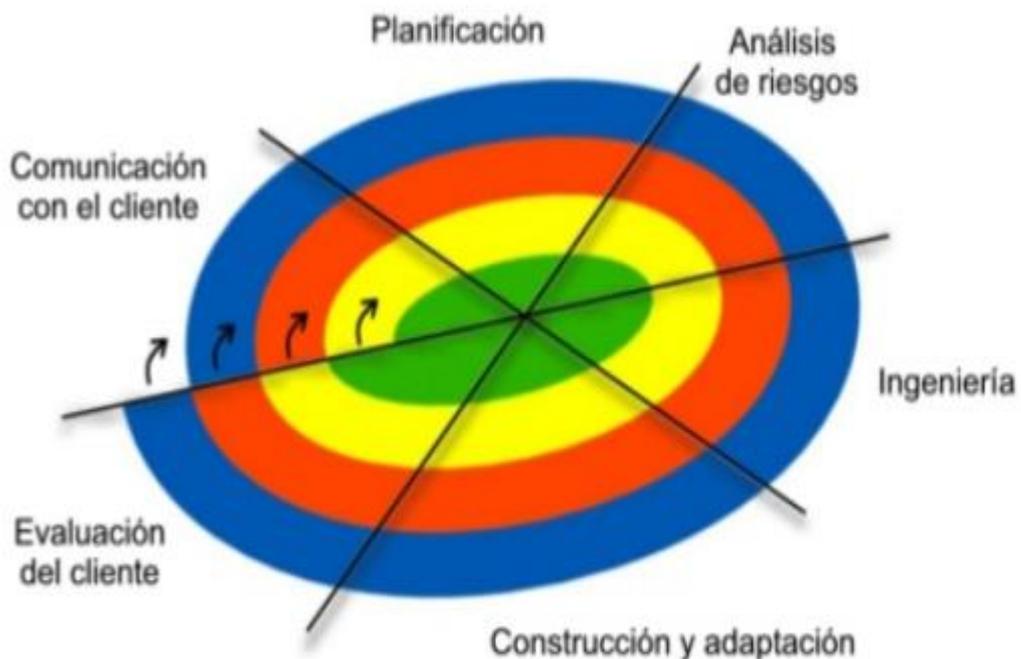


Figura 3. Modelo en Espiral.
Fuente: Elaboración propia basado en [5]

Este modelo está constituido en 6 fases que se describen a continuación [8]:

- Comunicación con el cliente: En esta etapa del desarrollo se especifican los objetivos, se identifican las restricciones tanto del producto como del proceso, se traza un plan de desarrollo y se identifican a su vez los riesgos del proyecto, dependiendo de los riesgos así mismo se trazan estrategias alternativas.
- Planificación: Después de que se evalúan los riesgos, se elige un modelo adecuado de desarrollo del sistema, además dependiendo de los riesgos así mismo pueden ser elegidos los sistemas, por ejemplo si tienen problemas de

seguridad el mejor sistema para implementarlo sería el sistema en transformaciones formales.

- Ingeniería: Las tareas que se requieren para construir una o varias tareas que tiene la aplicación.
- Análisis de Riesgo: En esta parte del modelo se lleva un análisis exhaustivo de cada riesgo y se estudia cómo reducir dichos riesgos.
- Evaluación del Cliente: El cliente en esta fase es donde observa el cómo quedo el producto, como funciona y cómo se maneja el sistema final, aquí se verán plasmados todos los requerimientos que el cliente quiso para el sistema.
- Construcción y Entrega: En esta etapa del modelo en espiral se revisa el proyecto y se debe de tomar la decisión si se continúa con los ciclos posteriores, si se continúa se desarrollan los planes de desarrollo de la siguiente fase del espiral.

Las ventajas y desventajas que tiene el modelo en espiral son [9]:

✓ Ventajas:

- El modelo en espiral puede adaptarse a lo largo de la vida del computador.
- Como cada vez el desarrollo del software avanza, el cliente y el equipo desarrollador comprenden mucho mejor los riesgos que se puedan presentar en la evolución del desarrollo.
- El modelo en espiral permite al desarrollador aplicar enfoques de construcción de prototipos en cualquier etapa del desarrollo.
- Cuando se utiliza en sistemas grandes de dobla la productividad.

✓ Desventajas:

- Este modelo es aconsejable no utilizarlo en sistemas pequeños.
- Este modelo es un poco costoso.
- Se requiere de experiencia en este modelo para identificar los riesgos.

3.2.4. Modelo de desarrollo Evolutivo o Prototipo

Este modelo se inicia con la definición de los objetivos globales, este modelo es utilizado para darle al usuario o cliente una vista preliminar del desarrollo, además se conoce como el modelo prueba y error ya que si al usuario o cliente no le gusta lo que ve del prototipo significa como una prueba fallida y se debe corregir hasta que el cliente lo apruebe como en la Figura 4 [10].



Figura 4. Modelo de desarrollo Evolutivo [5].

Las etapas del modelo de desarrollo evolutivo son las siguientes [11]:

- Recolección y refinamiento de requisitos.
- Modelado, Diseño Rápido.
- Construcción del Prototipo.
- Desarrollo, Evaluación del prototipo por el cliente.
- Refinamiento del prototipo.
- Producto de Ingeniería.

Las ventajas y desventajas de la metodología de prototipos son las siguientes [12]:

✓ Ventajas:

- El modelo permite las modificaciones tempranas del sistema de su desarrollo.
- Si se quiere que el prototipo sea un éxito se debe de esta retroalimentando en cada momento el proyecto.
- Permite al desarrollador que es lo que realmente desea el cliente.
- Permite al desarrollador cómo se va avanzando en el desarrollo.
- Este modelo es flexible y se adapta muy bien al cambio.

✓ Desventajas:

- Adoptarlo como sistema final aun cuando no se ha terminado o está incompleto.
- El cliente y el desarrollador se comunican muy poco en el inicio del proceso.

- Pueden surgir cambios imprevistos que atrasen de manera significativa el prototipo.

3.2.5. Metodología de Desarrollo Rápido de Aplicaciones (RAD)

El método comprende el desarrollo interactivo, la construcción de prototipos y el uso de la herramienta CASE (Ingeniería Asistida por Computadora) el funcionamiento de la metodología es mostrado en la Figura 5, Hoy se suele utilizar para el desarrollo rápido de interfaces como Glade, entornos de desarrollo integrado tales como Visual Studio, Lazarus, Gambas, Delphi, FoxPro [13].

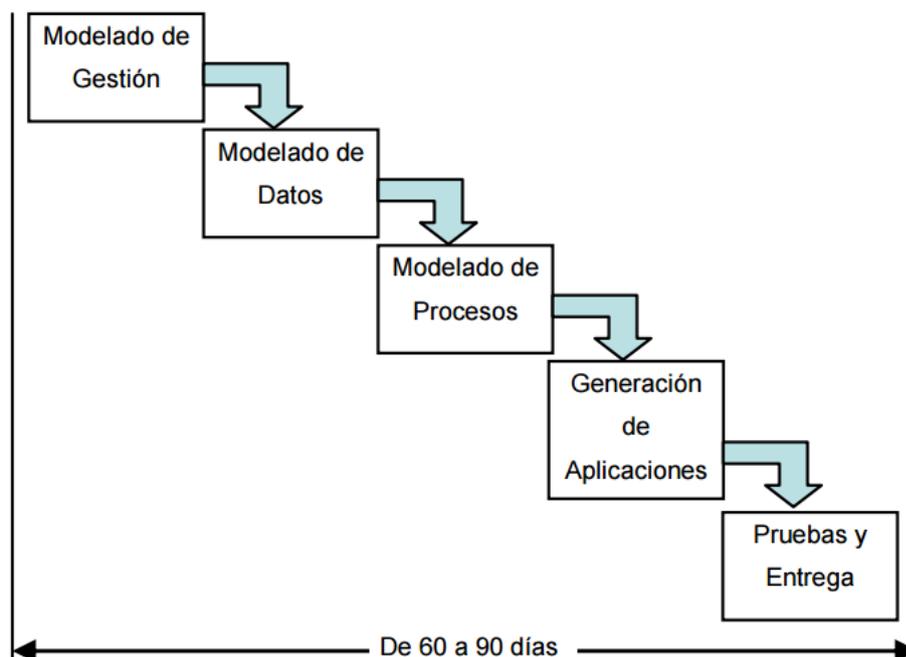


Figura 5: Metodología RAD [13].

Las fases de la Metodología de desarrollo rápido de aplicaciones (RAD) son [13]:

- Modelado de Gestión: El flujo de la información entre las funciones de gestión se modela de forma de preguntas tales como ¿Quién genera la información?, ¿A dónde va la información?, Etc.
- Modelado de Datos: Se definen los atributos de cada uno de los objetos y las relaciones entre los objetos.
- Modelado de Proceso: después de haber modelado los datos quedan listos para lograr la información necesaria para implementar una función de gestión. Es la comunicación entre objetos.

- Generación de aplicaciones: Esta metodología asume técnicas para trabajar en lenguajes de programación de cuarta generación, también se utiliza para volver a utilizar componentes de programas ya hechos o a crear componentes reutilizables.
- Pruebas de entrega: como esta metodología reutiliza componentes existentes este reduce el tiempo de pruebas. Siempre se deben de comprobarlos componentes para ejercitar las interfaces a fondo.

Las ventajas y desventajas de esta metodología son las siguientes [13]:

✓ Ventajas:

- Limitar la exposición del proyecto a las fuerzas de cambio.
- Ahorra tiempo de desarrollo a expensas de dinero o de la calidad del producto.
- Mayor flexibilidad.
- Menor codificación del desarrollo.
- Se reducen las fallas.
- Ciclos de desarrollo más pequeños.

✓ Desventajas:

- Es menos eficiente.
- El progreso es más difícil de medir.
- Se puede producir más fallas.
- Dependencia de componentes de terceros.

3.2.6. Metodologías Ágiles

Son muy útiles para visualizar y organizar las tareas a realizar y para mejorar el rendimiento y el trabajo en equipo. Permite tener seguimiento detallado de cada etapa de un proyecto, tanto a nivel personal como grupal, además, permiten visualizar de manera clara las tareas a realizar, gracias a la organización incremental la calidad del trabajo y disminuyen de manera considerable el tiempo que se invierte en cada tarea, además de lo anterior esto hace que la comunicación interna del equipo mejore [14].

3.2.6.1. Metodología SCRUM

Es un framework de desarrollo ágil, es un método iterativo que divide el desarrollo de software en ciclos, estos ciclos son llamados Spring, el equipo desarrollador prioriza la lista de requisitos, al acabar un ciclo el producto debe de estar terminado y entregable como en la figura 6 [15]

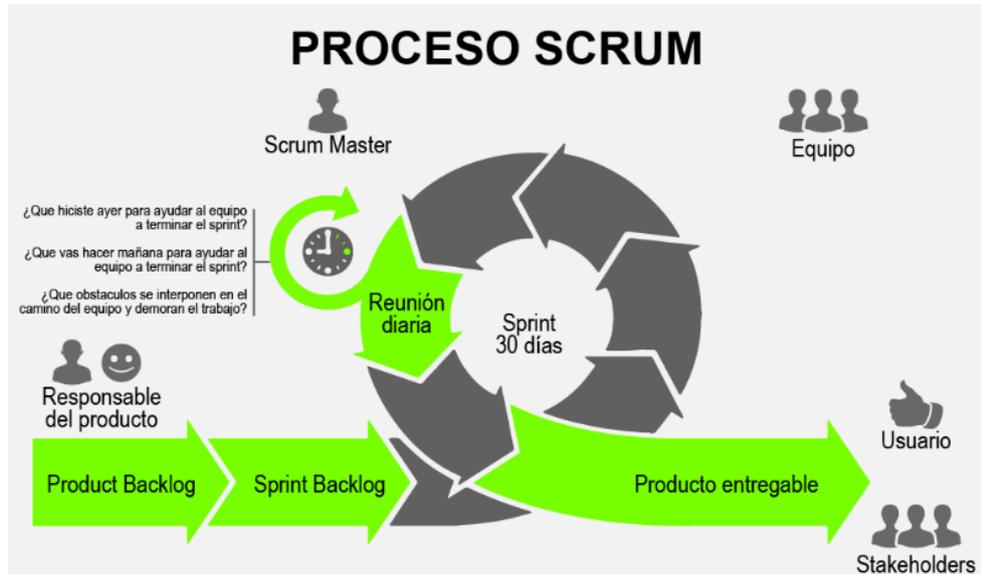


Figura 6. Metodología SCRUM.
Fuente: Elaboración propia basado en [15].

Esta metodología SCRUM tiene unas ventajas y desventajas que son [16]:

✓ Ventajas:

- Se tiene comunicación permanente con el cliente o usuario.
- Desarrollo incremental con entregas funcionales.
- Comunicación permanente con el equipo de desarrollo.
- Simplicidad.
- Entrega el mayor valor primero.

✓ Desventajas:

- No puede tener equipos demasiado grandes.
- Gente con muy poca experiencia.

3.2.6.2. Metodología de Programación Extrema

Es una metodología de desarrollo que tiene como objetivo aumentar la productividad a la hora de desarrollar software. El objetivo principal es entregar un software de calidad, la metodología de programación extrema tiene como base la simplicidad y la satisfacción del cliente como muestra la figura 7 [17].

Como hacemos funcionar la Metodología XP

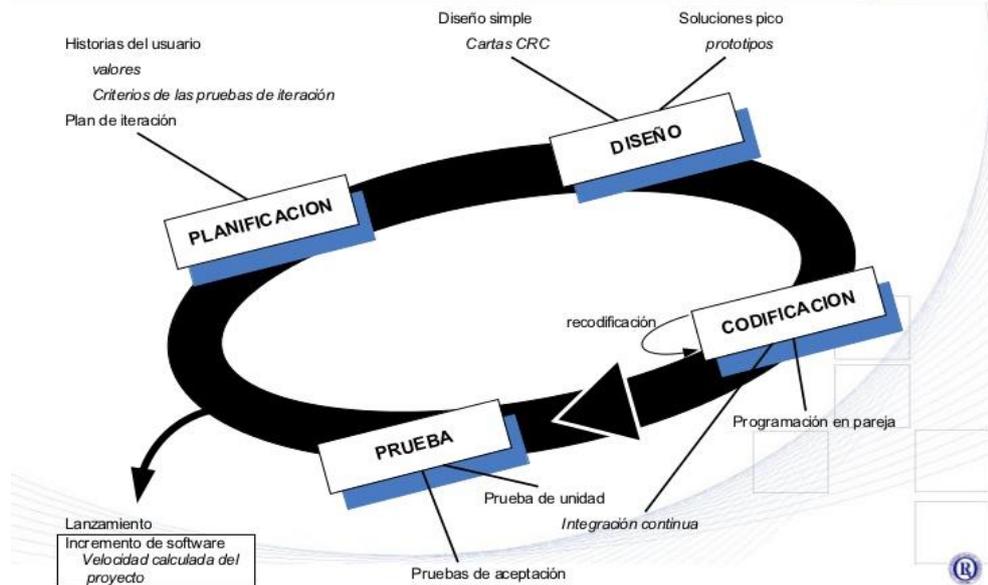


Figura 7: Metodología XP.
Fuente: Elaboración propia basado en [17].

La programación extrema tiene doce principios que son [17]:

- Principio de pruebas.
- Proceso de planificación.
- El cliente en el lugar.
- Programación en parejas.
- Integración continua.
- Refactorización
- Entregas pequeñas.
- Diseño simple.
- Metáforas.
- Propiedad colectiva del código.
- Estándar de decodificación.
- La semana es de 40 horas.

Esta programación extrema tiene ventajas y desventajas que son [18]:

✓ Ventajas:

- Es muy organizada la programación.
- Cuenta con una tasa de errores muy pequeñas.

- Fomenta la comunicación cliente-desarrolladores.
 - Facilita los cambios.
 - El cliente tiene el control sobre las prioridades.
 - Permite ahorrar tiempo y dinero.
- ✓ Desventajas:
- Es recomendable para proyectos a corto plazo.
 - Los fallos pueden salir caros.
 - Puede resultar más difícil para el desarrollo.

3.2.7. Arquitectura MVC

MVC es una arquitectura que separa los datos, la interfaz, la lógica de control en tres componentes diferentes, la llamada y el retorno que hace MVC se ve regularmente en las páginas web, Donde el HTML hace de vista, el modelo es el sistema de gestión y bases de datos, el controlador es el encargado de recibir la vista como puede verse en la Figura 8 [19].

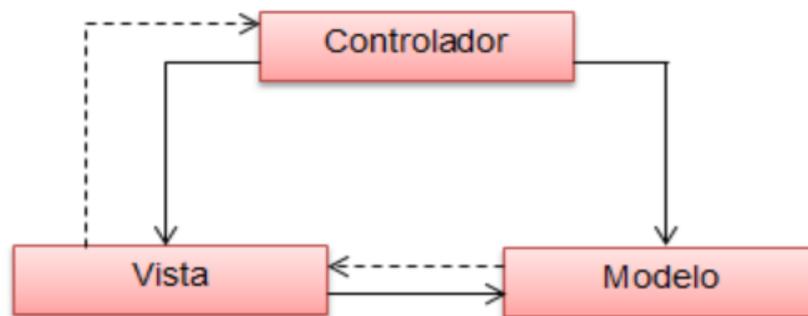


Figura 8: MVC.

Fuente. Elaboración propia basado en [19].

A continuación, se mostrarán algunas características de MVC [19]:

- El modelo es el objeto y representa los datos, es el mismo sistema es el responsable de mantener unidos el modelo y vistas, y notifica cuando al modelo le surge algún cambio.
- La vista es la que representa los datos en la parte visual, esto genera una vista que es la que se muestra al usuario.
- El controlador es el que proporciona las órdenes del usuario, este actúa como su nombre lo indica como el quien controla así sea cuando se produce un cambio o sea en el modelo o bien sea en la vista.
- Ventajas de utilizar MVC.

Este modelo hace una separación entre lógica y presentación, se le puede aplicar distintos módulos sin alterar la lógica, Cuando hace separación de capas lo hace para hacer arquitecturas consistentes, que se pueden reutilizar y su mantenimiento es mucho más fácil, en lo cual resulta muy rentable para trabajos posteriores [19].

3.2.7.1. Asp.Net

Es un modelo de desarrollo que incluye unos servicios unificados para crear aplicaciones web con el mínimo de código posible.

Asp.Net está integrado a .Net framework, el código de los proyectos pueden escribirse en un lenguaje compatible con el Common Language Runtime (CLR) entre otros Visual Basic, C#, JavaScript, .Net y J# estos lenguajes permiten realizar aplicativos en Asp.Net [20].

3.2.7.2. Entity Framework

Entity Framework es un conjunto de tecnologías ADO.NET que permite el desarrollo de aplicativos orientadas a datos, tanto arquitectos como diseñadores orientados a datos tienen la necesidad de obtener dos objetos muy diferentes, deben de modelar las relaciones y la lógica de los problemas que resuelven, y deben de trabajar con los motores de bases de datos que se usan en el almacenamiento y recuperación de datos. Los datos pueden tener varios sistemas de almacenamiento, cada uno con sus propios protocolos, aplicaciones que funcionan con un solo sistema de almacenamiento deben de emparejar con los requisitos del sistema de almacenamiento, con los requisitos de un código de aplicación seguro, eficaz y facilidad para el mantenimiento [21]

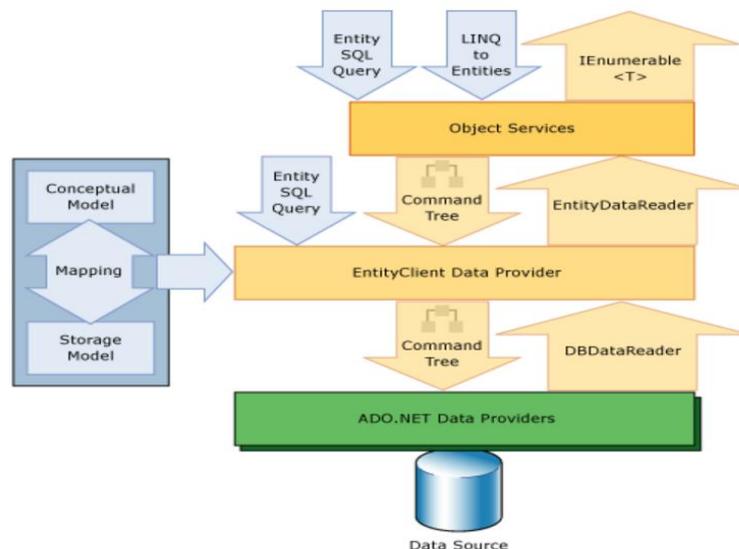


Figura 9. Entity Framework [21].

3.2.7.3. Visual Studio

Visual Studio es un programa de desarrollo integrado (IDE) para sistemas operativos. Este soporta varios lenguajes de programación como Visual C++, Visual C#, Visual J# como también Visual Basic, .Net al igual que desarrollos web como Asp.Net.

Visual Studio permite a los desarrolladores crear aplicativos, sitios y aplicativos web, así como en cualquier entorno que soporte .Net, esto hace que se puedan crear aplicativos que se comuniquen entre estaciones de trabajo, aplicativos web, móviles.

Microsoft Visual Studio es un programa completo de herramientas y servicios para desarrollar aplicativos para equipos de escritorio, la web, dispositivos la nube, tanto si va a crear su primer aplicativo, como si lo va a compilar en un sitio web compatible con los últimos exploradores, este aprovecha los conocimientos en el entorno de desarrollo que tiene Visual Studio para otros lenguajes [22].

3.2.8. Lenguajes de Programación

Es un sistema estructurado de comunicación, el cual permite la comunicación por medio de signos, palabras, sonidos y gestos, este sistema está organizado para que se entiendan entre si y a su vez haga interpretación de las instrucciones que se deben realizar, el termino programación se refiere a un conjunto de ordenes consecutivas y ordenadas que llevan a cabo una tarea específica, dichas tareas se llaman código fuente es cual es para cada lenguaje y está capacitado para una función o propósito específico, estas utilizan diferentes normas o bases para controlar un dispositivo también se pueden crear para programas informáticos [23].

3.2.8.1. Java

Sun Microsystem inventó el lenguaje 1991 el cual fue enfocado al paradigma de objetos y todos lo conocen como Java. El objetivo primario era utilizar a Java para un Set-top box, se trata de un sistema que ayuda a decodificar la señal de televisión, el primer nombre que se le dio a este lenguaje fue OAK, después Green y por último y como se conoce actualmente Java, la intención de Sun Microsystem era desarrollar un lenguaje con una estructura parecida a C, C++, aunque un lenguaje más simple y borrando herramientas de bajo nivel [24].

3.2.8.2. PHP

Por sus Siglas en Ingles PHP (Hypertext Pre-processor) es un lenguaje de código abierto adecuado para desarrollo web y puede incrustarse en HTML,

PHP está del lado del usuario como JavaScript que es el código ejecutado en el servidor, generando HTML y enviado al usuario, este recibe el resultado al ejecutar el script, aunque no podrá ver el código que lo hizo.

El servidor puede ser configurado para que procese ficheros HTML con PHP, lo mejor de PHP es que su entorno es muy fácil de manejar para la persona que apenas comienza a conocerlo, además ofrece características avanzadas para los programadores con más experiencia, aunque el desarrollo de PHP está centrado en la programación de scripts del lado del servidor, se puede utilizar para muchas otras cosas [25].

3.2.8.3. JavaScript

Es un lenguaje muy utilizado por los profesionales y para quienes están comenzando el desarrollo y diseño web, no necesita compilarse ya que este funcional del lado del usuario, los intérpretes de estos códigos con los navegadores, mucha gente piensa que JavaScript es Java y es ahí donde se equivocan, ambos lenguajes son muy diferentes y cada uno tiene sus características diferentes, JavaScript tiene una gran ventaja y es que puede ser incorporado, también puede ser ejecutado sin necesidad de otros programas para su funcionamiento y ser visualizado. Este lenguaje tiene como característica principal que es un lenguaje independiente de su plataforma, se puede crear cualquier programa y puede ser ejecutado en cualquier sistema operativo como Linux, Windows, Apple [26].

4. MODELO TEORICO

4.1. APORTE TEORICO Y PRÁCTICO

Este trabajo ayudará a los usuarios del programa de ingeniería industrial de la Universidad Católica de Pereira a gestionar los proyectos de grado, donde podrán estar al tanto de las novedades que se presentan.

Además ayudará a acelerar los procesos que se realizan en el programa de ingeniería industrial, para la inscripción de los proyectos de grado y en general para la Universidad Católica de Pereira.

Este proyecto se realiza con los procesos que actualmente se manejan en la facultad de Ciencias Básicas e Ingeniería, teniendo en cuenta como elemento de programación la metodología MVC y el motor de base de datos MySQL.

Para hacer posible un proceso de ingeniería de software, se identifican aquellos requisitos tanto de software como de hardware, bases de datos y otros elementos que hacen posible la realización del desarrollo del software.

4.1.1. Proceso de Ingeniería de Software

Debido a la gran cantidad de información que se administra en el programa de Ingeniería Industrial, se ha dado a la tarea de realizar un sistema de información para la gestión de proyectos de grado, para esto se ha utilizado el modelo para el desarrollo de software llamado modelo en cascada, el cual utiliza proceso de desarrollo secuencial por fases, debido a que se desarrolla de forma secuencial, esto quiere decir que sólo se podrá comenzar una fase cuando se termine la actual, este modelo aporta al desarrollo del proyecto organización y fiabilidad.

Un proceso es un conjunto de actividades interrelacionadas que transforman entradas en salidas.

Tabla 2: Proceso de Ingeniería del Software.

En la Tabla 2, se muestra el proceso o fases que se realizan para desarrollar un software.

Actividad	Descripción
Fase de Requisitos	<ul style="list-style-type: none">• Planificación del proyecto de desarrollo software.• Ingeniería de requisitos / Extracción de información.• Análisis (estudio) de esos requisitos.
Fase de Análisis	<ul style="list-style-type: none">• Desarrollo de requisitos estudiados en la anterior etapa.
Fase de Diseño	<ul style="list-style-type: none">• La Arquitectura obtenida en el análisis se transforma en una arquitectura especializada, donde se plantea el ambiente del sistema.• Diseño de Software.
Fase de Implementación	<ul style="list-style-type: none">• Se expresa en que arquitectura estará el sistema.• Implementación del código del sistema.
Fase de Pruebas	<ul style="list-style-type: none">• Corrección de errores y reajustes que a veces provienen de nuevos requisitos e implican repetir las actividades de fases anteriores.• Se hacen pruebas del software

Fuente: Elaboración propia basado en [27].

El objetivo principal de la Especificación de Requisitos del Sistema (ERS), es servir como medio de comunicación entre clientes, usuarios, ingenieros de requisitos y desarrolladores. En la ERS deben recogerse tanto las necesidades de clientes y usuarios como los requisitos que debe cumplir el software a desarrollar, para satisfacer dichas necesidades (requisitos del producto, también conocidos como requisitos de sistema o requisitos software). La ERS debe ser un documento consensuado entre todas las partes y tener un carácter contractual, de forma que cualquier cambio que se desee realizar en él, una vez acordada la primera línea base, deba aplicarse siguiendo el procedimiento de control de cambios establecido en el proyecto [5].

4.1.2. Requisitos.

A. Propósito.

El propósito de este documento, es orientar al lector de cómo obtener los requisitos necesarios para la elaboración de este proyecto, teniendo el tipo de software y la metodología, que se tiene en cuenta para la elaboración de este sistema de Información.

B. Alcance.

Este aplicativo se denomina Sistema de Información para la Gestión de Proyectos de Grado del programa Ingeniería Industrial de la Universidad Católica de Pereira. Pretende crear un sistema web que permita a docentes, estudiantes y administrador, extraer, compilar la información durante la formulación y finalización de los proyectos de grado de los estudiantes del programa.

C. Personal involucrado en el proyecto.

El personal involucrado se presenta en las tablas 3, tabla 4, tabla 5, tabla 6, tabla 7.

Tabla 3: Personal Involucrado: Administrador del Sistema.

Nombre	Administrador del Sistema.
Rol	Administrar el Sistema.
Categoría Profesional	Ingeniería.
Responsabilidades	Administración y manejo del sistema.
Información de Contacto	Universidad Católica de Pereira.
Aprobación	

Fuente: Elaboración propia.

Tabla 4: Personal Involucrado: Director del Proyecto.

Nombre	Juan Carlos Blandón Andrade.
Rol	Director del Proyecto.
Categoría Profesional	Ingeniero de Sistemas, M.Sc, Ph.D.
Responsabilidades	Director del proyecto.
Información de Contacto	Universidad Católica de Pereira.
Aprobación	

Fuente: Elaboración propia.

Tabla 5: Personal Involucrado: Desarrollador del Proyecto.

Nombre	Diego Mauricio Henao Zuleta
Rol	Desarrollador del Proyecto
Categoría Profesional	Estudiante de Ingeniería de Sistemas y Telecomunicaciones
Responsabilidades	Desarrollo del proyecto
Información de Contacto	Universidad Católica de Pereira
Aprobación	

Fuente: Elaboración propia.

Tabla 6: Personal Involucrado: Usuario.

Nombre	Estudiantes de Ingeniería Industrial
Rol	Usuario
Categoría Profesional	Conocimientos mínimos en informática
Responsabilidades	Manejo correcto del sistema
Información de Contacto	Universidad Católica de Pereira
Aprobación	

Fuente: Elaboración propia.

Tabla 7: Personal Involucrado: Evaluadores.

Nombre	Evaluadores
Rol	Usuario
Categoría Profesional	Conocimientos mínimos en informática
Responsabilidades	Manejo correcto del sistema
Información de Contacto	Universidad Católica de Pereira
Aprobación	

Fuente: Elaboración propia.

Tabla 8: Personal Involucrado: Comité Curricular.

Nombre	Comité Curricular
Rol	Usuarios
Categoría Profesional	Conocimientos mínimos en informática
Responsabilidades	Manejo correcto del sistema
Información de Contacto	Universidad Católica de Pereira
Aprobación	

Fuente: Elaboración propia.

D. Resumen.

Este Sistema de Información para para la gestión de los proyectos será un aplicativo enfocado para la carrera de Ingeniería Industrial, para el desarrollo de los proyectos de grado de una forma ágil y segura donde se puede gestionar la información de cada uno de los proyectos.

4.1.3. Diseño

A. Perspectiva del Sistema.

Este proyecto ayuda a la Gestión de los proyectos de grado del programa ingeniería industrial de la Universidad Católica de Pereira, este sistema se debe utilizar desde un pc o desde un móvil con internet, lo cual lo hace accesible y mediante un registro se puede acceder al sistema y desde allí gestionar de una manera eficaz los proyectos de grado de los estudiantes.

B. Funcionalidad del Sistema.

El sistema de Información para la gestión de proyectos de grado permite gestionar de una manera eficaz los proyectos, de una forma fácil y lógica.

4.1.4. Módulos y Funcionalidades del Sistema de Información para la Gestión de Proyectos de Grado

✓ Registro:

El usuario se debe de registrar mediante la entrega de una información que quedara almacenada en la base de datos del sistema.

✓ Logueo:

El usuario tendrá de manera fácil los datos que requiere para poder entrar al sistema, se logueará con email y contraseña.

✓ Proyectos:

En esta pestaña se recoge la información que se necesita para crear los proyectos de los estudiantes y estos campos son:

- Modalidad del proyecto.
- Nombre de la línea de investigación.
- Nombre del proyecto.
- Objetivos.
- Objetivos específicos.

✓ Registro de Acta:

Se llevará a cabo por el Director del programa Ingeniería Industrial, es donde se plasmará la información del proyecto de grado y llevará la siguiente información:

- Modalidad de Proyecto
- Línea de Investigación
- Nombre del Proyecto
- Autores del Proyecto
- Asesor
- Objetivo General
- Objetivos específicos
- Cronograma de actividades
- Compromisos de los Estudiantes.
- Firma de estudiantes y tutores.

✓ Cronogramas.

En esta fase del sistema en donde el estudiante del proyecto programa las actividades concernientes a los tiempos de desarrollo del proyecto dándole al proyecto metas a corto y largo plazo, los campos de información necesaria para crear el cronograma son:

- Actividad.
- Fecha inicial.
- Fecha final.

✓ Observaciones.

En este apartado del sistema en donde el tutor o director del programa le especifica al estudiante las observaciones que va encontrando a lo largo del desarrollo del proyecto, dándole al estudiante las pautas para seguir con su proyecto de grado para su posterior aprobación, los campos de información son:

- Nombre del proyecto.
- Observaciones del proyecto.
- Fecha y Hora.

✓ Usuarios.

En este apartado del sistema en donde se encuentra el espacio para el administrador del sistema, en esta pestaña es donde el administrador dará los roles concernientes al sistema, aquí aparecen las personas que están registradas con los siguientes campos de información como:

- Nombre del estudiante.
- Email.
- Roles.

✓ Roles.

Es aquí donde el administrador le da los permisos a todos los implicados en el sistema tanto a tutores como estudiantes, aquí se pueden adicionar o eliminar los permisos que el administrador crea necesarios.

✓ Modalidad.

En este apartado del sistema el administrador creara las modalidades que concerniesen a los proyectos de la carrera ingeniera industrial, estos campos de información para la creación de los cronogramas son:

- Nombre de la modalidad.

Aquí también el administrador puede gestionar los nombres de las modalidades con Editar, Detalles y Eliminar.

✓ Línea de Investigación.

En este apartado del sistema el administrador se encargará de ingresar los campos necesarios para la creación de la línea de investigación del proyecto, este campo de información es:

- Nombre de la línea de investigación del proyecto.

Aquí también el administrador puede gestionar los nombres de las modalidades con Editar, Detalles y Eliminar.

✓ Nota del Proyecto.

En este apartado es donde el tutor se encargara de poner la nota al proyecto de grado cuando este cumpla con los objetivos, este campo de información es:

- Nota del Proyecto.

C. Características de los usuarios.

Tabla 9: Características de los Usuarios.

Tipo De Usuario	Administrador
Formación	Persona que tenga habilidades mínimas en sistemas informáticos
Habilidades	Conocimientos básicos de sistemas operativos, digitación y búsquedas
Actividades	Manejo del sistema informático

Fuente: Elaboración propia.

Tabla 10: Características del Usuario.

Tipo De Usuario	Tutor
Formación	Persona que tenga habilidades mínimas en sistemas informáticos
Habilidades	Conocimientos básicos de sistemas operativos, digitación y búsquedas
Actividades	Revisar y resolver las correcciones hechas para los proyectos, registro y C.R.U.D de datos

Fuente: Elaboración propia.

Tabla 11: Características de los Usuarios.

Tipo De Usuario	Estudiantes
Formación	Básica en internet
Habilidades	Conocimientos básicos de sistemas operativos, digitación y búsquedas
Actividades	Registro y C.R.U.D de datos

Fuente: Elaboración propia.

4.1.5. Requisitos Específicos.

A. Captura de Requisitos.

En la captura de requisitos se implementó mediante la información que se le solicitó al interesado, permitió ver de manera objetiva el propósito y necesidades del aplicativo.

B. Requisitos comunes de las Interfaces.

- Interfaces de usuario
 - Entrada: Teclado y Mouse
 - Salida: Monitor
 - Software: Navegador Web.
- Interfaces de Hardware.
 - Entrada: teclado, Mouse.
 - Salida: Monitor.

- Interfaces de Software.
 - Tipo de base: Relacional con enfoque a objetos.
 - Arquitectura: MVC.
 - Motor base de datos: MySql.
 - Entorno de Desarrollo: Visual Studio 2017.
 - Lenguaje de programación: C#.
 - Patrón de diseño: MVC (modelo-Vista-Controlador).
 - Compatibilidad: Multiplataforma.

C. Interfaces de Comunicación.

- Conexión directa.

D. Requisitos Funcionales.

- R1: el sistema debe permitir ingresar campos para registro.
- R2: El sistema debe permitir ingresar un Email y contraseña para ingreso al sistema.
- R3: El sistema debe permitir gestionar los tutores para cada uno de los proyectos de los alumnos.
- R4: El sistema debe permitir hacer un seguimiento a los proyectos de grado por medio de observaciones hechas por el tutor.
- R5: El sistema debe permitir que los tutores tengan un permiso administrativo para gestionar las actas.
- R6: El sistema debe permitir registrar las fechas de seguimiento para saber si el tutor hace su respectivo trabajo.
- R7: El sistema debe permitir agregar un campo para la nota del tutor para saber si el alumno cumplió con los objetivos del proyecto.
- R8: El sistema debe permitir hacer un 1º Informe ejecutivo donde se cumplan los objetivos.
- R9: El sistema debe permitir un 2º Informe ejecutivo para la primera sustentación.
- R10: El sistema debe permitir un 3º Informe ejecutivo para la sustentación final.

- R11: El sistema debe permitir la asignación de jurado a las actas de grado.

E. Requisitos no Funcionales.

- R1: El sistema llevará un logo de la Universidad Católica de Pereira.
- R2: El sistema llevará la letra Arial de un tamaño de 12.
- R3: El sistema debe de operar adecuadamente cuando se tiene usuarios masivos en el sistema.
- R4: El sistema debe de contar con manuales de usuario para un mejor aprovechamiento del sistema.
- R5: La aplicación web debe poseer un diseño Responsive a fin de garantizar que se pueda ver en cualquier pc, Tablet o celular inteligente.
- R6: El sistema debe permitir tener la información a buen resguardo.
- R7: El tiempo de aprendizaje del sistema debe ser menor a 3 horas.
- R8: El sistema debe permitir restringir el acceso a personas que no pertenecen al programa de Ingeniería Industrial.

4.2. Listado de Requerimientos

- Registro: El sistema debe permitir registrar el Tutor y al Alumno. Para hacer un registro exitoso es necesario solicitar la siguiente información:
 - ✓ Nombre del Estudiante.
 - ✓ Cedula.
 - ✓ Email.
 - ✓ Teléfono.
 - ✓ Cedula.
 - ✓ Contraseña.
 - ✓ Repetir Contraseña.
- Logueo: El sistema debe permitir iniciar sesión al tutor y al estudiante. Para iniciar sesión se necesita:
 - ✓ Email.
 - ✓ Password.

- Adición de tutores: El sistema debe permitir adicionar los tutores para cada uno de los proyectos de los estudiantes.
- Seguimientos de proyectos: El sistema debe permitir hacer un seguimiento a los proyectos de grado por medio de observaciones hechas por el tutor.
- Actas de Grado: El sistema debe permitir que el administrador conceda el permiso administrativo para revisar las actas.
- Registro de Fechas: El sistema debe permitir registrar las fechas de seguimiento para saber si el tutor hace su respectivo trabajo.
- Nota Proyecto: El sistema debe permitir agregar un campo para la nota del tutor para saber si el alumno cumplió con los objetivos del proyecto.
- 1º Informe Ejecutivo: este informe es donde se estipula que el estudiante termino con los objetivos del proyecto.
- 2º Informe Ejecutivo: es donde al estudiante se le informará de la primera sustentación del proyecto.
- 3º Informe Ejecutivo: es donde al estudiante se le informará de la sustentación final del proyecto.
- El sistema debe permitir asignar los jurados a los trabajos de grado, para luego generar las actas respectivas.

Tabla 12: Requisito 1. Registro en el Sistema.

ID RF#	RF1	ESTADO	SOLICITADO
NOMBRE	Registro al sistema	Prioridad	Alta
DESCRIPCIÓN	El sistema debe permitir ingresar los campos que requiere para registro.		
ENTRADAS	<ul style="list-style-type: none"> - Nombre y Apellidos - Cedula - Teléfono - Correo Electrónico - Contraseña - Confirmar Contraseña. 		
SALIDAS	Vista del usuario correspondiente.		

RESTRICCIONES	Todos los campos del registro son necesarios, solo para estudiantes y tutores.		
OBSERVACIONES	El alumno debe registrarse para poder ingresar al sistema, al igual que tutores		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 13: Requisito 2. Logueo en el Sistema.

ID RF#	RF2	ESTADO	SOLICITADO
NOMBRE	Logueo al sistema	Prioridad	Alta
DESCRIPCIÓN	El sistema debe permitir ingresar los campos que requiere para loguearse al sistema.		
ENTRADAS	1. Email 2. Contraseña		
SALIDAS	Se dirige usuarios a la ventana principal		
RESTRICCIONES	Solo para estudiantes, tutores y administrador.		
OBSERVACIONES	Estudiantes, Tutores y administrador deben de llenar los campos necesarios para loguearse en el sistema		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia

Tabla 14: Requisito 3. Gestión de Tutores.

ID RF#	RF3	ESTADO	SOLICITADO
NOMBRE	Gestión de Tutores	Prioridad	Alta
DESCRIPCIÓN	El sistema debe permitir agregar el tutor para cada uno de los proyectos de grado.		
ENTRADAS	El sistema debe permitir que el administrador se logue con los siguientes campos: 1. Email. 2. Contraseña.		
SALIDAS	Sale a la vista de actas donde el administrador podrá		

	asignar el tutor al proyecto.		
RESTRICCIONES	Solo ingresara el administrador con su email y contraseña.		
OBSERVACIONES	Si el administrador ingresa datos incorrectos el sistema, saldrá mensaje de error de ingreso.		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 15: requisito 4. Seguimientos a Los Proyectos.

ID RF#	RF4	ESTADO	SOLICITADO
NOMBRE	Seguimientos a proyecto de grado	Prioridad	Alta
DESCRIPCIÓN	El sistema debe permitir que tanto estudiantes y Tutores puedan hacer seguimiento a los proyectos de grado por medio de un cronograma u observaciones del tutor.		
ENTRADAS	Se deben de loguear estudiantes y tutores para ingresar al sistema.		
SALIDAS	Sale a la vista principal y luego a proyectos		
RESTRICCIONES	Solo podrán Ingresar estudiantes, tutores y el administrador.		
OBSERVACIONES			
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 16: Requisito 5. Permisos del Sistema para Tutor.

ID RF#	RF5	ESTADO	SOLICITADO
NOMBRE	Permiso administrativo para el tutor.	Prioridad	Alta
DESCRIPCION	El sistema debe permitir el ingreso de fechas para que se pueda hacer seguimiento al trabajo del respectivo tutor.		
ENTRADAS	El administrador del programa debe de loguearse con los campos email y contraseña.		
SALIDAS	Se ingresa a la vista principal del administrador.		

RESTRICCIONES	Solo podrá ingresar el Director del Programa Ingeniería Industrial y tutores.		
OBSERVACIONES			
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 17: Requisito 6. Registro de Fechas.

ID RF#	RF6	ESTADO	SOLICITADO
NOMBRE	Registro de Fechas	Prioridad	Alta
DESCRIPCION	El sistema debe permitir ingresar las fechas para poder hacer seguimiento al trabajo del tutor.		
ENTRADAS	El administrador del programa debe de loguearse para gestionar los proyectos y observar si el tutor está haciendo el trabajo en las fechas estipuladas.		
SALIDAS	Ingresa a la vista de principal para luego abrir pestaña de observaciones		
RESTRICCIONES	Solo podrá ingresar el Director del Programa Ingeniería Industrial y tutores.		
OBSERVACIONES	Solo lo hará el administrador.		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 18: Requisito 7. Registro de Nota para Tutor.

ID: RF#	RF7	ESTADO	SOLICITADO
NOMBRE	Registro de nota	Prioridad	Alta
DESCRIPCION	El sistema debe permitir que el tutor pueda asignarle una nota al proyecto de grado para saber si el alumno cumplió con los objetivos del proyecto.		
ENTRADAS	El tutor deberá de hacer un seguimiento al proyecto de grado para ingresar la nota.		
SALIDAS	Sale a la vista de Actas de grado donde pondrá nota al proyecto		
RESTRICCIONES	Solo podrá acceder el tutor y el administrador del sistema		

OBSERVACIONES	El tutor es el encargado de poner la nota del proyecto		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 19: Requisito 8. 1º Informe Ejecutivo.

ID: RF#	RF8	ESTADO	SOLICITADO
NOMBRE	1º Informe Ejecutivo	Prioridad	Alta
DESCRIPCION	El sistema debe permitir que el administrador pueda hacer un primer informe ejecutivo donde conste que cumplió con los objetivos del proyecto.		
ENTRADAS	El administrador del sistema deberá realizar seguimiento al primer informe ejecutivo.		
SALIDAS	Ingresar a la vista de actas donde el administrador podrá hacer seguimiento del Informe para saber si se cumplen con los objetivos.		
RESTRICCIONES	Solo podrá acceder el administrador del sistema		
OBSERVACIONES	El administrador del sistema es el encargado de hacer seguimiento del informe ejecutivo.		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 20: Requisito 9. 2º Informe Ejecutivo.

ID: RF#	RF9	ESTADO	SOLICITADO
NOMBRE	2º Informe Ejecutivo	Prioridad	Alta
DESCRIPCION	El sistema debe permitir que el administrador del sistema pueda hacer un segundo informe ejecutivo donde se autoriza al alumno para hacer su primera sustentación.		
ENTRADAS	El administrador del sistema deberá de hacer seguimiento al segundo informe ejecutivo.		
SALIDAS	Sale a la vista de actas donde el administrador del sistema podrá hacer seguimiento del Informe para saber si el alumno es apto para sustentar el proyecto.		
RESTRICCIONES	Solo podrá acceder el administrador del sistema		

OBSERVACIONES	El administrador del sistema es el encargado de hacer seguimiento del informe ejecutivo.		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 21: Requisito 10. 3º Informe Ejecutivo.

ID: RF#	RF10	ESTADO	SOLICITADO
NOMBRE	3º Informe Ejecutivo	Prioridad	Alta
DESCRIPCION	El sistema debe permitir que el administrador del sistema pueda hacer un tercer informe ejecutivo donde se autoriza al alumno para hacer su sustentación final.		
ENTRADAS	El administrador del sistema deberá de hacer seguimiento al tercer informe ejecutivo.		
SALIDAS	Sale a la vista de actas donde el administrador del sistema podrá hacer seguimiento del Informe para saber si el alumno es apto para sustentación final del proyecto.		
RESTRICCIONES	Solo podrá acceder el administrador del sistema		
OBSERVACIONES	El administrador del sistema es el encargado de hacer seguimiento del informe ejecutivo.		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

Tabla 22: Requisito 11. Asignación de Jurado.

ID: RF#	RF11	ESTADO	SOLICITADO
NOMBRE	Asignación de Jurado para los proyectos	Prioridad	Alta
DESCRIPCION	El sistema debe permitir que el administrador del sistema pueda asignar los jurados para el proyecto de grado.		
ENTRADAS	El administrador asignara los nombres del jurado al acta del proyecto.		
SALIDAS	Sale a la vista de actas donde el administrador pueda ingresar los jurados del proyecto.		
RESTRICCIONES	Solo podrá acceder el administrador del sistema		

OBSERVACIONES	El administrador del sistema es el encargado de hacer seguimiento del informe ejecutivo.		
ENCARGADO	Desarrollador	Solicitante	Facultad de Ciencias Básicas e Ingeniería

Fuente: Elaboración propia.

4.3. Requisitos No Funcionales

Requisito de diseño: Este sistema se desarrolla para el programa de Ingeniería Industrial de la Facultad de Ciencias Básicas e Ingeniería de la Universidad Católica de Pereira.

- Seguridad

Este sistema es desarrollado con el IDE de Visual Studio por medio de la estructura de desarrollo MVC, programado con C#, para el Frontend y Backend se utiliza el ASP.Net esto permite que la seguridad del sistema sea eficiente para resguardar la información de los usuarios.

- Disponibilidad

Dado los requisitos del usuario y las condiciones del sistema de información, establece que la disponibilidad del sistema debe de estar disponible al 100% y con la disponibilidad de recuperación de fallas alrededor del 95%, Esto deberá ser planteado por el desarrollador para dar cumplimiento a este requisito.

- Mantenimiento

El rendimiento del sistema y el de la base de datos estará a cargo el administrador y las personas encargadas del sistema para poder gestionar la información de los proyectos y verificar en cada determinado tiempo se gestiona la información para que el sistema no se vea saturado con archivos temporales o en su caso eliminar la información que no sea necesaria para el buen funcionamiento del sistema.

Para realizar estas tareas de mantenimiento se necesita:

- Realizar Backups: Datos de Información de los proyectos y de usuarios.
- Eliminar Errores.
- Eliminar virus que pueda tener el sistema de Información.
- Corregir Bugs.
- Borrar la cache del sistema.

- Portabilidad.
 - o El Sistema al ser diseñado, cuenta con el sistema Responsive que hace que el sistema de información pueda ser utilizado en diferentes tipos de aparatos tecnológico tales como computadoras personales, Tablet, Teléfonos Inteligentes.

4.3.1. Análisis

Tanto el desarrollador como el administrador del sistema tienen un papel activo en la ingeniería de requisitos “un conjunto de actividades que son denominadas análisis”. El Administrador intenta replantear un sistema confuso, a nivel de descripción de datos, funciones y comportamiento, en detalles concretos. El desarrollador actúa como interrogador, como consultor, como persona que resuelve problemas y como negociador. El objetivo principal de "analizar los requisitos del sistema" es mejorar los requisitos, normalmente mediante la elaboración de modelos que permitan identificar posibles problemas y que detallen más la solución propuesta para resolver las necesidades de negocio de clientes y usuarios, incluyendo la arquitectura lógica del sistema y sus interfaces de usuario y de servicios [30].

4.4. Diagrama de Flujo

4.4.1. Descripción del Diagrama de Flujo

En la
Figura 10, se muestra el diagrama de flujo del sistema de Información.

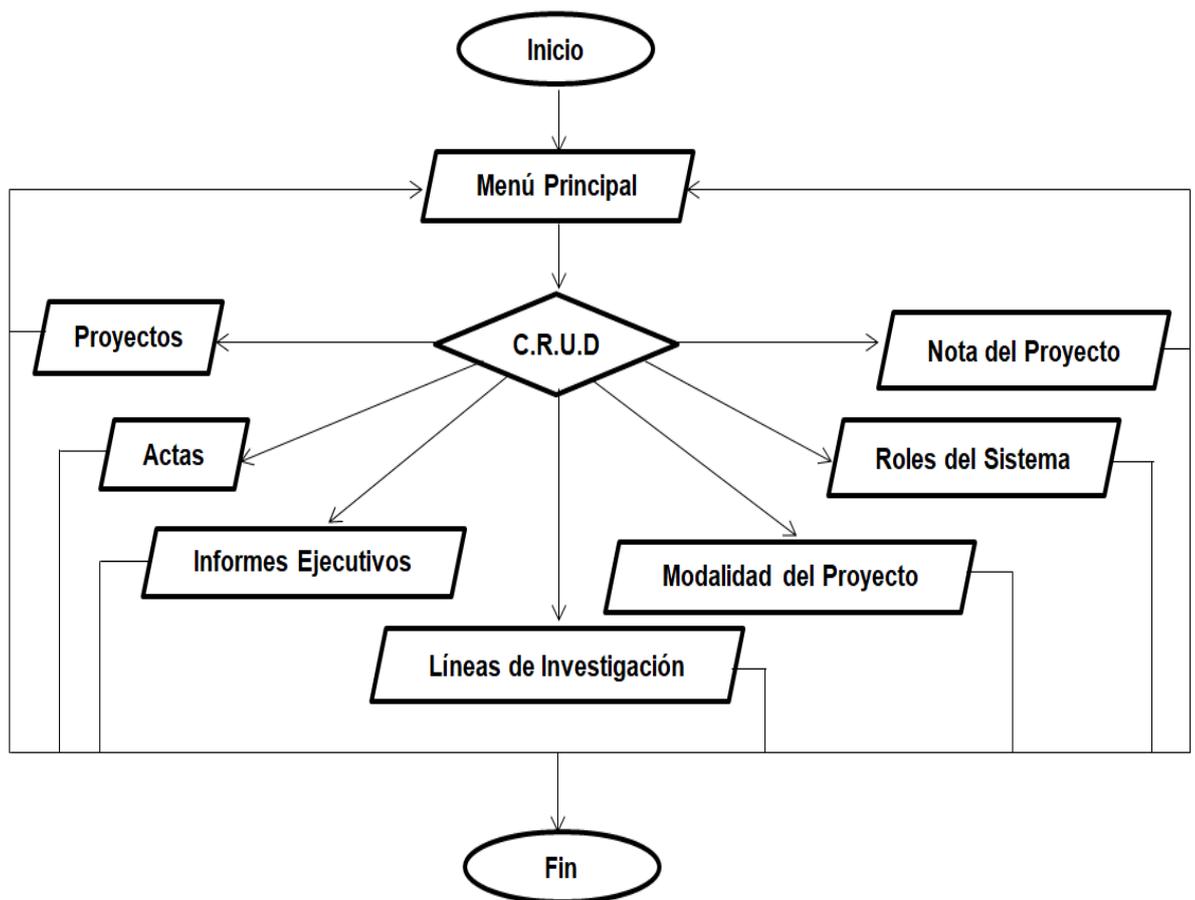


Figura 10: Diagrama de Flujo del Sistema.
Fuente: Elaboración propia.

- Registro al sistema: hace referencia a la acción de registrar con el Email y Password dado por un usuario ya registrado, incluye subprocesos como:
 - Verificación de la base de datos.
- Logueo del usuario: hace referencia al Logueo de todos los datos de un usuario nuevo, el cual podrá utilizar la aplicación, incluye subprocesos:
 - Digitación de campos requeridos.
 - Inclusión de información a la Base de datos.
- Registro de Actas de Inicio: hace referencia al ingreso de los datos necesarios para registrar un acta de Inicio de proyecto.
 - Digitación de campos requeridos.
 - Validación en la base de datos.

- **Proyectos:** hace referencia a la selección del proyecto de grado, Cronogramas y observaciones, incluye subprocesos:
 - Validación en la base de datos.
 - Digitación de campos requeridos.
- **Administración:** hace referencia a la selección de usuarios, Modalidad y línea de investigación, incluye subprocesos:
 - Validación en la base de datos.
 - Digitación de campos requeridos.
 - Asignación de parámetros en la base datos.
- **Nota del Proyecto:** hace referencia a la selección del proyecto para calificar por parte del tutor al que corresponde el proyecto, para saber si ya se cumplió con todos los requisitos necesarios para finalizar el proyecto y posterior nota.
 - Validar si existe el proyecto.
 - Digitación de campo requerido.
 - Guardar nota del proyecto.

4.5. Casos de Uso

En la Figura 11, se representa el diagrama de casos de uso nivel cero el cual se diseñó para la creación del sistema.

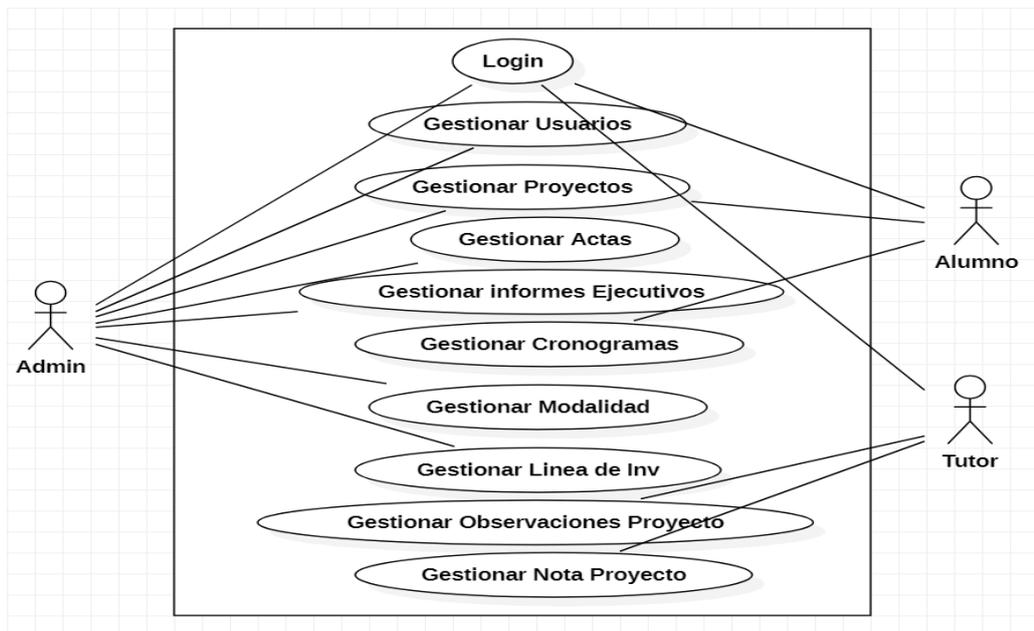


Figura 11: Caso de Uso General.
Fuente: Elaboración propia.

En la Figura 12, se presenta el caso de uso diseñado para la operación de registro en el sistema.

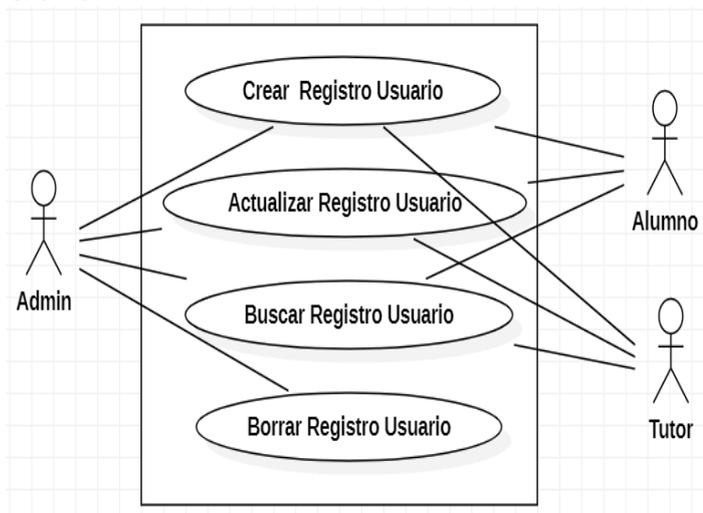


Figura 12: Caso de Uso Registrar Usuario.
Fuente: Elaboración propia.

En la Figura 13, se presenta el caso de uso diseñado para la operación de gestionar proyecto. En este apartado es donde tanto administrador como estudiante podrán crear el proyecto.

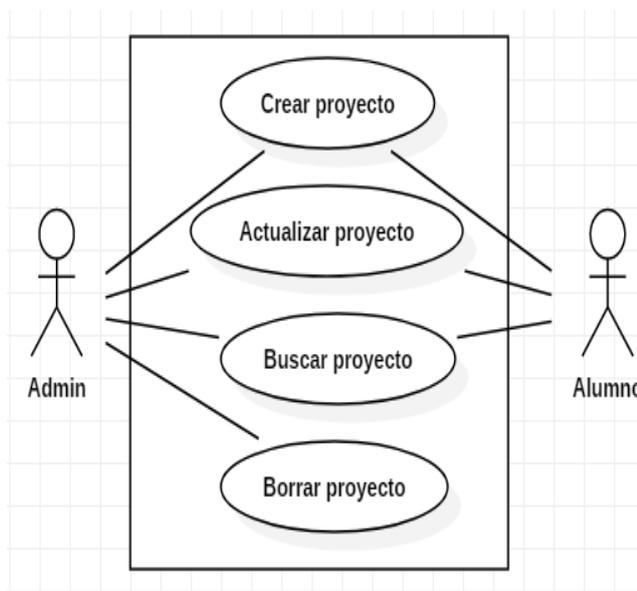


Figura 13: Caso de Uso Gestionar Proyecto.
Fuente: Elaboración propia.

En la Figura 14, se presenta el caso de uso diseñado para la operación de gestionar acta del proyecto, en este caso de uso es donde el administrador podrá crear el acta inicial del proyecto.

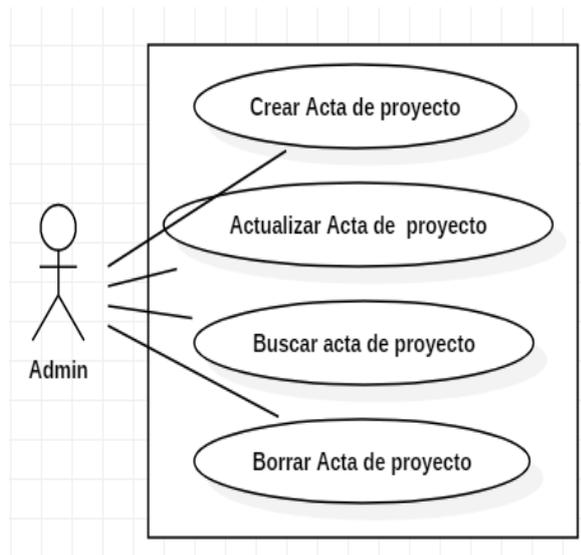


Figura 14: Caso de Uso Gestionar Actas.
Fuente: Elaboración propia.

En la Figura 15, se presenta el caso de uso diseñado para la operación de gestionar informes ejecutivos para el proyecto, en este caso de uso el administrador podrá crear un informe a partir del acta creada.

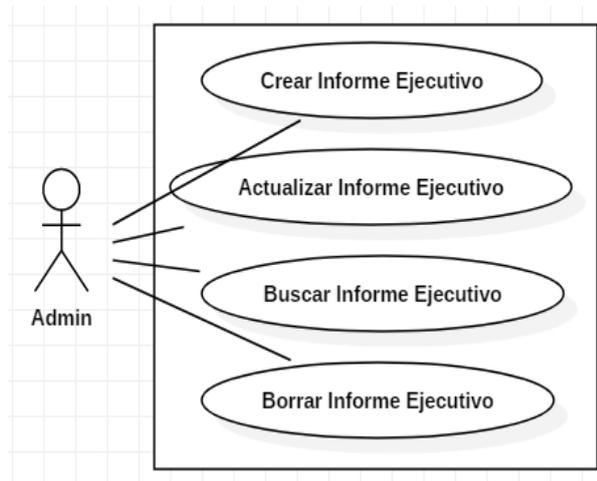


Figura 15: Caso de Uso de Gestionar Actas.
Fuente: Elaboración propia.

En la Figura 16, se presenta caso de uso diseñado para la operación de gestionar modalidad del proyecto, en este caso de uso es donde el Administrador creara las modalidades que corresponden al programa Ingeniería Industrial.

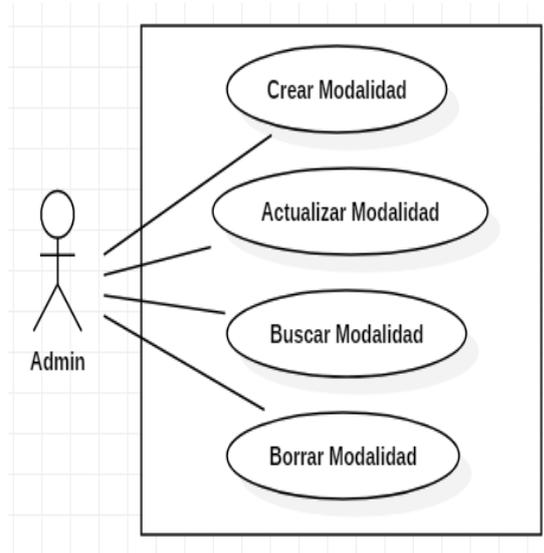


Figura 16: Caso De Uso Gestionar Modalidad.
Fuente: Elaboración propia.

En la Figura 17, se presenta el caso de uso para la operación de gestionar línea de investigación del proyecto, en este caso de uso es donde el administrador podrá crear las líneas de investigación que correspondan al programa Ingeniería Industrial.

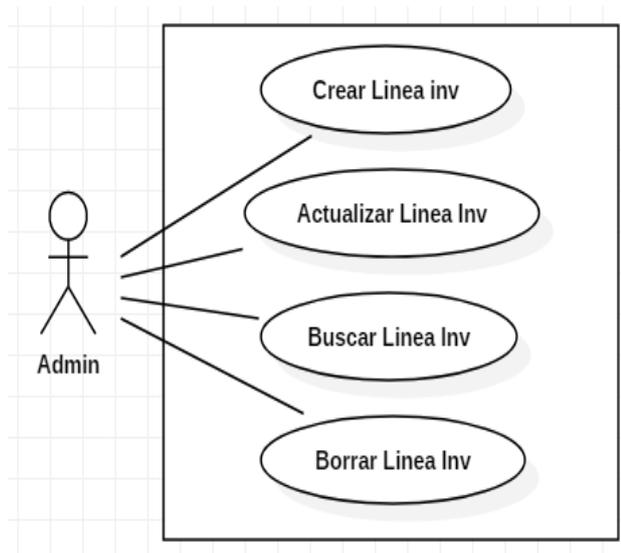


Figura 17: Caso de Uso Gestionar Línea de Investigación.
Fuente: Elaboración propia.

En la Figura 18, se presenta el caso de uso para la operación de gestionar permisos administrativos del sistema, en este caso de uso es donde el administrador podrá darles los roles pertinentes a los usuarios.

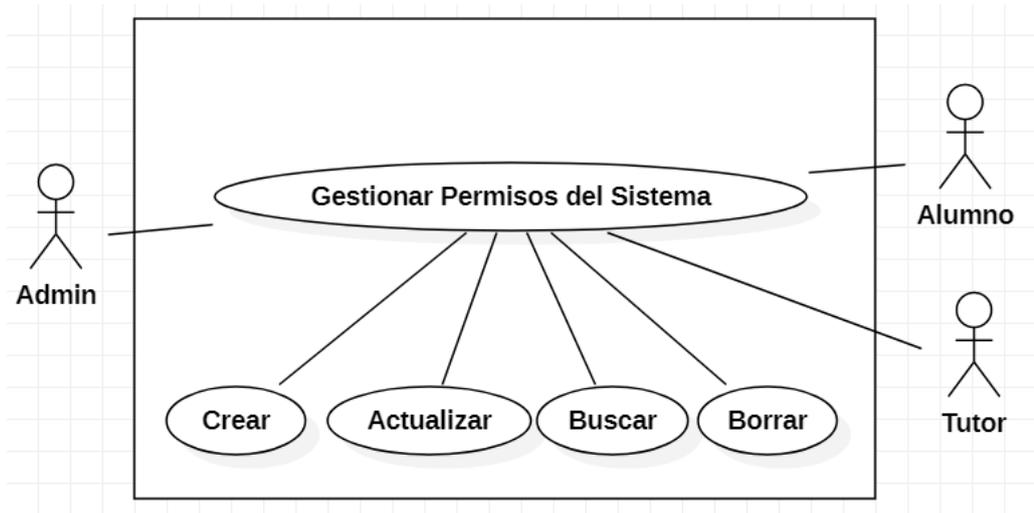


Figura 18: Caso de Uso Gestionar Permisos del Sistema.
Fuente: Elaboración propia.

En la Figura 19, se presenta el caso de uso para la operación de gestionar cronogramas para el proyecto, en este caso de uso el estudiante podrá crear el cronograma que seguirá el proyecto durante todo el proceso de culminación del mismo.

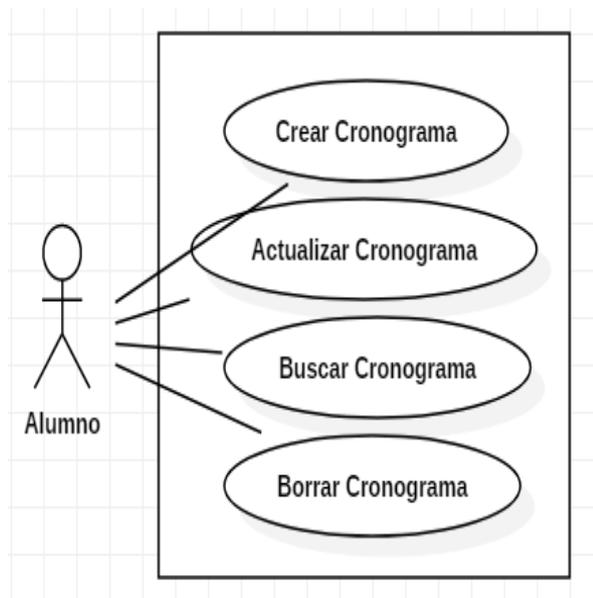


Figura 19: Caso de Uso Gestionar Cronograma.
Fuente: Elaboración propia.

En la Figura 20, se presenta el caso de uso para la operación de gestionar la nota para el proyecto, en este caso de uso es donde el tutor podrá poner la nota al proyecto cuando este haya culminado.

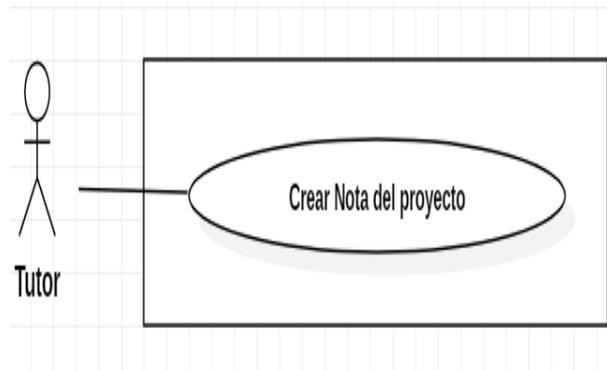


Figura 20: Caso de Uso Gestionar Nota Proyecto.
Fuente: Elaboración propia.

En la Figura 21, se presenta el caso de uso para la operación de gestionar observaciones del proyecto, en este caso de uso el tutor podrá hacer las observaciones pertinentes al proyecto, con esto ayudará a que el estudiante pueda desarrollar las observaciones hechas por el tutor.

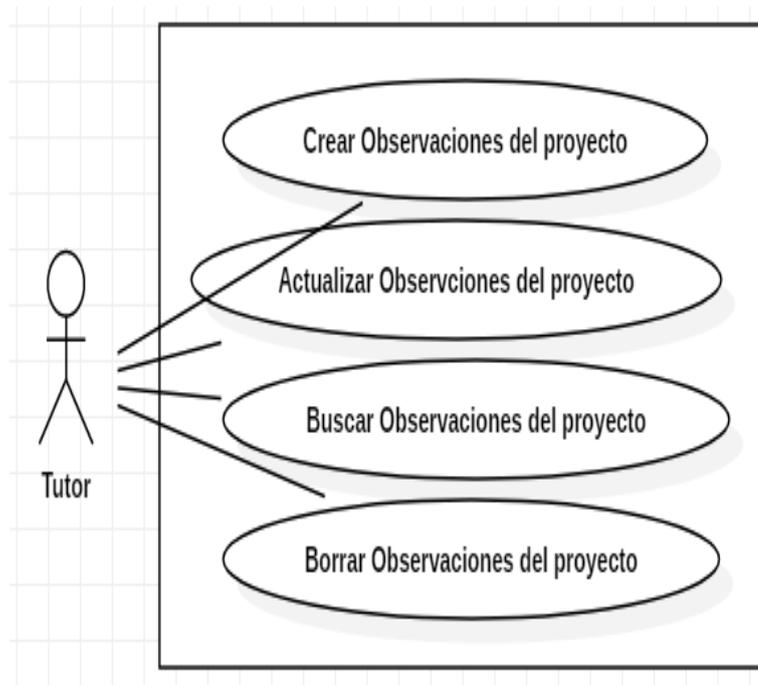


Figura 21: Caso de Uso Gestionar Observaciones.
Fuente: Elaboración propia.

4.5.1. Descripción de Casos de Uso.

Desde la tabla 23 hasta la tabla 46, se presentan las respectivas especificaciones de los casos de uso modelados para el sistema de información.

Tabla 23: Descripción Caso de Uso Registro.

C.U-01	- Registrar Usuarios
Actores	- Administrador, Tutor, Alumno.
Tipo	- Básico.
Propósito	- Admitir registro de usuario para ingresar al sistema.
Resumen	- Ingreso al sistema
Pre-condiciones	- Los Usuarios no deben de estar registrados en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El usuario debe de ingresar los campos de registro: <ul style="list-style-type: none"> - Nombres. - Apellidos. - Email. - Teléfono. - Cedula. - Contraseña - Repetir Contraseña. 2. El usuario hace clic en el botón registrar usuario. 3. El sistema validó la información ingresada. 4. El sistema ingresará a la página principal correspondiente de cada usuario.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos</p> <p>Descripción: Si en el paso 3 el usuario no ingresa un campo:</p> <ol style="list-style-type: none"> 1. El sistema muestra mensaje "campos incompletos". 2. El usuario debe de volver al paso 1. <p>Flujo Alt 2: Usuario ya registrado.</p> <p>Descripción: usuario registrado en el sistema.</p> <ol style="list-style-type: none"> 1. En caso de que un estudiante ya este registrado el sistema mostrará "Estudiante ya registrado".

Fuente: Elaboración propia.

Tabla 24: Descripción Caso de uso de logueo.

C.U-02	- Loguear Usuarios
Actores	- Administrador, Tutor, Alumno
Tipo	- Básico.
Propósito	- El sistema debe permitir que el usuario pueda loguearse para entrar al sistema.
Resumen	- El sistema debe permitir que el usuario pueda ingresar los datos de logueo que requiere para entrar al sistema.
Pre-condiciones	- El usuario debería de estar registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El usuario debe ingresar el correo electrónico en el campo email. 2. El usuario debe de ingresar password en el campo contraseña 3. El usuario dará clic en el botón “Iniciar Sesión. 4. El Sistema valida la información ingresada. 5. El usuario ingresa a la página principal correspondiente.
Flujo Alterno	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: si en el paso 3 del flujo normal los datos de logueo son incompletos.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará “Email o password incorrectos”. 2. El usuario debe de volver al paso 1.

Fuente: Elaboración propia.

Tabla 25: Descripción Caso de Uso Crear Proyecto.

C.U-03	- Crear proyectos para el administrador y alumno
Actores	- Administrador, Alumno
Tipo	- Básico.
Propósito	- Le permite al administrador y al alumno crear proyectos de grado
Resumen	- El Sistema debe permitir al administrador y al alumno que al ingresar los campos necesarios se deba crear los proyectos de grado.
Pre-condiciones	- El administrador y el alumno deberán estar registrados en la base de datos.

<p>Flujo Principal</p>	<ol style="list-style-type: none"> 1. El administrador y el alumno deberán ingresar los campos de logueo: <ul style="list-style-type: none"> - Email - Contraseña. 2. El administrador y el alumno deberán darle clic al botón Iniciar Sesión. 3. El administrador y el alumno ingresan a la vista que corresponden a cada uno. 4. El administrador y el usuario ingresarán a la vista crear proyecto con los siguientes campos: <ul style="list-style-type: none"> - Nombre del proyecto - Objetivos generales - Objetivos Específicos - Nombre de la modalidad - Nombre de la Línea de Investigación 5. El administrador y el alumno le darán al botón crear. 6. El sistema valida la Información y crea el proyecto en la base de datos. 7. regresa a la vista proyectos.
<p>Flujo Alternativo</p>	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El usuario debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Campos Incompletos crear proyecto. Descripción: Si en el paso 5 del flujo normal el usuario no ingresa un campo.</p> <ol style="list-style-type: none"> 1. El sistema mostrará mensaje "Campo incompleto. 2. El usuario debe volver al paso 4 del flujo normal. <p>Flujo Alt 3: No guardar Información. Descripción: Si en el paso 5 del flujo normal el estudiante no guarda información.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario no le dé clic al botón guardar se perderá la información. 2. El usuario no podrá ver el proyecto si no se guarda la información. 2. El usuario deberá volver al paso 4 del flujo normal.

Fuente: Elaboración propia.

Tabla 26: Descripción Caso de Uso Actualizar Proyecto.

C.U-04	- Actualizar proyectos para el administrador y Alumno
Actores	- Administrador, Alumno
Tipo	- Básico.
Propósito	- Le permite al Administrador y al alumno actualizar la información que hay en los proyectos de grado.
Resumen	- El Sistema me debe permitir al administrador y al alumno actualizar la información de los proyectos de grado.
Pre- condiciones	1. El administrador y al alumno deberán estar registrados en la base de datos
Flujo Principal	<p>1 El administrador y al alumno deben de ingresar los campos para iniciar sesión que son:</p> <ul style="list-style-type: none"> - Email. - Contraseña. <p>2. El administrador y el alumno darán clic al botón iniciar sesión.</p> <p>3. El sistema validó la información.</p> <p>4. El administrador y alumno ingresan a sus vistas respectivas en el sistema.</p> <p>5. El administrador y el alumno ingresan a proyectos y eligen proyecto a actualizar.</p> <p>6. El administrador y el alumno ingresan datos para actualizar el proyecto.</p> <p>7. El administrador y el alumno le dan al botón guardar.</p> <p>8. El sistema validó la información y la guarda.</p> <p>9. El administrador y el alumno regresan a sus respectivas vistas.</p>
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El usuario debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Campos Incompletos al crear proyecto. Descripción: Datos incompletos para crear un proyecto.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal campos de crear proyecto el sistema mostrará "ingresar información al campo

	<p>incompleto”.</p> <p>2. El usuario debe volver al paso 4 flujo normal.</p> <p>Flujo Alt 3: No guardar Información</p> <p>Descripción: Si en el paso 7 del flujo normal el usuario no guarda la Información Actualizada.</p> <p>1. En caso de no ingresar algún campo el sistema mostrará “campos incompletos”</p> <p>2. El usuario debe de volver al paso 4.</p> <p>3. Si el usuario no le da clic al botón guardar se perderá la información.</p> <p>4. el usuario debe de volver al paso 4 del flujo normal.</p>
--	--

Fuente: Elaboración propia.

Tabla 27: Descripción de Caso de Uso Buscar Proyecto.

C.U-05	- Buscar proyectos para el administrador, Alumno
Actores	- Administrador, Alumno.
Tipo	- Básico.
Propósito	- Le permite al Administrador y al alumno buscar el proyecto en el sistema.
Resumen	- El Sistema le debe permitir al administrador y al alumno buscar la información de los proyectos de grado.
Pre-condiciones	1. El administrador y al alumno deberán estar registrados en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador y al alumno deberán ingresar campos para ingresar al sistema, estos son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El sistema validó la información de logueo. 3. El administrador y alumno ingresan a sus respectivas vistas de proyectos. 4. El administrador y el alumno ingresan a proyectos y buscan el mismo. 5. El administrador y el alumno ingresan información del proyecto y guardan. 6. El sistema validó la información. 7. El administrador y el alumno regresan a sus respectivas vistas de proyectos.

Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo. 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El usuario debe de volver al paso 1 del flujo normal.</p> <p>Flujo Alt 2: Proyecto no guardado. Descripción: el usuario no encuentra el proyecto. 1. en caso de que el usuario no encuentre el proyecto este deberá volver a crearlo. 2. Debe de volver al paso donde se crean los proyectos.</p>
--------------------------	--

Fuente: Elaboración propia.

Tabla 28: Descripción Caso de Uso Borrar Proyecto.

C.U-06	- Borrar proyecto para el administrador.
Actores	- Administrador.
Tipo	- Básico.
Propósito	- El sistema debe permitir al administrador del sistema poder borrar cualquier proyecto.
Resumen	- El administrador se encargara de borrar los datos o proyectos que se requiera para esta acción.
Pre-condiciones	- Se requiere que el administrador este registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador deberá ingresar campos para logueo que son: <ul style="list-style-type: none"> - Email - Password o contraseña. 2. El administrador dará clic en el botón iniciar sesión. 3. El sistema validó la información. 4. El administrador ingresa a la vista proyectos. 5. El administrador elije el proyecto a borrar. 6. El administrador dará clic al botón eliminar. 7. El sistema validó la información y borra el proyecto del sistema. 8. El administrador vuelve a la ventana proyectos.
	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo. 1. En caso de que el usuario ingrese mal el email y</p>

Flujo Alternativo	<p>contraseña, el sistema mostrará "Email o password incorrectos".</p> <p>2. El usuario debe de volver al paso 1 del flujo normal.</p> <p>Flujo Alt 2: Borrar Proyecto equivocado.</p> <p>Descripción: Si en el paso 5 el administrador borra un proyecto equivocado.</p> <p>1. El administrador deberá crear de nuevo el proyecto que borro.</p> <p>2. El administrador deberá volver al paso de crear proyecto.</p>
--------------------------	---

Fuente: Elaboración propia.

Tabla 29: Descripción Caso de Uso Crear Acta del Proyecto.

C.U-07	- Crear actas del proyecto para el administrador.
Actores	- Administrador.
Tipo	- Básico.
Propósito	- Permitir al administrador poder crear el acta que del proyecto.
Resumen	- El sistema debe permitir que el administrador pueda crear el acta que se requiere para el proyecto.
Pre-condiciones	- Se requiere que el administrador esté registrado en la base de datos del sistema.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador ingresará los campos de logueo que son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El administrador dará clic al botón iniciar sesión. 3. El sistema validó la información. 4. El administrador ingresa en la vista principal y le da al botón crear Actas. 5. El sistema valida la petición. 6. El administrador ingresa a la vista crear actas, ingresa los campos para crear el acta y estos son: <ul style="list-style-type: none"> - Nombre del proyecto. - Observaciones del proyecto. - Fecha del acta. 7. El administrador dará clic al botón crear.

	<p>8. El sistema validó la información y si es correcta crea el acta.</p> <p>9. Al crear el acta el administrador deberá añadir la siguiente información al acta.</p> <ul style="list-style-type: none"> - Sustentación: donde se elige que clase de acta guarda y estas son: - Acta Inicial. - Cumplió Objetivos. - 1º Sustentación - Sustentación final. <p>- Además el administrador deberá elegir el tutor del proyecto.</p> <p>10. El administrador le dará al botón guardar acta.</p> <p>11. El sistema validó la información y guarda el acta en el sistema.</p> <p>12. El administrador vuelve a pantalla de actas.</p>
<p>Flujo Alternativo</p>	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El usuario debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: campos incompletos. Descripción: Si en el paso 6 del flujo normal el usuario ingresa mal un campo:</p> <ol style="list-style-type: none"> 1. el sistema mostrará un mensaje " campo incompleto" 2. si el usuario no le da clic al botón crear se perderá la información. 3. el usuario debe de volver al paso 6 del flujo normal. <p>Flujo Alt 3: campos incompletos Descripción: Si en el paso 9 del flujo normal el usuario no ingresa la información correcta.</p> <ol style="list-style-type: none"> 1. El usuario debe de ingresar manualmente la información del acta creada y del tutor para el proyecto. 2. En caso de no dale clic al botón guardar, se perderá la información. 3. El usuario debe de volver al paso 6 del flujo normal.

Fuente: Elaboración propia.

Tabla 30: Descripción Caso de Uso Actualizar Acta del Proyecto.

C.U-08	- Actualizar actas para el administrador.
Actores	- Administrador.
Tipo	- Básico.
Propósito	- Le permite al Administrador actualizar cualquier información que necesite de las actas.
Resumen	- El Administrador podrá gestionar cualquier información del sistema para actualizar actas.
Pre- condiciones	1. El administrador debe de estar registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador ingresará los campos de logueo para iniciar sesión y son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El administrador le dará al botón iniciar sesión. 3. El sistema validó la información. 4. El administrador ingresará a la vista actas y elegirá el acta a actualizar. 5. El administrador ingresará la información a actualizar. 6. El administrador dará clic al botón guardar. 7. El sistema validó la información y la guarda en la base de datos. 8. El administrador regresa a la vista Actas.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El usuario debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: campos incompletos. Descripción: Si en el paso 5 del flujo normal el usuario ingresa mal un campo:</p> <ol style="list-style-type: none"> 1. el sistema mostrará un mensaje " campo incompleto" 2. si el usuario no le da clic al botón crear se perderá la información.

Fuente: Elaboración propia.

Tabla 31: Descripción Caso de Uso Buscar Acta del Proyecto.

C.U-09	- Buscar actas para administrador.
Actores	- Administrador.
Tipo	- Básico.
Propósito	- El permite al Administrador buscar el acta que necesite.
Resumen	- El Sistema me debe permitir al administrador buscar las actas en el sistema.
Pre- condiciones	- El administrador deberá estar registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador debe ingresar campos para logue que son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El administrador le dará clic al botón iniciar sesión 3. El sistema validó la información. 4. El administrador ingresa a la vista actas y elije el acta a buscar. 5. El administrador dará clic al botón guardar. 6. El administrador regresa a la vista actas
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El usuario debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Acta no encontrada. Descripción: si en el paso 5 del flujo normal no se encuentra el acta.</p> <ol style="list-style-type: none"> 1. En caso de no encontrar el acta el usuario debe de volver a crearla. 2. El usuario debe de volver al paso de crea acta.

Fuente: Elaboración propia.

Tabla 32: Descripción Caso de Uso Borrar Acta del Proyecto.

C.U-10	- Borrar actas para el administrador.
Actores	- Administrador.
Tipo	- Básico.

Propósito	- El sistema debe permitir al administrador del sistema poder borrar cualquier información que se requiera.
Resumen	- El administrador se encarga de borrar los datos que se requieran en las actas para los proyectos de grado.
Pre-condiciones	- Se requiere que el administrador este registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador deberá ingresar campos para iniciar sesión que son: <ul style="list-style-type: none"> - Email. - Password y contraseña. - 2. El administrador le dará clic en el botón iniciar sesión. 3. El sistema validó la información. 4. El administrador se dirige a la vista de actas. 5. El administrador elige el acta y le da clic a borrar. 6. El sistema validó que los datos borrados sean correctos. 7. El administrador regresa a la vista actas.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El administrador debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Borrar acta incorrecta. Descripción: Si el administrador borra el acta Incorrecta.</p> <ol style="list-style-type: none"> 1. En caso de borrar una acta equivocada deberá volver a crear el acta borrada. 2. El administrador debe de volver al paso de crear acta.

Fuente: Elaboración Propia.

Tabla 33: Descripción Caso de Uso Crear Informe Ejecutivo.

C.U-11	- Crear un Informe Ejecutivo para Administrador.
Actores	- Administrador.
Tipo	- Básico.
Propósito	- EL sistema debe permitir crear un Informe Ejecutivo por el administrador.
Resumen	- El administrador es el encargado de crear el informe Ejecutivo los proyectos.
Pre-condiciones	- El administrador debe de esta registrado en la base de datos.
	1. El administrador debe ingresar campos para iniciar

<p>Flujo Principal</p>	<p>sesión y esos son:</p> <ul style="list-style-type: none"> - Email. - Password o contraseña. <p>2. El administrador se dará clic al botón iniciar sesión. 3. El sistema valida la información. 4. El administrador ingresa a la vista actas y completa los datos para crear el informe ejecutivo en el sistema que son:</p> <ul style="list-style-type: none"> - Autores de proyecto. - Cedula. - Sustentación - Nombre del proyecto - Tutor del proyecto - Título Proyecto. - Objetivo General. - Objetivos Específicos. - Modalidad del proyecto. - Línea de Investigación. - Cronograma de actividades. - Observaciones. <p>5. El administrador le da clic al botón guardar</p> <p>6. El sistema valida la información y la guarda.</p> <p>7. El administrador regresa a la vista actas y crea el informe Ejecutivo en pdf.</p>
<p>Flujo Alternativo</p>	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <p>1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El administrador debe de volver al paso 1 del flujo normal.</p> <p>Flujo Alt 2: campos incompletos. Descripción: Si en el paso 4 del flujo normal el usuario no ingresa un campo.</p> <p>1. En caso de no ingresar algún campo el sistema mostrará mensaje "campo incompleto". 2. El usuario debe de volver al paso 4 del flujo normal. 3. En caso de no darle clic al botón guardar la información se perderá. 4. El usuario debe de volver al paso 4 del flujo normal.</p>

Fuente: Elaboración Propia.

Tabla 34: Descripción Caso de Uso Actualizar Informe Ejecutivo.

C.U-12	- Actualizar informe ejecutivo para el Administrador
Actores	- Administrador.
Tipo	- Básico.
Propósito	- EL sistema debe permitir actualizar un informe ejecutivo hecho por el administrador.
Resumen	- El administrador es el encargado de actualizar el informe ejecutivo de los proyectos.
Pre- condiciones	1. El administrador deberá de estar registrado en la base de datos del sistema.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador ingresa campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El administrador le da clic al botón iniciar sesión. 3. El sistema valida la información. 4. El administrador ingresa a la vista actas. 5. El administrador elije el informe a actualizar. 6. El administrador ingresa campos a actualizar y le da al botón guardar. 7. El sistema valida la información 8. El administrador regresa a la vista actas.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El administrador debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Actualización incompleta. Descripción: Si en el paso 4 del flujo normal el usuario no ingresa la información</p> <ol style="list-style-type: none"> 1. En caso de no ingresar cualquier campo el sistema mostrará campo incompleto. 2. En caso de que el usuario no le dé clic al botón guardar se perderá la información. 3. El usuario deberá volver al paso 4 del flujo normal. <p>Flujo Alt 3: Informe no creado. Descripción: Si en el paso 4 del flujo normal no se encuentra el</p>

	<p>Informe.</p> <ol style="list-style-type: none"> 1. En caso de no encontrar el acta el administrador deberá volver a crearla. 2. Si el informe está incompleto el administrador deberá ingresar y completar la información para dejar el acta actualizada. 3. El administrador deberá volver al paso 4 del flujo normal.
--	---

Fuente: Elaboración Propia.

Tabla 35: Descripción Caso de Uso Buscar Informe Ejecutivo.

C.U-13	- Buscar informe ejecutivo.
Actores	- Administrador.
Tipo	- Básico.
Propósito	- EL sistema debe permitir buscar un informe ejecutivo hecho por el administrador.
Resumen	- El administrador es el encargado de buscar el informe ejecutivo de los proyectos.
Pre-condiciones	1. El administrador deberá estar registrado en la base de datos
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador deberá ingresar campos de iniciar sesión y esos son: <ul style="list-style-type: none"> - Email. - Password. 2. El administrador le dará clic al botón iniciar sesión. 3. El sistema valida la información. 4. El administrador ingresa en la vista de actas. 5. El administrador busca el informe ejecutivo. 6. El administrador regresa a la vista actas.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El administrador debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Informe No encontrado. Descripción: Si en el paso 4 del flujo normal el administrador no encuentra el informe.</p>

	<ol style="list-style-type: none"> 1. En caso de no encontrar el informe el administrador deberá volver a crearlo. 2. El administrador deberá volver al paso de crear informe.
--	--

Fuente: Elaboración Propia.

Tabla 36: Descripción Caso de Uso Borrar Informe Ejecutivo.

C.U-14	- Borrar informe ejecutivo para el Administrador
Actores	- Administrador.
Tipo	- Básico.
Propósito	- EL sistema debe permitir borrar un informe ejecutivo hecho por el administrador.
Resumen	- El administrador es el encargado de borrar el informe ejecutivo de los proyectos.
Pre- condiciones	1. El administrador deberá de estar registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador deberá ingresar campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password. 2. El administrador le dará al botón Iniciar sesión. 3. El sistema valida la información. 4. El administrador ingresa en la vista actas y elige el informe a eliminar. 5. El administrador dará clic al botón borrar. 6. El sistema valida la información y borra informe de la base de datos. 7. El administrador regresa a la vista actas.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El administrador debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Informe No encontrado. Descripción: Si en el paso 4 del flujo normal el administrador</p>

	<p>no encuentra el informe.</p> <ol style="list-style-type: none"> 1. En caso de no encontrar el informe el administrador deberá volver a crearlo. 2. El administrador deberá volver al paso de crear informe.
--	--

Fuente: Elaboración Propia.

Tabla 37: Descripción Caso de Uso Gestionar Permisos del Sistema.

C.U-15	- Gestionar permisos para el administrador.
Actores	- Administrador.
Tipo	- Básico.
Propósito	- El sistema debe permitir gestionar los permisos al administrador del sistema.
Resumen	- El administrador se encarga de gestionar los permisos que se requieran en el programa para su normal funcionamiento.
Pre-condiciones	- Se requiere que el administrador este registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador deberá ingresar campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password. 2. El administrador le da clic al botón iniciar sesión. 3. El sistema valida la información. 4. El administrador entra en la vista permisos administrativos. 5. El administrador elige al usuario al cual se le darán permisos administrativos le da clic al botón roles. 6. El sistema valida la información. 7. El administrador ingresa a la vista roles y le concede permisos administrativos al usuario le da clic al botón guardar. 8. El administrador vuelve a la vista principal.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El administrador debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Email no encontrado. Descripción: Si en el paso 5 del flujo normal el administrador</p>

	<p>no encuentra el email.</p> <ol style="list-style-type: none"> 1. El administrador deberá informar a la persona que no se encuentra en el sistema que se registre. 2. El usuario al estar registrado deberá aparecer en el sistema, En caso es contrario debe de informar al administrador.
--	---

Fuente: Elaboración Propia.

Tabla 38: Descripción Caso de Uso Crear Observaciones del Proyecto.

C.U-16	- Crear observaciones del proyecto para el tutor.
Actores	- Tutor.
Tipo	- Básico.
Propósito	- Permitir al tutor poder crear las observaciones que requiera el proyecto.
Resumen	- El sistema debe permitir que el tutor pueda crear las observaciones que se requieren en el proyecto.
Pre-condiciones	- El tutor deberá estar registrado en el sistema.
Flujo Principal	<ol style="list-style-type: none"> 1. El tutor deberá ingresar campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password. 2. El tutor dará clic en el botón de iniciar sesión. 3. El sistema valida la información. 4. El tutor ingresa a la vista de Observaciones. 5. El tutor ingresa los campos para hacer la observación del proyecto y estos son : <ul style="list-style-type: none"> - Nombre del proyecto. - Descripción de la observación. - Fecha. 6. El tutor le da clic al botón crear observaciones. 7. El sistema valida la información. 8. El tutor regresa a la vista observaciones.
	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos".

Flujo Alternativo	<p>2. El Tutor debe de volver al paso 1 del flujo normal.</p> <p>Flujo Alt 2: Si en el paso 5 del flujo normal el tutor no ingresa la información.</p> <ol style="list-style-type: none"> 1. El sistema mostrará un mensaje “campos incompletos” 2. El tutor deberá darle clic al botón crear, de lo contrario se perderá la información. 3. El tutor deberá volver al paso 5.
--------------------------	---

Fuente: Elaboración Propia.

Tabla 39: Descripción Caso de Uso Actualizar Observaciones del Proyecto.

C.U-17	- Actualizar observaciones del proyectos para tutor
Actores	- Tutor.
Tipo	- Básico.
Propósito	- Permitir al tutor poder actualizar cualquier observación que requiera el proyecto.
Resumen	- El sistema debe permitir que el tutor pueda actualizar las observaciones que se tiene en el proyecto.
Pre-condiciones	- Se requiere que el tutor esté registrado en el sistema.
Flujo Principal	<ol style="list-style-type: none"> 1. El tutor deberá ingresar campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El tutor deberá darle clic al botón iniciar sesión. 3. El sistema valida la información. 4. El tutor ingresa a la vista de observaciones. 5. El tutor elegirá la observación a actualizar 6. El tutor ingresara la observación a actualizar 7. El tutor le dará clic al botón crear observación. 8. El sistema valida la información. 9. El tutor regresa a la vista observaciones.
Flujo Alternativo.	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará “Email o password incorrectos”. 2. El Tutor debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Observación no encontrada. Descripción Si en el paso 5 del flujo normal el tutor no encuentra la observación.</p> <ol style="list-style-type: none"> 1. El tutor deberá a volver a crear la observación. 2. El tutor debe de volver al paso 5 del flujo normal. <p>Flujo Alt 3: Información no guardada. Descripción: En el paso 5 del flujo normal el tutor deberá</p>

	<p>guardar la información.</p> <ol style="list-style-type: none"> 1. El tutor deberá ingresar la información que desea actualizar. 2. En caso de que el tutor no le dé clic al botón guardar se perderá la información. 3. El tutor deberá volver al paso 5.
--	---

Fuente: Elaboración Propia.

Tabla 40: Descripción Caso de Uso Buscar Observaciones del Proyecto.

C.U-18	- Buscar observaciones del proyecto para tutor.
Actores	- Tutor.
Tipo	- Básico.
Propósito	- Permitir al tutor poder buscar las observaciones hechas para el proyecto.
Resumen	- El sistema debe permitir que el tutor pueda buscar las observaciones que se le haga al proyecto para su terminación.
Pre-condiciones	- El tutor deberá estar registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El tutor deberá ingresar campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password. 1. El tutor le dará clic al botón iniciar sesión. 2. El sistema valida la información. 3. El tutor abrirá el proyecto y buscará las observaciones que tenga el proyecto. 4. El tutor ingresará las observaciones hechas al proyecto. 5. El tutor le dará clic al botón crear observación. 6. El sistema valida la información. 8. El tutor regresa a la vista observaciones.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El tutor debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Observación no encontrada. Descripción Si en el paso 5 del flujo normal el tutor no encuentra la observación.</p> <ol style="list-style-type: none"> 1. El tutor deberá a volver a crear la observación. 2. El tutor debe de volver al paso 5 del flujo normal.

Fuente: Elaboración Propia.

Tabla 41. Descripción Caso de Uso Borrar Observaciones del Proyecto.

C.U-19	- Borrar observaciones del proyecto para tutor.
Actores	- Tutor.
Tipo	- Básico.
Propósito	- Permitir al tutor poder borrar las observaciones hechas para el proyecto.
Resumen	- El sistema debe permitir que el tutor pueda borrar las observaciones que se le haga al proyecto.
Pre-condiciones	- Se requiere que el tutor esté registrado en el sistema.
Flujo Principal	<ol style="list-style-type: none"> 1. El tutor deberá ingresar campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El tutor deberá darle clic al botón iniciar sesión. 3. El sistema valida la información. 4. El tutor ingresará en la vista observaciones. 5. El tutor elegirá el proyecto al cual se le borrarán las observaciones y le clic a borrar. 6. El sistema valida la información y se borrarán las observaciones de la base de datos. 7. El tutor regresa a la vista Observaciones
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El Tutor debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Observación no encontrada. Descripción Si en el paso 5 del flujo normal el tutor no encuentra la observación.</p> <ol style="list-style-type: none"> 1. El tutor deberá a volver a crear la observación. 2. El tutor debe de volver al paso 5 del flujo normal.

Fuente: Elaboración Propia.

Tabla 42. Descripción Caso de Uso Crear Nota para Proyectos.

C.U-20	- Crear Nota del proyecto para tutor.
Actores	- Tutor.
Tipo	- Básico.
Propósito	- Permitir al tutor poder calificar los proyectos por medio de una nota para el proyecto.
Resumen	- El sistema debe permitir que el tutor pueda ingresar las notas para calificar los proyectos de grado.

Pre- condiciones	<ul style="list-style-type: none"> - Se requiere que el tutor esté registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El tutor deberá ingresar campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El tutor deberá darle clic al botón iniciar sesión. 3. El sistema valida la información 4. El tutor ingresará a la vista nota del proyecto e ingresará los campos: <ul style="list-style-type: none"> - Nombre del proyecto. - Nombre del tutor. - Nota del proyecto. 5. El tutor dará clic en el botón crear nota. 6. El sistema valida la información. 7. El tutor regresa a la vista nota del proyecto.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El Tutor debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Campos incompletos. Descripción: En el paso 4 del flujo normal en caso de que le tutor ingrese campo erróneo.</p> <ol style="list-style-type: none"> 1. El sistema muestra mensaje "campos incompletos" 2. En caso de que el tutor no le dé clic al botón guardar se perderá la información. 3. El tutor deberá volver al paso 4 del flujo normal. <p>Flujo Alt 3: Nota no encontrada. Descripción Si en el paso 5 del flujo normal el tutor no encuentra la nota.</p> <ol style="list-style-type: none"> 1. El tutor deberá a volver a crear la nota del proyecto. 2. El tutor debe de volver al paso 5 del flujo normal.

Fuente: Elaboración Propia.

Tabla 43: Descripción Caso de Uso Crear Cronograma del Proyecto.

C.U-24	- Crear cronograma del proyecto para estudiante.
Actores	- Estudiante.
Tipo	- Básico
Propósito	- Permitir al estudiante poder crear el cronograma del proyecto.
Resumen	- El sistema debe permitir que el estudiante pueda crear el cronograma que tendrá el proyecto para tener una guía para llevarlo a su terminación.
Pre-condiciones	- Se requiere que el estudiante este registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El estudiante deberá ingresar los campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El estudiante deberá darle clic al botón iniciar sesión. 3. El sistema valida la información. 4. El estudiante ingresa a la vista cronogramas. 5. El estudiante ingresa campos para hacer el cronograma al proyecto y estos campos son: <ul style="list-style-type: none"> - Proyecto. - Actividad. - Fecha Inicial. - Fecha final. 6. El estudiante dará clic al botón crear cronograma. 7. El sistema valida la información. 8. El cronograma ya se creó para el proyecto seleccionado. 9. El estudiante regresa a la vista cronogramas.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos". 2. El estudiante debe de volver al paso 1 del flujo normal.

	<p>Flujo Alt 2: Camps Incompletos. Descripción: Si en el paso 6 del flujo normal el estudiante no ingresa un campo.</p> <ol style="list-style-type: none"> 1. El sistema mostrará mensaje “campo incompleto” 2. El estudiante deberá volver al paso 5 del flujo normal.
--	--

Fuente: Elaboración Propia.

Tabla 44: Descripción Caso de Uso Actualizar Cronograma del Proyecto.

C.U-25	- Actualizar cronograma del proyecto para estudiante.
Actores	- Estudiantes.
Tipo	- Básico
Propósito	- Permitir al estudiante poder Actualizar el cronograma del proyecto.
Resumen	- El sistema debe permitir que el estudiante pueda Actualizar el cronograma que tendrá el proyecto para tenerlo al orden del día y llevarlo a su terminación.
Pre-condiciones	- Se requiere que el estudiante este registrado en la base de datos.
Flujo Principal	<ol style="list-style-type: none"> 1. El estudiante ingresará campos para iniciar sesión y estos son: <ul style="list-style-type: none"> - Email. - Password o contraseña. 2. El estudiante deberá darle clic al botón iniciar sesión. 3. El sistema valida la información. 4. El estudiante ingresa a la vista cronogramas. 5. El estudiante elige el cronograma a actualizar. 6. El estudiante ingresa la información del cronograma a actualizar y le da al botón actualizar. 7. El sistema valida la información y la guarda en la base de datos. 8. El estudiante regresa a la vista cronogramas.
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <ol style="list-style-type: none"> 1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará “Email o password incorrectos”. 2. El estudiante debe de volver al paso 1 del flujo normal. <p>Flujo Alt 2: Camps Incompletos. Descripción: Si en el paso 6 del flujo normal el estudiante no ingresa un campo.</p> <ol style="list-style-type: none"> 1. El sistema mostrará mensaje “campo incompleto”

	<p>2. El estudiante deberá volver al paso 5 del flujo normal. Flujo Alt 3: Cronograma no encontrado. Descripción: En el paso 4 del flujo normal el estudiante deberá encontrar el cronograma.</p> <p>1. En caso de no encontrar el cronograma el estudiante deberá crearla de nuevo. 2. El estudiante deberá volver al paso 5 del flujo normal. Flujo Alt 4: Fecha Invalida. Descripción: Si en el paso 6 el estudiante no ingresa la fecha correcta.</p> <p>1. El sistema no dejara crear el cronograma si la fecha esta errónea. 2. El estudiante deberá volver al paso 4 del flujo normal.</p>
--	---

Fuente: Elaboración Propia.

Tabla 45: Descripción Caso de Uso Buscar Cronograma del Proyecto.

C.U-26	- Buscar cronograma del proyecto para estudiante.
Actores	- estudiante.
Tipo	- Básico
Propósito	- Permitir al tutor poder buscar el cronograma del proyecto.
Resumen	- El sistema debe permitir que el estudiante pueda buscar el cronograma que tendrá el proyecto para tenerlo al orden del día y llevarlo a su terminación.
Pre-condiciones	- Se requiere que el estudiante este registrado en la base de datos.
Flujo Principal	<p>1. El estudiante deberá ingresar los campos para iniciar sesión y estos son:</p> <ul style="list-style-type: none"> - Email. - Password o contraseña. <p>2. El estudiante deberá darle clic al botón iniciar sesión.</p> <p>3. El sistema valida la información.</p> <p>4. El estudiante buscará el cronograma del proyecto seleccionado y gestiona información si así lo requiere.</p> <p>5. El estudiante regresara a la vista cronogramas.</p>
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <p>1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos".</p>

	<p>2. El estudiante debe de volver al paso 1 del flujo normal. Flujo Alt 2: Cronograma no encontrado. Descripción: En el paso 4 del flujo normal el estudiante deberá encontrar el cronograma.</p> <p>1. En caso de no encontrar el cronograma el estudiante deberá crearla de nuevo. 2. El estudiante deberá volver al paso 4 del flujo normal.</p>
--	--

Fuente: Elaboración Propia.

Tabla 46: Descripción Caso de Uso Borrar Cronograma del Proyecto.

C.U-27	- Borrar cronograma del proyecto para estudiante.
Actores	- estudiante.
Tipo	- Básico
Propósito	- Permitir al estudiante poder borrar el cronograma del proyecto si así lo requiere.
Resumen	- El sistema debe permitir que el estudiante pueda borrar el cronograma en caso de borrar alguna información que sea relevante para el proyecto.
Pre-condiciones	- Se requiere que el estudiante deberá estar registrado en la base de datos.
Flujo Principal	<p>1. El estudiante deberá ingresar campos de iniciar sesión y estos son:</p> <ul style="list-style-type: none"> - Email. - Password. <p>2. El estudiante deberá darle clic al botón iniciar sesión.</p> <p>3. El sistema valida la información.</p> <p>4. El estudiante ingresa a la vista cronograma.</p> <p>5. El estudiante elije el cronograma que da clic en borrar.</p> <p>6. El sistema valida la información y borra el cronograma de la base de datos.</p> <p>7. El estudiante regresa a la vista cronograma</p>
Flujo Alternativo	<p>Flujo Alt 1: Campos incompletos de logueo. Descripción: Datos incompletos de los campos de logueo.</p> <p>1. En caso de que el usuario ingrese mal el email y contraseña, el sistema mostrará "Email o password incorrectos".</p> <p>2. El estudiante debe de volver al paso 1 del flujo normal.</p> <p>Flujo Alt 2: Borrar cronograma equivocado.</p>

	<p>Descripción: En caso de que el estudiante borre el cronograma equivocado.</p> <ol style="list-style-type: none"> 1. El estudiante deberá volver a crear el cronograma que borro por equivocación. 2. El estudiante debe de volver al paso crear cronograma.
--	--

Fuente: Elaboración Propia.

4.6. Diseño

En esta sección se presenta el modelo relacional de la base de datos, el diagrama de secuencias, la Base se SQL Server Management, la arquitectura general de proyecto, el diagrama de clases y finalmente el diseño de las interfaces graficas del sistema.

4.6.1. Modelo Relacional

El modelo relacional (ver Figura 22) es la representación lógica de los datos, luego éste puede ser implementado en un motor de base de datos, teniendo en cuenta las entidades y las relaciones entre ellas.

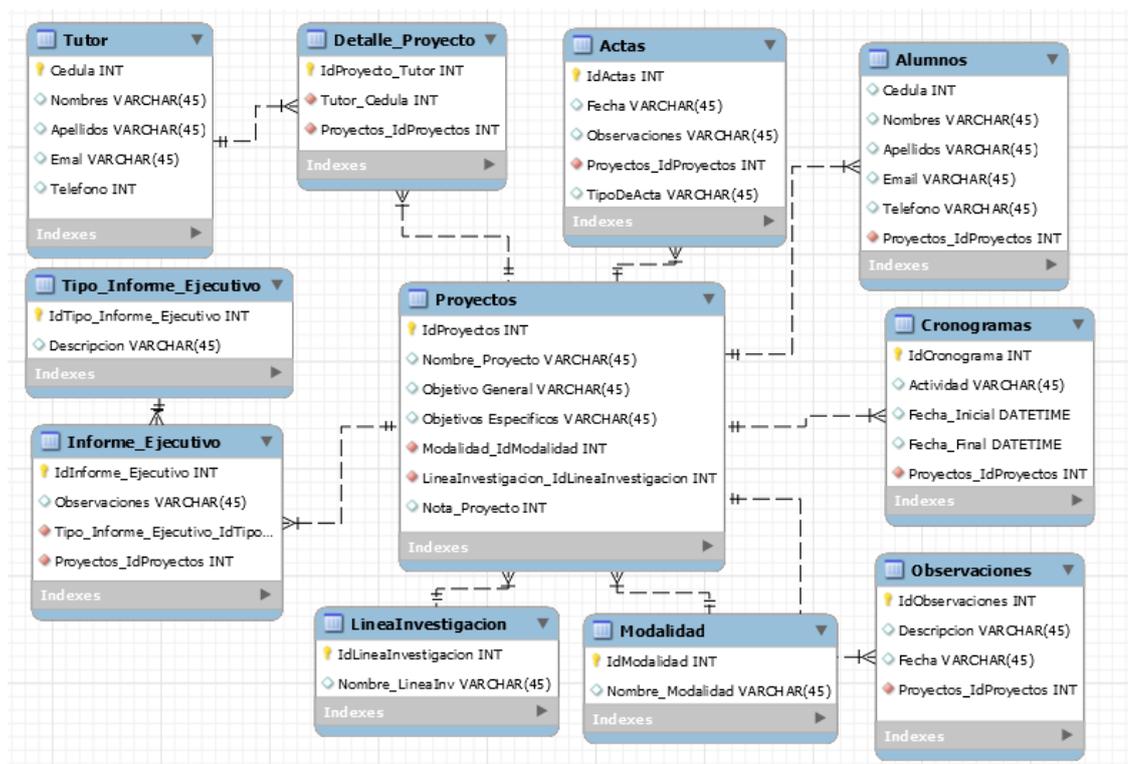


Figura 22: Modelo Relacional.
Fuente: Elaboración Propia.

4.6.2. Base de Datos de SQL Server Management.

En la Figura 23, se muestra cómo está diseñada la base de datos en el programa SQL Server Management 2014.

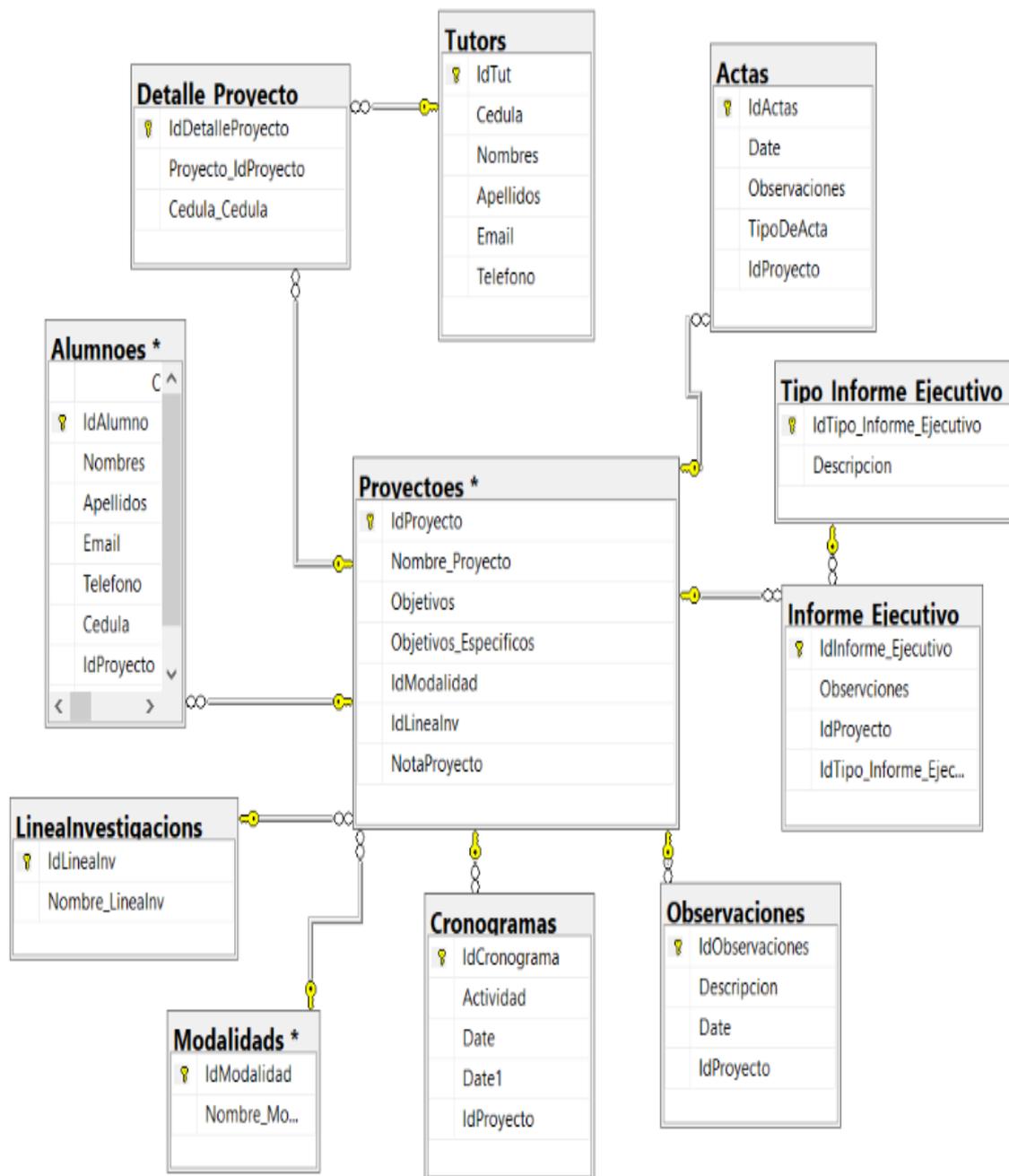


Figura 23: Base de Datos SQL Server.
Fuente: Elaboración propia.

4.6.3. Diagrama de Secuencias

En la Figura 24, se muestra la secuencia de cómo se registra la información de los usuarios en el sistema de gestión de información.

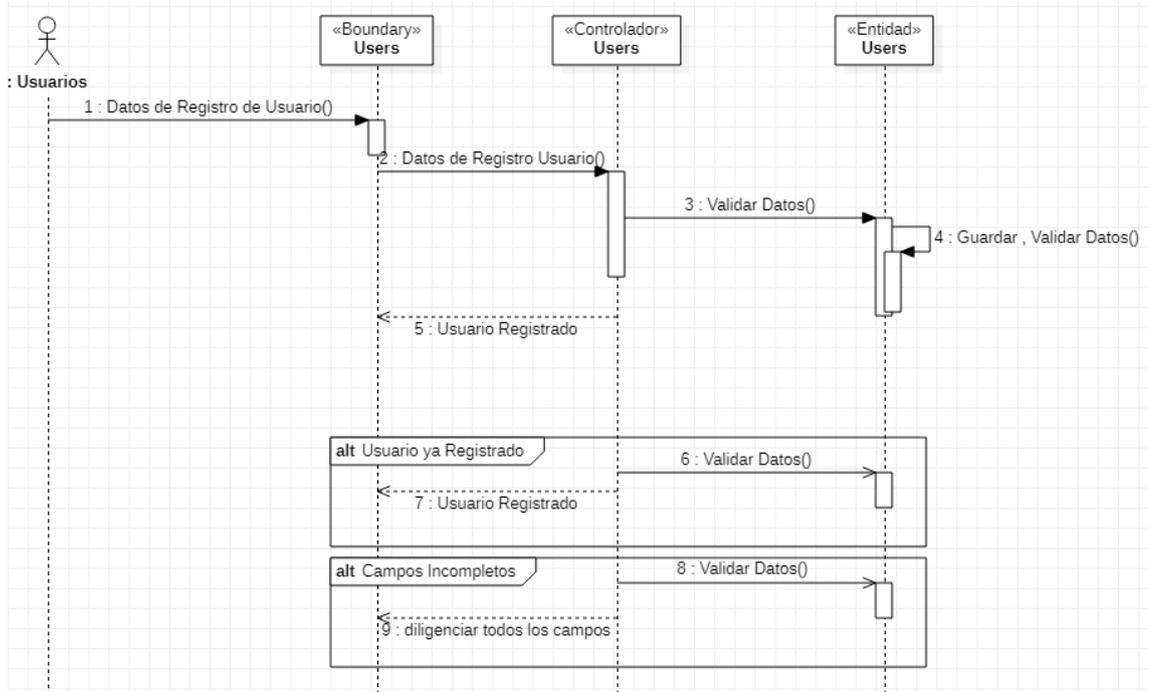


Figura 24: Diagrama de Secuencia Registro de Usuarios.
Fuente: Elaboración Propia.

En la Figura 25, se evidencia la secuencia de logueo que debe de tener el sistema de gestión de información.

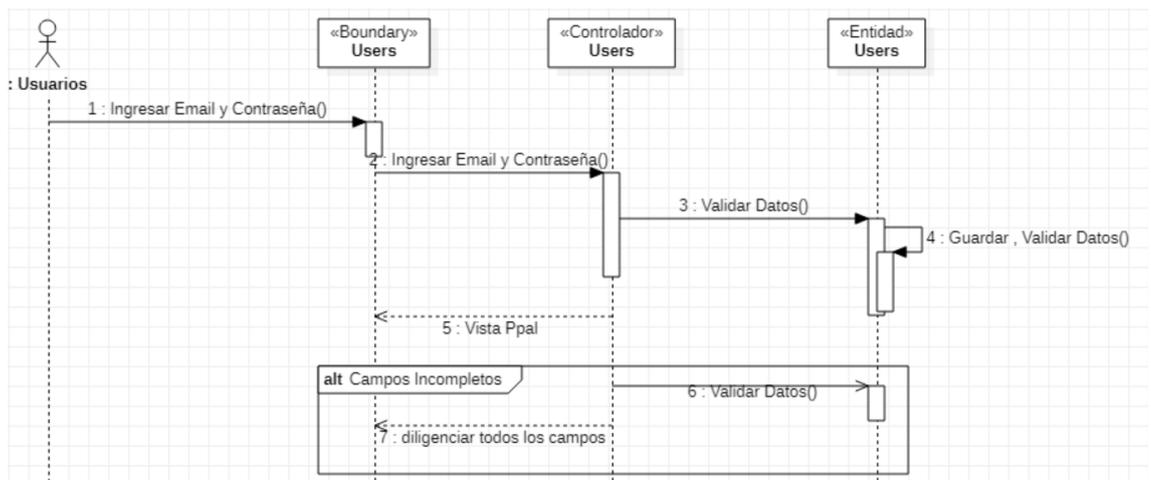


Figura 25: Diagrama de Secuencia Logueo Usuarios.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 26, se muestra la manera de crear un proyecto en el sistema de gestión, además validando la información con un flujo alternativo en caso de realizar una validación de si el proyecto esta creado, y un flujo alternativo en caso de que no se ingrese algún campo.

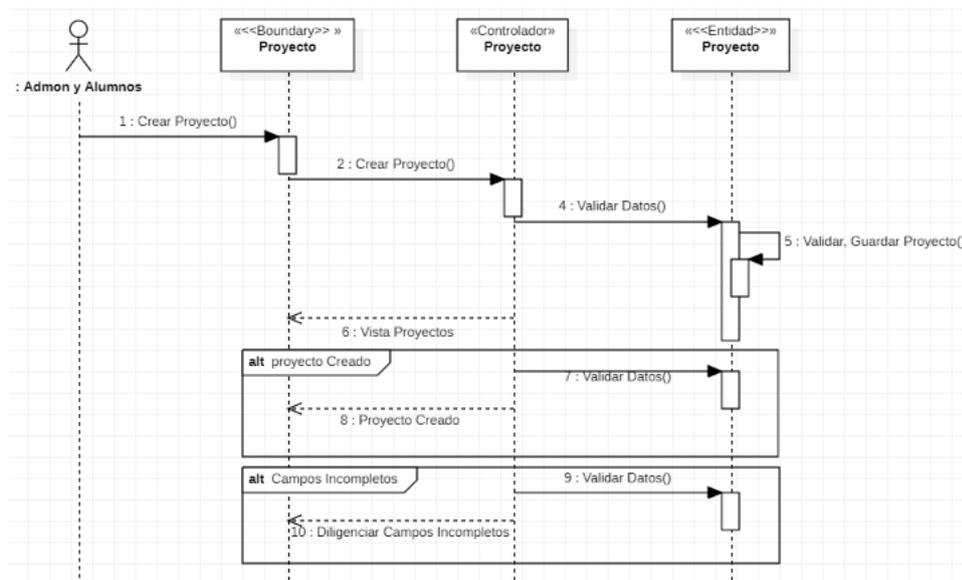


Figura 26: Diagrama de secuencia Crear Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 27, se muestra el proceso para actualizar un proyecto de grado, además se valida la información con un flujo alternativo en caso de no ingresar algún campo.

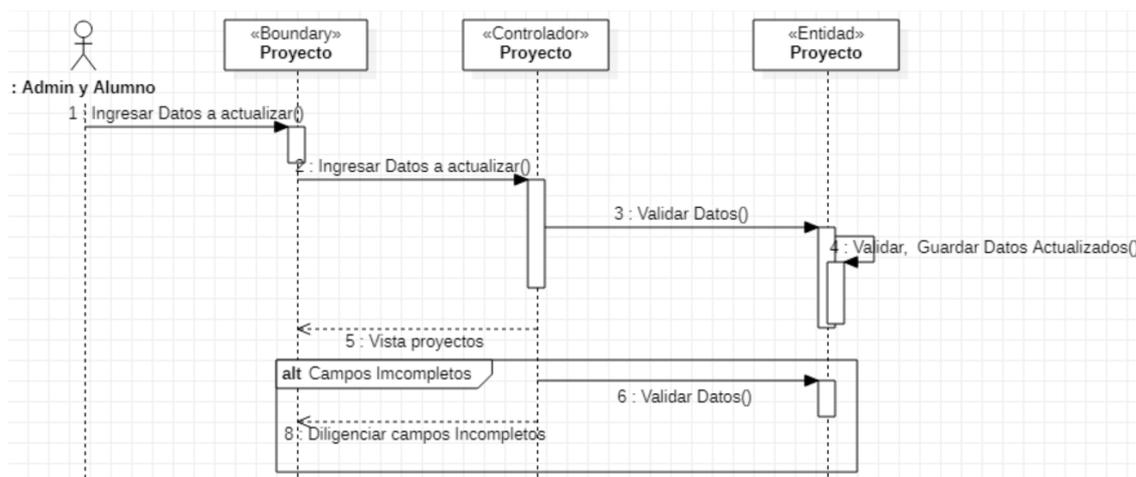


Figura 27: Diagrama de Secuencia Actualizar Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 28, se muestra la manera de guardar un proyecto o la información en el sistema de gestión, además se valida la información con un flujo alternativo en caso de ingresar mal la información.

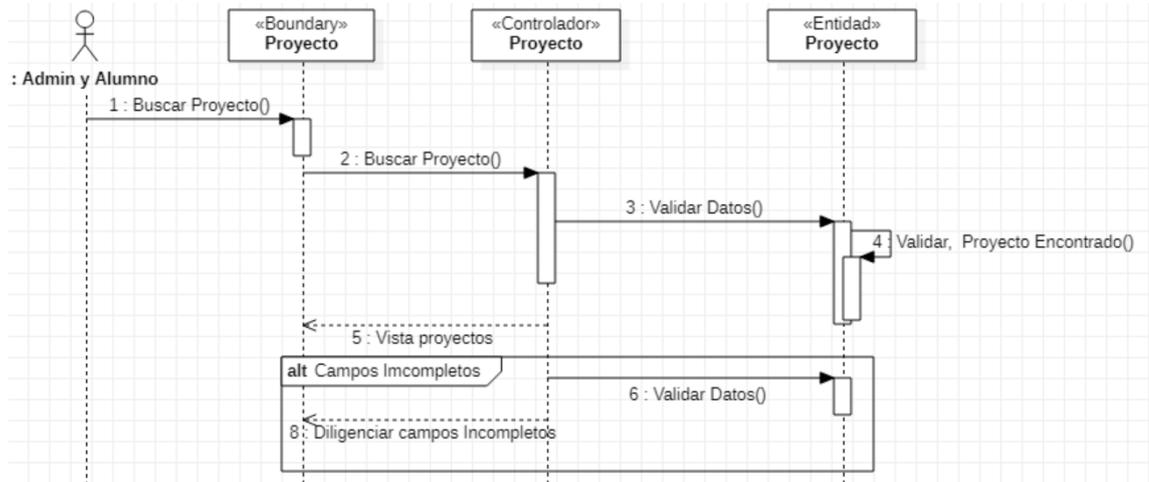


Figura 28: Diagrama de Secuencia Buscar Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 29, se muestra como se elimina un proyecto del sistema de gestión.

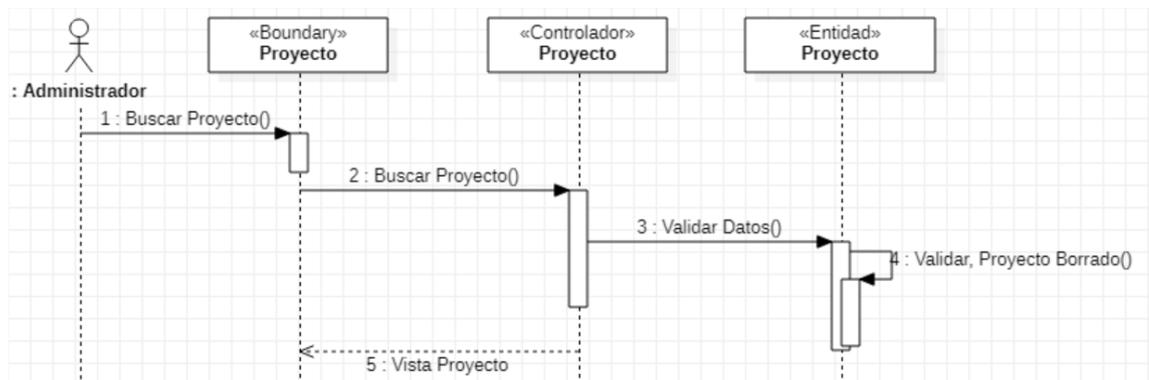


Figura 29: Diagrama de Secuencia Borrar Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 30, se muestra como se da la creación de un acta de proyecto, validando la información con un flujo alternativo, es caso de que la acta inicial ya este creada, se valida la información con un flujo alternativo, en caso de que no ingrese algún campo en el sistema de gestión.

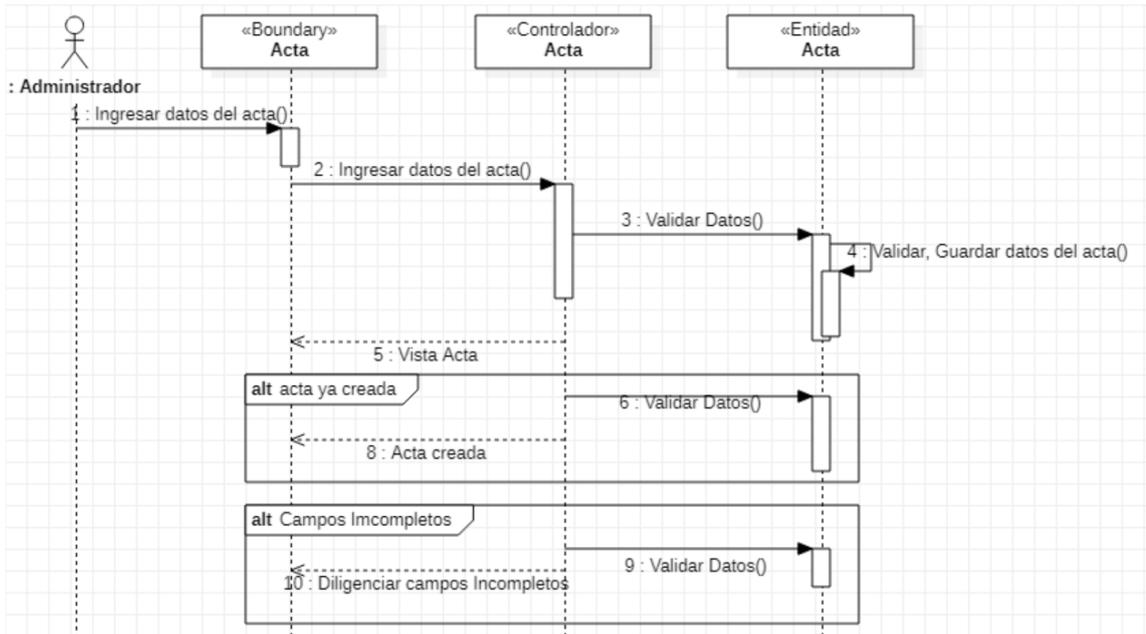


Figura 30: Diagrama de Secuencia Crear Acta del Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 31, se muestra el proceso para actualizar la información de una acta de proyecto, además se valida la información con un flujo alternativo para verificar que se ingresen todos los campos en el sistema de gestión.

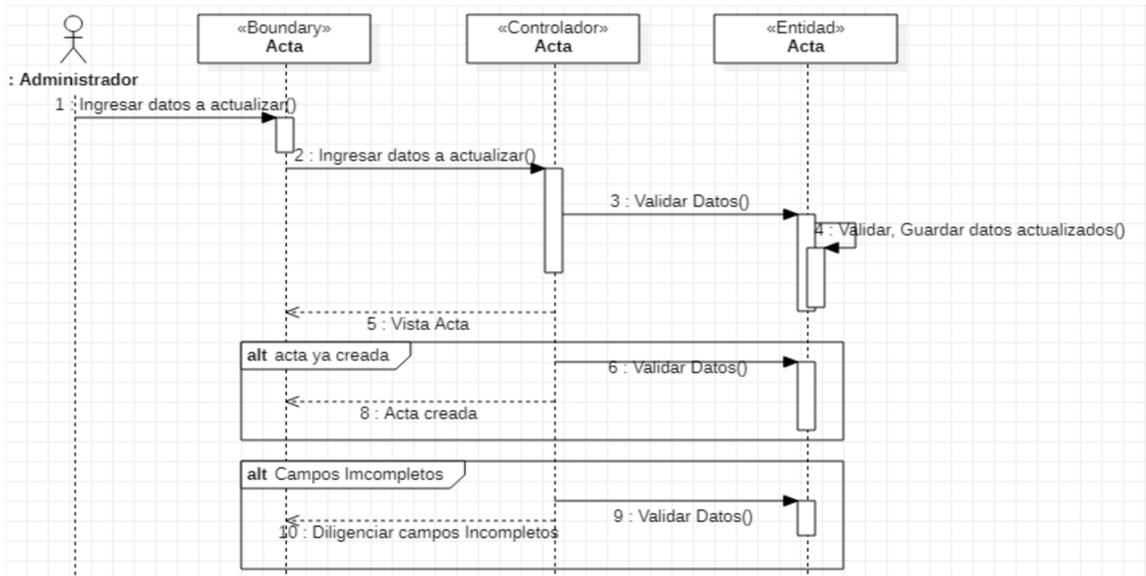


Figura 31: Diagrama de Secuencia Actualizar acta.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 32, se muestra como se busca un acta del proyecto, además se valida la información con un flujo alternativo para verificar que ingrese la información en el sistema de gestión.

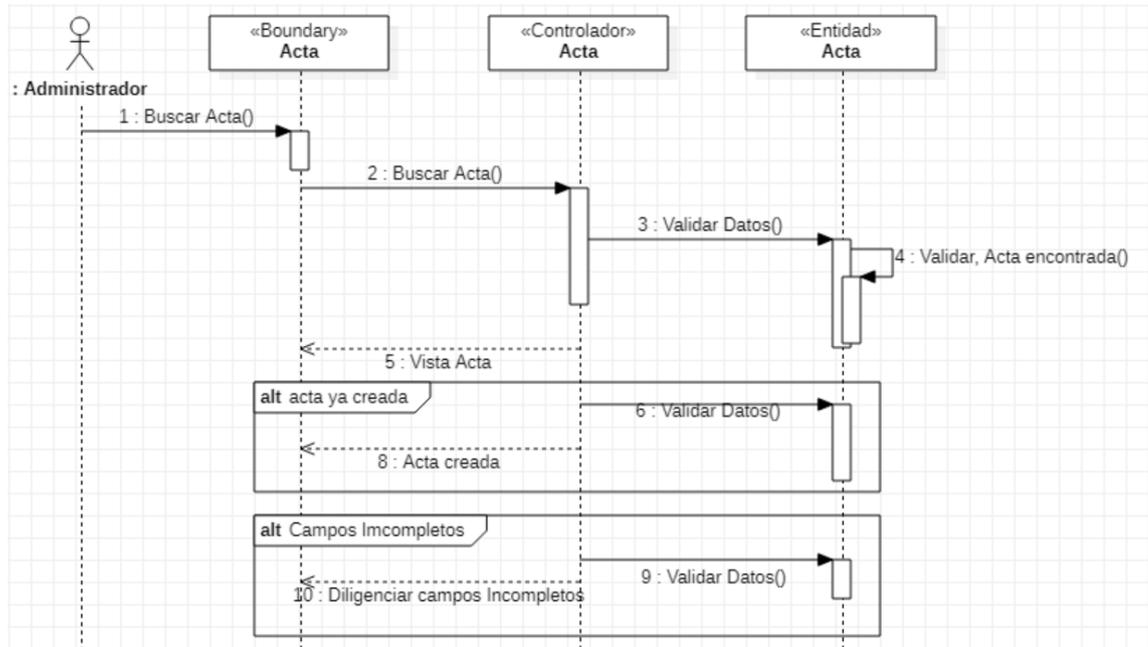


Figura 32: Diagrama de Secuencia Buscar Acta.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 33, se presenta como se borra un acta del sistema de gestión, además se evalúan los datos con un flujo alternativo para saber si el acta fue borrada de forma permanente en el sistema de gestión.

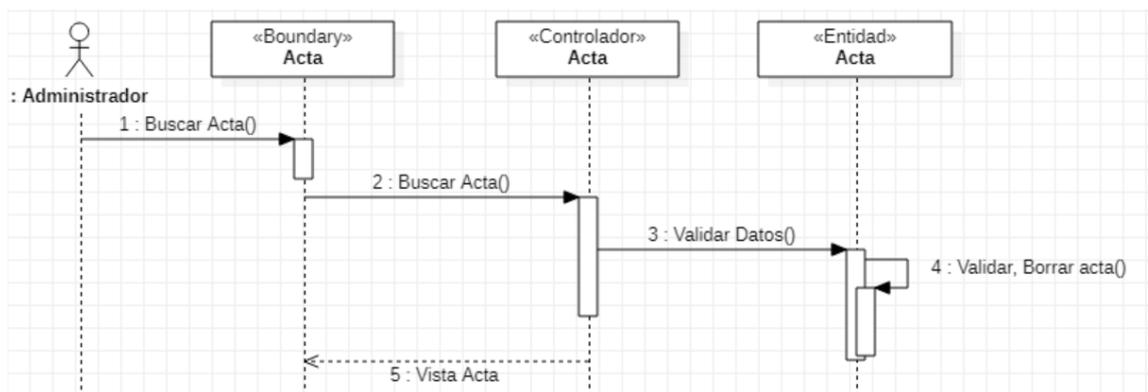


Figura 33: Diagrama de Secuencia Borrar Acta.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 34, se presenta la creación de un informe ejecutivo del cual se encarga el administrador, además se evalúa la información con un flujo alternativo en caso de que el acta este creada, y se evalúa la información con un flujo alternativo en caso de que no se ingresen todos los campo en el sistema de gestión.

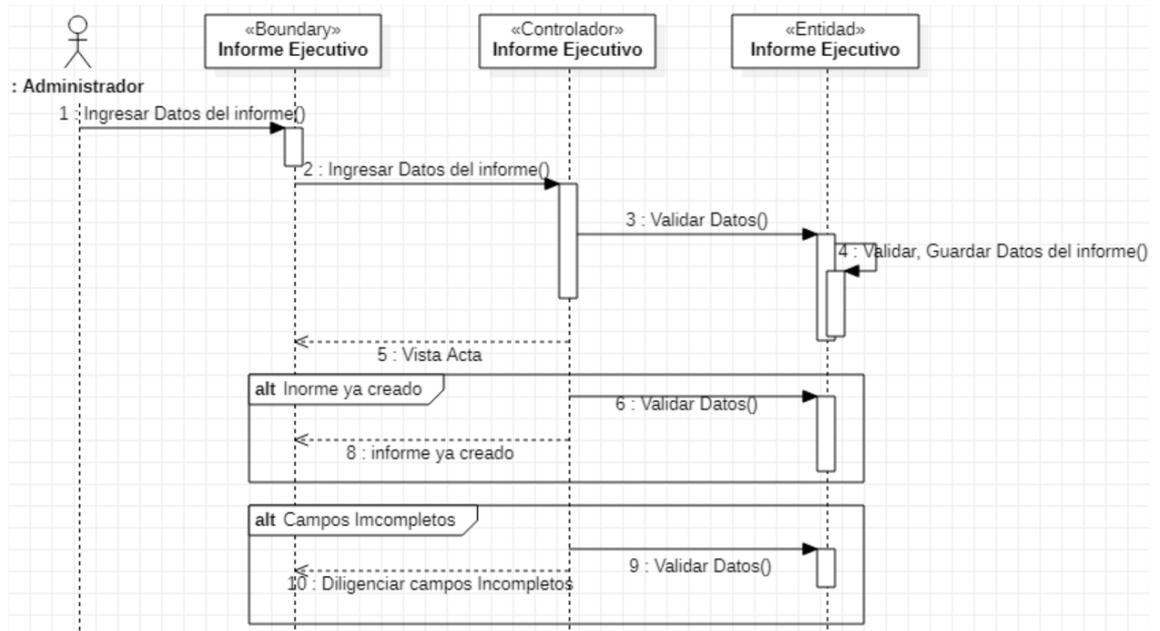


Figura 34: Diagrama de Secuencia Crear Informe Ejecutivo.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 35, se muestra la forma de actualizar un informe ejecutivo hecho por el administrador, además se evalúa la información con un flujo alternativo en caso que no se ingresen todos los campos a actualizar en el sistema de gestión.

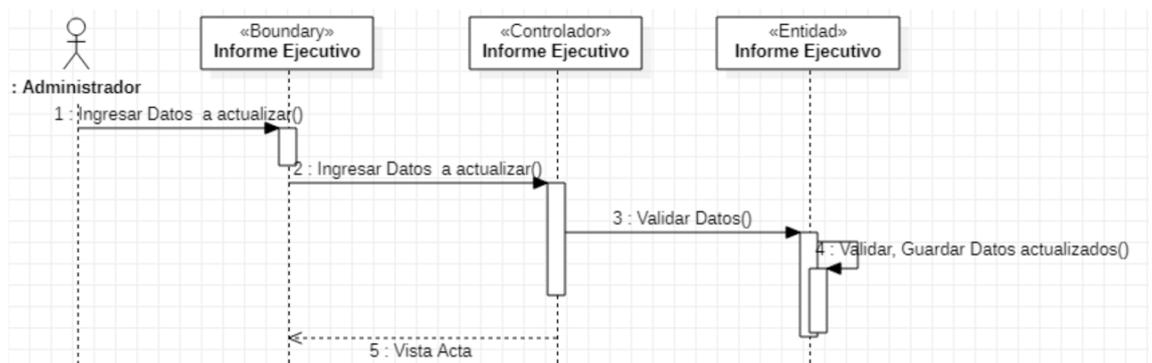


Figura 35: Diagrama de Secuencia Actualizar Informe Ejecutivo.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 36, se presenta como se guarda la información de los informes ejecutivos creados por el administrador, además se evalúa la información con un flujo alternativo guardando la información en el sistema de gestión.

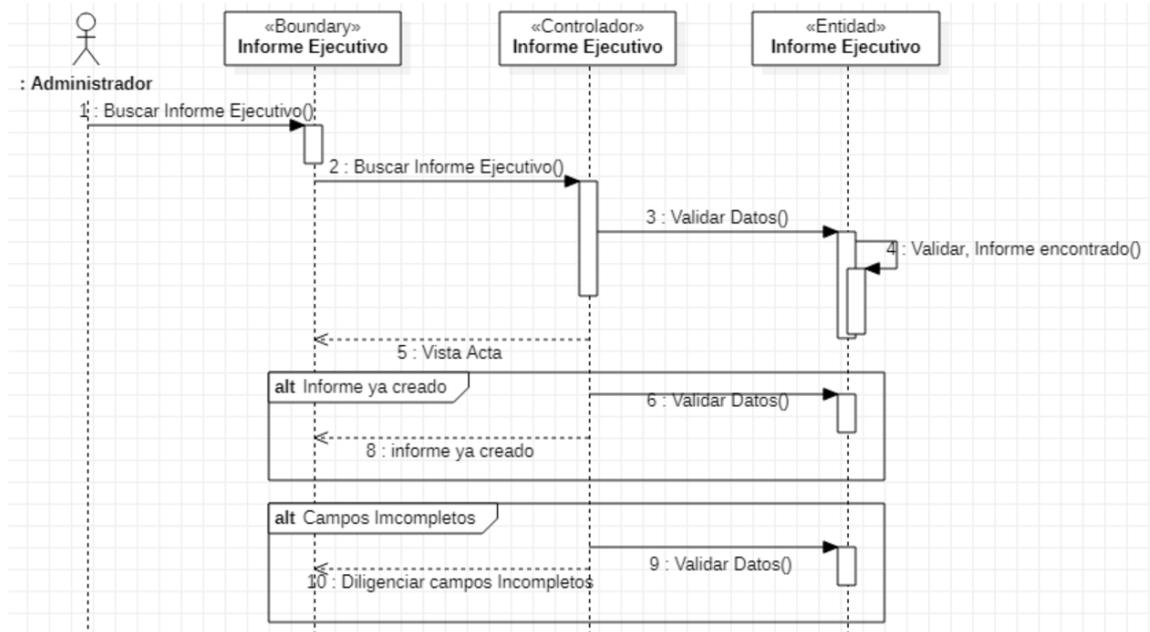


Figura 36: Diagrama de Secuencia Buscar Informe Ejecutivo.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 37, se muestra cómo se borra un informe ejecutivo del sistema esto solo podrá hacerlo el administrador, además se evalúa la información con un flujo alternativo en caso de que no se ingresen todos los campos

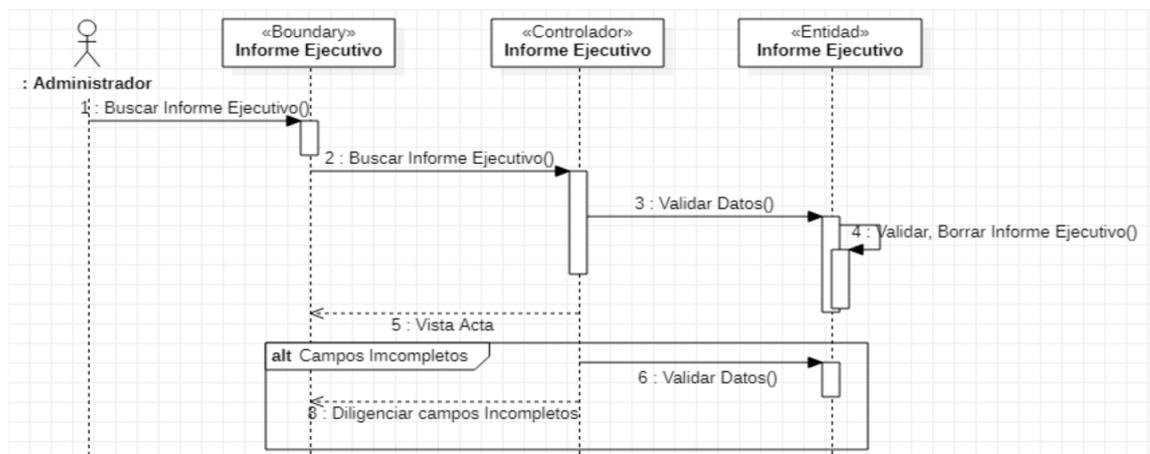


Figura 37: Diagrama de Secuencia Borrar Informe Ejecutivo.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 38, se muestra la forma en la que el administrador concede permisos a los usuarios en el sistema de gestión.

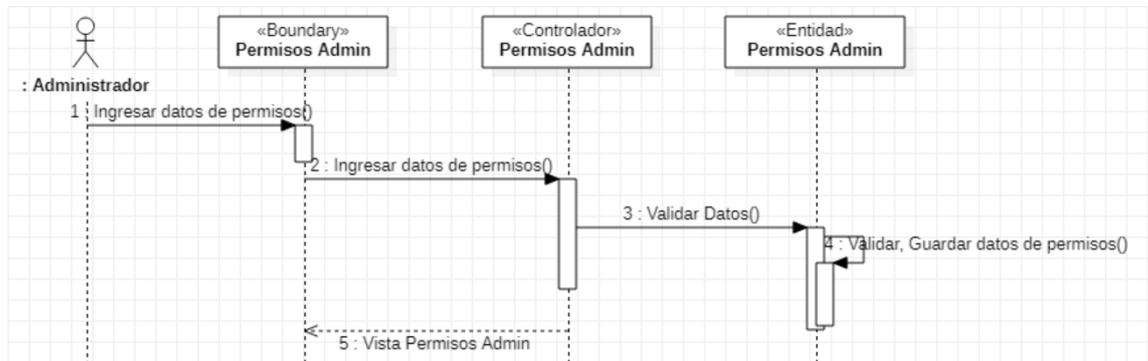


Figura 38: Diagrama de Secuencia Gestionar Permisos del Sistema.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 39, se muestra la forma de crear una observación del proyecto hecha por el tutor, además se evalúa la información con un flujo alternativo para corroborar que la observación no está creada, además se evalúa la información para indicar que no se ingresaron los campos en el sistema de gestión.

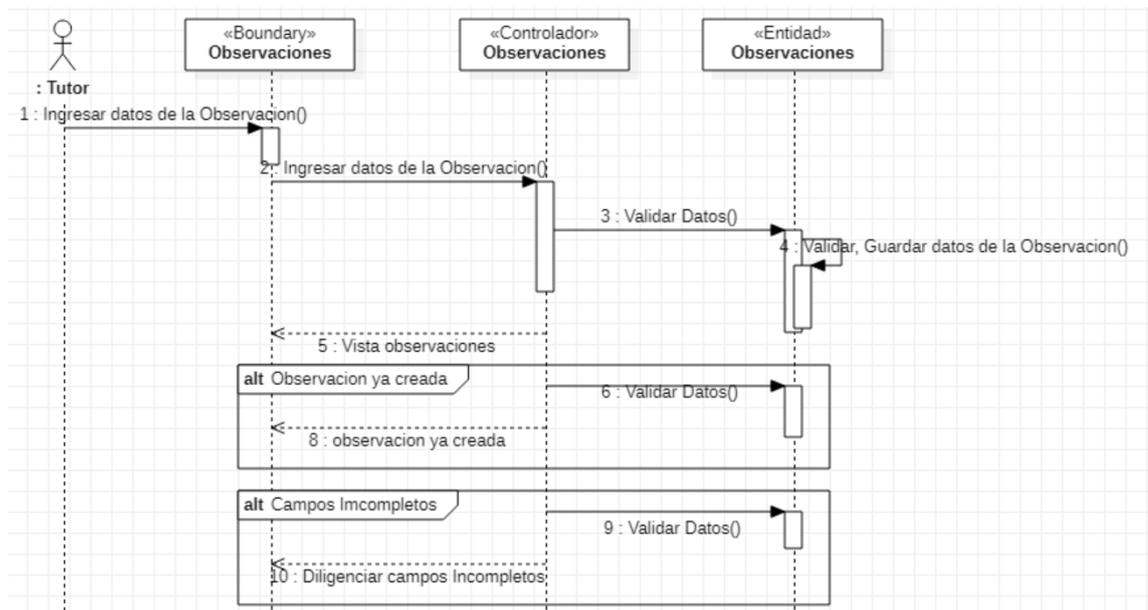


Figura 39: Diagrama de Secuencia Crear Observaciones del Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 40, se muestra la forma de actualizar las observaciones hechas por el tutor del proyecto, además se

evalúa la información con un flujo alternativo para saber si le falta ingresar algún campo en el sistema de gestión.

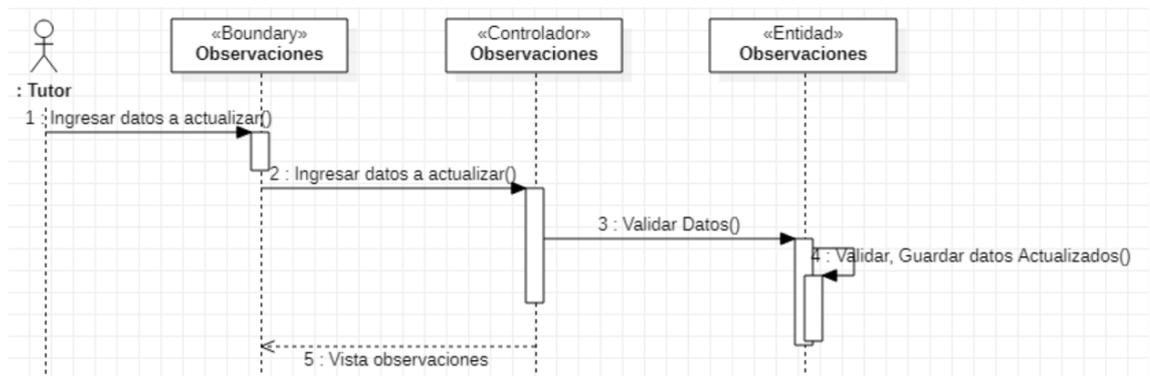


Figura 40: Diagrama de Secuencia Actualizar Observaciones del Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 41, se muestra la forma de guardar la información en el sistema, además se evalúa la información con un flujo alternativo para saber si se ingresaron todos los datos requeridos para la guardar la observación en el sistema de gestión.

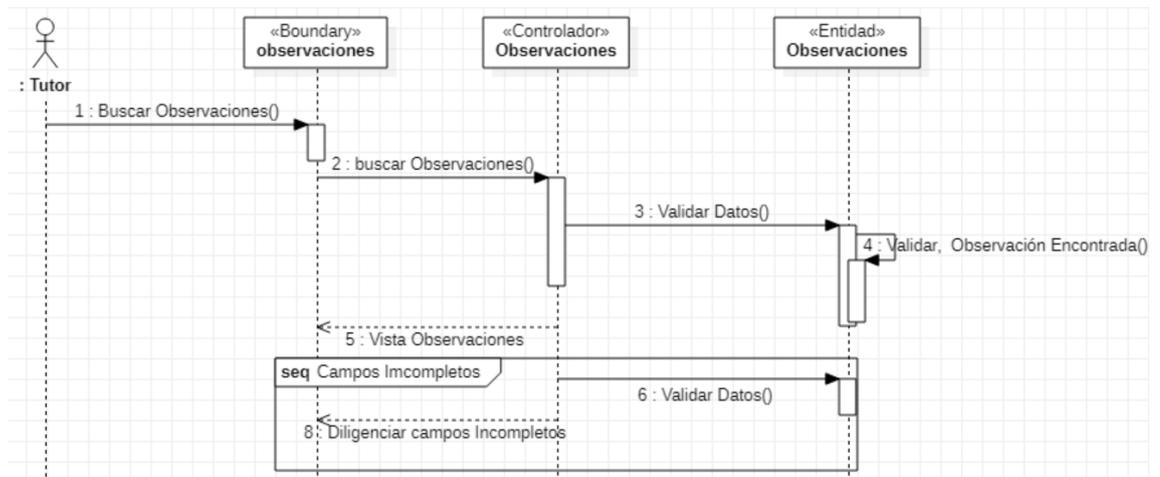


Figura 41: Diagrama de Secuencia Buscar Observaciones.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 42, se muestra la forma de borrar una observación del proyecto en el sistema de gestión.

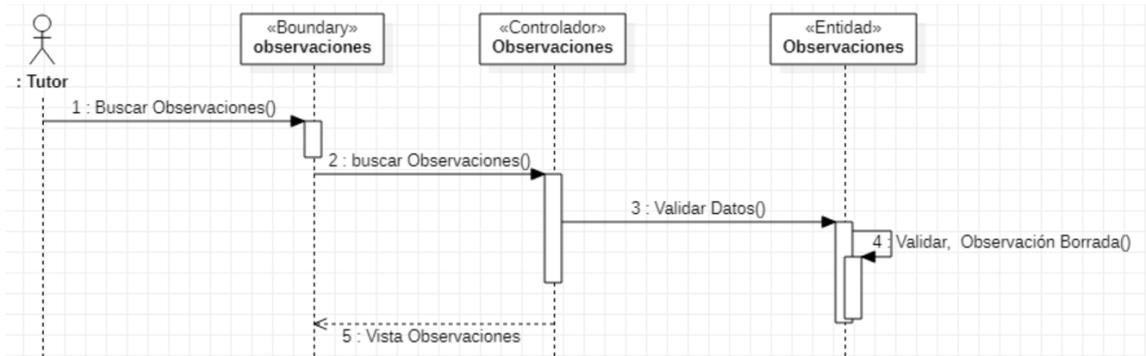


Figura 42: Diagrama de Secuencia Borrar Observaciones del Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 43, se muestra la forma de como el tutor califica el proyecto de grado, además se evalúan los datos por medio de un flujo alternativo para verificar que la nota no esté creada, y un flujo alternativo para verificar que se ingresó la nota del proyecto en el sistema de gestión.

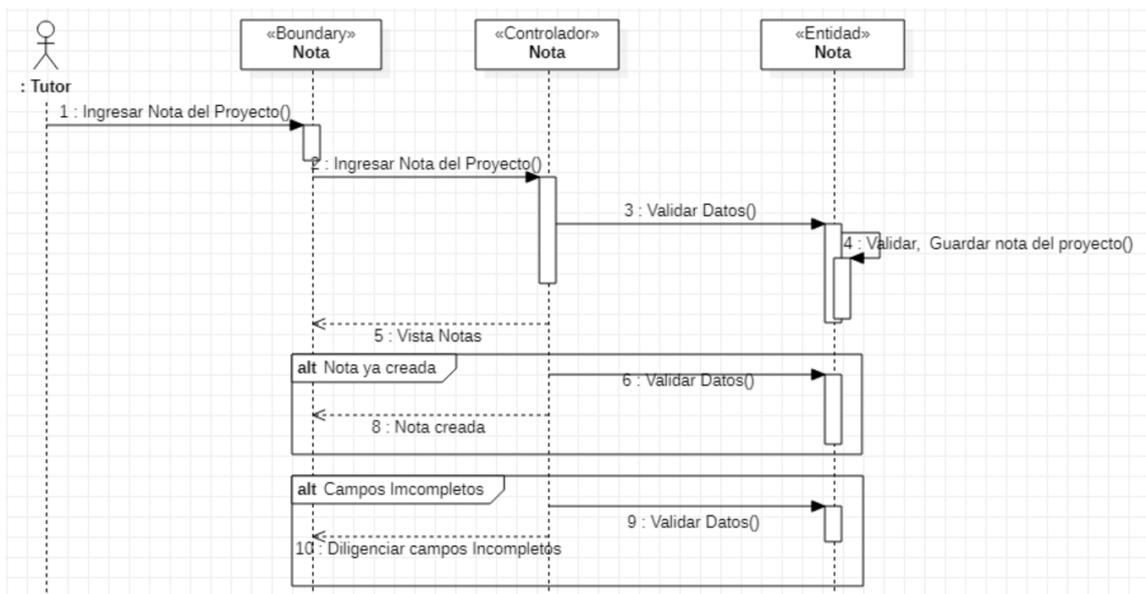


Figura 43: Diagrama de Secuencia Nota del Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 44, se muestra la forma de crear un cronograma del proyecto de grado del alumno, se crea un flujo alternativo para saber si ya existe el cronograma del proyecto, y se crea flujo alternativo para saber si el alumno ingreso todos los campos del cronograma en el sistema de gestión.

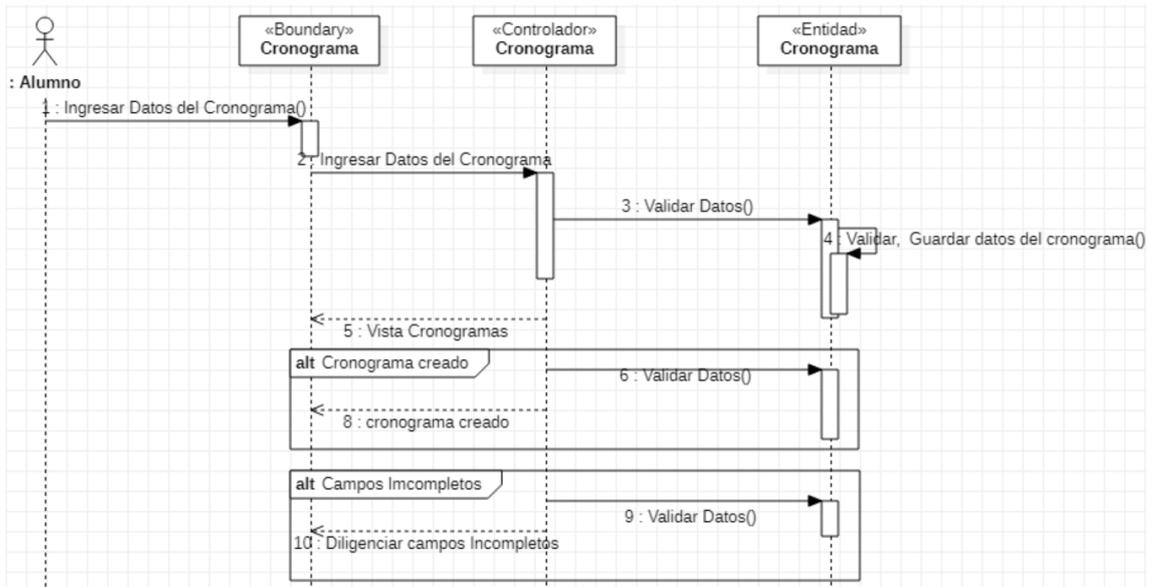


Figura 44: Diagrama de Secuencia Crear Cronograma del Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 45, se muestra la forma de actualizar la información en el sistema, se crea un flujo alternativo en caso de que no se ingresen todos los campos en el sistema de gestión.

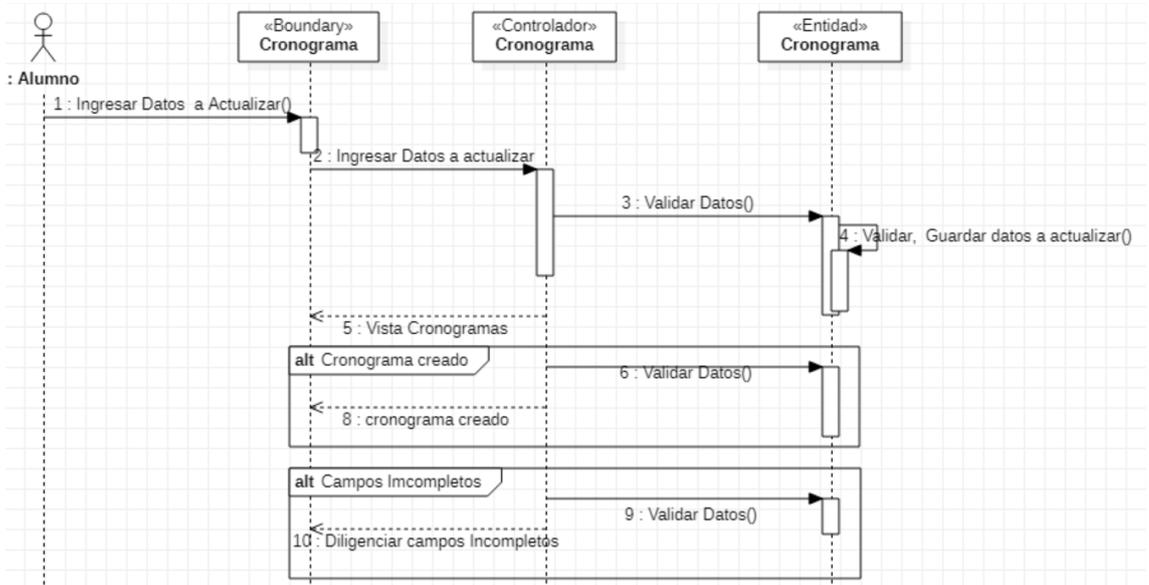


Figura 45: Diagrama de Secuencia Actualizar Cronograma del Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 46, se muestra la forma de guardar la información de los cronogramas de los proyectos de grado, se crea

un flujo alternativo para saber que el cronograma este creado, como también de que todos los campos estén ingresados en el sistema de gestión.

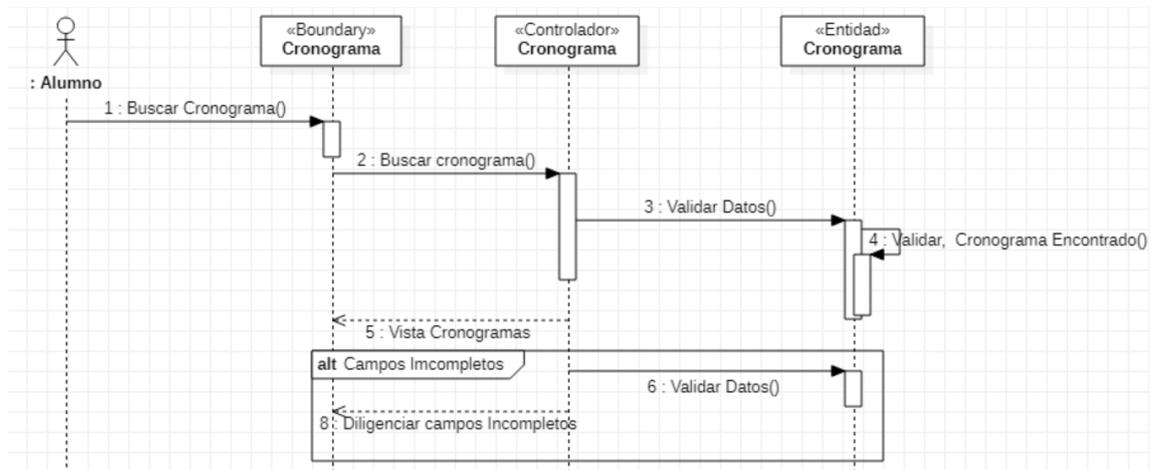


Figura 46: Diagrama de Secuencia Buscar Cronograma del Proyecto.
Fuente: Elaboración Propia.

En el diagrama de secuencia mostrado en la Figura 47, se muestra la forma de como el tutor borra un cronograma del proyecto.

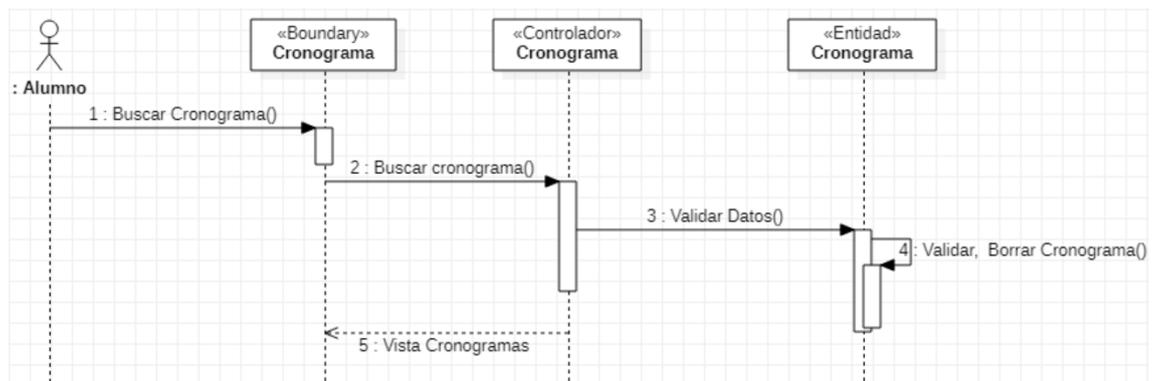


Figura 47: Diagrama de Secuencia Borrar Cronograma del Proyecto.
Fuente: Elaboración Propia.

4.6.4. Arquitectura

Asp.Net MVC es una alternativa al modelo de formularios de Asp.Net para crear aplicativos web, este marco de presentación es fácil de manejar y comprobar que las aplicaciones con formularios web forms se integra fácilmente a las características de ASP.NET que existan.

El marco de ensamblado de MVC es System.Web.Mvc, los componentes que incluye el marco de MVC son:

Modelos: los objetos de modelo son los que implementan la lógica de dominio de datos del aplicativo. Los objetos de modelo almacenan y recuperan el estado del modelo en la base de datos. El modelo es una separación conceptual en lugar de la física.

Vistas: son los componentes que muestra la interfaz del usuario en el aplicativo. Esta interfaz se crea a partir de datos del modelo.

Controladores: son los componentes que controla la interacción con el usuario, trabajan con el modelo y esta representa una vista para entregar al usuario, en MVC solo muestra la información, el controlador responde a los datos que proporciona el usuario o cliente, con estos a su vez puede consultar la base de datos [19].

4.6.5. Diagrama de Clases.

En la Figura 48, se muestra cómo está diseñado el diagrama de clases del sistema de información en el programa StarUML.

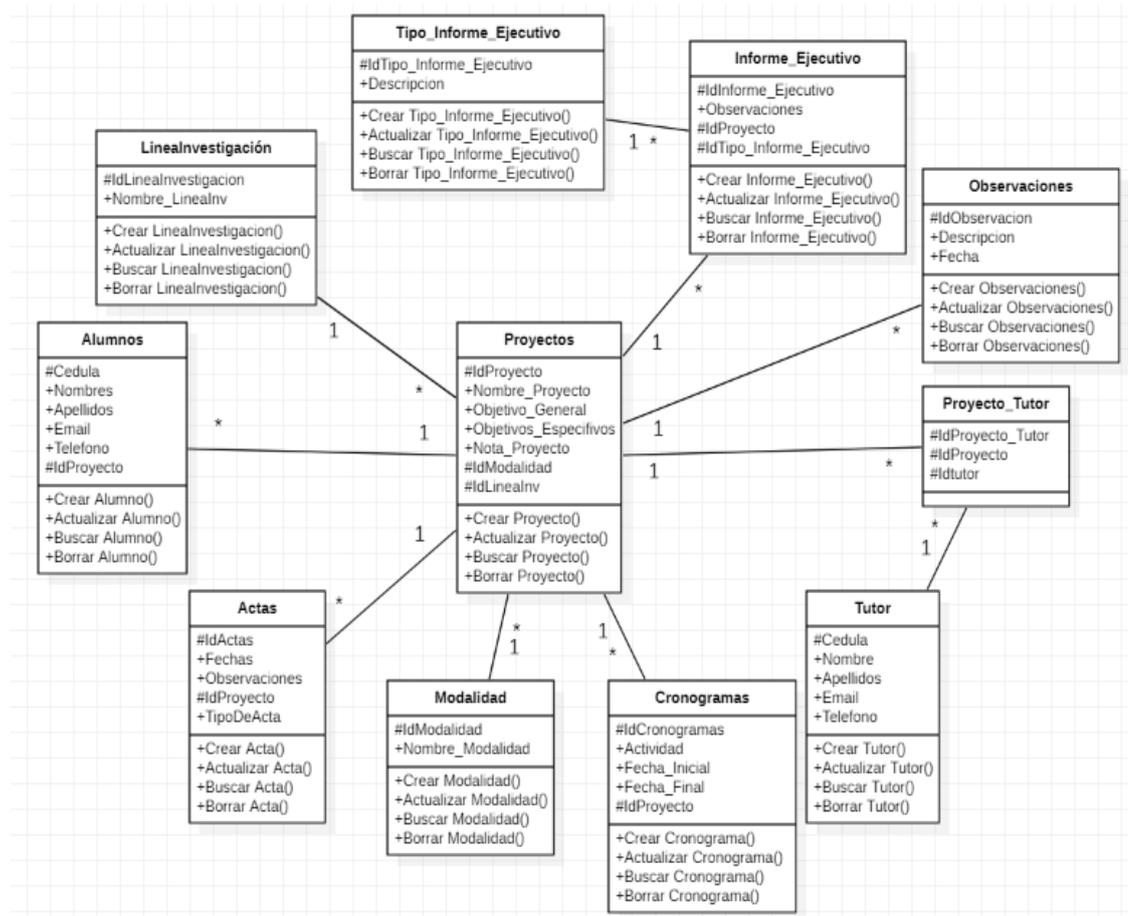


Figura 48: Diagrama de Clases.
Fuente: Elaboración Propia.

4.6.6. Interfaz Gráfica

- Pantalla Inicial.

En la Figura 49, se muestra la pantalla principal del sistema de Información.



Figura 49: Pantalla Principal del Sistema.
Fuente: Elaboración Propia.

- Pantalla de Registro de Usuarios.

En la Figura 50, se muestra la pantalla de ingresar la información de registro del estudiante.



Figura 50: Pantalla de Registrar Usuarios.
Fuente: Elaboración Propia.

- Pantalla de Iniciar Sesión.

En la Figura 51, se muestra la pantalla de ingresar el email y la contraseña para ingresar al sistema de información.



Figura 51: Pantalla de Iniciar Sesión.
Fuente: Elaboración Propia.

- Pantalla de Proyectos.

En la Figura 52, se muestra la pantalla para ingresar la información para crear el proyecto de grado del estudiante.



Figura 52: Pantalla de Crear Proyectos.
Fuente: Elaboración Propia.

- Pantalla de Actas.

En la Figura 53, se muestra la pantalla para ingresar la información de la creación del acta Inicial del proyecto.

The screenshot shows a web interface for 'ACTAS DE PROYECTOS DE GRADO'. At the top, there is a header with the university logo and name 'UNIVERSIDAD CATOLICA DE PEREIRA', navigation links 'Acerca de' and 'Contacto', and user information 'Hola carolina.lima@ucp.edu.col' and 'Cerrar sesión'. The main content area is titled 'ACTAS DE PROYECTOS DE GRADO' and 'Crear Acta'. It contains a form with the following fields: 'Nombre del Proyecto' (dropdown menu with 'Como Crear un Programa Informáticc'), 'Observaciones' (text input with 'debe mejorar arquitectura'), and 'Fecha' (text input with '22/11/2019'). Below the form are two buttons: 'Crear Acta' and 'Regresar'. The footer contains the copyright notice '© 2019 - MauricioHenao'.

Figura 53: Pantalla de Crear Actas del Proyecto.
Fuente: Elaboración Propia.

- Pantalla de Cronogramas.

En la Figura 54, se muestra la pantalla de ingresar la información para crear el cronograma del proyecto por parte del estudiante.

The screenshot shows a web interface for 'Crear Cronograma'. At the top, there is a header with the university logo and name 'Universidad Católica de Pereira', navigation links 'Acerca de' and 'Contacto', and user information 'Hola Ferman@hotmail.com!' and 'Cerrar sesión'. The main content area is titled 'Crear Cronograma' and 'Cronograma'. It contains a form with the following fields: 'Proyecto' (dropdown menu with 'Como Crear un Programa Informáticc'), 'Actividad' (text input with 'debe trabajar mas en la arquitectura'), 'Fecha Inicial' (text input with '22/11/2019'), and 'Fecha Final' (text input with '29/11/2019'). Below the form are two buttons: 'Crear' and 'Regresar'. The footer contains the copyright notice '© 2019 - MauricioHenao'.

Figura 54: Pantalla de Crear Cronogramas.
Fuente: Elaboración Propia.

- Pantalla de Observaciones.

En la Figura 55, se muestra la pantalla para ingresar la información para crear las observaciones del proyecto por parte del tutor.

The screenshot shows a web interface for creating project observations. The header includes the logo and name 'JCP - Tutor', the page title 'Observaciones', and user information 'Hola Resfa@ucp.edu.co!' with a 'Cerrar sesión' link. The main heading is 'Crear Observaciones del Proyecto' with a sub-heading 'Observaciones'. The form contains three input fields: 'Proyecto' with a dropdown menu showing 'Como Crear un Programa Informatic...', 'Descripcion' with the text 'Mejorar Tablas y Figuras', and 'Fecha' with the date '23/11/2019'. Below the fields are two buttons: 'Crear' and 'Regresar'. The footer shows the copyright '© 2019 - MaoMix'.

Figura 55: Pantalla de Crear Observaciones del Proyecto.
Fuente: Elaboración Propia.

- Pantalla de Permisos Administrativos.

En la Figura 56, se muestra la pantalla de gestionar los permisos que concede el administrador del sistema de información.

The screenshot shows a web interface for managing system permissions. The header includes the logo and name 'UNIVERSIDAD CATOLICA DE PEREIRA', navigation links 'Acerca de' and 'Contacto', user information 'Hola carolina.lima@ucp.edu.co!' with a 'Cerrar sesión' link, and the page title 'PERMISOS DEL SISTEMA'. The main content is a table with two columns: 'Nombre' and 'Email'. The table lists several users and their roles. Below the table is a 'Regresar' button. The footer shows the copyright '© 2019 - MauricioHenao'.

Nombre	Email	Roles
3@gmail.com	3@gmail.com	Roles
Ricar@yahoo.com	Ricar@yahoo.com	Roles
mau@gmail.com	mau@gmail.com	Roles
2@gmail.com	2@gmail.com	Roles
Resfa@ucp.edu.co	Resfa@ucp.edu.co	Roles
Marc@ucp.edu.co	Marc@ucp.edu.co	Roles
Fernan@hotmail.com	Fernan@hotmail.com	Roles
carolina.lima@ucp.edu.co	carolina.lima@ucp.edu.co	Roles

Figura 56: Pantalla de gestionar Permisos del Sistema.
Fuente: Elaboración Propia.

- Pantalla de Modalidad del Proyecto.

En la Figura 57, se muestra la pantalla de ingresar información para crear la modalidad del proyecto por parte del administrador.



Figura 57: Pantalla de Crear Modalidad del Proyecto.
Fuente: Elaboración Propia.

- Pantalla de Línea de Investigación.

En la Figura 58, se muestra la pantalla de ingresar información para crear la Línea de Investigación el proyecto por parte del administrador.



Figura 58. Pantalla de Crear Línea De Investigación del Proyecto.
Fuente: Elaboración Propia.

4.7. CONCRECIÓN DEL MODELO

4.7.1. IMPLEMENTACIÓN

Una vez que sabe qué funciones debe desempeñar el sistema de información (análisis) y se determina cómo se organizarán sus distintos componentes (diseño), se pasa a la etapa de implementación. Antes de desarrollar el código, es fundamental haber comprendido bien el problema que se pretende resolver y haber aplicado principios básicos de diseño que permitan construir un sistema de información de calidad.

En la fase de implementación se determinarán las herramientas adecuadas, un entorno de desarrollo que facilite este desarrollo y un lenguaje de programación apropiado para este tipo de sistema que se creará. La elección de estas herramientas dependerá en gran parte de las decisiones de diseño y del entorno en el que el sistema deberá funcionar.

A continuación, se mostrará el resultado de la implementación del sistema, en esta parte del programa se encuentra la página principal del sistema de gestión, como se evidencia en la Figura 59.



Figura 59: Implementación Pantalla principal.
Fuente: Elaboración Propia.

En esta parte del sistema es donde se deben de registrar todos los usuarios del sistema como son alumnos, tutores y administrador con los campos nombres,

apellidos, email, teléfono, cedula, contraseña, confirmar contraseña. Además se muestra en la Figura 60 parte del código de dicha función del programa.



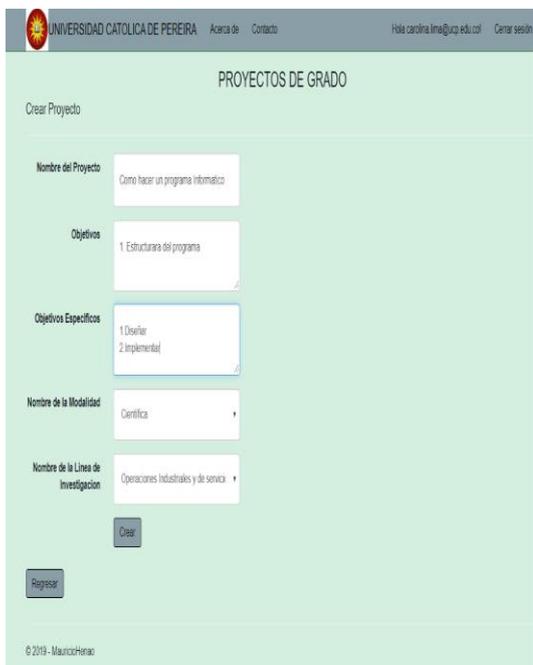
```

case SignInStatus.Success:
    var usr = UserManager.FindByName(model.Email);
    if (UserManager.IsInRole(usr.Id, "User"))
    {
        returnUrl = "/Home/User";
    }
    if (UserManager.IsInRole(usr.Id, "Tutor"))
    {
        returnUrl = "/Home/Tutor";
    }
    if (UserManager.IsInRole(usr.Id, "Admin"))
    {
        returnUrl = "/Home/Admin";
    }
    return RedirectToLocal(returnUrl);

```

Figura 60: implementación registro de Usuarios.
Fuente: Elaboración Propia.

En esta parte del programa es donde se crean los proyectos de grado de los estudiantes de la Universidad Católica de Pereira. En la Figura 61 se muestra parte del código requerido para dicha función.



```

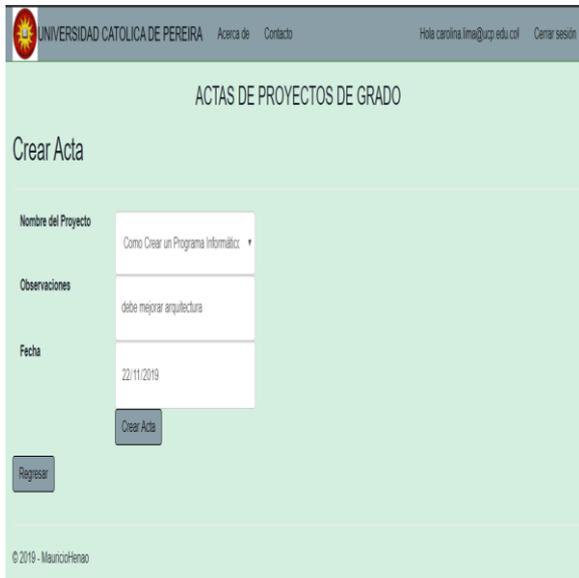
Proyecto proyecto = db.Proyecto.Find(id);
if (proyecto == null)
{
    return HttpNotFound();
}
return View(proyecto);
}

// GET: Proyectos/Create
// [Authorize(Roles = "Admin, User")]
public ActionResult Create()
{
    ViewBag.IdLinealnv = new SelectList(db.LinealInvestigacion, "IdLinealnv", "Nombre_Linealnv");
    ViewBag.IdModalidad = new SelectList(db.Modalidad, "IdModalidad", "Nombre_Modalidad");
    return View();
}

```

Figura 61: Implementación de Crear Proyectos.
Fuente: Elaboración Propia.

En esta parte del sistema es donde se crean las actas de proyecto, esta es creada por el administrador al momento de crear el proyecto, además se muestra parte del código como se ve en la Figura 62 .



The screenshot shows a web application interface for creating project records. The header includes the university logo and navigation links. The main content area is titled 'ACTAS DE PROYECTOS DE GRADO' and contains a form titled 'Crear Acta'. The form has three input fields: 'Nombre del Proyecto' (with a dropdown menu), 'Observaciones' (with a text input), and 'Fecha' (with a date picker). Below the form are 'Crear Acta' and 'Regresar' buttons. The footer shows the copyright year 2019 and the author's name, Mauricio Henao.

```

Actas actas = db.Actas.Find(id);
if (actas == null)
{
    return HttpNotFound();
}
return View(actas);
}

// GET: Actas/Create
[Authorize(Roles = "Admin")]
public ActionResult Create()
{
    ViewBag.IdProyecto = new SelectList(db.Proyecto, "IdProyecto", "Nombre_Proyecto");
    return View();
}

```

Figura 62: implementación Crear Actas del Proyecto.
Fuente: Elaboración Propia.

En esta parte del programa es donde el estudiante puede crear el cronograma para el proyecto, esto con el fin de llevar una organización de las actividades que se tengan en el proyecto, además se muestra parte del código de la función como se muestra en la Figura 63.



The screenshot shows a web application interface for creating project schedules. The header includes the university logo and navigation links. The main content area is titled 'Crear Cronograma' and contains a form titled 'Cronograma'. The form has four input fields: 'Proyecto' (with a dropdown menu), 'Actividad' (with a text input), 'Fecha Inicial' (with a date picker), and 'Fecha Final' (with a date picker). Below the form are 'Crear' and 'Regresar' buttons. The footer shows the copyright year 2019 and the author's name, Mauricio Henao.

```

Cronograma cronograma = db.Cronogramas.Find(id);
if (cronograma == null)
{
    return HttpNotFound();
}
return View(cronograma);
}

// GET: Cronogramas/Create
public ActionResult Create()
{
    ViewBag.IdProyecto = new SelectList(db.Proyecto, "IdProyecto", "Nombre_Proyecto");
    return View();
}

```

Figura 63: Implementación Crear Cronogramas del Proyecto.
Fuente: Elaboración Propia.

En esta parte del programa es donde el tutor podrá hacer las observaciones pertinentes y así hacer un seguimiento efectivo del proyecto, además se muestra parte del código de la función como se muestra en la Figura 64.

```
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create([Bind(Include =
"IdObservaciones,Descripcion,Date,IdProyecto")] Observaciones observaciones)
{
    if (ModelState.IsValid)
    {
        db.Observaciones.Add(observaciones);
        db.SaveChanges();
        return RedirectToAction("Index");
    }

    ViewBag.IdProyecto = new SelectList(db.Proyecto, "IdProyecto",
"Nombre_Proyecto", observaciones.IdProyecto);
    return View(observaciones);
}
```

Figura 64: Implementación Crear Observaciones del Proyecto.
Fuente: Elaboración Propia.

En esta parte del programa es donde el administrador del sistema dará permisos (Roles) a cada uno de los usuarios del sistema, además se muestra parte del código de la función como muestra la Figura 65.

Nombre	Email	Roles
3@gmail.com	3@gmail.com	Roles
Ricar@yahoo.com	Ricar@yahoo.com	Roles
mau@gmail.com	mau@gmail.com	Roles
2@gmail.com	2@gmail.com	Roles
Resfa@ucp.edu.co	Resfa@ucp.edu.co	Roles
Marc@ucp.edu.co	Marc@ucp.edu.co	Roles
Fernan@hotmail.com	Fernan@hotmail.com	Roles
carolina.lima@ucp.edu.co	carolina.lima@ucp.edu.co	Roles

```

case SignInStatus.Success:
    var usr = UserManager.FindByName(model.Email);
    if (UserManager.IsInRole(usr.Id, "User"))
    {
        returnUrl = "/Home/User";
    }
    if (UserManager.IsInRole(usr.Id, "Tutor"))
    {
        returnUrl = "/Home/Tutor";
    }
    if (UserManager.IsInRole(usr.Id, "Admin"))
    {
        returnUrl = "/Home/Admin";
    }
}
return RedirectToLocal(returnUrl);

```

Figura 65: Implementación Gestionar permisos del Sistema.
Fuente: Elaboración Propia.

En esta parte del programa el administrador del sistema podrá crear las modalidades de la carrera ingeniería industrial de la Universidad Católica de Pereira, además se muestra parte de código de la función como muestra la Figura 66.

```

[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create
    ([Bind(Include = "IdModalidad,Nombre_Modalidad")] Modalidad
modalidad)
{
    if (ModelState.IsValid)
    {
        db.Modalidad.Add(modalidad);
        db.SaveChanges();
        return RedirectToAction("Index");
    }
    return View(modalidad);
}

```

Figura 66: Implementación Crear modalidad del Proyecto.
Fuente: Elaboración Propia.

En esta parte del sistema el administrador podrá crear las líneas de investigación que crea convenientes para la carrera de ingeniería industrial de la Universidad Católica de Pereira, además se muestra parte del código de la función como muestra la Figura 67.

```
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create
    ([Bind(Include = "IdLineaInv,Nombre_LineaInv")] LineaInvestigacion
lineaInvestigacion)
{
    if (ModelState.IsValid)
    {
        db.LineaInvestigacion.Add(lineaInvestigacion);
        db.SaveChanges();
        return RedirectToAction("Index");
    }

    return View(lineaInvestigacion);
}
```

Figura 67: Implementación Crear línea de Investigación.
Fuente: Elaboración Propia.

En esta parte del sistema se muestra cómo se crea la nota para el proyecto, además se muestra parte del código de la función como esta en la Figura 68.

```

@Html.ValidationSummary(true, "", new { @class = "text-danger" })
    <div class="form-group">
        @Html.LabelFor(model => model.IdActas, "Nota del Proyecto",
htmlAttributes: new { @class = "control-label col-md-2" })
        <div class="col-md-10">
            <input type="number" id="NotaProyecto" name="NotaProyecto"
step="0.1">

```

Figura 68: Implementación Crear Nota del Proyecto.
Fuente: Elaboración Propia.

4.7.2. PRUEBAS

La etapa de pruebas tiene como objetivo detectar problemas que hayan quedado después de la implementación del proyecto y que sean importantes para cumplir con los requisitos. La búsqueda de errores que se realiza en la etapa de pruebas pueda encontrarse errores que antes no pudieron ser detectados.

En la Figura 69 se presentan las pruebas realizadas en el programa Visual Studio, al Sistema de Gestión para los proyectos de grado de la Universidad Católica de Pereira:

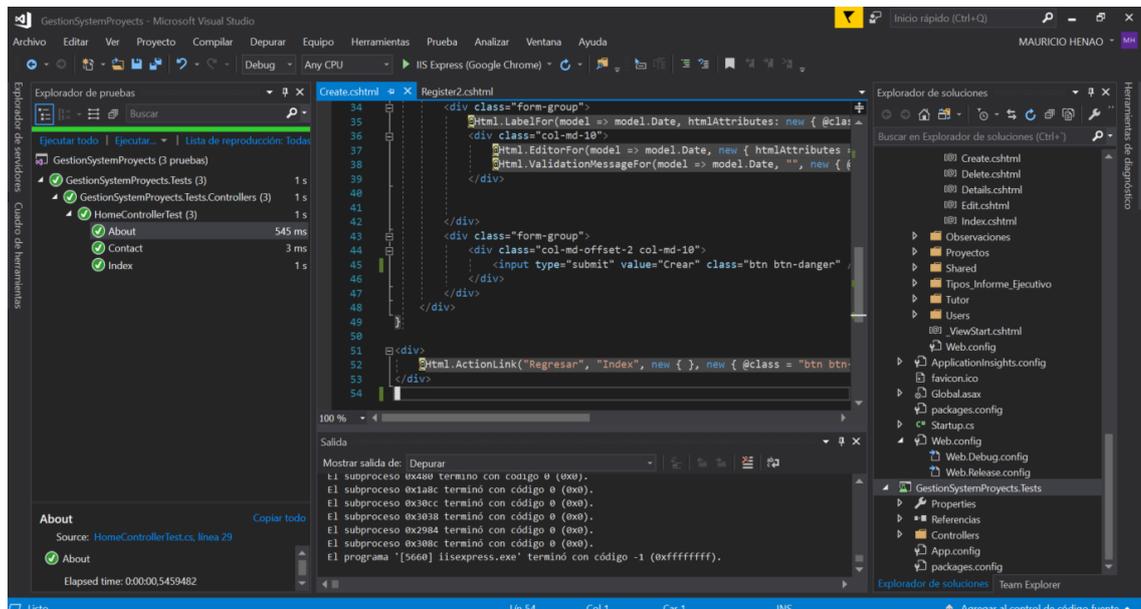


Figura 69: Pruebas Unitarias del Sistema.
Fuente: Elaboración Propia.

4.7.2.1. ¿Qué son Pruebas unitarias?

Se utilizan para preservar el estado del código, garantizar cobertura del código, y detectar errores y fallos antes de que lo hagan los usuarios, se deben de ejecutar las pruebas unitarias a menudo para asegurarse que el código funciona correctamente.

En la Figura 70 se muestra la manera como se elige el proyecto al cual se le va a hacer la prueba unitaria.

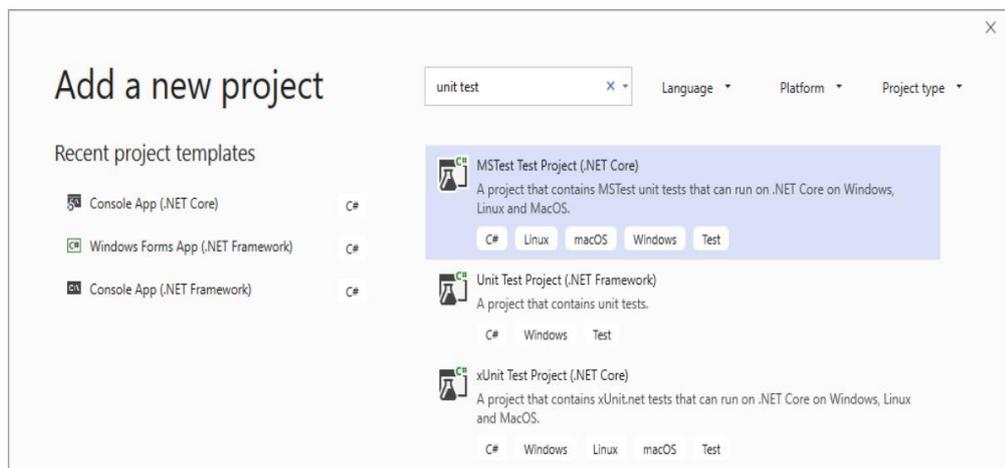


Figura 70: Pruebas Unitarias: Seleccionar Proyecto para Pruebas.
Fuente: Elaboración propia basado en [28].

En la Figura 71 se muestra el proyecto de prueba unitaria, se agrega una referencia al proyecto que quiera probarse y se crea.

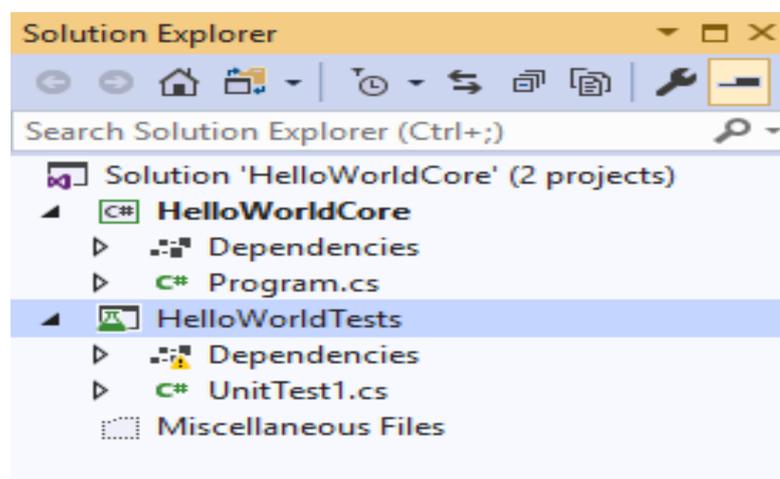


Figura 71: Pruebas Unitarias: test del Sistema.
Fuente: Basado en [28].

En la Figura 72 se agrega el código al método de prueba unitaria.

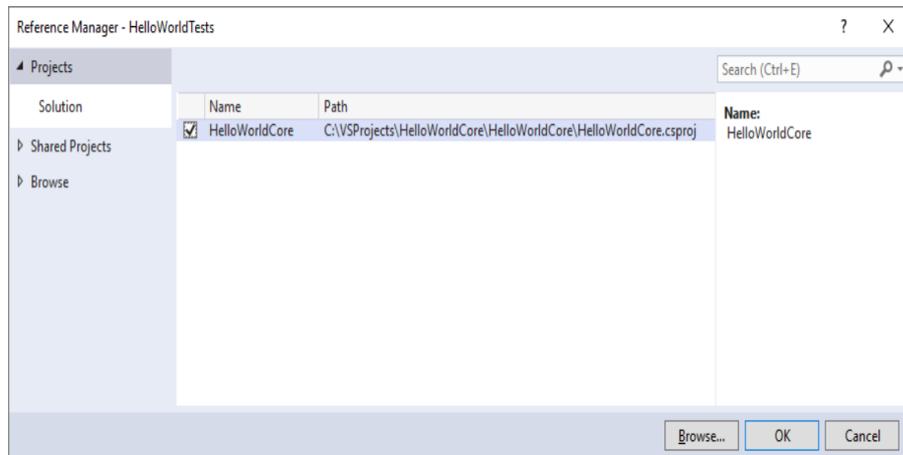


Figura 72: Pruebas Unitarias: Guardado del Test.
Fuente: Basado en [28].

En la Figura 73 se muestra el código para ejecutar las pruebas unitarias.

```
namespace GestionSystemProjects.Controllers
{
    ///[Authorize(Roles = "Admin, user")]
    public class ProyectosController : Controller
    {
        private Context.GestionSystemProjectsContext db = new
        Context.GestionSystemProjectsContext();
        private ApplicationDbContext Conexion = new ApplicationDbContext();
        // GET: Proyectos
        public ActionResult Index()
        {
            var userId = User.Identity.GetUserId();
            var userManager = new UserManager<ApplicationUser>(new
            UserStore<ApplicationUser>(Conexion));
            var usr = userManager.FindById(userId);
            var proyecto = db.Proyecto.Include(p => p.LineaInvestigacion)
                .Include(p => p.Modalidad);
            if (userManager.IsInRole(usr.Id, "User"))
            {
                ApplicationUser userView = userManager.FindById(userId);
                proyecto = db.Proyecto.Include(p => p.LineaInvestigacion)
                    .Include(p => p.Modalidad)
                    .Where(u => u.IdProyecto == userView.IdProyecto);
                List<string> Tutores = new List<string>();
            }
        }
    }
}
```

Figura 73: Pruebas Unitarias: Código del Sistema.
Fuente: Elaboración Propia.

En la Figura 74 se muestra cómo se abre el explorador de pruebas y desde la barra de menú se le da ejecutar la prueba.

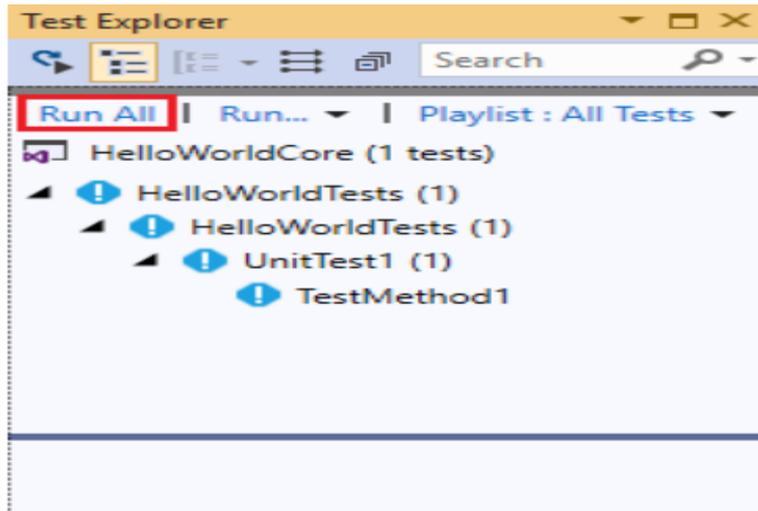


Figura 74: Pruebas Unitarias: Pruebas del Test.
Fuente: Basado en [28].

En la Figura 75 se muestra como se ejecuten las pruebas y si salen exitosas se mostrará un icono verde en el proyecto, y con un icono X indica que las pruebas salieron fallidas.

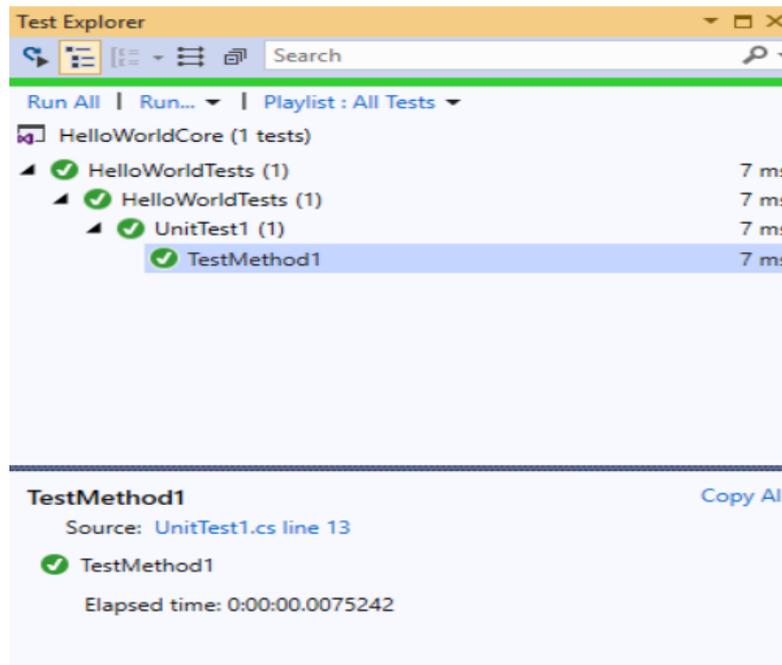


Figura 75. Pruebas Unitarias: Pruebas exitosas.
Fuente: Basado en [28].

En la Figura 76, se muestra cómo se registra un estudiante en el sistema.

The screenshot shows a registration form titled "Registrar Nuevo Estudiante." with the subtitle "Cree un nuevo registro." The form includes the following fields: "Nombres", "Apellidos", "Email", "Telefono", "Cedula", "Contraseña", and "Confirmar contraseña". A "Guardar" button is located below the "Contraseña" field. At the bottom left, there is a "Registro Tutor" button. The top navigation bar includes the university logo, "UNIVERSIDAD CATOLICA DE PEREIRA", and links for "Iniciar sesión", "Registrarse", "Acerca de", and "Contacto".

Figura 76: Prueba de Registro del Sistema.
Fuente: Elaboración Propia.

Se guarda la información para el registro de los alumnos, como muestra la Figura 77, donde aparece el email del usuario en la parte derecha de arriba.

The screenshot displays a dashboard for "INGENIERÍA INDUSTRIAL" titled "Sistema de Información para la Gestión de Proyectos de Grado". It features an aerial photograph of the university campus. Below the photo are two main sections: "Cronogramas" with a "Cronogramas" button, and "Proyectos" with a "Crear Nuevo Proyecto" button. At the bottom, a quote reads: "Somos apoyo para llegar a ser gente, gente de bien y profesionalmente capaz". The top navigation bar shows the university logo, "Universidad Católica de Pereira", the user email "Hola Fernan@hotmail.com", and a "Cerrar sesión" link. A copyright notice "© 2019 - MauricioHenao" is visible at the bottom left.

Figura 77: Prueba de acceso a estudiantes.
Fuente: Elaboración Propia.

Se guarda información para el registro del tutor, como muestra la Figura 78.

The screenshot shows a registration form titled "Registrar Nuevo Tutor." with the subtitle "Cree una cuenta nueva." The form includes the following fields: "Nombres", "Apellidos", "Email", "Telefono", "Cedula", "Contraseña", and "Confirmar contraseña". A "Registrar Tutor" button is located at the bottom of the form. The page header includes the university logo and name, along with links for "Iniciar sesión", "Registrarse", "Acerca de", and "Contacto". The footer contains the copyright notice "© 2019 - MauricioHenao".

Figura 78: Prueba de Acceso Tutor.
Fuente: Elaboración Propia.

Se hace la prueba para verificar que el tutor acceda a la pantalla correcta como muestra la Figura 79.

The screenshot displays the tutor dashboard. The header shows the user's profile as "UCP - Tutor" with "Observaciones" and the email "Hola Resfa@ucp.edu.co! Cerrar sesión". The main content area features a large "TUTOR" title, a left navigation arrow, and two primary options: "Observaciones Del Proyecto" and "Nota Del Proyecto", each with an "Ingresar" button. The footer includes the copyright notice "© 2019 - MaoMix".

Figura 79. Pruebas de Logueo para Tutor.
Fuente: Elaboración Propia.

Se guarda la información del administrador como muestra la Figura 80.



Figura 80: Pruebas de Acceso al Administrador.
Fuente: Elaboración Propia.

Se hace la prueba para verificar que el administrador acceda a la pantalla correcta como muestra la Figura 81.

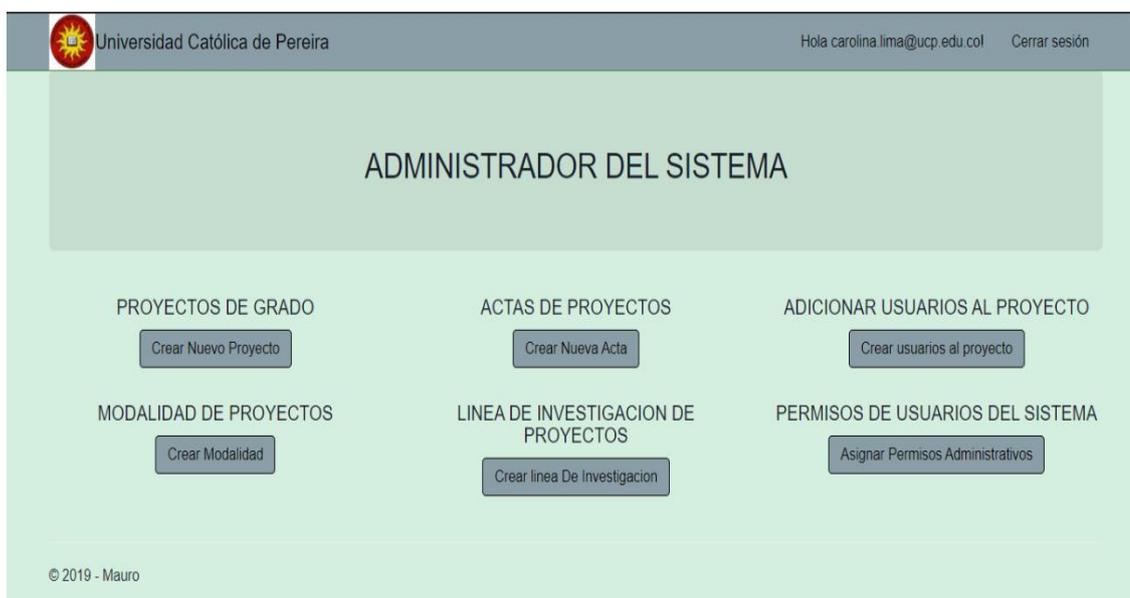


Figura 81: Pruebas de Acceso al Administrador.
Fuente: Elaboración Propia.

Se guarda la información de la creación del proyecto de grado como muestra la Figura 82.

Figura 82: Pruebas de Crear un Proyecto.
Fuente: Elaboración Propia.

Se hace la prueba de que el proyecto quede guardado en el sistema como muestra la Figura 83.

Nombre Del Proyecto	Objetivos Generales	Objetivos Especificos	Modalidad	Linea de Investigacion	Nota
Como Crear un Programa Informático	1. Como crear un programa desde cero	1. Diseñar 2.. Implementar	Cientifica	Operaciones Industriales y de servicios	Editar Detalles Eliminar

Figura 83: Pruebas de Acceso de Crear un Proyecto.
Fuente: Elaboración Propia.

Se guarda la información de la actividad o cronograma que se tiene para seguir un orden de trabajo con el alumno y el tutor, como se muestra en la Figura 84.

Universidad Católica de Pereira Hola Fernan@hotmail.com! Cerrar sesión

Crear Cronograma

Cronograma

Proyecto Como Crear un Programa Informáticc ▾

Actividad debe trabajar mas en la arquitectura

Fecha Inicial 22/11/2019

Fecha Final 29/11/2019

© 2019 - MauricioHenao

Figura 84: Prueba de Crear un Cronograma.
Fuente: elaboración Propia.

Se hace la prueba para saber que el sistema guarda la información en el sistema como muestra la Figura 85.

Universidad Católica de Pereira Hola Fernan@hotmail.com! Cerrar sesión

Cronograma del Proyecto

Nombre_Proyecto	Actividad	Date	Date1	
Como Crear un Programa Informático	Se debe de trabajar mas en la arquitectura	21/11/2019	30/11/2019	Editar Detalles Eliminar

© 2019 - MauricioHenao

Figura 85: Prueba de Crear un Cronograma.
Fuente: Elaboración Propia.

Se guarda la información de la actividad o cronograma que se tiene para seguir un orden de trabajo con el alumno y el tutor, como se muestra en la Figura 84.

The screenshot shows a web interface for creating project observations. At the top, there is a header with the UCP - Tutor logo, the text 'Observaciones', and user information 'Hola Resfa@ucp.edu.col' and 'Cerrar sesión'. The main heading is 'Crear Observaciones del Proyecto' with a sub-heading 'Observaciones'. The form contains three input fields: 'Proyecto' with a dropdown menu showing 'Como Crear un Programa Informáticc', 'Descripcion' with a text input containing 'Debe de mejorar la arquitectura', and 'Fecha' with a date input showing '22/11/2019'. Below these fields are two buttons: 'Crear' and 'Regresar'. At the bottom left, there is a copyright notice '© 2019 - MaoMix'.

Figura 86: Prueba de Crear Observaciones del Proyecto.
Fuente: Elaboración Propia.

Se hace la prueba de que el sistema guarda los datos almacenados como muestra la Figura 87.

The screenshot shows a web interface displaying a list of project observations. At the top, there is a header with the UCP - Tutor logo, the text 'Observaciones', and user information 'Hola Resfa@ucp.edu.col' and 'Cerrar sesión'. The main heading is 'Observaciones del Proyecto'. Below the heading is a table with three columns: 'Nombre_Proyecto', 'Descripcion', and 'Date'. The table contains one row of data. Below the table are two buttons: 'Regresar' and 'Crear Observacion'. At the bottom left, there is a copyright notice '© 2019 - MaoMix'.

Nombre_Proyecto	Descripcion	Date	
Como Crear un Programa Informático	Debe de mejorar la arquitectura	22/11/2019 12:00:00 a. m.	Editar Detalles Eliminar

Figura 87: Prueba de Crear Observaciones del Proyecto.
Fuente: Elaboración Propia.

En la prueba de creación de usuarios el sistema muestra el email de la persona que se registra en el sistema como se aprecia en la Figura 88.

Nombre	Email	Roles
3@gmail.com	3@gmail.com	Roles
Ricar@yahoo.com	Ricar@yahoo.com	Roles
mau@gmail.com	mau@gmail.com	Roles
2@gmail.com	2@gmail.com	Roles
Resfa@ucp.edu.co	Resfa@ucp.edu.co	Roles
Marc@ucp.edu.co	Marc@ucp.edu.co	Roles
Fernan@hotmail.com	Fernan@hotmail.com	Roles
carolina.lima@ucp.edu.co	carolina.lima@ucp.edu.co	Roles

Regresar

© 2019 - MauricioHenao

Figura 88: Pruebas de Gestionar Permisos del Sistema.
Fuente: Elaboración Propia.

Se hace la prueba donde el sistema guarda los roles que se les dan a todos los usuarios, cómo muestra la Figura 89.

ROLES PARA USUARIOS DEL SISTEMA

Roles para el usuario: carolina.lima@ucp.edu.co

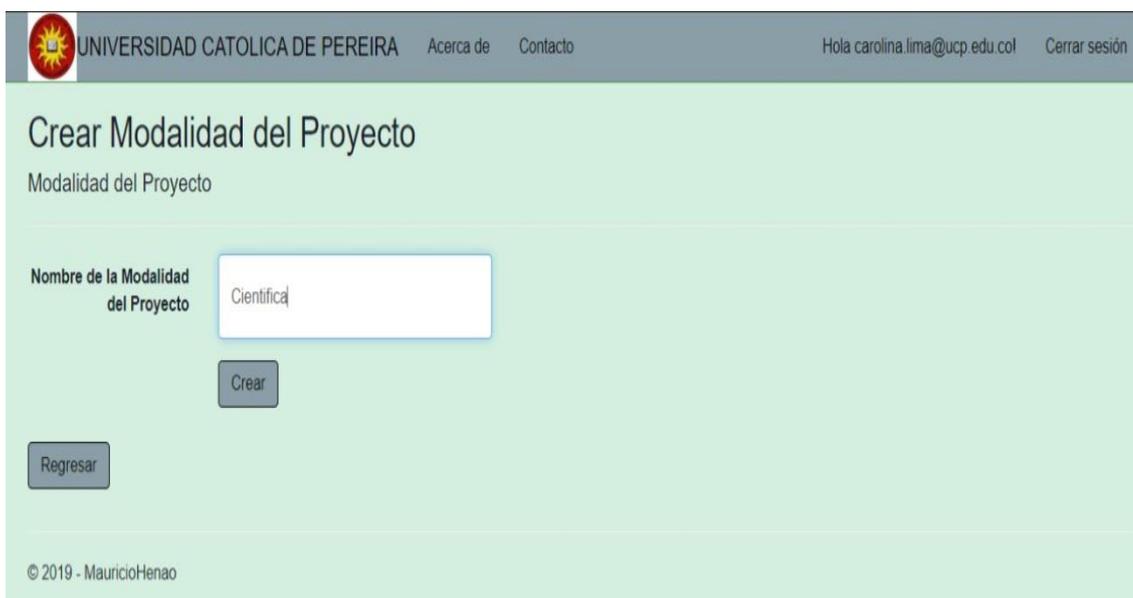
User	Eliminar Rol de Usuario
Tutor	Eliminar Rol de Usuario
Admin	Eliminar Rol de Usuario

Adicionar Rol Regresar a Usuarios

© 2019 - MauricioHenao

Figura 89: Pruebas de Gestionar Permisos del Sistema.
Fuente: elaboración Propia.

En la prueba de modalidades del proyecto se ingresa la modalidad que se requiere y se le da al botón crear, como muestra la Figura 90.



The screenshot shows the 'Crear Modalidad del Proyecto' (Create Project Modality) form. At the top, there is a header with the university logo and name 'UNIVERSIDAD CATOLICA DE PEREIRA', navigation links 'Acerca de' and 'Contacto', and user information 'Hola carolina.lima@ucp.edu.co!' and 'Cerrar sesión'. The main heading is 'Crear Modalidad del Proyecto' with the subtitle 'Modalidad del Proyecto'. The form contains a label 'Nombre de la Modalidad del Proyecto' next to a text input field containing the word 'Científica'. Below the input field is a 'Crear' button. To the left of the 'Crear' button is a 'Regresar' button. At the bottom left, there is a copyright notice '© 2019 - MauricioHenao'.

Figura 90: Prueba de Crear Modalidad del Proyecto.
Fuente: Elaboración Propia.

Se hace la prueba donde el sistema guarda la modalidad que va a tener el proyecto, cómo muestra la Figura 91.



The screenshot shows the 'MODALIDAD DEL PROYECTO' (Project Modality) list view. The header is the same as in Figure 90. The main heading is 'MODALIDAD DEL PROYECTO'. Below it, there is a section titled 'Modalidad' containing a list item for 'Científica'. To the right of 'Científica' are the actions 'Editar | Detalles | Eliminar'. Below the list item are two buttons: 'Crear Modalidad del Proyecto' and 'Regresar'. At the bottom left, there is a copyright notice '© 2019 - MauricioHenao'.

Figura 91: Prueba para Crear Modalidad del Proyecto.
Fuente: Elaboración Propia.

En la prueba de creación de línea de investigación se ingresa información y se le da crear, como muestra la Figura 92.



The screenshot shows a web interface for creating a research line. At the top, there is a header with the university logo and name 'UNIVERSIDAD CATOLICA DE PEREIRA', navigation links 'Acerca de' and 'Contacto', and user information 'Hola carolina.lima@ucp.edu.col' and 'Cerrar sesión'. The main heading is 'Crear Línea de Investigación'. Below it, the text 'Línea de Investigación' is displayed. A form field labeled 'Nombre de la Línea de Investigación del Proyecto' contains the text 'Operaciones Industriales y de servicios'. Below the form field is a 'Crear' button. To the left of the form field is a 'Regresar' button. At the bottom left, there is a copyright notice '© 2019 - MauricioHenao'.

Figura 92: Prueba de Crear Línea de Investigación del Proyecto.
Fuente: Elaboración Propia.

Se hace la prueba para ver si el sistema guarda la información digitada, como muestra la Figura 93.



The screenshot shows the result of creating a research line. The header is the same as in Figure 92. The main heading is 'LINEA DE INVESTIGACIÓN DEL PROYECTO'. Below it, the text 'Línea de Investigación' is displayed. The text 'Operaciones Industriales y de servicios' is shown, followed by links 'Editar | Detalles | Eliminar'. Below this, there are two buttons: 'Crear Línea de Investigación del Proyecto' and 'Regresar'. At the bottom left, there is a copyright notice '© 2019 - MauricioHenao'.

Figura 93: Prueba de Crear Línea de Investigación del Proyecto.
Fuente: Elaboración Propia.

En la prueba de creación de actas se selecciona proyecto, se ingresa la información que se requiere para crear el acta como muestra la Figura 94.

UNIVERSIDAD CATOLICA DE PEREIRA Acerca de Contacto Hola carolina.lima@ucp.edu.col Cerrar sesión

ACTAS DE PROYECTOS DE GRADO

Crear Acta

Nombre del Proyecto Como Crear un Programa Informático ▾

Observaciones debe mejorar arquitectura

Fecha 22/11/2019

© 2019 - MauricioHenao

Figura 94: Prueba de Crear Actas del Proyecto.
Fuente: Elaboración Propia.

Se hace la prueba para observar si el sistema guarda la información suministrada, como muestra la Figura 95.

UNIVERSIDAD CATOLICA DE PEREIRA Acerca de Contacto Hola carolina.lima@ucp.edu.col Cerrar sesión

ACTA DE INICIACIÓN DEL PROYECTO

Fecha del Acta: 22/11/2019 12:17:34 a. m.

Participantes del Proyecto	Ricardo Alberto Goenz Jimenez
Cédula	10231654
Participantes del Proyecto	Fernando Leon Aparicio Gomez
Cédula	10025297
Sustentación	Acta Inicial ▾
Nombre del Proyecto	Como Crear un Programa Informático
Nombre del Tutor	Rafael Tula Garcia Renfijo ▾
Objetivos Generales	1. Como crear un programa desde cero
Objetivos Especificos	1. Diseñar 2. Implementar
Modalidad del Proyecto	Científica
Línea de Investigación	Operaciones Industriales y de servicios

Cronograma de Actividades	Descripción de la actividad	Fecha Inicial	Fecha Final
	Se debe de trabajar más en la arquitectura	21/11/2019	30/11/2019

Observaciones	Fecha de la Observación
trabajar más en el objetivo general	29/11/2019

- COMPROMISOS DEL (LOS) ESTUDIANTE (S):

- De acuerdo al artículo 21 del reglamento de los trabajos de grado.
 - a. Programar con el director de trabajo de grado el cronograma de las asesorías y cumplir con el mismo.
 - b. Informar de inmediato y por escrito, al Director del trabajo y a la dirección del programa respectivo, sobre cualquier cambio o asunto que dificulte y obstaculice el normal desarrollo del trabajo, para que ellos autoricen o no las modificaciones correspondientes.
 - c. Acatar las recomendaciones del Director de trabajo de grado y entregar los avances acordados con él.
 - d. Entregar el informe final del trabajo de grado al Director del programa, dentro del cronograma establecido por la universidad.
 - e. Sustentar el informe final del trabajo de grado ante el jurado que designe la Dirección del programa y atender sus observaciones y recomendaciones.
- Otros compromisos acordados con el director de trabajo de grado.

f.
g.

Firma Estudiante: _____ Firma Estudiante: _____
Firma Tutor: _____ Firma Director de Programa: _____

© 2019 - MauricioHenao

Figura 95: Prueba de Crear Acta del Proyecto.
Fuente: Elaboración Propia.

En la prueba de crear nota para proyecto se ingresan o se buscan los campos Nombre del proyecto, Nombre del tutor, Nota, como muestra la Figura 96.

Figura 96: Prueba de Crear nota del Proyecto.
Fuente: Elaboración Propia.

4.7.2.2. Prueba de Integración Descendente

La prueba de integración descendente es un enfoque incremental a la construcción de la arquitectura de software. Los módulos se integran al moverse hacia abajo a través de la jerarquía de control, comenzando con el módulo de control principal (programa principal). Los módulos subordinados al módulo de control principal se incorporan en la estructura en una forma de primero en profundidad o primero en anchura[5].

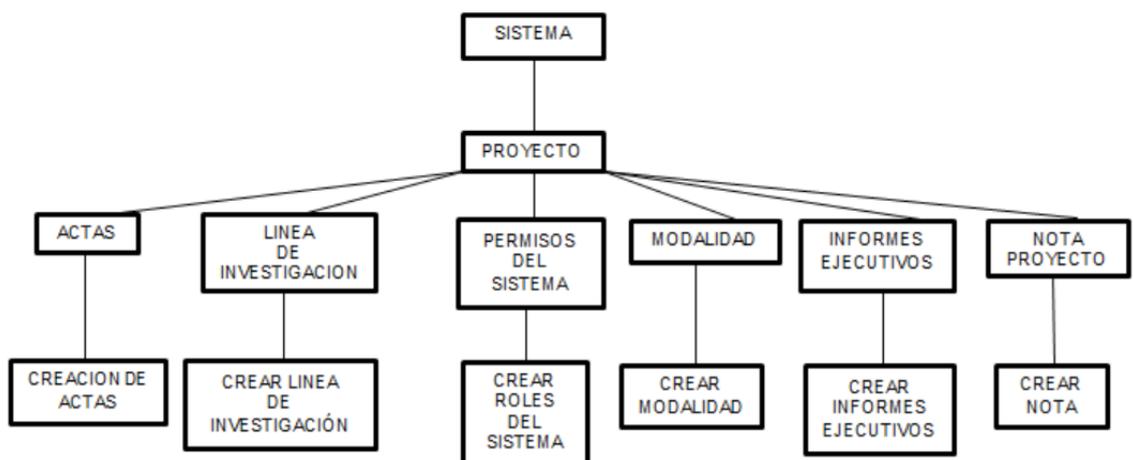


Figura 97: Prueba de Integración Descendente.
Fuente: Elaboración Propia.

4.7.2.3. Diseño de Casos de Prueba del Sistema

En la Figura 98, se muestra las pruebas funcionales que se hicieron a cada uno de los módulos del sistema de información.

PRUEBAS A EJECUTAR EN EL SISTEMA DE INFORMACIÓN PARA LA GESTIÓN DE PROYECTOS DE GRADO			RESULTADO SATISFACTORIO		
PRUEBAS FUNCIONALES DEL SISTEMA			SI	NO	OBSERVACION
ESCENARIO	NOMBRE DE LA PRUEBA A EJECUTAR	RESULTADO ESPERADO POR EL PROGRAMA DE INGENIERÍA IND	SI	NO	OBSERVACION
0	Registro de Usuarios en el sistema				
Paso 1	Ingresar Nombres	Ingreso correcto del nombre del usuario	si		
Paso 2	Ingresar Apellidos	Ingreso correcto del apellido del usuario	si		
Paso 3	Ingresar email	Ingreso correcto del email del usuario	si		
Paso 4	Ingresar Telefono	Ingreso correcto del telefono del usuario		no	El campo telefono solo admite 6 dígitos
Paso 5	Ingresar Cedula del Usuario	Ingreso correcto de la cedula del usuario	si		
Paso 6	Ingresar contraseña del Usuario	Ingreso Correcto de la contraseña del usuario	si		
Paso 7	Repetir Ingreso de Contraseña	Ingreso correcto de la repeticion de contraseña del usuario	si		
ESCENARIO	NOMBRE DE LA PRUEBA A EJECUTAR	RESULTADO ESPERADO POR EL PROGRAMA INGENIERIN IND	SI	NO	OBSERVACION
0	Iniciar Sesión en el Sistema		si		
Paso 1	Ingresar campo email	Ingreso correcto del Email del Usuario	si		
Paso 2	Ingresar campo contraseña	Ingreso correcto del campo contraseña del usuario	si		
Paso 3	Presionar boton de Iniciar sesión	Ingreso correcto al sistema	si		El usuario al ingresar al sistema aparece su email en la parte superior derecha indicando que el ingreso fue correcto
ESCENARIO	NOMBRE DE LA PRUEBA A EJECUTAR	RESULTADO ESPERADO POR EL PROGRAMA INGENIERÍA IND	SI	NO	OBSERVACION
0	Crear Proyecto de Grado		si		
Paso 1	Ingresar campo Nombre del proyecto	Ingreso correcto del nombre del proyecto	si		
Paso 2	Ingresar campo Objetivos del proyecto	Ingreso correcto del objetivo del proyecto	si		
Paso 3	Ingresar campo Objetivos especificos del proyecto	Ingreso correcto de los Objetivos especificos del proyecto	si		
Paso 4	Ingresar el campo modalidad del proyecto	Ingreso correcto de la modalidad del proyecto	si		En el momento de crear el proyecto el administrador debio crear previamente la modalidad en el sistema
Paso 5	Ingresar campo Linea de Investigación	Ingreso correcto de la linea de investigación del proyecto	si		En el momento de crear el proyecto el administrador debio crear previamente la linea de investigación en el sistema
Paso 6	presionar boton para crear proyecto	Creación correcta del proyecto de grado	si		
ESCENARIO	NOMBRE DE LA PRUEBA A EJECUTAR	RESULTADO ESPERADO POR EL PROGRAMA INGENIERÍA IND	SI	NO	OBSERVACION
Paso 1	Crear Cronograma para el proyecto de grado		si		
Paso 2	Ingresar campo actividad	Ingreso correcto de campo actividad	si		
Paso 3	Ingresar campo de proyecto	Ingreso correcto del campo de proyecto	si		
Paso 4	Ingresar campo fecha inicial	Ingreso correcto de fecha inicial	si		
Paso 5	Ingresar campo fecha final	Ingreso correcto de fecha final	si		
Paso 6	Presionar boton de crear cronograma	Creación correcta del cronograma del poryecto	si		
ESCENARIO	NOMBRE DE LA PRUEBA A EJECUTAR	RESULTADO ESPERADO POR EL PROGRAMA DE INGENIERÍA IND	SI	NO	OBSERVACION
Paso 1	Crear Observaciones del proyecto		si		
Paso 2	Ingresar campo de proyecto	Ingreso correcto del campo proyecto	si		
paso 3	Ingresar campo para observación del proyecto	Ingreso correcto para campo onservación delproyecto	si		
Paso 4	Ingresar campo fecha para la observación del proyecto	Ingreso correcto del campo fecha para la observación del proyecto	si		

Paso 5	Presionar boton para la creación del observaciones	Creación correcta del observaciones del proyecto	si		
ESCENARIO	NOMBRE DE LA PRUEBA A EJECUTAR	RESULTADO ESPERADO POR EL PROGRAMA INGENIERÍA IND	SI	NO	OBSERVACIÓN
Paso 1	Crear actas de proyecto		si		
Paso 2	Ingresar campo nombre del proyecto	Ingreso correcto del campo nombre del proyecto	si		
Paso 3	Ingresar campo observación del acta	Ingreso correcto del campo observacion del acta	si		que estos componen toda la acta como son:Autores del proyecto,cedula,Sustentacion entre otros.
Paso 4	Presionar boton de crear acta del proyecto	Creación correcta con el boton crear acta	si		
ESCENARIO	NOMBRE DE LA PRUEBA A EJECUTAR	RESULTADO ESPERADO POR EL PROGRAMA INGENIERÍA IND	SI	NO	OBSERVACIÓN
Paso 1	Crear modalidad para el proyecto		si		
Paso 2	Ingresar campo para la modalidad del proyecto	Ingreso correcto del campo modalidad del proyecto	si		
Paso 3	Presionar boton de creacion de la modalida del proyecto	Creación correcta de la modalidad del proyecto	si		
ESCENARIO	NOMBRE DE LA PRUEBA A EJECUTAR	RESULTADO ESPERADO POR EL PROGRAMA INGENIERÍA IND	SI	NO	OBSERVACIÓN
Paso 1	Crear linea de Investigación del proyecto		si		
Paso 2	Ingresar campo para la linea de investigación	Ingreso correcto del campo linea de investigación del proyecto	si		
Paso 3	Presionar boton de creación de la linea de investigación del proyecto	Creación correcta de la linea de investigacion del proyecto	si		

Figura 98: Diseños de Casos de Prueba.
Fuente: Elaboración propia.

4.7.2.4. Bugs del Sistema.

En la Figura 99 se muestra el código con un fallo en la tabla cronogramas e índex del sistema.

Lista de errores de búsqueda						
	Código	Descripción	Proyecto	Archivo	Lí...	Estado suprimido
✖	CS1026	Se esperaba)	1_Views_Cronogramas_In...	Index.cshtml	11	Activo
✖		An opening "(" is missing the corresponding closing ")".	GestionSystemProjects	Index.cshtml	11	

Figura 99: Bugs del Sistema.
Fuente: Elaboración Propia.

En la Figura 100, se muestra parte del código con un error de compilación en el archivo Global.asax.cs.

Error de servidor en la aplicación '/'.

Error relacionado con la red o específico de la instancia mientras se establecía una conexión con el servidor SQL Server. No se encontró el servidor o éste no estaba accesible. Compruebe que el nombre de la instancia es correcto y que SQL Server está configurado para admitir conexiones remotas. (provider: SQL Network Interfaces, error: 26 - Error al buscar el servidor o instancia especificado)

Descripción: Excepción no controlada al ejecutar la solicitud Web actual. Revise el seguimiento de la pila para obtener más información acerca del error y dónde se originó en el código.

Detalles de la excepción: System.Data.SqlClient.SqlException: Error relacionado con la red o específico de la instancia mientras se establecía una conexión con el servidor SQL Server. No se encontró el servidor o éste no estaba accesible. Compruebe que el nombre de la instancia es correcto y que SQL Server está configurado para admitir conexiones remotas. (provider: SQL Network Interfaces, error: 26 - Error al buscar el servidor o instancia especificado)

Error de código fuente:

```
Línea 75:         var roleManager = new RoleManager<IdentityRole>(new RoleStore<IdentityRole>(db));
Línea 76:
Línea 77:         if (!roleManager.RoleExists("Admin"))
Línea 78:         {
Línea 79:             roleManager.Create(new IdentityRole("Admin"));
```

Archivo de origen: C:\Users\Mauri\Desktop\GestionSystemProjects\GestionSystemProjects\Global.asax.cs **Línea:** 77

Seguimiento de la pila:

Figura 100: Bugs del Sistema.
Fuente: Elaboración Propia.

En la Figura 101, se muestra el código con un error de compilación en CreateRoles.

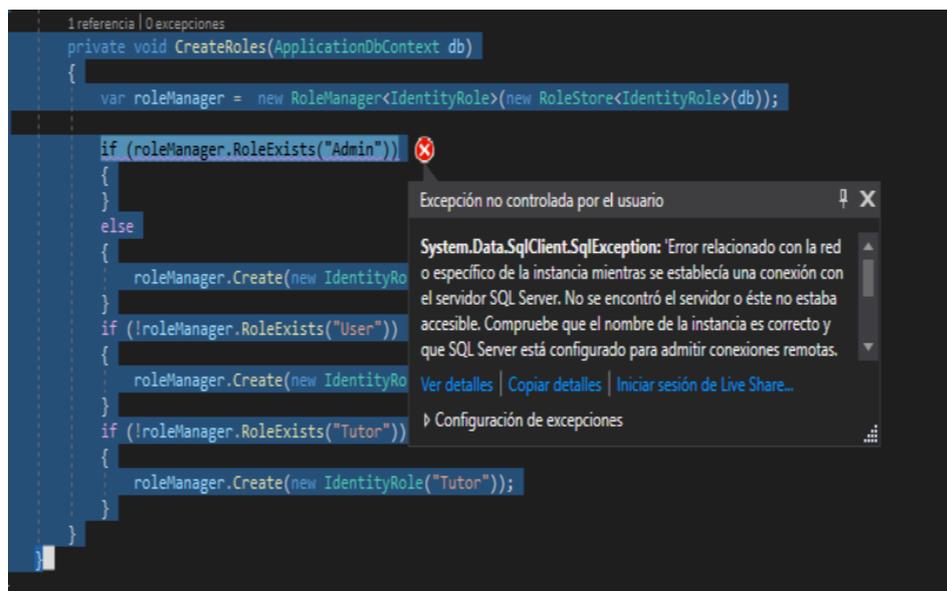


Figura 101: Bugs del Sistema.
Fuente: Elaboración Propia.

4.7.3. MANTENIMIENTO

La etapa de mantenimiento es cuando después de hacer la implementación se corrigen los defectos que se puedan presentar.

Dada la naturaleza del software, que ni se rompe ni se desgasta con el uso, su mantenimiento incluye tres facetas diferentes:

- ✓ Eliminar los defectos que se detecten durante su vida útil (mantenimiento correctivo).
- ✓ Adaptarlo a nuevas necesidades (mantenimiento adaptativo), cuando el sistema ha de funcionar sobre una nueva versión del sistema operativo o en un entorno hardware diferente, por ejemplo.
- ✓ Añadirle nueva funcionalidad (mantenimiento perfectivo), cuando se proponen características deseables que supondrían una mejora del Sistema ya existente [5].

5. CONCLUSIONES

En este trabajo se presenta el desarrollo de un sistema para la gestión de la información de los trabajos de grado del programa Ingeniería Industrial de la Universidad Católica de Pereira. Para el desarrollo del software se utilizó el modelo en cascada y se siguieron todas las etapas que éste plantea.

En la etapa del análisis se encontraron varios requisitos funcionales los cuales fueron modelados mediante diagramas de casos de uso con su respectiva especificación, además se modelaron los diagramas de secuencias. Para la implementación se utilizó Visual Studio 2017 y se modeló una base de datos que se implementó en SQL Server Management 2014.

Después de las pruebas de aceptación realizadas con el administrador del sistema, se puede concluir que el desarrollo del sistema podrá ser de ayuda para la gestión de los trabajos de grado del programa de Ingeniería Industrial.

Se tuvo múltiples problemas en la codificación del sistema, dos de ellos son, en la creación del acta del proyecto, se solucionó mediante la creación de un (create) donde llama a los campos IdActa y IdProyecto para poder traer la información que contiene el proyecto, y en la relación del estudiante con el proyecto de grado, se solucionó mediante la creación de una tabla llamada alumno donde se relaciona al proyecto por medio de la llave principal de la tabla proyecto (IdProyecto).

También se pudo evidenciar que con el Sistema de gestión de información de los proyectos de grado del programa Ingeniería Industrial de la Universidad Católica de Pereira, se avanza hacia la sistematización de la gestión los proyectos, además con este sistema de gestión se reducen las pérdidas de información y se agiliza la gestión de información tanto para alumnos como tutores del programa.

6. RECOMENDACIONES

Se recomienda utilizar la seguridad que brinda Visual Studio, porque ayudará a asegurar el acceso al código de la aplicación creada, a la seguridad basada en roles, además el programa posee una guía donde se proporcionan instrucciones para clasificar los componentes a fin de solucionar los problemas de seguridad.

Como principal inconveniente para tener en cuenta es que la distribución de los componentes obliga a crear y mantener un mayor número de archivos que ocupan mucho espacio en el disco duro, MVC no sustituye a WebForms, y además este sistema solo soporta proyectos en .net.

Como trabajo futuro se pueden mejorar algunos de los módulos, por ejemplo el de los cronogramas, donde pueden ser actualizados con un cronograma de Gantt. También se puede implementar la opción de crear actas en masa.

7. REFERENCIA

- [1] J. C. O. CAMILO ANDRÉS VACA HERNANDEZ, «SISTEMA D E INFORMACIÓN PARA LA GESTIÓN DE PROYECTOS DE GRADO EN LA CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS C ENTRO REGIONAL SOA HA "UNIPROYECTOS",» Bogota, 2012.
- [2] J. C. D. P. Marcelo Solís Poveda, «SISTEMA DE INFORMACIÓN PARA LA GESTIÓN DE LOS TRABAJOS DE GRADO.,» CARTAGENA, 2013.
- [3] E. L. N. M. CLARA BAYARRI, «GESTION DE PROYETOS FINAL DEL GRADO,» BARCELONA, ESPAÑA, 2011, 2012.
- [4] L. M. G. TAPIERO, «PROTOTIPO DE UNA APLICACIÓN WEB PARA LA GESTIÓN EN EL PROCESO DE TRABAJOS DE GRADO DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD EAN.,» BOGOTA., 2013.
- [5] R. S. PRESSMAN, «INGENIERIA DEL SOFTWARE,» 2010.
- [6] U. D. BARCELONA, «PROS Y CONTRAS DEL MODELO EN CASCADA,» BARCELONA ESPAÑA, 2018.
- [7] ACADEMIA.EDU, «VENTAJAS Y DESVENTAJAS DEL MODELO EN V,» 2018.
- [8] G. E. PHP, «Fases de Desarrollo del Modelo en Espiral,» 2009.
- [9] <https://sites.google.com/site/proyectoadpmodelosdedesarrollo/home/modelos-de-desarrollo/modelo-espinal-ventajas-y-desventajas>, «Ventajas y desventajas del Modelo en Espiral.,» 2015.
- [10] A. G. H. Dr. Francisco José García Peñalvo, «Modelo de Desarrollo Evolutivo o Prototipo,» Salamanca, España, 2018.
- [11] <https://sites.google.com/site/modelodeprototipo/etapas-para-la-elaboracion-del-modelo-de-prototipos>, «Etapas del Modelo de Desarrollo Evolutivo.,» 2015.
- [12] <https://sites.google.com/site/modelodeprototipo/ventajas-y-desventajas>, «Ventajas y Desventajas del Modelo en la Metodología de Prototipos.,» 2017.
- [13] [http://metodologiarad.weebly.com/.](http://metodologiarad.weebly.com/), «Metodología de Desarrollo RAD,» 2018.
- [14] Mario Oleg García González, «Metodologías Agiles,» Medellin, Colombia, 2016,2017.
- [15] <https://proyectosagiles.org/que-es-scrum/>, «Metodología SCRUM,» 2015.

- [16] A. Martel, «Ventajas y Desventajas de la Metodología SCRUM.,» 2016.
- [17] Mario Perez Estes, «Metodología de Programación Extrema y sus Principios.,» 2014.
- [18] <https://sites.google.com/site/xpmetodologia/marco-teorico/ventajas>, «Ventajas y Desventajas de la Programación Extrema.,» 2013.
- [19] Miguel Angel Alvarez, «Arquitectura MVC.,» 2014.
- [20] Miguel Angel Alvarez, «Que es Asp.Net.,» 2017.
- [21] <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/ef/overview>, «Que es Entity Framework.,» 2017.
- [22] <https://docs.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2019>, «Que es Visual Studio.,» 2019.
- [23] Ricardo Morales, «Que son Lenguajes de Programación.,» 2014.
- [24] Miguel Angel Alvarez, «Que es Java.,» 2019.
- [25] <https://www.php.net/manual/es/intro-what-is.php>, «Que es Php.,» 2019.
- [26] Richard Stallman, «Que es JavaScript.,» 2019.
- [27] A. Weitzenfeld, «ingeniería del Software Orientada a Objetos con Uml, Java e Internet.,» 2005.
- [28] M. V. Studio., «Introducción a la Pruebas Unitarias.,» 2017.

8. ANEXOS

8.1. Manual del Usuario

Manual del Usuario para el Sistema de Información para la Gestión de Proyectos de grado del programa Ingeniería Industrial de la Universidad Católica de Pereira.

Como empezar.

Este software se puede utilizar de una manera fácil y lograr administrar muy bien la información suministrada para este manual y para el producto.

Para empezar a navegar en este sistema de información debe de seguir unos pasos que le ayudarán a entender mejor el sistema:

En la Figura 102, se muestra la pantalla principal del sistema de información para la gestión de proyectos de grado del programa Ingeniería Industrial de la Universidad Católica de Pereira (página web).



Figura 102: Manual del Usuario: Pantalla Principal del Sistema.
Fuente: Elaboración Propia.

En la Figura 103, se muestra la pantalla para registrarse en el sistema de información, luego se debe de ingresar a la pestaña de registrar usuario, donde se le deberá ingresar la información que se requiere para el registro.



The screenshot shows a web page for 'UNIVERSIDAD CATOLICA DE PEREIRA'. The header includes navigation links: 'Iniciar sesión', 'Registrarse', 'Acerca de', and 'Contacto'. The main heading is 'Registrar Nuevo Estudiante.' with the subtext 'Cree un nuevo registro.'. The form contains several input fields: 'Nombres', 'Apellidos', 'Email', 'Telefono', 'Cedula', 'Contraseña', and 'Confirmar contraseña'. Below the fields are two buttons: 'Guardar' and 'Registro Tutor'.

Figura 103: Manual del Usuario: Registro en el sistema.
Fuente: Elaboración Propia.

En la Figura 104, se muestra como aparece el ingreso del usuario después de un registro exitoso, después de haberse registrado se visualizará los datos guardados en el parte superior derecha donde se aprecia el email del usuario en el sistema como en la Figura 105.



The screenshot shows a web page for 'INGENIERIA INDUSTRIAL' under the 'UNIVERSIDAD CATOLICA DE PEREIRA' header. The header also includes 'Iniciar sesión', 'Registrarse', 'Acerca de', and 'Contacto'. The main heading is 'INGENIERIA INDUSTRIAL'. The subtext is 'Ingrese datos para iniciar sesión.'. The form contains input fields for 'Correo electrónico' and 'Contraseña'. Below the fields is a checkbox labeled '¿Recordar cuenta?' and an 'Iniciar sesión' button. On the right side, there is a photograph of two industrial workers in blue shirts and yellow hard hats looking at a control panel. At the bottom left, there is a copyright notice: '© 2019 - MauricioHenao'.

Figura 104: Manual del Usuario: Pantalla de Logueo en el Sistema.
Fuente: Elaboración Propia.



Figura 105: Manual del Usuario: Ingreso Pantalla Principal.
Fuente: Elaboración Propia.

En la Figura 106, se muestra cómo se crea un proyecto en el Sistema de Información, el estudiante ingresará la información que se requiere del proyecto para su registro, luego le dará al botón crear para guardar la información, donde también podrá gestionar la información de una manera rápida.

Figura 106: Manual del Usuario: Creación de proyectos.
Fuente: Elaboración Propia.

En la Figura 107, después de haber creado el proyecto en el sistema, anteriormente la persona encargada de la dirección del programa de ingeniería industrial, deberá haber creado tanto la modalidad y la línea de investigación, a la vista de actas es necesario ingresar los campos para buscarlo en el sistema después de haber guardado el proyecto.



Figura 107: Manual del Usuario: Creación del Proyecto.
Fuente: Elaboración Propia.

En la Figura 108, se muestra la manera cómo se debe de ingresar la información acerca de la modalidad del proyecto, luego se le da crear, y se podrá gestionar una vez guardada en el sistema como muestra la Figura 109.

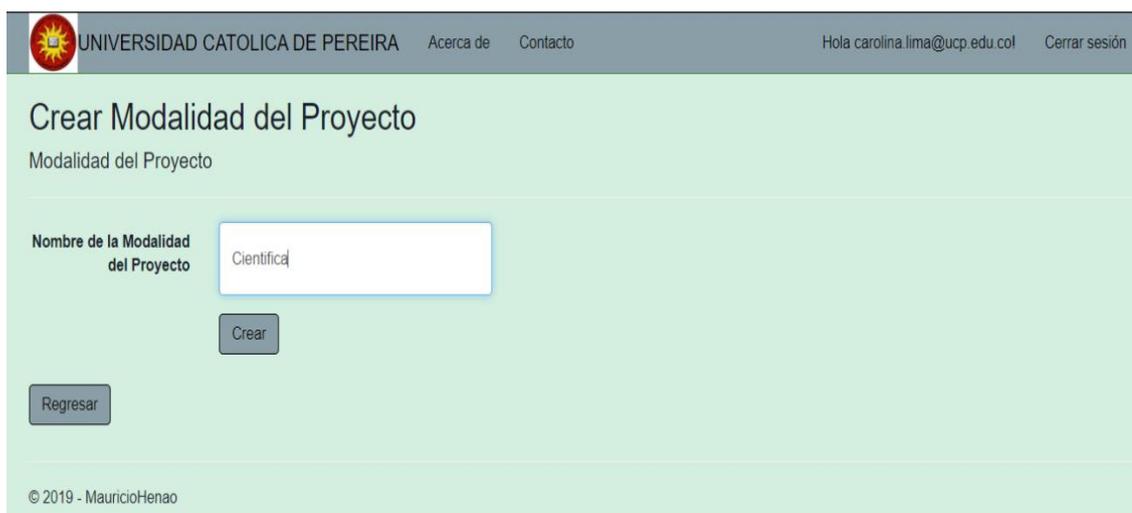


Figura 108: Manual del Usuario: Creación Modalidad del Proyecto.
Fuente: Elaboración Propia.



Figura 109: Manual del Usuario: Creación Modalidad del Proyecto.
Fuente: Elaboración Propia.

En la Figura 110, se muestra cómo se ingresa el nombre de la línea de investigación, se ven reflejadas todas las líneas de investigación guardadas por el administrador del sistema, desde aquí se puede gestionar la información como muestra la Figura 111.



Figura 110: Manual del Usuario: Creación Línea de Investigación del Proyecto.
Fuente: Elaboración Propia.



Figura 111: Manual del Usuario: Creación Línea de Investigación del Proyecto.
Fuente: Elaboración Propia.

En la Figura 112, se muestra la creación del cronograma del proyecto, después de haber creado el proyecto el estudiante deberá crear el cronograma que tendrá para llevar una gestión organizada del proyecto.

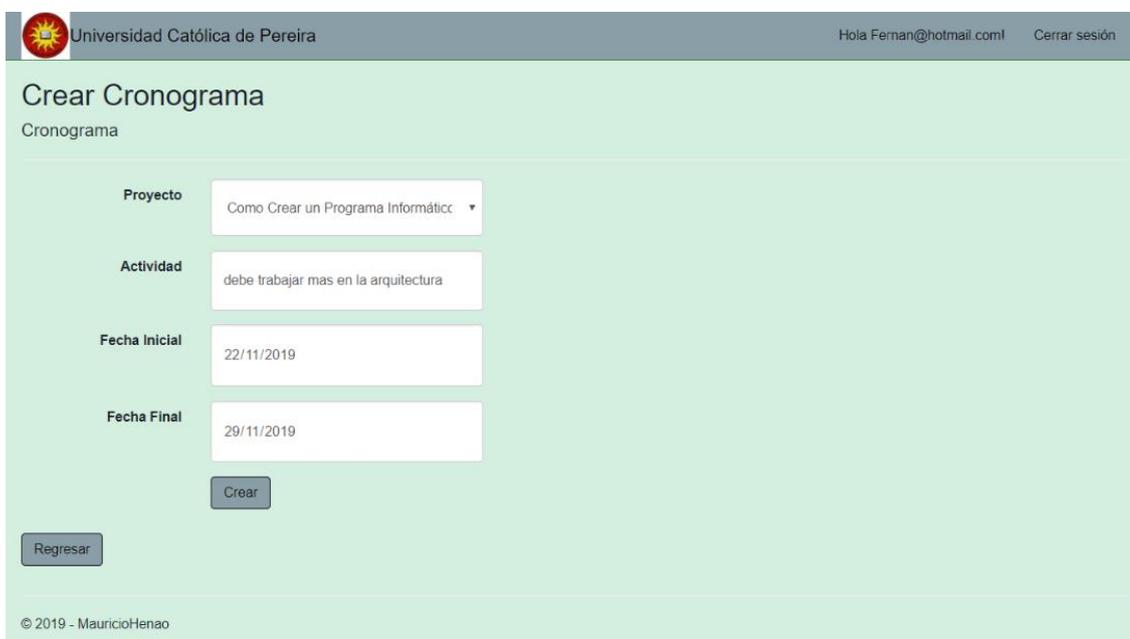


Figura 112: Manual del Usuario: Creación del Cronograma del Proyecto.
Fuente: Elaboración Propia.

En la Figura 113, se muestra cómo queda guardado el cronograma del proyecto. Ahí se verá reflejada la actividad que se requiere del proyecto, también desde esa parte del sistema se podrán gestionar todas las actividades requeridas.



Figura 113: Manual del Usuario: Creación Cronogramas del Proyecto.
Fuente: Elaboración Propia.

En la Figura 114, se muestra cómo se crearán las observaciones que hará el tutor al proyecto, de acuerdo al avance que se hacen en las actividades que tiene el estudiante sobre el proyecto, como se muestra en la Figura 115.

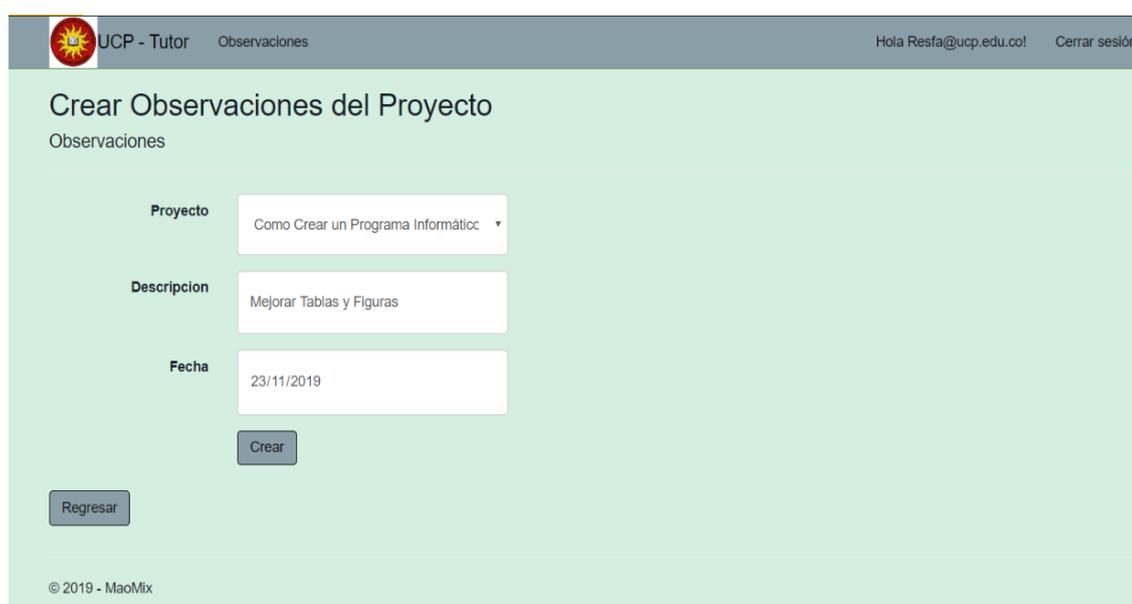


Figura 114: Manual del Usuario: Creación Observaciones del Proyecto.
Fuente: Elaboración Propia.

UCP - Tutor Observaciones Hola Resfa@ucp.edu.co! Cerrar sesión

Observaciones del Proyecto

Nombre_Proyecto	Descripcion	Date	
Como Crear un Programa Informático	Debe de mejorar la arquitectura	22/11/2019 12:00:00 a. m.	Editar Detalles Eliminar

[Regresar](#) [Crear Observacion](#)

© 2019 - MaoMix

Figura 115: Manual del Usuario: Creación Observaciones del Proyecto.
Fuente: Elaboración Propia.

En la Figura 116, se muestra cómo se dispone a ingresar en la vista de usuarios donde el administrador dará los roles (Permisos Administrativos) a cada una de las personas que se registran en el sistema, cada vez que alguien se registra en la vista usuarios se actualiza automáticamente.

UNIVERSIDAD CATOLICA DE PEREIRA Acercas de Contacto Hola carolina.lima@ucp.edu.co! Cerrar sesión

PERMISOS DEL SISTEMA

Nombre	Email	
3@gmail.com	3@gmail.com	Roles
Ricar@yahoo.com	Ricar@yahoo.com	Roles
mau@gmail.com	mau@gmail.com	Roles
2@gmail.com	2@gmail.com	Roles
Resfa@ucp.edu.co	Resfa@ucp.edu.co	Roles
Marc@ucp.edu.co	Marc@ucp.edu.co	Roles
Fernan@hotmail.com	Fernan@hotmail.com	Roles
carolina.lima@ucp.edu.co	carolina.lima@ucp.edu.co	Roles

[Regresar](#)

© 2019 - MauricioHenao

Figura 116: Manual del Usuario: Creación Permisos Administrativos.
Fuente: Elaboración Propia.

En la Figura 117, se muestra cómo al administrador del sistema se le concedieron los permisos requeridos para gestionar el sistema de una manera correcta.



Figura 117: Manual del Usuario: Creación Permisos Administrativos.
Fuente: Elaboración Propia.

En la Figura 118, se muestra cómo se crea un acta lo primero es buscar el nombre del proyecto, después de haber ingresado toda la información concerniente del proyecto, se dispone a crear el acta para ese proyecto con toda la información relevante del mismo cómo se muestra en la Figura 119.

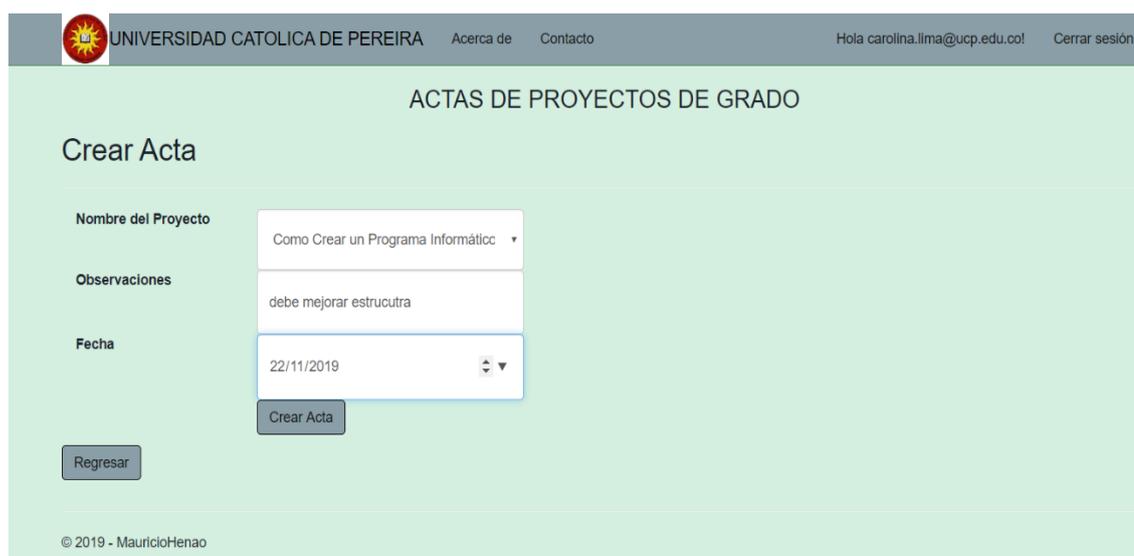


Figura 118: Manual del Usuario: Creación Actas del Proyecto.
Fuente: Elaboración Propia.

UNIVERSIDAD CATOLICA DE PEREIRA Acerca de Contacto [Hola carolina.lima@ucp.edu.co](mailto:Hola.carolina.lima@ucp.edu.co) Cerrar sesión

ACTA DE INICIACIÓN DEL PROYECTO

Fecha del Acta: 22/11/2019 12:17:34 a. m.

Participantes del Proyecto	Ricardo Alberto Goemz Jimenez		
Cedula	10231654		
Participantes del Proyecto	Fernando Leon Aparicio Gomez		
Cedula	10025287		
Sustentación	Acta Inicial ▼		
Nombre del Proyecto	Como Crear un Programa Informático		
Nombre del Tutor	Resfa Tulia Garcia Renfijo ▼		
Objetivos Generales	1. Como crear un programa desde cero		
Objetivos Especificos	1. Diseñar 2.. Implementar		
Modalidad del Proyecto	Científica		
Linea de Investigación	Operaciones Industriales y de servicios		
Cronograma de Actividades	Descripción de la actividad	Fecha inicial	Fecha Final
	Se debe de trabajar mas en la arquitectura	21/11/2019	30/11/2019
Observaciones	Observaciones	Fecha de la Observación	
	trabajar mas en el otjetivo general	29/11/2019	

- COMPROMISOS DEL (LOS) ESTUDIANTE (S):

- De acuerdo al artículo 21 del reglamento de los trabajos de grado.

a. Programar con el director de trabajo de grado el cronograma de las asesorías y cumplir con el mismo.
b. Informar de inmediato y por escrito, al Director del trabajo y a la dirección del programa respectivo, sobre cualquier cambio o asunto que dificulte y obstaculice el normal desarrollo del trabajo, para que ellos autorizan no las modificaciones correspondientes.
c. Acatar las recomendaciones del Director de trabajo de grado y entregar los avances acordados con el.
d. Entregar el informe final del trabajo de grado al Director del programa, dentro del cronograma establecido por la universidad.
e. Sustentar el informe final del trabajo de grado ante el jurado que designe la Dirección del programa y atender sus observaciones y recomendaciones.

- Otros compromisos acordados con el director de trabajo de grado.

f.
g.

Firma Estudiante: _____ Firma Estudiante: _____

Firma Tutor: _____ Firma Director de Programa: _____

© 2019 - MauricioHenao

Figura 119: Manual del Usuario: Creación de Actas del Proyecto.
Fuente: Elaboración Propia.