

Presentación

Descripción del Proyecto

Para garantizar una educación de calidad, es fundamental llevar a cabo un seguimiento y evaluación del desempeño del docente universitario, comenzando con la planificación de cada semestre académico para cada asignatura y docente. El portafolio docente incluye elementos que permiten visualizar la organización y estructura de los cursos que imparte cada profesor. Todos estos elementos deben ser examinados y analizados, lo que requiere la participación de todo el personal docente de la facultad, donde los docentes revisan los portafolios de sus colegas. Además, es crucial contar con un medio que facilite este proceso y conecte el trabajo de docentes, revisores y responsables de asignar tareas. Ante esta necesidad en el ámbito universitario, se propone el desarrollo e implementación de un sistema web dirigido a docentes, que permitirá el registro y revisión de los portafolios correspondientes a cada semestre académico, ofreciendo las herramientas necesarias para la presentación de los portafolios, así como para la revisión y corrección de las observaciones.

Metodología Utilizada

Para el desarrollo del sistema, se ha utilizado la metodología Scrum, una metodología ágil que facilita la gestión eficiente del proyecto mediante ciclos iterativos llamados sprints. Cada integrante del equipo de trabajo desempeña un rol clave para asegurar que el proyecto avance de manera coordinada.

Los roles principales son:

- ❖ **Scrum Master y Desarrollador:** Lidera el equipo y gestiona las ceremonias de Scrum. (Daily Stand-ups, Sprint Planning, Sprint Review y Retrospective)
- ❖ **Desarrollador Full Stack:** Desarrolla tanto el frontend como el backend del sistema.
- ❖ **Desarrollador Frontend:** Implementa la interfaz de usuario del sistema.
- ❖ **Desarrollador Backend:** Desarrolla la lógica de negocio y la integración con la base de datos.
- ❖ **Diseñador UX/UI:** Responsable de la experiencia de usuario y el diseño visual del sistema.

Pasos Seguidos en el Proyecto

- **Análisis de Problemas y Necesidades:** Se detectaron múltiples inconvenientes en el proceso de revisión de portafolios, tales como la falta de un sistema, las demoras en la entrega de informes.
- **Definición de la Solución:** Se propuso el desarrollo de una plataforma web utilizando tecnologías como Laravel y MySQL.

- **Desarrollo del Sistema:** Durante varios sprints, el equipo se enfocó en implementar las funcionalidades principales, que incluyen el registro, la revisión y la generación automática de informes para los portafolios docentes.
- **Pruebas y Validación:** Se llevarán a cabo pruebas exhaustivas para verificar que el sistema satisfaga tanto los requisitos funcionales como los no funcionales, asegurando una interfaz intuitiva y un funcionamiento eficaz.
- **Generación de Reportes:** El sistema ofrece la opción de crear informes detallados sobre el estado de los portafolios, lo que facilita el seguimiento de los avances académicos.

Recomendaciones para la Implementación

- ☐ **Capacitaciones:** Se recomienda llevar a cabo sesiones de formación para los docentes y revisores, con el fin de que se familiaricen con el uso de la plataforma.
- ☐ **Mantenimiento:** Se propone crear un plan de mantenimiento periódico para asegurar la estabilidad del sistema y contemplar futuras actualizaciones que permitan la integración de nuevas funcionalidades.
- ☐ **Retroalimentación:** Es importante establecer un sistema de retroalimentación continua con los usuarios finales para detectar posibles áreas de mejora.

II. Organigrama del proyecto

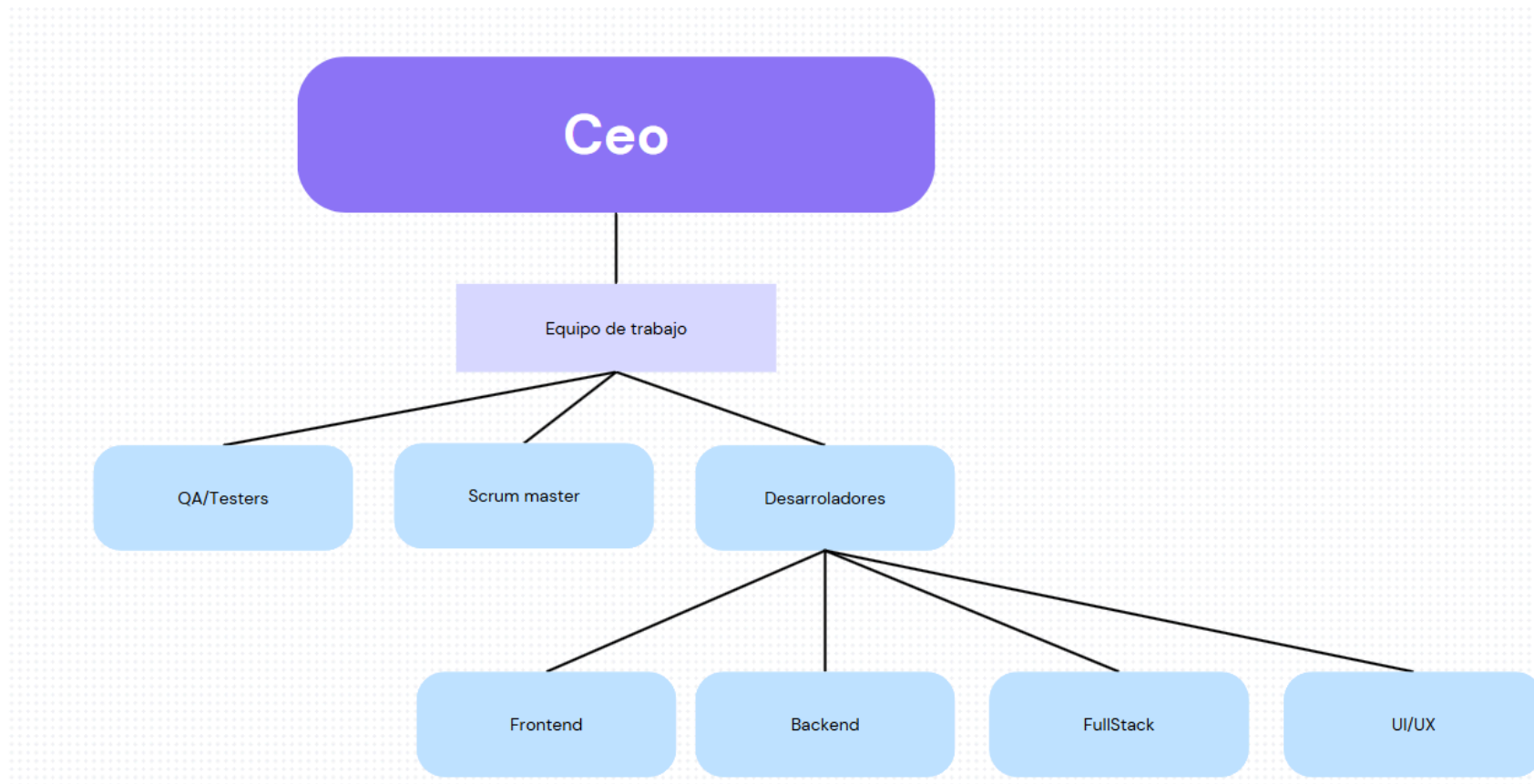


Tabla 1.2 Organigrama del proyecto

III. Plan del proyecto.

ACTIVIDADES		SETIEMBRE				OCTUBRE				NOVIEMBRE				DICIEMBRE			
		S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4
N°	INTRODUCCION Y ORGANIZACION																
1	Redacción de la Introducción																
2	Distribucion del Grupo de Trabajo																
3	Identificación y Documentación de Procesos																
	Identificación de Problemas																
	JUSTIFICACIÓN Y PLANIFICACIÓN																
4	Justificación del Proyecto																
5	Análisis de Factibilidad y Viabilidad																
6	Redacción y Firma del Contrato																
	PLANIFICACIÓN Y DOCUMENTOS DE REQUERIMIENTOS																
7	Requerimientos Funcionales																
8	Requerimientos No Funcionales																
9	Matriz de Rastreabilidad de Requerimientos																
	DISEÑO DEL SISTEMA																
10	Arquitectura Funcional del Sistema																
11	Arquitectura Técnica del Sistema																
12	Topología del Software																
13	Topología del Hardware																
14	Casos de Uso																
15	Diagramas Secuenciales																
	CONSTRUCCIÓN Y PRUEBAS																
16	Modelo Lógico de la Base de Datos																
17	Modelo Físico de la Base de Datos																
	ENTREGA Y DOCUMENTACIÓN																
18	Pruebas Finales del Sistema																
19	Documentación y Entrega Final																

Tabla 1.3 Plan del proyecto

IV. Glosario de términos propios de la implementación

1. Portafolio docente

Conjunto de documentos y recursos que muestran la planificación, organización y desempeño académico del docente en un semestre, incluyendo syllabus, evaluaciones, planes de clase y retroalimentación.

2. Sistema web

Aplicación accesible a través de un navegador que permite gestionar información de manera centralizada, facilitando la interacción entre usuarios en línea.

3. Revisión de portafolios

Proceso mediante el cual se evalúan y analizan los portafolios presentados por los docentes, verificando que cumplan con los criterios académicos y administrativos establecidos.

4. Laravel

Framework de desarrollo web basado en PHP que proporciona una arquitectura eficiente para construir aplicaciones robustas y escalables, utilizado en el backend del proyecto.

5. MySQL

Sistema de gestión de bases de datos relacional utilizado para almacenar y gestionar los datos del sistema web, como usuarios, portafolios y reportes.

6. Scrum

Metodología ágil de gestión de proyectos utilizada por el equipo para organizar el trabajo colaborativo mediante sprints, ceremonias diarias (Daily Scrum), revisiones y retrospectivas.

7. Scrum Master

Rol del equipo encargado de facilitar las ceremonias Scrum y asegurar el cumplimiento del marco metodológico, además de resolver impedimentos para el avance del proyecto.

8. Desarrollador Full Stack

Profesional capaz de trabajar tanto en el frontend (interfaz de usuario) como en el backend (lógica y procesamiento de datos) del sistema.

9. Frontend

Parte del desarrollo que se enfoca en la interfaz gráfica del usuario (UI), garantizando que las interacciones sean intuitivas y funcionales.

10. Backend

Sección del desarrollo enfocada en la lógica de negocio, procesamiento de datos y comunicación con la base de datos.

11. Diseño UX/UI

Diseño orientado a mejorar la Experiencia de Usuario (UX) y la Interfaz de Usuario (UI), asegurando que el sistema sea fácil de usar y visualmente atractivo.

12. Daily Scrum

Reunión breve que se realiza diariamente donde los miembros del equipo responden a tres preguntas:

¿Qué avancé ayer?

¿Qué planeo hacer hoy?

¿Hay impedimentos en mi trabajo?

13. Directiva N° 003-2018-VRAC-UNSAAC

Normativa interna de la Universidad Nacional de San Antonio Abad del Cusco que establece criterios y procesos para la gestión y revisión del portafolio docente.

14. ICACIT

Organismo acreditador de programas educativos en ingeniería, tecnología y ciencias, cuya evaluación se toma en cuenta al diseñar los procesos de calidad del sistema académico.

15. Retroalimentación

Proceso mediante el cual los revisores proporcionan comentarios y observaciones a los docentes sobre sus portafolios, permitiendo mejoras continuas.

16. Sprint

Intervalo de tiempo definido (generalmente de 1 a 4 semanas) durante el cual el equipo Scrum trabaja en la implementación de funcionalidades específicas del sistema.

17. Pruebas de integración

Proceso de verificación para asegurar que los diferentes módulos del sistema (frontend y backend) funcionan de manera conjunta sin errores.

18. Factibilidad económica

Análisis de la viabilidad del proyecto en función de los costos asociados al desarrollo, implementación y mantenimiento, considerando el uso de tecnologías open-source para reducir gastos.

19. Viabilidad técnica

Capacidad del equipo y del entorno tecnológico para implementar y mantener el sistema propuesto, asegurando la integración con otras plataformas universitarias.

V. Documento de Requerimientos

5.1. Requerimientos funcionales

Requerimientos Generales	
RF 1	El sistema deberá permitir el inicio de sesión para los usuario
RF 2	El sistema deberá permitir el cambio de contraseña de cada usuario
RF3	El sistema debe permitir ingresar y modificar los datos personales

Tabla 1.3 Requerimientos generales

Requerimientos del usuario Docente	
RF 4	El sistema debe permitir que se indique los elementos a enviar
RF 5	El sistema debe permitir la visualización de las observaciones realizadas por el Revisor
RF 6	El sistema debe permitir la subsanación de las observaciones obtenidas

Tabla 1.4 Requerimientos del usuario docente

Requerimientos del Revisor	
RF 7	El sistema debe permitir la visualización de los elementos remitidos por el docente
RF 8	El sistema debe permitir indicar y enviar las observaciones encontradas
RF 9	El sistema debe permitir la aprobación de los elementos remitidos por el docente

Tabla 1.5 Requerimientos del revisor

Requerimientos del Administrador	
RF 10	El sistema deberá permitir la creación y habilitación de usuarios Revisores
RF 11	El sistema deberá permitir la creación y habilitación de usuarios Docentes
RF 12	El sistema debe permitir la asignación de docentes para cada Revisor
RF 13	El sistema debe permitir la Apertura y Cierre del proceso de revisión
RF 14	El sistema deberá permitir la generación de reportes

Tabla 1.6 Requerimientos del administrador

5.2. Requerimientos no funcionales

Rendimiento	
RNF 1	El sistema debe mantener un tiempo de respuesta promedio inferior a 2 segundos para la mayoría de las operaciones, incluso bajo una carga moderada de hasta 100 usuarios concurrentes.
RNF 2	La carga de archivos de hasta 10 MB por los docentes debe completarse en un máximo de 5 segundos. (Alternativa) La lectura de archivos por docente debe completarse en un máximo de 5 segundos.
RNF 3	La revisión de documentos debe procesarse en menos de 1 segundo desde que el revisor da el check.

Tabla 1.6 Requerimientos no funcionales rendimiento

Escalabilidad	
RNF 4	El sistema puede crecer o manejar una mayor carga de trabajo cuando aumenta el número de usuarios o la cantidad de datos subidos al sistema.

Tabla 1.7 Requerimientos no funcionales escalabilidad

Seguridad	
RNF 5	Garantizar que datos de docentes, revisores y archivos sensibles estén seguros en todo momento.(corregir)
RNF 6	Garantizar el acceso basado en roles , asegurando que los docentes, revisores y administradores solo tengan acceso a las funcionalidades que correspondan a su rol.
RNF 7	Garantizar la restricción de acceso a datos sensibles, de forma que los usuarios solo puedan ver y manipular la información asociada a sus respectivos roles.

Tabla 1.8 Requerimientos no funcionales seguridad

Usabilidad	
RNF 8	Interfaz intuitiva tanto para docentes como para revisores.

Tabla 1.9 Requerimientos no funcionales usabilidad

Disponibilidad	
RNF 9	El sistema debe de estar disponible en todo momento, especialmente en periodos de alta actividad como lo es el inicio de semestre

Tabla 2.0 Requerimientos no funcionales disponibilidad

Confiabilidad	
RNF 10	El sistema tolera fallos mediante mecanismos de recuperación automática, garantizando que cualquier interrupción se resuelva en el menor tiempo posible sin pérdida de datos.
RNF 11	Los datos deben ser respaldados automáticamente en intervalos de 24 horas, con al menos dos copias almacenadas en ubicaciones seguras para evitar la pérdida de información.(corregir)

Tabla 2.1 Requerimientos no funcionales confiabilidad

Mantenimiento	
RNF 12	El sistema debe ser fácilmente actualizable, así permitiendo la corrección de errores y adición de funcionalidades sin afectar a los usuarios.
RNF 13	El sistema debe tener una documentación actualizada para facilitar el mantenimiento y la incorporación de nuevas funciones.

Tabla 2.2 Requerimientos no funcionales mantenimiento

5.3. Matriz de rastreo de los requerimientos (Rastreabilidad)

		Lista de Casos de Uso													
		CU1	CU2	CU3	CU4	CU5	CU6	CU7	CU8	CU9	CU10	CU11	CU12	CU13	CU14
Requerimientos	R1	X													
	R2		X												
	R3			X											
	R4				X										
	R5					X									
	R6						X								
	R7							X							
	R8								X						
	R9									X					
	R10										X				
	R11											X			
	R12												X		
	R13													X	
	R14														X

Tabla 2.3 Matriz de rastreo de los requerimiento

VI. Documentos de Diseño

6.1. Arquitectura funcional del sistema

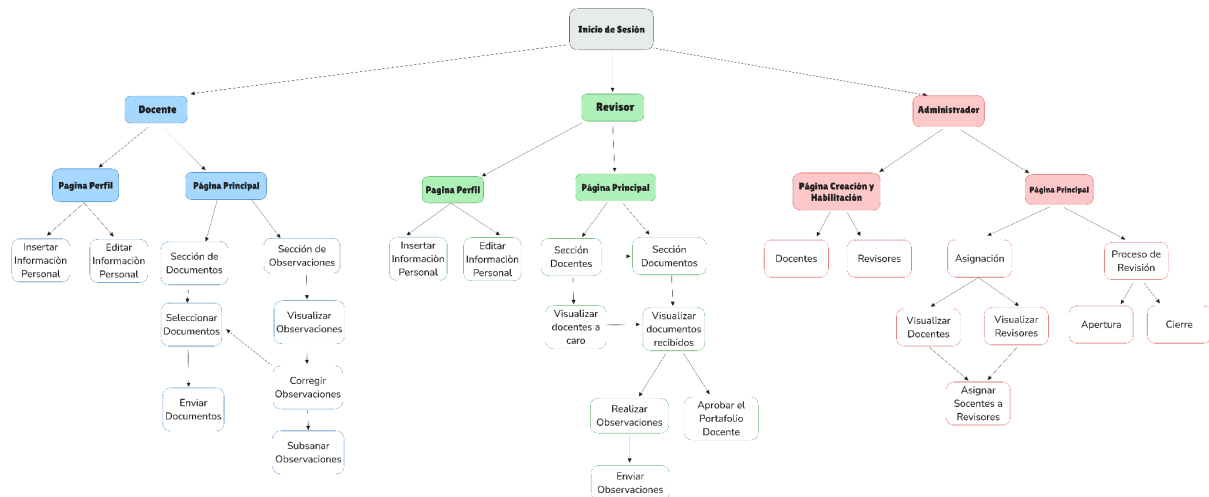


Tabla 2.4 Arquitectura funcional del sistema

6.2. Arquitectura técnica del sistema

6.2.1. Topología de software

Arquitectura multicapas

Se ha adoptado una arquitectura de tres capas para asegurar modularidad, escalabilidad y flexibilidad en el despliegue, tanto local como en la web. Esta arquitectura, que separa la interfaz de usuario, la lógica del negocio y el acceso a datos, facilita el mantenimiento y la futura expansión del sistema.

- 1) **Capa de Presentación (Frontend):** Esta sería la interfaz con la que interactúan los docentes y revisores. Aquí se usaría Laravel Blade para diseñar las pantallas donde los usuarios podrán ver las revisiones y recibir notificaciones sobre observaciones. Esta capa permite que la interacción sea intuitiva y agradable para los usuarios.
- 2) **Capa de Lógica de Negocio (Backend):** En esta capa se maneja toda la lógica que hace funcionar el sistema. Laravel se encargará de gestionar todo lo relacionado con la asignación de revisores, la generación de informes y la administración de las observaciones de los portafolios.

- 3) **Capa de Acceso a Datos (Base de Datos):** La base de datos en MySQL, almacenará toda la información del sistema, como los usuarios, las revisiones y las observaciones. Esta capa está separada para asegurar que los datos sensibles sean accesibles de manera controlada y segura, evitando que cualquier usuario o parte no autorizada acceda directamente.

6.2.2. Topología del hardware

La topología del hardware del sistema está optimizada para una alta disponibilidad y un rendimiento óptimo:

1. Servidor Web:

Funciones: Alojamiento del servidor web (Laravel)

Especificaciones Recomendadas:

- CPU: Mínimo 4 núcleos.
- RAM: Mínimo 8 GB (se recomienda más si se espera una alta carga de usuarios).(corregir)

2. Servidor de Base de Datos:

Funciones: Almacenamiento y gestión de datos del sistema (MySQL).

Especificaciones Recomendadas:

- CPU: Mínimo 4 núcleos.
- RAM: Mínimo 4 GB.

3. Maquina Funcional en Desarrollo (corregir, donde se despliega la informacion):

CPU: Intel Core i5-10400F (6 núcleos, 12 hilos, 2.9 GHz base, hasta 4.3 GHz en turbo boost)

Memoria RAM: 16GB DDR4 a 2666 MHz

Disco Duro/SSD: 512GB SSD NVMe

Gráficos: Integrados (Intel UHD Graphics 630)

Sistema Operativo: Windows 10

Otros periféricos: Monitor Full HD de 24 pulgadas, teclado y ratón estándar

6.3. Documentos basados en UML

6.3.1. Casos de uso

Casos de uso de alto nivel

Caso de uso 1: Permitir el inicio de sesión para los usuarios

CU01	Permitir el inicio de sesión para los usuarios
Actor	Usuario
Tipo	Primario
Descripción	El sistema deberá permitir el inicio de sesión de los usuarios mediante un proceso seguro que valide sus credenciales de acceso. Esto garantizará que solo los usuarios autorizados puedan acceder a sus datos

Tabla 2.5 Caso de uso permitir el inicio de sesión para los usuarios

Caso de uso 2: Permitir el cambio de contraseña de cada usuario

CU02	Permitir el cambio de contraseña de cada usuario
Actor	Usuario
Tipo	Primario
Descripción	El sistema deberá permitir a cada usuario cambiar su contraseña de manera segura, facilitando el acceso y la gestión de su cuenta. Este proceso incluirá medidas de autenticación para garantizar que solo el usuario autorizado pueda realizar el cambio.

Tabla 2.6 Caso de uso permitir cambio de contraseña de cada usuario

Caso de uso 3: Permitir ingresar y modificar los datos personales

CU03	Permitir ingresar y modificar los datos personales
Actor	Docente
Tipo	Primario

Descripción	El sistema deberá permitir a los usuarios ingresar y modificar sus datos personales de forma sencilla y segura. Esto garantizará que la información esté siempre actualizada y refleje con precisión las necesidades de cada usuario.
--------------------	---

Tabla 2.7 Caso de uso permitir ingresar y modificar los datos personales

Caso de uso 4: Permitir que se indique los elementos a enviar

CU04	Permitir que se indique los elementos a enviar
Actor	Docente
Tipo	Primario
Descripción	El sistema deberá permitir a los usuarios indicar los elementos que desean enviar, facilitando así la gestión y seguimiento de sus solicitudes.

Tabla 2.8 Caso de uso permitir que se indique los elementos a enviar

Caso de uso 5: Permitir la visualización de las observaciones realizadas por el Revisor

CU05	Permitir la visualización de las observaciones realizadas por el Revisor
Actor	Docente
Tipo	Primario
Descripción	El sistema deberá permitir a los docentes visualizar las observaciones realizadas por el revisor, proporcionando transparencia en el proceso de revisión. Esta función facilitará la comprensión de los comentarios y sugerencias (correo electrónico), mejorando la comunicación entre las partes involucradas.

Tabla 2.9 Caso de uso permitir la visualización de las observaciones realizadas por el revisor

Caso de uso 6: Permitir la subsanación de las observaciones obtenidas

CU06	Permitir la subsanación de las observaciones obtenidas
Actor	Docente
Tipo	Primario
Descripción	El sistema deberá permitir a los docentes subsanar las observaciones obtenidas, brindando la oportunidad de corregir cualquier error o deficiencia señalada. Esta funcionalidad fomentará la mejora continua y asegurará que los usuarios puedan cumplir con los requisitos establecidos.

*Tabla 3.0 Caso de uso permitir la subsanación de las observaciones obtenidas***Caso de uso 7: Permitir la visualización de los elementos remitidos por el docente**

CU07	Permitir la visualización de los elementos remitidos por el docente
Actor	Revisor
Tipo	Primario
Descripción	El sistema deberá permitir al revisor visualizar los documentos enviados por el docente, facilitando así la evaluación.

*Tabla 3.1 Caso de uso permitir la visualización de los elementos remitidos por el docente***Caso de uso 8: Permitir indicar y enviar las observaciones encontradas**

CU08	Permitir indicar y enviar las observaciones encontradas
Actor	Revisor
Tipo	Primario
Descripción	El sistema deberá permitir a los revisores indicar y enviar las observaciones encontradas durante el proceso de revisión. Esta funcionalidad garantizará una evaluación más clara.

Tabla 3.2 Caso de uso Permitir indicar y enviar las observaciones encontradas

Caso de uso 9: Permitir la aprobación de los elementos remitidos por el docente

CU09	Permitir la aprobación de los elementos remitidos por el docente
Actor	Revisor
Tipo	Primario
Descripción	El sistema deberá permitir la aprobación de los elementos remitidos por el docente (aprobado, en espera, correcciones, incompleto), asegurando un proceso de evaluación eficiente y transparente. Esta función facilitará la validación de los documentos enviados.

*Tabla 3.3 Caso de uso Permitir la aprobación de los elementos remitidos por el docente***Caso de uso 10: Permitir la creación y modificación de usuarios revisores**

CU10	Permitir la creación y modificación de usuarios revisores
Actor	Administrador
Tipo	Primario
Descripción	El sistema deberá permitir la creación y modificación de usuarios revisores, garantizando la gestión adecuada de sus roles y permisos. Esta funcionalidad asegurará que sólo personas autorizadas puedan acceder y evaluar la documentación.

*Tabla 3.4 Caso de uso Permitir la creación y modificación de usuarios revisores***Caso de uso 11: Permitir la creación y modificación de usuarios docentes**

CU11	Permitir la creación y modificación de usuarios docentes
Actor	Administrador
Tipo	Primario
Descripción	El sistema deberá permitir la creación y modificación de docentes, garantizando la gestión adecuada de sus roles y permisos. Esta funcionalidad

	aseguraré que sólo docentes autorizados puedan acceder y evaluar la documentación.
--	--

Tabla 3.5 Caso de uso Permitir la creación y modificación de usuarios docentes

Caso de uso 12: Permitir la asignación de docentes para cada Revisor

CU12	Permitir la asignación de docentes para cada Revisor
Actor	Administrador
Tipo	Primario
Descripción	El sistema deberá permitir la asignación de docentes a cada Revisor, facilitando la gestión y supervisión de las evaluaciones. Esta función asegurará una distribución equitativa de responsabilidades

Tabla 3.6 Caso de uso Permitir la asignación de docentes para cada Revisor

Caso de uso 13: Permitir la Apertura y Cierre del proceso de revisión

CU13	Permitir la Apertura y Cierre del proceso de revisión
Actor	Administrador
Tipo	Primario
Descripción	El sistema deberá permitir la apertura y cierre del proceso de revisión, proporcionando un control efectivo sobre las etapas de evaluación. Esta funcionalidad asegurará que los usuarios puedan gestionar los plazos y el flujo del proceso de revisión de manera organizada y eficiente

Tabla 3.7 Caso de uso Permitir la Apertura y Cierre del proceso de revisión

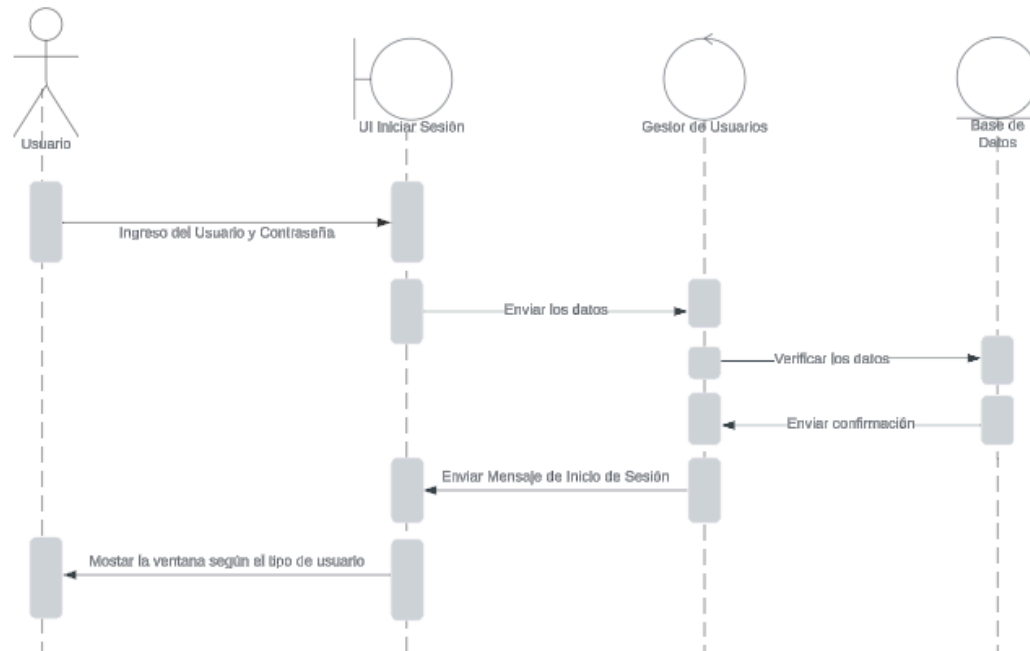
Caso de uso 14: Permitir la generación de reportes

CU14	Permitir la generación de reportes
Actor	Administrador
Tipo	Primario
Descripción	El sistema deberá permitir la generación de reportes, brindando a los usuarios información detallada sobre el proceso de revisión.

Tabla 3.8 Caso de uso Permitir la generación de reportes

6.3.3. Diagramas secuenciales

a. Iniciar sesión



3.9 Diagrama de secuencia iniciar sesión

b. Cambiar contraseña

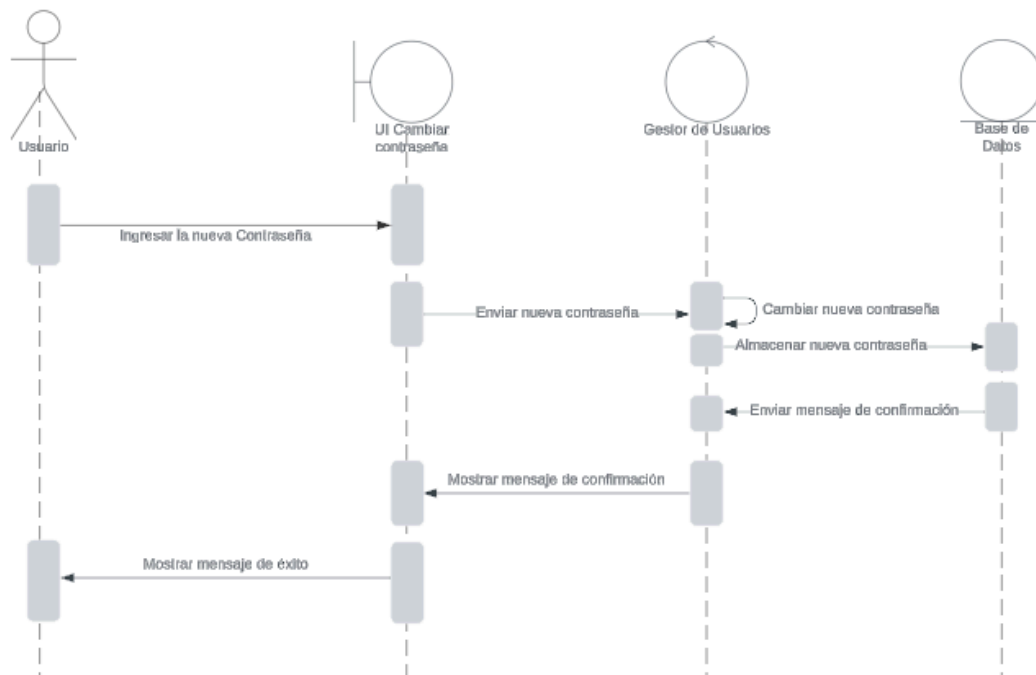


Tabla 4.0 Diagrama de secuencia iniciar sesión

c. Modificar datos personales

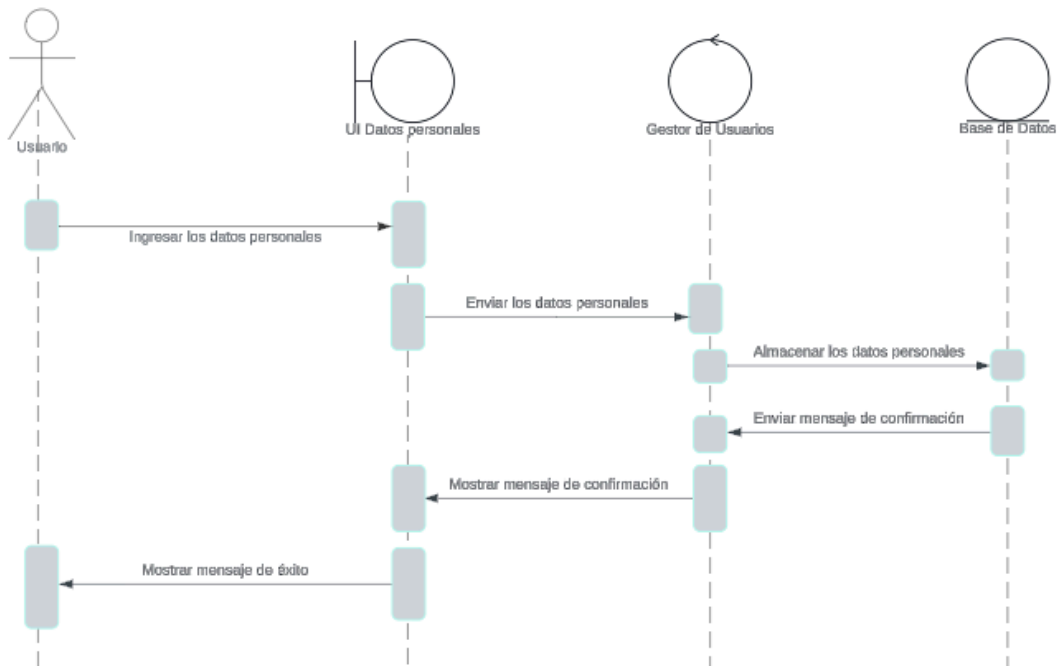


Tabla 4.1 Diagrama de secuencia modificar datos personales

d. Indicar los elementos a enviar

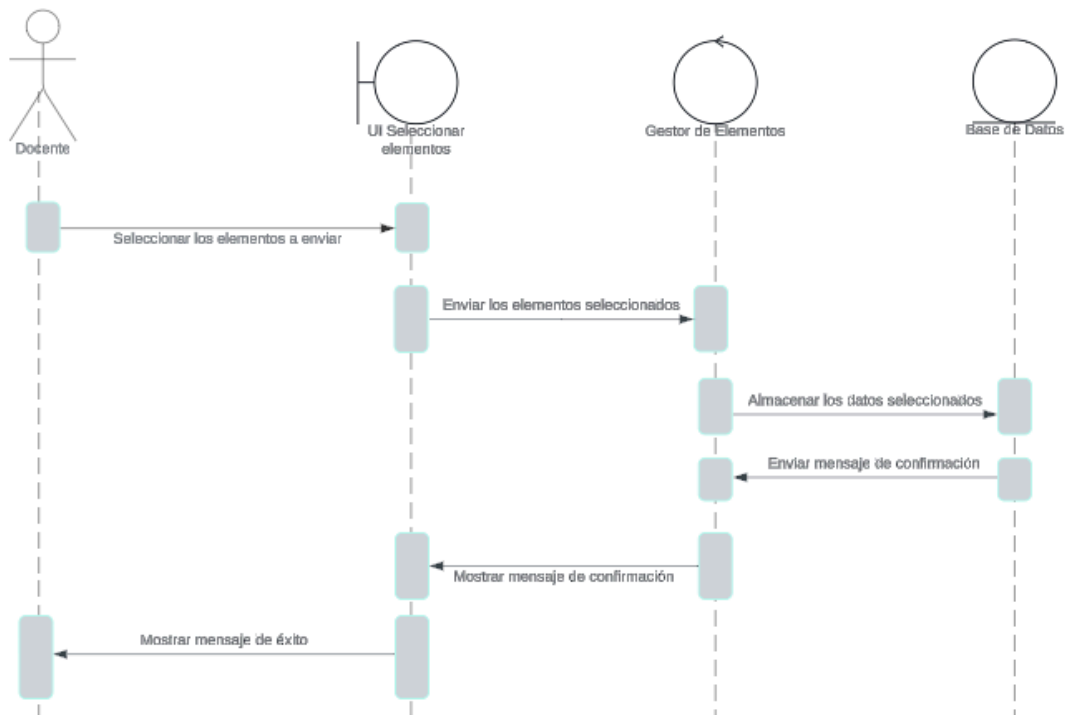


Tabla 4.2 Diagrama de secuencia indicar los elementos a enviar

e. Visualizar las observaciones realizadas por el revisor

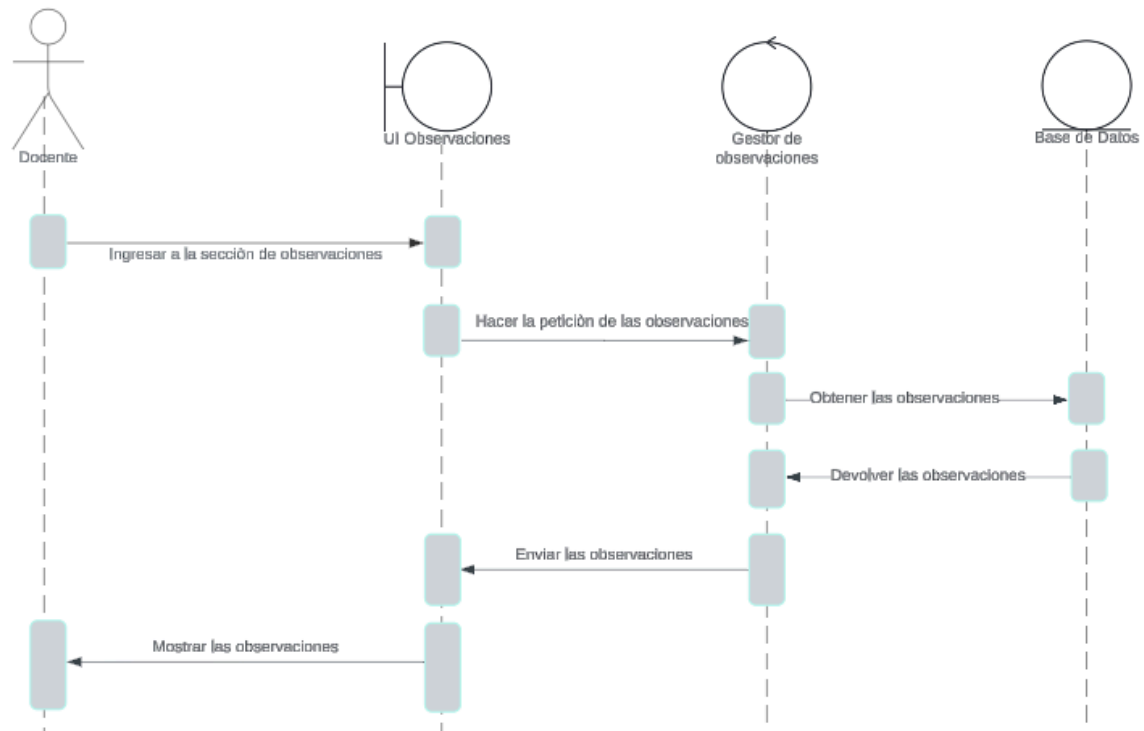


Tabla 4.3 Diagrama de secuencia visualizar las observaciones

f. Subsanan las observaciones (Docente)

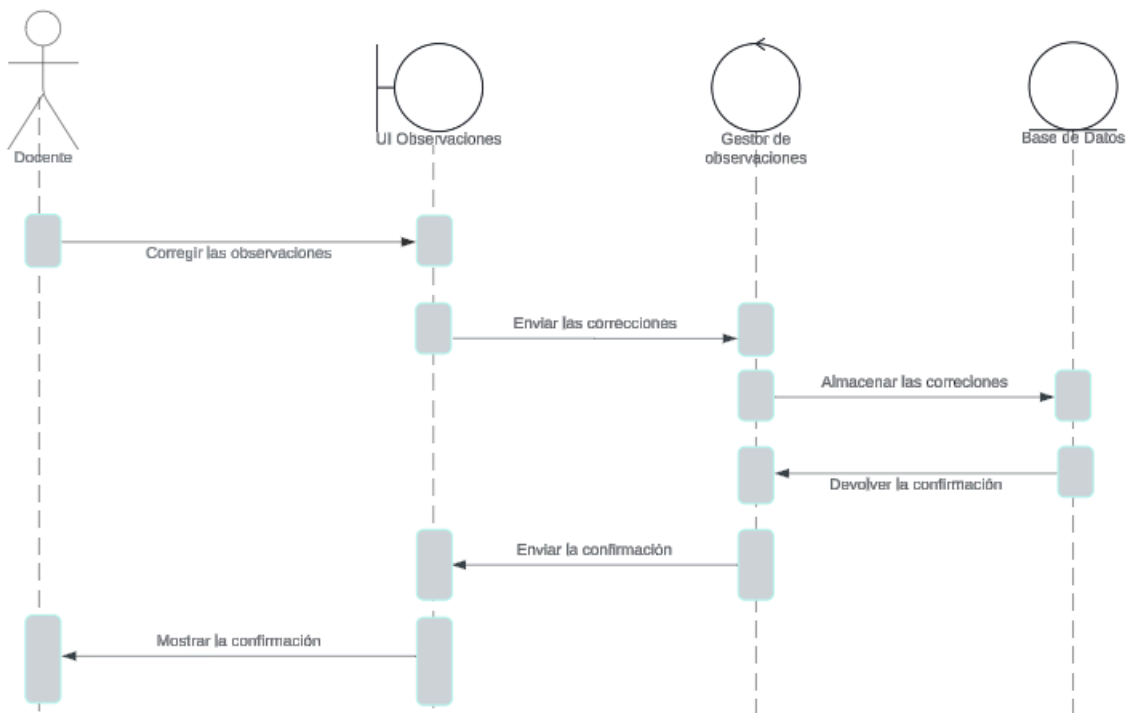


Tabla 4.4 Diagrama de secuencia subsanar las observaciones

g. Visualizar documentos

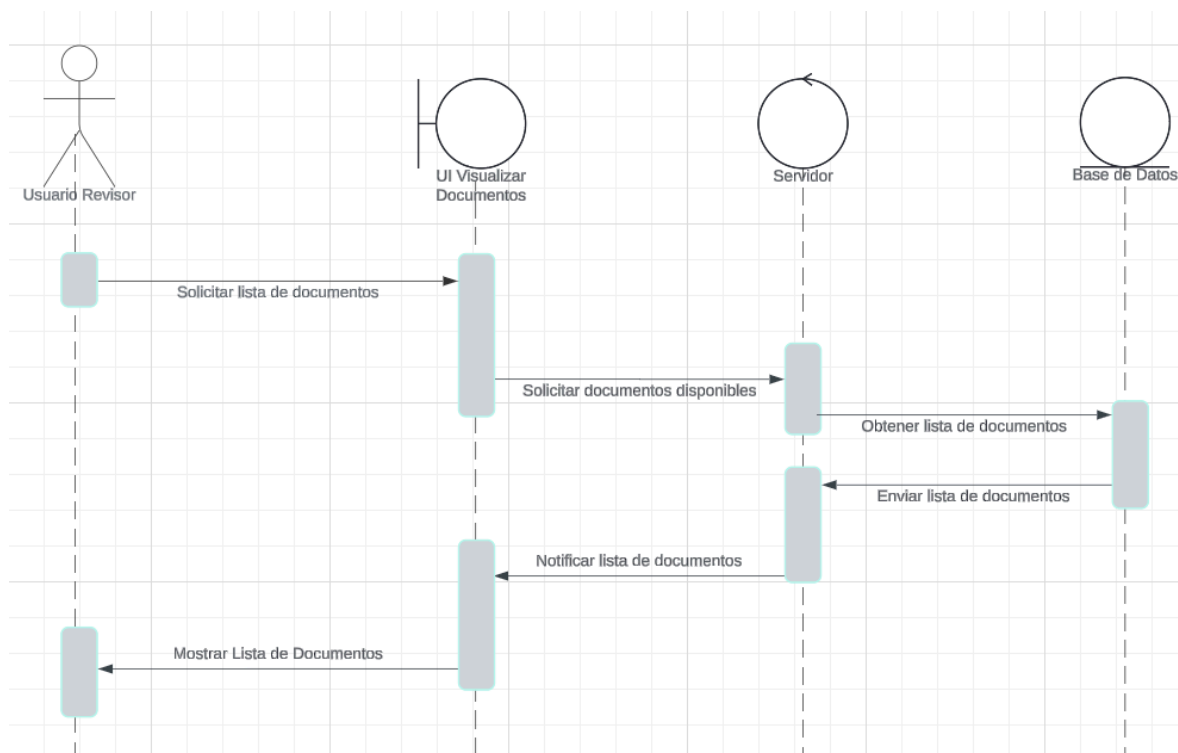


Tabla 4.5 Diagrama de secuencia visualizar documentos

h. Indicar y enviar observaciones

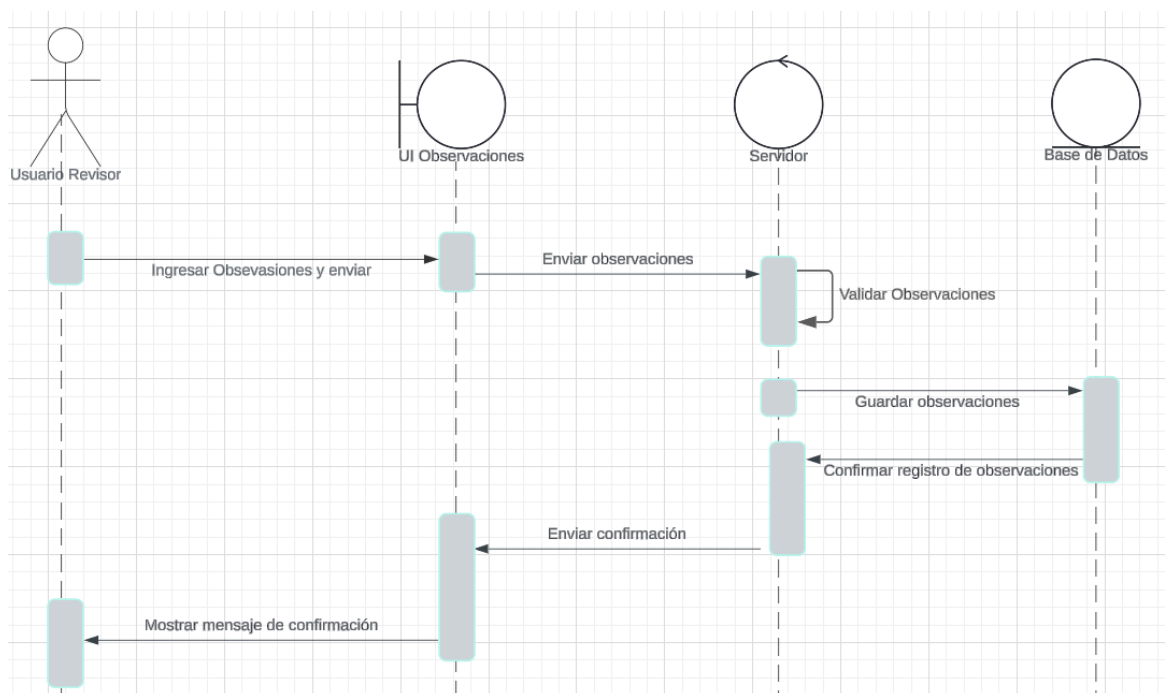


Tabla 4.6 Diagrama de secuencia indicar y enviar observaciones

i. Aprobar los elementos del portafolio docente

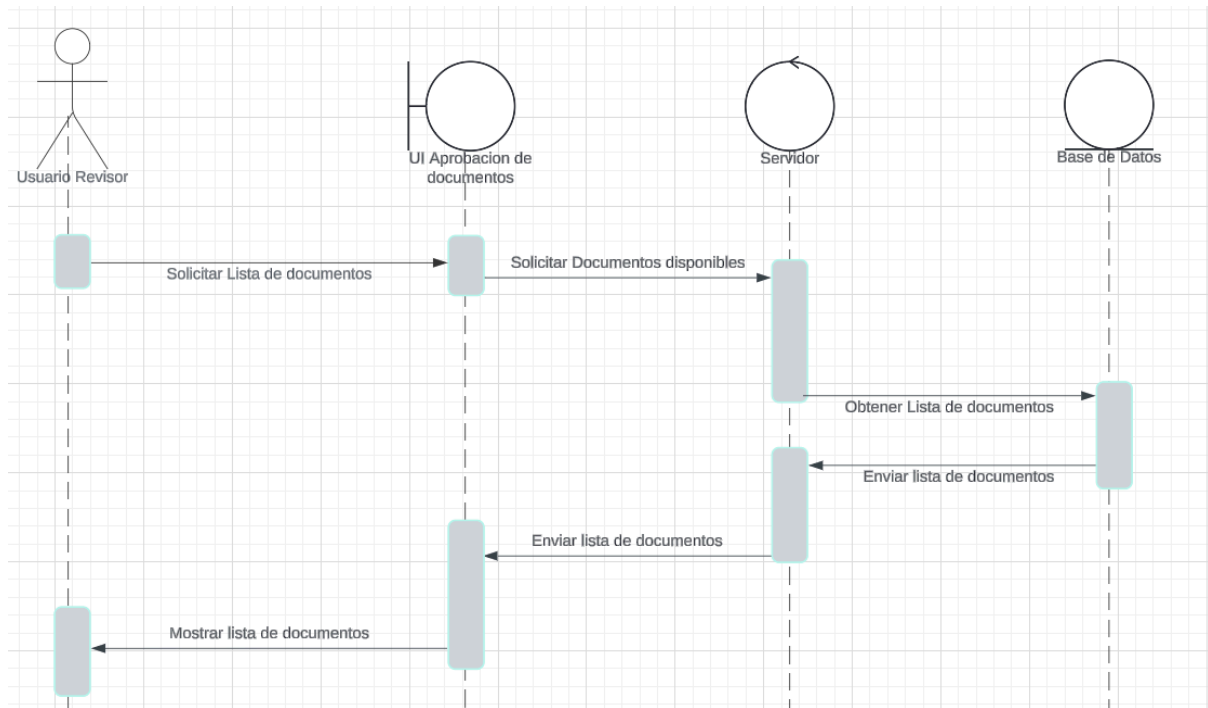


Tabla 4.7 Diagrama de secuencia aprobar los elementos del portafolio docente

j. Crear y habilitar a los Revisores

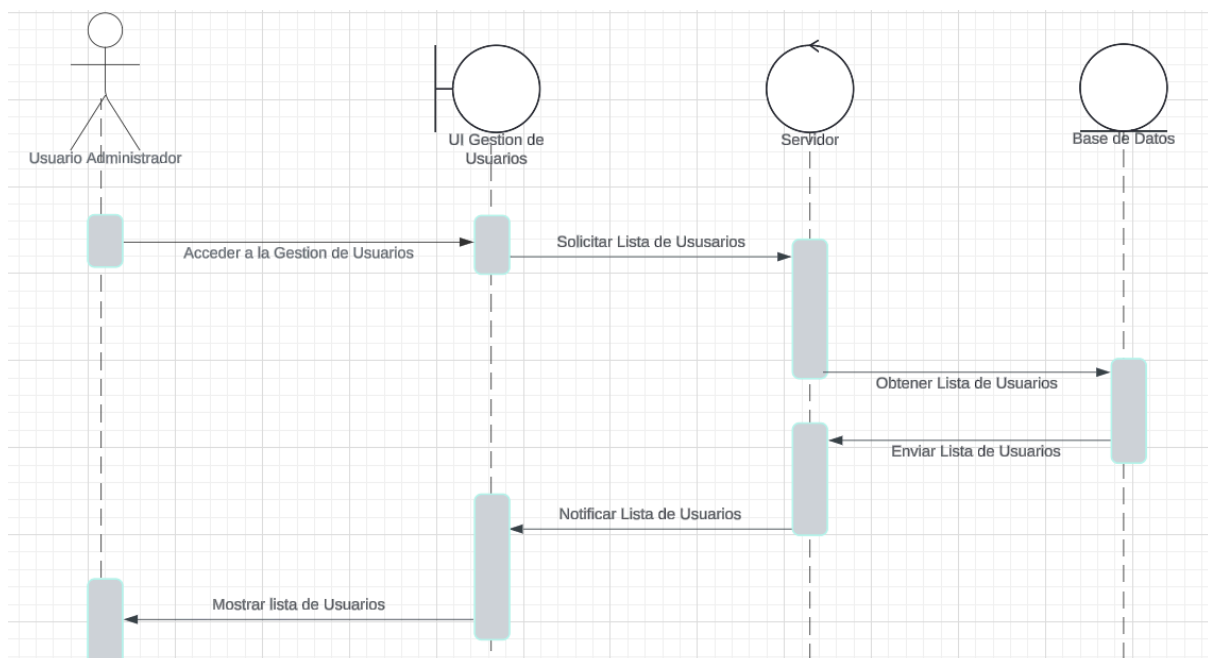


Tabla 4.8 Diagrama de secuencia crear y habilitar a los revisores

k. Crear y habilitar a los Docentes

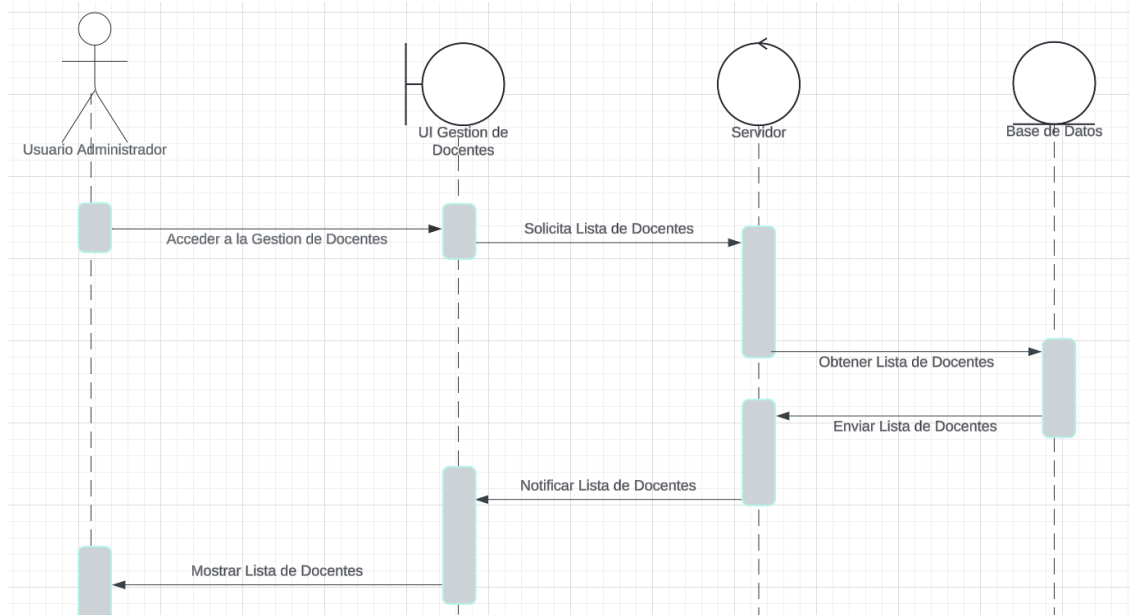


Tabla 4.9 Diagrama de secuencia crear y habilitar a los docentes

l. Creación Asignación de Docentes a cada Revisor

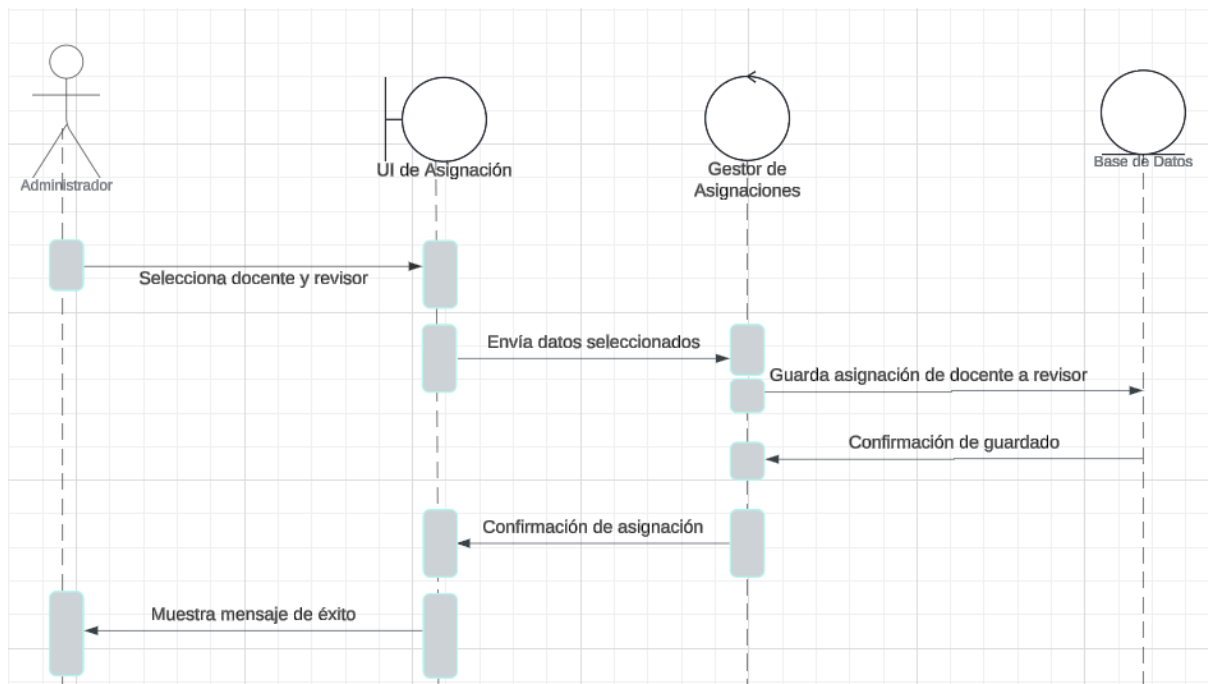


Tabla 5.0 Diagrama de secuencia creación asignación de docentes a cada revisor

m. Aperturar y Cerrar el proceso de revisión

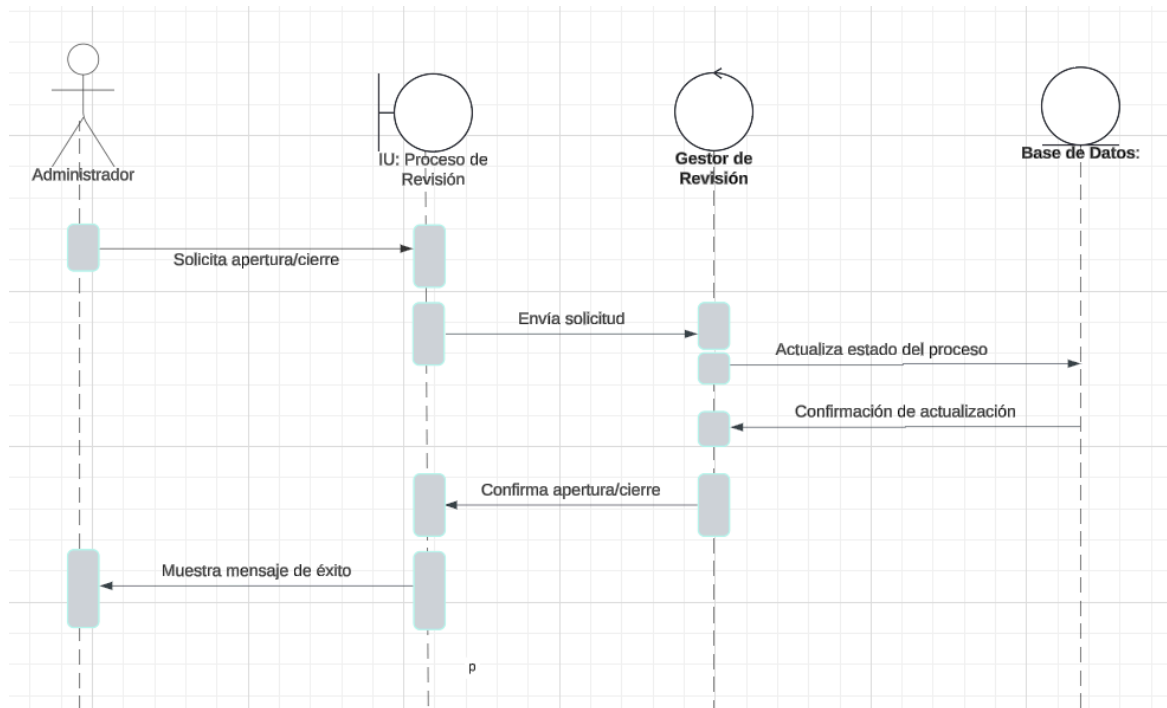


Tabla 5.1 Diagrama de secuencia aperturar y cerrar el proceso de revisión

n. Generar reportes

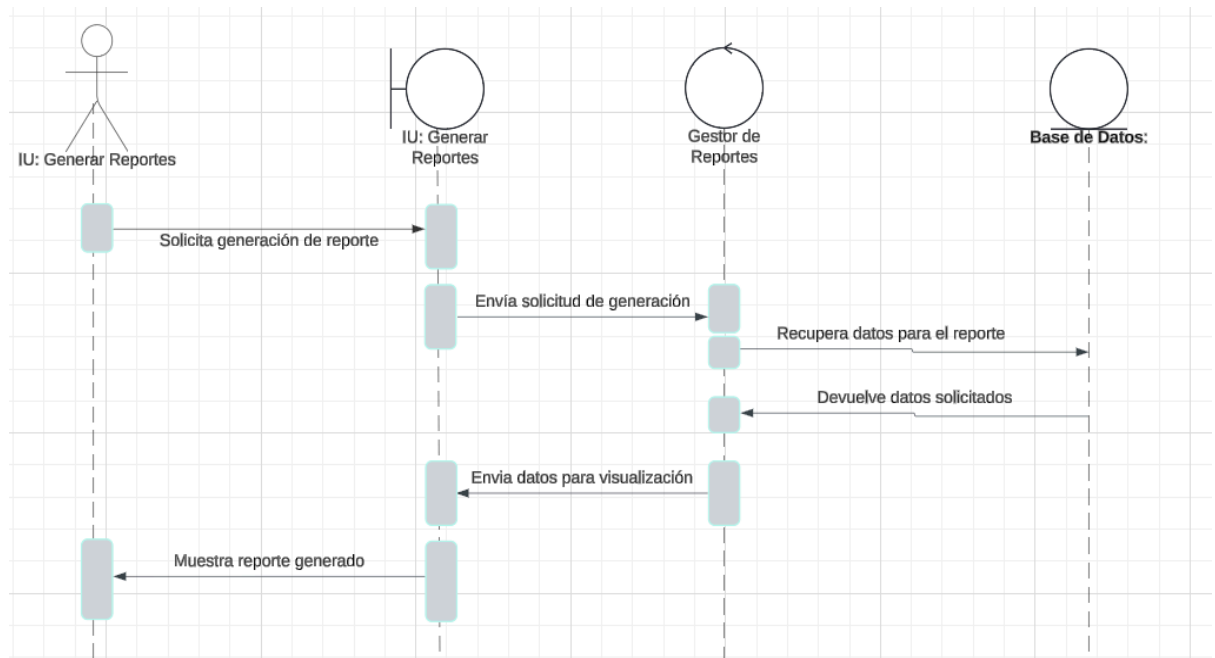


Tabla 5.2 Diagrama de secuencia generar reportes

6.4. Descripción de la metodología ágil utilizada de manera específica

Sprint 0: Planificación del proyecto

Inicio: 01/09/2024

Fin: 15/09/2024

Historia de usuario H01: Organización del Proyecto en Azure DevOps	
Como Scrum Master, quiero tener todo el proyecto y los integrantes del equipo, estén organizados en azure DevOps para tener una mayor coordinación en el trabajo para ayudar a mis compañeros, así como una mejor documentación.	
estimación (horas):	1
Prioridad:	8
Dependencia	ninguna

Tabla 5.3 Cuadro historia de usuario h01

Historia de usuario H02 : Monitoreo del Avance del Proyecto	
Como Product Owner, quiero estar actualizado en todo lo que se avanza en el proyecto, para poder tener la certeza de que el proyecto se está avanzando en los tiempos previstos.	
estimación (horas):	1
Prioridad:	8
Dependencia	ninguna

Tabla 5.4 Cuadro historia de usuario h02

Historia de usuario H03:Uso de Laragon para la Adaptación del Equipo	
Como Desarrollador full stack, quiero poder usar Laragon, para que nadie en el equipo tenga problemas a la hora de adaptarse al trabajo.	
estimación (horas):	<i>1</i>
Prioridad:	<i>8</i>
Dependencia	<i>ninguna</i>

Tabla 5.5 Cuadro historia de usuario h03

Historia de usuario H04: Creación de la Interfaz Visual del Proyecto	
Como desarrollador Front end, quiero usar html, css, javascript y una plantilla, para poder crear la parte visual del trabajo, debido a que son las herramientas que más conozco y con que tengo más comodidad.	
estimación (horas):	<i>1</i>
Prioridad:	<i>8</i>
Dependencia	<i>ninguna</i>

Tabla 5.6 Cuadro historia de usuario h04

Historia de usuario H05: Desarrollo de la Lógica y Conexiones de Base de Datos	
Como desarrollador Back End, quiero usar las herramientas MySQL y php, para poder desarrollar la parte de las conexiones con la base de datos, así como la lógica que tendrá el programa.	
estimación (horas):	<i>1</i>
Prioridad:	<i>8</i>
Dependencia	<i>ninguna</i>

Tabla 5.7 Cuadro historia de usuario h05

Historia de usuario H06 : Uso de Figma para Diseño UX/UI	
Como diseñador UX/UI quiero usar el software de figma, debido a que es un software que te da un mayor control a la hora de diseñar proyectos, además de ser una herramienta con la cual ya estoy familiarizado.	
<i>estimación (horas):</i>	<i>1</i>
<i>Prioridad:</i>	<i>8</i>
<i>Dependencia</i>	<i>ninguna</i>

Tabla 5.8 Cuadro historia de usuario h06

Sprint 1: Presentación 1ra del proyecto

Inicio: 15/09/2024**Fin:** 29/09/2024

Historia de usuario H07: DOCUMENTACIÓN	
Como Product owner quiero tener completo la primera parte de entrega del proyecto, para su debida exposición	
estimación (horas):	8
Prioridad:	13
Dependencia	ninguna

Tabla 5.9 Cuadro historia de usuario h07

Tarea T01: Introduction	
Realizar la introducción de la documentación de manera escrita, justo después del índice. La introducción no deberá ser más extensa de media hoja. Aquí se hará la descripción de cómo es el proceso planteado.	
estimación (horas)	1
Prioridad	13
Dependencia	ninguna
Historia de usuario	H07

Tabla 6.0 Cuadro tarea T01: Introducción

Tarea T02: Descripción del grupo de trabajo	
Realizar la descripción del grupo de trabajo. Para poder tener de manera más clara los roles que tendrá cada persona, dado que usamos una metodología ágil, estos no son rígidos.	
estimación (horas)	1

Prioridad	<i>13</i>
Dependencia	<i>T01</i>
Historia de usuario	<i>H07</i>

Tabla 6.1 Cuadro tarea T02: Descripción del grupo de trabajo

Tarea T03: descripción del proceso	
Realizar la descripción del proceso, de como funciona actualmente la revisión de portafolios, usando BPMN. Esto va después de la descripción del grupo de trabajo. Se aconseja el uso del software bizagi	
estimación (horas)	<i>1</i>
Prioridad	<i>13</i>
Dependencia	<i>T02</i>
Historia de usuario	<i>H07</i>

Tabla 6.2 Cuadro tarea T03: Descripción del grupo de trabajo

Tarea T03: descripción del proceso	
Realizar la descripción del proceso, de como funciona actualmente la revisión de portafolios, usando BPMN. Esto va después de la descripción del grupo de trabajo. Se aconseja el uso del software bizagi	
estimación (horas)	<i>1</i>
Prioridad	<i>13</i>
Dependencia	<i>T02</i>
Historia de usuario	<i>H07</i>

Tabla 6.3 Cuadro tarea T03: Descripción del proceso

Tarea T04: problemas del proceso actual	
Realizar la descripción escrita de los problemas, que tiene el proceso actual de la revisión de portafolios. Esto va después de la descripción del proceso.	
<i>estimación (horas)</i>	<i>1</i>
<i>Prioridad</i>	<i>13</i>
<i>Dependencia</i>	<i>T03</i>
<i>Historia de usuario</i>	<i>H07</i>

Tabla 6.4 Cuadro tarea T04: Problemas del proceso actual

Tarea T05: justificación	
Realizar de manera escrita la justificación del proyecto actual, del porqué es importante desarrollar el proyecto. Esto va después de la descripción del problema.	
<i>estimación (horas)</i>	<i>1</i>
<i>Prioridad</i>	<i>13</i>
<i>Dependencia</i>	<i>T04</i>
<i>Historia de usuario</i>	<i>H07</i>

Tabla 6.5 Cuadro tarea T05: Justificación

Tarea T06: Factibilidad y viabilidad	
Realizar de manera escrita la sección de factibilidad y viabilidad, después de la justificación.	
<i>estimación (horas)</i>	<i>1</i>
<i>Prioridad</i>	<i>13</i>
<i>Dependencia</i>	<i>T05</i>
<i>Historia de usuario</i>	<i>H07</i>

Tabla 6.6 Cuadro tarea T06: Factibilidad y viabilidad

Tarea T07: Contrato	
Realizar el contrato entre la directora de escuela profesional y un representante de la compañía encargada de realizar el desarrollo del software. Esto va después de la factibilidad y viabilidad.	
estimación (horas)	<i>1</i>
Prioridad	<i>13</i>
Dependencia	<i>T06</i>
Historia de usuario	<i>H07</i>

Tabla 6.7 Cuadro tarea T07: Contrato

Tarea T08: Reportes	
Realizar la descripción escrita de los reportes de podrá generar el software. Esto va después del contrato.	
estimación (horas)	<i>1</i>
Prioridad	<i>13</i>
Dependencia	<i>T06</i>
Historia de usuario	<i>H07</i>

Tabla 6.8 Cuadro tarea T08: Reportes

Historia de usuario H08: CORRECCIONES	
Como Product Owner quiero que se realicen las correcciones explicadas en la exposición, para su debida calificación del avance del proyecto.	
estimación (horas):	<i>2</i>
Prioridad:	<i>13</i>
Dependencia	<i>H06</i>

Tabla 6.9 Cuadro historia de usuario H08

Sprint 2: Presentación 2da del proyecto

Inicio: 8/10/2024

Fin: 22/10/2024

Historia de usuario H09:	
Como Product Owner quiero que se realice todo lo concerniente a la 2da parte de la entrega del proyecto, para su respectiva exposición	
estimación (horas):	9
Prioridad:	13
Dependencia	H06

Tabla 7.0 Cuadro historia de usuario H09

Tarea T09: Documento de Presentación	
Realizar el documento de presentación donde se describe el proyecto y se resumen las metodologías, los pasos seguidos y las recomendaciones hechas para implementar el proyecto.	
estimación (horas)	2
Prioridad	13
Dependencia	T08
Historia de usuario	H09

Tabla 7.1 Cuadro tarea T08: Documento de presentación

Tarea T10: Documento de Presentación	
Realizar el organigrama del proyecto, describiendo sus partes esenciales para incluirlo en la documentación final.	
estimación (horas)	1

<i>Prioridad</i>	8
<i>Dependencia</i>	T09
<i>Historia de usuario</i>	H09

Tabla 7.2 Cuadro tarea T10: Documento de presentación

Tarea T11: Plan del proyecto	
Elaborar el plan del proyecto, creando un cronograma detallado que incluya las tareas a realizar en cada sprint, reflejando el avance del proyecto en cada fase.	
<i>estimación (horas)</i>	2
<i>Prioridad</i>	8
<i>Dependencia</i>	T10
<i>Historia de usuario</i>	H09

Tabla 7.3 Cuadro tarea T1: Plan del proyecto

Tarea T12: Glosario de términos	
Elaborar un glosario de términos con definiciones de palabras comúnmente utilizadas en el proyecto para facilitar la comprensión de los términos técnicos.	
<i>estimación (horas)</i>	1
<i>Prioridad</i>	8
<i>Dependencia</i>	T11
<i>Historia de usuario</i>	H09

Tabla 7.4 Cuadro tarea T12: Glosario de términos

Tarea T13: Requerimientos	
Documentar los requerimientos funcionales y no funcionales del sistema, además de desarrollar la matriz de rastreo de requerimientos para asegurar que se cumplan los objetivos del proyecto.	
<i>estimación (horas)</i>	3
<i>Prioridad</i>	8
<i>Dependencia</i>	T12
<i>Historia de usuario</i>	H09

Tabla 7.5 Cuadro tarea T13: Requerimientos

Tarea T14: Documentos de diseño	
<p>Completar la documentación del diseño del sistema, que incluirá:</p> <ul style="list-style-type: none"> • Arquitectura del sistema • Diagramas UML • Descripción de la metodología ágil • Modelo lógico y físico de la base de datos 	
<i>estimación (horas)</i>	4
<i>Prioridad</i>	13
<i>Dependencia</i>	T13
<i>Historia de usuario</i>	H09

Tabla 7.6 Cuadro tarea T14: Documentos de diseño

Sprint 3: Codificación y Entrega del Proyecto

Inicio: 23/10/2024

Fin: 06/11/2024

Historia de usuario H08: CORRECCIONES	
Como Product Owner quiero que se realicen las correcciones explicadas en la exposición, para su debida calificación del avance del proyecto.	
estimación (horas):	2
Prioridad:	13
Dependencia	H06

Tabla 7.7 Cuadro historia de usuario H09

Historia de usuario H10: CODIFICACIÓN Y ENTREGA FINAL	
Como Product Owner , quiero que se realice toda la codificación final del sistema y se haga la entrega del proyecto, para que pueda ser evaluado y utilizado en producción.	
estimación (horas):	10
Prioridad:	15
Dependencia	H09

Tabla 7.8 Cuadro historia de usuario H09

Tarea T15: Revisión final del código	
Realizar una revisión exhaustiva del código del proyecto, asegurando que cumpla con los estándares de calidad, sea eficiente y esté correctamente documentado.	
estimación (horas)	2
Prioridad	13

Dependencia	
Historia de usuario	<i>H10</i>

Tabla 7.9 Cuadro tarea T15: Revisión final del código

Tarea T16: Pruebas finales	
Llevar a cabo pruebas funcionales y de rendimiento para verificar que el sistema funcione según los requerimientos y no tenga errores críticos.	
estimación (horas)	<i>3</i>
Prioridad	<i>13</i>
Dependencia	<i>T15</i>
Historia de usuario	<i>H10</i>

Tabla 8.0 Cuadro tarea T16: Pruebas finales

Tarea T17: Documentación de pruebas	
Elaborar un informe de las pruebas realizadas, detallando los casos de prueba, los resultados obtenidos y cualquier hallazgo relevante.	
estimación (horas)	<i>2</i>
Prioridad	<i>13</i>
Dependencia	<i>T16</i>
Historia de usuario	<i>H10</i>

Tabla 8.1 Cuadro tarea T17: Documentación de pruebas

Tarea T18: Preparación de la entrega	
Preparar toda la documentación y los archivos necesarios para la entrega final del proyecto. Esto incluirá tanto el código fuente como los manuales de usuario y de instalación.	
estimación (horas)	2
Prioridad	13
Dependencia	T17
Historia de usuario	H10

Tabla 8.2 Cuadro tarea T18: Preparación de la entrega

Tarea T19: Presentación final del proyecto	
Realizar la presentación oficial del proyecto ante el comité evaluador, detallando los resultados alcanzados y respondiendo las preguntas del jurado.	
estimación (horas)	2
Prioridad	13
Dependencia	T18
Historia de usuario	H10

Tabla 8.3 Cuadro tarea T19: Presentación final del proyecto

6.5. Modelo lógico de la base de datos

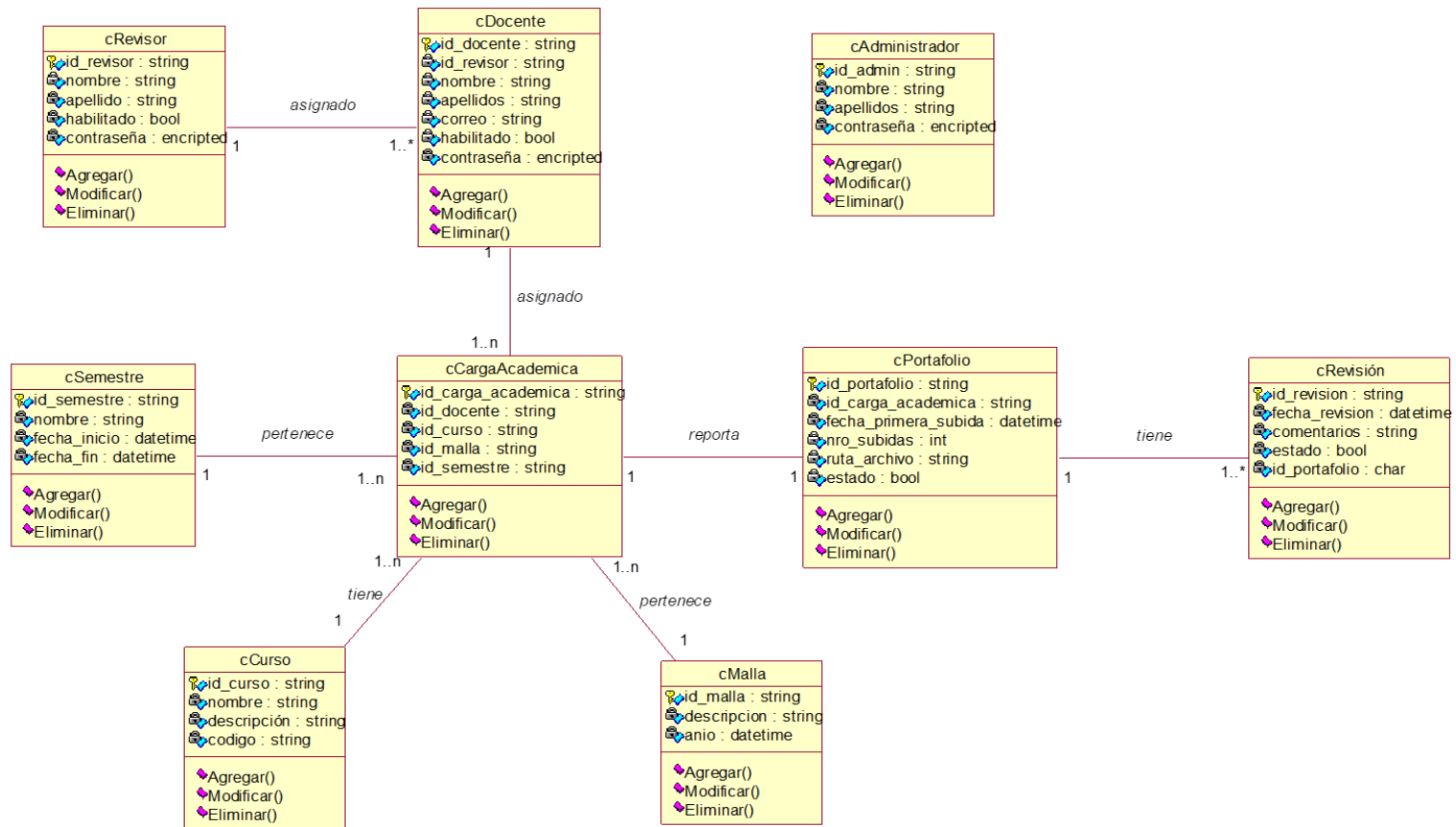


Tabla 8.4 Modelo lógico de la base de datos

6.6. Modelo físico de la base de datos

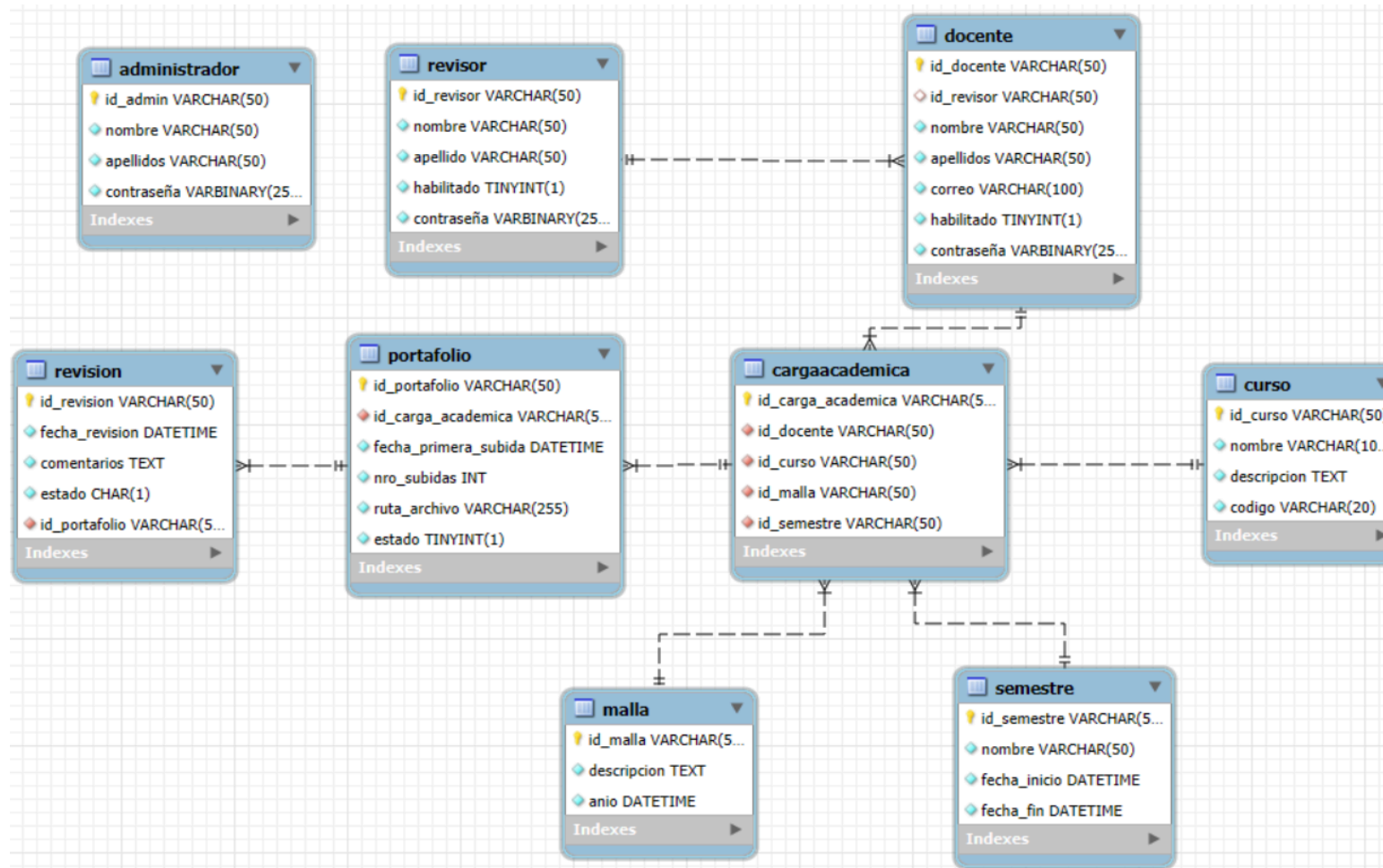


Tabla 8.5 Modelo físico de la base de datos

Script mySQL

```

CREATE DATABASE IF NOT EXISTS bd_portafolios;
USE bd_portafolios;

-- Tabla Revisor
CREATE TABLE Revisor (
    id_revisor VARCHAR(50) PRIMARY KEY,
    nombre VARCHAR(50) NOT NULL,
    apellido VARCHAR(50) NOT NULL,
    habilitado BOOLEAN NOT NULL,
    contraseña VARBINARY(255) NOT NULL
);

-- Tabla Docente
CREATE TABLE Docente (
    id_docente VARCHAR(50) PRIMARY KEY,
    id_revisor VARCHAR(50),
    nombre VARCHAR(50) NOT NULL,
    apellidos VARCHAR(50) NOT NULL,
    correo VARCHAR(100) NOT NULL,
    habilitado BOOLEAN NOT NULL,
    contraseña VARBINARY(255) NOT NULL,
    FOREIGN KEY (id_revisor) REFERENCES Revisor(id_revisor)
);

-- Tabla Administrador
CREATE TABLE Administrador (
    id_admin VARCHAR(50) PRIMARY KEY,
    nombre VARCHAR(50) NOT NULL,
    apellidos VARCHAR(50) NOT NULL,
    contraseña VARBINARY(255) NOT NULL
);

-- Tabla Semestre
CREATE TABLE Semestre (
    id_semestre VARCHAR(50) PRIMARY KEY,
    nombre VARCHAR(50) NOT NULL,
    fecha_inicio DATETIME NOT NULL,
    fecha_fin DATETIME NOT NULL
);

-- Tabla Malla
CREATE TABLE Malla (
    id_malla VARCHAR(50) PRIMARY KEY,
    descripcion TEXT NOT NULL,
    anio DATETIME NOT NULL
);

-- Tabla Curso
CREATE TABLE Curso (
    id_curso VARCHAR(50) PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL,
    descripcion TEXT NOT NULL,
    codigo VARCHAR(20) NOT NULL
);

```

```

-- Tabla CargaAcademica
CREATE TABLE CargaAcademica (
    id_carga_academica VARCHAR(50) PRIMARY KEY,
    id_docente VARCHAR(50) NOT NULL,
    id_curso VARCHAR(50) NOT NULL,
    id_malla VARCHAR(50) NOT NULL,
    id_semestre VARCHAR(50) NOT NULL,
    FOREIGN KEY (id_docente) REFERENCES Docente(id_docente),
    FOREIGN KEY (id_curso) REFERENCES Curso(id_curso),
    FOREIGN KEY (id_malla) REFERENCES Malla(id_malla),
    FOREIGN KEY (id_semestre) REFERENCES Semestre(id_semestre)
);

-- Tabla Portafolio
CREATE TABLE Portafolio (
    id_portafolio VARCHAR(50) PRIMARY KEY,
    id_carga_academica VARCHAR(50) NOT NULL,
    fecha_primera_subida DATETIME NOT NULL,
    nro_subidas INT NOT NULL,
    ruta_archivo VARCHAR(255) NOT NULL,
    estado BOOLEAN NOT NULL,
    FOREIGN KEY (id_carga_academica) REFERENCES CargaAcademica(id_carga_academica)
);

-- Tabla Revision
CREATE TABLE Revision (
    id_revision VARCHAR(50) PRIMARY KEY,
    fecha_revision DATETIME NOT NULL,
    comentarios TEXT NOT NULL,
    estado CHAR(1) NOT NULL,
    id_portafolio VARCHAR(50) NOT NULL,
    FOREIGN KEY (id_portafolio) REFERENCES Portafolio(id_portafolio)
);

```

Tabla 8.6 Código Script mySQL