

**Análisis Y Diseño De Un Sistema De Información Para La Gestión De Trabajo De  
Grado En Las Modalidades de Tesis, Pasantía y Producción Por Línea de Investigación En  
El Instituto Tecnológico Del Putumayo**

**Kelium Zeus Antonio Cesar Jojoa Urrutia**

**Luis Carlos Jurado Ortega**

**Edinson José Meneses Díaz**

**Instituto Tecnológico del Putumayo**

**Facultad de Ingeniería**

**Tecnología en Desarrollo de Software**

**Mocoa, Junio de 2019**

**Análisis Y Diseño De Un Sistema De Información Para La Gestión De Trabajo De  
Grado En Las Modalidades de Tesis, Pasantía y Producción Por Línea de Investigación En  
El Instituto Tecnológico Del Putumayo**

**Kelium Zeus Julio Cesar Jojoa Urrutia**

**Luis Carlos Jurado Ortega**

**Edinson José Meneses Díaz**

**Informe final modalidad línea de investigación por semillero de investigación para optar  
el título de Tecnólogos en Desarrollo de Software**

**Ing. Adrián Izquierdo**

**Asesor**

**Instituto Tecnológico del Putumayo**

**Facultad de Ingeniería**

**Tecnología en Desarrollo de Software**

**Mocoa, Junio de 2019**

## Tabla de Contenido

Resumen.....	9
Abstrast.....	9
Introducción .....	10
1. Generalidades .....	10
1.1. Título.....	10
1.2. Problema.....	11
1.2.1. Descripción de problema .....	11
1.2.2. Formulación del Problema.....	12
1.3. Objetivos .....	12
1.3.1. Objetivo General.....	12
1.3.2. Objetivos específicos. ....	12
1.4. Justificación.....	12
2. Marco De Referencia .....	13
2.1. Marco Contextual .....	13
2.1.1. Generalidades.....	13
2.1.2. Contexto.....	14
2.2. Estado del arte .....	15
2.3. Marco Teórico .....	16
2.3.1. Software.....	16
2.3.2. Software de Desarrollo. ....	17
2.3.3. Desarrollo web. ....	17
2.3.4. Backend. ....	17
2.3.5. Java. ....	17
2.3.6. Bases de datos. ....	18

2.3.7.	Frontend.....	18
2.3.8.	Metodologías de desarrollo.....	20
2.4.	Marco Conceptual.....	23
2.5.	Marco Legal.....	25
2.5.1.	Normatividad Educativa Superior en Colombia .....	25
2.5.2.	Normatividad Institucional ITP .....	26
2.5.3.	Normatividad de derechos de autor y desarrollo de software.....	27
3.	Metodología.....	28
3.1.	Tipo de Investigación.....	28
3.2.	Línea de investigación .....	28
3.3.	Sublínea de investigación.....	28
3.4.	Tabla Metodológica .....	29
4.	Análisis De Resultados .....	31
4.1.	Análisis del Proceso Actual.....	31
4.2.	Requerimiento de los actores involucrados.....	35
4.2.1.	Análisis de requerimientos .....	36
4.2.2.	Diseño.....	39
4.3.	Prototipos de interfaz de usuario .....	100
4.3.1.	Interfaz para Estudiante .....	101
	Conclusiones .....	126
	Recomendaciones .....	126
	Bibliografía .....	127

## Índice de Tablas

Tabla 1.	Ventajas y desventajas de las arquitecturas.....	22
----------	--	----

Tabla 2. Tabla Metodológica .....	29
Tabla 3. Tabla de requerimiento de Stakeholders .....	36
Tabla 4 Tabla de tiempos para cada proceso de acuerdo al usuario.....	97

## Índice de Imágenes

Imagen 1. Fotografía Satelital Ubicación I.T.P. Sede Mocoa.....	14
Imagen 2. Arquitectura de software .....	98
Imagen 3. Interfaz ingreso información general de la propuesta .....	101
Imagen 4. Interfaz ingreso de los elementos de la propuesta.....	102
Imagen 5. Interfaz de ingreso de presupuesto de la propuesta.....	102
Imagen 6. Interfaz para ingresar los resultados esperados.....	103
Imagen 7. Interfaz para ingresar los impactos esperados .....	103
Imagen 8. Interfaz para ingresar las entidades financieras .....	104
Imagen 9. Interfaz para ingresar los cronogramas de actividades.....	104
Imagen 10. Interfaz para revisar la propuesta en desarrollo .....	105
Imagen 11- Interfaz para revisar solicitudes .....	106
Imagen 12. Interfaz para realizar una Nueva Solicitud .....	106
Imagen 13. Interfaz para la revisión de observaciones.....	107
Imagen 14. Interfaz de observaciones del jurado .....	108
Imagen 15 Interfaz para inscribir propuesta por pasantía.....	108
Imagen 16 Interfaz ingresar elementos de la propuesta de pasantía .....	109
Imagen 17 Interfaz para ingresar Cronograma.....	109
Imagen 18. Interfaz de asignación fecha de sustentación.....	110
Imagen 19. Interfaz para la asignar jurado a la propuesta .....	111
Imagen 20 Interfaz para agregar nueva empresa.....	111
Imagen 21 Interfaz Evaluar Propuesta.....	112
Imagen 22 Interfaz Evaluar Elementos de la propuesta .....	112
Imagen 23. Viabilizar Propuesta .....	113
Imagen 24 Instalación de Jhipster en Windows desde PowerShell(Administrador).....	114
Imagen 25 Características de un proyecto generado con Jhipster.....	115
Imagen 26 Arquitectura de una aplicación monolítica.....	116

Imagen 27 Estructura del proyecto para spring framework(backend) desde visual studio code	117
Imagen 28 Estructura del proyecto para angular(frontend)	118
Imagen 29 <b>Aplicación de spring framework</b>	118
Imagen 30 Implementación del servicio RESTful en la entidad PROYECTO, con jhipster	119
Imagen 31 Servicio de angular que se conecta al api RESTful por medio de la URL	120
Imagen 32 Diagrama del diseño de la base de datos en postgresql	121
Imagen 33 Diagrama de la base de datos con lenguaje JDL	121
Imagen 34 Crud de la tabla usuarios	122
Imagen 35 Crud de la tabla proyectos	122
Imagen 36 Crud de la tabla solicitudes	123
Imagen 37 Crud de la tabla presupuesto valor	123
Imagen 38 Vista para crear o editar una solicitud	124
Imagen 39 Vista para crear o editar el presupuesto	124
Imagen 40 Vista para crear o editar un usuario	125
Imagen 41 Vista para crear o editar un proyecto	125

## Índice de Diagramas

Diagrama 1. Diagrama de procesos tesis BPMN	33
Diagrama 2. Diagramas de proceso de semillero BPMN	34
Diagrama 3. Diagramas de proceso de grupo de investigación BPMN	35
Diagrama 4. Diagrama de dominio inicial	39
Diagrama 5. Caso de uso de estudiantes	40
Diagrama 6. Caso de uso inscripción propuesta de trabajo de grado	41
Diagrama 7. Diagrama de robustez Inscribir propuesta	43
Diagrama 8. Diagrama de secuencia inscribir propuesta	44
Diagrama 9. Caso de uso Ingresar Elementos	44
Diagrama 10. Diagrama de robutez ingresar elementos	46
Diagrama 11. Diagrama de secuencia ingresar elementos	47
Diagrama 12. Caso de uso ingresar resultados esperados	48
Diagrama 13. Diagrama de robustez ingresar resultados esperados	49

Diagrama 14. Diagrama de secuencia ingresar resultados esperados .....	50
Diagrama 15. Caso de uso ingresar impactos esperados .....	51
Diagrama 16.de robustez ingresar impactos esperados .....	52
Diagrama 17.de secuencia ingresar impactos esperados .....	53
Diagrama 18.Caso de uso Añadir entidades financiadoras.....	54
Diagrama 19.Diagrama de Robustez añadir entidades financiera.....	55
Diagrama 20.De secuencia añadir entidades financiera .....	56
Diagrama 21. Caso de uso subir documento de la propuesta.....	57
Diagrama 22. Diagrama de robustez subir documento a la propuesta .....	58
Diagrama 23. Diagrama de secuencia subir documento a la propuesta .....	59
Diagrama 24. Caso de uso para el actor asesor .....	60
Diagrama 25. Requerimientos del sistema CIECYT.....	60
Diagrama 26. Diagrama de Robustez realizar solicitud .....	61
Diagrama 27. Diagrama de secuencias realizar solicitud .....	63
Diagrama 28. Diagrama de observaciones de la propuesta .....	64
Diagrama 29. Diagrama de robustez realizar observaciones a al propuesta.....	65
Diagrama 30. Diagrama de secuencia realizar observaciones a la propuesta.....	66
Diagrama 31. Estado de la propuesta .....	67
Diagrama 32. Diagrama de robustez estado de la propuesta .....	68
Diagrama 33. Diagrama de secuencia estado de la propuesta .....	69
Diagrama 34. Caso de uso de presupuesto.....	70
Diagrama 35 Diagrama de robustez de ingresar presupuesto .....	71
Diagrama 36 diagrama de secuencia de ingresar presupuesto .....	72
Diagrama 37. Caso de Uso Evaluación de la propuesta por el jurado .....	73
Diagrama 38. Diagrama de Robustez evaluación de la propuesta .....	74
Diagrama 39. Diagrama de secuencia evaluación de la propuesta .....	75
Diagrama 40. Caso de uso solicitud de novedades .....	76
Diagrama 41. Diagrama de robustez solicitud de novedad.....	77
Diagrama 42. Diagrama de secuencia solicitud de novedad.....	78
Diagrama 43. Caso de uso viabilidad para sustentación.....	79
Diagrama 44. Diagrama de robustez viabilizar para sustentar.....	80

Diagrama 45. Diagrama de Secuencia viabilizar para sustentar .....	81
Diagrama 46. Casos de uso del administrador .....	82
Diagrama 47. Diagrama de robustez crear usuario .....	83
Diagrama 48. Diagrama de secuencia crear usuario .....	84
Diagrama 49. Caso de uso actualizar usuario .....	85
Diagrama 50. Diagrama de robustez actualizar usuario .....	86
Diagrama 51. Diagrama de secuencia .....	86
Diagrama 52. Caso de uso cambio de asesor .....	88
Diagrama 53. Diagrama de robustez .....	89
Diagrama 54. Diagrama de secuencia .....	90
Diagrama 55. Caso de uso estado de solicitud .....	91
Diagrama 56. Diagrama de robustez .....	92
Diagrama 57. Registrar datos empresa .....	93
Diagrama 58 .Caso de Uso Registrar Datos de Empresa.....	94
Diagrama 59. Diagrama de Robustez Registrar Datos de Empresa .....	95
Diagrama de secuencia 60 Registrar Datos Empresa .....	96
Diagrama 61 Caso de uso: definir tiempo de los procesos .....	97
Diagrama 62 Diseño de diagrama de clases para el caso de uso generar alertas .....	99

## Índice de Anexos

Anexo 1. Diccionario de Datos .....	131
Anexo 2 Diagrama de la Metodología ICONIX .....	144
Anexo 3 Ejemplo de un diagrama de modelo de dominio.....	144
Anexo 4 Ejemplo de diagrama de casos de uso .....	145
Anexo 5 Ejemplo de un diagrama de robustez correcto .....	145
Anexo 6 Elementos de un diagrama de secuencia .....	146
Anexo 7 Ejemplo de diagrama de clases relacionado con sub clases .....	146

## **Resumen**

En el presente trabajo se realizó el análisis, diseño y desarrollo de un prototipo de un sistema de información para la gestión de trabajos de grado en el Instituto Tecnológico del Putumayo, teniendo en cuenta los requerimientos de los diferentes usuarios que tendrá el sistema desde la parte administrativa de los mismos, así como quienes desarrollan y asesoran los diferentes trabajos de grado en la institución.

Para su desarrollo se utiliza la metodología semi ágil ICONIX, se realizó el levantamiento de requerimientos, creación de casos de uso, diagramas, modelado de procesos y diseño de prototipos a manera visual, además se desarrolló un prototipo con funcionalidades básicas como crear, guardar y visualizar información de trabajos de grado relacionado con los tiempos, requisitos, asignación de jurados y sustentaciones.

Esto permitirá al Instituto Tecnológico del Putumayo sede Mocoa obtener una solución acorde a la necesidad establecida por funcionarios del CIECYT, que aseguran que actualmente no cuenta con un Sistema de información para administrar los trabajos de grado en sus distintas modalidades y que esta situación les genera inconvenientes y retrasos.

## **Abstrast**

In the present work was carried out the analysis, design and development of a prototype of an information system for the management of degree works in the Technological Institute of Putumayo, taking into account the requirements of the different users that the system will have from the administrative, as well as those who develop and advise the different degree works in the institution.

For its development the semi-agile ICONIX methodology was used, the requirements were raised, use cases were created, diagrams, process modeling and prototypes design in a visual way, and a prototype was developed with basic functionalities such as creating, saving and View information about degree works related to times, requirements, jury assignments and support.

This will allow the Technological Institute of Putumayo Mocoa headquarters to obtain a solution according to the need established by CIECYT officials, who assure that currently it does not have an Information System to administer the degree works in its different modalities and that this situation generates inconveniences and delays.

## **Introducción**

La gestión de trabajos de grado es de gran importancia en las instituciones de educación superior, facilita porque de esta manera se realiza un seguimiento, control y evaluación de los distintos trabajos de grado, sin embargo, la organización y estructura de la información a largo plazo puede volverse un problema, por lo tanto, se construyen sistemas especializados o bases de datos centralizadas para una correcta administración de la información.

El Instituto Tecnológico del Putumayo (ITP) sede Mocoa aún no cuenta con un Sistema para la gestión de trabajos de grado, su Área de investigación aun realiza los procesos administrativos de forma manual, manejando papeleo de todo tipo de procesos y formatos como: la inscripción de propuesta, la evaluación de los jurados, la asignación de fechas de sustentación, el formato de correcciones entre otros que hacen parte del estatuto general.

En este trabajo de investigación se realizó el análisis, diseño y desarrollo de un prototipo de un sistema de información para la gestión de trabajos de grado aplicando la metodología de desarrollo ágil ICONIX, aunque el instituto cuenta con diversas modalidades de grado nos centramos en la Tesis para fomentar una mayor utilización de la misma como opción de grado para los estudiantes del ciclo profesional, no obstante el diseño puede ser extendido fácilmente para otros tipos de modalidades como la Línea de investigación por semilleros y grupos de investigación que fueron tenidos en cuenta al modelar sus procesos por medio de diagramas BPMN.

Para realizar análisis se utilizó diagramas BMPN para un fácil entendimiento de los procesos que lleva el ITP, en la fase de diseño nos centramos en los casos de uso y la realización de los diagramas que propone la metodología ICONIX es decir diagramas de robustez, secuencia y clases, junto con su diagrama de dominio que da una base inicial del diagrama de clases.

### **1. Generalidades**

#### **1.1. Título**

Análisis Y Diseño De Un Sistema De Información Para La Gestión De Trabajo De Grado En Las Modalidades de Tesis, Pasantía y Producción Por Línea de Investigación En El Instituto Tecnológico Del Putumayo.

## **1.2. Problema**

### **1.2.1. Descripción de problema**

Los sistemas de información han logrado sistematizar la mayor parte de los procesos que se lleva a cabo en las organizaciones, empresas o instituciones logrando así que puedan ser más eficientes y eficaces a ofrecer a sus usuarios la calidad del servicio, se hace necesario la búsqueda de soluciones que faciliten los procedimientos para que sean efectivos.

En el Instituto Tecnológico del Putumayo durante cada semestre se acepta la inscripción, asignación y desarrollo de números trabajos de grado, en las diferentes modalidades estatutarias, sin embargo, se ha venido presentado una dificultad en la gestión, seguimiento y control de estos puesto que los procesos se llevan de manera tradicional, y la información es manejada físicamente, lo que incrementa el tiempo en cada proceso y algunas veces se pierde la información y en otros casos no se pueden cumplir los tiempos establecidos debido a que la comunicación entre los diferentes actores se torna deficiente debido a que todo se hace a través de correo electrónico y algunas veces los buzones están llenos, no son revisados o simplemente falla la conectividad.

De igual manera, el manejo actual de los trabajos de grado obliga al estudiante a desplazarse continuamente para su control e información en el desarrollo de su trabajo, generando dificultades en la consulta de información, los tiempos de respuestas, deficiencia en el seguimiento del desarrollo de la propuesta.

El Centro de investigaciones y Extensión Científica y Tecnológica (CIECYT) en el Instituto Tecnológico del Putumayo se maneja la información a través de aplicativo de la suite de Microsoft “Excel” tradicionalmente para el registro, modificaciones, consultas entre otras actividades que tienen que ver con los diferentes procesos que se lleva a cabo en los trabajos de grado, perdiendo en algunos casos los historiales debido a que la información guardada solo es la básica de identificación del proyecto y del estudiante.

De acuerdo a lo expuesto en lo anterior se formula el problema de investigación “falta de un sistema de información que permita la gestión y control de trabajos de grado”.

### **1.2.2. Formulación del Problema**

¿Cómo un sistema de información permite mejorar el acompañamiento y control continuo del estudiante en el proceso de inscripción, desarrollo y evaluación de su trabajo de grado en el Instituto Tecnológico del Putumayo?

### **1.3. Objetivos**

#### **1.3.1. Objetivo General.**

Diseñar un sistema de información para la gestión de trabajos de grado en sus diferentes modalidades y proyectos de investigación en el Instituto Tecnológico del Putumayo que permita la administración, seguimiento y control oportuno de los mismos.

#### **1.3.2. Objetivos específicos.**

Analizar los procesos actuales para la gestión de trabajos de grado y proyectos de investigación que se llevan a cabo en el centro de investigación y extensión (CIECYT) en el Instituto Tecnológico del Putumayo

Establecer los requerimientos básicos para el sistema de información CYECYT para las modalidades de grado y proyectos de investigación en el I.T.P.

Diseñar y desarrollar un prototipo del sistema de información para para la gestión de trabajos de grado en CIECYT del I.T.P

### **1.4. Justificación**

En la formación tecnológica y profesional que ofrecen las instituciones de educación superior, se constituye como condición obligatoria la presentación de un trabajo de grado, que está sujeto a las diferentes modalidades que establezca cada una de ellas; el I.T.P. también cumple con esta normativa de la educación superior y ha asignado al CIECYT para el desarrollo y acompañamiento de esta tarea con los estudiantes que están culminando cada uno de los ciclos propedéuticos que se llevan a cabo.

El proceso que llevan a cabo los estudiantes y funcionarios de CIECYT para la inscripción, seguimiento, control, asignación de jurado y sustentaciones se desarrolla de manera manual en EXCEL, lo que algunas veces retrasa los tiempos establecidos y se genera pérdidas de información que obstaculizan el cumplimiento de cada una de las partes. De igual manera, la generación de

estadísticas reales en momentos requeridos, es poco confiable, dado la dispersión de la información actual.

Teniendo en cuenta que el I.T.P. ofrece, dentro de sus programas de formación profesional, el programa de Ingeniería de Sistemas articulado a la Tecnología en Desarrollo de Software y al evidenciar esta problemática que se presente semestre a semestre, como estudiantes de Desarrollo de software se plantea como aporte a la institución el análisis y desarrollo de un sistema de información que administre, gestione, controle y genere información estadística real cuando se requiera; lo cual contribuirá a la organización y consolidación de información y al cumplimiento de los tiempos y requisitos establecidos dentro de cada modalidad.

Para el desarrollo del sistema se utilizará la metodología iconix que brinda una guía de desarrollo ágil para el diseño del software que satisfaga los requerimientos de cada uno de los diferentes actores y les permita a las directivas obtener datos valiosos en todo momento para la toma de decisiones asertivas.

Con el diseño de este sistema se pretende consolidar la base para la creación del sistema de gestión de trabajo de grado porque le brinda las herramientas, el análisis documental, la lógica del negocio, diagramas y un acercamiento al software por medio de los prototipos, además un base preliminar del diseño de la base de datos, utilizando tecnologías actuales y que se encuentran posicionados fuertemente en mercado.

## **2. Marco De Referencia**

### **2.1. Marco Contextual**

#### **2.1.1. Generalidades**

Reseña histórica

El Instituto Tecnológico del Putumayo es una Institución de Educación Superior creada mediante Ley 65 de 1989 como Establecimiento Público, de carácter Académico del Orden Departamental, con Personería Jurídica, Autonomía Administrativa y Patrimonio Independiente, adscrito al Departamento del Putumayo (ITP, s/f). Surge en el territorio putumayense como respuesta a una necesidad académica de profesionalización, con capacidad de compromiso frente

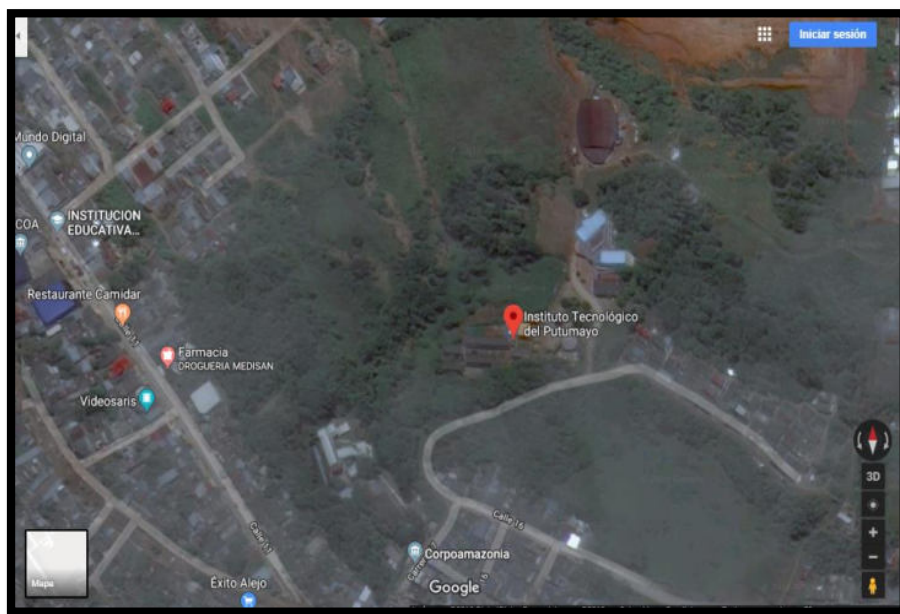
a las propuestas idóneas de desarrollo regional, en un proceso formativo para que se consoliden los discursos y responsabilidades en torno a la tarea histórica que debemos asumir.

Como se expone en la reseña histórica “El 10 de noviembre de 1994, el Gobierno Nacional mediante Decreto No. 2434 se nombró oficialmente como Rector del Instituto al Dr. FRANCISCO VELA HERRERA. En tales condiciones legales, se logra la histórica inauguración del nuevo ente educativo para el PUTUMAYO, el 27 de febrero de 1995 en Mocoa.” (ITP, s.f.).

### 2.1.2. Contexto

El Instituto Tecnológico del Putumayo se encuentra ubicado en el departamento del Putumayo, ciudad de Mocoa, barrio Luis Carlos Galán sede “aire libre”.

Imagen 1. Fotografía Satelital Ubicación I.T.P. Sede Mocoa



Fuente: google maps <https://www.google.com/maps/@1.1537154,-76.6529015,15z>

El Instituto Tecnológico del Putumayo, cuenta con 1810 estudiantes matriculados en el primer semestre del 2019, asistiendo a las sedes de Mocoa y sus extensiones en Valle del Guamuez y Puerto Asís; también tiene una subsede, la cual se encuentra ubicada en el municipio de Colon en el alto Putumayo donde se matricularon 457 estudiantes para los diferentes programas.

La investigación para el análisis, diseño y desarrollo de una versión preliminar de un software para la gestión de trabajos de grado se realizó en el municipio de Mocoa capital del departamento del Putumayo; específicamente en el Instituto Tecnológico del Putumayo (I.T.P).

El I.T.P cual cuenta con un Centro de investigaciones y Extensión Científica y Tecnológica (CIECYT) entre sus objetivos tenemos el siguiente “Generar una cultura investigativa que priorice la consolidación de los grupos de investigación existentes y la formación de otros nuevos, de carácter interdisciplinario y multidisciplinario, capaces de integrarse a redes nacionales e internacionales, en los distintos campos del saber.” (I.T.P, s.f.)

El CIECYT lleva una administración manual de los trabajos de grado es decir que la inscripción de propuestas, asesoramiento y corrección de trabajos de grado se hace por medio de papeleo.

## **2.2. Estado del arte**

En el Instituto Tecnológico del Putumayo se llevan a cabo los procesos de titulación enmarcados en las modalidades establecidas en el Estatuto Estudiantil y los acuerdos de Líneas y Sublíneas de Investigación establecidos por el CIECYT mediante aprobación de los Consejos Académicos y Directivo institucionales. Desde sus inicios y hasta las actuales promociones de tecnólogos egresados hasta la actualidad y egresados profesionales se han presentado cambios en las modalidades de trabajo de grado de acuerdo a los requerimientos institucionales y regionales, con el fin de mejorar los procesos investigativos y de formación para la vida profesional de cada uno de sus titulados.

A pesar de que el número de egresados cada semestre es relativamente alto en sus diferentes ciclos propedéuticos, el registro de trabajos de grado en CIECYT es bajo, dado que durante el año inmediatamente anterior, solo se tiene un registro de 25 trabajos de grado entre tesis y monografías en revisión y aceptación para sustentación y no hay información sobre el seguimiento y estadísticas de trabajos a partir de las otras modalidades, lo cual puede afectar para la toma de decisiones en este aspecto.

Según información emitida por la docente Valentina Ordoñez, hace unos años se desarrolló como tesis de grado del programa de ingeniería de sistemas un trabajo similar al presente, sin embargo, no se ha encontrado registro en Biblioteca ni en CIECYT que son las dependencias encargadas de llevar los registros de trabajos de grado en I.T.P.

Un sistema de información de trabajo de grado, ha permitido a instituciones como la Universidad Nacional de Colombia, tenga control y estadísticas reales en todo momento sobre los trabajos de grado que se han desarrollado y los que se encuentran en desarrollo, evitando así duplicidad en los mismo y exigiendo el cumplimiento de tiempos y requisitos exigidos dentro de su normatividad (Gonzales, 2011)., así mismo, su plataforma permite realizar las validaciones antiplagio y publicar los documentos desarrollados por estudiantes de las diferentes facultades.

Así mismo, universidades privadas como la Javeriana, tienen un sistema completo que permite realizar en línea desde la inscripción hasta el seguimiento, observaciones, solicitudes de ajustes hasta la aprobación de los diferentes trabajos de grado, lo cual le permite al estudiante estar en contacto con el desarrollo desde cualquier lugar del mundo y en todo momento, dado que se generan alertas cuando se han realizado cambios (Pontificia Universidad Javeriana, s/f).

## **2.3. Marco Teórico**

### **2.3.1. Software.**

Son el conjunto de programas de cómputo, procedimientos, reglas, documentación y datos asociados mediante procesos lógicos que hacen posible la realización de una tarea específica. A groso modo, es la parte intangible que hace posible el funcionamiento de los programas de un ordenador. (CISSET, 2019)

Dentro de los tipos de software existentes se utilizará

Software de sistemas que consiste en un conjunto de programas escritos para dar servicio a otros programas. (por ejemplo, compiladores, editores y herramientas para administrar archivos) se encarga del proceso de estructuras de información complejas pero deterministas. Otras aplicaciones de sistemas (por ejemplo, componentes de sistemas operativos, manejadores, software de redes, procesadores de telecomunicaciones) procesan sobre todo datos indeterminados. En cualquier caso, el área de software de sistemas se caracteriza por: gran interacción con el hardware de la computadora, uso intensivo por parte de usuarios múltiples, operación concurrente que requiere la secuenciación, recursos compartidos y administración de un proceso sofisticado, estructuras complejas de datos e interfaces externas múltiples. (Pressman, 2010)

### **2.3.2. Software de Desarrollo.**

El software de programación son aquellas aplicaciones y herramientas que utilizan los programadores para desarrollar nuevo software. Entre ellos se encuentran editores de texto, compiladores, intérpretes, enlazadores, depuradores y entornos de desarrollo integrados. Cada una de estas herramientas de programación puede ser utilizada con uno o más lenguajes de programación (Uriarte J. , 2017)

### **2.3.3. Desarrollo web.**

Desarrollo web significa construir y mantener sitios web; es el trabajo que tiene lugar en un segundo plano y que permite que una web tenga una apariencia impecable, un funcionamiento rápido y un buen desempeño para permitir la mejor experiencia de usuario. Los desarrolladores web son como duendes con poderes: nunca los ves, pero son los que hacen que todo esté bien y funcione de manera rápida y eficiente (Berenguer, 2014)

### **2.3.4. Backend.**

El back-end es el encargado del procesamiento de los datos de entrada desde el front-end, siendo este el controlador de la vista y realizando la conexión con la base de datos. (Nieto Rodrigo, 2016)

### **2.3.5. Java.**

Es un lenguaje fuertemente tipado y estricto. Por ejemplo, la declaración de las variables debe ser obligatoriamente explícita en java. Se verifica el código (sintaxis, tipos) en el momento de la compilación y también de la ejecución, lo que permite reducir los errores y los problemas de incompatibilidad de versiones

Además, java se encarga totalmente de la gestión de los punteros y el programador no tiene manera de acceder a ello, lo que evita la sobre-escritura accidental de datos en memoria y la manipulación de datos corruptos (JOSE MIGUEL ORDAX CASSÁ, PILAR ARANZAZU DIAZ-UFANO, SIN FECHA)

#### **2.3.5.1.Spring.**

Spring proporciona una serie de características, entre las que tenemos la inyección de dependencia, la gestión de transacciones, el soporte automatizadas y el soporte orientado a aspectos de programación

Spring framework es un software libre, desarrollado por la spring Sourcer. Se puede utilizar en contenedores web, dispensadores servidores de aplicaciones JEE como Glassfish y JBoss. También se puede utilizar para aplicaciones de escritorio (GONZALEZ, SIN FECHA)

### **2.3.6. Bases de datos.**

Una base de datos es un conjunto de datos almacenados sin redundancia innecesaria en un soporte informático y accesible simultáneamente por distintos usuarios y aplicaciones. Los datos deben estar estructurados y almacenados de forma totalmente independiente de las aplicaciones que la utilizan. (Yera, s.f.)

#### ***2.3.6.1. Sistemas gestores de bases de datos***

Un sistema gestor de base de datos es un software o conjunto de programas que permiten crear y mantener una base de datos. El SGBD actúa como interfaz entre los programas de aplicación (Usuario) y el sistema operativo. El objetivo principal de un SGBD es proporcionar un entorno eficiente a la hora de almacenar y recuperar la información de la base de datos (Yera, s.f.)

#### ***2.3.6.2. Postgresql***

Gestor de bases de datos orientadas a objetos (SGBDOO o ORDBMS en sus siglas en inglés) muy conocido y usado en entornos de software libre porque cumple los estándares SQL92 y SQL99, y también por el conjunto de funcionalidades avanzadas que soporta, lo que lo sitúa al mismo o a un mejor nivel que muchos SGBD comerciales.

### **2.3.7. Frontend**

el Frontend es la parte que se ejecuta en el navegador del usuario final, se encarga de mostrar la información de una manera atractiva y comunicarse con el Backend para la creación de datos y visualizarlos. (Azaustre, 2014)

#### ***2.3.7.1. JavaScript***

JavaScript es un lenguaje interpretado usado para múltiples propósitos, pero solo considerado como un complemento hasta ahora. Una de las innovaciones que ayudó a cambiar el modo en que vemos JavaScript fue el desarrollo de nuevos motores de interpretación, creados para acelerar el procesamiento de código. La clave de los motores más exitosos fue transformar el código JavaScript en código máquina para lograr velocidades de ejecución similares a aquellas

encontradas en aplicaciones de escritorio. Esta mejorada capacidad permitió superar viejas limitaciones de rendimiento y confirmar el lenguaje JavaScript como la mejor opción para la web. (Gauchat, 2012)

#### **2.3.7.2.HTML 5**

El Web tenía que ser distribuido: La información repartida en páginas no muy grandes enlazadas entre sí.

El Web tenía que ser hipertexto y debía ser fácil navegar por él.

Tenía que ser compatible con todo tipo de ordenadores (PCs, Macintosh, estaciones de trabajo...) y con todo tipo de sistemas operativos (Windows, MS-DOS, UNIX, MAC-OS).

Debía ser dinámico: el proceso de cambiar y actualizar la información debía ser ágil y rápido.

Estas características son las que marcaron el diseño de todos los elementos del WWW incluida la programación de páginas Web. Como respuesta a todos estos requisitos se creó el lenguaje HTML (HiperText Markup Language), cuyas siglas significan "lenguaje hipertexto de marcas". Este lenguaje será el encargado de convertir un inocente archivo de texto inicial en una página web con diferentes tipos y tamaños de letra, con imágenes impactantes, animaciones sorprendentes, formularios interactivos, etc. (Ferrer, García, & García)

#### **2.3.7.3.CSS**

CSS es un lenguaje que trabaja junto con HTML para proveer estilos visuales a los elementos del documento, como tamaño, color, fondo, bordes, etc... (Gauchat, 2012)

CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas. (Pérez, 2009)

#### **2.3.7.4.Angular**

Fue concebido con la idea de que la programación declarativa es mejor opción para construir interfaces de usuario que la programación imperativa, donde encaja mejor la lógica de negocio de la aplicación. Como resultado AngularJS aprovecha el HTML tradicional añadiendo nuevas etiquetas a su lenguaje. De hecho, Adam Abrons fue el encargado de bautizar al framework cómo

AngularJS por los diples (angle brackets en inglés) que componen los elementos de HTML. (Sánchez, 2018)

### **2.3.8. Metodologías de desarrollo**

Es un nuevo enfoque de desarrollo que según (Figueroa, Solís, & Cabrera) se basa en dos aspectos puntuales “el retrasar las decisiones y la planificación adaptativa; permitiendo potencia aún más el desarrollo de software a gran escala.”

Por ende, genera una nueva teoría que tiene como principales características:

- Los individuos y las interacciones entre ellos son más importantes que las herramientas y los procesos empleados.
- Es más importante crear un producto software que funcione que escribir documentación exhaustiva.
- La colaboración con el cliente debe prevalecer sobre la negociación de contratos.
- La capacidad de respuesta ante un cambio es más importante que el seguimiento estricto de un plan. (Figueroa, Solís, & Cabrera)

#### **2.3.8.1. Iconix**

ICONIX está entre la complejidad del RUP (Rational Unified Processes) y la simplicidad y pragmatismo del XP (Extreme Programming), sin eliminar las tareas de análisis y de diseño que XP no contempla. ICONIX es un proceso simplificado en comparación con otros procesos más tradicionales, que unifican un conjunto de métodos de orientación a objetos con el objetivo de abarcar todo el ciclo de vida de un proyecto. (Rosenberg & Scott, 1999)

La característica fundamental de ICONIX es Iterativo e incremental: Varias iteraciones ocurren entre el desarrollo del modelo del dominio y la identificación de los casos de uso. El modelo estático es incrementalmente refinado por los modelos dinámicos.

#### **2.3.8.2. Arquitectura Inicial**

Existe literatura que intenta realizar categorización de los tipos de arquitecturas posibles que pueden implementarse en la actualidad, se habla precisamente de este tema es: Software Architecture Patterns, Understanding Common Architecture Patterns and When to Use hem de Mark Richards. En este libro se establecen:

**Arquitectura por capas:** “Cada capa realiza una función, las capas más comunes son: presentación, lógica del negocio, y persistencia, aunque se pueden agregar más capas. Este tipo de arquitectura es muy común en aplicaciones web monolíticas y generalmente constituyen una solución a un sistema para una pequeña o mediana empresa que quiere sistematizar sus procesos en línea”. (Richards, 2015)

**Arquitectura dirigida por eventos:** “El patrón de arquitectura impulsado por eventos es un patrón popular de arquitectura asíncrono distribuido utilizado para producir aplicaciones altamente escalables. También es altamente adaptable y se puede usar para aplicaciones pequeñas y grandes y complejas. La arquitectura dirigida por eventos consta de componentes de procesamiento de eventos de propósito único altamente desacoplados que reciben y procesan eventos de forma asíncrona”. (Richards, 2015)

**Arquitectura microkernel:** “El patrón de arquitectura de microkernel (a veces denominado patrón de arquitectura de plug-in) es un patrón natural para implementar aplicaciones basadas en productos. Una aplicación basada en productos es una que está empaquetada y disponible para descargar en versiones como un producto típico de terceros. Sin embargo, muchas compañías también desarrollan y lanzan sus aplicaciones comerciales internas como productos de software, completas con versiones, notas de lanzamiento y características conectables. Estos también son un ajuste natural para este patrón. El patrón de la arquitectura del microkernel le permite agregar características adicionales de la aplicación como complementos a la aplicación central, proporcionando extensibilidad así como también separación y aislamiento de las características”. (Richards, 2015)

**Patrón de Arquitectura de Microservicios:** “...cada componente de la arquitectura de microservicios se implementa como una unidad separada, lo que permite una implementación más sencilla a través de un sistema de entrega eficaz y optimizado, una mayor escalabilidad y un alto grado de desacoplamiento de la aplicación y los componentes dentro de su aplicación. Quizás el concepto más importante a entender con este patrón es la noción de un componente de servicio. En lugar de pensar en servicios dentro de una arquitectura de microservicios, es mejor pensar en los componentes del servicio, que pueden variar en la granularidad de un solo módulo a una gran

parte de la aplicación. Los componentes del servicio contienen uno o más módulos (por ejemplo, clases de Java) que representan una función de un solo propósito (por ejemplo, proporcionar el clima para una ciudad o pueblo específico) o una parte independiente de una aplicación comercial grande (por ejemplo, la colocación de acciones comerciales o la determinación de las tasas de seguro de automóviles). Diseñar el nivel correcto de granularidad de componentes de servicio es uno de los mayores desafíos dentro de la arquitectura de microservicios”. (Richards, 2015)

**Arquitectura basada en el espacio:** “El patrón de arquitectura basada en el espacio está diseñado específicamente para abordar y resolver problemas de escalabilidad y concurrencia. También es un patrón de arquitectura útil para aplicaciones que tienen volúmenes de usuarios concurrentes variables e impredecibles. Solucionar arquitectónicamente el problema extremo y variable de la escalabilidad es a menudo un enfoque mejor que intentar escalar una base de datos o actualizar las tecnologías de almacenamiento en caché en una arquitectura no escalable”. (Richards, 2015)

De acuerdo a Richard, 2015, en la tabla 1, se muestran las ventajas (flecha verde arriba) y desventajas (flecha roja arriba) de cada una de las arquitecturas

Tabla 1. Ventajas y desventajas de las arquitecturas

	Layered	Event-driven	Microkernel	Microservices	Space-based
Overall Agility	↓	↑	↑	↑	↑
Deployment	↓	↑	↑	↑	↑
Testability	↑	↓	↑	↑	↓
Performance	↓	↑	↑	↓	↑
Scalability	↓	↑	↓	↑	↑
Development	↑	↓	↓	↑	↓

Fuente: (Richards, 2015)

## 2.4. Marco Conceptual

**Arquitectura Del Sistema:** Arquitectura de sistemas: la estructura del sistema fundamental y de unificación definida en términos de comportamientos, restricciones, procesos, interfaces y elementos del sistema (INCOSE, 1996)

**Caso de Uso:** Los casos de uso le brindan una forma estructurada de capturar los requisitos de comportamiento de un sistema, de modo que usted pueda crear razonablemente un diseño a partir de ellos.

Te ayudan a responder algunas preguntas fundamentales:

¿Qué intentan hacer los usuarios del sistema? ¿Cuál es la experiencia del usuario? Una cantidad sorprendente de lo que hace su software está dictada por la forma en que los usuarios deben interactuar con él.

Los casos de uso le ofrecen algo para lo que puede diseñar, y desde el cual puede estimar de manera confiable el tiempo y el esfuerzo.

Este libro le enseña cómo escribir casos de uso como el primer paso para hacer un buen diseño orientado a objetos y como un medio para ayudarlo a llegar rápidamente a un nivel alto. Código de calidad. (Scott & Rosenberg, 1999)

**Modelo de dominio:** El modelo de dominio es un artefacto colaborativo en vivo. Se refina y actualiza a lo largo del proyecto, de modo que siempre refleja la comprensión actual del espacio del problema.

El modelado de dominios es la tarea de construir un glosario de proyecto o un diccionario de términos utilizados en su proyecto. El modelo de dominio para un proyecto define el alcance y constituye la base sobre la cual construir sus casos de uso.

Como se acaba de mencionar, un modelo de dominio es, esencialmente, un glosario de proyecto: un diccionario "en vivo" de todos los términos utilizados en su proyecto. (Scott & Rosenberg, 1999)

**Diagrama de robustez:** Un diagrama de robustez es una imagen de objeto de un caso de uso. El diagrama de robustez y el texto del caso de uso deben coincidir con precisión, por lo que el

diagrama de robustez obliga a vincular el texto del caso de uso a los objetos. Esto le permite impulsar diseños orientados a objetos desde casos de uso, y esto es realmente la "magia" del análisis de robustez. (Scott & Rosenberg, 1999)

**Diagrama de Secuencia:** El diseño detallado es, por el contrario, el comportamiento de asignación (también conocido como asignación de comportamiento), es decir, asignar las funciones de software que ha identificado en el conjunto de Clases que descubriste durante el diseño preliminar.

Cuando dibuja diagramas de secuencia, está realizando otro barrido a través del diseño preliminar, agregando en detalle.

Al dibujar diagramas de secuencia, estás explorando meticulosamente los detalles del diseño detallado para cada caso de uso, en detalle microscópico. Esto significa explorar no solo el curso básico, sino también todos los cursos alternativos de acción en cada caso de uso. (Scott & Rosenberg, 1999)

**Diagrama de clases:** Para modelar clases, incluidos sus atributos, operaciones, relaciones y asociaciones con otras clases, el UML proporciona un diagrama de clase, que aporta una visión estática o de estructura de un sistema, sin mostrar la naturaleza dinámica de las comunicaciones entre los objetos de las clases.

Los diagramas de clase también pueden mostrar relaciones entre clases. Una clase que sea una subclase de otra clase se conecta con ella mediante una flecha con una línea sólida y con una punta triangular hueca. (Roger S. Pressman)

**CIECYT:** Centro de Investigaciones, Extensión Científica y Tecnológica del Instituto Tecnológico del Putumayo.

**Diccionario De Datos:** El diccionario de datos está formado por un conjunto de metadatos, las cuales tienen diferentes características que son lógicas y puntuales de datos. Estos nos describen el funcionamiento de estas características, las cuales deben incluir nombre, descripción, alias del contenido y la organización. El diccionario identifica los procesos donde son utilizados los datos y los sitios en donde se necesita el acceso rápido a la información que se requiera. Este es un complemento del flujo de datos que se encuentre en algún sistema, ya que en el encontramos un

listado de todos los elementos que integran esa parte. Como ya sabemos el flujo de datos está compuesto por dos elementos principales: almacén de datos y procesos, en donde el diccionario de datos guarda los complementos y describe todos estos elementos. (Red Tercer Milenio S.C, 2012)

**Modelo De Negocio:** un modelo de proceso de negocio describe cómo funciona el negocio, es decir, describe las actividades involucradas en el negocio y la manera en que se relacionan unas con otras e interactúan con los recursos necesarios para lograr la meta del proceso. (Ruiz, García, & Piattini)

**Requerimientos:** Son la base fundamental de cualquier software como se especifica en el estándar “Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo” (IEEE, 1990)

**Testing:** es toda una disciplina en la ingeniería de software permite tener procesos, métodos de trabajo y herramientas para identificar defectos en el software alcanzando un proceso de estabilidad del mismo. El Testing no es una actividad que se piensa al final del desarrollo del software, va paralelo a este. Permite que lo que se está construyendo, se realice de manera correcta de acuerdo a lo que necesita un usuario final. (Hoyos, s/f)

## **2.5. Marco Legal**

Constitución Política de Colombia. Artículo No 67 Derecho a la educación “de la persona y un servicio público que tiene una función social; con ella se busca el acceso al conocimiento, a la ciencia, a la técnica, y a los demás bienes y valores de la cultura”. (Asamblea Nacional Constituyente, 1991).

### **2.5.1. Normatividad Educativa Superior en Colombia**

**Ley 30 de 1992** que define el carácter y autonomía de las Instituciones de Educación Superior -IES-, el objeto de los programas académicos y los procedimientos de fomento, inspección y vigilancia de la enseñanza (Congreso de la República, 1992).

**El Decreto 2566 de 2003** reglamentó las condiciones de calidad y demás requisitos para el ofrecimiento y desarrollo de programas académicos de educación superior, norma que fue derogada con la Ley 1188 de 2008 que estableció de forma obligatoria las condiciones de calidad

para obtener el registro calificado de un programa académico, para lo cual las Instituciones de Educación Superior, además de demostrar el cumplimiento de condiciones de calidad de los programas, deben demostrar ciertas condiciones de calidad de carácter institucional (Mineducación, 2003).

**Decreto No. 1075 del 26 de mayo de 2015:** El objetivo de este Decreto, es compilar y racionalizar las normas de carácter reglamentario que rigen en el Sector Educación; así como otorgar un instrumento jurídico único, para el sector. (FODESEP, 2016)

### **2.5.2. Normatividad Institucional ITP**

**Estatuto general:** Artículo 53: Es estudiante de la Institución la persona que posea matrícula vigente para un programa y periodo académico (ITP, s/f).

**Reglamento Estudiantil:** Artículo 54. La Institución tendrá un Reglamento Estudiantil, expedido mediante acuerdo del Consejo Directivo a propuesta del Consejo Académico, que regule entre otros los siguientes aspectos: Requisitos de inscripción, admisión y matrícula, (ITP, s/f)

**PEI:** les otorga identidad y sentido a los procesos institucionales, delineando y conformando comunidades pedagógicas, académicas y científico – tecnológicas (ITP, s/f).

#### **Estatuto estudiantil**

**Acuerdo No.06** agosto 04 de 2017. Artículo 111. Definición. El Trabajo de grado, es una actividad dirigida de investigación y desarrollo sobre un tema específico, estrechamente relacionado con el campo de formación del estudiante, de importancia, actualidad y utilidad social.

Producción Académica Por Líneas De Investigación y Semilleros De Investigación. (INSTITUTO TECNOLÓGICO DEL PUTUMAYO, 2010)

**Estatuto Docente:** Mejorar el nivel académico a través del ejercicio de la investigación, la docencia y la extensión.

**Acuerdo 014 de 2007.** “Por el cual se adoptan los Programas, Líneas y Sublíneas de Investigación institucionales en el Instituto Tecnológico del Putumayo” (ITP, 2007).

### **2.5.3. Normatividad de derechos de autor y desarrollo de software**

Para el desarrollo existe normatividad de derechos de autor a nivel internacional y nacional para proteger los derechos de autor.

**Ley número 23 de 1982.** Artículo 2: Los derechos de autor recaen sobre las obras científicas, literarias y artísticas las cuales se comprenden todas las creaciones del espíritu en el campo científico, literario y artístico, cualquiera que sea el modo o forma de expresión y cualquiera que sea su destinación (...), inclusive los videogramas; (...) (Congreso de la Republica, 1982)

**Decreto número 1360 de 1989** (junio 23): Artículo 1: de conforme con lo previsto en la ley 23 de 1982 sobre derechos de autor, el soporte lógico (software) se considera una creación propia del dominio literario.

Artículo 2: el soporte lógico (software) comprende uno o varios de los siguientes elementos: el programa de computadora, la descripción de programas y el material auxiliar (Anonimo, 1989)

#### **Régimen Común Sobre Derecho De Autor Y Derechos Conexos**

**Decisión 351.** Artículo 2.- Cada País Miembro concederá a los nacionales de otro país, una protección no menos favorable que la reconocida a sus propios nacionales en materia de Derecho de Autor y Derechos Conexos.

Artículo 3.- A los efectos de esta Decisión se entiende por: Programa de ordenador (Software): Expresión de un conjunto de instrucciones mediante palabras, códigos, planes o en cualquier otra forma que, al ser incorporadas en un dispositivo de lectura automatizada, es capaz de hacer que un ordenador -un aparato electrónico o similar capaz de elaborar informaciones-, ejecute determinada tarea u obtenga determinado resultado. El programa de ordenador comprende también la documentación técnica y los manuales de uso.

Artículo 4.- La protección reconocida por la presente Decisión recae sobre todas las obras literarias, artísticas y científicas que puedan reproducirse o divulgarse por cualquier forma o medio conocido o por conocer, y que incluye, entre otras, las siguientes:

- 1) Los programas de ordenador;

II) Las antologías o compilaciones de obras diversas y las bases de datos, que por la selección o disposición de las materias constituyan creaciones personales (Anonimo, 1993)

### **3. Metodología**

#### **3.1. Tipo de Investigación**

La investigación se desarrolla a partir del tipo de investigación mixta partiendo de la investigación Descriptiva (Universia, 2017) que se utiliza para describir la realidad de situaciones, eventos, personas, grupos o comunidades que se estén abordando y que se pretenda analizar; permitiendo describir los procesos que se llevan a cabo para los trabajos de grado partiendo de un análisis de documentos y entrevista directa y la investigación exploratoria como menciona (Universia, 2017) que ofrece un primer acercamiento al problema que se pretende estudiar y conocer; permitiendo identificar y conocer los diferentes procesos que se lleva a cabo en el Instituto Tecnológico del Putumayo referente al trabajo de grado, y los actores que intervienen en el proceso, el manejo de la información, el estudio de las diferentes modalidades, el proceso de inscripción, desarrollo y evaluación de los trabajos de grado.

#### **3.2. Línea de investigación**

Sistemas de Información (ITP, 2007).

#### **3.3. Sublínea de investigación**

Diseño asistido por computador (ITP, 2007).

### 3.4. Tabla Metodológica

Tabla 2. Tabla Metodológica

OBJETIVO GENERAL	OBJETIVOS	ACTIVIDADES	HERRAMIENTAS	RESULTADOS ESPERADOS
Diseñar un sistema de información para la gestión de trabajos de grado en sus diferentes modalidades y proyectos de investigación en el Instituto Tecnológico del Putumayo que permita la administración, seguimiento y control oportuno de los mismos.	Analizar los procesos actuales para la gestión de trabajos de grado que se llevan a cabo dentro del centro de investigación y extensión (CIECYT) en el Instituto Tecnológico del Putumayo	Realizar una reunión con los funcionarios encargados del manejo de Trabajos de grado en CIECYT ITP	Entrevista directa	Comprender el manejo que se da actualmente a los procesos en la gestión y administración de los trabajos de grado en CIECYT ITP
		Analizar el documento P-INV-001 modalidades de trabajo de grado	Observación Directa – Análisis Documental	Comprender las modalidades de grado
		Analizar los procesos actuales que se desarrollan por parte de los funcionarios de CIECYT	Análisis de procesos	Comprender de manera más realista los requisitos del sistema
		Hacer diagramas BPMN	Diagramación a través de Visual paradigm Enterprise	Comprender e identificar las actividades que ocurren durante cada uno de los procesos
	Establecer los requerimientos básicos para el sistema de información CYECYT para las modalidades de grado dentro del ITP	Realizar una reunión con los Stakeholders CIECYT ITP	Grupo Focal Entrevistas directas	Comprender los requerimientos de cada uno de los actores involucrados para establecerlos en el programa a desarrollar.
		Levantamiento de requerimientos	Historias de usuario	Establecer los requerimientos para el sistema
		Elaborar diagramas	Enterprise Architect	Modelar los procesos y funcionamiento del sistema tomando en cuenta los requerimientos
		Diseñar prototipos	Pencil, Adobe XD version gratuita	Tener una perspectiva o idea de lo que se espera obtener en el diseño del sistema.
		Socializar prototipos	Exposición	Consulta y manejo de expectativa frente al diseño del sistema

				participación activa y resolver dudas
	Diseñar y desarrollar un prototipo del sistema de información para la gestión de trabajos de grado en CIECYT del ITP	Crear una base de datos	Postgresql, pg admin 4, Jhipster	Diseño y desarrollo de base de datos para administrar trabajos de grado.
		Elaborar el FRONTEND	Angular 6, TipeScript	Diseño del prototipo de la interfaz gráfica del software para administrar trabajos de grado
		Elaborar el BACKEND	Spring boot 2, java	Implementación de la lógica del lado del servidor

Fuente: Los autores

## **4. Análisis De Resultados**

### **4.1. Análisis del Proceso Actual**

Inicialmente, se hizo necesario un estudio a través de una reunión con los diferentes interesados como son: administradores del centro de investigación y extensión científica y tecnológica, estudiantes del Instituto Tecnológico del Putumayo y docentes que participan en la evaluación de propuestas para las diferentes modalidades, también se hace un análisis del documento modalidad de trabajos de grado el cual describe cada uno de los procesos que se deben llevar en las diferentes modalidades de grado que están vigentes en el instituto tecnológico del putumayo; de esta forma se visualiza la necesidad del software y se delimita los procesos o modalidades que se incluirán en el proyecto

Como resultado de las actividades anteriormente mencionadas se establece los procesos que se llevan a cabo para dentro del proceso para las modalidades tesis, pasantía y línea de investigación.

#### Descripción del proceso para tesis

Para la inscripción de una propuesta de trabajo de grado el estudiante debe entregar un oficio firmado por el asesor y oficio de entrega de la propuesta del trabajo de grado en CIECYT ; después de que el estudiante presente la propuesta en medio físico y magnético. el comité de investigación y extensión asigna un jurado, el cual es el encargado de hacer la evaluación y decidir si la propuesta es viable o no es viable. En caso de ser devuelta con observaciones, los estudiantes deberán hacer las respectivas correcciones y enviarlo nuevamente al Comité de Investigaciones y Extensión, en un plazo máximo de diez (10) días hábiles, este a su vez lo remitirá nuevamente a los jurados lectores para su respectiva evaluación, la cual no debe pasar de cinco (5) días hábiles.

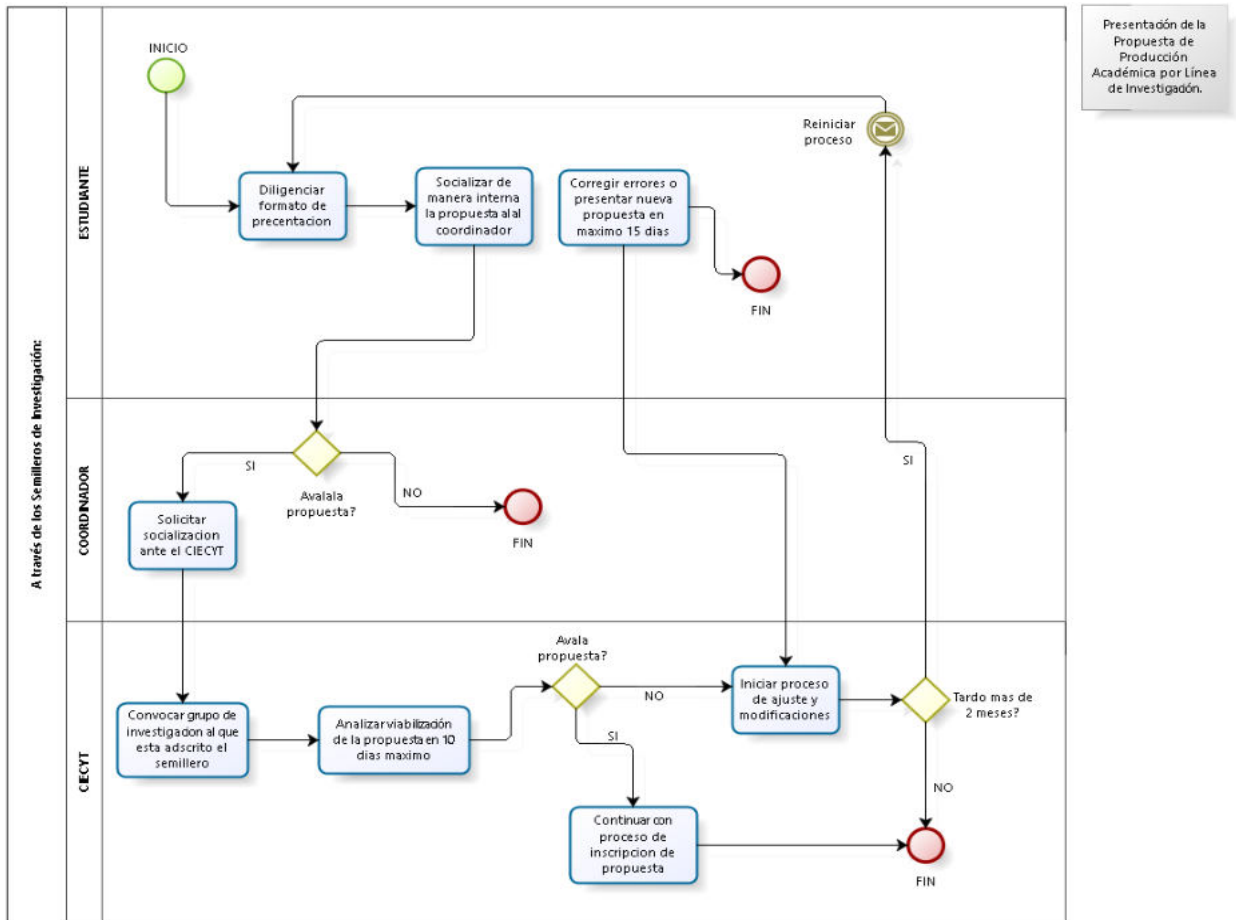
Si la evaluación por parte del jurado es positiva, los estudiantes darán inicio a la investigación, en caso contrario deberán realizar las correcciones o ajustes hasta que el jurado evaluador lo apruebe. Una vez aprobada la propuesta tendrán una duración mínima de seis (6) meses, la propuesta aprobada no podrá ser modificada en aspectos de fondo (objetivos, metodología, cronograma, resultados esperados). En caso de ser necesarios, el asesor y los estudiantes presentaran un oficio con la solicitud de novedades ante el Comité de Investigaciones y Extensión, quien analizará el caso y aprobará o negará la misma

Durante el proceso de investigación el estudiante puede hacer solicitud de novedades ya sea para renuncia, cambio de asesor o para realizar cambios en la propuesta inicial.

Para la entrega del informe final el estudiante debe entregar el documento junto con un oficio dirigido al CIECYT quien a su vez envía el documento al jurado quien tendrá quince (15) días hábiles para su revisión; luego de la revisión el jurado puede pedir correcciones si es necesario o aprobar para sustentación si ya está lista

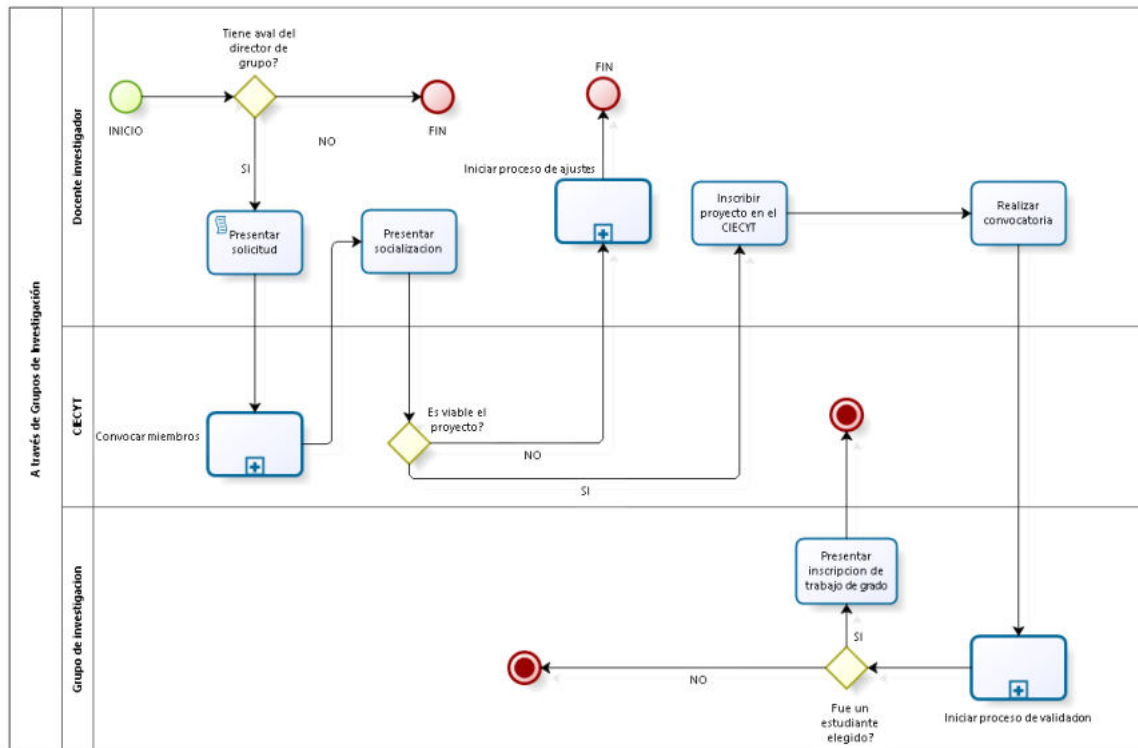


Diagrama 2. Diagramas de proceso de semillero BPMN



Fuente: Los autores

Diagrama 3. Diagramas de proceso de grupo de investigación BPMN



Powered by  
bizagi  
Modeler

Fuente: Los autores

#### 4.2. Requerimiento de los actores involucrados

El levantamiento de requerimientos se realizó en base a una reunión con los Stakeholders (interesados), donde a partir de la herramienta de trabajo grupal “lluvia de ideas” se capturaron algunas características funcionales del sistema, sin embargo, la mayor cantidad de requerimientos se obtuvieron del documento P-INV-001 MODALIDAD DE TRABAJOS DE GRADO donde se describen todos los procedimientos de las distintas modalidades de grado desde la inscripción hasta la sustentación

Tabla 3. Tabla de requerimiento de Stakeholders

<b>STAKEHOLDERS</b>	<b>REQUERIMIENTOS</b>
<b>Estudiantes</b>	Realizar la inscripción de la propuesta en línea
	Visualizar el estado de la propuesta
	Subir versiones del documento del proyecto
	Visualizar las correcciones del asesor asignado
	Visualizar las correcciones del jurado asignado
	Visualizar una propuesta creada
	Realizar una solicitud
<b>Asesor</b>	Observar la respuesta de una solicitud
	Mirar las propuestas que tiene asignadas
	Realizar una solicitud
	Hacer revisiones a una propuesta
	Filtrar propuestas por fase
<b>Jurado</b>	Observar la respuesta de una solicitud
	Mirar las propuestas que tiene asignadas
	Filtrar propuestas por fase
	Seleccionar una propuesta para su evaluación
	Determinar el grado de cumplimiento de los elementos de una propuesta
	Realizar una observación a un elemento de una propuesta seleccionada, en base a una pregunta
	Evaluar el informe final(propuesta en fase final)
	Asignar una puntuación a cada elemento del informe final
Explicar una puntuación en de un elemento de una propuesta seleccionada, en base a una pregunta	
<b>Administrador</b>	Llenar un formato de ficha técnica
	Crear usuarios para el sistema
	Asignar una fecha de sustentación a una propuesta
	Asignar un jurado a una propuesta
	Mirar la ficha técnica de un jurado
	Asignar un asesor a una propuesta
	Mirar todas las propuestas de la base de datos
Filtrar propuestas por fase	

Fuente: Los autores

#### 4.2.1. Análisis de requerimientos

Teniendo en cuenta el tamaño del sistema a continuación se lista lo que podría ser los requisitos y tiempos necesarios para las principales funcionalidades del sistema de información.

## Módulos del sistema de información

Sistema	N°	Detalle del sistema	Semanas
CIECYT	1	implementación de la base de datos	* * *
	2	creación del control de usuarios	*
	3	Creación de interfaz para la inscripción de propuestas	*
	4	interfaz para guardar versiones de informes	*
	5	Creación de interfaz para las preguntas y respuestas (corrección y evaluación)	* *
	6	Interfaz para la asignación de jurados	*
	7	Interfaz para la creación de solicitudes	*
	8	interfaz para la asignación de fechas	*
	9	interfaz para las observaciones del asesor	*
	10	interfaz para la búsqueda de trabajos de grado	*
	11	Crear los triggers, vistas, funciones y procedimientos necesarios	**
	12	Crear servicios web para las funcionalidades principales	***
	13	Crear un sistema de alertas para el control de tiempos	***
TOTAL			21

Fuente: Los autores

### 4.2.1.1. Posibles iteraciones para el desarrollo del sistema

Las iteraciones dentro del desarrollo son muy importantes porque nos ayudan a descubrir errores que al principio puede ser difícil notar, además nos permite controlar mejor nuestro tiempo para desarrollar los módulos del sistema. Cabe aclarar que todas estas iteraciones serían a modo de ejemplo porque no se pretende desarrollar el sistema en su totalidad.

La cantidad total de tiempo que se considera emplear sería de 21 semanas para el desarrollo de los principales módulos del proyecto, sin tener en cuenta el conocimiento que tiene cada desarrollador sobre las tecnologías planteadas en el proyecto tales como: angular, spring y desarrollo web en general.

### Iteración 1

Se podría considerar la implementación de la base de datos y la creación de un control de usuario para tener en cuenta la seguridad que propone la arquitectura RESTful, además de esta

manera se podría crear usuarios con sus respectivos roles. Consideramos que tardaríamos 3 semanas en implementar este módulo.

### **Iteración 2**

Esta iteración puede durar una semana y la utilizaremos para crear una de las funcionalidades más importantes del sistema, es decir la inscripción de propuestas y el almacenamiento de los informes en sus distintas fases.

### **Iteración 3**

Continuando con el flujo del proceso este tardará solo dos semanas, aquí se creará una interfaz para la asignación de jurados, además de una funcionalidad muy útil la cual es la evaluación de la propuesta por parte de un jurado.

### **Iteración 4**

Para este módulo se realizará la interfaz para el envío y recepción de solicitudes, además se creará una interfaz para la asignación de fechas, este es un módulo muy importante para el administrador del ciecyt, pensamos que podríamos tardar dos semanas en implementarlo.

### **Iteración 5**

Aquí pretendemos crear el módulo del asesor para que este pueda realizar observaciones a los elementos e informes de una propuesta. También se adaptarán algoritmos de búsqueda para realizar búsquedas eficientes de los trabajos de grado. El tiempo para realizar este módulo sería de 2 semanas.

### **Iteración 6**

En esta iteración se crearían los Triggers, vistas, funciones y procedimientos necesarios en la base de datos todo ello para enfatizar en la eficiencia de las consultas o peticiones, además se implementarán servicios web basados en la arquitectura RESTful. Pensamos que tardaremos 5 semanas en desarrollar este módulo.

### **Iteración 7**

Este módulo es muy importante para ayudar al usuario a controlar mejor su tiempo, se pretende desarrollar un sistema de alertas para notificarle al usuario el tiempo que le queda para realizar

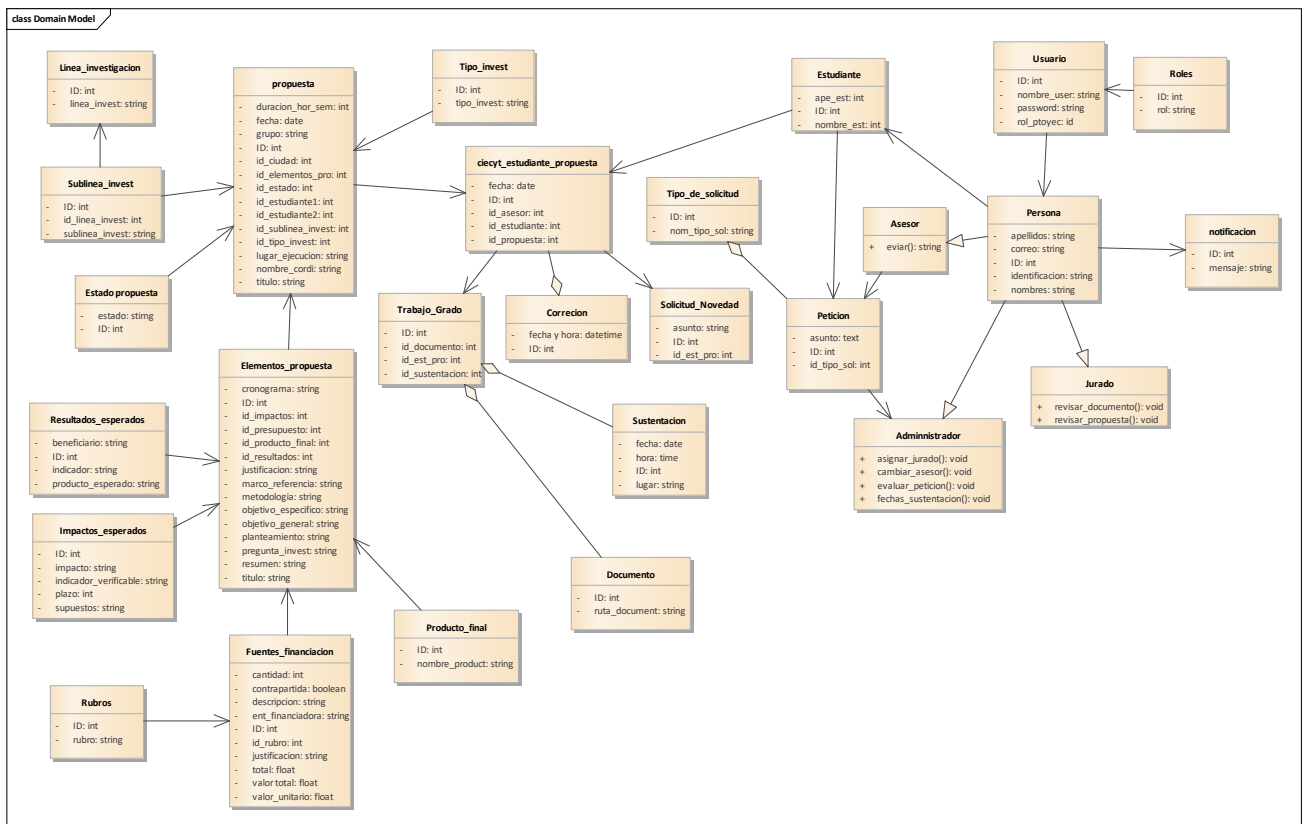
cierta acción como: realizar una corrección, revisar una propuesta, asignar una fecha entre otras, se evaluara y valorara las distintas tecnologías para realizar peticiones al servidor de manera eficiente. Creemos que tardaríamos 3 semanas en programar este sistema de alertas.

## 4.2.2. Diseño.

### 4.2.2.1. Modelo de dominio.

En este modelo se plantea las diferentes clases del dominio u objetos potenciales que harán parte del sistema de gestión de trabajo de grado del ciecyt tomando como referencia su contexto

Diagrama 4. Diagrama de dominio inicial



Fuente: Los autores

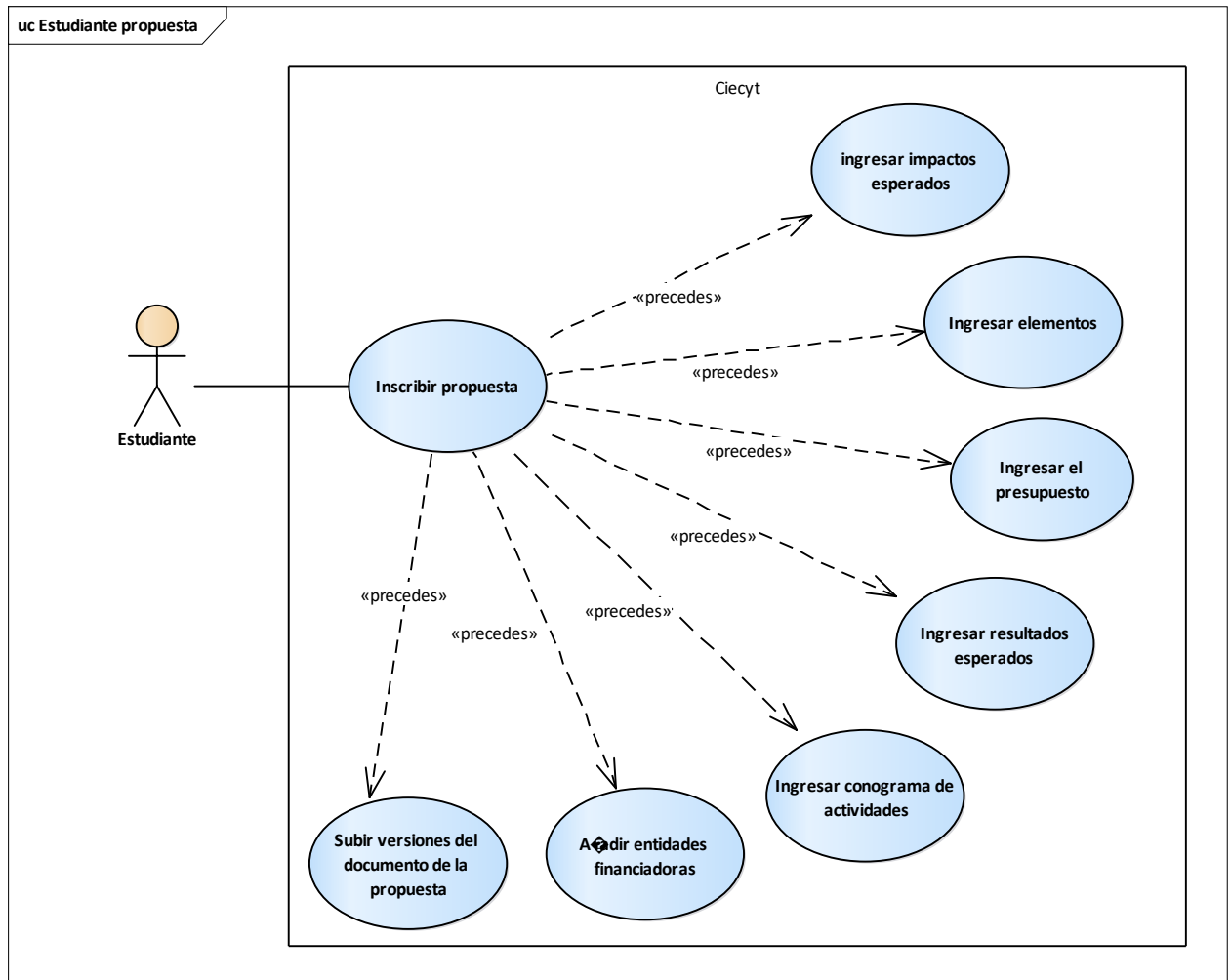
### 4.2.2.2. Modelo de casos de usos

Este modelo ayuda a tomar en cuenta a partir de los requerimientos las acciones que cada actor puede realizar dentro del sistema

## Caso de uso estudiante

Se describe las diferentes acciones que puede realizar con el rol de estudiante dentro del sistema.

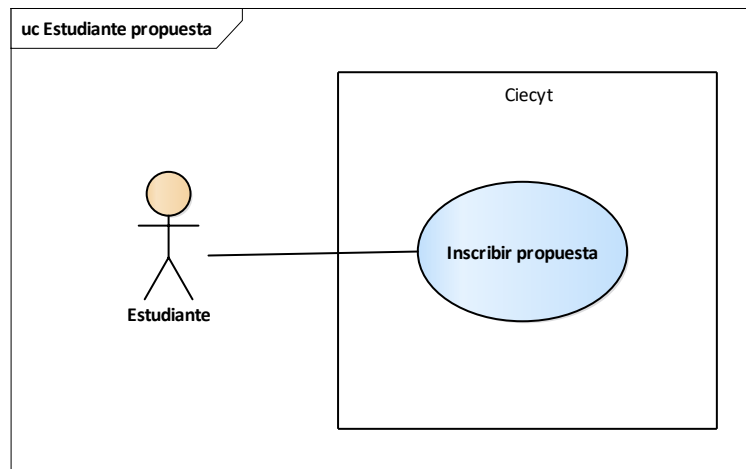
Diagrama 5.Caso de uso de estudiantes



Fuente: Los autores

## Análisis de requerimientos del sistema CIECYT

Diagrama 6. Caso de uso inscripción propuesta de trabajo de grado



Fuente: Los autores

### Descripción del caso de uso: Inscribir propuesta

#### Curso Básico

El sistema muestra la página inscribir propuesta. El estudiante ingresa datos a un formulario con diferentes campos. Ingresará el título de la propuesta (máximo doce palabras), tendrá la opción de elegir el semillero o grupo de investigación (seleccionables opcionales), de la misma forma elegirá la facultad, programa, línea, sublínea y tipo de investigación. Además, ingresará el lugar de ejecución y la duración de su propuesta (en meses), luego se mostrarán campos donde se ingresará la fecha de inicio y la fecha final. También ingresará valores para los siguientes campos: contrapartida en especie (dato decimal), contrapartida en efectivo (dato decimal), palabras clave y nombre de la convocatoria.

Una vez termine de diligenciar datos hace clic en el botón enviar. El sistema se encargará de verificar que los campos obligatorios estén llenos, y el resto de campos como fechas, texto y números tengan datos válidos. El sistema también verifica que no exista otra propuesta con el mismo título. El sistema crea entonces la propuesta en la base de datos. Luego el sistema muestra la página “agregar elementos”.

### Curso Alternativo

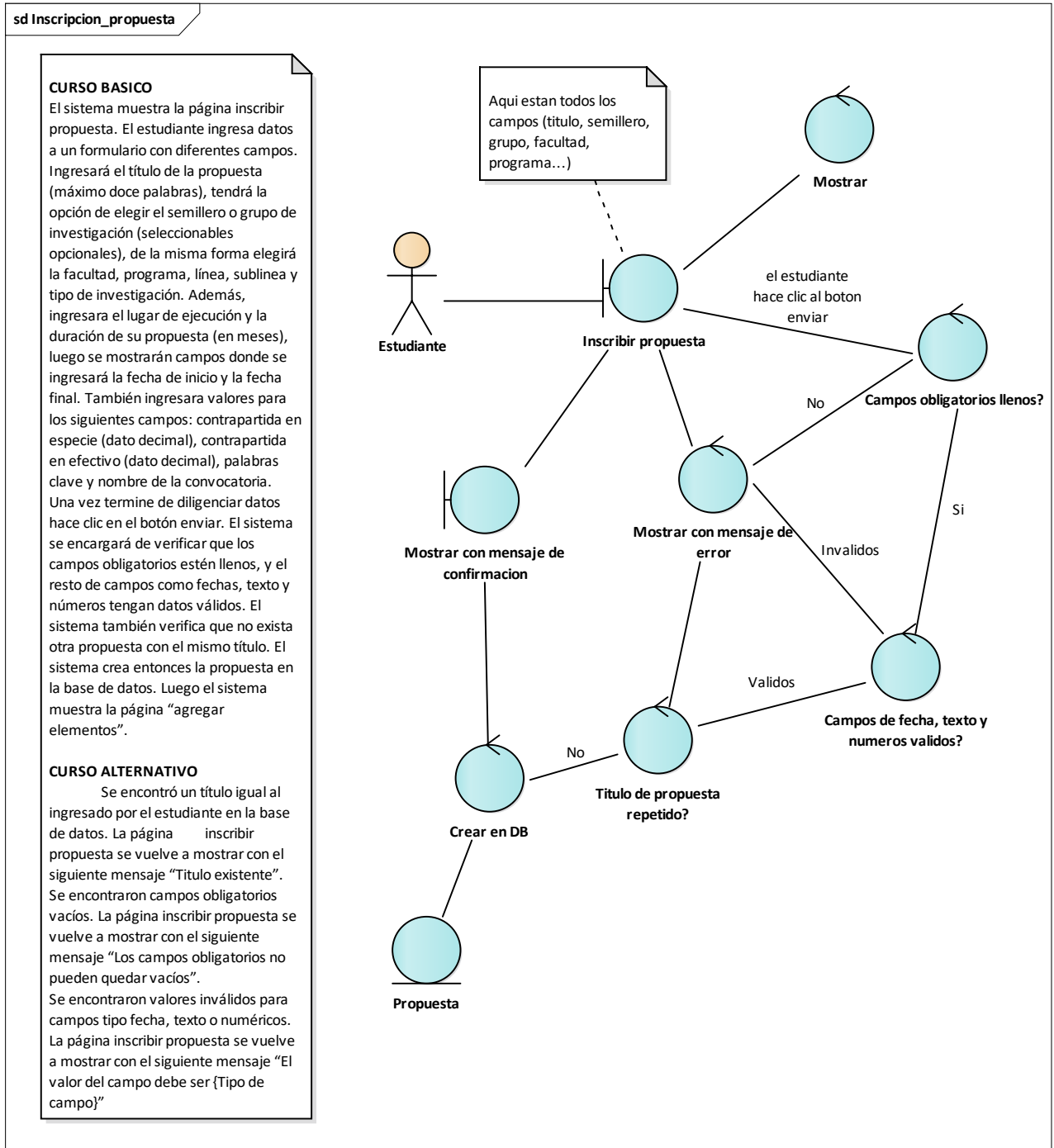
Se encontró un título igual al ingresado por el estudiante en la base de datos. La página inscribir propuesta se vuelve a mostrar con el siguiente mensaje “Titulo existente”.

Se encontraron campos obligatorios vacíos. La página inscribir propuesta se vuelve a mostrar con el siguiente mensaje “Los campos obligatorios no pueden quedar vacíos”.

Se encontraron valores inválidos para campos tipo fecha, texto o numéricos. La página inscribir propuesta se vuelve a mostrar con el siguiente mensaje “El valor del campo debe ser {Tipo de campo}”

## Diagrama de robustez

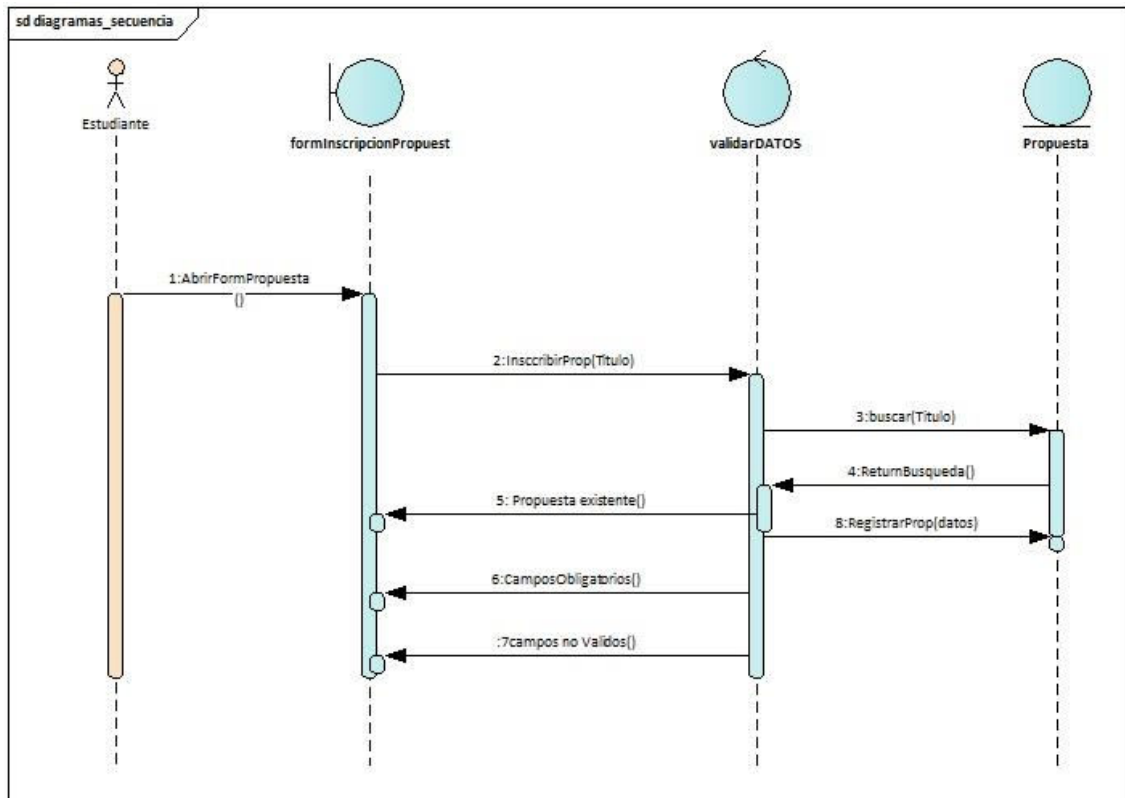
Diagrama 7. Diagrama de robustez Inscribir propuesta



Fuente: Los autores

## Diagrama de secuencia

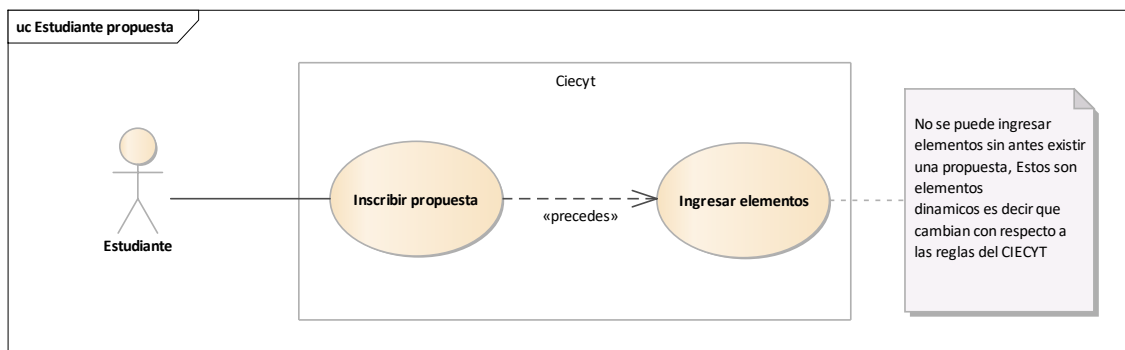
Diagrama 8. Diagrama de secuencia inscribir propuesta



Fuente: Los autores

## Diagrama de casos de uso

Diagrama 9. Caso de uso Ingresar Elementos



Fuente: Los autores

## **Descripción del caso de uso: ingresar elementos**

### Curso Básico

Si el estudiante ha inscrito una propuesta tendrá la opción de ingresar elementos, cuando el estudiante da clic al botón elementos el sistema muestra la página inscribir elementos. El estudiante ingresa valores a los campos disponibles en el formulario (título, objetivo general, objetivos específicos, resumen ejecutivo...).

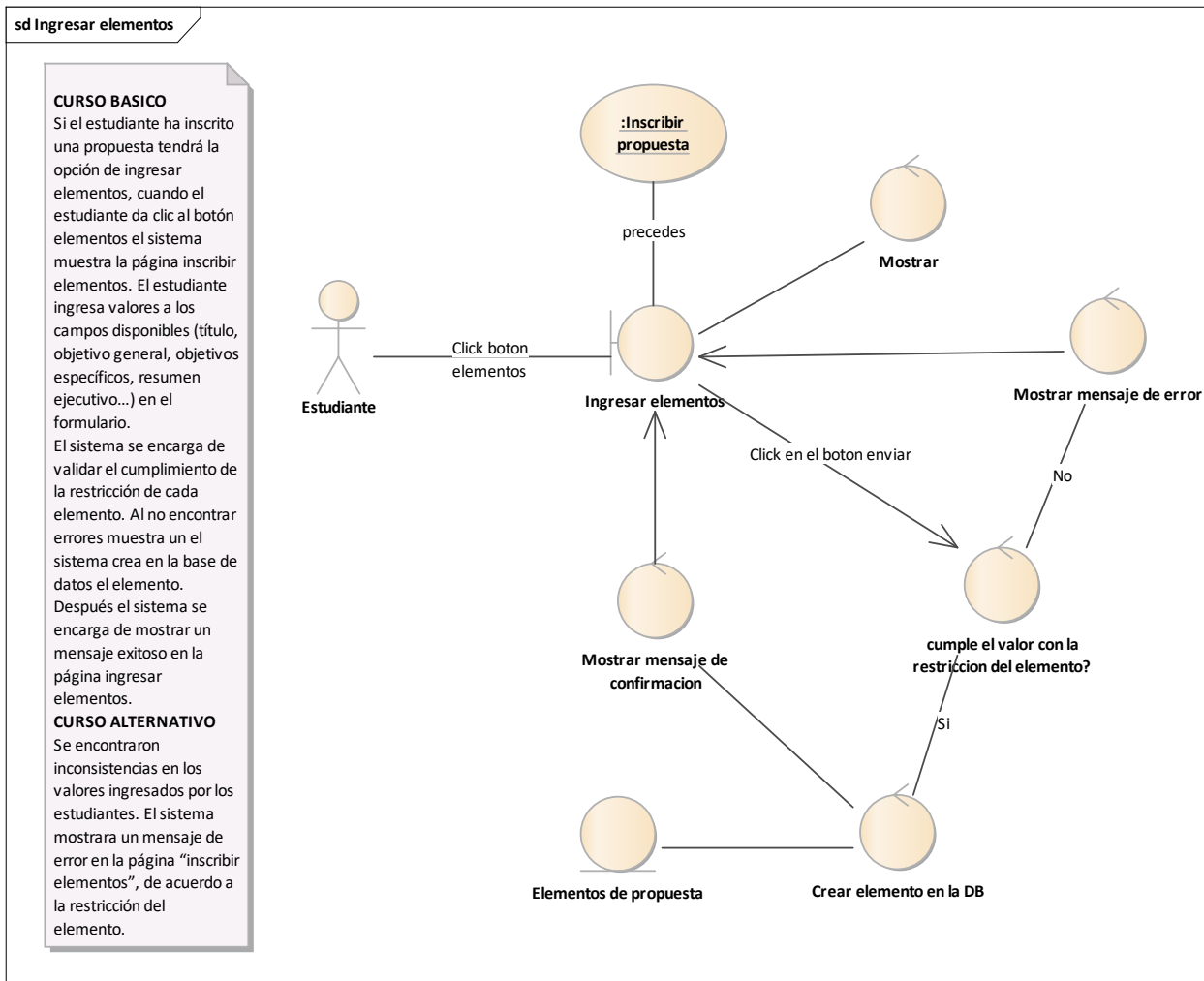
El sistema se encarga de validar el cumplimiento de la restricción del elemento ingresado por el estudiante. Al no encontrar errores el sistema registra el valor del elemento en la base de datos. Después el sistema se encarga de mostrar un mensaje exitoso en la página ingresar elementos.

### Curso Alternativo

Se encontraron inconsistencias en el dato ingresados por el estudiante. El sistema mostrara un mensaje de error en la página “inscribir elementos”.

## **Diagrama de robustez**

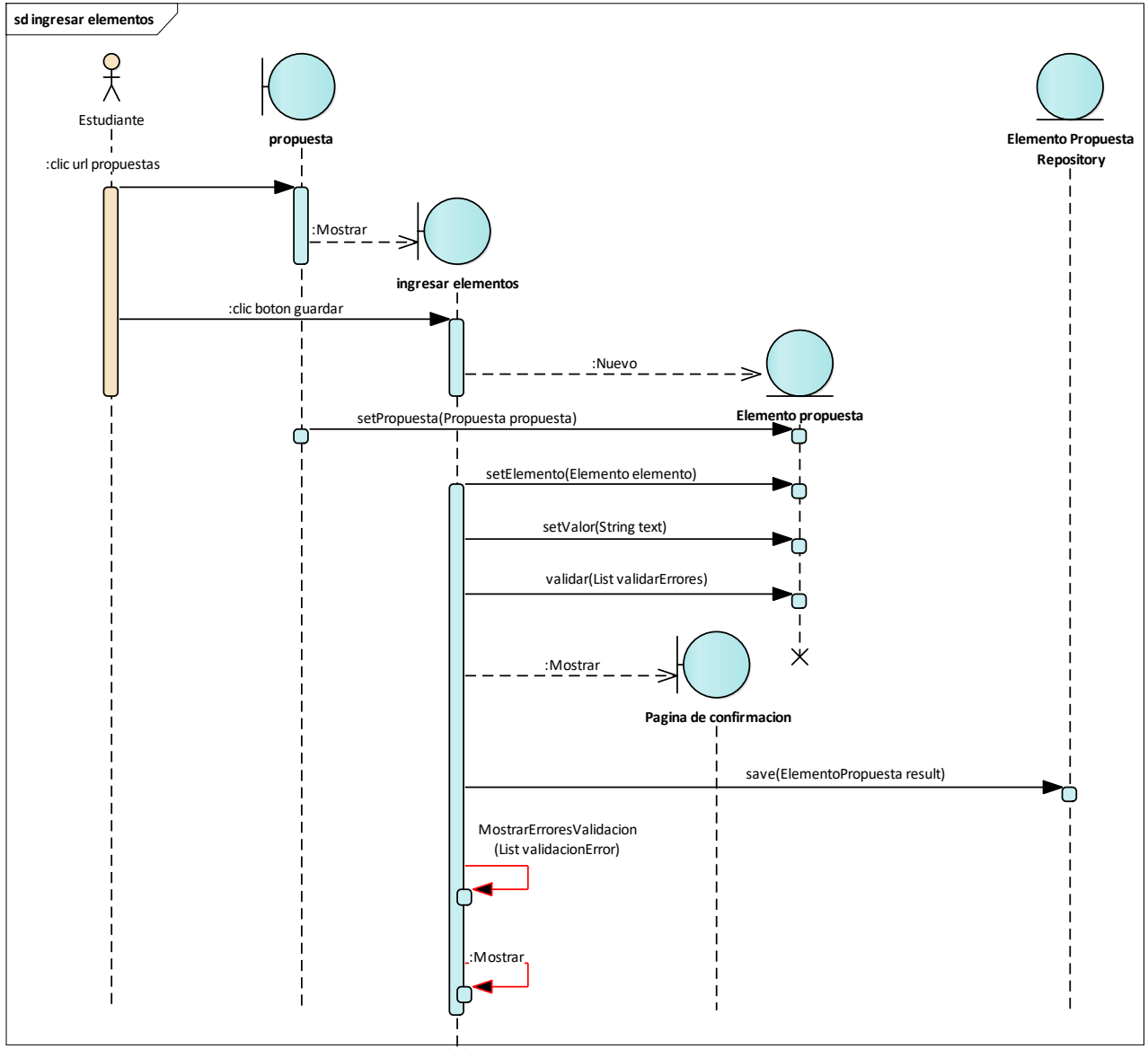
Diagrama 10. Diagrama de robotez ingresar elementos



Fuente: Los autores

## Diagrama de secuencia

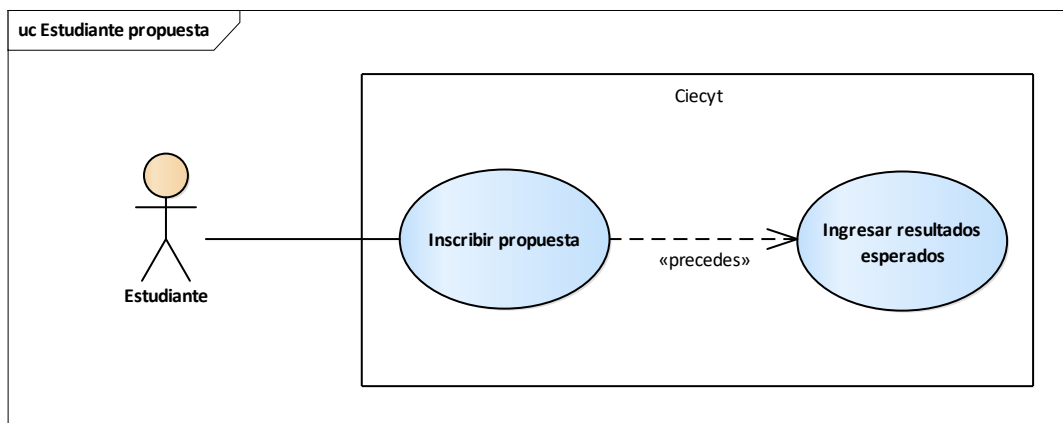
Diagrama 11. Diagrama de secuencia ingresar elementos



Fuente: Los autores

## Caso de uso ingresar resultados esperados

Diagrama 12. Caso de uso ingresar resultados esperados



Fuente: Los autores

### Descripción del caso de uso: Ingresar resultados esperados

#### Curso Básico

Una vez exista una propuesta registrada en la base de datos el estudiante podrá hacer clic a la pestaña “resultados esperados” donde ingresara el nombre del resultado, el indicador y el beneficiario. El estudiante le da clic al botón agregar el sistema se encargará de validar los datos ingresados que coincidan con el tipo de dato predeterminado y que cumplan con las restricciones preestablecidas. Si no encontró errores el sistema guarda los valores en la base de datos y vuelve a mostrar la página con un mensaje de éxito.

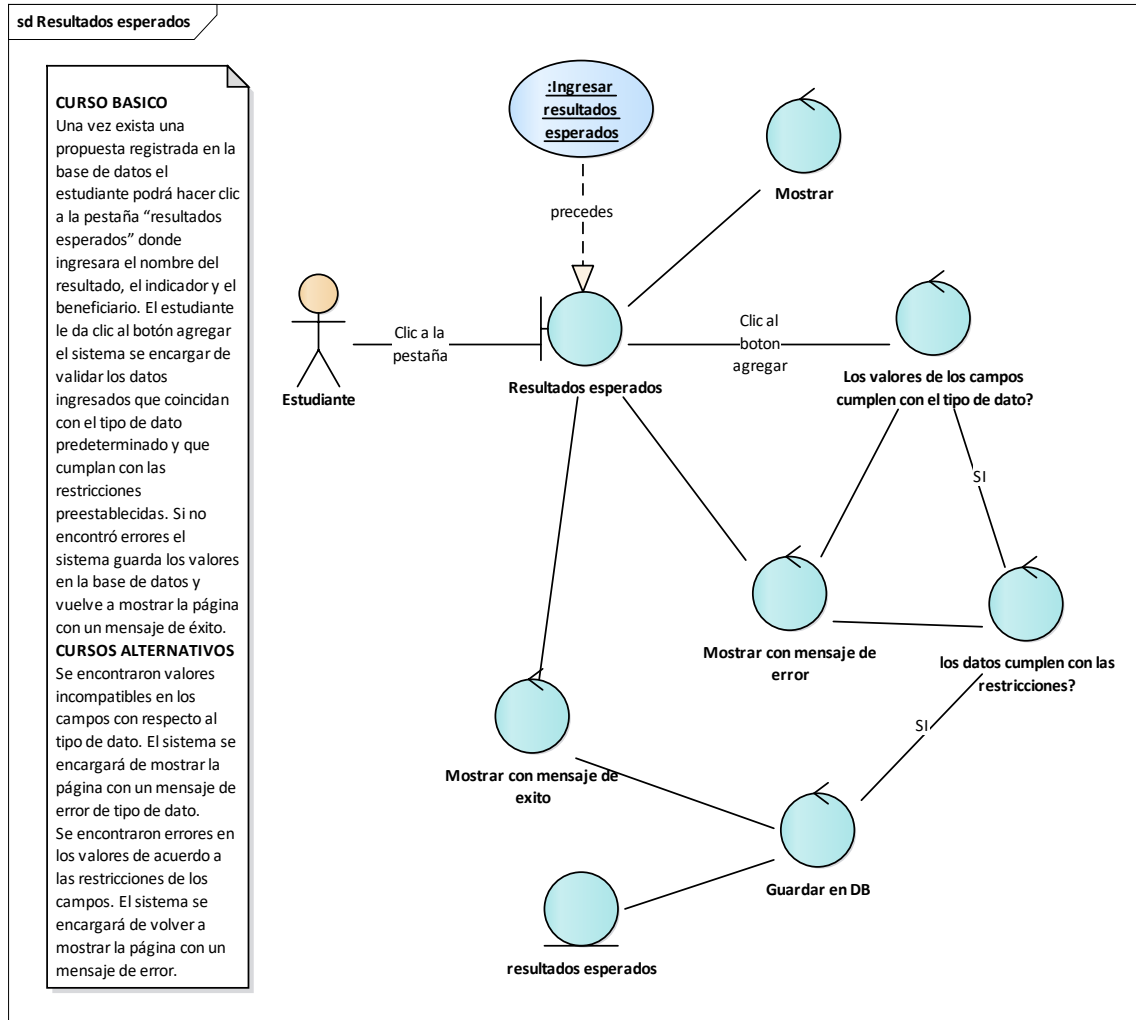
#### Cursos Alternativos

Se encontraron valores incompatibles en los campos con respecto al tipo de dato. El sistema se encargará de mostrar la página con un mensaje de error de tipo de dato.

Se encontraron errores en los valores de acuerdo a las restricciones de los campos. El sistema se encargará de volver a mostrar la página con un mensaje de error.

## Diagrama de robustez

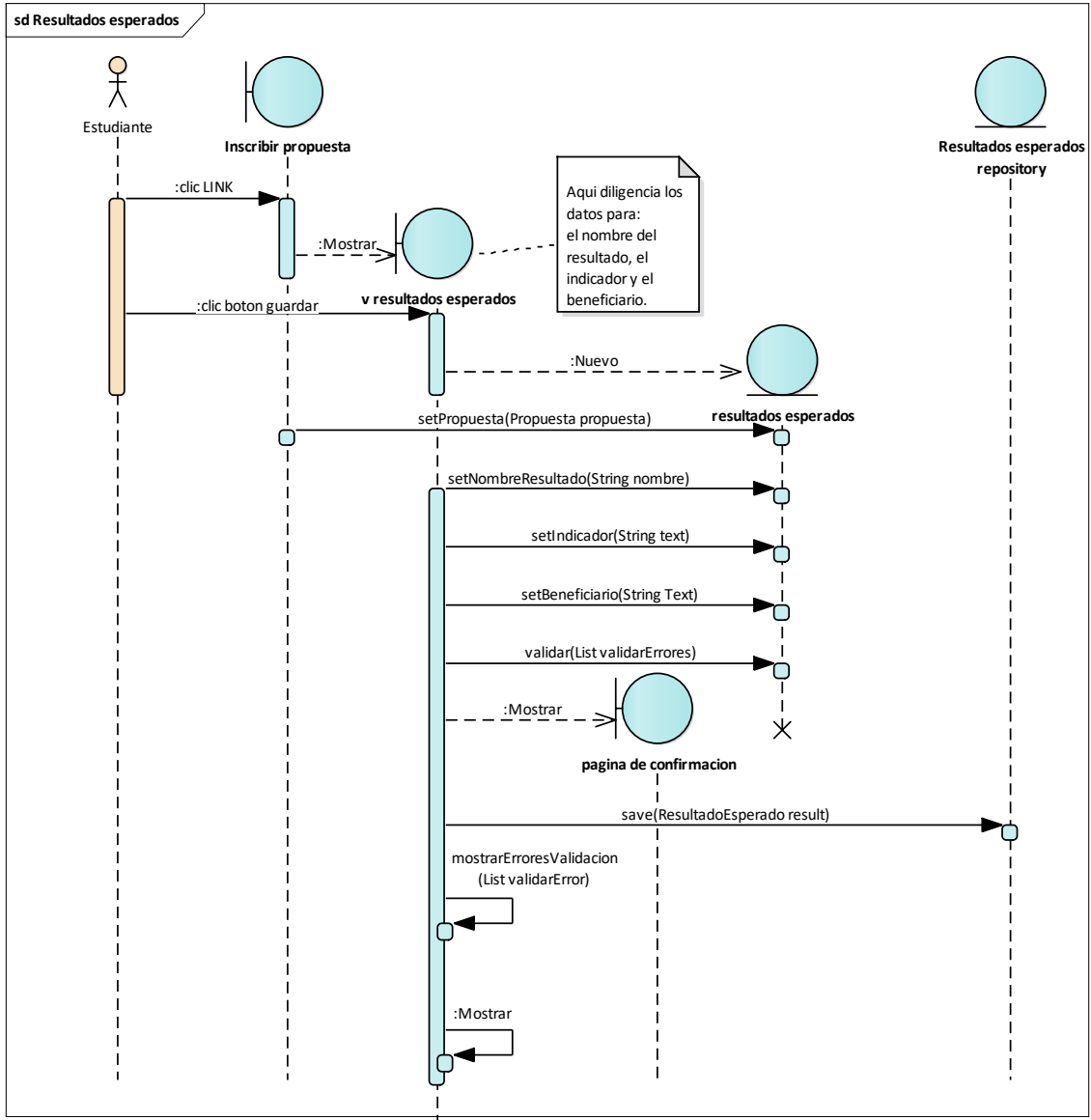
Diagrama 13. Diagrama de robustez ingresar resultados esperados



Fuente: Los autores

## Diagrama de secuencia

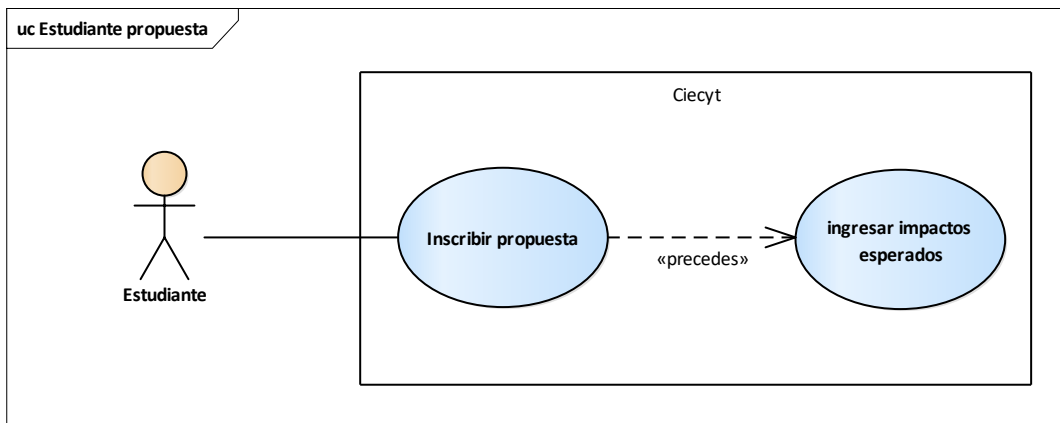
Diagrama 14. Diagrama de secuencia ingresar resultados esperados



Fuente: Los autores

## Descripción del caso de uso: Ingresar impactos esperados

Diagrama 15. Caso de uso ingresar impactos esperados



Fuente: Los autores

El sistema permitirá al estudiante ingresar los impactos esperados de su propuesta a través de un formulario, el cual tendrá los siguientes campos: impacto esperado (pequeña descripción de 300 caracteres), debe elegir el tiempo de duración en años de la siguiente manera: corto (1-4), mediano (5-9), largo (10 o más). también el campo indicador verificable (tipo texto) además de los supuestos (tipo texto).

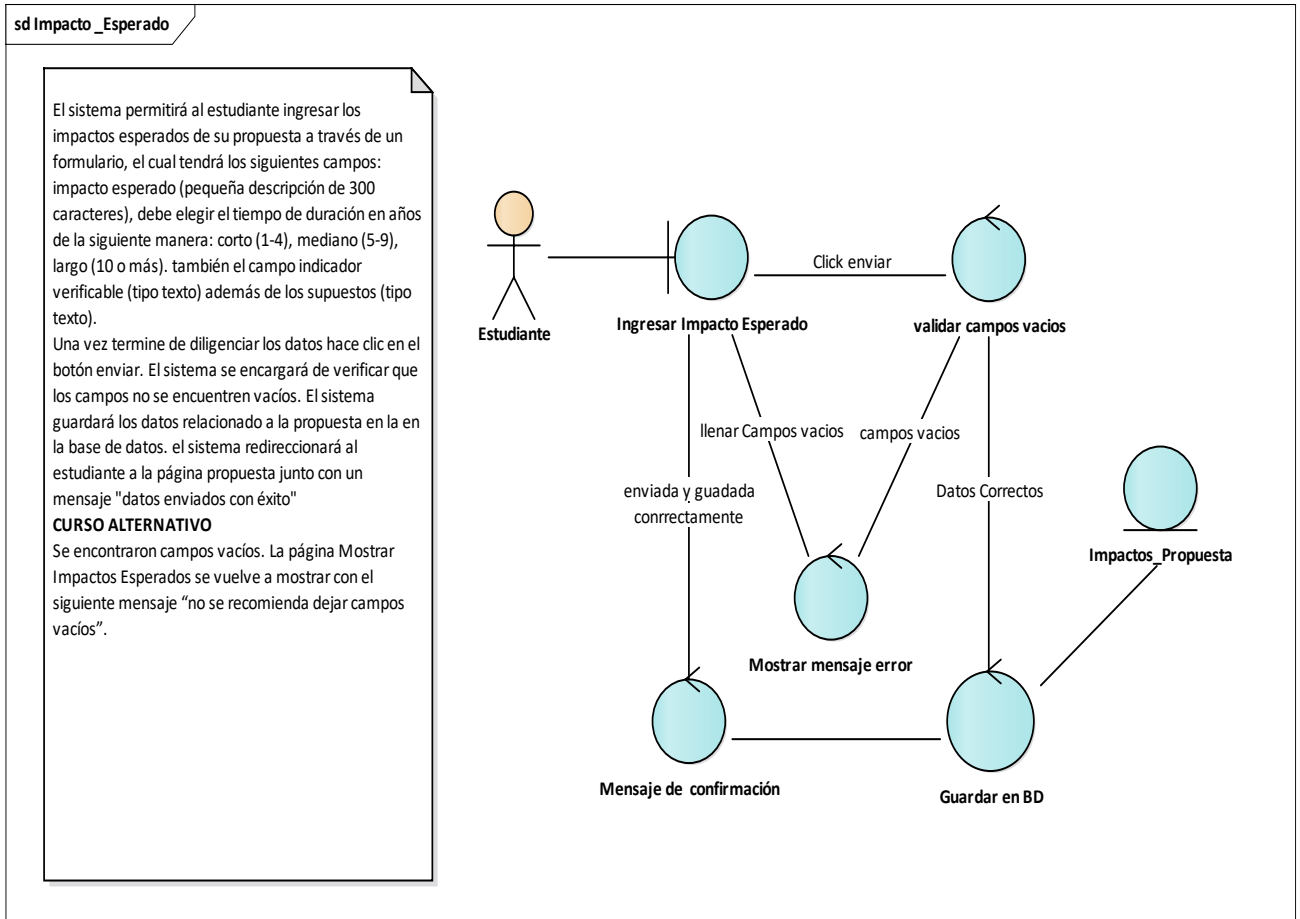
Una vez termine de diligenciar los datos hace clic en el botón Guardar. El sistema se encargará de verificar que los campos no se encuentren vacíos. El sistema guardará los datos relacionados a la propuesta en la base de datos. el sistema re direccionará al estudiante a la página propuesta junto con un mensaje "datos enviados con éxito"

### Curso Alternativo

Se encontraron campos vacíos. La página Mostrar Impactos Esperados se vuelve a mostrar con el siguiente mensaje "no se recomienda dejar campos vacíos".

## Diagrama de robustez

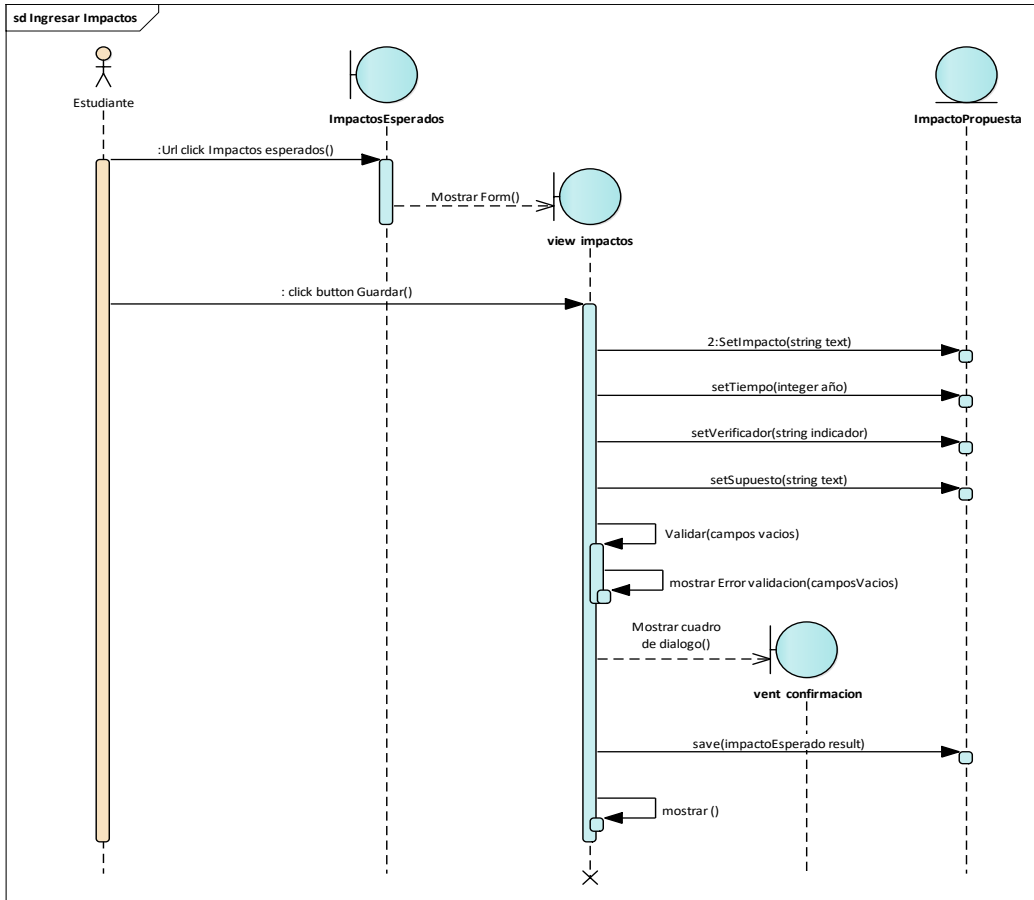
Diagrama 16.de robustez ingresar impactos esperados



Fuente: Los autores

# Diagrama de secuencia

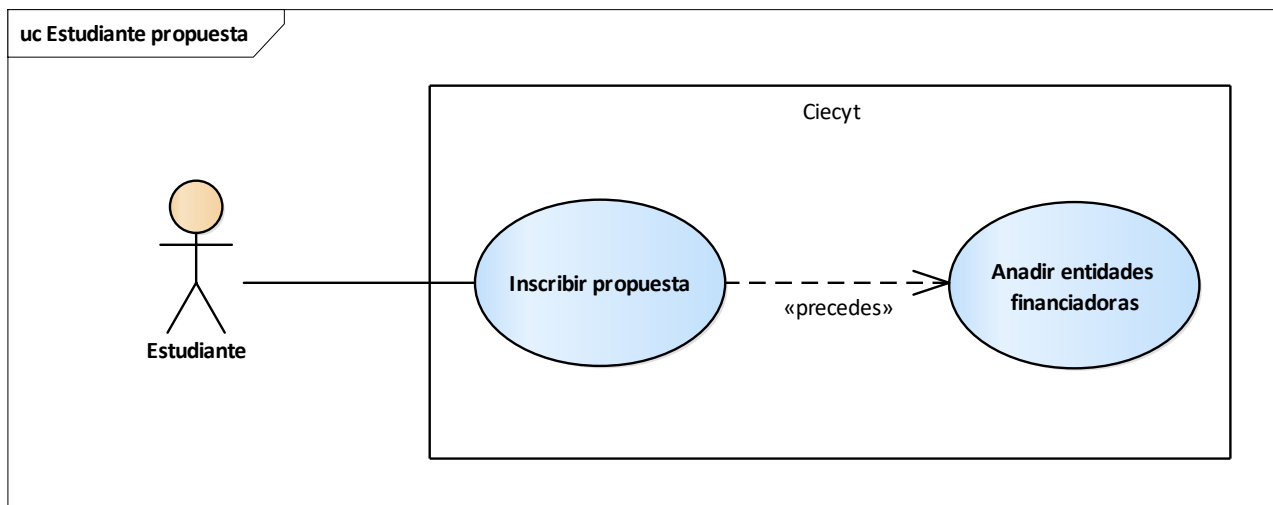
Diagrama 17.de secuencia ingresar impactos esperados



Fuente: Los autores

## Descripción del caso de uso: AÑADIR ENTIDADES FINANCIERAS

Diagrama 18.Caso de uso Añadir entidades financiadoras



Fuente: Los autores

### Curso Básico

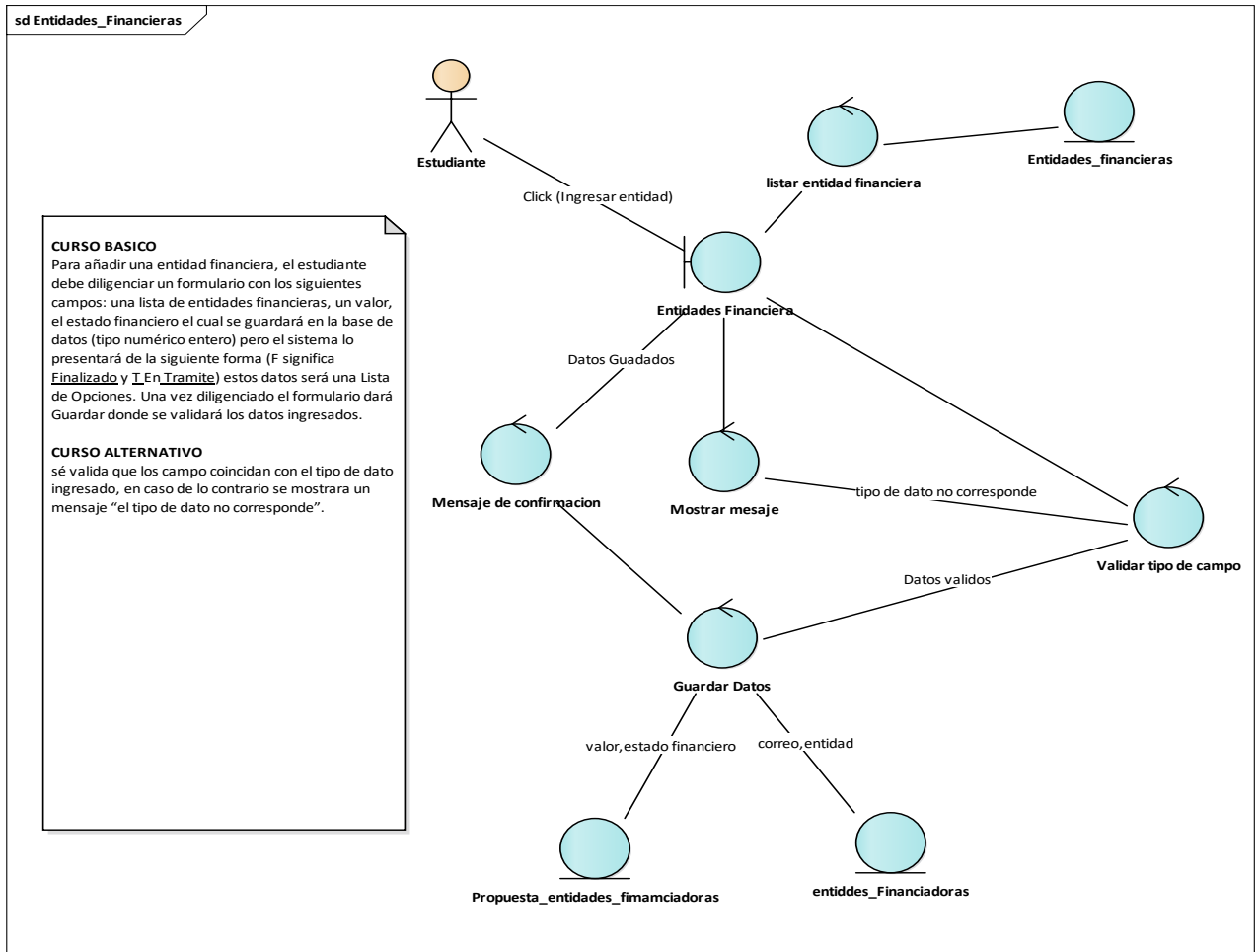
Para añadir una entidad financiera, el estudiante debe diligenciar un formulario con los siguientes campos: una lista de entidades financieras, un valor, el estado financiero el cual se guardará en la base de datos (tipo numérico entero) pero el sistema lo presentará de la siguiente forma (F significa Finalizado y T En Tramite) estos datos será una Lista de Opciones. Una vez diligenciado el formulario dará Guardar donde se validará los datos ingresados.

### Curso Alternativo

Se valida que los campo coincidan con el tipo de dato ingresado, en caso de lo contrario se mostrara un mensaje “el tipo de dato no corresponde”.

## Diagrama de robustez

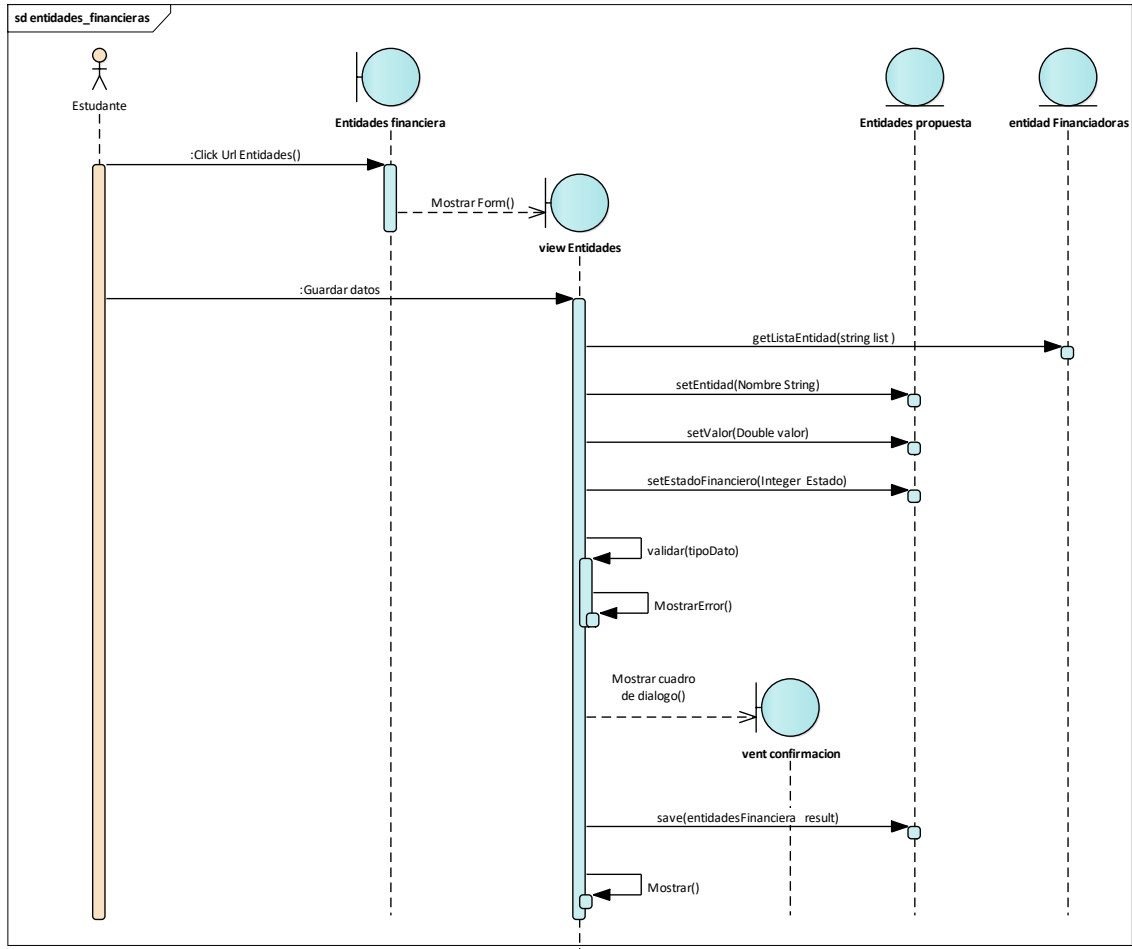
Diagrama 19. Diagrama de Robustez añadir entidades financiera



Fuente: Los autores

## Diagrama de secuencia

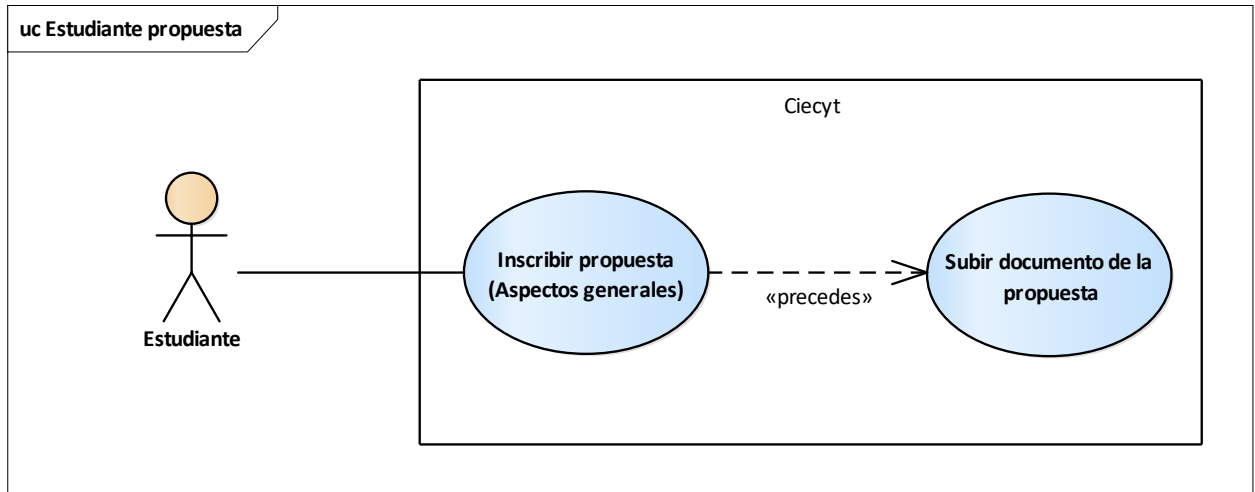
Diagrama 20. De secuencia añadir entidades financiera



Fuente: Los autores

## Descripción del caso de uso: SUBIR DOCUMENTO DE LA PROPUESTA

Diagrama 21. Caso de uso subir documento de la propuesta



Fuente: Los autores

### Curso Básico

Se necesita que el estudiante tenga los aspectos generales de la propuesta registrados en el sistema. Cuando el estudiante de clic al botón “Subir documento” se le desplegará un campo drag and drop donde podrá colocar un documento de Word o eligiendo directamente desde su ordenador. Si guarda los cambios el sistema se encargará de verificar que el campo no este vacío después subirá el archivo al servidor y guardara la ruta del documento en la base de datos, también controlará la versión del documento por medio de un contador que se incremente a medida que el estudiante suba varias veces el documento. Al no existir errores el sistema se encarga de mostrar un mensaje de éxito.

### Curso Alternativo

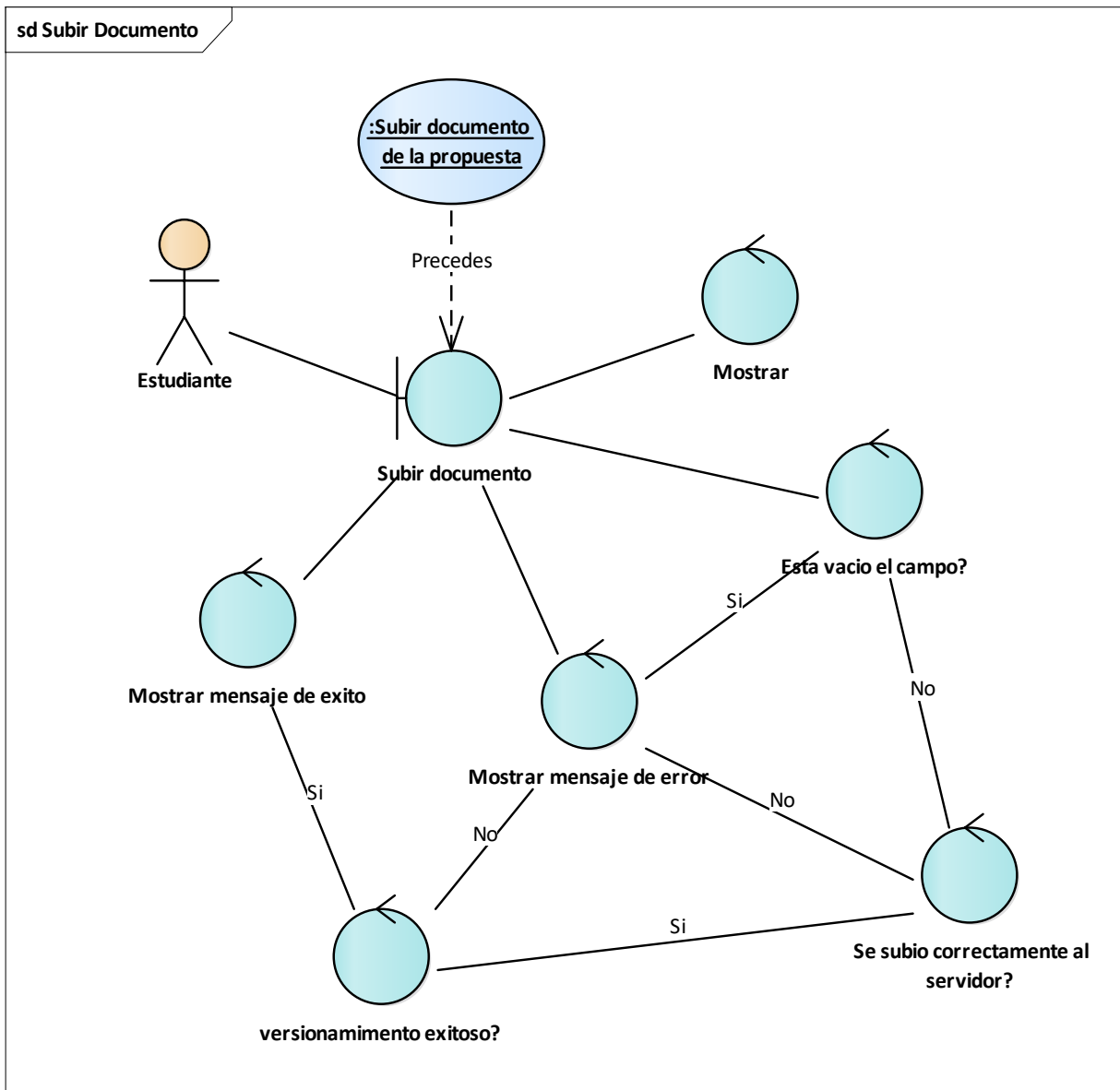
El campo está vacío, el sistema se encarga de mostrar un mensaje de error solicitando el diligenciamiento del campo.

Sucedió algún error al subir el archivo al servidor, el sistema mostrara el mensaje de error del servidor.

Sucedió algún error al generar el número de la versión del documento, el sistema mostrara un mensaje de error de numero de versión.

### Diagrama de robustez

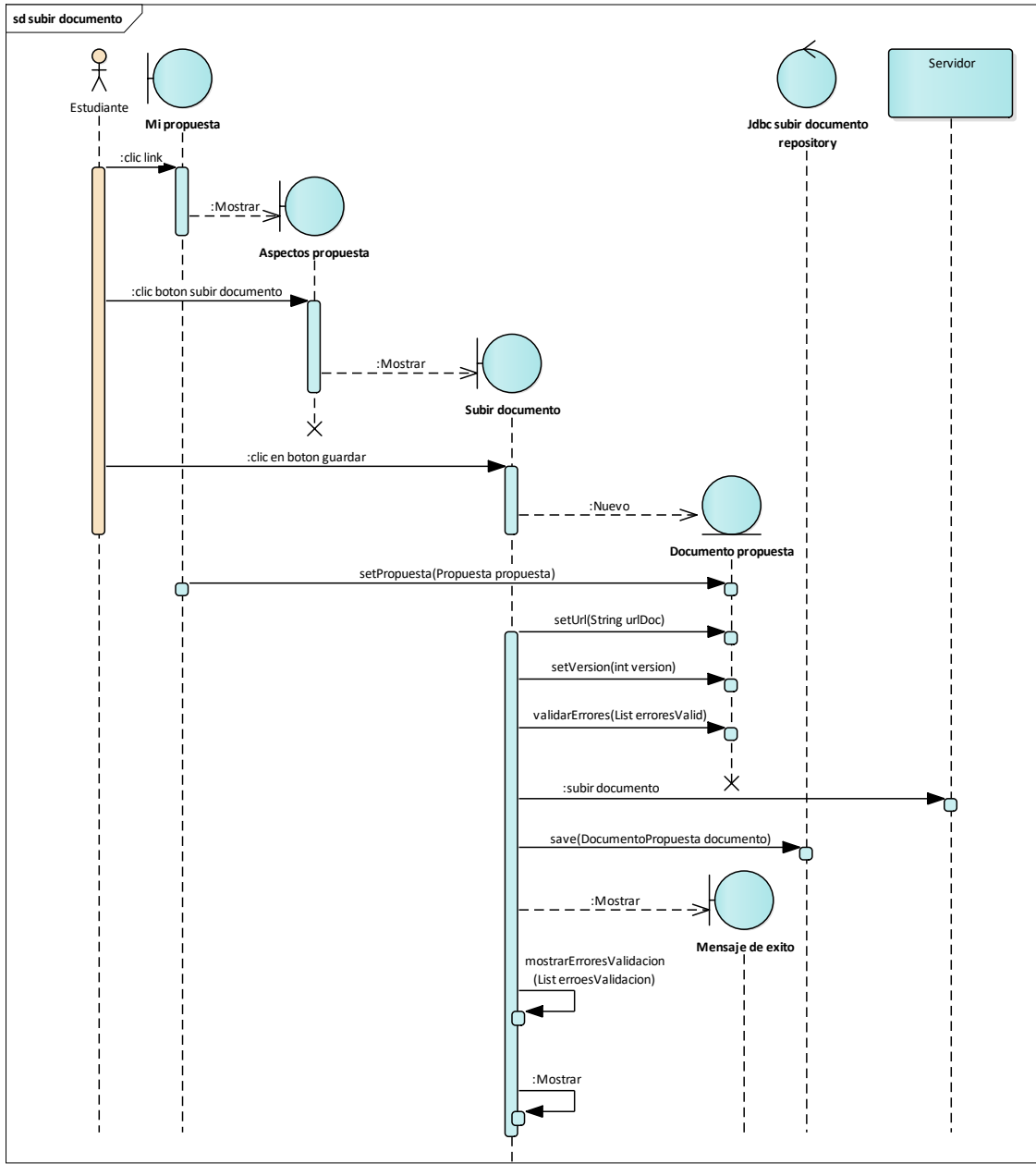
Diagrama 22. Diagrama de robustez subir documento a la propuesta



Fuente: Los autores

# Diagrama de secuencia

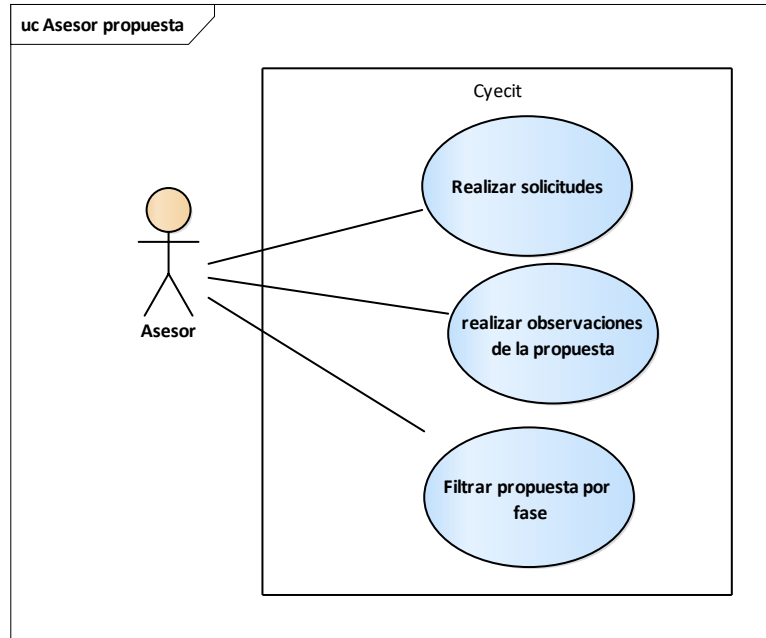
Diagrama 23. Diagrama de secuencia subir documento a la propuesta



Fuente: Los autores

## Casos de uso para el actor Asesor

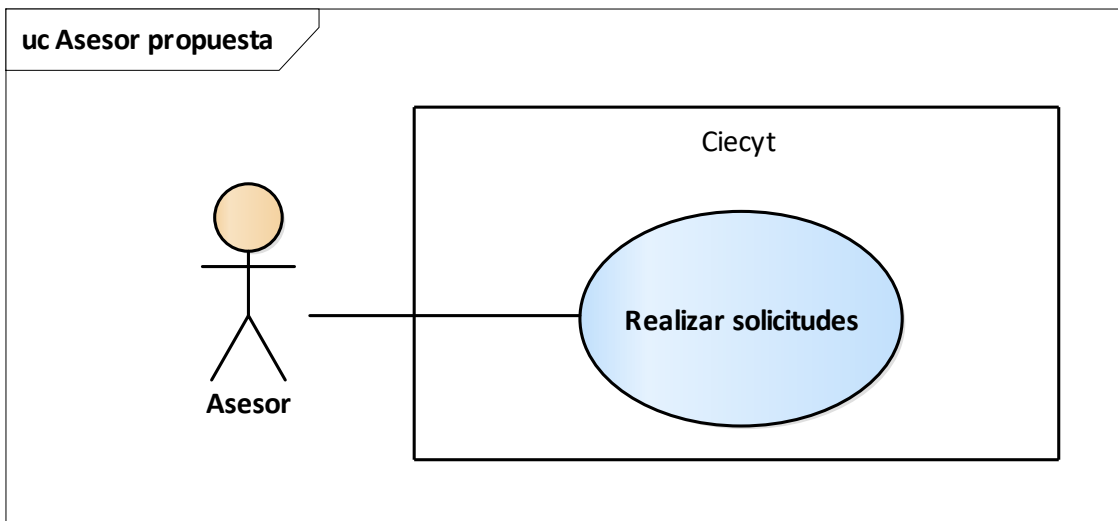
Diagrama 24. Caso de uso para el actor asesor



Fuente: Los autores

## Análisis de requerimientos del sistema CIECYT

Diagrama 25. Requerimientos del sistema CIECYT



Fuente: Los autores

## **Descripción del caso de uso: Realizar solicitudes**

### Curso Básico

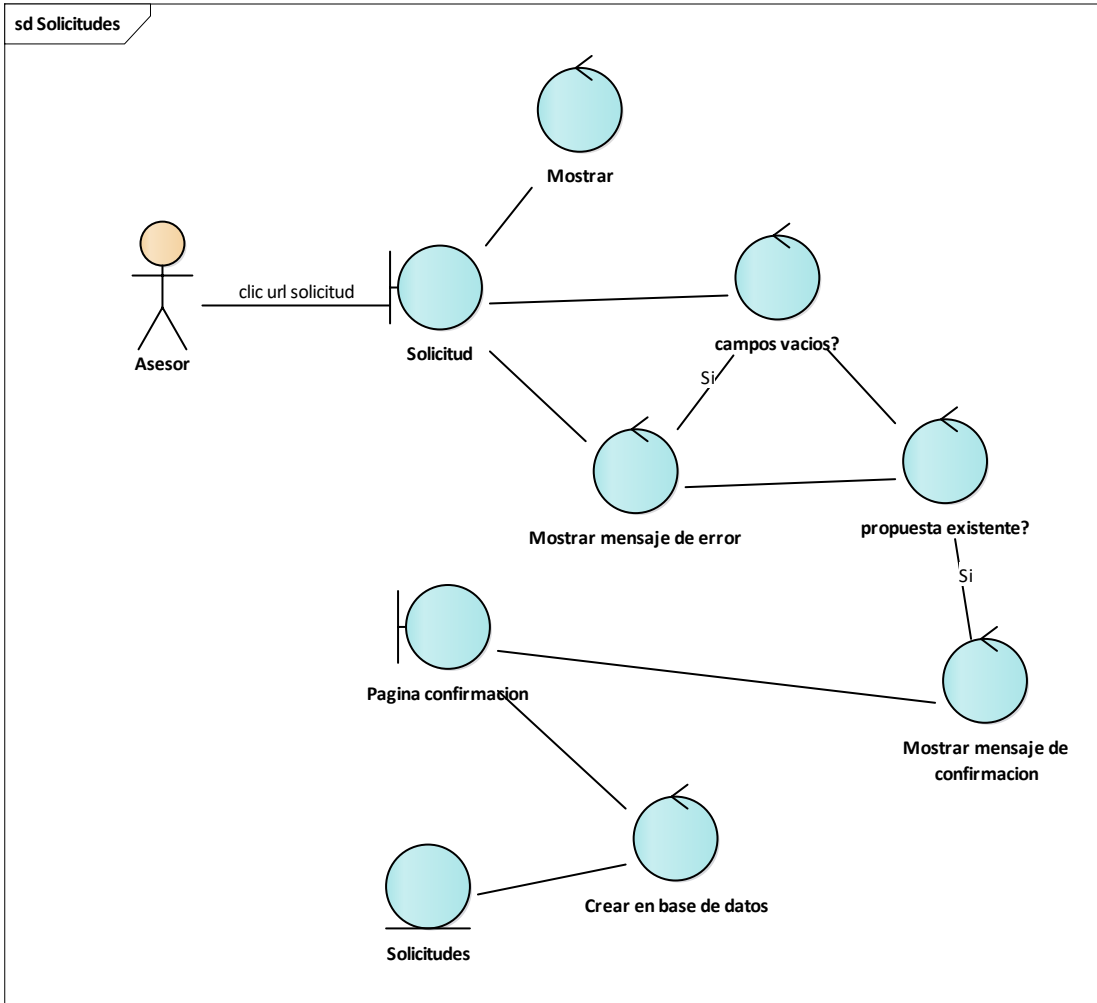
El asesor da clic a la URL solicitudes se le mostrará un formulario donde buscará la propuesta por su nombre, e ingresará el asunto. El usuario envía la solicitud, el sistema se encarga de validar que no haya campos vacíos y exista una propuesta. Si no existen errores en la validación el sistema se encargará de mostrar un mensaje de confirmación después guardará la solicitud en la base de datos para que la evalúe el administrador.

### Curso Alternativo

Si algún campo del formulario se envió vacío el sistema se encargará de mostrar un mensaje de error en la misma página. Si no se encuentra la propuesta mostrara el mensaje “propuesta inexistente”.

## **Diagrama de robustez**

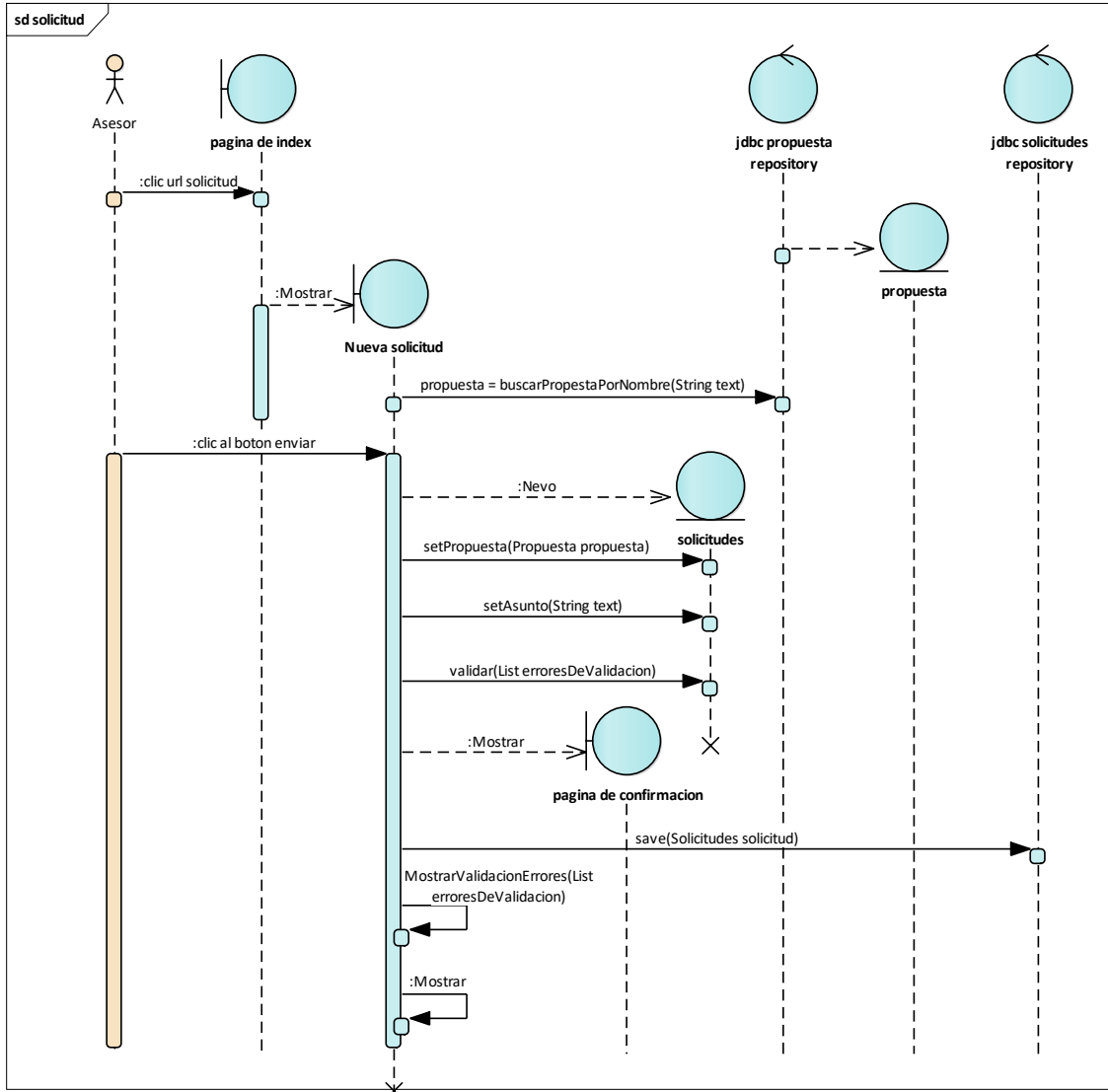
Diagrama 26. Diagrama de Robustez realizar solicitud



Fuente: Los autores

## Diagrama de secuencia

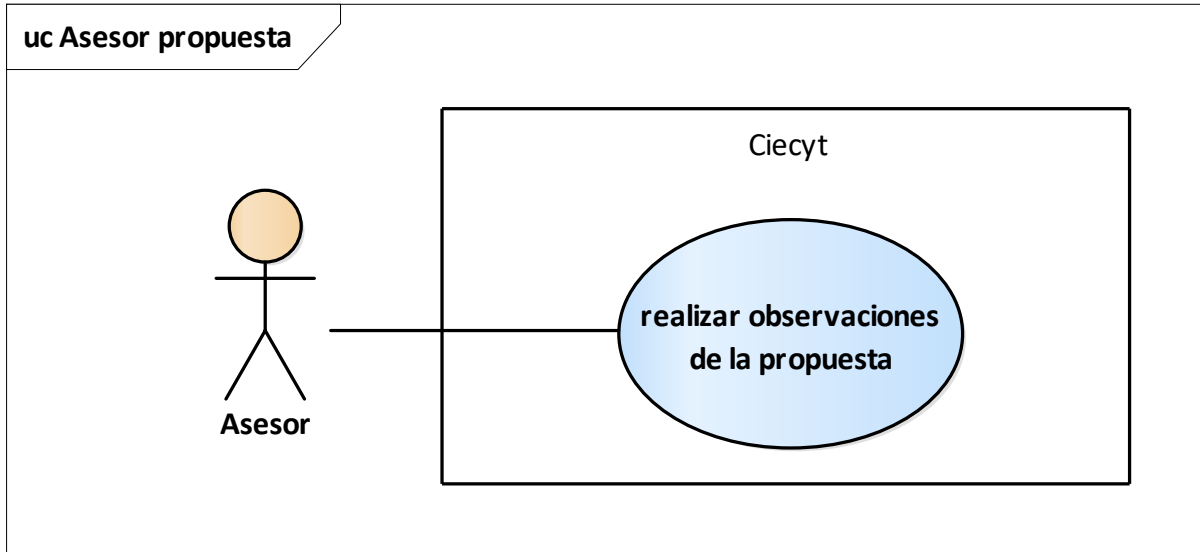
Diagrama 27. Diagrama de secuencias realizar solicitud



Fuente: Los autores

## Diagrama de casos de uso

Diagrama 28. Diagrama de observaciones de la propuesta



Fuente: Los autores

### Descripción del caso de uso: Realizar observaciones de la propuesta

#### Curso Básico

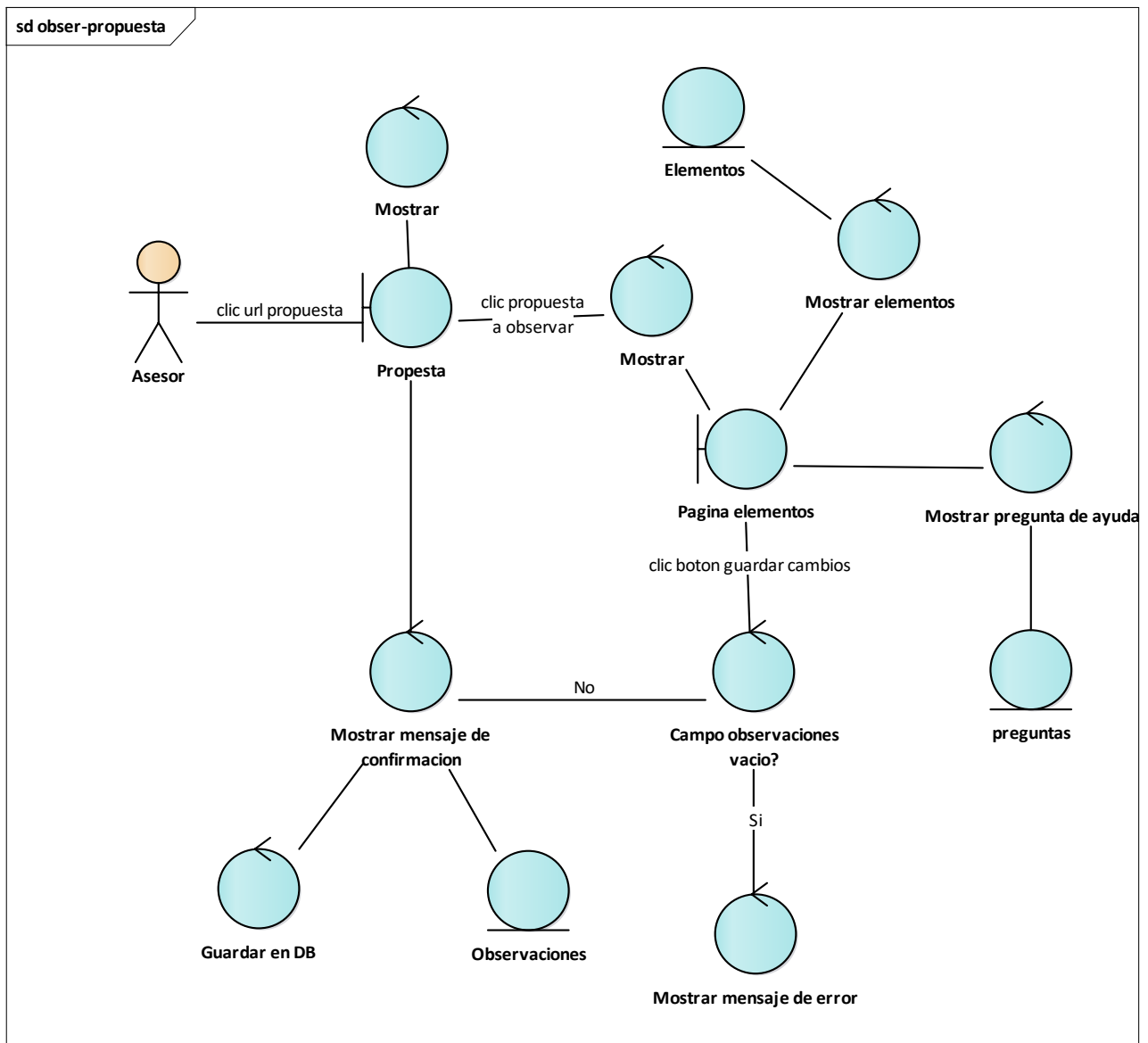
Cuando el asesor da clic a la URL “propuesta” el sistema se encarga de listar todas las propuestas que tenga asignadas, cuando el asesor da clic a una propuesta el sistema en una nueva página desplegará todos los elementos de la propuesta, pero aparecerá por defecto el elemento título, en donde el sistema mostrara: el valor del elemento, una pregunta de ayuda y un campo para ingresar una observación. Cuando el asesor guarda los cambios el sistema se encarga de verificar que el campo observación no este vacío después muestra un mensaje de confirmación y guarda los cambios en la base de datos. Si el asesor da clic a cualquier otro elemento tales como: objetivos, cronograma de actividades, presupuesto entre otros; el proceso a seguir será igual al mencionado anteriormente.

## Curso Alternativo

Si el asesor deja el campo observación vacío el sistema se encargará de mostrar un mensaje de error.

### Diagrama de robustez

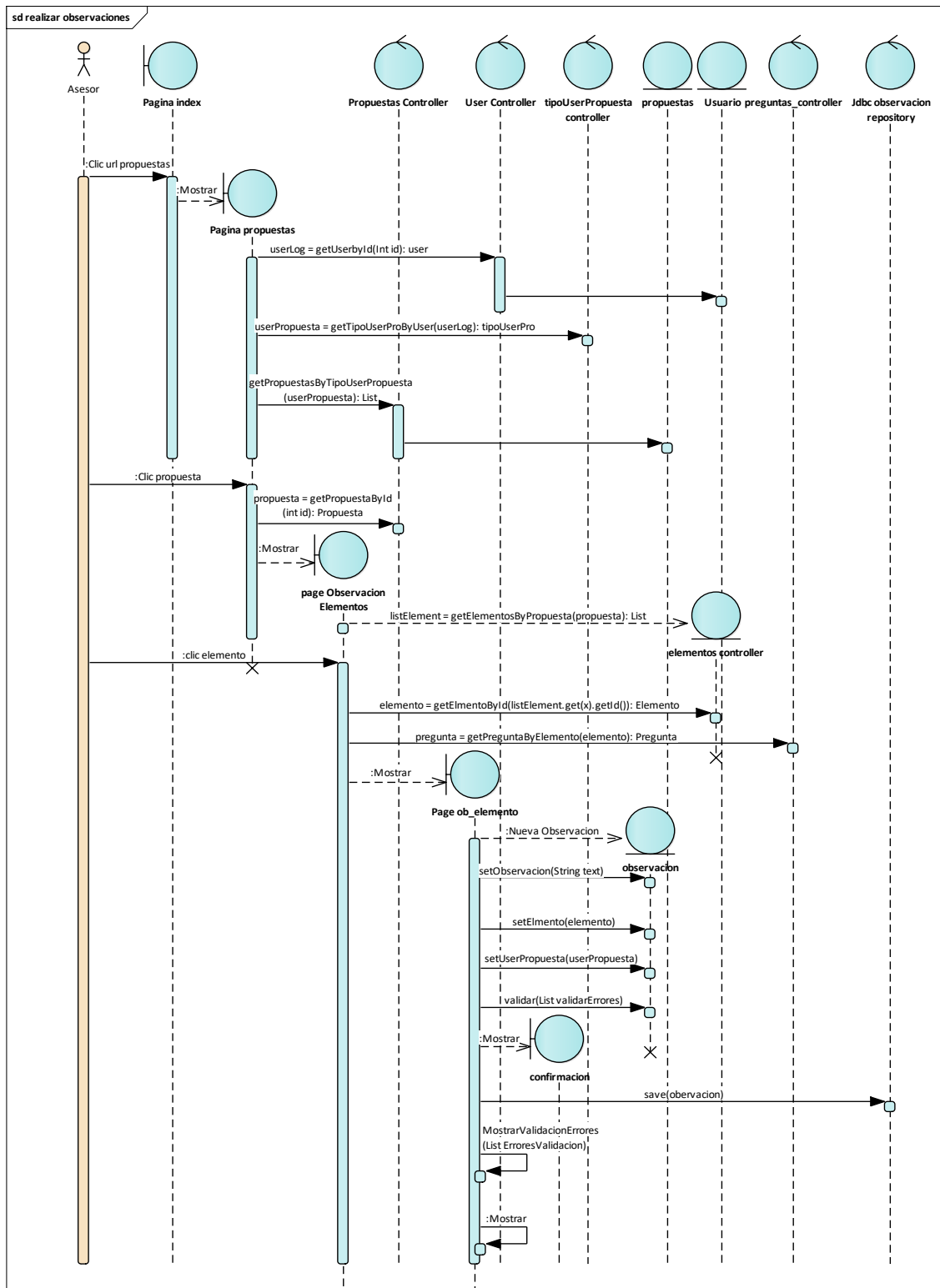
Diagrama 29. Diagrama de robustez realizar observaciones a al propuesta



Fuente: Los autores

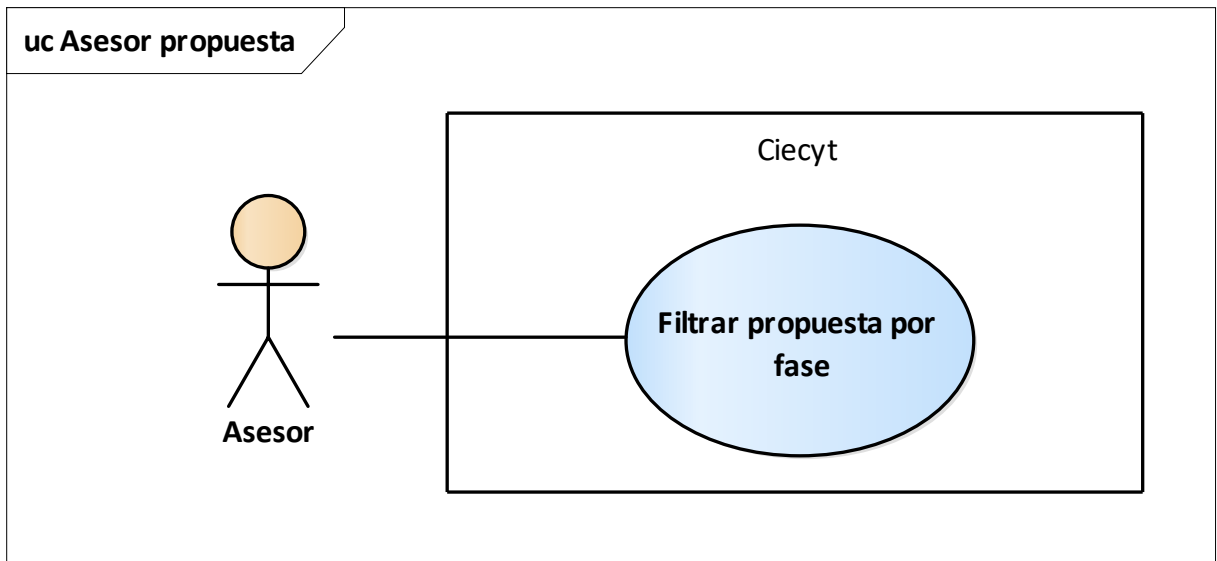
## Diagrama de secuencia

Diagrama 30. Diagrama de secuencia realizar observaciones a la propuesta



Fuente: Los autores

Diagrama 31. Estado de la propuesta



Fuente: Los autores

### **Descripción del caso de uso: Filtrar propuesta por fase**

#### Curso Básico

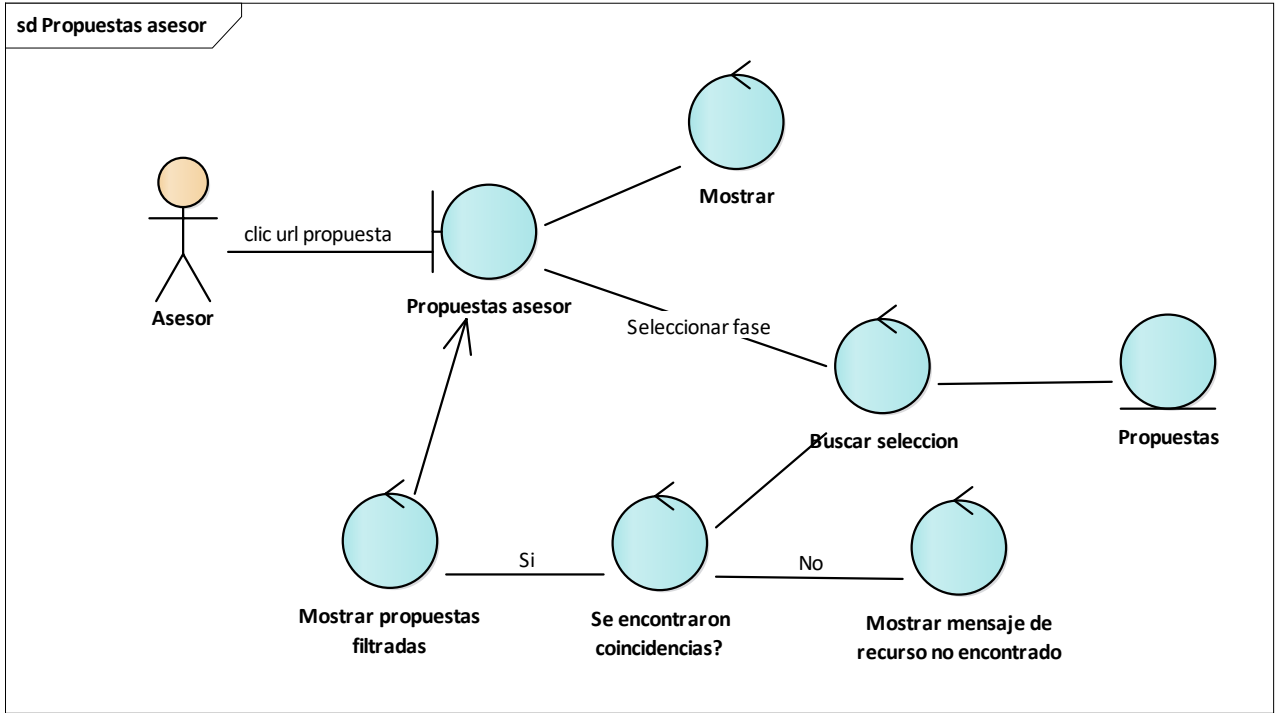
Cuando el usuario da clic a la URL de la propuesta se mostrará la página propuestas junto con una opción de filtro. Si el usuario selecciona alguna de las opciones se filtrarán las propuestas de acuerdo a la opción seleccionada por el usuario, en este punto es sistema se encargará de realizar una búsqueda en la base de datos tratando de mostrar todas las coincidencias encontradas.

#### Cursos Alternativos

Si no se encuentran resultados de la búsqueda el sistema se encargará de mostrar un mensaje de recurso no encontrado junto con la una tabla vacía.

## Diagrama de robustez

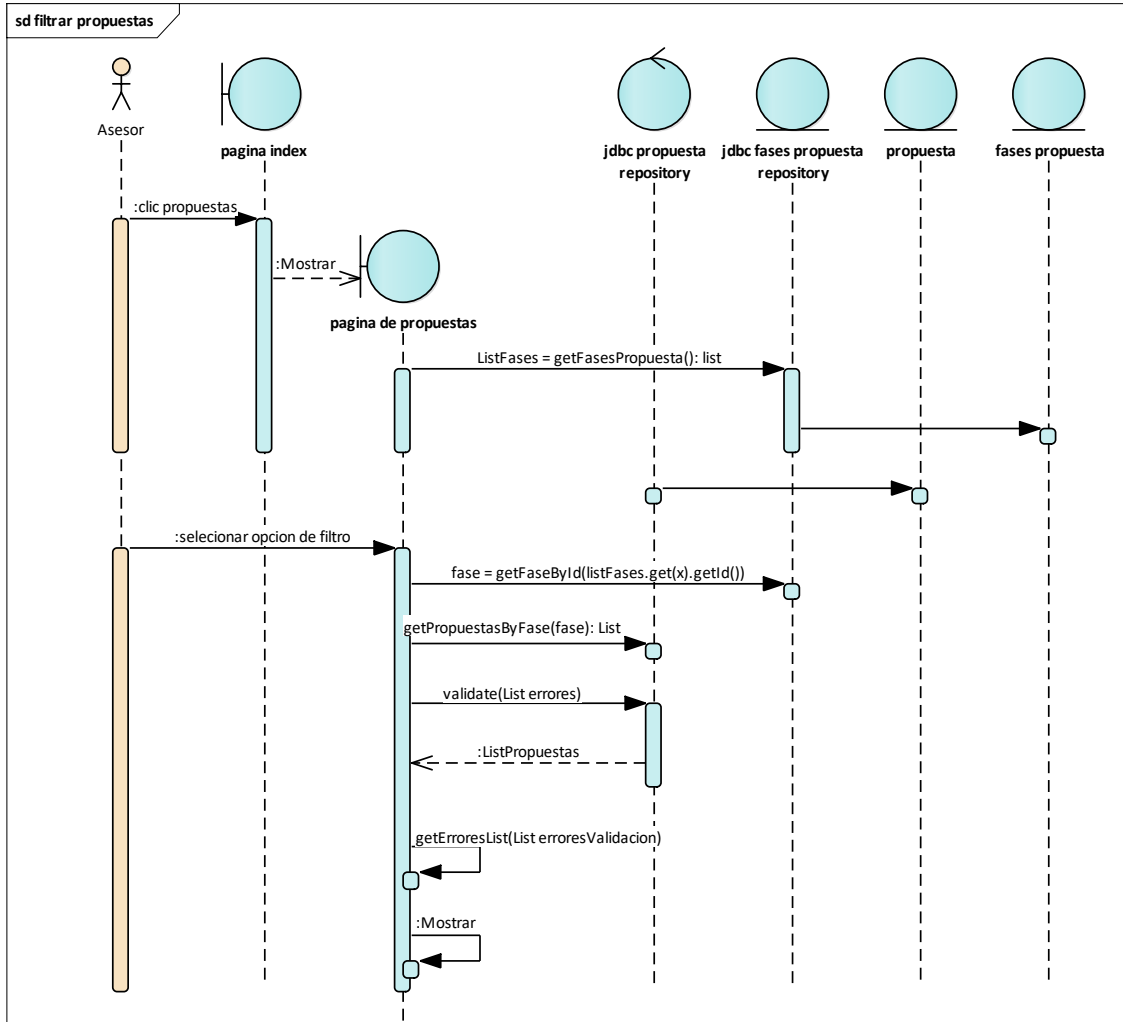
Diagrama 32. Diagrama de robustez estado de la propuesta



Fuente: Los autores

## Diagrama de secuencia

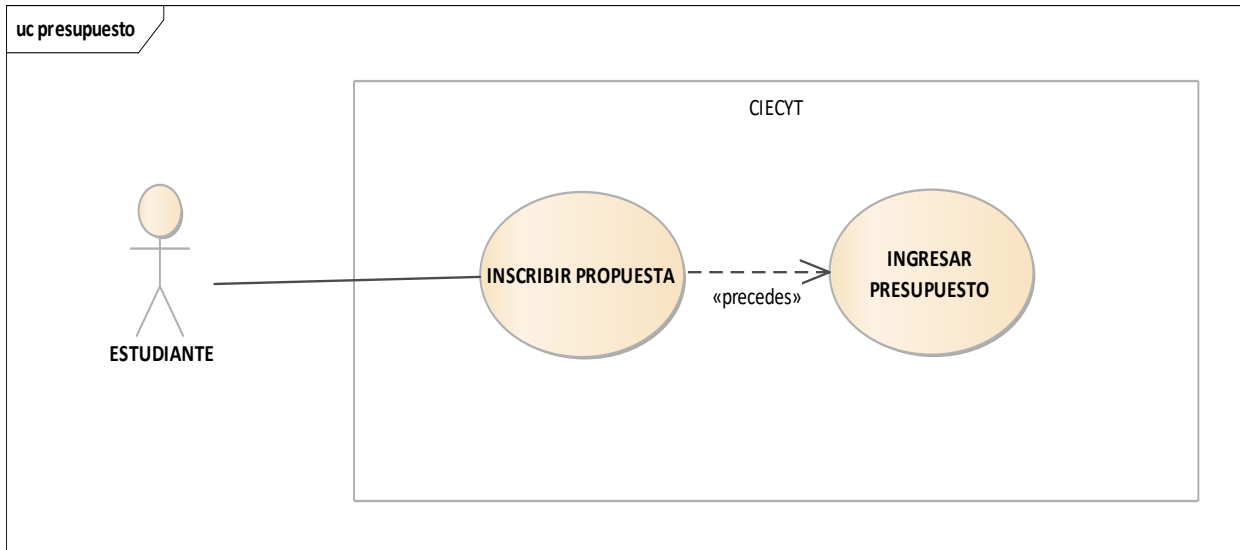
Diagrama 33. Diagrama de secuencia estado de la propuesta



Fuente: Los autores

## Diagrama De Casos De Uso De Presupuesto

Diagrama 34. Caso de uso de presupuesto



Fuente: Los autores

### Curso Básico

Si el estudiante ha inscrito una propuesta tendrá la opción de ingresar el presupuesto, cuando el estudiante da clic al botón presupuesto el sistema muestra la página ingresar presupuesto. El sistema listará los rubros para que el estudiante puede seleccionar uno y llenar los siguientes campos (descripción, justificación, cantidad, valor unitario...); cuando de clic en guardar o cancelar podrá seleccionar otro rubro.

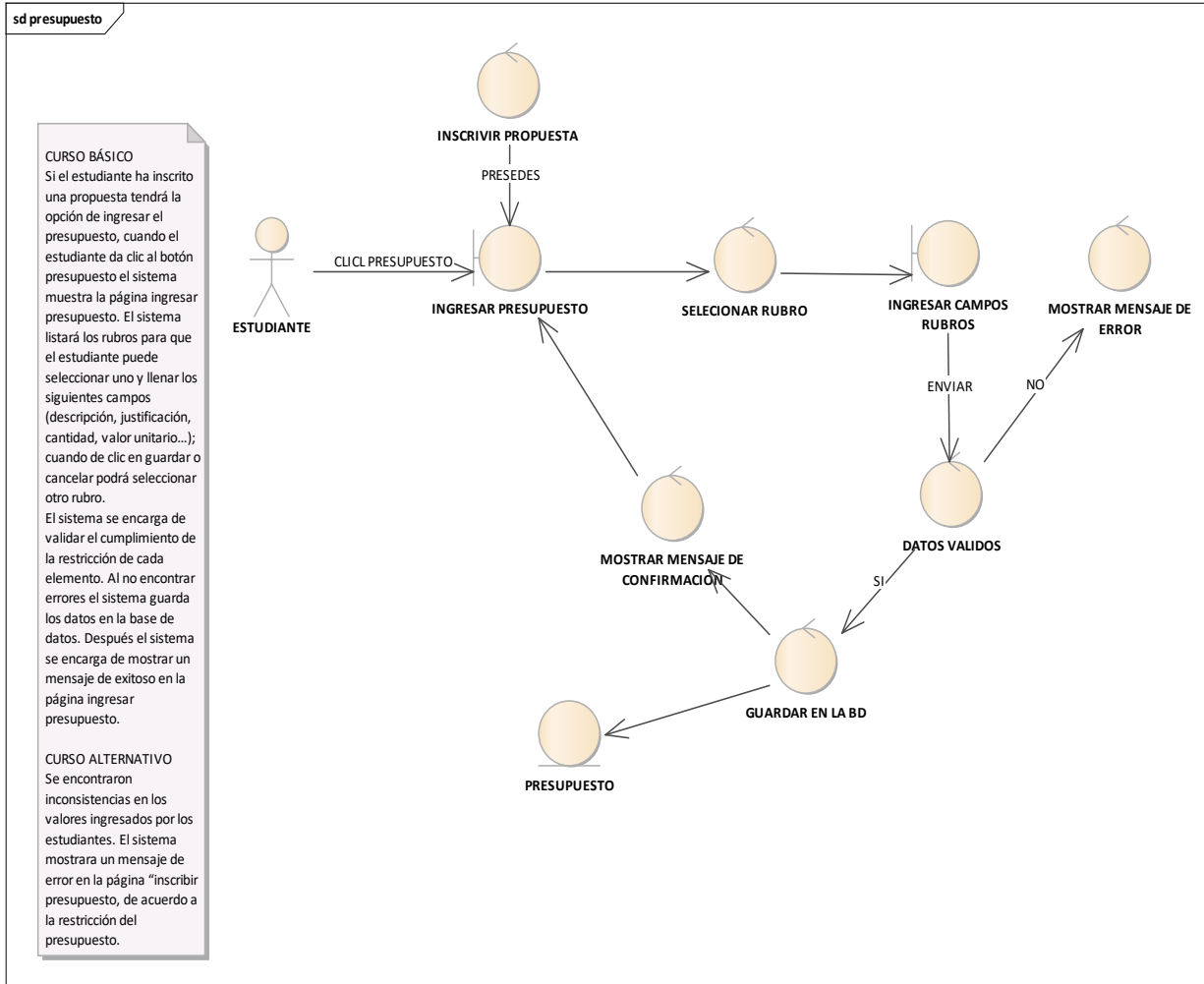
El sistema se encarga de validar el cumplimiento de la restricción de cada elemento. Al no encontrar errores el sistema guarda los datos en la base de datos. Después el sistema se encarga de mostrar un mensaje de exitoso en la página ingresar presupuesto.

### Curso Alternativo

Se encontraron inconsistencias en los valores ingresados por los estudiantes. El sistema mostrara un mensaje de error en la página “inscribir presupuesto, de acuerdo a las restricciones del presupuesto.

# Diagrama De Robustez

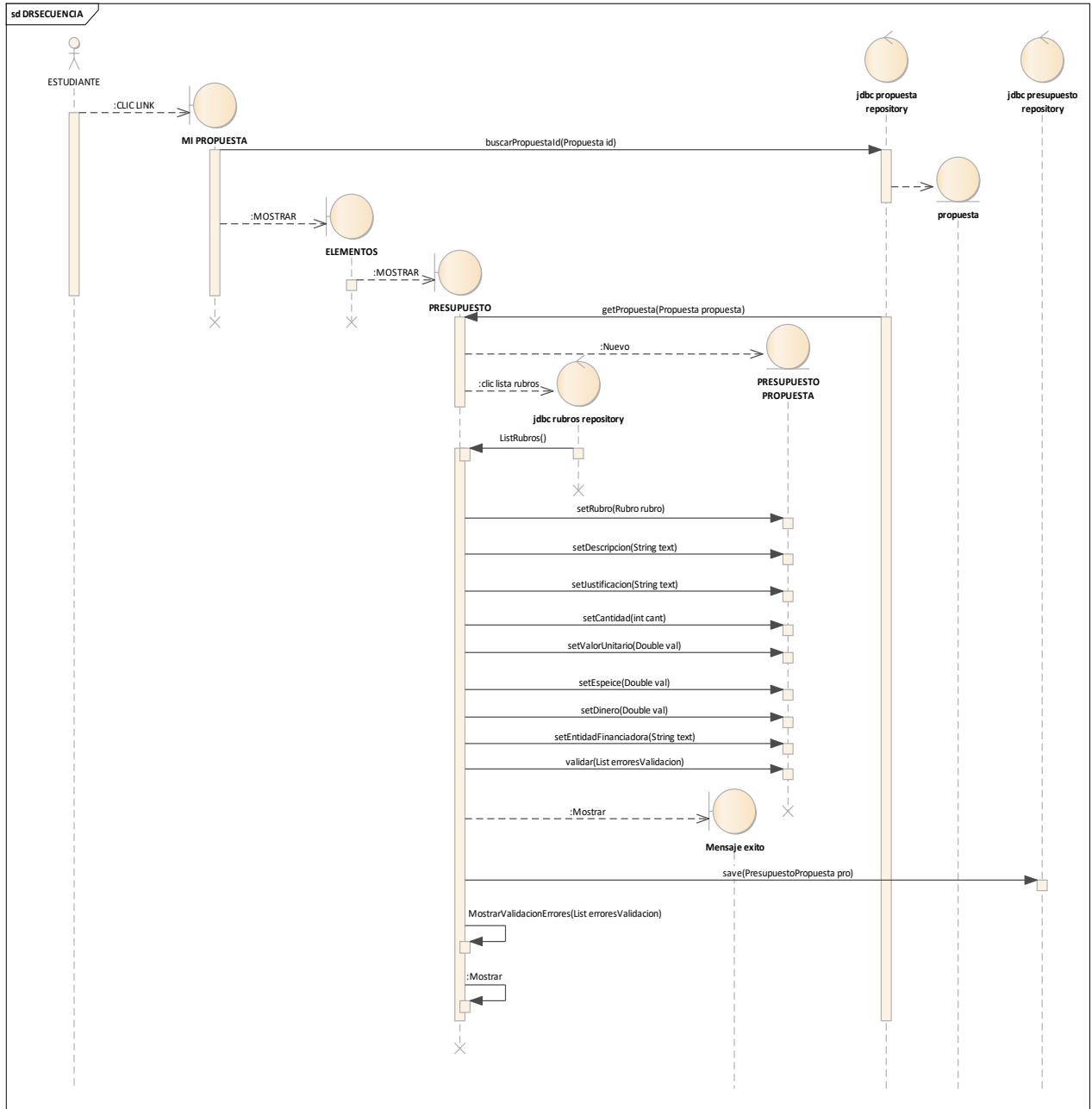
## Diagrama 35 Diagrama de robustez de ingresar presupuesto



Fuente: Los autores

# Diagrama De Secuencia

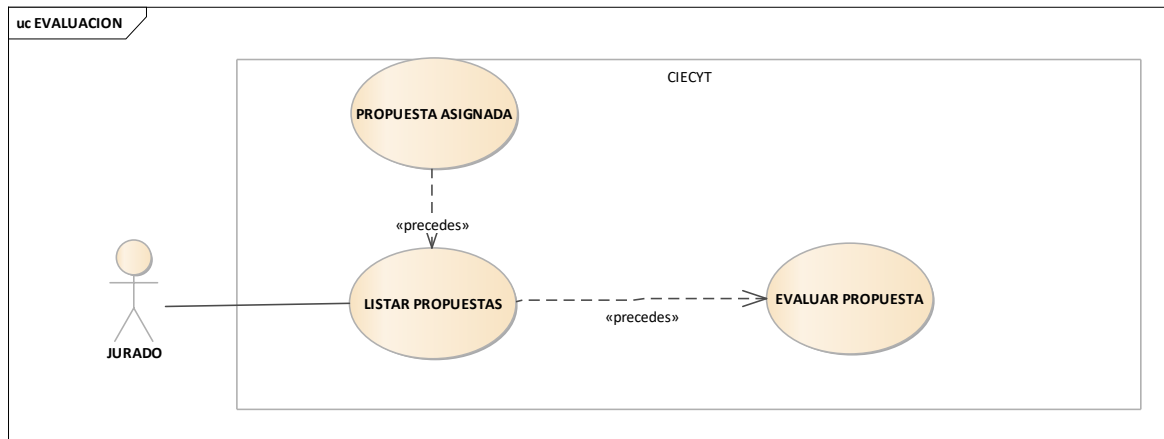
Diagrama 36 diagrama de secuencia de ingresar presupuesto



Fuente: Los autores

## Diagrama De Casos De Uso Del Jurado - Evaluación Propuesta

Diagrama 37. Caso de Uso Evaluación de la propuesta por el jurado



Fuente: Los autores

### Curso Básico

El sistema le permitirá al jurado listar las propuestas que le hayan sido designadas previamente. Cuando el jurado de CLIC en la propuesta se le mostrarán todos los elementos (título, problema, justificación...) si el jurado selecciona un elemento; el sistema le mostrará los siguientes campos: cumple, no cumple o N/A. además le mostrará una pregunta para que compare si el elemento cumple con los parámetros ya establecidos. Cuando el jurado presiona el botón guardar el sistema se encarga de validar que todos los campos estén llenos y también que el número de palabras para las observaciones no supere los límites que se han dado en el estatuto.

Cuando el jurado de clic en enviar el sistema valida que todos los elementos ya hayan sido evaluados por el jurado; si ya están completos el sistema guarda los datos en la base de datos y habilita la opción categorizar; cuando el jurado de clic en el botón categorizar se mostrarán tres campos: viabilizar, pendiente y no viable.

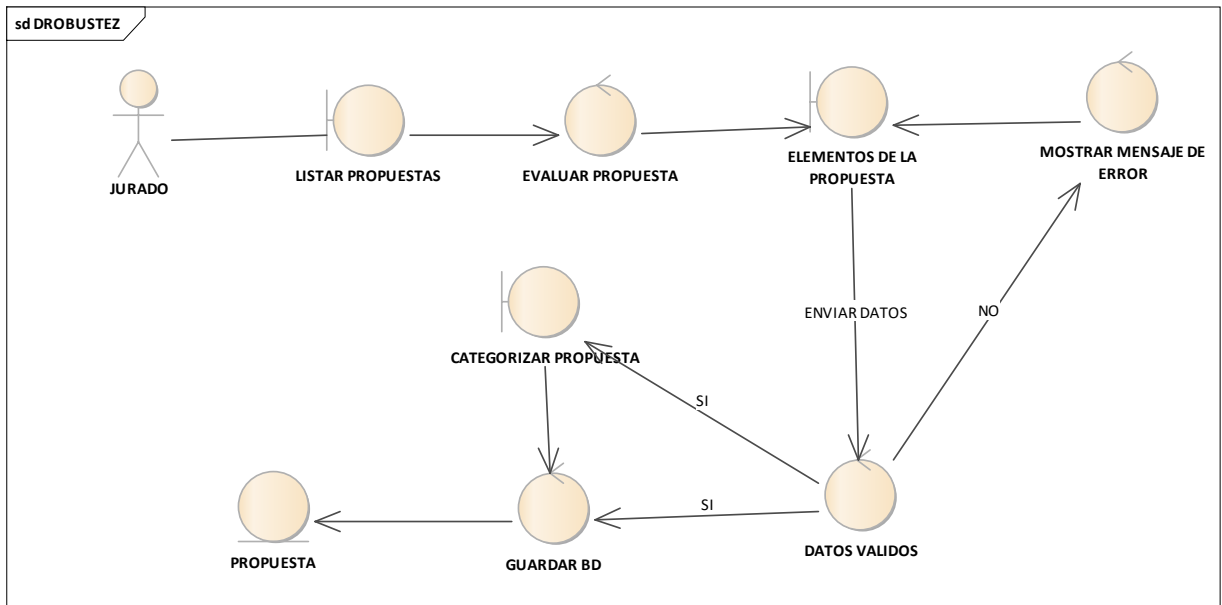
Después de que el jurado haya categoriza la propuesta el sistema le mostrará los resultados al estudiante el cual podrá realizar correcciones si son solicitadas por el jurado y reenviar la propuesta para que el jurado revise las correcciones.

## Curso Alternativo

Si se encontraron inconsistencias en los valores ingresados por los jurados, o campos obligatorios vacíos. El sistema mostrara un mensaje de error en la página “evaluar propuesta, de acuerdo a las restricciones de la evaluación.

### **Diagrama De Robustez**

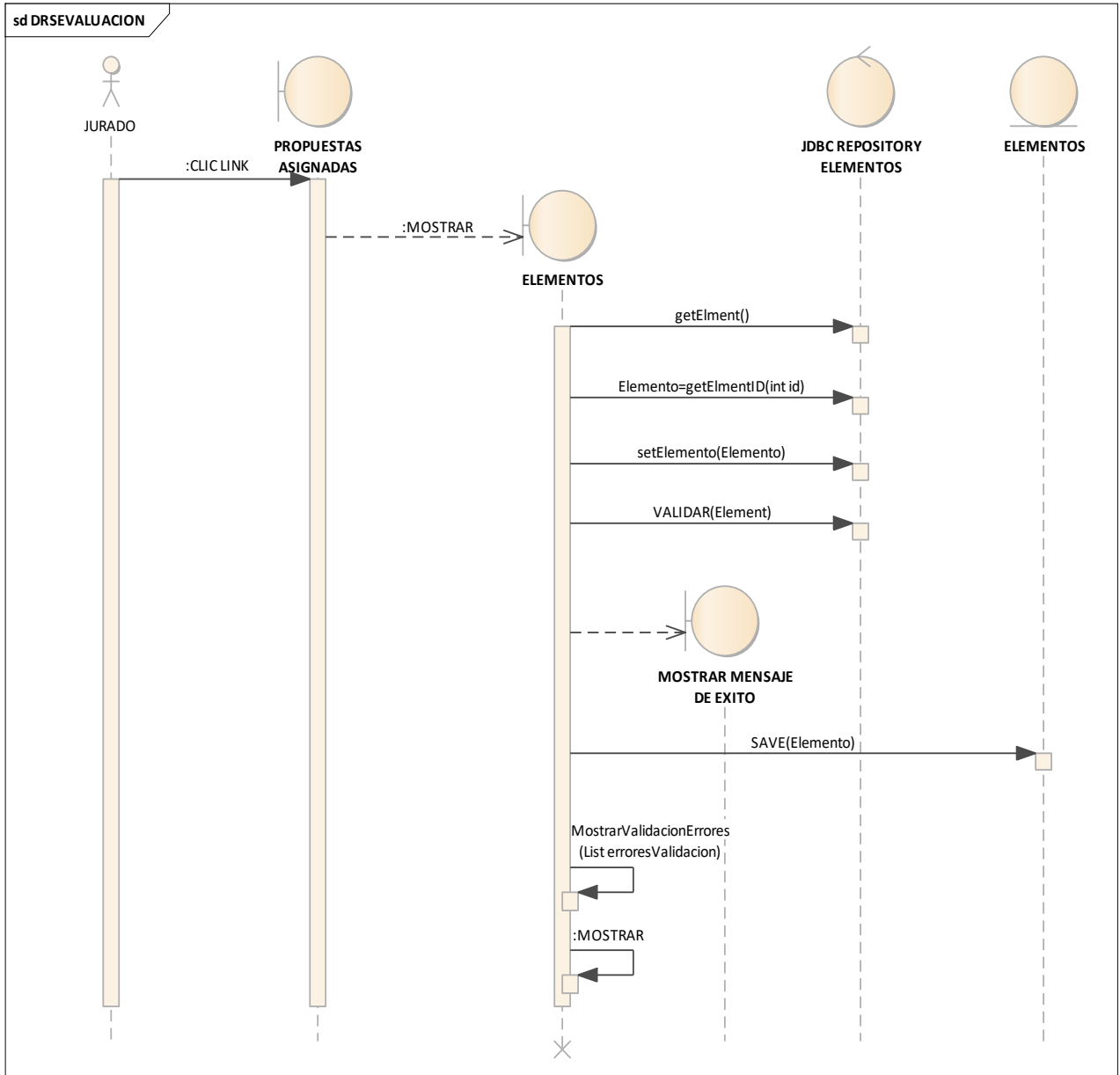
Diagrama 38. Diagrama de Robustez evaluación de la propuesta



Fuente: Los autores

## Diagrama De Secuencia

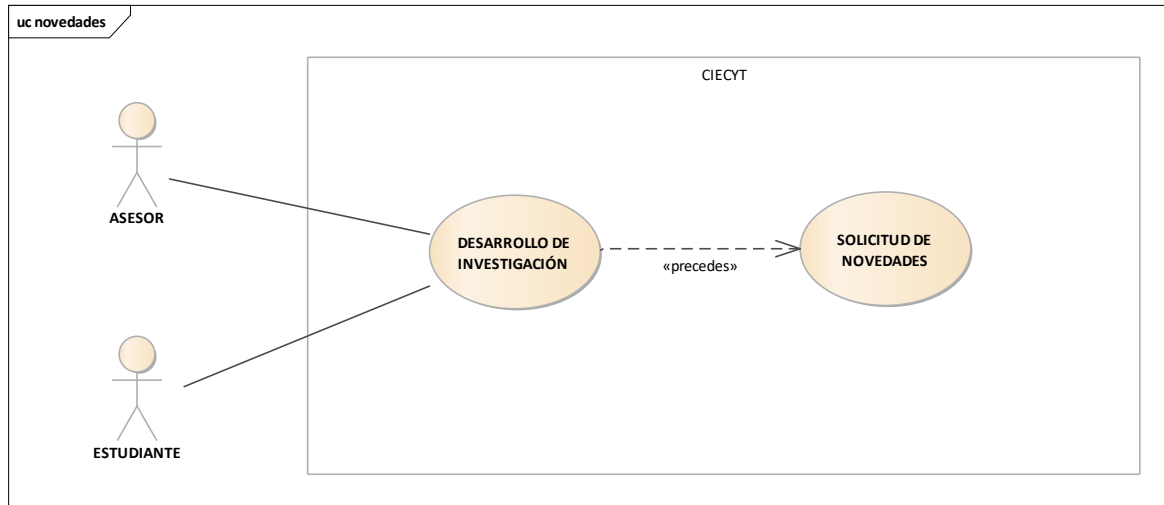
Diagrama 39. Diagrama de secuencia evaluación de la propuesta



Fuente: Los autores

## Solicitud De Novedades Diagrama De Casos De Uso

Diagrama 40. Caso de uso solicitud de novedades



Fuente: Los autores

### Curso Básico

El sistema le permitirá al jurado listar las tesis y el estado de cada una de ellas; al dar clic en las investigaciones que estén listas para evaluación, el sistema le mostrará una ficha técnica en la cual deberá llenar los campos (nombre, cedula, universidad...), además se mostrara un campo en el que puede se le solicita los datos de un evaluador experto en el tema del proyecto que se está evaluando

También el sistema le mostrara el botón SIGUIENTE para evaluar cada uno de los elementos de la investigación; cada vez que el jurado evaluador de CLIC en siguiente el sistema validara si los campos obligatorios están llenos para permitirle seguir con la evolución del siguiente elemento.

Para cada elemento el sistema le mostrara una pregunta para que compare el elemento si cumple con los parámetros ya establecidos y dos campos, en algunos casos, uno es para darle un valor numérico y el otro para mostrar el valor máximo que se le puede asignar al elemento; en los otros casos le mostrara las opciones SI y NO para valorar si el elemento cumple o no cumple los parámetros.

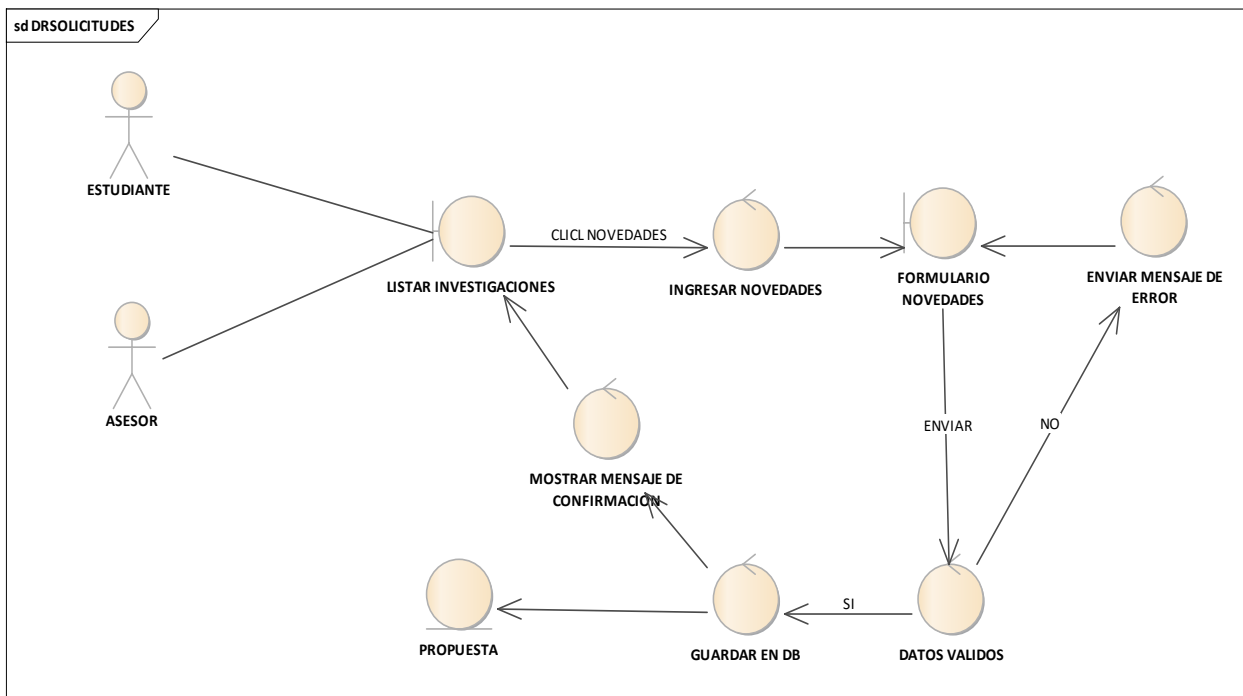
La finalizar los elementos el sistema le mostrar el campo observaciones, para que el jurado haga las observaciones de ser necesario.

## Curso Alternativo

Si se encontraron inconsistencias en los valores ingresados por los jurados, o campos obligatorios vacíos. El sistema mostrara un mensaje de error en la página “evaluar tesis”.

### Diagrama De Robustez

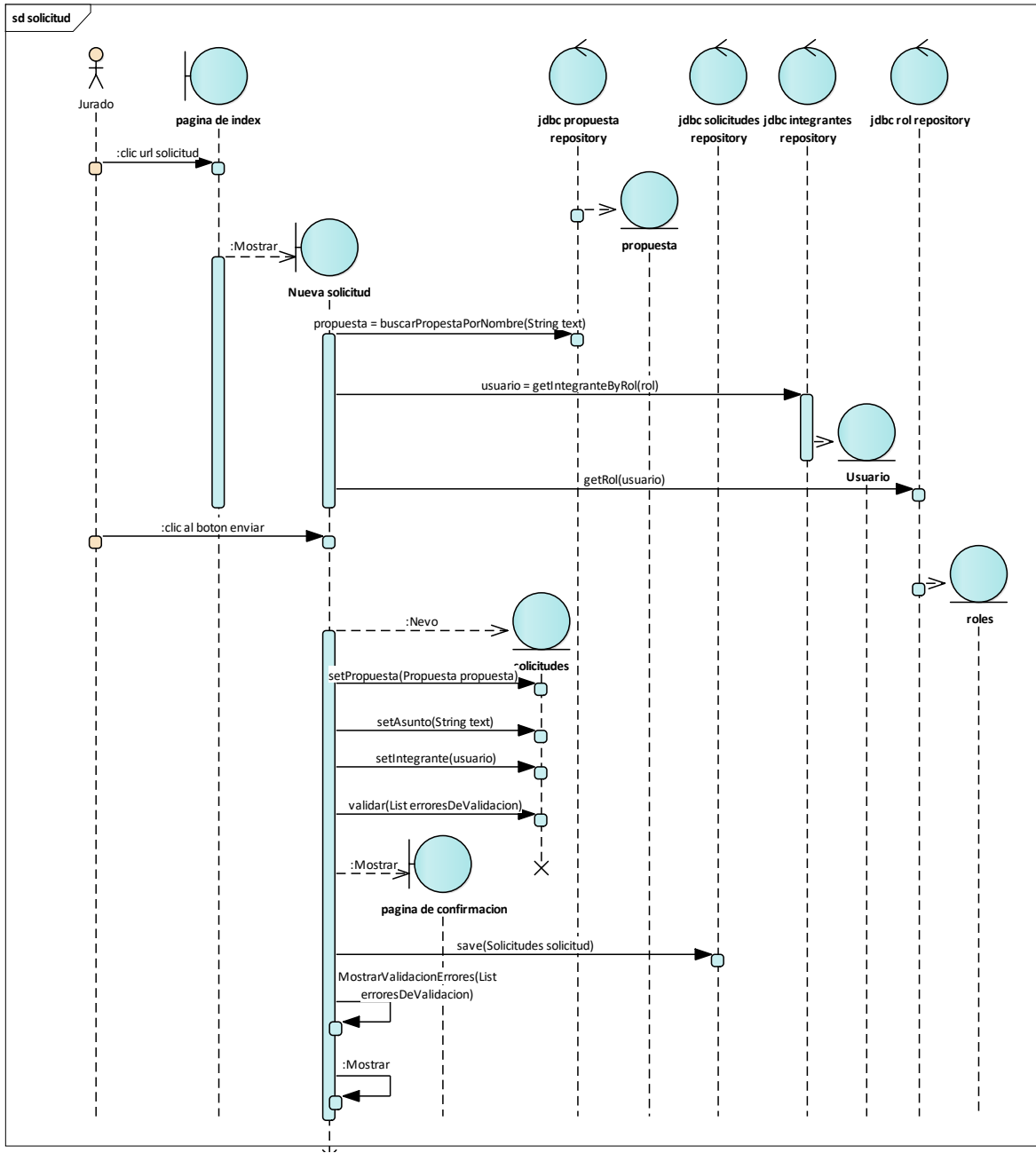
Diagrama 41. Diagrama de robustez solicitud de novedad



Fuente: Los autores

# Diagrama de secuencia

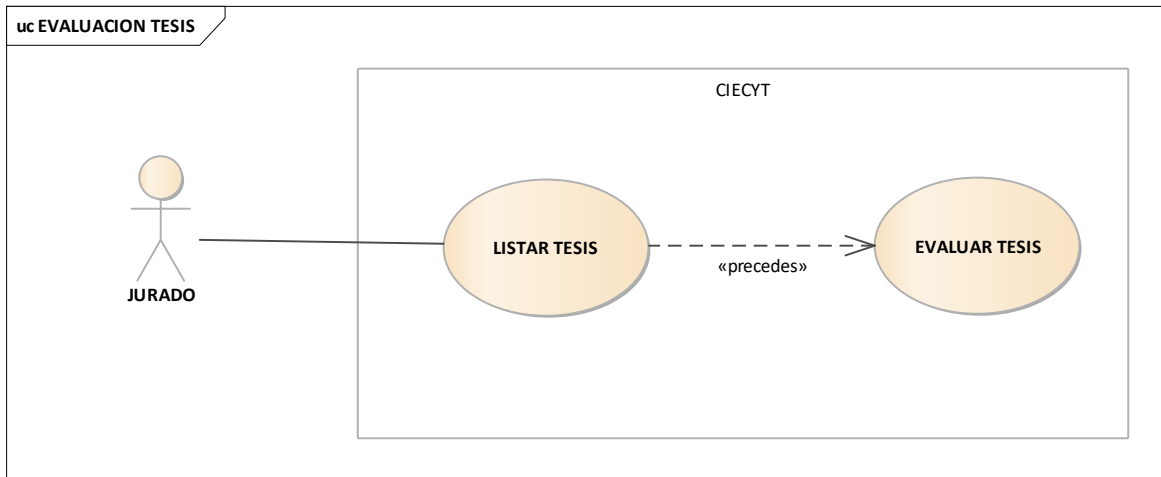
Diagrama 42. Diagrama de secuencia solicitud de novedad



Fuente: Los autores

## Diagrama De Casos De Uso Viabilizar Para Sustentar

Diagrama 43. Caso de uso viabilidad para sustentación



Fuente: Los autores

### Curso Básico

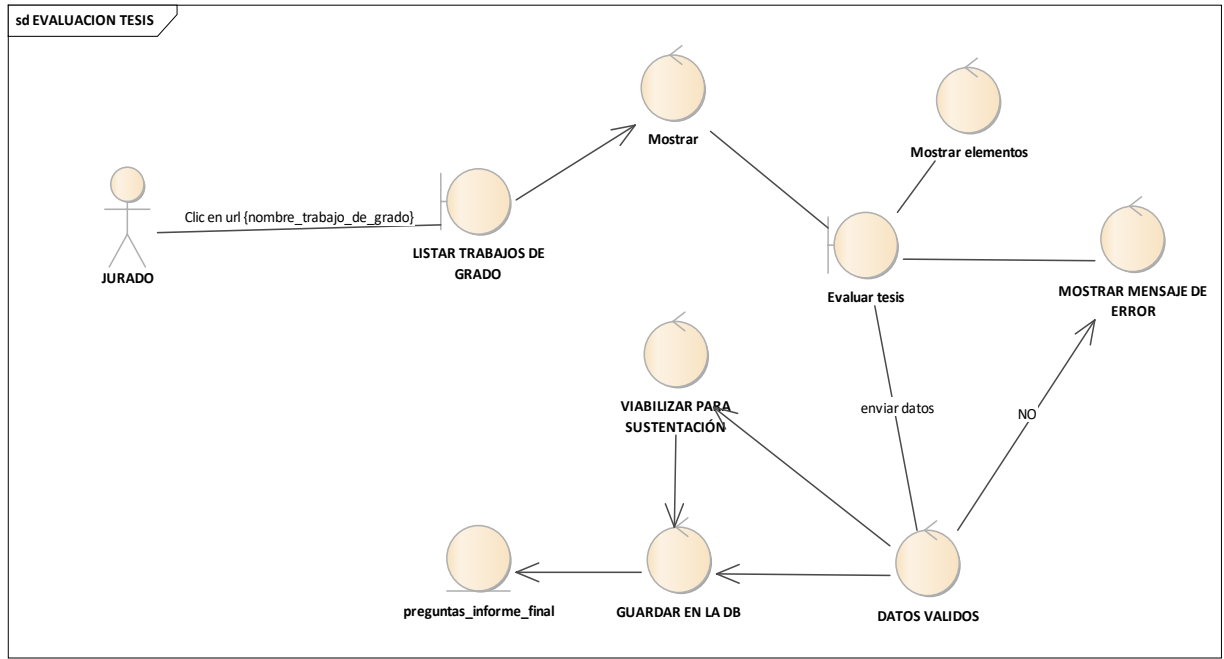
Cuando el Jurado da clic a la URL “propuesta” el sistema se encarga de listar todas las propuestas que tenga asignadas, cuando el jurado da clic a una propuesta el sistema en una nueva página desplegará todos los elementos de la propuesta, pero aparecerá por defecto el elemento título, en donde el sistema mostrara: el valor del elemento, una pregunta de ayuda y un campo para ingresar una observación. Cuando el asesor guarda los cambios el sistema se encarga de verificar que el campo observación no este vacío después muestra un mensaje de confirmación y guarda los cambios en la base de datos. Si el Jurado da clic a cualquier otro elemento tales como: objetivos, cronograma de actividades, presupuesto entre otros; el proceso a seguir será igual al mencionado anteriormente.

### Curso Alternativo

Si el Jurado deja el campo observación vacío el sistema se encargará de mostrar un mensaje de error.

## Diagrama De Robustez

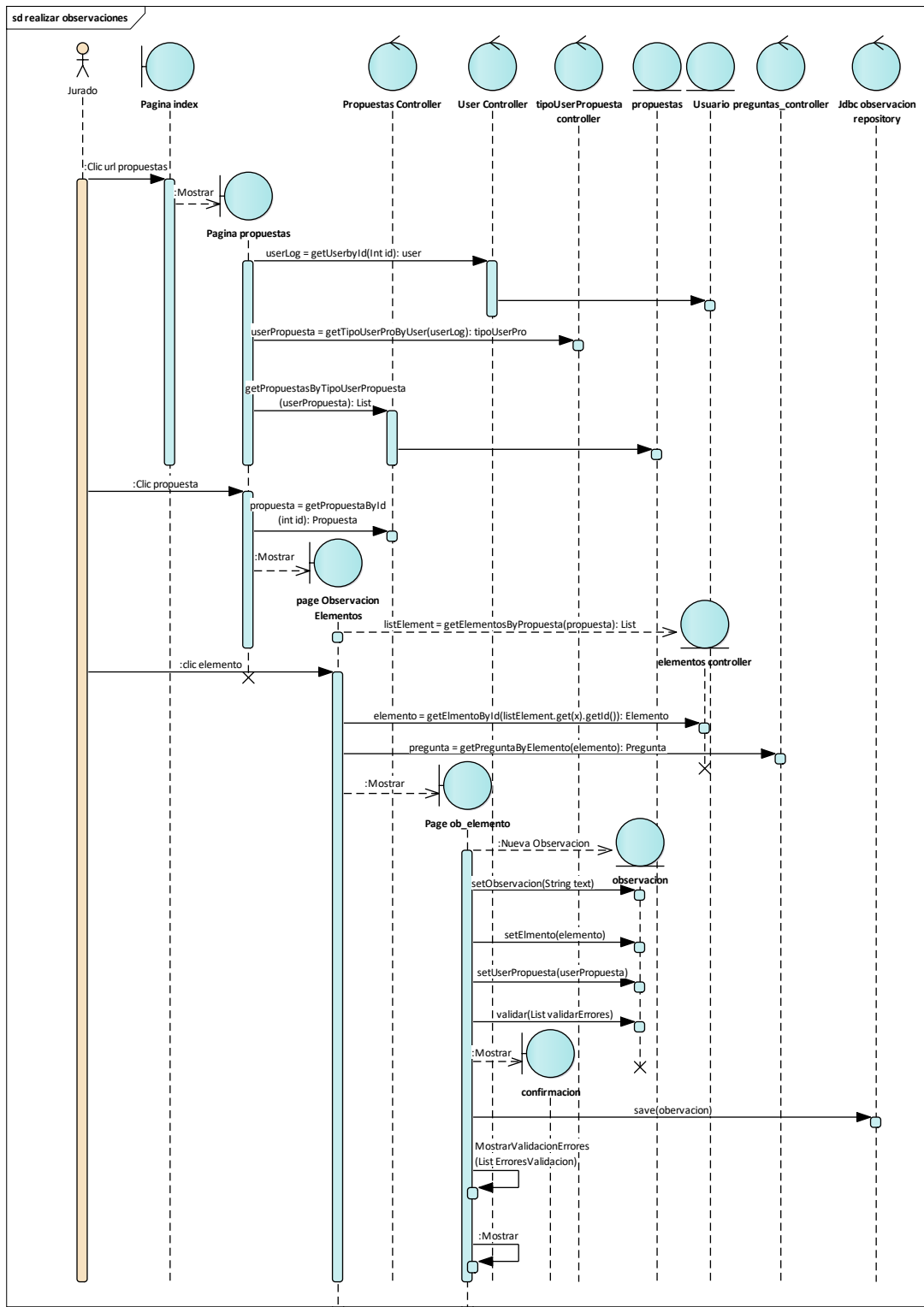
Diagrama 44. Diagrama de robustez viabilizar para sustentar



Fuente: Los autores

# Diagrama De Secuencia

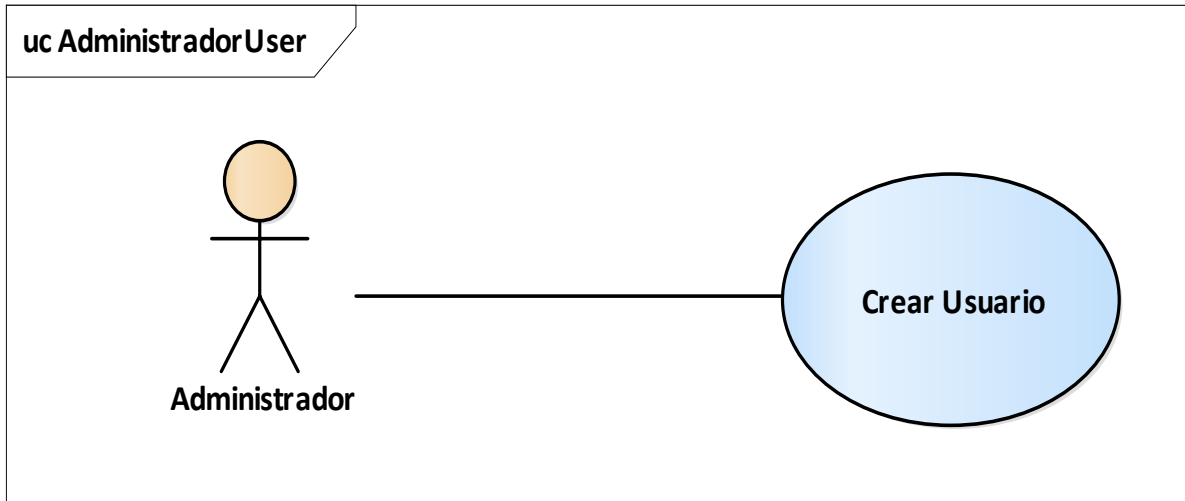
Diagrama 45. Diagrama de Secuencia viabilizar para sustentar



Fuente: Los autores

## Casos De Uso Del Administrador

Diagrama 46. Casos de uso del administrador



Fuente: Los autores

### Descripción del caso de uso: CREAR USUARIO

#### Curso Básico

El sistema permitirá al administrador crear un usuario a partir de un formulario donde deberá llenar los siguientes datos: Identificación, nombres, apellidos, email, estado Usuario, el tipo de usuario se mostrará en una lista seleccionable, fecha de expedición, lugar de expedición.

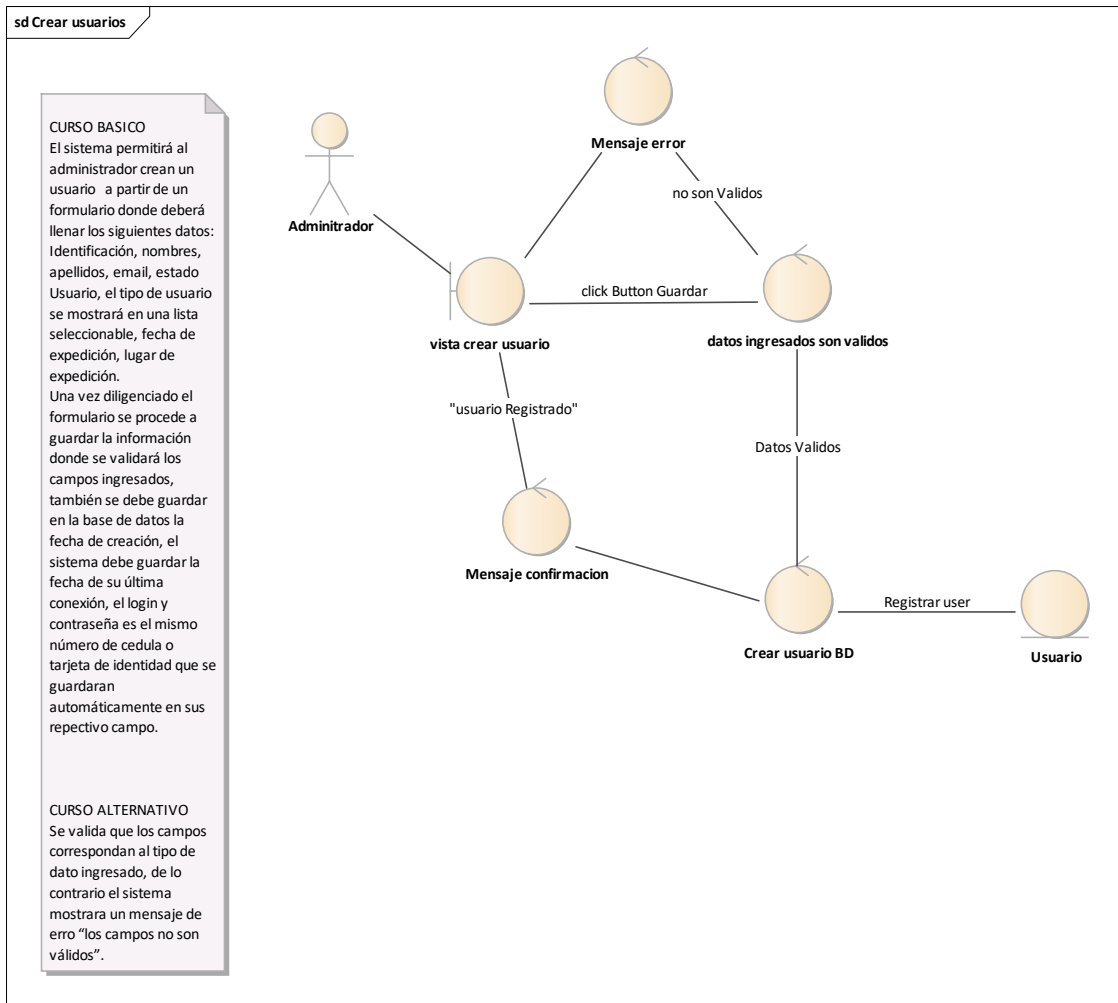
Una vez diligenciado el formulario se procede a guardar la información donde se validará los campos ingresados, también se debe guardar en la base de datos la fecha de creación, el sistema debe guardar la fecha de su última conexión, el login y contraseña es el mismo número de cedula o tarjeta de identidad que se guardaran automáticamente.

#### Curso Alternativo

Se valida que los campos correspondan al tipo de dato ingresado, de lo contrario el sistema mostrara un mensaje de error “los campos no son válidos”.

Diagrama de robustez

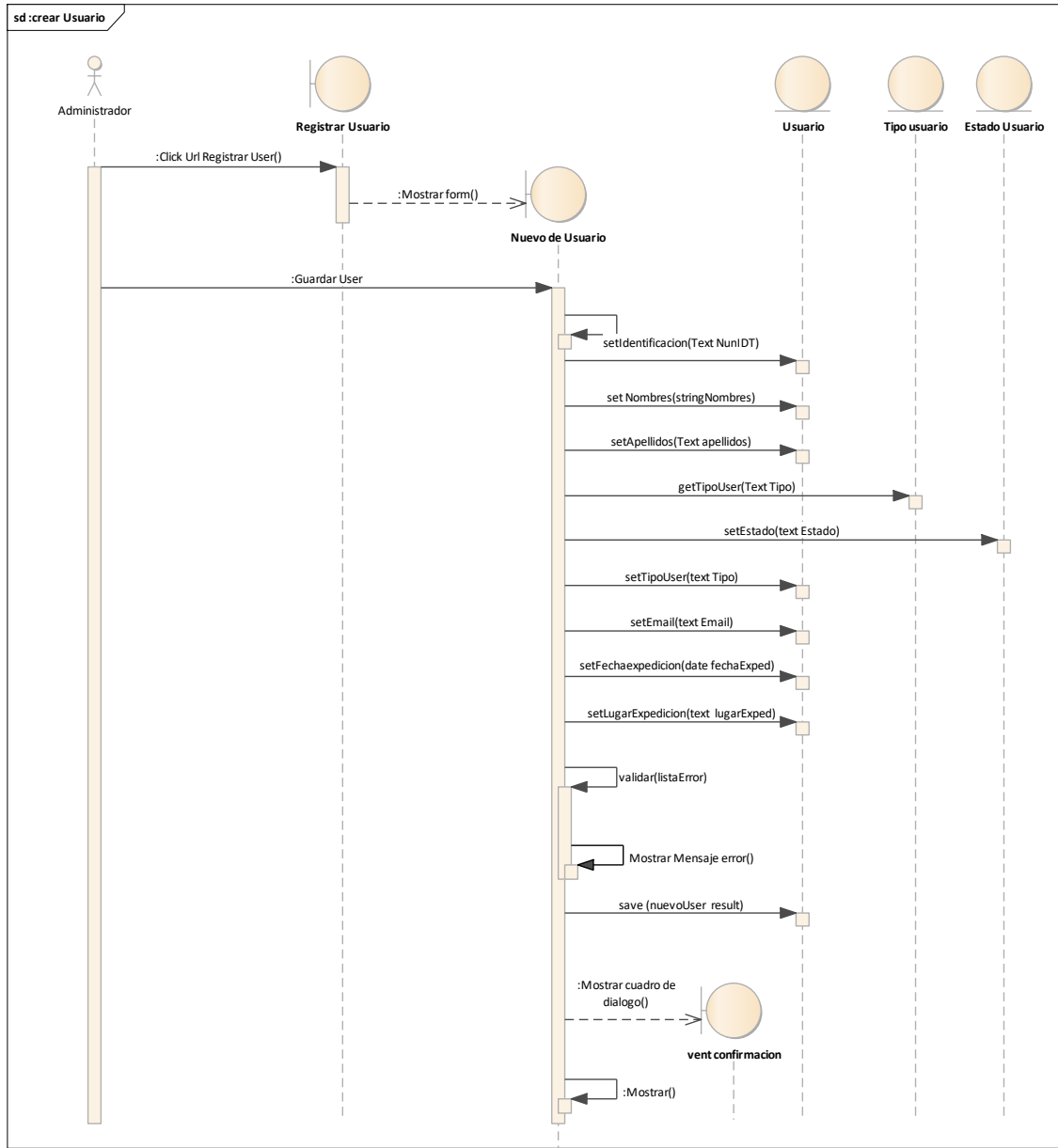
Diagrama 47. Diagrama de robustez crear usuario



Fuente: Los autores

## Diagrama de secuencia

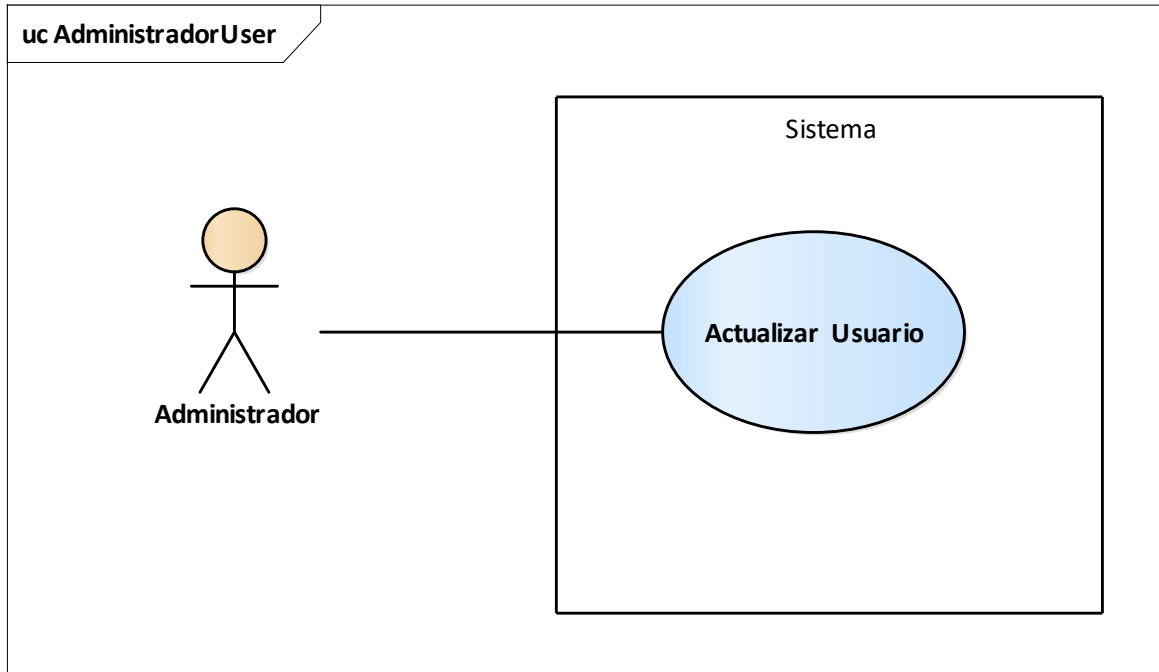
Diagrama 48. Diagrama de secuencia crear usuario



Fuente: Los autores

## Caso de uso actualizar usuario

Diagrama 49. Caso de uso actualizar usuario



Fuente: Los autores

### Descripción del caso de uso: actualizar Usuario

#### Curso Básico

El sistema le permite al administrador actualizar los datos de los usuarios, los cuales se encuentra listados en una tabla, debe permitirle la búsqueda de algún usuario en específico, puede seleccionar el nombre de usuario, se desplegará una ventana con los datos del usuario, debe seleccionar campo que desea modificar.

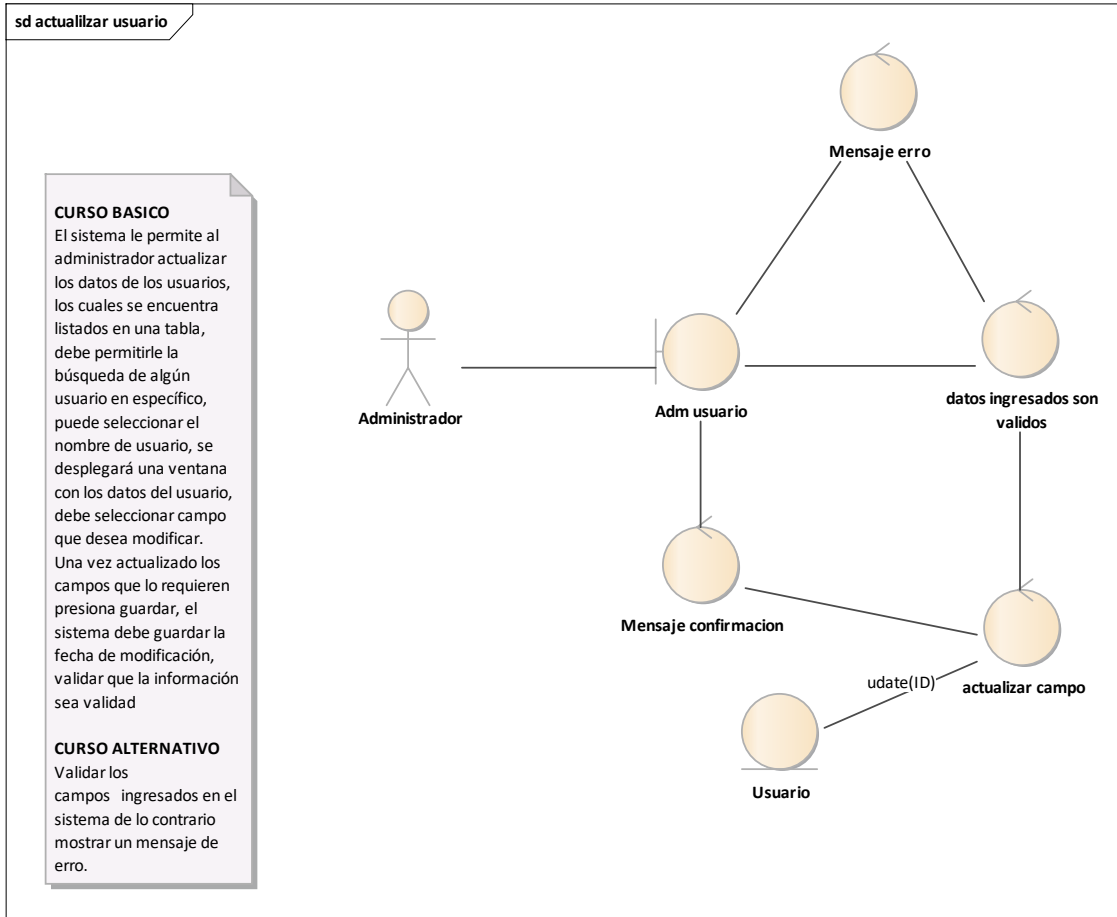
Una vez actualizado los campos que lo requieren presiona guardar, el sistema debe guardar la fecha de modificación, validar que la información sea valida

#### **Curso Alternativo**

Validar los campos ingresados en el sistema de lo contrario mostrar un mensaje de error.

## Diagrama de robustez

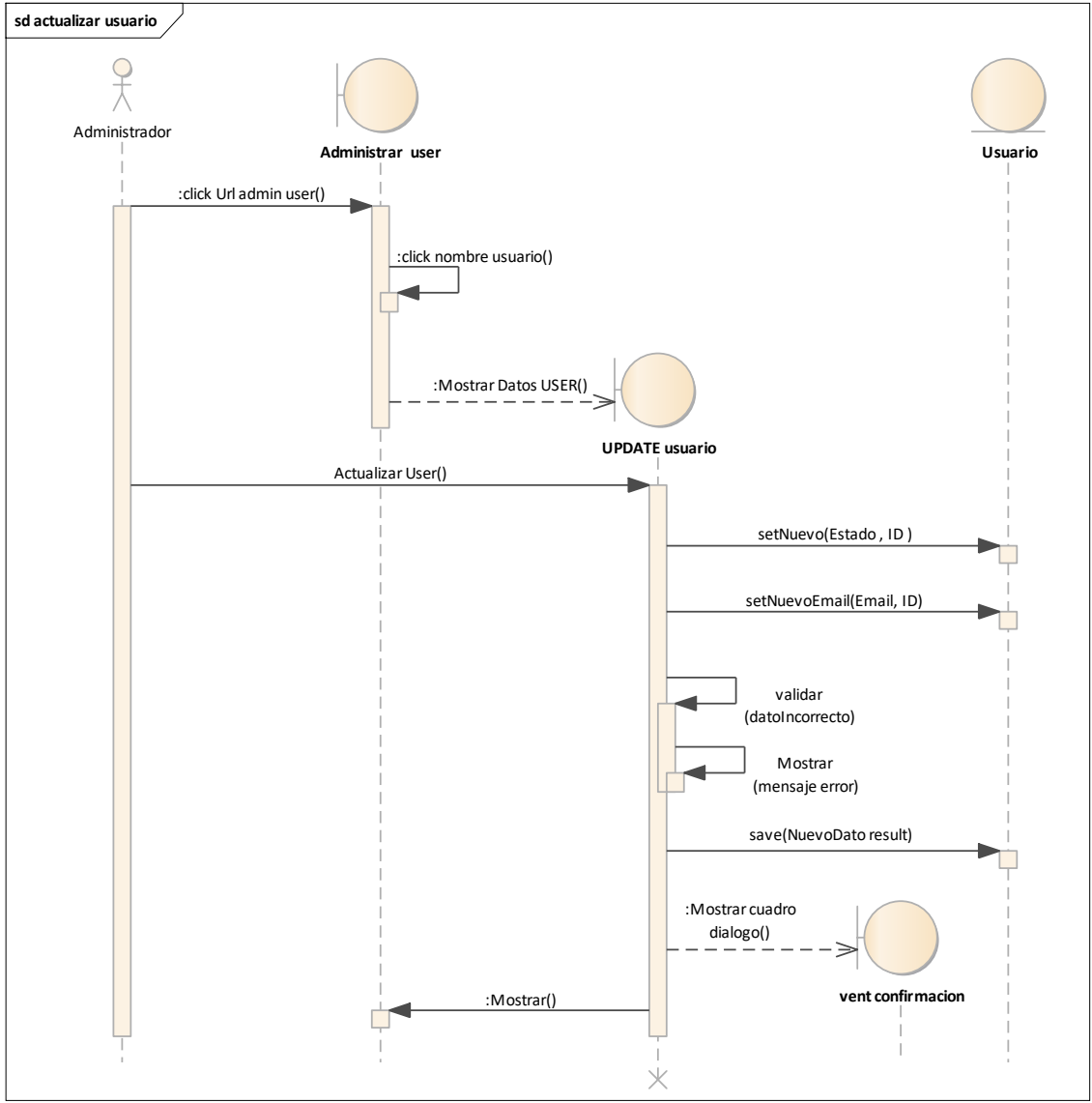
Diagrama 50. Diagrama de robustez actualizar usuario



Fuente: Los autores

## Diagrama de secuencia

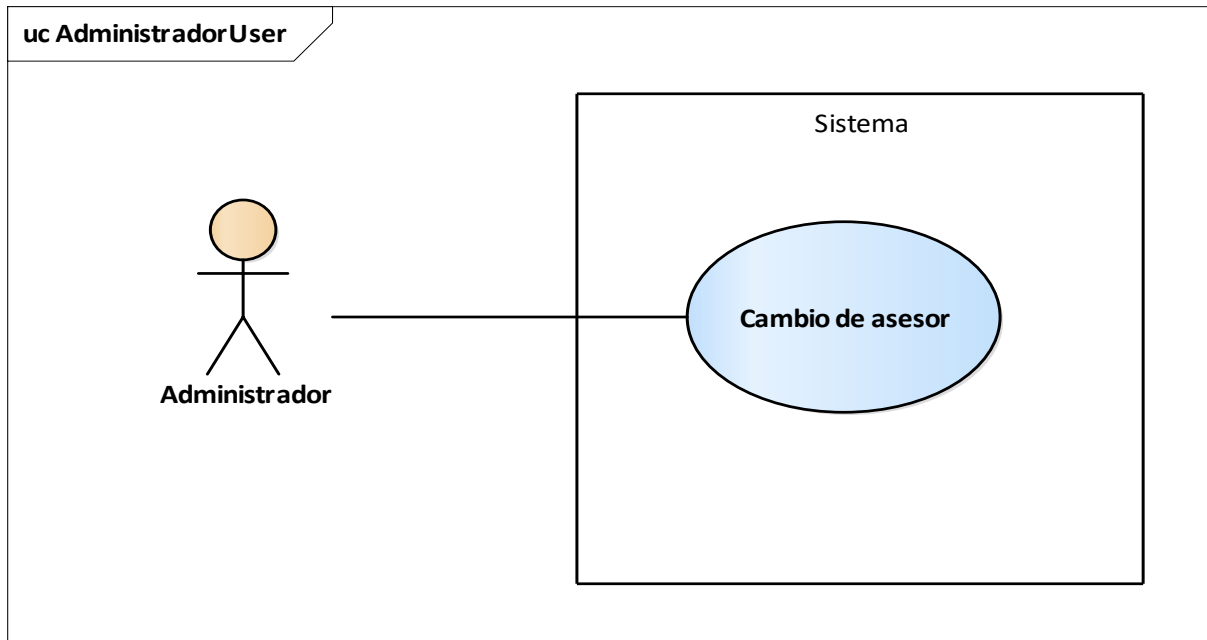
Diagrama 51. Diagrama de secuencia



Fuente: Los autores

## Caso de uso cambio de asesor

Diagrama 52. Caso de uso cambio de asesor



Fuente: Los autores

### Descripción de case de uso: solicitud de cambio de asesor

#### Curso Básico

El sistema permitirá al administrador hacer el cambio de asesor mediante una ventana con los siguientes datos: ID, nombre del nuevo asesor.

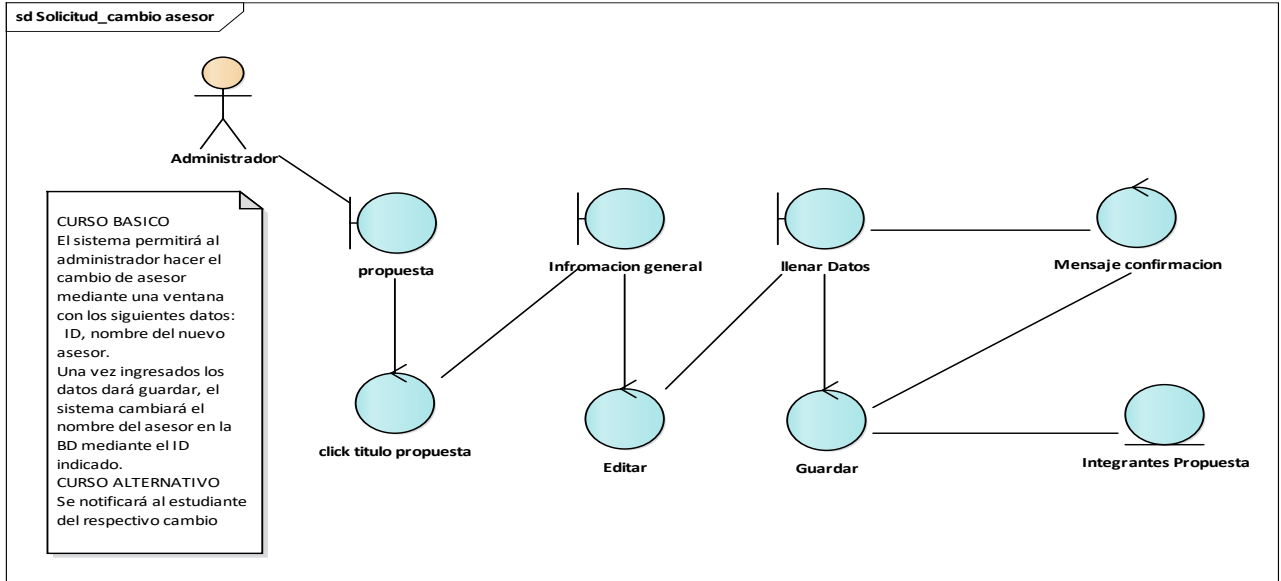
Una vez ingresados los datos dará guardar, el sistema cambiará el nombre del asesor en la base de datos mediante el ID indicado.

#### Curso Alternativo

Se notificará al estudiante del respectivo cambio

## Diagrama de robustez

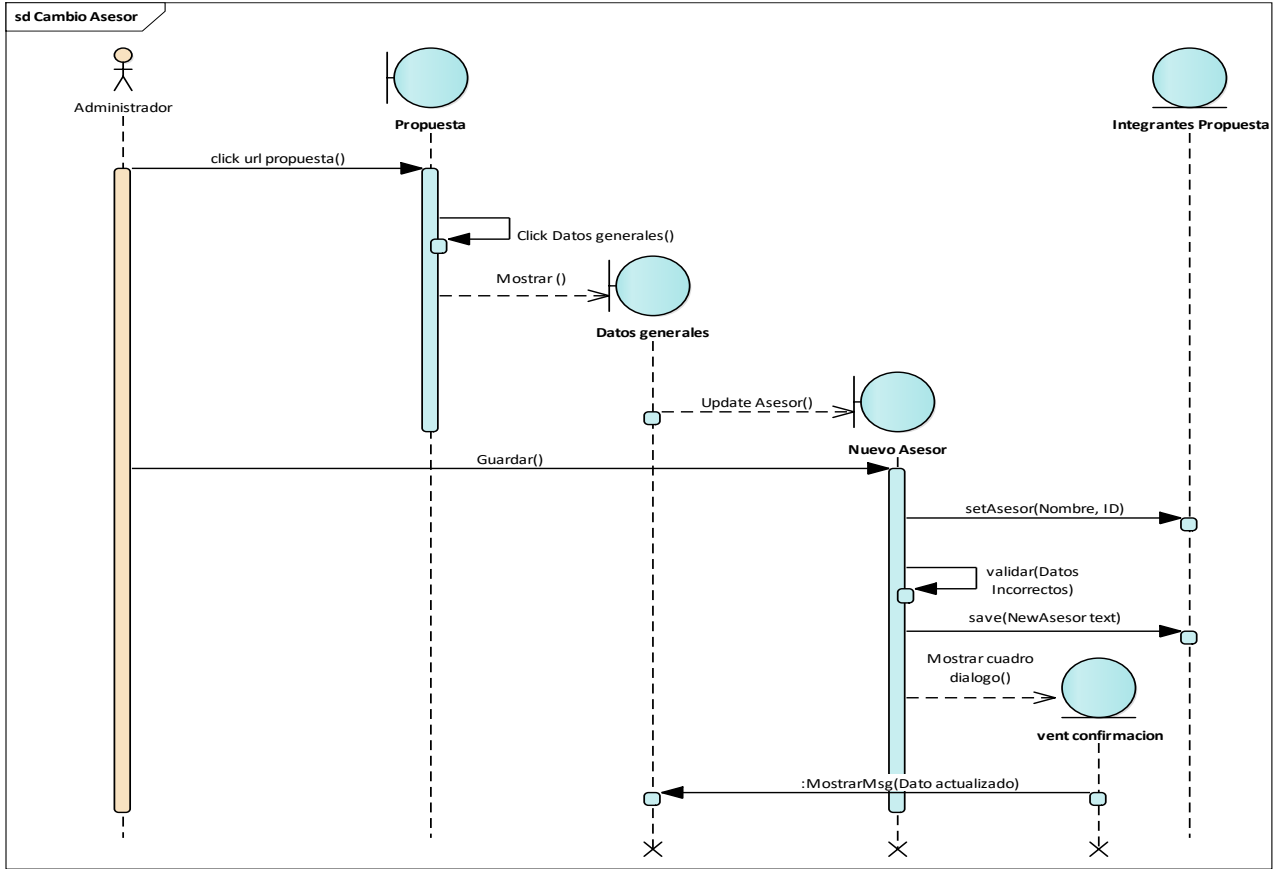
Diagrama 53. Diagrama de robustez



Fuente: Los autores

# Diagrama de secuencia

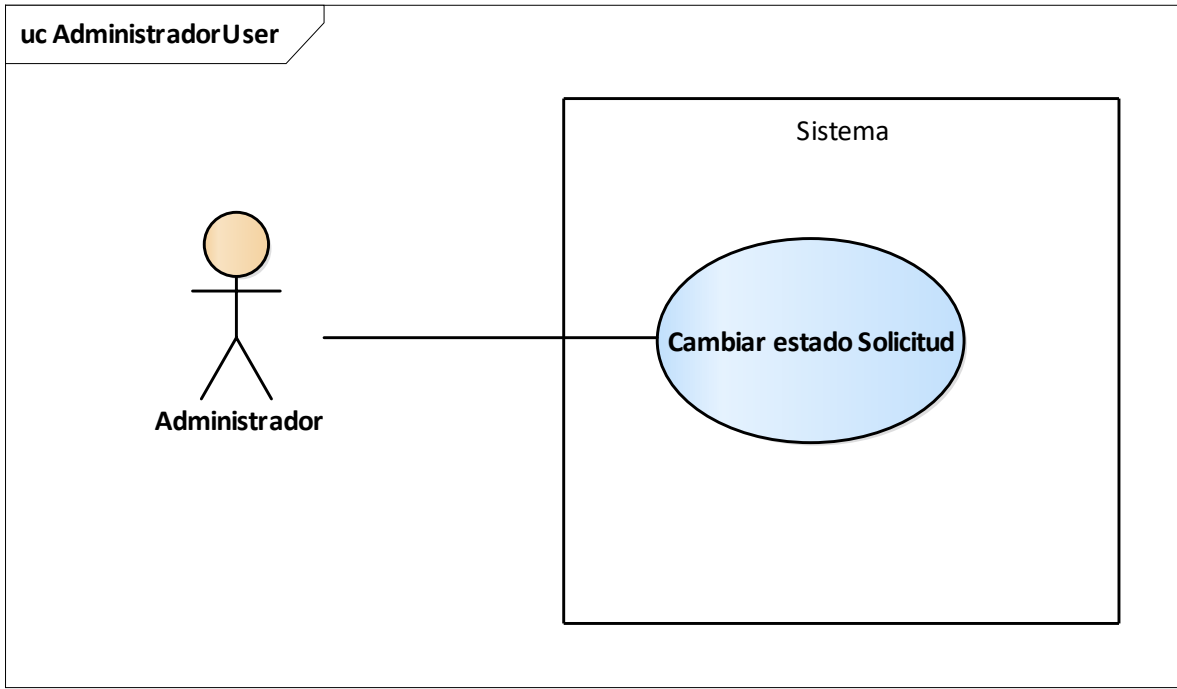
Diagrama 54. Diagrama de secuencia



Fuente: Los autores

## Caso de uso estado solicitud

Diagrama 55. Caso de uso estado de solicitud



Fuente: Los autores

### Descripción de case de uso: cambiar estado solicitud

#### Curso Básico

El sistema permitirá al administrador visualizar la solicitud de cambio por parte del Asesor donde se mostrará los siguientes datos: fecha, Asunto, integrantes, además podrá dar una respuesta positiva o negativa mediante las dos opciones APROBADO, RECHAZADO.

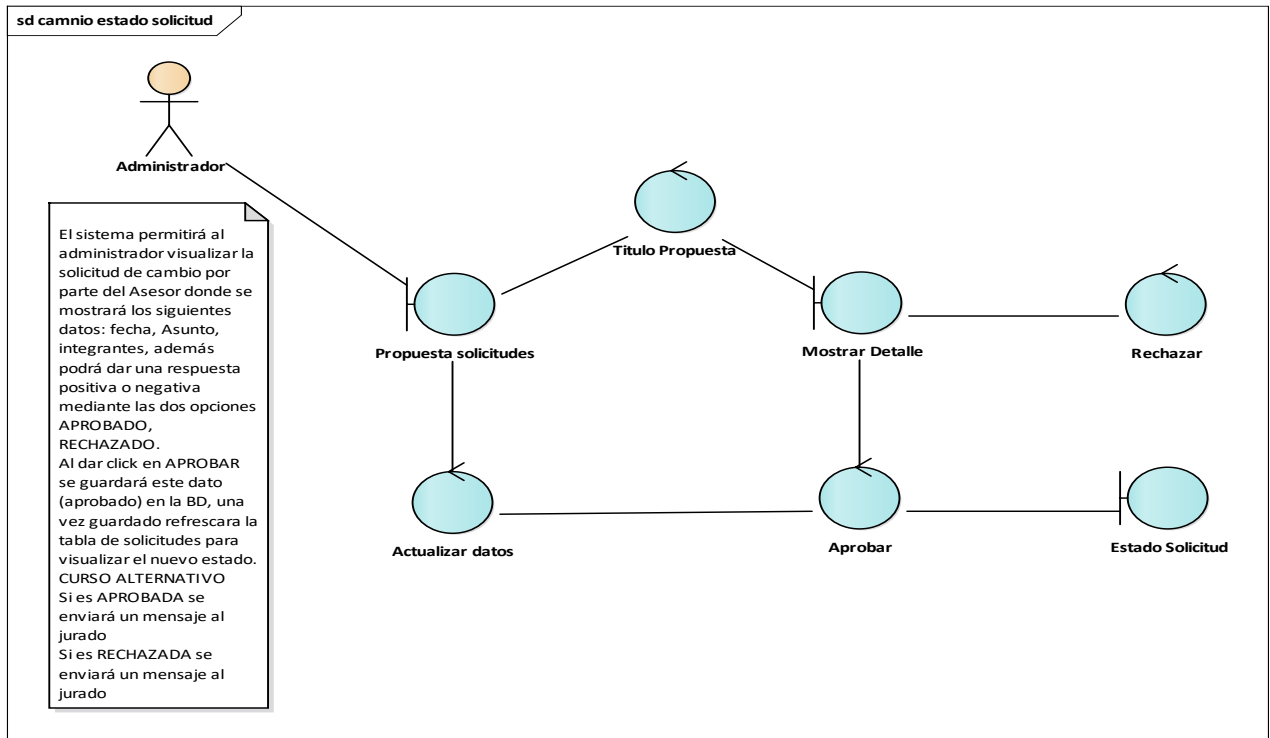
Al dar click en APROBAR se guardará este dato (aprobado) en la base de datos, una vez guardado refrescará la tabla de solicitudes para visualizar el nuevo estado.

#### Curso Alternativo

Si es APROBADA se enviará un mensaje al jurado, Si es RECHAZADA se enviará un mensaje al jurado

## Diagrama de robustez

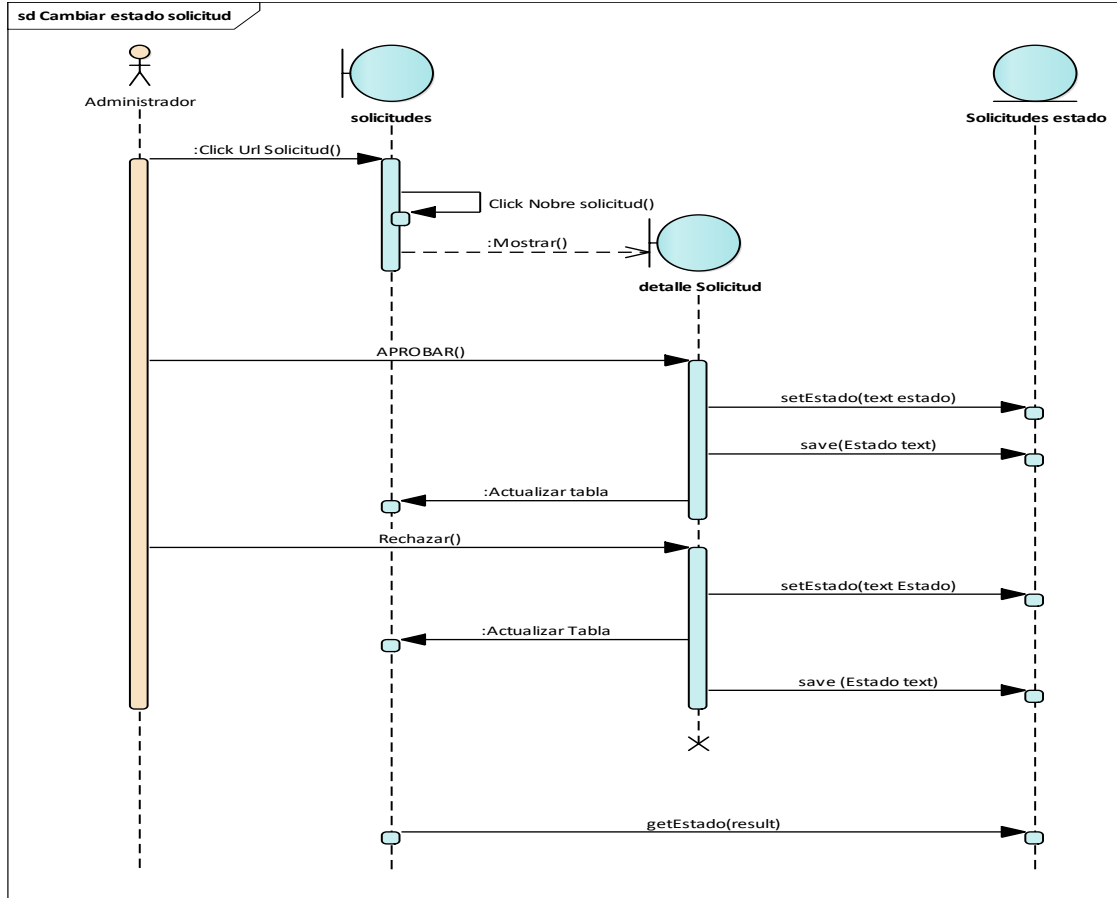
Diagrama 56. Diagrama de robustez



Fuente: Los autores

# Diagrama de secuencia

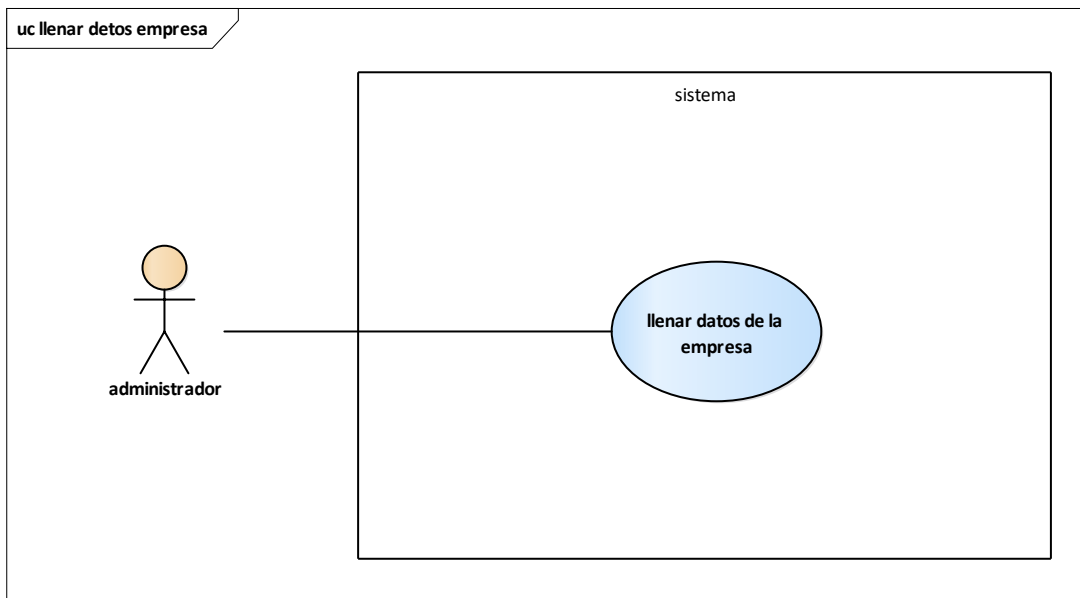
Diagrama 57. Registrar datos empresa



Fuente: Los autores

## Caso de uso registrar empresa

Diagrama 58 .Caso de Uso Registrar Datos de Empresa



### Curso Básico

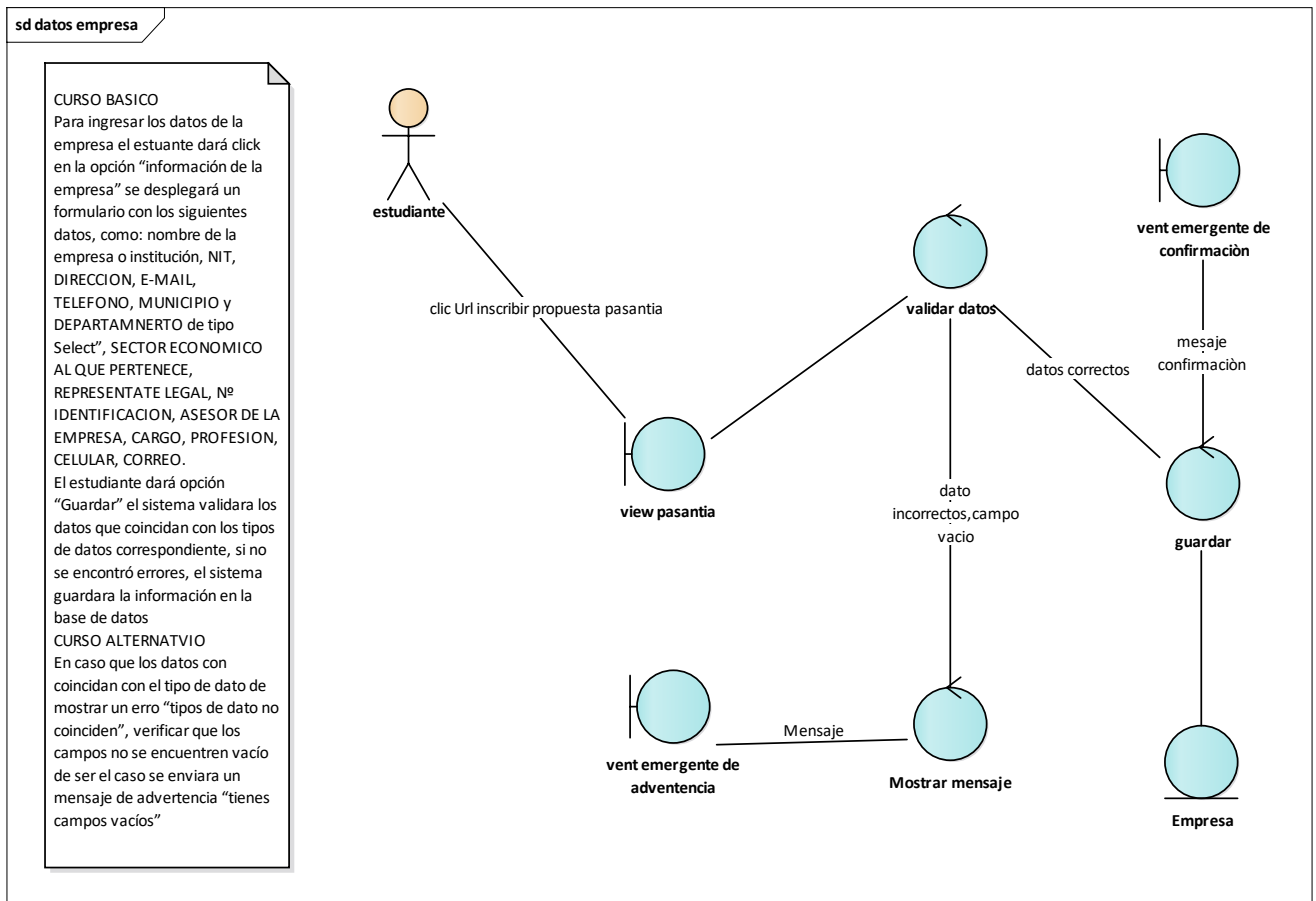
Para ingresar los datos de la empresa el estuante dará click en la opción “información de la empresa” se desplegará un formulario con los siguientes datos, como: nombre de la empresa o institución, NIT, DIRECCION, E-MAIL, TELEFONO, MUNICIPIO y DEPARTAMNERTO de tipo Select”, SECTOR ECONOMICO AL QUE PERTENECE, REPRESENTATE LEGAL, N° IDENTIFICACION, ASESOR DE LA EMPRESA, CARGO, PROFESION, CELULAR, CORREO.

El estudiante dará opción “Guardar” el sistema validara los datos que coincidan con los tipos de datos correspondiente, si no se encontró errores, el sistema guardara la información en la base de datos

## Curso Alternativo

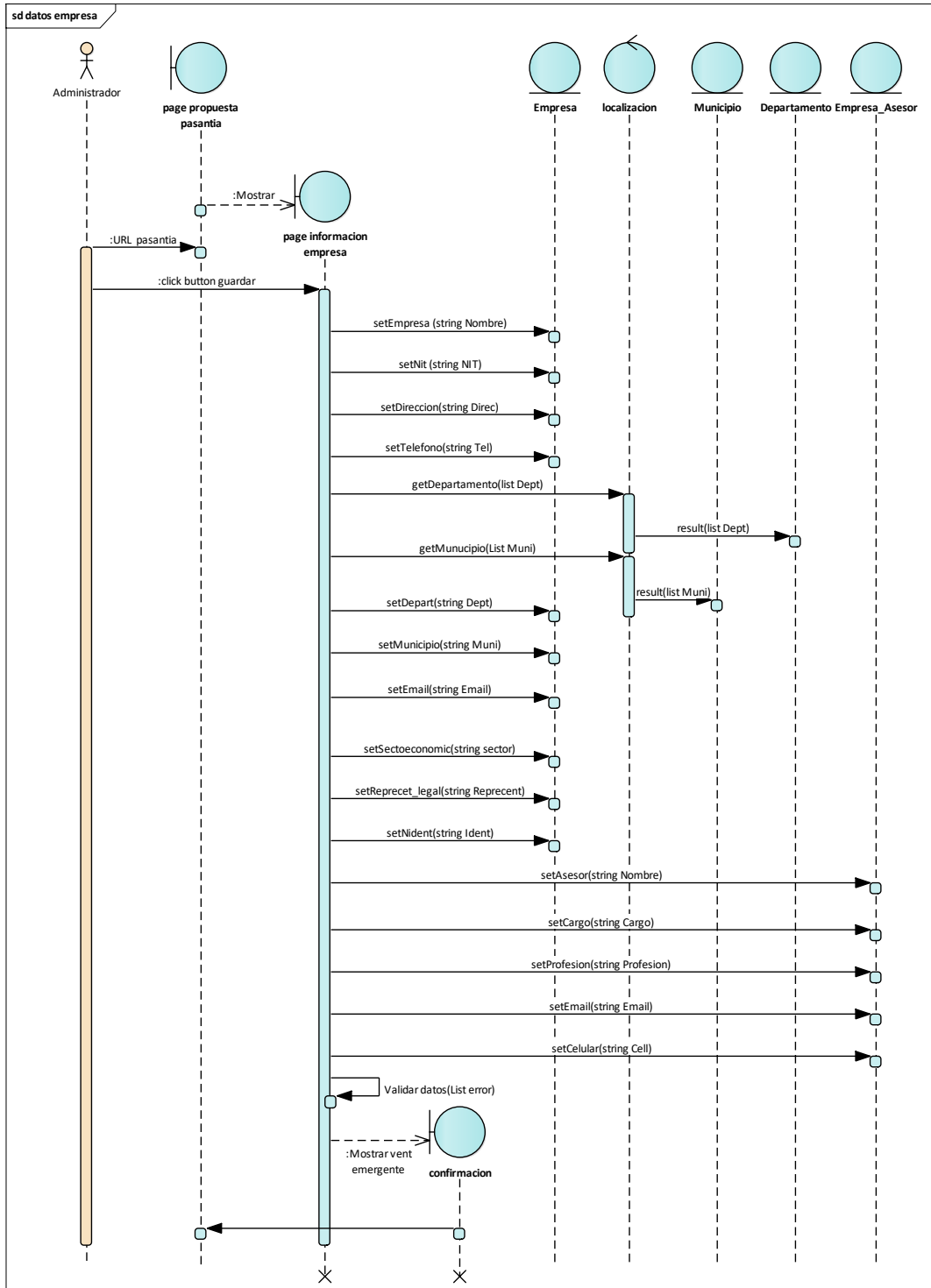
En caso que los datos con coincidan con el tipo de dato de mostrar un erro “tipos de dato no coinciden”, verificar que los campos no se encuentren vacío de ser el caso se enviara un mensaje de advertencia “tienes campos vacíos”.

Diagrama 59. Diagrama de Robustez Registrar Datos de Empresa



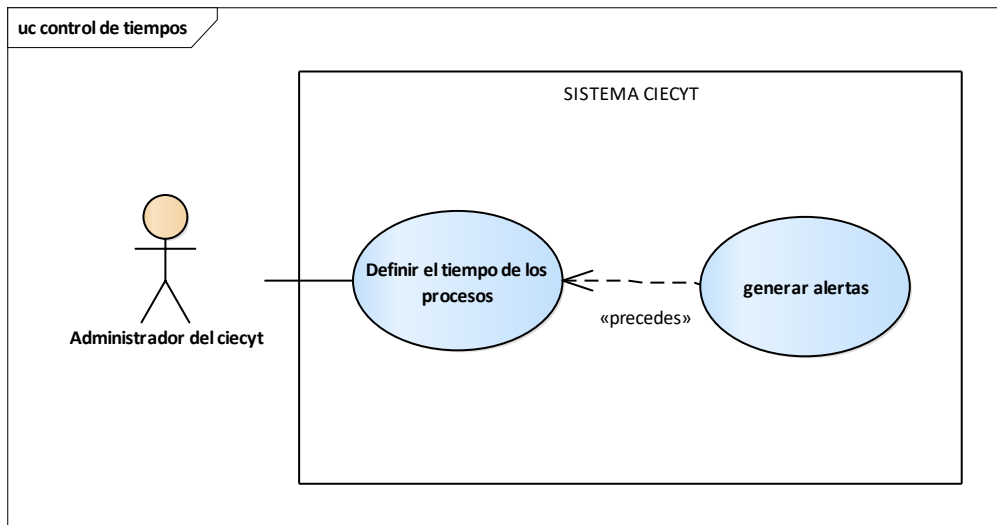
# Diagrama de secuencia

## Diagrama de secuencia 60 Registrar Datos Empresa



Fuente: Los autores

Diagrama 61 Caso de uso: definir tiempo de los procesos



Fuente: Los autores

Curso básico

El administrador del **CIECYT** tendrá la opción de definir los tiempos de cada proceso, se tendrán en cuenta los siguientes aspectos:

Tabla 4 Tabla de tiempos para cada proceso de acuerdo al usuario

PROCESO	TIEMPO NORMAL	TIEMPO EXTRA	USUARIO
Inscripción propuesta	10 días	5 días	estudiante
Revisión propuesta	15 días	10 días	Jurado
Informe	6 meses	1 mes	Estudiante
Informe final	2 meses	15 días	Estudiante
Asignar fecha	3 días	2 días	Administrador ciecyt

Fuente: Los autores

Una vez termine de diligenciar los datos se mostrar un mensaje de confirmación y se procederá a guardar en la base de datos.

**Curso Alternativo**

Si el administrador del ciecyt deja algún campo vacío se mostrará un mensaje de error genérico.

Se validará que los valores para los tiempos no sean menores a cero.

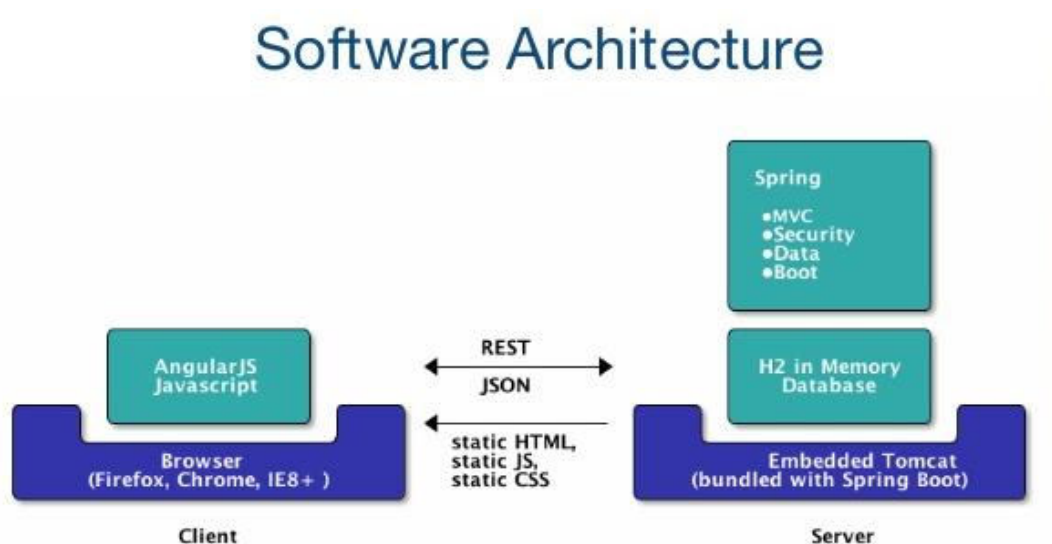
### 4.2.2.3.Arquitectura Inicial:

En el desarrollo de software con procesos ágiles como es el caso de ICONIX, la arquitectura no se diseña completamente, lo que se hace es elegir una arquitectura inicial, con la cual el sistema puede empezar a desplegarse, para de esta manera hacer pruebas y mirar si esa arquitectura se soporta, por tanto se arranca con una arquitectura candidata, la cual puede ser al final la definitiva si el sistema la soporta, pero también pueden haber cambios y al final se adquiera la arquitectura ideal para el sistema

En este trabajo se eligió trabajar con el patrón de Arquitectura por Capas la cual tiene ventajas en la facilidad de desarrollo y pruebas.

La arquitectura por capas se elige como arquitectura inicial, debido a que no se conocen muy a fondo las otras tecnologías y además el tamaño, número de usuarios y tiempos de respuesta no amerita elegir otras que podrían ser más complejas a la hora del desarrollo. Es posible que en otras iteraciones del desarrollo se elija otro patrón como por ejemplo el patrón dirigido por eventos, que podría también ser una solución cuando el sistema escale y sobre todo para el manejo asíncrono de mensajes, en este caso se tendría que refactorizar para adaptar el cambio.

Imagen 2. Arquitectura de software



Fuente: Wahlen, Dr. Oliver <https://www.slideshare.net/owahlen/angularjs-spring4>. (Wahlen, 2014)

En este caso se puede observar una arquitectura que en el Backend trabaja con Java Spring MVC para el manejo de vistas, Security para la seguridad, Data con JPA para el acceso a datos, Boot para la configuración de Spring. En la persistencia se trabajará con H2 un motor de bases de datos ligero (que posteriormete se cambiará a PostgreSQL), y en el despliegue se usará un motor Apache Tomcat. El Backend dispondrá de endpoints API/REST para el servicio de datos en formato JSON.

En el FrontEnd se trabajará con Angular y JavaScript (o TypeScript según se decida posteriormente), este frontend se encarga de crear las vistas de la aplicación, las cuales serán finalmente visualizadas en los navegadores de los clientes

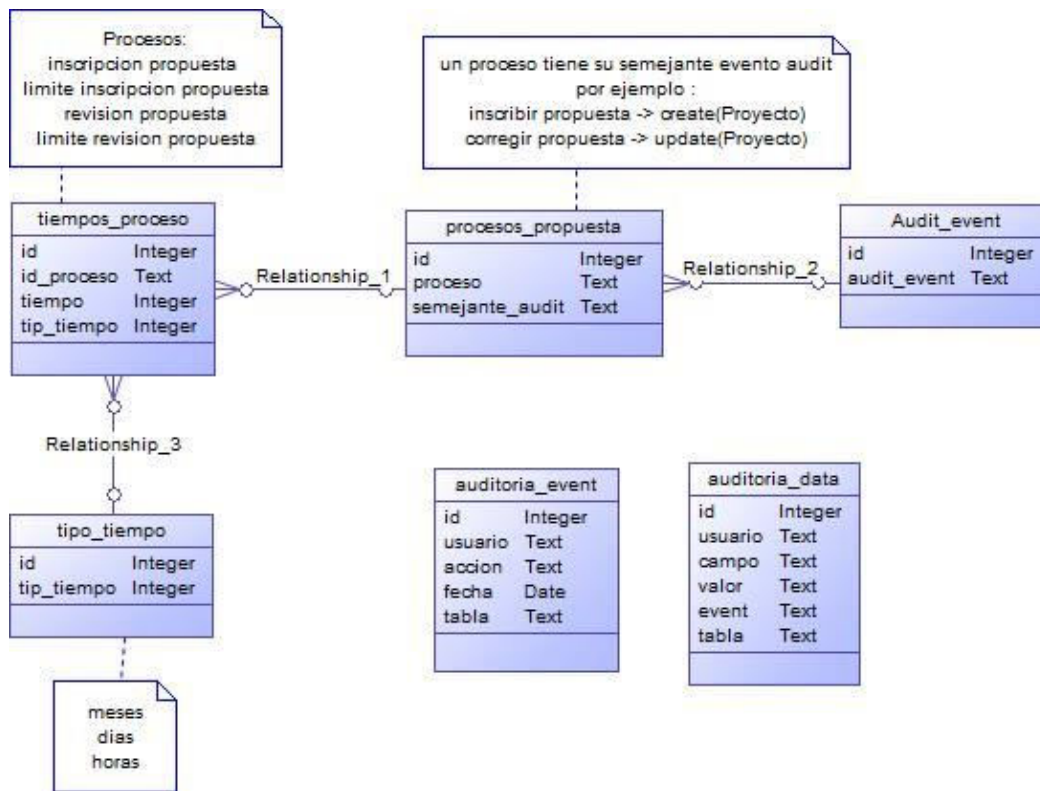
#### ***4.2.2.4. Generar alertas***

La generación de alertas será una terea del sistema para realizarla se tendrá en cuenta lo siguiente:

Primero nos basaremos en los días que ha guardado el administrador para mostrar las alertas cada cierto tiempo, se creara disparadores para las tablas “proyecto” y “pregunta\_respuesta” lo cual será relacionado con la tabla “auditoria\_event”, “auditoria\_data” las cuales guardaran los registros de la creación y actualización de datos realizadas por algún usuario registrado en la base de datos.

También se relacionará los posibles eventos de las tablas como crear, actualizar, eliminar, ver; junto con los procesos como inscribir propuesta, corregir, evaluar, asignar fecha entre otros. De esta manera se puede automatizar las alertas de acuerdo a los eventos registrados en las tablas de auditoria teniendo en cuenta el proceso y el tiempo asignado por los usuarios con el rol dentro del sistema “administrador del CIECYT”.

Diagrama 62 Diseño de diagrama de clases para el caso de uso generar alertas



Fuente: los autores

Para que la comunicación con el servidor sea más eficiente y la generación de las alertas sea más efectiva se podría implementar WebSockets, el cual se define como “una tecnología avanzada que permite abrir una comunicación interactiva canal entre el navegador del usuario y un servidor. Con esta API, puedes enviar mensajes a un servidor y reciba respuestas dirigidas por eventos sin tener que sondear el servidor para obtener una respuesta. WebSockets también ha sido llamado "TCP para la web".” (Raible, The JHipster Mini-Book, 2019), esta tecnología podría ser una opción a tener en cuenta.

### 4.3. Prototipos de interfaz de usuario

Los prototipos de usuario son una vista previa del sistema y ayudan a explicar mejor su funcionamiento, para el desarrollo de prototipos de usuario se tomaron en cuenta los siguientes:

### 4.3.1. Interfaz para Estudiante

Interfaz para inscribir la propuesta

Para el proceso de inscribir una propuesta se llevó a cabo a través de seis interfaces las cuales juntas hacen el proceso de la inscripción

#### Interfaz para ingresar información general de la propuesta.

Cuando el estudiante ingrese por la opción “inscribir propuesta” se mostrará la siguiente interfaz donde podrá ingresar la información requerida de una forma secuencial donde primero debe ingresar la información general, elementos, presupuesto, resultados esperados, impactos esperados, entidades financieras, cronograma de actividades. Cada una de estas opciones lo llevara un interfaz para realizar su respectivo proceso.

Imagen 3. Interfaz ingreso información general de la propuesta

The screenshot displays a web application interface for submitting a proposal. The main content area is titled 'INFORMACION GENERAL DE LA PROPUESTA' and contains a form with the following fields and sections:

- Navigation:** A top menu with 'Menú' and a search bar. A left sidebar contains 'Propuesta' (with sub-items 'Mi propuesta' and 'Inscribir Propuesta'), 'Solicitudes' (with sub-items 'Mis solicitudes' and 'Nueva solicitud'), 'Documento Final', and 'Observaciones' (with sub-items 'Jurado' and 'Asesor').
- Form Tabs:** A horizontal tab bar with 'Información General' (selected), 'Elementos', 'Presupuesto', 'Resultados Esperados', 'Impactos Esperados', 'Entidades financieras', and 'Cronograma de Actividades'.
- Form Fields:**
  - Ciudad: [ ] Fecha: [ ]
  - Título Provisional de la Propuesta: [ ] Tesis: [ ]
  - Grupo o semillero de Investigación: [ ]
  - Facultad/Programa: [ ]
  - Asesor: [ ] Coordinador: [ ] Investigador Principal: [ ] Dedicación Hr/Sem: [ ]
  - Integrantes del Proyecto de Investigación (Listado):
    - Nombre Completo: [ ] No. de Identificación: [ ] Fecha Y Lugar de Expedición: [ ]
  - Programa Académico: [ ] Rol: [ ]
  - Teléfono: [ ] E-mail: [ ]
  - Línea de Investigación: [ ]
  - Sublínea de Investigación: [ ]
  - Lugar de Ejecución de la Proyecto: [ ]
  - Duración del Proyecto Investigativo (meses): [ ]
  - Fecha de Inicio: [ ] Fecha de Terminación: [ ]
  - Tipo de Proyecto: [ ]
  - Investigación Básica  Investigación Aplicada  Desarrollo Tecnológico o Experimental
  - Valor Contrapartida I.T.P.: [ ] Efectivo: (En miles) [ ] Especie: (En miles) [ ]
  - Descriptorios / Palabras Claves: [ ]
  - Nombre de la Convocatoria a la cual se Presenta el Proyecto: (Aplica para convocatoria interna) [ ]
- Buttons:** A 'Guardar' button is located at the bottom center of the form area.

Fuente: Los autores

## Interfaz para ingresar los elementos de la propuesta.

Imagen 4. Interfaz ingreso de los elementos de la propuesta

The screenshot shows a web application interface for submitting a proposal. At the top, there is a 'Menú' bar with a search field and user profile icons. On the left, a sidebar menu lists 'Propuesta' (with sub-items 'Mi propuesta' and 'Inscribir Propuesta'), 'Solicitudes' (with sub-items 'Mis solicitudes' and 'Nueva solicitud'), 'Documento Final', and 'Observaciones' (with sub-items 'Jurado' and 'Asesor'). The main content area is titled 'Inscribir Propuesta' and features a horizontal navigation bar with tabs: 'Información General', 'Elementos' (highlighted), 'Presupuesto', 'Resultados Esperados', 'Impactos Esperados', 'Entidades financieras', and 'Cronograma de Actividades'. Below the tabs is a large empty rectangular form area. To the right of this area is a list of elements to be entered: 'Elementos' followed by 'Titulo Provisional', 'Resumen Ejecutivo', 'Planteamiento del Problema y Preguntas de Investigación', 'Justificación en Términos de Necesidades y Pertinencia', 'Objetivo General', 'Objetivos Específicos', 'Marco de Referencia', and 'Metodología'. A 'Guardar' button is located at the bottom center of the form area.

Fuente: Los autores

## Interfaz para ingresar el presupuesto

Imagen 5. Interfaz de ingreso de presupuesto de la propuesta

The screenshot shows the 'Mi Propuesta' interface with the 'PRESUPUESTO' tab selected. The sidebar menu is identical to the previous image. The main content area has a navigation bar with tabs: 'Información General', 'Elementos', 'Presupuesto' (highlighted), 'Resultados Esperados', 'Impactos Esperados', 'Entidades financieras', and 'Cronograma de Actividades'. The main form area contains several input fields: 'DESCRIPCIÓN' and 'JUSTIFICACIÓN' (text boxes), 'CANTIDAD', 'VALOR UNITARIO', and 'VALOR TOTAL' (text boxes), 'CONTRAPARTIDA' (with sub-fields 'Especie' and 'Dinero'), 'ENTIDAD FINANCIADORA' (a dropdown menu), and 'TOTAL' (text box). To the right of the form is a list of 'RUBROS' (budget categories): 'Equipos y Software', 'Personal', 'Viajes', 'Salida de Campo', 'Materiales y Suministros', 'Servicio Técnico', 'Publicaciones y Patentes', 'Material Bibliográfico', 'Construcciones', 'Mantenimiento', and 'Administración'. A 'Guardar' button is located at the bottom center of the form area.

Fuente: Los autores

Imagen 6. Interfaz para ingresar los resultados esperados

Fuente: Los autores

Imagen 7. Interfaz para ingresar los impactos esperados

Fuente: Los autores

Imagen 8. Interfaz para ingresar las entidades financieras

The screenshot shows a web application interface for managing financial entities. At the top, there is a 'Menú' (Menu) and a search bar. The main navigation bar includes 'Mi Propuesta' and 'ENTIDADES FINANCIADORAS'. The 'ENTIDADES FINANCIADORAS' section is active, with sub-tabs for 'Información General', 'Elementos', 'Presupuesto', 'Resultados Esperados', 'Impactos Esperados', 'Entidades financieras', and 'Cronograma de Actividades'. The 'Entidades financieras' tab is selected. On the left, there is a sidebar with 'Propuesta' (Mi propuesta, Inscribir Propuesta), 'Solicitudes' (Mis solicitudes, Nueva solicitud), 'Documento Final', and 'Observaciones' (Jurado, Asesor). The main content area has a form with fields for 'Entidad financiera' (a dropdown menu), 'valor' (a text input), and 'Estado Financiero' (a dropdown menu). A green 'Agregar' button is below the form. To the right is a table with three columns: 'Entidades', 'Valor', and 'Estado Financiero'. A 'Guardar' button is at the bottom.

Fuente: Los autores

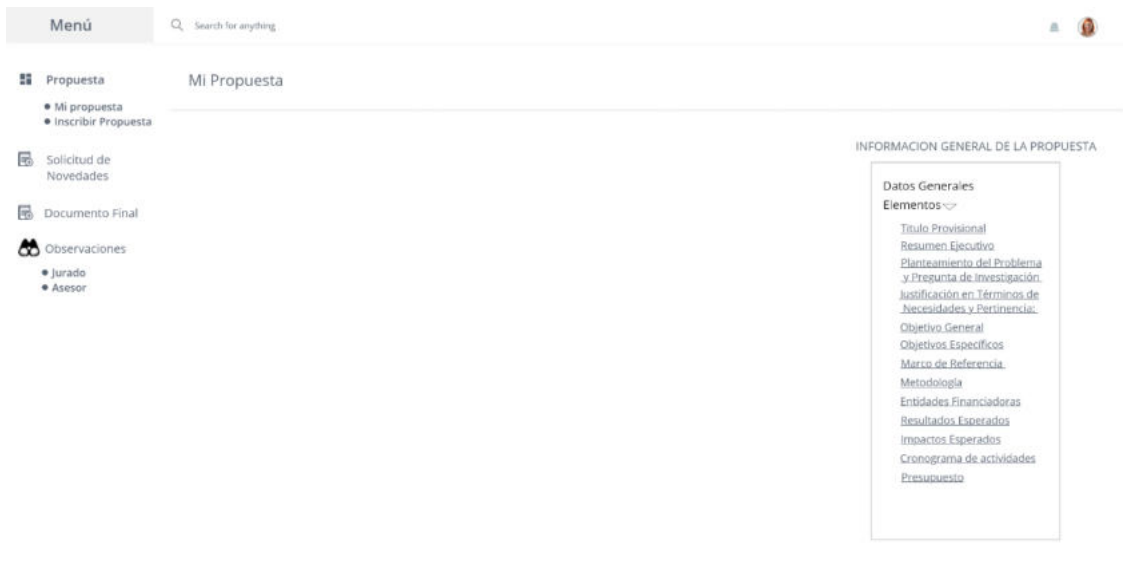
Imagen 9. Interfaz para ingresar los cronogramas de actividades

The screenshot shows the 'Cronograma de Actividades' (Activity Schedule) interface. The top navigation is similar to the previous image, but the 'Cronograma de Actividades' tab is selected under 'ENTIDADES FINANCIADORAS'. The sidebar is the same. The main content area has a heading 'Cronograma de Actividades:' followed by the instruction: 'Se describen las diferentes actividades a realizar según la metodología, indicando el tiempo (meses) programado para su ejecución.' Below this are three text input fields labeled 'Actividades:', 'Metodología:', and 'Tiempo(meses):'. A 'Guardar' button is at the bottom.

Fuente: Los autores

Cuando el estudiante ingrese por la opción de “mi propuesta” se mostrará la siguiente interfaz.

Imagen 10. Interfaz para revisar la propuesta en desarrollo



Fuentes: Los autores

Esta opción le permitirá al estudiante poder revisar la información ingresada de su propuesta, para ello tendrá un menú donde con cada uno de los campos de su propuesta que le permitirá moverse más rápido para buscar datos específicos con solo selecciona el campo de la propuesta que desea ver.

Imagen 11- Interfaz para revisar solicitudes



Fuente: Los autores

Imagen 12. Interfaz para realizar una Nueva Solicitud



Fuente: Los autores

## Interfaz para la revisión de observaciones

Cuando el estudiante ingrese en la opción “observaciones jurado” se mostrará la siguiente interfaz, para la opción “observaciones asesor” se mostrará de la misma manera como la de jurado solo cambia la información del remitente.

Imagen 13. Interfaz para la revisión de observaciones

Menú Search for anything

Propuesta

- Mi propuesta
- Inscribir Propuesta

Solicitudes

- Mis solicitudes
- Nueva solicitud

Documento Final

Observaciones

- Jurado
- Asesor

Observaciones OBSERVACIONES JURADO

Nombre	Estado	Fecha
MARIO ENRIQUE MUÑOS	INICIAL	19/03/2019
MARIO ENRIQUE MUÑOS	INICIAL	19/03/2019

10 | pagina 1 de 30 | << Siguiente Anterior >>

Fuente: Los autores

En esta página nos alistará el listado de solicitudes por parte del jurado, cuando seleccione el nombre de jurado se mostrará la interfaz ().

Imagen 14. Interfaz de observaciones del jurado

Menú Search for anything

Propuesta

- Mi propuesta
- Inscribir Propuesta

Solicitudes

- Mis solicitudes
- Nueva solicitud

Documento Final

Observaciones

- Jurado
- Asesor

Observaciones OBSERVACIONES JURADO

Elementos	Observación
Objetivo	Debes mejorar el Objetivo porque hay incurrencia.....
Cronograma	

10 pagina 1 de 30 << Sigüente Anterior >>

Fuente: Los autores

Imagen 15 Interfaz para inscribir propuesta por pasantía

Menú Search for anything

Propuesta

- Mi propuesta
- Inscribir Propuesta
  - Tesis
  - Pasantía
  - Investigación
  - Grupo
  - Seminario
- Solicitudes
  - Mis solicitudes
  - Nueva solicitud
- Documento Final
- Observaciones
  - Jurado
  - Asesor

Inscribir Propuesta "pasantia"

Información del Estudiante

Elementos

- Título de la Propuesta
- Planteamiento del Problema
- Justificación
- Objetivo General
- Objetivos Especificos
- Metodología
- Cronograma
- Funciones a Realizar
- Resultados Esperados

INFORMACIÓN DEL ESTUDIANTE

Ciudad Fecha Radicado

Nombre del Estudiante

Facultad

- Ingenierías y Ciencias Básicas
- Ciencias Administrativas, Económicas y Contables
- Educación y Ciencias Humanas

Programa

Número de horas

- Técnico Profesional 480 h
- Tecnólogo 640 h
- Profesional Universitario 880 h

Dirección Correo electrónico Teléfono

Días en los que se realiza

L  M  M  J  v  S  D

Horario Total horas/mes

Apoyos Aprobados

- Bono Alimenticio
- Auxilio de Transporte
- Capacitación
- Apoyo Económico
- Otro
- ¿Cuál?:

Salir Guardar

Fuente: Los autores

Imagen 16 Interfaz ingresar elementos de la propuesta de pasantía

The screenshot shows a web application interface for submitting a proposal. The main heading is "Inscribir Propuesta 'pasantia'". On the left, there is a navigation menu with categories like "Propuesta", "Mi propuesta", "Solicitudes", "Documento Final", and "Observaciones". The central area is titled "INFORMACIÓN GENERAL DE LA PROPUESTA" and contains a section for "RESULTADOS ESPERADOS". This section has a text input field with the placeholder "Ingresar datos.....". Below the input field are two buttons: "Salir" and "Guardar".

Fuente: Los autores

Imagen 17 Interfaz para ingresar Cronograma

The screenshot shows the same web application interface, but now displaying the "CRONOGRAMA" section. The heading is "CRONOGRAMA". Below the heading, there is a description: "Cronograma: Se describen las diferentes actividades a realizar según la metodología, indicando el tiempo (meses) programado para su ejecución." There are two input fields: "Actividades" and "Sustentación". The "Sustentación" field has a dropdown menu with "M1" selected. To the right, there is a "TIEMPO(meses)" dropdown menu with options "M1", "M2", "M3", and "M4", where "M4" is currently selected. An "Agregar" button is positioned to the right of the "TIEMPO(meses)" dropdown. At the bottom of the section are "Salir" and "Guardar" buttons.

Fuente: Los autores

## Interfaz del administrador

Interfaz para la asignación de la fecha de sustentación

Dentro de la opción “propuesta → nombre de la propuesta → asignar fecha” se mostrara los datos necesario para la asignación de la fecha de sustentación como se muestra en la siguiente imagen.

Imagen 18. Interfaz de asignación fecha de sustentación

Menú

Propuesta  
Investigación  
Administración  
• Usuarios  
• Nuevo Usuario  
Documento Final

Información General | Elementos | Presupuesto | Resultados Esperados | Impactos Esperados | Entidades financieras | Jurado | Solicitudes | **Asignar Fecha**

Recuperación de los humedales en el municipio de mocoa

INTEGRANTES DE LA PROPUESTA:

Nombre y Apellidos completos	Documento Identidad	Programa Academico

FECHA

HORA

Guardar

Fuente: Los autores

Imagen 19. Interfaz para la asignar jurado a la propuesta

Menú

Propuesta

Investigación

Administración

- Usuarios
- Nuevo Usuario

Documento Final

Información General | Elementos | Presupuesto | Resultados Esperados | Impactos Esperados | Entidades financieras | **Jurado** | Solicitudes | Asignar Fecha

Search for anything

SISTEMAS

CONTADIRIA

ADMINISTRADOR

Nombre	Fecha tecnica	Fecha
MARIO ENRIQUE MUÑOS	ver	19/03/2019
MARIO ENRIQUE MUÑOS	ver	19/03/2019

pagina 1 de 30

<< Siguientes Anterior >>

10

Fuente: Los autores

Imagen 20 Interfaz para agregar nueva empresa

Menú

Search for anything

Propuesta

Administración

- Usuarios
- Nuevo Usuario
- Registrar Empresa

Documento Final

Registrar Nueva Empresa

INFORMACIÓN DE LA EMPRESA

Nombre de la Empresa/Organización  NIT

Dirección  E-mail

Teléfono de contacto:  Municipio  Departamento

Sector económico al cual pertenece la empresa

Representante Legal  No. de identificación

Asesor de la Empresa  Cargo

Profesión  E-mail

Celular

Salir Guardar

Fuente: los autores

## Interfaz para jurado

### Imagen 21 Interfaz Evaluar Propuesta



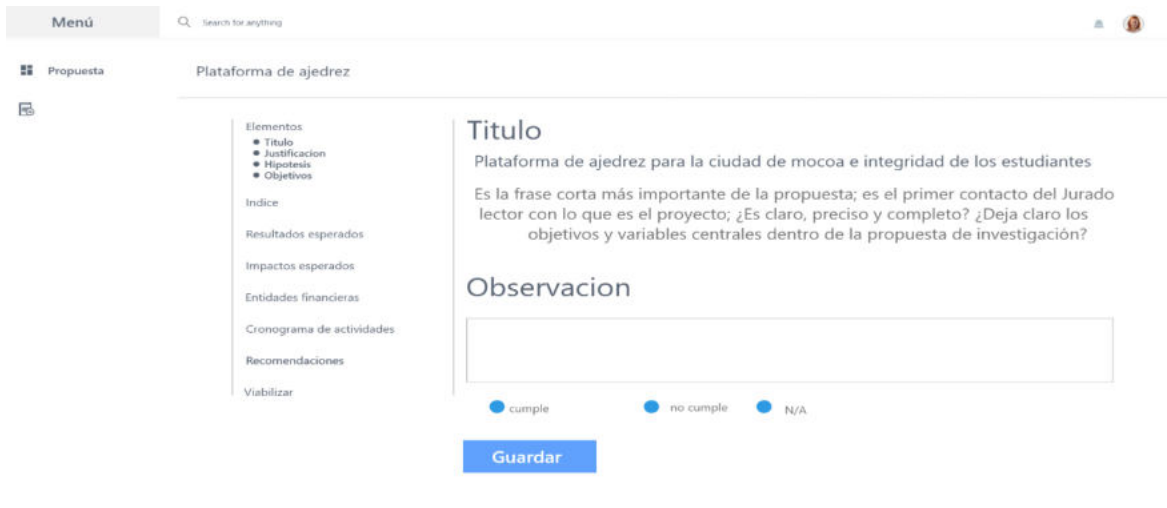
The screenshot shows a web interface for evaluating proposals. At the top, there is a 'Menú' button and a search bar. Below the navigation bar, the page title is 'Propuestas asignadas'. A dropdown menu is set to 'FASE EN DESARROLLO DE LA PROPUUESTA'. The main content is a table with the following data:

Nombre	Estado	Fecha	Documento	Modalidad
Plataforma de ajedrez	APROBADO	19/03/2019	ajedrez.docx	Tesis
Reforestación de montaña	APROBADO	19/03/2019	reforestación.docx	Semillero

At the bottom of the table, there is a pagination control showing '10 pagina 1 de 30' and navigation buttons: '<<', 'Siguiente', 'Anterior', and '>>'.

Fuente: Los autores

### Imagen 22 Interfaz Evaluar Elementos de la propuesta



The screenshot shows the evaluation form for the 'Plataforma de ajedrez' proposal. The interface includes a 'Menú' button, a search bar, and a navigation bar with 'Propuesta' and 'Plataforma de ajedrez'. A sidebar on the left lists evaluation elements: 'Elementos' (with sub-items: Título, Justificación, Hipotesis, Objetivos), 'Indice', 'Resultados esperados', 'Impactos esperados', 'Entidades financieras', 'Cronograma de actividades', 'Recomendaciones', and 'Viabilizar'. The main content area is titled 'Titulo' and contains the text: 'Plataforma de ajedrez para la ciudad de mocoa e integridad de los estudiantes'. Below this, a paragraph explains the importance of the title: 'Es la frase corta más importante de la propuesta; es el primer contacto del Jurado lector con lo que es el proyecto; ¿Es claro, preciso y completo? ¿Deja claro los objetivos y variables centrales dentro de la propuesta de investigación?'. The 'Observacion' section has a text input field. At the bottom, there are three radio buttons for evaluation: 'cumple', 'no cumple', and 'N/A'. A blue 'Guardar' button is located at the bottom center.

Fuente: Los autores

Imagen 23. Viabilizar Propuesta

The screenshot shows a web application interface for 'Plataforma de ajedrez'. At the top, there is a 'Menú' button and a search bar with the placeholder text 'Search for anything'. Below the search bar, the page title 'Plataforma de ajedrez' is displayed. On the left side, there is a sidebar menu with the following items: 'Propuesta', 'Elementos' (with sub-items: 'Titulo', 'Justificacion', 'Hipotesis', 'Objetivos'), 'Presupuesto', 'Resultados esperados', 'Impactos esperados', 'Entidades financieras', 'Cronograma de actividades', 'Recomendaciones', and 'Viabilización'. The main content area is titled 'VIABILIZACIÓN' and contains the text: 'La propuesta de investigación cumple con todos los requerimientos establecidos, se aprueba con:'. Below this text are three radio button options: 'Sin modificaciones', 'Con modificaciones', and 'No Viable'. Underneath the radio buttons is a section titled 'CONCEPTO' with a large text input field. At the bottom of the form is a blue button labeled 'Guardar'.

Fuente: Los autores

#### 4.4. Prototipo

Desarrollo del prototipo del sistema de información para el centro de investigación del Instituto Tecnológico del Putumayo con la herramienta JHIPSTER

La fase del desarrollo se limitó a estructurar y corregir la cantidad de archivos que genera la Herramienta de desarrollo JHIPSTER a causa del extenso tamaño del proyecto, el cual fue dado en base al análisis y diseño de objetivos anteriores, donde se utilizó la metodología ICONIX para llegar a este punto.

#### Jhipster

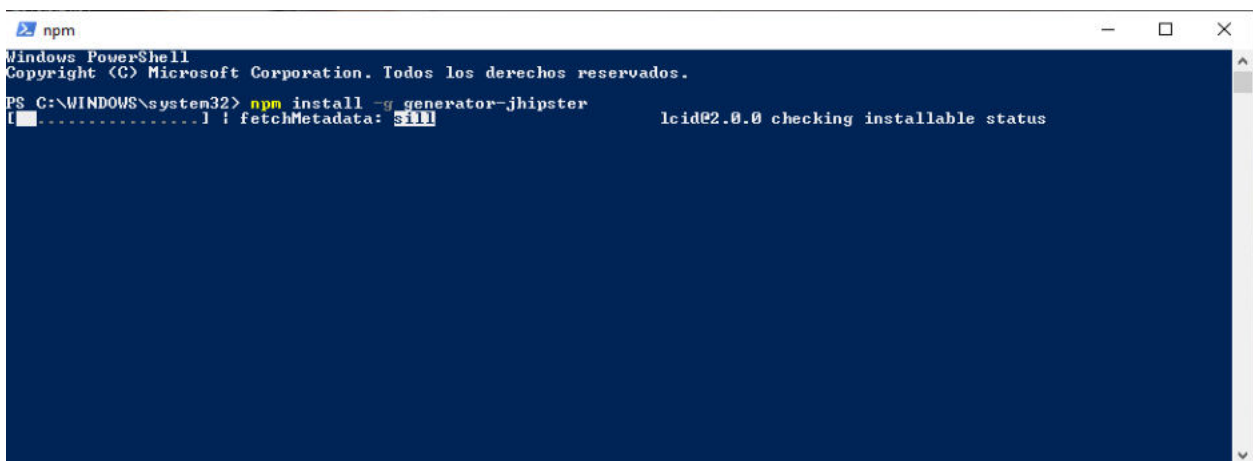
Primero comprendamos que es Jhipster para tener una idea más clara de esta metodología de trabajo de acuerdo a (Raible, The JHipster Mini-Book, 2019) “En su núcleo, JHipster es un generador Yeoman. Yeoman es un generador de código que ejecuta con un comando yo para generar aplicaciones completas o partes útiles de una aplicación. Los generadores de Yeoman

promueven lo que el equipo de Yeoman llama "flujo de trabajo de Yeoman". Este es un conjunto de herramientas de opinión del lado del cliente que puede ayudar a los desarrolladores a crear rápidamente aplicaciones web hermosas. Se encarga de proporcionar todo lo necesario para trabajar sin los problemas habituales asociados con la configuración manual.”

### Instalación

La instalación de Jhipster es muy sencilla, primero instalamos el gestor de paquetes de Node.js desde su web oficial, después simplemente digitamos el siguiente comando en PoweShell: `npm install -g generator-jhipster`, al igual que se observa en la siguiente imagen.

Imagen 24 Instalación de Jhipster en Windows desde PowerShell(Administrador)



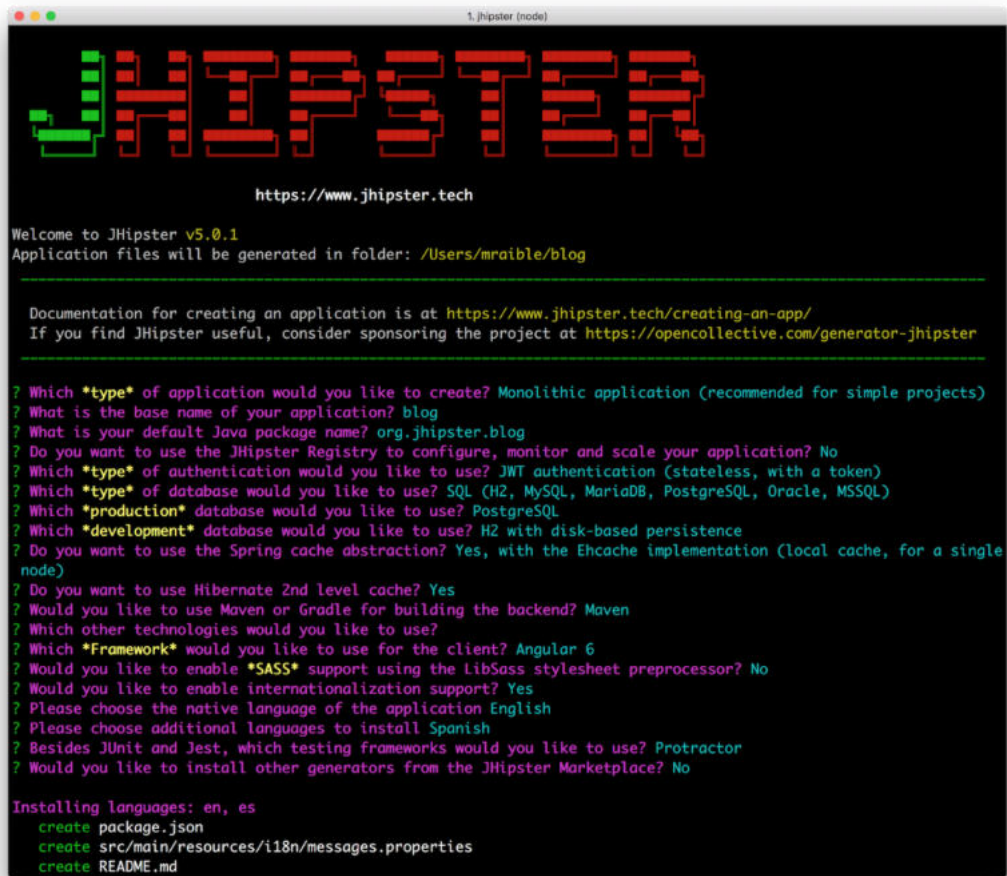
```
npm
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.
PS C:\WINDOWS\system32> npm install -g generator-jhipster
[.....] | fetchMetadata: still          lcid@2.0.0 checking installable status
```

Fuente: Los autores generada por el sistema

### **Generación del proyecto**

La generación de un proyecto de Jhipster se realiza a modo de pregunta respuesta donde el generador nos pregunta las características que nosotros queremos para el proyecto, por ejemplo, el nombre, el nombre base, el tipo de seguridad, el tipo de base de datos relacional o no relacional, la internacionalización, la tecnología para el frontend, entre otras opciones las cuales se transformarían en el core(Base) de nuestra aplicación.

Imagen 25 Características de un proyecto generado con Jhipster



```
1_jhipster (node)
JHIPSTER
https://www.jhipster.tech

Welcome to JHipster v5.0.1
Application files will be generated in folder: /Users/mraible/blog

Documentation for creating an application is at https://www.jhipster.tech/creating-an-app/
If you find JHipster useful, consider sponsoring the project at https://opencollective.com/generator-jhipster

? Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? What is the base name of your application? blog
? What is your default Java package name? org.jhipster.blog
? Do you want to use the JHipster Registry to configure, monitor and scale your application? No
? Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? Which *production* database would you like to use? PostgreSQL
? Which *development* database would you like to use? H2 with disk-based persistence
? Do you want to use the Spring cache abstraction? Yes, with the Ehcache implementation (local cache, for a single node)
? Do you want to use Hibernate 2nd level cache? Yes
? Would you like to use Maven or Gradle for building the backend? Maven
? Which other technologies would you like to use?
? Which *Framework* would you like to use for the client? Angular 6
? Would you like to enable *SASS* support using the LibSass stylesheet preprocessor? No
? Would you like to enable internationalization support? Yes
? Please choose the native language of the application English
? Please choose additional languages to install Spanish
? Besides JUnit and Jest, which testing frameworks would you like to use? Protractor
? Would you like to install other generators from the JHipster Marketplace? No

Installing languages: en, es
create package.json
create src/main/resources/i18n/messages.properties
create README.md
```

Fuente: (Raible, DocGist, 2019)

Cabe resaltar que es muy importante la arquitectura seleccionada porque de acuerdo a esta se generara una estructura de carpetas y archivo diferente, además de las distintas características extra que puede presentar cada una de estas. Para este prototipo elegimos la arquitectura monolítica.

### Monolítica

Las aplicaciones monolíticas son muy simples a la vista de un desarrollador puesto que su estructura de carpetas tanto del frontend como del backend están alojadas en un mismo proyecto esto nos genera una facilidad a la hora de movernos entre archivos puesto que podríamos pasar muy fácil de programar una funcionalidad en spring framework a crear un nuevo componente en angular podríamos tener en cuenta la siguiente definición “Una arquitectura monolítica, es una

simple unidad, donde las aplicaciones están construidas en tres partes: una Interfaz de Usuario Cliente, Aplicación del lado del Servidor y una Base de Datos (Chiza, 2018)

Imagen 26 Arquitectura de una aplicación monolítica

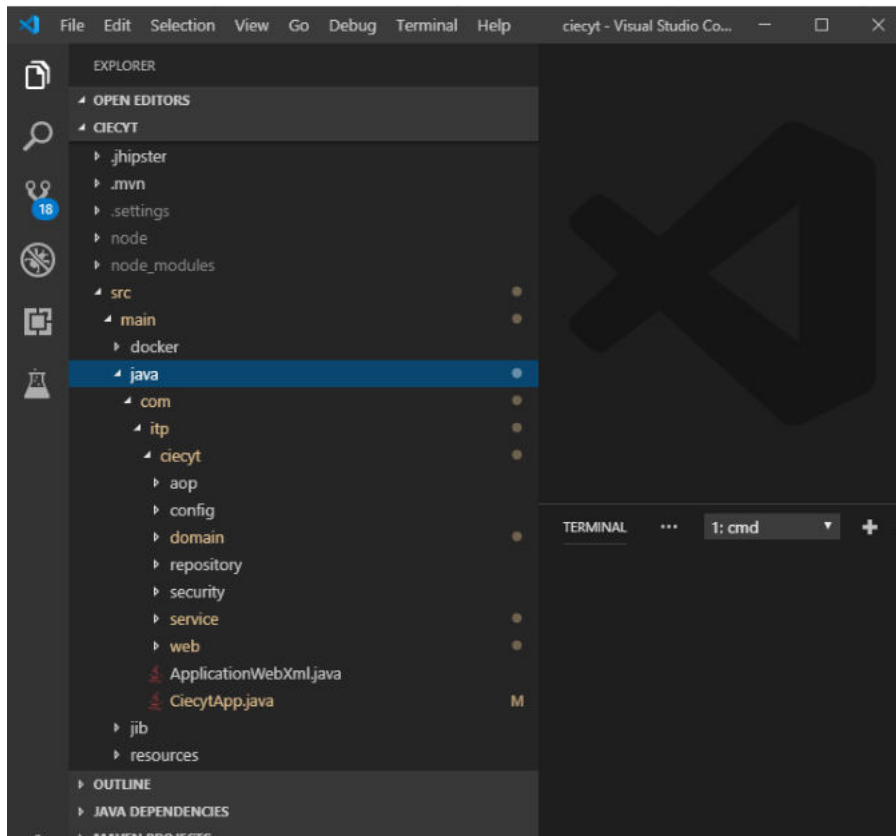


Fuente: Análisis De Rendimiento Entre Una Arquitectura Monolítica (Chiza., 2018)

### **Estructura del proyecto**

La estructura de un proyecto para una aplicación monolítica generada por jhipster en el lado del backend se apoya en maven el cual es “una herramienta de gestión y comprensión de proyectos de software. Basándose en el concepto de un modelo de objeto de proyecto (POM), Maven puede gestionar la compilación, los informes y la documentación de un proyecto a partir de una información central.” (apache, s.f.). En visual studio code se vería de la siguiente forma.

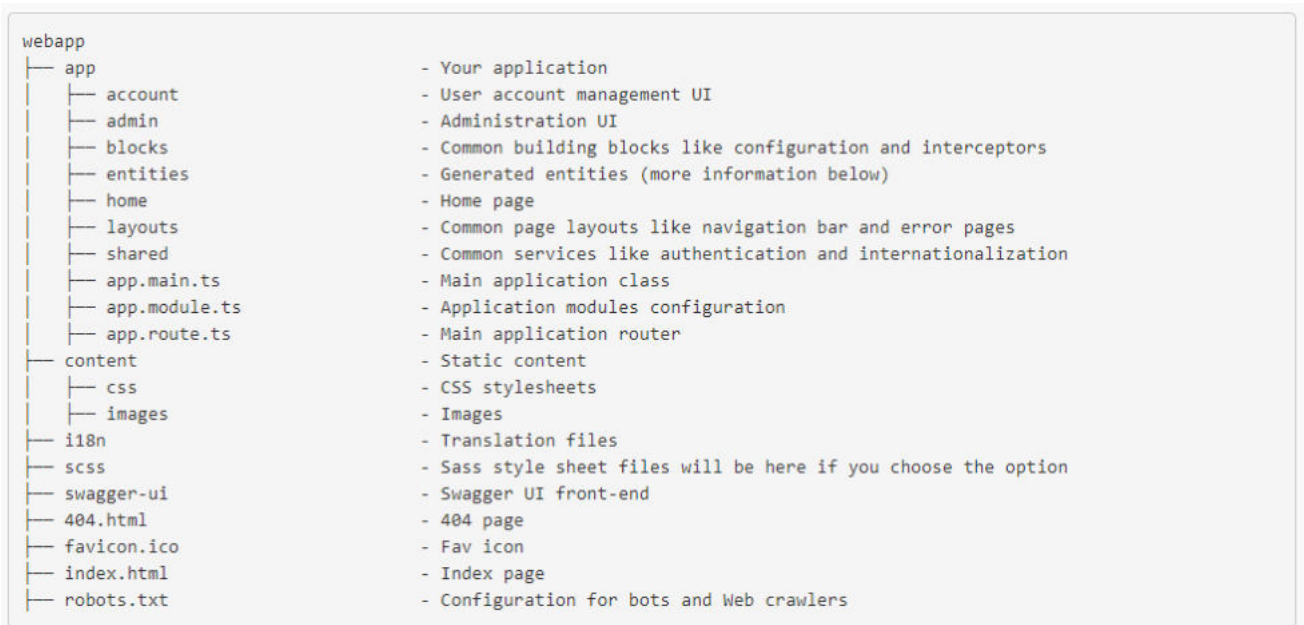
Imagen 27 Estructura del proyecto para spring framework(backend) desde visual studio code



Fuente: los autores

Para generar la estructura de angular jhipster se apoya en webpack como lo dicen en su página oficial “Angular está utilizando TypeScript en lugar de JavaScript, y como resultado, se necesitan algunas herramientas específicas para trabajar de manera eficiente con él.” (jhipster, s.f.) pero que es webpack de acuerdo a su página oficial sería “En su núcleo, webpack es un paquete de módulos estáticos para aplicaciones JavaScript modernas. Cuando el paquete web procesa su aplicación, genera internamente un gráfico de dependencia que mapea cada módulo que su proyecto necesita y genera uno o más paquetes.” (webpack, s.f.) básicamente empaqueta y exporta los ficheros.

Imagen 28 Estructura del proyecto para angular(frontend)



(jhipster, s.f.)

Imagen 29 Aplicación de spring framework

Jhipster genera una estructura de ficheros óptima para la creación de servicios web, utilizando el framework spring, y simplificando su configuración con spring-boot, al trabajar con tecnologías distintas en este caso angular y spring, el envío y recepción de datos se realiza por medio de servicios RESTful la cual se basa en la arquitectura REST.

### Servicios REST

Se lo define de la siguiente manera: “REST define un conjunto de principios arquitectónicos por los que se pueden diseñar servicios Web que se centran en los recursos de un sistema, lo que incluye la forma en que los estados de los recursos se dirigen y transfieren a través de HTTP por un amplio rango de clientes que están escritos en diferentes lenguajes. Si lo medimos por el número de servicios Web que lo utilizan, en los últimos años REST ha emergido como un modelo de diseño predominante para los servicios Web. De hecho, REST ha tenido tanto impacto en la Web que ha desplazado enormemente al diseño de interfaces basado en SOAP y WSDL, porque es un estilo considerablemente más fácil de utilizar”**Fuente especificada no válida.**, esto dio paso para la creación de los servicios RESTful , un servicio REST es una arquitectura que permite la conexión

de múltiples tecnologías con diferentes servidores es decir muchas tecnologías intercambian datos con múltiples servidores de esta forma no existe un limitante con respecto a

## Servicio RESTful

(Bellido, 2011) “Una telaraña de recursos interconectados identificados con URLs, que pueden ser manipulados a través de interfaces uniformes (métodos HTTP), cuyo estado es servido a través de representaciones (p. e. página HTML) que contienen enlaces y controles (p. e. un formulario indicado una operación POST), los cuales definen el modelo hipermedia que determina no solo la relación entre recursos, sino también la posible red de transiciones de estado del recurso”

Imagen 30 Implementación del servicio RESTful en la entidad PROYECTO, con jhipster

```
28 /**
29  * REST controller for managing {@link com.itp.ciecyt.domain.Proyecto}.
30  */
31 @RestController
32 @RequestMapping("/api")
33 public class ProyectoResource {
34
35     private final Logger log = LoggerFactory.getLogger(ProyectoResource.class);
36
37     private static final String ENTITY_NAME = "proyecto";
38
39     @Value("${jhipster.clientApp.name}")
40     private String applicationName;
41
42     private final ProyectoService proyectoService;
43
44     public ProyectoResource(ProyectoService proyectoService) {
45         this.proyectoService = proyectoService;
46     }
47
48     /**
49     * {@code POST} /proyectos : Create a new proyecto.
50     *
51     * @param proyecto the proyecto to create.
52     * @return the {@link ResponseEntity} with status {@code 201 (Created)} and with body the new proyecto, or with status {@code 400 (Bad
53     * @throws URISyntaxException if the Location URI syntax is incorrect.
54     */
55     @PostMapping("/proyectos")
56     public ResponseEntity<Proyecto> createProyecto(@RequestBody Proyecto proyecto) throws URISyntaxException {
57         log.debug("REST request to save Proyecto : {}", proyecto);
58         if (proyecto.getId() != null) {
59             throw new BadRequestAlertException("A new proyecto cannot already have an ID", ENTITY_NAME, "idexists");
60         }
61         Proyecto result = proyectoService.save(proyecto);
62         return ResponseEntity.created(new URI("/api/proyectos/" + result.getId()))
63             .headers(HeaderUtil.createEntityCreationAlert(applicationName, true, ENTITY_NAME, result.getId().toString()))
64             .body(result);
65     }
66 }
```

Fuente: los autores

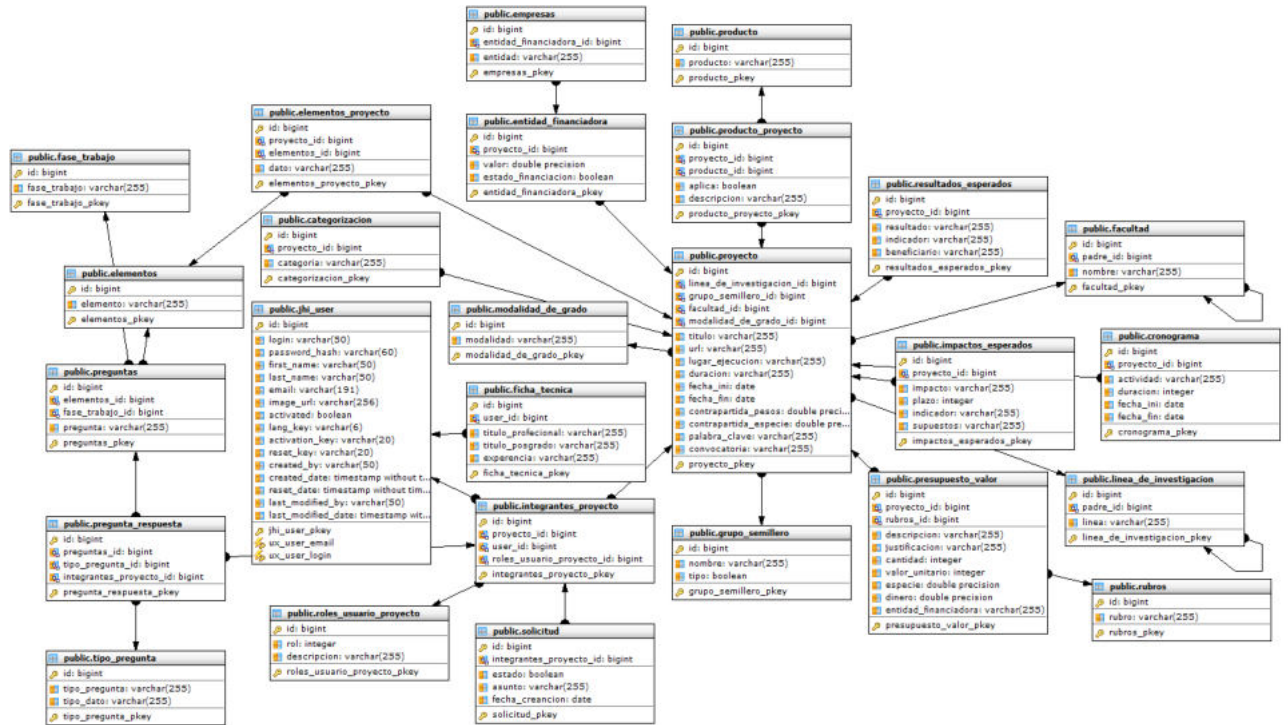
Imagen 31 Servicio de angular que se conecta al api RESTful por medio de la URL

```
12  type EntityResponseType = HttpResponse<IProyecto>;
13  type EntityArrayResponseType = HttpResponse<IProyecto[]>;
14
15  @Injectable({ providedIn: 'root' })
16  export class ProyectoService {
17    public resourceUrl = SERVER_API_URL + 'api/proyectos';
18
19    constructor(protected http: HttpClient) {}
20
21    create(proyecto: IProyecto): Observable<EntityResponseType> {
22      const copy = this.convertDateFromClient(proyecto);
23      return this.http
24        .post<IProyecto>(this.resourceUrl, copy, { observe: 'response' })
25        .pipe(map((res: EntityResponseType) => this.convertDateFromServer(res)));
26    }
27
28    update(proyecto: IProyecto): Observable<EntityResponseType> {
29      const copy = this.convertDateFromClient(proyecto);
30      return this.http
31        .put<IProyecto>(this.resourceUrl, copy, { observe: 'response' })
32        .pipe(map((res: EntityResponseType) => this.convertDateFromServer(res)));
33    }
34
35    find(id: number): Observable<EntityResponseType> {
36      return this.http
37        .get<IProyecto>(`${this.resourceUrl}/${id}`, { observe: 'response' })
38        .pipe(map((res: EntityResponseType) => this.convertDateFromServer(res)));
39    }

```

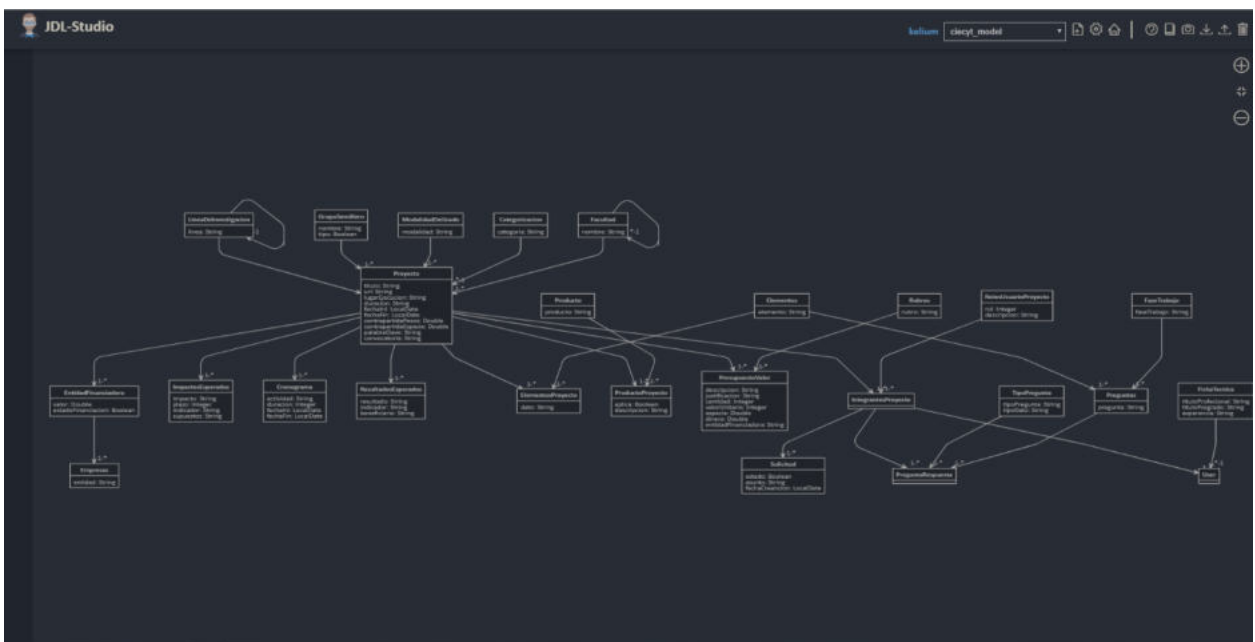
Fuente: Los autores

Imagen 32 Diagrama del diseño de la base de datos en postgresql



Fuente: Los autores

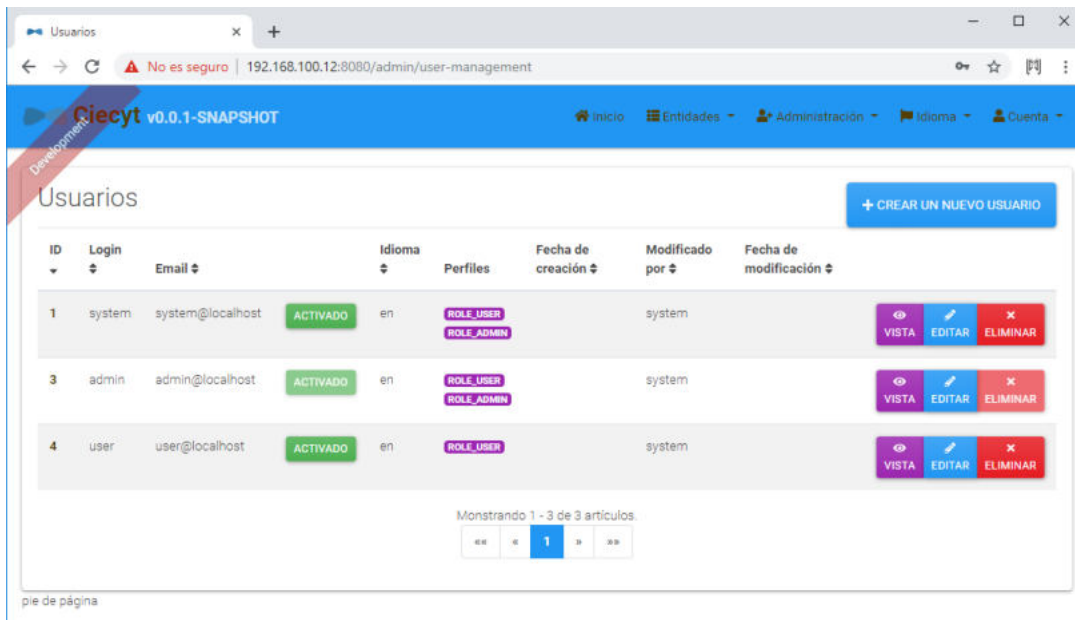
Imagen 33 Diagrama de la base de datos con lenguaje JDL



Fuente: Los autores

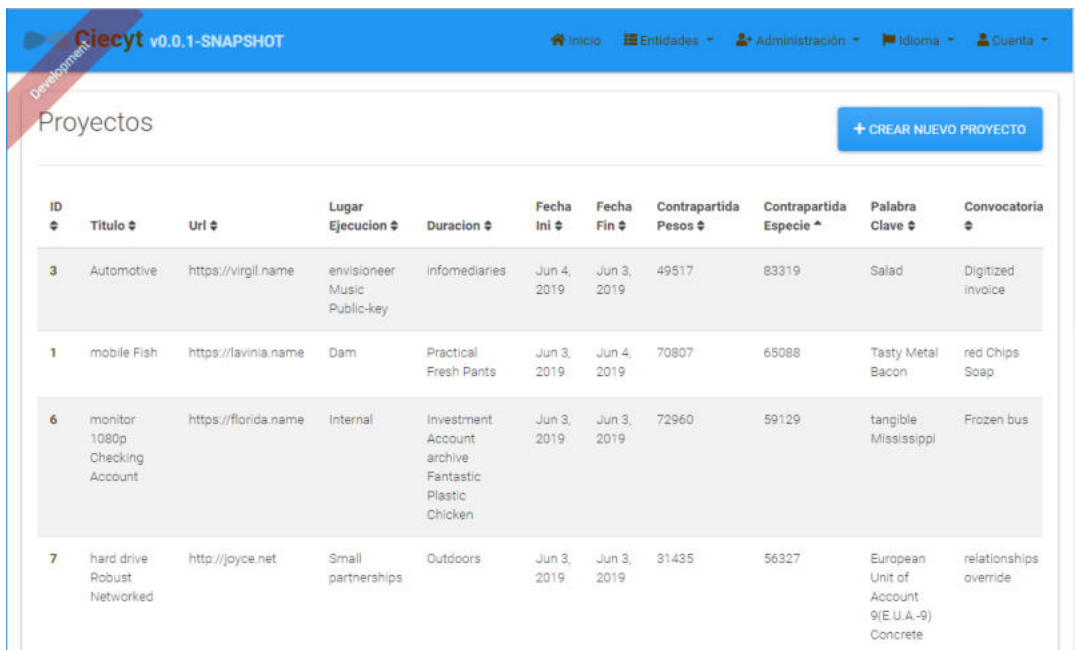
## Vistas principales

Imagen 34 Crud de la tabla usuarios



Fuente: Los autores

Imagen 35 Crud de la tabla proyectos



Fuente: Los autores

Imagen 36 Crud de la tabla solicitudes

**Solicitudes** + CREAR NUEVO SOLICITUD

ID	Estado	Asunto	Fecha Creacion	Integrantes Proyecto
1	true	Money Market Account Credit Card Account withdrawal	Jun 4, 2019	VISTA EDITAR ELIMINAR
2	true	navigating seize generate	Jun 3, 2019	VISTA EDITAR ELIMINAR
3	false	Chips microchip	Jun 4, 2019	VISTA EDITAR ELIMINAR
4	false	compress Fantastic Fresh Ball	Jun 3, 2019	VISTA EDITAR ELIMINAR
5	false	monitor methodology West Virginia	Jun 3, 2019	VISTA EDITAR ELIMINAR
6	false	Factors Soft	Jun 3, 2019	VISTA EDITAR ELIMINAR
7	false	Beauty	Jun 3, 2019	VISTA EDITAR ELIMINAR
8	false	backing up indexing	Jun 3, 2019	VISTA EDITAR ELIMINAR
9	true	extensible	Jun 3, 2019	VISTA EDITAR ELIMINAR

Fuente: Los autores

Imagen 37 Crud de la tabla presupuesto valor.

**Presupuesto Valores** + CREAR NUEVO PRESUPUESTO VALOR

ID	Descripcion	Justificacion	Cantidad	Valor Unitario	Especie	Dinero	Entidad Financiadora	Proyecto	Rubros
1	Principal Senior	Paraguay	96930	7543	52309	16621	magenta heuristic		VISTA EDITAR ELIMINAR
2	Credit Card Account Maryland Legacy	Product	95870	95041	5786	89026	Court Money Market Account		VISTA EDITAR ELIMINAR
3	Mouse	Junctions one-to-one	58365	26593	7773	36034	zero defect Ridges Beauty		VISTA EDITAR ELIMINAR
4	groupware	Credit Card Account clear-thinking Frozen	7620	82253	94053	94979	hard drive Personal Loan Account		VISTA EDITAR ELIMINAR
5	Unbranded SCSI	Rustic Steel Shirt drive	81055	48037	98727	98528	engage		VISTA EDITAR ELIMINAR
6	disintermediate Assistant	Practical Steel Shirt Plastic	7535	86472	69171	81290	Producer Agent		VISTA EDITAR ELIMINAR

Fuente: Los autores

Imagen 38 Vista para crear o editar una solicitud

The screenshot shows a web application interface for creating or editing a request. The header is blue with the Giecyt logo and version 'v0.0.1-SNAPSHOT'. A red 'Development' banner is on the left. The main content area is titled 'Crear o editar Solicitud' and contains several input fields: 'Estado' with a checkmark icon, 'Asunto', 'Fecha Creacion' with a calendar icon, and 'Integrantes Proyecto' with a dropdown arrow. At the bottom are 'CANCELAR' and 'GUARDAR' buttons. A small 'pie de página' label is at the bottom left.

Fuente: Los autores

Imagen 39 Vista para crear o editar el presupuesto

The screenshot shows a web application interface for creating or editing a budget value. The header is blue with the Giecyt logo and version 'v0.0.1-SNAPSHOT'. A red 'Development' banner is on the left. The main content area is titled 'Crear o editar Presupuesto Valor' and contains several input fields: 'Descripcion', 'Justificacion', 'Cantidad', 'Valor Unitario', 'Especie', and 'Dinero'. A vertical scrollbar is visible on the right side.

Fuente: Los autores

Imagen 40 Vista para crear o editar un usuario

The screenshot shows a web application interface for creating or editing a user. The header is blue with the text 'Ciecyt v0.0.1-SNAPSHOT' and navigation links: 'Inicio', 'Entidades', 'Administración', 'Idioma', and 'Cuenta'. A red diagonal banner on the left says 'Development'. The main content area is titled 'Crear o editar un usuario' and contains the following fields: 'Login', 'Nombre', 'Apellidos', 'Email', a checkbox for 'Activado', a dropdown for 'Idioma', and a dropdown for 'Perfiles'.

Fuente: Los autores

Imagen 41 Vista para crear o editar un proyecto

The screenshot shows a web application interface for creating or editing a project. The header is blue with the text 'Ciecyt v0.0.1-SNAPSHOT' and navigation links: 'Inicio', 'Entidades', 'Administración', 'Idioma', and 'Cuenta'. A red diagonal banner on the left says 'Development'. The main content area is titled 'Crear o editar Proyecto' and contains the following fields: 'Titulo', 'Url', 'Lugar Ejecucion', 'Duracion', 'Fecha Ini' (with a calendar icon), and 'Fecha Fin' (with a calendar icon).

Fuente: Los autores

## **Conclusiones**

La metodología ICONIX permitió diseñar de manera completa y clara las diferentes fases de análisis y diseño que se requieren para atender los requerimientos que tienen los Stakeholders de trabajos de grado en el ITP.

El análisis de procesos y establecimiento de requerimientos en la gestión y administración de trabajos de grado en el ITP, permitió comprender las falencias que se presentan en este proceso al desarrollarse de manera manual, y la pérdida de información que se puede presentar.

El fortalecimiento en la gestión y administración de trabajos de grado, permitirá a la institución mostrar de manera protegida los diferentes procesos de investigación que se llevan a cabo por parte de sus estudiantes, contrario a lo que se presenta actualmente donde los diferentes trabajos de grado son desconocidos.

En el desarrollo de la propuesta se adquirieron nuevos conocimientos y herramientas que refuerzan el aprendizaje adquirido durante el transcurso de nuestra carrera universitaria permitiendo así, a aplicar lo aprendido en nuevos proyectos en el futuro.

Se cumplió con el objetivo de realizar un prototipo con funcionalidades básicas como crear, actualizar, eliminar y visualizar la información de las principales tablas de la base de datos, entre las cuales tenemos: tabla usuarios, proyecto, presupuesto entre otras.

## **Recomendaciones**

Se recomienda extender las funcionalidades del prototipo para que evolucionen a un software completo y robusto para la gestión de trabajos de grado, claro está basándose en el análisis y diseño que se plasma en los diagramas obtenidos con la metodología ICONIX.

Realizar el análisis de las modalidades de trabajo de grado no tratadas en el presente documento, con el fin de realizar el acompañamiento adecuado durante todo el proceso de inscripción y desarrollo.

Crear un módulo que permita la publicación protegida de los trabajos de grado desarrollados con el fin de mostrar a la institución en sus diferentes campos de investigación.

## Bibliografía

- Anonimo. (23 de junio de 1989). *derechodeautor.gov.co*. Recuperado el 15 de 01 de 2019, de [derechodeautor.gov.co](http://derechodeautor.gov.co):  
<http://derechodeautor.gov.co/documents/10181/187670/DECRETO+1360+DE+1989.pdf/db076bd8-f8b8-41d2-8f39-76faa2e212db>
- Anonimo. (17 de Diciembre de 1993). *www.wipo.int*. Obtenido de Decision 351 - Régimen Común sobre Derecho de Autor y Derechos Conexos:  
<https://www.wipo.int/edocs/lexdocs/laws/es/can/can010es.pdf>
- apache. (s.f.). *apache maven project*. Obtenido de apache maven project :  
<https://maven.apache.org/>
- Asamblea Nacional Constituyente. (1991). *Secretaria del Senado de la República*. Recuperado el 3 de 04 de 2019, de [http://www.secretariasenado.gov.co/senado/basedoc/constitucion\\_politica\\_1991.html](http://www.secretariasenado.gov.co/senado/basedoc/constitucion_politica_1991.html)
- Azaustre, C. (2014). *Desarrollo web ágil con Angular.js*.
- Bellido, J. A. (2011). *Web Linking-based protocols for guiding RESTful M2M interaction*.
- Berenguer, F. (2014). *Riunet*. Recuperado el 23 de 09 de 2018, de [https://riunet.upv.es/bitstream/handle/10251/49757/MEMORIA\\_Barba%20Soler%2C%20Juan%20Pedro.pdf?sequence=1](https://riunet.upv.es/bitstream/handle/10251/49757/MEMORIA_Barba%20Soler%2C%20Juan%20Pedro.pdf?sequence=1)
- Chiza, A. F. (2018). *UNIVERSIDAD TÉCNICA DEL NORTE*. Obtenido de ANÁLISIS DE RENDIMIENTO ENTRE UNA ARQUITECTURA MONOLITICA Y UNA ARQUITECTURA DE MICROSERVICIO: <file:///C:/Users/edin-/Downloads/PG%20694%20Trabajo%20de%20Grado.pdf>
- Chiza., A. F. (2018). *ANÁLISIS DE RENDIMIENTO ENTRE UNA ARQUITECTURA MONOLITICA Y ARQUITECTURA DE MICROSERVICIOS TECNOLOGIA BASADA EN CONTENEDORES*. Obtenido de <file:///C:/Users/edin-/Downloads/PG%20694%20Trabajo%20de%20Grado.pdf>

CISET. (2019). *CENTRO DE INNOVACION Y SOLUCIONES EMPRESARIALES Y TECNOLOICAS*. Obtenido de <https://www.ciset.es/>

Congreso de la Republica. (28 de 01 de 1982). *derechosdeautor.gov.con*. Obtenido de [derechosdeautor.gov.con](http://derechosdeautor.gov.con):  
<http://derechosdeautor.gov.co/documents/10181/182597/23.pdf/a97b8750-8451-4529-ab87-bb82160dd226>

Congreso de la República. (1992). *Consejo Nacional de Acreditación*. Recuperado el 13 de 02 de 2019, de [https://www.cna.gov.co/1741/articles-186370\\_ley\\_3092.pdf](https://www.cna.gov.co/1741/articles-186370_ley_3092.pdf)

Ferrer, J., García, V., & García, R. (s.f.). *Curso completo de HTML*.

Gauchat, J. D. (2012). *El gran libro de HTML5, CSS3 y Javascript*. MARCOMBO.

Gonzales, A. (2011). *Universidad Nacional de Colombia*. Obtenido de <file:///C:/Users/LUIS%20CARLOS/Downloads/Registro%20de%20Trabajo%20de%20Grado.pdf>

GONZALEZ, G. M. (SIN FECHA). *Aprende a Desarrollar con Spring Framework*. Obtenido de *Aprende a Desarrollar con Spring Framework*:  
[https://books.google.es/books?hl=es&lr=&id=4BbfDQAAQBAJ&oi=fnd&pg=PA6&dq=+ventajas+de+spring+framework&ots=V0sKcJ\\_WPR&sig=HLA4LUd9GAe1ujjaRmzGBAoMAks#v=onepage&q=ventajas%20de%20spring%20framework&f=false](https://books.google.es/books?hl=es&lr=&id=4BbfDQAAQBAJ&oi=fnd&pg=PA6&dq=+ventajas+de+spring+framework&ots=V0sKcJ_WPR&sig=HLA4LUd9GAe1ujjaRmzGBAoMAks#v=onepage&q=ventajas%20de%20spring%20framework&f=false)

Hoyos, P. (s/f). *PacifiTIC*. Obtenido de <https://pacifitic.org/que-es-el-testing-de-software-y-porque-es-tan-importante-en-el-desarrollo-de-software/>

ieec. (s.f.). *Systems and software engineering — Architecture description ISO/IEC/IEEE 42010*. Obtenido de *Systems and software engineering — Architecture description ISO/IEC/IEEE 42010*: <http://www.iso-architecture.org/ieec-1471/defining-architecture.html>

ITP. (2007). *Instituto Tecnológico del Putumayo*. Recuperado el 07 de 03 de 2019, de <file:///C:/Users/LUIS%20CARLOS/Downloads/acuerdo%20de%20lineas%20de%20investigacion%20-%20copia.pdf>

ITP. (s/f). *Instituto Tecnológico del Putumayo*. Obtenido de <http://www.itp.edu.co/web2016/index.php/la-universidad/acerca-del-itp/historia>

jhipster. (s.f.). *jhipster*. Obtenido de jhipster: <https://www.jhipster.tech/using-angular/>

JOSE MIGUEL ORDAX CASSÁ, PILAR ARANZAZU DIAZ-UFANO. (SIN FECHA). *Ministerio de Educacion, Cultura y Deporate*. Obtenido de Ministerio de Educacion, Cultura y Deporate: [https://books.google.es/books?hl=es&lr=&id=ry8bAgAAQBAJ&oi=fnd&pg=PP1&dq=1+enuaje+de+programacion+java&ots=Q6dpDux\\_9z&sig=oJrXJXblAsDcRrrSuSUfz8Gj94A#v=onepage&q&f=false](https://books.google.es/books?hl=es&lr=&id=ry8bAgAAQBAJ&oi=fnd&pg=PP1&dq=1+enuaje+de+programacion+java&ots=Q6dpDux_9z&sig=oJrXJXblAsDcRrrSuSUfz8Gj94A#v=onepage&q&f=false)

Mineducación. (2003). *Ministerio de Educación*. Recuperado el 20 de 02 de 2019, de [https://www.mineducacion.gov.co/1621/articles-86425\\_Archivo\\_pdf.pdf](https://www.mineducacion.gov.co/1621/articles-86425_Archivo_pdf.pdf)

Nieto Rodrigo Jorge, M.-R. R. (2016). Obtenido de <https://riunet.upv.es/bitstream/handle/10251/75120/NIETO%20-%20Desarrollo%20de%20una%20aplicaci%C3%B3n%20web%2C%20con%20Front-end%20y%20Back-end%20para%20compraventa%20de%20segunda%20mano.pdf?sequence=1>

Nieto Rodrigo, J. (2016). *UNIVERSIDAD POLITECNICA DE VALENCIA*. Obtenido de *UNIVERSIDAD POLITECNICA DE VALENCIA*: <https://riunet.upv.es/bitstream/handle/10251/75120/NIETO%20-%20Desarrollo%20de%20una%20aplicaci%C3%B3n%20web,%20con%20Front-end%20y%20Back-end%20para%20compraventa%20de%20segunda%20mano.pdf?sequence=1>

Pérez, J. E. (2009). *Introducción a CSS*.

Pontificia Universidad Javeriana. (s/f). *Universidad Javeriana*. Obtenido de <https://www.javeriana.edu.co/migrado/pasos>

Pressman, R. S. (2010). *INGENIERÍA DEL SOFTWARE. UN ENFOQUE PRÁCTICO*. México : McGraw-hill.

- Pressman, R. S. (2010). *INGENIERÍA DEL SOFTWARE. UN ENFOQUE PRÁCTICO*. MEXICO: McGraw-hill.
- Raible, M. (13 de 06 de 2019). *DocGist*. Obtenido de DocGist: <http://gist.asciidoctor.org/?github-mraible/jhipster5-demo//README.adoc>
- Raible, M. (2019). *The JHipster Mini-Book*. C4Media.
- Red Tercer Milenio S.C. (2012). *ANÁLISIS DE SISTEMAS DE INFORMACIÓN*.
- Richards, M. (2015). *Software Architecture Patterns*.
- Roger S. Pressman, P. (s.f.). *INGENIERÍA DEL SOFTWARE. UN ENFOQUE PRÁCTICO*. The McGraw-Hill.
- Scott , K., & Rosenberg, D. (1999). *Use Case Driven Object Modeling with UML: A Practical Approach (Object Technology Series)*.
- Uriarte, J. (2017). *Características*. Recuperado el 09 de 01 de 2019, de <https://www.caracteristicas.co/software/>
- Uriarte, J. M. (27 de septiembre de 2017). *Características*. Obtenido de <https://www.caracteristicas.co/software/>
- Wahlen, D. O. (2014). *Testdrive AngularJS with Spring 4* .
- webpack. (s.f.). *webpack*. Obtenido de webpack: <https://webpack.js.org/concepts>
- Yera, Á. C. (s.f.). *Diseño y Prgramacion de Bases de Datos* . Obtenido de Diseño y Prgramacion de Bases de Datos : <https://books.google.es/books?hl=es&lr=&id=anCDr9N-kGsC&oi=fnd&pg=PA7&dq=bases+de+datos&#v=onepage&q&f=false>

## Anexos

### Anexo 1. Diccionario de Datos




#### 1.1. : public.categorizacion (categorias de evaluacion)

En esta tabla se establecerán los distintos estados de la propuesta como:

- viable
- pendiente
- no viable
- corregida
- en correccion

Entre otros que puedan requerirse.

#### Columns

	Name	Data type	Description / Attributes
	id	bigint	Llave primaria
	categoria (Categorizacon de la propuesta)	character varying(255)	<b>Nullable</b>
	proyecto_id (Propuesta asociada)	bigint	<b>Nullable</b> <b>References:</b> public.proyecto






#### 1.1. Tabla: public.cronograma (Cronograma de actividades)

Aquí se guardará cada actividad a realizar de una propuesta relacionada especificando las fechas de cuando inicia y termina;

los campos principales de la tabla son:

- actividad
- duracion

#### Columnas

	Name	Data type	Description / Attributes
	id	bigint	Llave primaria autoincremental
	actividad (Actividad a desarrollar)	character varying(255)	<b>Nullable</b>
	duracion (Tiempo de realizacion)	integer	La duracion de la actividad en meses <b>Nullable</b>
	fecha_ini (fecha de inicio)	date	<b>Nullable</b>
	fecha_fin (fecha final )	date	<b>Nullable</b>

**1.1.** Tabla: public.databasechangelog (Auditoria de la base de datos)

En esta tabla se guardara cada accion del usuario desde la creacion, actualizacion y eliminacion tecnicamente funciona como una tabla de auditoria para cada tabla de la base de datos. Sus principales columnas son :

- autor
- descripcion
- etiqueta

Columns

	Name	Data type	Description / Attributes
	id	character varying(255)	
	author (Autor )	character varying(255)	
	filename (Nombre del archivo)	character varying(255)	
	dateexecuted	timestamp without time zone	
	orderexecuted	integer	
	exectype	character varying(10)	
	md5sum	character varying(35)	<b>Nullable</b>
	description (descripcion del cambio)	character varying(255)	<b>Nullable</b>
	comments (comentarios)	character varying(255)	<b>Nullable</b>
	tag (etiqueta)	character varying(255)	<b>Nullable</b>
	liquibase	character varying(20)	<b>Nullable</b>
	contexts	character varying(255)	<b>Nullable</b>
	labels	character varying(255)	<b>Nullable</b>
	deployment_id	character varying(10)	<b>Nullable</b>





**1.1.** Table: public.elementos (Elementos de la propuesta)

Columns

	Name	Data type	Description / Attributes
	id	bigint	
	elemento (Elemento propuesta)	character varying(255)	Son los elementos de la propuesta de acuerdo al documento F-INV-008 <b>Nullable</b>

#### 1.1. public.empresas (Empresas Asociadas)

##### Columns





	Name	Data type	Description / Attributes
 	id	bigint	
	entidad (Empresa)	character varying(255)	<b>Nullable</b>
	entidad_financiadora_id	bigint	<b>Nullable</b> <b>References:</b> public.entidad_financiadora

#### 1.1. Table: public.entidad\_financiadora (Empresas financiadoras)

En esta tabla se guardan todas las empresas que podrian financiar la propuesta del estudiante para el desarrollo de la misma. Sus principales columnas son:

- **entidad**
- **valor**
- **estado**
- **propuesta**

##### Columns

	Name	Data type	Description / Attributes
 	id	bigint	
	valor (Dinero aportado)	double precision	El monto economico de ayuda a la propuesta <b>Nullable</b>
	estado_financiacion (Estado del proceso)	boolean	<b>Nullable</b>





**1.1.** Table: public.facultad (Facultades y programas)

En esta tabla se guardaran todas las facultades y programas de la institucion las cuales podran ser referenciadas por la proupesta. las columnas principales de esta tabla son:

- **Facultad o programa**
- **Facultad padre(Programa)**

Cabe aclarar que en esta tabla se usa recursion es decir que la se referencia asi misma.

Columns







	Name	Data type	Description / Attributes
 	id	bigint	Llave primaria autoincremental
	nombre (Facultad o programa)	character varying(255)	En este campo puede colocarse el nombre de una facultad o un programa. <b>Nullable</b>
	padre_id (Facultad)	bigint	Una facultad tiene programas <b>Nullable</b> <b>References:</b> public.facultad

**1.1.** Table: public.ficha\_tecnica (Fhica tecnica del jurado)

En esta tabla se especificaran algunos datos de caracter tecnico educativo del jurado, a modo de referencia para la asignacion de una propuesta. sus principales campos son :

- **titulo profesional**
- **experiencia**

Columns







	Name	Data type	Description / Attributes
 	id	bigint	
	titulo_profecional	character varying(255)	<b>Nullable</b>
	titulo_posgrado	character varying(255)	<b>Nullable</b>
	experiencia	character varying(255)	<b>Nullable</b>
	user_id	bigint	<b>Nullable</b> <b>References:</b> public.jhi_user

**1.1.** Table: public.impactos\_esperados (Impactos esperados de la propuesta)

Se definiran uno a uno los impactos que se esperan de una propuesta asociada a la tabla, sus principales columnas son:

- **impacto**
- **plazo**
- **supuesto**

Columns





	Name	Data type	Description / Attributes
	id	bigint	Llave primaria incremental
	impacto	character varying(255)	<b>Nullable</b>
	plazo	integer	Tiempo en meses <b>Nullable</b>
	indicador	character varying(255)	<b>Nullable</b>
	supuestos	character varying(255)	<b>Nullable</b>
	proyecto_id	bigint	Referencia a la propuesta <b>Nullable</b> <b>References:</b> public.proyecto

**1.1.** Table: public.integrantes\_proyecto (Integrantes del proyecto de invsetigacion)

En esta tabla se guardaran los integrantes de una propuesta como jurado, asesor y estudiantes entre otros. Sus principales campos son:

- **usuario**
- **propuesta**

Columns


	Name	Data type	Description / Attributes
	id	bigint	
	proyecto_id (propuesta de investigacion)	bigint	Referencia a la propuesta <b>Nullable</b> <b>References:</b> public.proyecto
	user_id (usuario)	bigint	Referencia a los usuarios registrados <b>Nullable</b> <b>References:</b> public.jhi_user
	roles_usuario_proyecto_id (Roles en la propuesta)	bigint	roles dentro de la propuesta <b>Nullable</b> <b>References:</b> public.roles_usuario_proyecto

**1.1.** Table: public.jhi\_authority (Roles dentro del sistema)

En esta tabla se especifican los roles globales como

- asesor
- Jurado
- estudiante
- anonimo
- admin
- Administrador ciecyt

Columns






	Name	Data type	Description / Attributes
 	name	character varying(50)	

**1.1.** Table: public.jhi\_persistent\_audit\_event (Auditoria de eventos)

Se guardan los eventos realizados dentro del sistema por ejemplo si se crea, actualiza, elimina etc, sus principales columnas son:

- **tipo de evento**
- **fecha**






Columns

	Name	Data type	Description / Attributes
 	event_id	bigint	
	principal	character varying(50)	
	event_date (Fecha de accion)	timestamp without time zone	<b>Nullable</b>
	event_type (tipo de evento)	character varying(255)	<b>Nullable</b>

**1.1.** Table: public.jhi\_persistent\_audit\_evt\_data (Auditoria de datos)

En esta tabla se guarda el valor y el nombre del campo afectado por el usuario.

















Columns

	Name	Data type	Description / Attributes
 	event_id	bigint	<b>References:</b> public.jhi_persistent_audit_event
 	name	character varying(150)	
	value	character varying(255)	<b>Nullable</b>

**1.1.** Table: public.jhi\_user (Usuarios del sistema)

En esta tabla se guardaran los datos basicos del los usuarios del sistema como su nombre, login y contraseña entre otros.



Columns

	Name	Data type	Description / Attributes
	id	bigint	
	login (nombre de usuario)	character varying(50)	
	password_hash (contraseña)	character varying(60)	
	first_name (nombre)	character varying(50)	<b>Nullable</b>
	last_name (apellido)	character varying(50)	<b>Nullable</b>
	email	character varying(191)	<b>Nullable</b>
	image_url	character varying(256)	<b>Nullable</b>
	activated	boolean	
	lang_key	character varying(6)	<b>Nullable</b>
	activation_key	character varying(20)	<b>Nullable</b>
	reset_key	character varying(20)	<b>Nullable</b>
	created_by	character varying(50)	
	created_date	timestamp without time zone	<b>Nullable</b>
	reset_date	timestamp without time zone	<b>Nullable</b>
	last_modified_by	character varying(50)	<b>Nullable</b>
	last_modified_date	timestamp without time zone	<b>Nullable</b>

**1.1.** Table: public.jhi\_user\_authority (Roles del usuario dentro del sistema)

Aqui se especifica el grado de permisos que tendra cada usuario de acuerdo al rol que se le asigne.



Columns

	Name	Data type	Description / Attributes
	user_id (Usuario)	bigint	<b>References:</b> public.jhi_user
	authority_name (Rol dentro del sistema)	character varying(50)	<b>References:</b> public.jhi_authority

**1.1.** Table: public.linea\_de\_investigacion (Linea y sublinea de investigacion)





En esta tabla recursiva se guarda la línea o sublínea de investigación y se referencia a una línea de investigación de acuerdo a las necesidades.

Columns

	Name	Data type	Description / Attributes
	id	bigint	
	linea (Linea o sublinea)	character varying(255)	Nombre de la línea o sublínea de investigación <b>Nullable</b>

**1.1.** Table: public.pregunta\_respuesta (Revision de la propuesta)





En esta tabla se guardaran las explicaciones de los jurados a preguntas que se proponen de acuerdo a un elemento de la propuesta.

	Name	Data type	Description / Attributes
	id	bigint	
	preguntas_id (Pregunta)	bigint	<b>Nullable</b> <b>References:</b> public.preguntas
	tipo_pregunta_id (tipo de pregunta)	bigint	<b>Nullable</b> <b>References:</b> public.tipo_pregunta
	integrantes_proyecto_id (Usuario)	bigint	<b>Nullable</b> <b>References:</b> public.integrantes_proyecto

**1.1.** Table: public.preguntas (Preguntas de elementos)

En esta tabla se guardaran todas las posibles preguntas que se puedan hacer para un elemento.

Columns












	Name	Data type	Description / Attributes
	id	bigint	
	pregunta	character varying(255)	<b>Nullable</b>
	elementos_id (elementos)	bigint	<b>Nullable</b> <b>References:</b> public.elementos
	fase_trabajo_id (fase propuesta)	bigint	<b>Nullable</b> <b>References:</b> public.fase_trabajo

**1.1.** Table: public.presupuesto\_valor (Presupuesto de la propuesta)

Se guardaran los principales aspectos del presupuesto de una propuesta en base a los rubros seleccionados, sus principales campos son:

- **rubro**
- **descripcion**
- **cantidad**
- **valor unitario**




Columns

	Name	Data type	Description / Attributes
 	id	bigint	
	descripcion	character varying(255)	<b>Nullable</b>
	justificacion	character varying(255)	<b>Nullable</b>
	cantidad	integer	<b>Nullable</b>
	valor_unitario	integer	<b>Nullable</b>
	especie	double precision	<b>Nullable</b>
	dinero	double precision	<b>Nullable</b>
	entidad_financiadora (entidad financiadora)	character varying(255)	<b>Nullable</b>
	proyecto_id (Propuesta)	bigint	<b>Nullable</b> <b>References:</b> public.proyecto
	rubros_id (rubros)	bigint	<b>Nullable</b> <b>References:</b> public.rubros

**1.1.** Table: public.producto (Productos )

En esta tabla se guardaran los productos finales como : articulo, libro, norma tecnica etc; establecidos por el instituto.

Columns







	Name	Data type	Description / Attributes
 	id	bigint	
	producto	character varying(255)	<b>Nullable</b>

1.1. Table: public.producto\_proyecto (Productos esperados de la propuesta)

Aqui se guardaran todos los productos esperados de una propuesta, las principales columnas de esta tabla son:

- **producto**
- **propuesta**
- **descripcion**

Columns











	Name	Data type	Description / Attributes
 	id	bigint	
	aplica	boolean	0 no aplica, 1 aplica <b>Nullable</b>
	descripcion	character varying(255)	<b>Nullable</b>
	proyecto_id (propuesta de investigacion)	bigint	<b>Nullable</b> <b>References:</b> public.proyecto
	producto_id (Productos)	bigint	<b>Nullable</b> <b>References:</b> public.producto

#### 1.1. Table: public.proyecto (Proyecto investigativo)

En esta tabla se guardaran todos los aspectos generales de una propuesta basandonos en el formato F-INV-008. Sus principales campos son:

- **titulo**
- **modalidad de grado**
- **linea y sublinea de investigacion**
- **categoria**
- **url**

#### Columns






	Name	Data type	Description / Attributes
 	id	bigint	
	titulo (Titulo de la propuesta)	character varying(255)	<b>Nullable</b>
	url	character varying(255)	Ruta donde se guardara la version final del proyecto <b>Nullable</b>
	lugar_ejecucion (Lugar de ejecucion)	character varying(255)	<b>Nullable</b>
	duracion	character varying(255)	<b>Nullable</b>
	fecha_ini (fecha de inicio)	date	<b>Nullable</b>
	fecha_fin (Fecha de finalizacion)	date	<b>Nullable</b>
	contrapartida_pesos	double precision	<b>Nullable</b>
	contrapartida_especie	double precision	<b>Nullable</b>
	palabra_clave	character varying(255)	<b>Nullable</b>
	convocatoria	character varying(255)	<b>Nullable</b>
	linea_de_investigacion_id (Linea y sublinea de investigacion)	bigint	<b>Nullable</b> <b>References:</b> public.linea_de_investigacion
	grupo_semillero_id (Grupo o semillero)	bigint	<b>Nullable</b> <b>References:</b> public.grupo_semillero
	facultad_id (Facultad y Programa)	bigint	<b>Nullable</b> <b>References:</b> public.facultad
	modalidad_de_grado_id (Modalidad de grado)	bigint	<b>Nullable</b> <b>References:</b> public.modalidad_de_grado

**1.1.** Table: public.resultados\_esperados (Resultados esperados de la propuesta)

Aquí guardaremos cada resultado esperado de una propuesta que se asigne donde sus campos más importantes son:

- **resultado esperado**
- **propuesta de investigación**




Columns

	Name	Data type	Description / Attributes
	id	bigint	
	resultado (resultado esperado)	character varying(255)	<b>Nullable</b>
	indicador	character varying(255)	<b>Nullable</b>
	beneficiario	character varying(255)	<b>Nullable</b>
	proyecto_id (Propuesta de investigación)	bigint	<b>Nullable</b> <b>References:</b> public.proyecto

**1.1.** Table: public.roles\_usuario\_proyecto (Roles dentro de la propuesta)

En esta tabla se especificarán los posibles roles que tenga un usuario dentro de una propuesta.



Columns

	Name	Data type	Description / Attributes
	id	bigint	
	rol (rol en la propuesta)	integer	<b>Nullable</b>
	descripcion (Descripción del rol)	character varying(255)	se describe que acciones puede realizar el usuario al que se le asigne el rol. <b>Nullable</b>

**1.1.** Table: public.rubros (Rubros)

Esta tabla guardará los rubros especificados por el instituto tales como: Personal, viajes, servicios técnicos, equipos y software entre otros.

Columns







	Name	Data type	Description / Attributes
	id	bigint	
	rubro	character varying(255)	<b>Nullable</b>

**1.1.** Table: public.solicitud (Solicitud de novedades)

En esta tabla se guardaran todas las solicitudes ya sea de cambios o renuncias de algun tema en especifico. sus campos principales son:

- **Asunto**
- **Estado**
- **Fecha**





Columns

	Name	Data type	Description / Attributes
 	id	bigint	
	estado (Estado solicitud)	boolean	0 es rechazado, 1 es aceptado <b>Nullable</b>
	asunto	character varying(255)	<b>Nullable</b>
	fecha_creacion (fecha de creacion)	date	<b>Nullable</b>
	integrantes_proyecto_id (Integrantes de la propuesta)	bigint	<b>Nullable</b> <b>References:</b> public.integrantes_proyecto

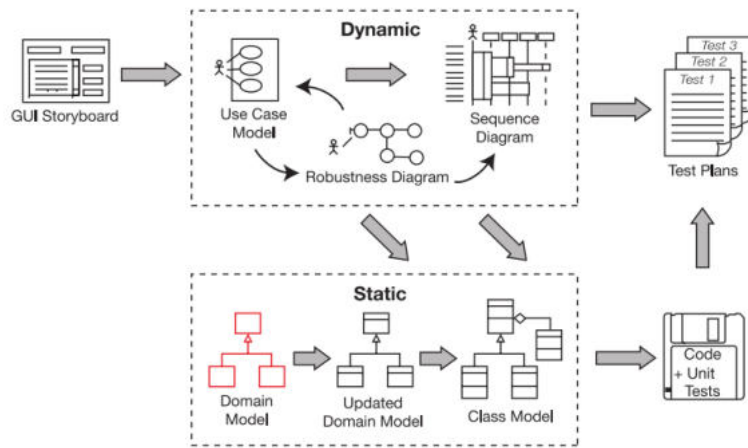
**1.1.** Table: public.tipo\_pregunta (Tipos de preguntas)

En esta tabla guardaremos los tipos de preguntas o variaciones de acuerdo al documento F-INV-033 y el documento F-INV-014.

Columns

	Name	Data type	Description / Attributes
 	id	bigint	
	tipo_pregunta (Tipo de pregunta)	character varying(255)	<b>Nullable</b>
	tipo_dato (tipo de dato)	character varying(255)	Se especifica si sera un string, integer, date, booleano entre otros <b>Nullable</b>

## Anexo 2 Diagrama de la Metodología ICONIX



(Scott & Rosenberg, 1999)

## Anexo 3 Ejemplo de un diagrama de modelo de dominio

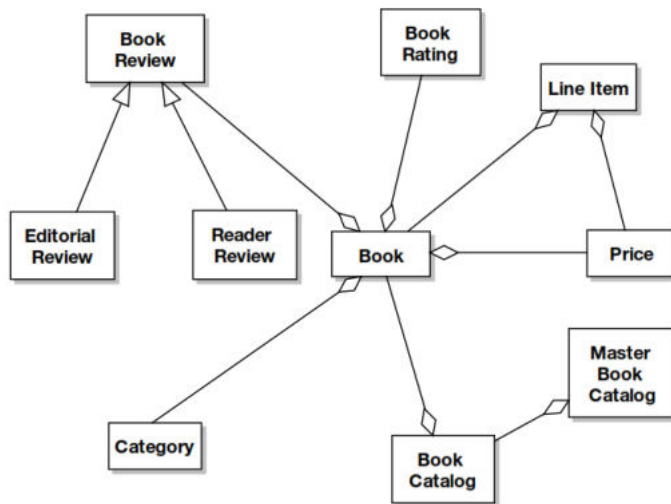


Figure 2-1. Example of a domain model diagram

(Scott & Rosenberg, 1999)

Anexo 4 Ejemplo de diagrama de casos de uso

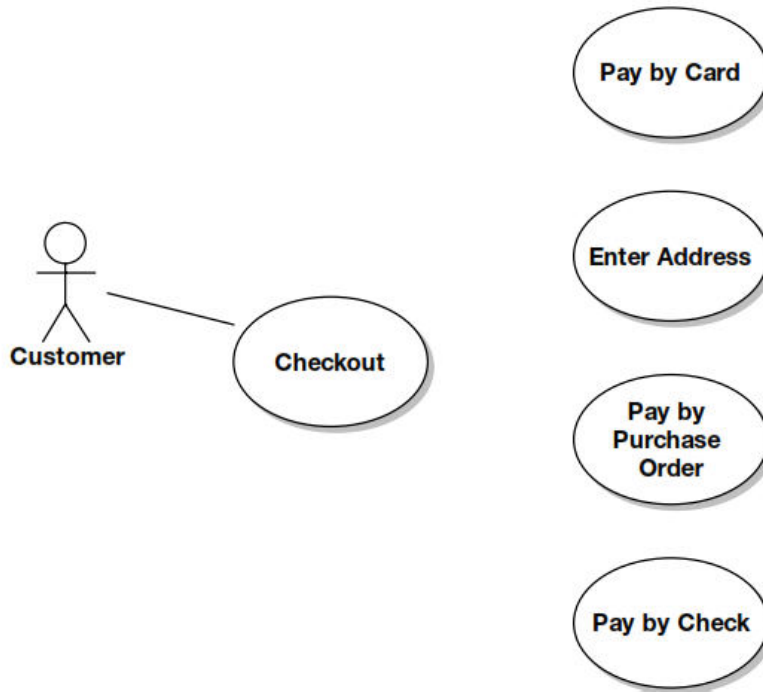


Figure 3-1. Example use case diagram

(Scott & Rosenberg, 1999)

Anexo 5 Ejemplo de un diagrama de robustez correcto

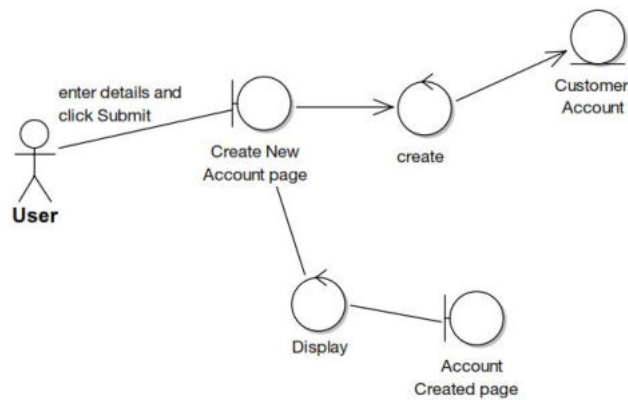


Figure 5-25. The corrected robustness diagram excerpt for Exercise 5-1

(Scott & Rosenberg, 1999)

## Anexo 6 Elementos de un diagrama de secuencia

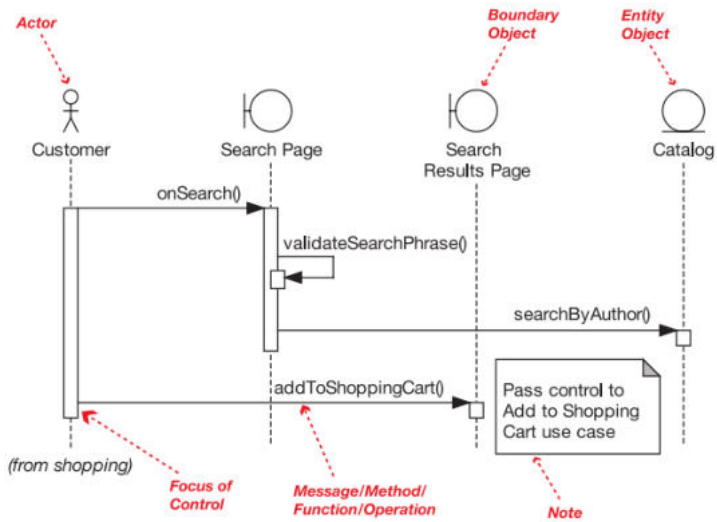
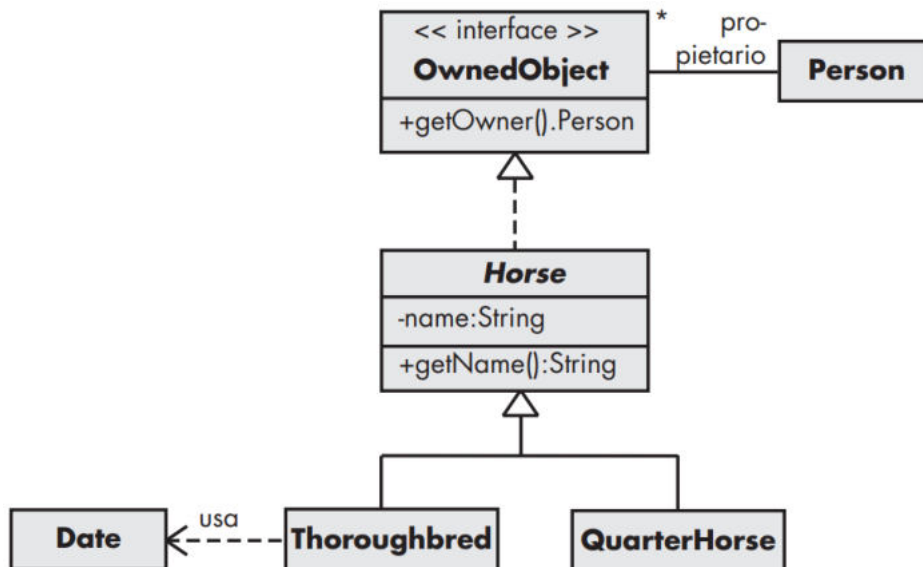


Figure 8-1. Sequence diagram notation

(Scott & Rosenberg, 1999)

## Anexo 7 Ejemplo de diagrama de clases relacionado con sub clases



(Roger S. Pressman)