

Universidad Nacional de San Antonio Abad del Cusco

Facultad de Ingeniería Eléctrica, Electrónica,
Informática y Mecánica



Proyecto de Ingeniería de Software I

Sistema web para la Revisión de Portafolios de Docentes del
Departamento

Profesor: Mgt. Willian Zamalloa Paro

Autores: Arana Flores Shaiel Almendra
Cabrera Mejia Cristian Andy
Callañaupa Sallo Julio Cesar
Gallegos Cjuiro Luis Alberto
Lima Esperilla Katerine Candy
Huerta Medina Vito Jhon
Mamani Meza Johana Maria
Montañez Choque Willians
Poma Supo Juan Gabriel
Reynaga Flores Angela Vanessa
Sumire Ccahuana Kevin Aron

Fecha de Presentación: febrero 1, 2024

Índice

1. Resumen	3
2. Introducción	4
3. Objetivos	5
3.1. Objetivos Generales	5
3.2. Objetivos Específicos	5
4. Organigrama del Proyecto	6
4.1. Líder del Proyecto	6
4.2. Analista de requerimientos	6
4.3. Diseñador UX/IU	6
4.4. Desarrolladores Frontend	6
4.5. Desarrolladores Backend	6
4.6. Tester / QA	7
4.7. DBA (Administrador de base de datos)	7
4.8. Despliegue / DevOps	7
4.9. Documentación y soporte	7
4.10. Gestor de proyectos / Scrum Master	7
4.11. Experto en seguridad	8
5. Glosario de términos	8
6. Plan de proyecto	10
7. Requerimientos	11
7.1. Requerimientos Funcionales:	11
7.2. Requerimientos No Funcionales:	11
7.3. Matriz de rastreo de los requerimientos:	11
8. Diseño	12
8.1. Arquitectura Funcional del Sistema	12
8.1.1. Contexto y alcance del Sistema	12
8.1.2. Componentes de la Arquitectura del Sistema	12
8.1.3. Flujo de Procesos	12
8.1.4. Gestión de datos	13
8.2. Arquitectura Técnica del Sistema	13
8.2.1. Topología de Software	13

8.2.2. Topología de Hardware	16
8.3. Documentos Basados en UML	17
8.3.1. Casos de Uso	17
8.3.2. Tabla extendida de Casos de de Uso	18
8.3.3. Diagrama de Secuencias	20
8.4. Descripción de la metodología	21
8.5. Modelo lógico de la DB	21
9. Costos	22
9.1. COSMIC:	22
9.1.1. Costo del equipo de trabajo de desarrollo de software	24
9.1.2. Costo del proyecto de software	25
9.2. COCOMO II (Constructive Cost Model):	26
9.2.1. Costos por Hardware	26
9.2.2. Costos por Personal	27
9.2.3. Costo en la elaboración del proyecto	27
9.2.4. Costo total del Software	27
10. Anexos	28
11. Referencias	53

1. Resumen

El proyecto consiste en el desarrollo de un sistema de información web para la revisión de portafolios docentes en el Departamento Académico de Ingeniería Informática (DAII) de la Universidad Nacional de San Antonio Abad del Cusco (UNSAAC). Este sistema permitirá gestionar los portafolios, asignar revisores, notificar a los docentes sobre las observaciones o faltas en sus portafolios y generar informes al final de cada revisión.

El equipo del proyecto estará compuesto por un líder de proyecto, un analista de requerimientos, un diseñador UX/IU, desarrolladores frontend y backend, un tester/QA, un administrador de base de datos (DBA), un experto en despliegue/DevOps, un encargado de documentación y soporte, un gestor de proyectos/Scrum Master y un experto en seguridad.

El proyecto seguirá un plan detallado que incluirá la definición de los requerimientos funcionales y no funcionales, el diseño de la arquitectura funcional y técnica del sistema, la creación de documentos basados en UML, la descripción de la metodología, la definición del modelo lógico y físico de la base de datos, la programación, las pruebas, la estimación de costos, el despliegue de la aplicación y la documentación de mantenimiento.

2. Introducción

La Universidad Nacional de San Antonio Abad del Cusco (UNSAAC), a través de su Departamento Académico de Ingeniería Informática (DAII), ha identificado la necesidad de un sistema de información web que facilite y mejore los procesos relacionados con la revisión de los portafolios docentes. Este sistema será una herramienta esencial para el cumplimiento del criterio 6 (CUERPO DE PROFESORES) establecido por la comisión de acreditación ICACIT.

El portafolio docente es un elemento crucial en la evaluación del desempeño docente. Está compuesto por varios documentos que reflejan la labor educativa del profesor durante un semestre académico. Estos documentos pueden variar dependiendo de si las asignaturas impartidas son teóricas, prácticas o una combinación de ambas. Además, es importante destacar que la carga académica de un docente puede variar cada semestre, lo que añade una capa adicional de complejidad al proceso.

Actualmente, cada semestre, cada docente debe presentar su portafolio a una comisión revisora compuesta por un presidente y entre 6 a 15 integrantes. Esta comisión tiene la responsabilidad de revisar los portafolios y determinar si cumplen con los requisitos establecidos. Durante el semestre se realizan tres revisiones, y para cada una de ellas se asignan docentes a un revisor. Sin embargo, este proceso puede ser desafiante y consumir mucho tiempo. Si un portafolio no cumple con los requisitos, el docente debe ser notificado e informado sobre las observaciones o faltas detectadas. Además, al finalizar las fechas de revisión, la comisión debe emitir un informe dirigido al jefe del departamento de la escuela profesional. Este informe debe indicar si se ha cumplido o no con lo requerido en la revisión correspondiente.

Por lo tanto, el objetivo principal de este proyecto es desarrollar un sistema que automatice estos procesos y facilite la gestión de los portafolios docentes. Este sistema permitirá a la UNSAAC mejorar la eficiencia y eficacia de las revisiones, asegurando así que se cumplan los estándares de calidad educativa.

3. Objetivos

3.1. Objetivos Generales

Desarrollar un sistema de información web que facilite y mejore los procesos de revisión de los portafolios docentes en el DAIH de la UNSAAC.

3.2. Objetivos Específicos

- Crear una interfaz de usuario intuitiva y fácil de usar para los docentes y los miembros de la comisión revisora.
- Implementar una funcionalidad para la gestión de portafolios que permita a los docentes subir sus documentos y a los revisores acceder a ellos.
- Desarrollar un sistema de asignación de revisores que permita asignar docentes a un revisor para cada revisión.
- Implementar un sistema de notificaciones para informar a los docentes sobre las observaciones o faltas en sus portafolios.
- Generar informes al final de cada revisión que indiquen si se ha cumplido o no con lo requerido.
- Asegurar la seguridad de la información y los datos del sistema.
- Realizar pruebas exhaustivas para garantizar la funcionalidad y eficiencia del sistema.
- Desplegar la aplicación y proporcionar documentación y soporte para su mantenimiento.

4. Organigrama del Proyecto

4.1. Líder del Proyecto

- Responsabilidades:
 - Coordinar y supervisar el equipo.
 - Tomar decisiones clave.
 - Comunicar el progreso del proyecto a los interesados.
- Perfil:
 - Persona con experiencia en desarrollo web, liderazgo y habilidades de comunicación.

4.2. Analista de requerimientos

- Responsabilidades:
 - Recolectar y documentar los requisitos del cliente.
 - Crear casos de uso y especificaciones técnicas.
- Perfil:
 - Persona con habilidades de análisis y comunicación.

4.3. Diseñador UX/IU

- Responsabilidades:
 - Crear wireframes y prototipos.
 - Diseñar la interfaz de usuario.
- Perfil:
 - Persona con experiencia en diseño de interfaces y experiencia de usuario.

4.4. Desarrolladores Frontend

- Responsabilidades:
 - Implementar la interfaz de usuario.
 - Trabajar con HTML, CSS y JavaScript.
- Perfil:
 - Personas con experiencia en desarrollo web frontend.

4.5. Desarrolladores Backend

- Responsabilidades:
 - Desarrollar la lógica de negocio y la base de datos.
 - Trabajar con Laravel y PHP.
- Perfil:
 - Personas con experiencia en desarrollo web backend.

4.6. Tester / QA

- Responsabilidades:
 - Realizar pruebas de calidad
 - Reportar y rastrear errores.
- Perfil:
 - Persona con habilidades de prueba y atención al detalle.

4.7. DBA (Administrador de base de datos)

- Responsabilidades:
 - Diseñar y mantener la base de datos.
 - Optimizar consultas y asegurar la integridad de los datos.
- Perfil:
 - Persona con experiencia en administración de bases de datos.

4.8. Despliegue / DevOps

- Responsabilidades:
 - Configurar y mantener el entorno de desarrollo y producción.
 - Automatizar el despliegue y las pruebas.
- Perfil:
 - Persona con habilidades de DevOps y configuración de servidores.

4.9. Documentación y soporte

- Responsabilidades:
 - Crear documentación técnica y de usuario.
 - Brindar soporte a los usuarios finales.
- Perfil:
 - Persona con habilidades de documentación y soporte.

4.10. Gestor de proyectos / Scrum Master

- Responsabilidades:
 - Facilitar reuniones de planificación y seguimiento.
 - Mantener actualizada la pizarra Kanban o tablero de tareas.
- Perfil:
 - Persona con experiencia en gestión de proyectos ágiles.

4.11. Experto en seguridad

- Responsabilidades:
 - Identificar y mitigar vulnerabilidades de seguridad.
 - Implementar buenas prácticas de seguridad en el desarrollo.
- Perfil:
 - Persona con experiencia en seguridad.

5. Glosario de términos

- **Portafolio:** Colección de documentos que un profesor debe presentar para la revisión. Estos documentos reflejan el trabajo y las actividades realizadas durante un semestre académico.
- **Revisor:** Un revisor es un profesor que tiene la responsabilidad adicional de revisar los portafolios de otros profesores. No todos los profesores son revisores.
- **Teórico/Práctico:** Curso que combina elementos teóricos y prácticos. Los requisitos para los portafolios de estos cursos pueden diferir de los cursos puramente teóricos o prácticos.
- **Regularizar:** Proceso de asegurarse de que todos los documentos requeridos estén presentes en el portafolio y cumplan con los estándares establecidos.
- **Arquitectura funcional:** Estructura de alto nivel de un sistema que muestra cómo se organizan y se relacionan entre sí las diferentes funciones del sistema.
- **Laravel (PHP Framework):** Laravel es un marco de trabajo de código abierto de PHP que proporciona una amplia gama de características y herramientas para el desarrollo web. Laravel facilita la gestión de rutas, controladores, modelos y vistas, y ofrece una sólida seguridad y capacidades de gestión de bases de datos.
- **PHP (Lenguaje de Programación):** Lenguaje de programación de código abierto muy popular, especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.
- **MySQL:** Es un sistema de gestión de bases de datos relacional de código abierto basado en SQL (Structured Query Language).
- **HTML, CSS y JavaScript:** Son los lenguajes fundamentales para la creación de sitios web. HTML se utiliza para estructurar el contenido, CSS para el diseño y la presentación, y JavaScript para la interactividad en el lado del cliente.
- **Git (Control de Versiones):** Git es un sistema de control de versiones distribuido que permite a varios desarrolladores trabajar en un proyecto al mismo tiempo. Se utiliza para rastrear y gestionar cambios en el código fuente y coordinar el trabajo entre los miembros del equipo.
- **Base de Datos:** Es donde el sistema almacena toda la información. En este caso, se utilizará MySQL en la versión de 64 bits.
- **Servidor de aplicaciones:** Ejecuta toda la lógica del negocio y es el único que se conecta con la base de datos.
- **Servidor Web:** Es un software que sirve contenido web a través de Internet. Recibe una solicitud del cliente web, la procesa y luego envía la respuesta al cliente web. Servidor encargado de atender las solicitudes al sistema desde clientes web.
- **Cliente:** Permite a los usuarios acceder a las funcionalidades del sistema.
- **Cliente Web:** Componente que permite a los usuarios ingresar al sistema mediante un navegador Web.

- **Flujo de procesos:** Secuencia de pasos que los usuarios deben seguir para interactuar con el sistema.
- **Gestión de datos:** Sistema maneja y organiza la información. Incluye aspectos como la creación de una base de datos, la gestión de usuarios y permisos, el almacenamiento de cargas académicas y la integración con sistemas externos.
- **Arquitectura MVC (Modelo-Vista-Controlador):** Patrón de diseño de software que separa una aplicación en tres componentes principales: modelo, vista y controlador. Se utiliza para organizar el código de manera que sea más fácil de mantener y entender.
- **Modelo:** En el patrón MVC, el modelo representa los datos y la lógica de negocio de la aplicación. Los modelos se utilizan para interactuar con la base de datos y recuperar o almacenar información.
- **Vista:** En el patrón MVC, la vista se encarga de la presentación de la información al usuario. Las vistas definen cómo se muestran los datos y la interfaz de usuario.
- **Controlador:** En el patrón MVC, el controlador actúa como intermediario entre el modelo y la vista. Los controladores gestionan las solicitudes HTTP, recuperan datos del modelo y los pasan a la vista para su representación.
- **Middlewares:** Son componentes que permiten realizar tareas antes o después de que una solicitud llegue al controlador. Esto se utiliza comúnmente para aplicar autenticación, autorización y otros filtros a las solicitudes.
- **Servicios:** En una aplicación basada en Laravel, es común utilizar servicios para abstraer la lógica de negocio compleja y mantener los controladores más delgados. Estos servicios encapsulan operaciones específicas y se pueden inyectar en los controladores cuando sea necesario.
- **Bibliotecas y Paquetes:** En PHP, es común utilizar bibliotecas y paquetes de terceros para simplificar el desarrollo. Estos pueden proporcionar funcionalidades adicionales o facilitar la realización de ciertas tareas.
- **Módulos Front-end:** Componentes del lado del cliente, como JavaScript, CSS y bibliotecas de front-end, que se utilizan para mejorar la experiencia del usuario y la interfaz de la aplicación web.

6. Plan de proyecto

	Dia 1-3	Dia 4-7	Dia 8-11	Dia 12-15	Dia 16-19	Dia 20-23	Dia 24-27	Dia 28-31	Dia 32-35	Dia 36-39	Dia 40-43	Dia 44-47	Dia 48-53	Dia 54-57	Dia 58-62
Escalabilidad y mantenimiento															
Aprobacion final															
Despliegue															
Generacion de manuales y documentacion															
Pruebas de Aceptacion															
Optimizacion de rendimiento															
Pruebas de funcionamiento completo															
Acomprar los modulos															
Pruebas de cada Modulo															
Implementar modulo de Generacion de informes															
Implementar modulo de Revision de portafolios															
Implementar modulo Gestion de portafolios															
Implementar modulo de Gestion de usuarios															
Implementacion modulo de autentificacion															
configuracion entorno															
Definir las herramientas de desarrollo															
diseño de BD															
diseño de arquitectura															
casos de uso															
modelado de procesos															
analisis y diseño															
Identificacion de requerimientos															
asignacion de roles															

Figura 1: Plan de proyecto

7. Requerimientos

7.1. Requerimientos Funcionales:

- Autenticación: La interfaz debe permitir el inicio de sesión para los tres tipos de usuarios: administrador, revisor y profesor.
- Administrador: El administrador debe tener acceso a las funciones CRUD para gestionar revisores y profesores. Debe poder asignar revisores a profesores y viceversa. Se debe mostrar una interfaz que permita al administrador visualizar la distribución de profesores y revisores.
- Un docente que es también revisor debe tener la opción de acceder como docente o como revisor. Un docente que también es administrador y revisor debe tener la opción de acceder como cualquiera de esos tres.
- Revisor/Profesor: Un revisor puede ser asignado a varios profesores para revisar sus portafolios. La relación entre revisores y profesores debe reflejarse claramente en la interfaz, mostrando los docentes a cargo de un revisor.
- Un revisor puede acceder solo al portafolio de los docentes que tiene a su cargo.

7.2. Requerimientos No Funcionales:

- La interfaz será programada en Laravel.
- Se utilizará MySQL y Workbench como sistema de gestión de bases de datos.
- El control de versiones se realizará mediante Github.

7.3. Matriz de rastreo de los requerimientos:

Requerimientos	Funcionalidad	Análisis	Diseño	Codificación	Pruebas	Validación
Interfaz	Funcional	2	2	2	1	1
Administrador CRUD	Funcional	2	2	2	1	1
Mostrar asignaciones	Funcional	2	2	2	1	1
Mostrar revisores con sus docentes	Funcional	2	2	2	1	1
interfaz en Laravel	No Funcional	2	1	1	1	1
Base de Datos en MySQL	No Funcional	2	1	1	1	1
Control de Versiones en GitHub	No Funcional	2	1	1	1	1

Cuadro 1: Matriz de rastreo de requerimientos

8. Diseño

8.1. Arquitectura Funcional del Sistema

8.1.1. Contexto y alcance del Sistema

La arquitectura funcional de nuestra aplicación web de gestión académica tiene como objetivo principal optimizar la planificación de cursos, horarios y recursos docentes en el contexto de una escuela profesional. Este marco conceptual proporcionará una visión clara de cómo la aplicación abordará las complejidades asociadas con la administración de cargas académicas.

8.1.2. Componentes de la Arquitectura del Sistema

En el siguiente diagrama se muestran los diferentes componentes del sistema.

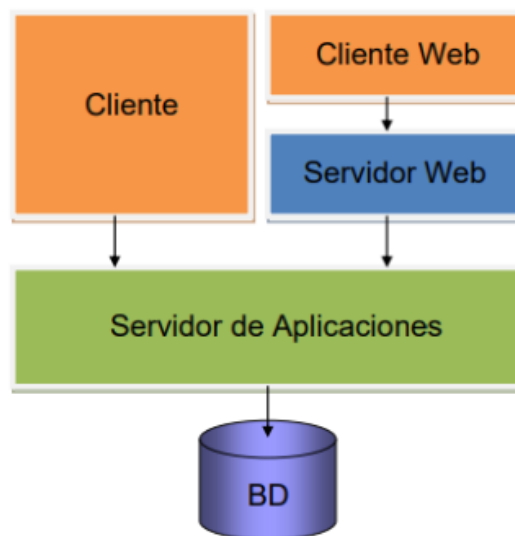


Figura 2: componentes del sistema

1. Base de Datos: Donde el sistema almacena toda la información. Para este proyecto, la base de datos a usar será MySQL en la versión de 64 bits, y usaremos Workbench para la administración y diseño de las bases de datos.
2. Servidor de Aplicaciones: Ejecuta toda la lógica del negocio. El servidor de aplicaciones es el único que se conecta con la base de datos.
3. Servidor Web: Servidor encargado de atender las solicitudes al sistema desde clientes web.
4. Cliente: Permite a los usuarios acceder a las funcionalidades del sistema, para ello utiliza el servidor de aplicaciones donde se realiza el procesamiento de la operación requerida por el usuario.
5. Cliente Web: Componente que permite a los usuarios ingresar al sistema mediante un navegador web. La comunicación entre este cliente y el servidor web está optimizada de modo que minimiza la cantidad de datos que son transferidos y optimizan el tiempo de respuesta del sistema.

8.1.3. Flujo de Procesos

1. **Inicio de sesión:** Los usuarios accederán a la aplicación mediante un proceso de inicio de sesión seguro. Se verificarán las credenciales para garantizar el acceso autorizado.

2. **Panel Principal:** Después del inicio de sesión, los usuarios serán dirigidos al panel principal. Aquí encontrarán un resumen de las cargas académicas actuales y herramientas para gestionarlas.
3. **Gestión de Cursos:** Los administradores podrán agregar, editar o eliminar cursos. Podrán asignar profesores a cursos específicos y establecer requisitos si es necesario.
4. **Planificación de Horarios:** La función de planificación permitirá a los administradores y profesores organizar horarios de clases. Se asignarán aulas y recursos de manera eficiente.
5. **Asignación de Recursos Docentes:** Los administradores podrán asignar profesores a cursos y horarios según la disponibilidad y competencias. Se evitarán conflictos de programación.
6. **Consulta de Cargas Académicas:** Profesores y estudiantes podrán consultar fácilmente sus cargas académicas, incluyendo detalles de cursos, horarios y ubicaciones.
7. **Generación de Reportes:** La aplicación permitirá la generación de informes personalizados sobre la planificación académica, facilitando el análisis y la toma de decisiones.
8. **Notificaciones y Alertas:** Se implementarán notificaciones automáticas para alertar sobre cambios en horarios, asignaciones o eventos importantes relacionados con la gestión académica.
9. **Cierre de Sesión:** Los usuarios podrán cerrar sesión de manera segura al finalizar sus actividades en la aplicación.

8.1.4. Gestión de datos

1. **Base de Datos:** La aplicación utilizará una base de datos relacional para almacenar información crítica. Se crearán tablas específicas para cursos, profesores, estudiantes, horarios y otras entidades relevantes.
2. **Usuarios y Permisos:** Se establecerá un sistema de gestión de usuarios que almacene información de inicio de sesión de administradores, profesores y estudiantes. Se asignarán permisos específicos para garantizar el acceso adecuado a las funciones de la aplicación.
3. **Almacenamiento de Cargas Académicas:** La información sobre las cargas académicas, incluyendo detalles de cursos, horarios y asignaciones, se almacenará de manera estructurada y se vinculará a las entidades correspondientes en la base de datos.
4. **Integración con Sistemas Externos:** En caso de integraciones con sistemas externos (por ejemplo, plataformas de gestión estudiantil), se establecerán protocolos seguros para el intercambio de datos.

8.2. Arquitectura Técnica del Sistema

8.2.1. Topología de Software

Arquitectura del Sistema

Arquitectura MVC (Modelo-Vista-Controlador): La aplicación web se basa en el patrón de arquitectura Modelo-Vista-Controlador (MVC), que es característico de Laravel. Esta arquitectura se divide en tres componentes principales:

- **Modelo (Model):** Representa los datos y la lógica de negocio de la aplicación. Los modelos se utilizan para interactuar con la base de datos y recuperar o almacenar información.
- **Vista (View):** Se encarga de la presentación de la información al usuario. Las vistas definen cómo se muestran los datos y la interfaz de usuario.
- **Controlador (Controller):** Actúa como intermediario entre el modelo y la vista. Los controladores gestionan las solicitudes HTTP, recuperan datos del modelo y los pasan a la vista para su representación.

Componentes de Software

Es fundamental comprender los componentes de software que componen el sistema, estos componentes se dividen en funciones específicas y desempeñan un papel importante en la funcionalidad y operación de la aplicación. A continuación, se describen algunos de los componentes:

- **Controladores (Controllers):** En una aplicación Laravel, los controladores juegan un papel crucial. Estos componentes manejan las solicitudes HTTP y actúan como intermediarios entre las rutas y la lógica de negocio. Cada controlador se encarga de una funcionalidad específica y controla la lógica para gestionar esa parte del sistema.
- **Modelos (Models):** Los modelos en Laravel representan las entidades de la base de datos. Definen cómo se accede a los datos y cómo se relacionan entre sí. Cada modelo generalmente corresponde a una tabla de la base de datos y se utiliza para realizar operaciones de consulta y manipulación de datos.
- **Vistas (Views):** Las vistas son responsables de la presentación de datos al usuario. En una aplicación web, las vistas pueden estar compuestas por código HTML, CSS y plantillas de Blade (el motor de plantillas de Laravel). Las vistas muestran la información procesada por los controladores y modelos.
- **Rutas (Routes):** Las rutas definen cómo se accede a las diferentes partes de la aplicación a través de URLs. En Laravel, las rutas se configuran en archivos de rutas y se asignan a controladores y métodos específicos.
- **Middlewares:** Los middlewares son componentes que permiten realizar tareas antes o después de que una solicitud llegue al controlador. Esto se utiliza comúnmente para aplicar autenticación, autorización y otros filtros a las solicitudes.
- **Servicios (Services):** En una aplicación Scrum basada en Laravel, es común utilizar servicios para abstraer la lógica de negocio compleja y mantener los controladores más delgados. Estos servicios encapsulan operaciones específicas y se pueden inyectar en los controladores cuando sea necesario.
- **Bibliotecas y Paquetes:** En PHP, es común utilizar bibliotecas y paquetes de terceros para simplificar el desarrollo. Puedes mencionar cualquier biblioteca o paquete específico que utilices en tu proyecto.
- **Módulos Front-end:** Además de los componentes del lado del servidor, es importante mencionar los componentes del lado del cliente, como JavaScript, CSS y bibliotecas de front-end, que se utilizan para mejorar la experiencia del usuario y la interfaz de la aplicación web.

Tecnologías y Lenguajes

Se utilizan diversas tecnologías y lenguajes para desarrollar y mantener la aplicación. Estas tecnologías son fundamentales para garantizar el rendimiento, la eficiencia y la funcionalidad de la aplicación. Aquí se describen las tecnologías y lenguajes clave:

- Laravel (Framework)
- PHP (Lenguaje de backend)
- Base de Datos (MySQL)
- HTML, CSS y JavaScript (Lenguajes de frontend)
- Git (Controlador de Versiones)

Comunicación entre Componentes

La comunicación efectiva entre los componentes del sistema es esencial para garantizar que la aplicación funcione de manera eficiente y cumpla con sus objetivos:

- **Controlador a Vista (View):** En Laravel, los controladores recuperan datos y lógica de negocio, transmitiéndolos a las vistas para la renderización de la interfaz de usuario en el navegador.
- **Controlador a Modelo (Model):** Los controladores se comunican con los modelos para acceder y manipular datos en la base de datos, utilizando la capa de abstracción de base de datos proporcionada por Laravel.
- **Rutas a Controladores:** Las rutas en Laravel dirigen las solicitudes HTTP a controladores específicos, estableciendo una conexión crucial entre las solicitudes entrantes y el manejo correspondiente.
- **Middleware a Controlador:** Los middlewares intervienen antes de llegar al controlador, aplicando tareas como autenticación y validaciones para garantizar el manejo seguro de las solicitudes.
- **Servicios y Controladores:** Para mantener controladores delgados y modularizar la lógica de negocio, los servicios encapsulan operaciones complejas, permitiendo que los controladores se comuniquen con ellos según sea necesario.
- **Front-End y API:** La interacción entre el front-end y el controlador se realiza mediante solicitudes HTTP, mientras que una API, si está disponible, permite la integración externa a través de solicitudes para acceder a datos y funcionalidades del sistema.
- **Base de Datos y Modelo:** Los modelos de Laravel gestionan la comunicación con la base de datos, realizando operaciones de lectura y escritura mediante SQL o utilizando el sistema de consultas Eloquent.
- **Control de Versiones y Colaboración:** Git y sistemas de control de versiones facilitan la colaboración al gestionar cambios en el código fuente y coordinar el trabajo del equipo.
- **Interacción con la Base de Datos:** La comunicación entre los componentes del software y el DBMS se realiza a través de consultas SQL o el sistema de consultas Eloquent en Laravel, proporcionando una interfaz de alto nivel para interactuar con la base de datos.

Flujo de Datos

El flujo de datos describe cómo la información y los datos se mueven a través del sistema, desde la entrada del usuario hasta la base de datos y viceversa:

- **Entrada del Usuario:** Usuarios interactúan mediante formularios, navegación o solicitudes.
- **Rutas y Controladores:** Rutas dirigen solicitudes a controladores para gestionar la lógica de negocio.
- **Modelos y Base de Datos:** Controladores interactúan con modelos para acceder y manipular datos en la base de datos.
- **Lógica de Negocio y Servicios:** Lógica compleja se encapsula en servicios para tareas especializadas.
- **Respuesta al Usuario:** Controladores generan respuestas, ya sea vistas HTML o JSON para API.
- **Interfaz de Usuario (UI):** Respuestas se envían al navegador para mostrar la interfaz.
- **Actualización de Datos:** Controladores actualizan la base de datos en acciones como enviar formularios.
- **Comunicación Externa:** Posibilidad de comunicarse con sistemas externos mediante API.
- **Registro de Auditoría y Seguridad:** Se registra la actividad del usuario, y Laravel ofrece seguridad incorporada.

8.2.2. Topología de Hardware

Requisitos de Hardware

Los requisitos de hardware son fundamentales para garantizar que la aplicación web funcione de manera óptima y que pueda gestionar la carga de usuarios prevista.

- **Servidor Web:**
 - **Mínimo:** Recomendado servidor con al menos 2 núcleos de CPU y 4 GB de RAM.
 - **Recomendado:** Servidor con 4 o más núcleos, 8 GB de RAM o más, y almacenamiento SSD.
- **Sistema Operativo:**
 - **Mínimo:** Linux o Windows compatible con versiones de Laravel y PHP.
 - **Recomendado:** Distribución de Linux actualizada con parches de seguridad.
- **Base de Datos:**
 - **Mínimo:** Instancia de base de datos compatible con Laravel y requisitos mínimos.
 - **Recomendado:** Base de datos escalable según necesidades y recursos adecuados.
- **Recursos de Red:**
 - **Mínimo:** Conexión a Internet estable con suficiente ancho de banda.
 - **Recomendado:** Conexión de alta velocidad y configuraciones de red redundantes.
- **Almacenamiento:**
 - **Mínimo:** Almacenamiento adecuado para la aplicación y base de datos.
 - **Recomendado:** Almacenamiento SSD de alto rendimiento para una respuesta más rápida.
- **Memoria:**
 - **Mínimo:** 4 GB de RAM para funcionamiento básico.
 - **Recomendado:** 8 GB de RAM o más para mayor capacidad de respuesta.
- **Procesador:**
 - **Mínimo:** CPU de al menos 2 núcleos.
 - **Recomendado:** CPU de 4 núcleos o más para manejar más solicitudes y procesamiento.
- **Escalabilidad:**
 - Infraestructura escalable que permite aumentar recursos según demanda y cantidad de usuarios.
- **Seguridad:**
 - Medidas de seguridad como cortafuegos y monitoreo para proteger la aplicación y datos.
- **Respaldo y Recuperación:**
 - Sistema de respaldo y recuperación para garantizar disponibilidad e integridad de datos en fallas del hardware.

8.3. Documentos Basados en UML

8.3.1. Casos de Uso

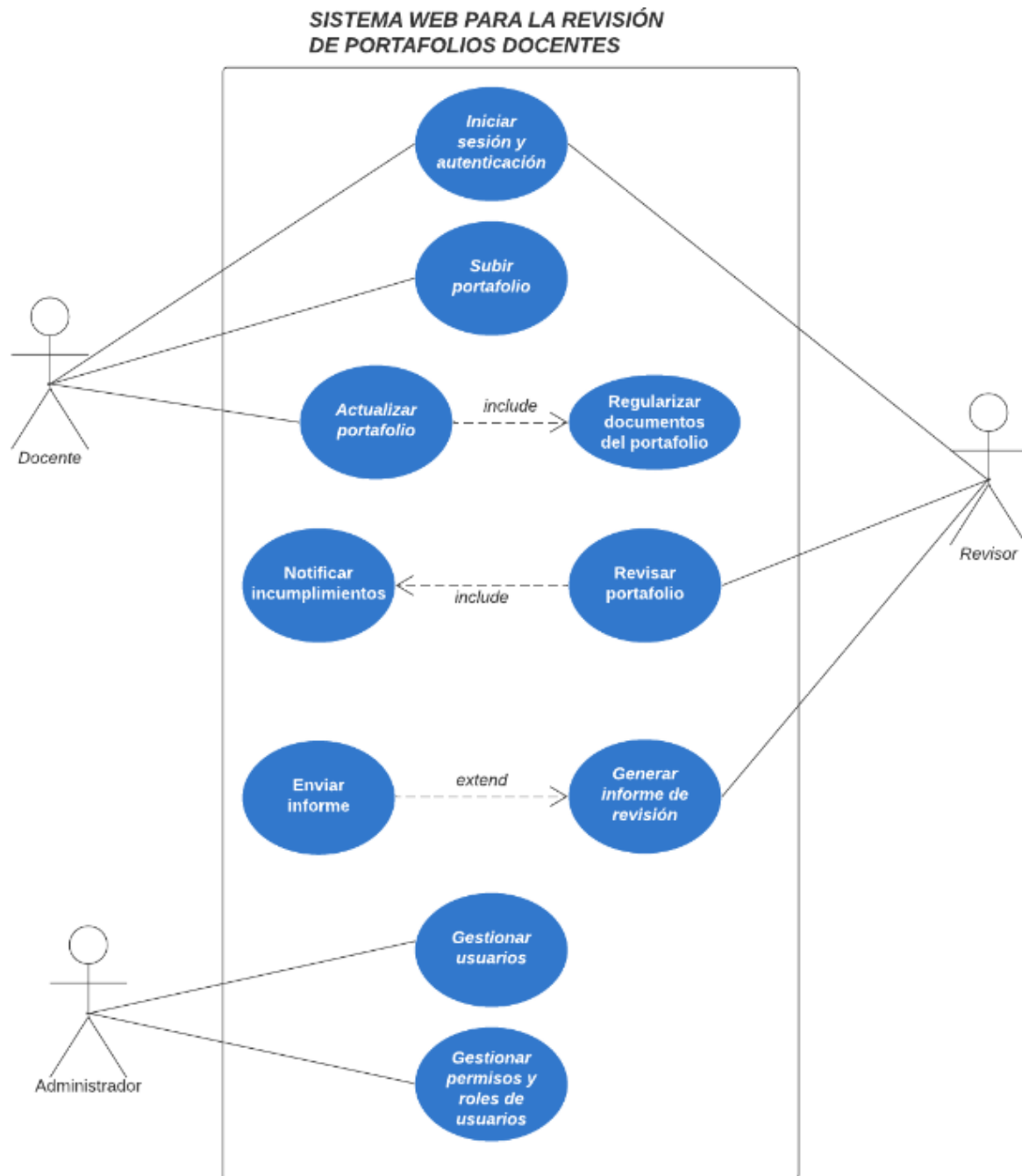


Figura 3: componentes del sistema

8.3.2. Tabla extendida de Casos de de Uso

Caso de Uso	Iniciar sesión y autenticación
Tipo	Secundario
Actores	Docente, Revisor
Descripción	Consiste en permitir que los usuarios autorizados ingresen al sistema.
Caso de Uso	Subir portafolio
Tipo	Primario
Actores	Docente
Descripción	Los docentes cargan sus portafolios al sistema.
Caso de Uso	Actualizar portafolio
Tipo	Secundario
Actores	Docente
Descripción	Consiste en permitir a los docentes realizar la modificación, eliminación y poder actualizar su portafolio y así, subirlo al sistema.
Caso de Uso	Regularizar documentos
Tipo	Secundario
Actores	Docente
Descripción	Consiste en permitir a los docentes terminar de subir los documentos que les faltan o en caso de que estén mal volverlos a enviar.
Caso de Uso	Revisar portafolio
Tipo	Primario
Actores	Revisor
Descripción	Consiste en permitir a los revisores verificar los portafolios de los docentes que se les asignaron, para poder confirmar que esté correcto.

Caso de Uso	Notificar incumplimientos
Tipo	Secundario
Actores	Revisor
Descripción	Consiste en enviar una notificación al profesor después de revisar su portafolio y ver que falten documentos o en caso de que falten documentos en el portafolio, se le enviará un recordatorio 24 horas antes de la fecha establecida de entrega.
Caso de Uso	Generar informe de la revisión
Tipo	Secundario
Actores	Revisor
Descripción	Los revisores luego de hacer la verificación podrán generar un reporte que contenga la información de que la entrega de los portafolios fueron correctas.
Caso de Uso	Enviar informe
Tipo	Secundario
Actores	Revisor
Descripción	Los revisores deben entregar el informe luego de generarlo.
Caso de Uso	Gestionar usuarios
Tipo	Primario
Actores	Administrador
Descripción	El administrador del sistema podrá crear, modificar y eliminar cuentas de docentes
Caso de Uso	Gestionar permisos y roles de usuarios
Tipo	Primario
Actores	Administrador
Descripción	El administrador podrá asignar roles y permisos a los docentes, a los revisores y otros usuarios.

8.3.3. Diagrama de Secuencias

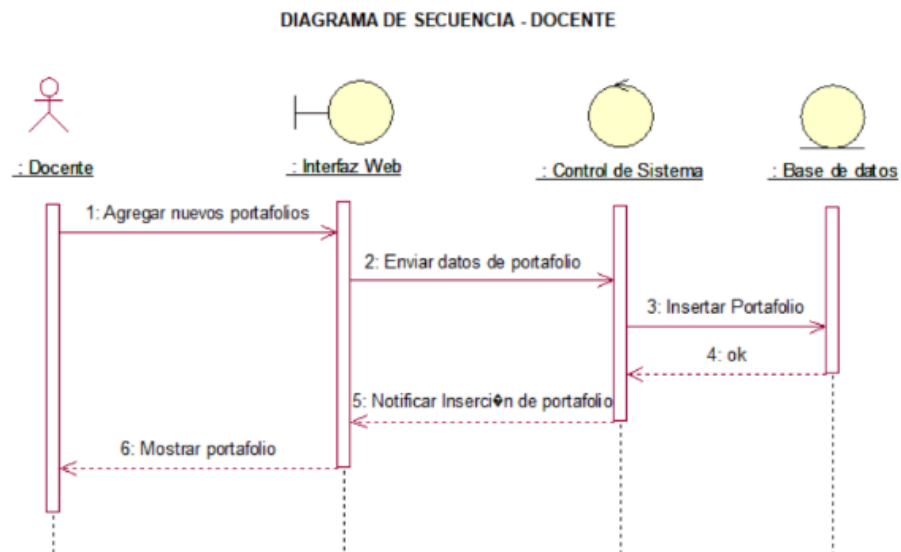


Figura 4: componentes del sistema

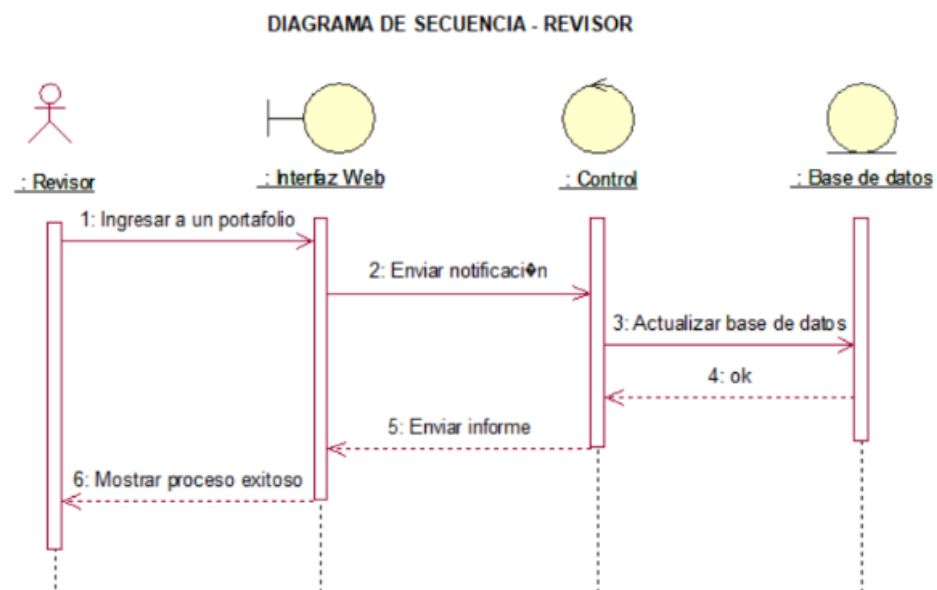


Figura 5: componentes del sistema

8.4. Descripción de la metodología

La metodología que emplearemos será Scrum, el enfoque de trabajo se sustenta en ciclos de trabajo o sprints, los cuales tienen una duración que puede variar desde semanas hasta meses, y en cada uno de ellos se entrega una parte del proyecto. En cada sprint se crea una versión mejorada del producto respecto a la anterior. En Scrum, la base fundamental de su metodología ágil radica en tomar todas las decisiones en base a la información disponible y a la experiencia propia de los miembros del equipo.[2]

■ Roles:

- **Product Owner:** Su responsabilidad principal recae en maximizar el valor del producto que surge del esfuerzo del equipo Scrum. Es responsable de comprender las necesidades y motivaciones de los clientes.
- **Scrum Master:** Se encarga de ayudar al equipo a entender y adherirse a la teoría de Scrum, asegurando que se lleven a cabo las reuniones importantes y motivando al equipo.
- **Scrum Team:** Es el equipo responsable de desarrollar y entregar el producto, siendo una estructura horizontal autosuficiente y autónoma en su gestión.
- **Stakeholders:** Individuos o grupos con interés e influencia en la organización, cuyas acciones determinan los resultados obtenidos.

■ Eventos:

- **Sprint:** Un sprint o "time-box" de un mes o menos se establece para crear un incremento de producto "Terminado", listo para ser utilizado y potencialmente desplegado.
- **Reunión de Planificación de Sprint:** El plan se desarrolla en conjunto por todo el Equipo Scrum. La duración máxima es de ocho horas.
- **Objetivo del Sprint:** Se establece como una meta alcanzable durante el Sprint, brindando al equipo de desarrollo flexibilidad en la funcionalidad implementada.
- **Scrum Diario:** Reunión diaria de 15 minutos donde el Equipo de Desarrollo se sincroniza, inspecciona el trabajo realizado y planifica para las próximas 24 horas.
- **Revisión de Sprint:** Colaboración entre el Equipo Scrum y los interesados para discutir y evaluar lo logrado durante el Sprint.
- **Retrospectiva de Sprint:** Brinda al Equipo Scrum la oportunidad de autocriticarse y elaborar un plan de mejoras para abordar en el siguiente Sprint.

8.5. Modelo lógico de la DB

Para implementar un sistema de roles y permisos en Laravel, usaremos el paquete Laravel Permissions de Spatie. Este paquete crea las siguientes tablas:

- **roles:** Almacena los roles de usuario.
- **permissions:** Almacena los permisos.
- **model_has_roles:** Relaciona modelos con roles, permitiendo que un modelo tenga varios roles y que un rol pueda ser asignado a varios modelos.
- **model_has_permissions:** Relaciona modelos con permisos, permitiendo que un modelo tenga varios permisos y que un permiso pueda ser asignado a varios modelos.
- **role_has_permissions:** Almacena la relación entre un rol y los permisos que ese rol puede realizar.

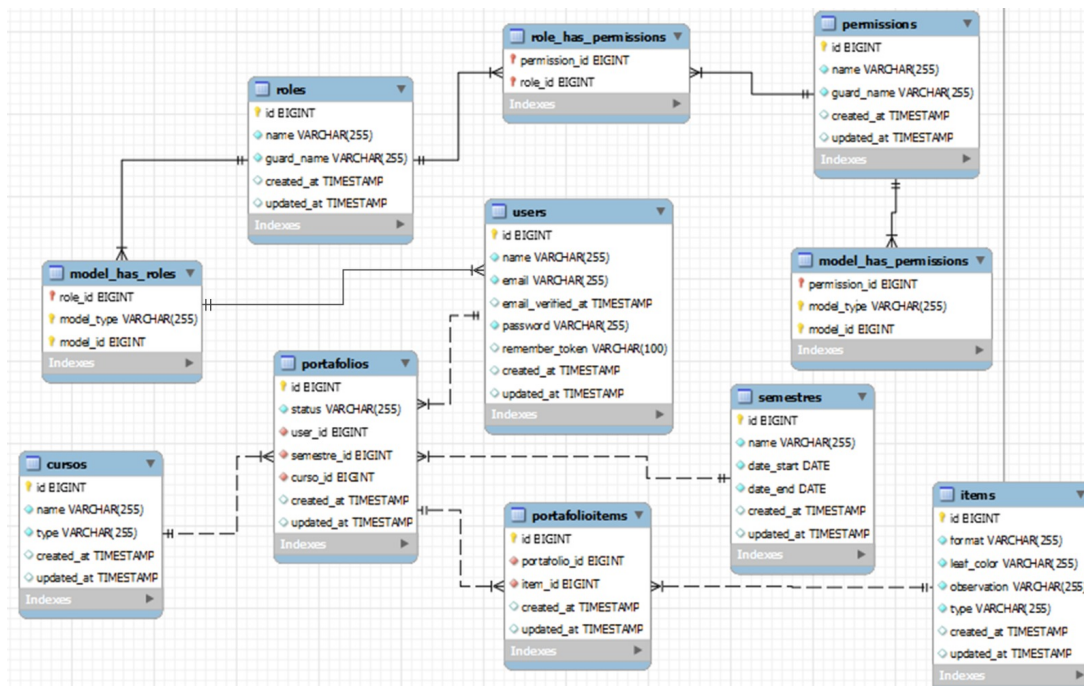


Figura 6: componentes del sistema

9. Costos

A continuación, se presenta una estructura detallada de una aproximación de los costos para el "Sistema Web para la Revisión de Portafolios de Docentes del Departamento". Haciendo uso de la Metodología COSMIC, donde se hará la Identificación de Funciones del Software según la metodología y la Medición de Tamaño Funciona basado en las funciones identificadas.

9.1. COSMIC:

- ★ Identificación de Funciones del Software: Identificación y clasificación de las funciones del software según la metodología COSMIC.
- ★ Medición de Tamaño Funcional:

- Fase 1: Estrategia de medición.

La determinamos identificando cuales son los requerimientos funcionales a medir, cual es el propósito de la medición y quienes son los usuarios funcionales.

Los Requerimientos funcionales del proyecto son:

- Autenticación (EI): Inicio de sesión para administrador, revisor y profesor.
- Gestión de Usuarios (EO): CRUD de revisores y profesores, asignación de revisores a profesores.
- Visualización de Distribución (EO): Visualización de la distribución de profesores y revisores.
- Acceso para Docentes (EI): Elección de rol al iniciar sesión para docentes con múltiples roles.
- Asignación y Acceso de Portafolios (EI y EO): Asignación de revisores a profesores, acceso de revisores a los portafolios de sus docentes asignados.

Propósito de la medición:

Realizar la estimación funcional de la mejora al sistema de sistema de información web para la revisión de portafolios docentes en el Departamento Académico de Ingeniería Informática (DAII) de la Universidad Nacional de San Antonio Abad del Cusco (UNSAAC), comprendida por los requisitos de Autenticación, Gestión de Usuarios, Visualización de Distribución, Acceso para Docentes, Asignación y Acceso de Portafolios.

Usuarios:

Los usuarios son, el Docente, el Revisor y al Administrador.

– Fase 2: Mapeo/Medición

En el método COSMIC, utilizamos la ingeniería de software de nuestro proyecto para determinar cuáles son los procesos funcionales y movimientos de datos que lo componen. Posteriormente, asignamos un punto de función COSMIC por cada movimiento de datos identificado.

○ **Proceso funcional: Autenticación**

Movimientos de datos:

- ☞ Entrada: Iniciar sesión como administrador, revisor o profesor.
- ☞ Entrada: Ingresar como docente o revisor.
- ☞ Entrada: Ingresar como administrador, docente o revisor.

Por lo tanto los **Puntos de función COSMIC** son: 3 CFP.

○ **Proceso funcional: Gestión de Usuarios**

Movimientos de datos:

- ☞ Entrada: Crear, Leer, Actualizar y Eliminar datos de revisores y profesores.
- ☞ Lectura: Leer datos de revisores y profesores para su gestión.
- ☞ Escritura: Registrar cambios realizados en la información de revisores y profesores.
- ☞ Entrada: Seleccionar revisor y profesor para asignar mutuamente.
- ☞ Lectura: Leer datos de revisores y profesores para validar asignaciones existentes.
- ☞ Escritura: Registrar la asignación de revisores a profesores y viceversa.
- ☞ Entrada: Seleccionar parámetros de visualización de distribución.
- ☞ Lectura: Obtener datos de distribución según los parámetros seleccionados.

Por lo tanto los **Puntos de función COSMIC** son: 8 CFP

○ **Proceso funcional: Acceso para Docente/Revisor/Administrador**

Movimientos de datos:

- ☞ Entrada: Seleccionar tipo de acceso (docente, revisor o administrador).

Puntos de función COSMIC: 1 CFP.

○ **Proceso funcional: Revisor/Profesor**

Movimientos de datos:

- ☞ Entrada: Seleccionar revisor y profesor para asignar.

- ✎ Escritura: Registrar la asignación de revisores a profesores.
- ✎ Entrada: Seleccionar docente para acceder a su portafolio.
- ✎ Lectura: Leer datos del portafolio del docente seleccionado.

Por lo tanto los **Puntos de función COSMIC** son: 4 CFP.

De esta forma, hemos determinado que nuestro proyecto tiene una medición de:

16 puntos de función COSMIC .

9.1.1. Costo del equipo de trabajo de desarrollo de software

Se considera el número de personas, cual es la remuneración de cada rol, también tener en cuenta otros gastos del personal como lo son beneficios de fin de año, seguros, y también el costo administrativo de cada persona.

- Número de Personal: 11
- Roles: Gestor de Proyectos/Scrum Master, Analista de Requerimientos, Diseñador UX/UI, Diseñadores Frontend(2), Diseñadores Backend(3), Tester/QA, DBA(Administrador de Base de Datos), Despliegue/DevOps, Documentación y Soporte, Experto de Seguridad.

Para que esta medición sea exacta, debemos considerar que un punto de función está desarrollado solamente cuando está completamente listo e instalado en ambiente de producción.

Cuadro 2: Relación de Personas y sus Roles

Nombre	Roles	Nro. de personas por cada Rol
Willians Montañez Choque	Gestor de Proyectos/Scrum Master	1
Shaiel Almendra Arana Flores	Analista de Requerimientos	1
Julio Cesar Callañaupa Sallo	Diseñador UX/UI	1
Cristian Andy Cabrera Mejia, Johana Maria Mamani Meza	Diseñadores Frontend	2
Willians Montañez Choque, Angela Vanessa Reynaga Flores, Kevin Aron Sumire Ccahuana	Diseñadores Backend	3
Juan Gabriel Poma Supo	Tester/QA	1
Angela Vanessa Reynaga Flores	DBA(Administrador de Base de Datos)	1
Luis Alberto Gallegos Cjuiro	Despliegue/DevOps	1
Katerine Candy Lima Esperilla	Documentación y Soporte	1
Vito Jhon Huerta Medina	Experto de Seguridad	1

Roles	Tarifa por día (8 horas)	Horas estimadas	Días estimados	Remunera-ción	Remunera-ción Total
Gestor de Proyectos/Scrum Master	90.00	368 horas	52 días	90.00 * 52 días	4680.00
Analista de Requerimientos	80.00	45 horas	19 días	80.00 * 19 días	1520.00
Diseñador UX/UI	85.00	121 horas	16 días	85.00 * 16 días	1360.00
Diseñadores Frontend (cada uno)	85.00	138 horas	18 días	(85.00 * 18 días)	3060.00
Diseñadores Backend (cada uno)	85.00	230 horas	30 días	(85.00 * 30 días)	4590.00
Tester/QA	80.00	184 horas	24 días	80.00 * 24 días	1920.00
DBA (Administrador de Base de Datos)	80.00	138 horas	18 días	80.00 * 18 días	1440.00
Despliegue /DevOps	75.00	46 horas	6 días	75.00 * 6 días	450.00
Documentación y Soporte	70.00	138 horas	18 días	70.00 * 18 días	1260.00
Experto de Seguridad	70.00	92 horas	12 días	70.00 * 12 días	840.00
Costo Total del equipo trabajo					S/.21 120

Determinar el costo por unidad de medida

Para determinar cuánto cuesta desarrollar cada punto de función se utiliza la siguiente fórmula:

$$\text{Costo por punto de función} = \text{Costo del equipo de trabajo} / \text{puntos de función}$$

Para nuestro proyecto se consideró:

$$\text{Costo por punto de función} = S/. 21\ 120 / 8 \text{ puntos de función por mes}$$

$$\text{Costo por punto de función} = S/. 2\ 640$$

9.1.2. Costo del proyecto de software

Una vez que contamos con la medición del tamaño del software y el costo por unidad de medida, podemos determinar el costo del proyecto de software usando la siguiente formula:

$$\text{Costo de un proyecto de software} = \text{Tamaño del software} \times \text{Costo por punto de función}$$

Para nuestro proyecto se consideró:

$$\text{Costo de un proyecto de software} = 16 \text{ CFP} \times S/. 2\ 640$$

$$\text{Costo de un proyecto de software} = S/. 42\ 240$$

9.2. COCOMO II (Constructive Cost Model):

- ★ Una de las tareas cruciales en la gestión de proyectos de software es la estimación, que implica determinar, con un grado de certeza razonable, los recursos necesarios, tanto en hardware como software, así como los costos, el tiempo y el esfuerzo requerido para el desarrollo del proyecto. COCOMO II es un modelo de estimación que utiliza el tamaño del software y factores de costo y escala para prever el esfuerzo y tiempo en el desarrollo de un producto. se clasifica como un modelo algorítmico que establece relaciones matemáticas. COCOMO define tipos de desarrollo, como:

- Orgánico: Proyecto simples
- Semi-acoplado: Proyecto de complejidad intermedia, que es lo que usaremos para nuestro proyecto.
- Empotrado: Proyectos complejos con poca experiencia.

Y por otro lado existen diferentes modelos que define COCOMO:

Modelo básico: Se basa exclusivamente en el tamaño expresado en LDC.

Modelo intermedio: Además del programa incluye un conjunto de medidas subjetivas llamadas conductores de costes.

modelo avanzado: Incluye todo lo del tamaño del programa incluye un conjunto de medidas subjetivas llamadas conductores de costes.

Modelo Avanzado: Incluye todo lo del modelo intermedio además del impacto de cada conductor de costes en las distintas fases de desarrollo.

Para nuestro proyecto utilizaremos el modelo intermedio, dado que realiza las estimaciones con precisión. Formulas que se usara será las siguientes:

$$E = a \cdot \text{KLDC} \cdot E^e \cdot \text{FAE (persona} \cdot \text{mes)}$$

$$T = c \cdot E^d \text{ (meses)}$$

$$P = \frac{E}{T} \text{ (personas)}$$

Lenguaje	Kilo Lineas de Código (KLDC)
.json	85
.xml	32
.php	4409
.css	51
.blade.php	140
Total	5218

Cuadro 3: Numero de lineas de código de los programa utilizados.

9.2.1. Costos por Hardware

- **Servidores y Alojamiento de Local:** Se requerirá la adquisición de servidores para alojar la aplicación web y garantizar su disponibilidad, también se incluirán costos asociados con el espacio físico.
 - ◊ Adquisición de servidores: S/.1500.00
 - ◊ Espacio físico y climatización: S/.5000.00
- **Equipos de Desarrollo:** Estimación de costos asociados con la adquisición y mantenimiento de computadoras y dispositivos necesarios para el desarrollo del proyecto.
 - ◊ Adquisición de computadoras para el equipo de desarrollo: S/.10000.00
 - ◊ Mantenimiento y actualización de hardware: S/.500.00
- **Costo Total: S/.17000.00**

9.2.2. Costos por Personal

Se considera el tamaño del proyecto en líneas de código (LOC) como el factor principal para la estimación de costos.

Roles	Horas Estimadas	Días Estimados	Remuneración	Costo Total
Gestor de Proyectos/Scrum Master	368	52	S/.4128	S/.4128
Analista de Requerimientos	45	19	S/.9243	S/.9243
Diseñador UX/UI	121	16	S/.2353	S/.2353
Diseñadores Frontend (2)	276	18	S/.4858	S/.4858
Diseñadores Backend (3)	690	30	S/.11402	S/.11402
Tester/QA	184	24	S/.2070	S/.2070
DBA (Administrador de Base de Datos)	138	18	S/.1131	S/.1131
Despliegue/DevOps	46	6	S/.442	S/.442
Documentación y Soporte	138	18	S/.1485	S/.1485
Experto de Seguridad	92	12	S/.1200	S/.1200
Total	—	—	—	S/.36827.00

9.2.3. Costo en la elaboración del proyecto

Los costos de elaboración del proyecto se refieren principalmente a los gastos que realizaron a lo largo de las diferentes fases del proyecto. Estas las podemos ver expresadas en la siguiente tabla.

Detalle	Importe(S/.)
Material de escritorio	120
Internet	240
Total	260

9.2.4. Costo total del Software

El costo total del software se obtiene de la sumatorio de los diferentes costos vistos hasta el momento, costo de desarrollo, implementación y elaboración del proyecto. Esto lo vemos expresado en la siguiente tabla.

Detalle	Importe(S/.)
Costos de Hardware	17000.00
Costos de personal	36827
Costo en la elaboración del proyecto	260
Total	54087

10. Anexos

- Anexo A: Contenido de portafolio Docente de Practica



UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y
MECÁNICA

DEPARTAMENTO ACADÉMICO DE INGENIERÍA
INFORMÁTICA

PORTAFOLIO DOCENTE DE PRÁCTICA (JEFES DE PRÁCTICA)

CONTENIDO E INSTRUCTIVO
SEMESTRE 2023-1

1. Presentación del portafolio para los Jefes de Práctica que regentan los cursos de laboratorio

- 1.1. Carátula (FORMATO02: Caratula.doc)
- 1.2. Carga académica (lectiva y no lectiva) del semestre (FORMATO: PD-001)
- 1.3. Filosofía general del docente (FORMATO03: Filosofia docente.doc)
- 1.4. Currículo vitae de docente (FORMATO04: CV.doc)

2. Contenido por asignatura

Documentación general

- 2.1. Avance académico por - Formato Plan de sesiones de sesiones clase (FORMATO PD-012)
- 2.2. Registro de asistencia de estudiantes (FORMATO PD-004 ó formato simple o propio se sobreentiende que aquí esta incluida la relacion de estudiantes matriculados)

3. Actividad de enseñanza - Materiales de clase

- 3.1. Evidencia de actividades de enseñanza (FORMATO PD-014)

4. Evaluaciones

- 4.1. Registro de notas de prácticas (FORMATO PD-015). Para las asignaturas de laboratorio.

5. Proyectos individuales/grupales (si hubiera)

- Anexo B: Contenido de portafolio Docente (Contenido e Instructivo)



UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA

DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA

PORTAFOLIO DOCENTE CONTENIDO E INSTRUCTIVO

SEMESTRE 2023-1

1. Presentación de portafolio

- 1.1. Carátula (FORMATO02: Caratula.doc) **Necesario**
- 1.2. Carga académica (lectiva y no lectiva) del semestre (FORMATO: PD-001) **Necesario, lo requiere el DAI**
- 1.3. Filosofía general del docente (FORMATO03: Filosofía docente.doc) **Necesario, por Directiva N° 003-2018-VRAC-UNSAAC**
- 1.4. Currículo vitae de docente (FORMATO04: CV.doc) **Necesario para el criterio 6 de ICACIT (Cuerpo de Profesores)**

2. Contenido por asignatura

Documentación general

- 2.1. Silabo formato UNSAAC (FORMATO Centro de Cómputo) **Necesario**
- 2.2. Avance académico por sesiones - Formato Plan de sesiones de clase (FORMATO PD-012) **Necesario, lo requiere el DAI.**
- 2.3. Registro de entrega de silabo. (FORMATO PD-002 o Formato de CC) **Necesario. Lo requiere el DAI**
- 2.4. Registro de asistencia de estudiantes, Se entiende que en la asistencia Incluye relación de estudiantes matriculados, no es necesario adjuntarlo) (FORMATO CENTRO DE COMPUTO obligatorio desde el próximo semestre 2023-2 por un tema de estandarización del formato) **Necesario, por licenciamiento y Directiva N° 003-2018-VRAC-UNSAAC**

3. Actividad de enseñanza - Materiales de clase

- 3.1. Evidencia de actividades de enseñanza (FORMATO PD-014) **Necesario, por Directiva N° 003-2018-VRAC-UNSAAC**

4. Evaluaciones **Necesario.**

Artículo 46°, reglamento Académico

Evaluación de entrada


- 4.1. Enunciado y resolución de evaluación de entrada (Puede ser un solo archivo, es decir la prueba resuelta).
- 4.2. Informe de resultado de la evaluación de entrada (FORMATO Informe Resultado de la evaluación de entrada).

- 4.3. Enunciado y resolución de la evaluación de conocimiento parcial de la primera, segunda y tercera unidad (En caso de que el proyecto sea considerado como nota de la última parcial deberá ceñirse al esquema de proyectos ítem 5 Proyectos individuales/grupales)
- 4.4. Registro de asistentes a la resolución de examen de primera, segunda y tercera unidad (FORMATO PD-006 o Formato CC).
- 4.5. Registro de ingreso de notas en FORMATO de Centro de Cómputo. CONSOLIDADO FINAL

5. Proyectos individuales/grupales (si hubiera)

- 5.1. Rúbrica (FORMATO PD-007) u otro instrumento para la calificación del proyecto individual y grupal
 - 5.2. Documento de asignación de proyectos individuales y/o grupales (FORMATO PD-010)
 - 5.3. Informe sobre entrega final de proyectos, evaluación y resultados de calificación. (FORMATO FORMATO PD-011)
 - 5.4. Otras evaluaciones (Opcional)
 - Rúbrica (FORMATO PD-007) u otro instrumento para la calificación del proyecto individual y grupal
6. Cierre de portafolio (FORMATO: Informe Cierre de portafolio.doc) Necesario, lo requiere el DAI

- Anexo C: Formatos para portafolios docente

	A	B	C	D	E	F
1		0				
2		FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA				
3		DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA				
4						
5		FORMATOS PARA EL PORTAFOLIO DOCENTE				
6						
7	<p>Estimado colega:</p> <p>Los formatos establecidos en el presente archivo son de uso OBLIGATORIO en la documentación de su portafolio semestral. Las instrucciones, colores y observaciones tienen por finalidad estandarizar y facilitar la gestión de los portafolios, por los que se le solicita su cumplimiento escrupuloso.</p> <p>Los formatos son propiedad intelectual del Departamento Académico de Ingeniería Informática de la UNSAAC, por lo que se les solicita manejarlos con discreción, bajo responsabilidad.</p> <p>Cualquier duda u observación comunicarlo a la comisión.</p>					
8						
9	CODIGO	FORMATO	COLOR DE HOJA	OBS.		
10	PD-001	CARGA ACADEMICA DOCENTE	Blanco	Llenar e imprimir a color		
11	PD-002	EXPOSICIÓN Y ENTREGA DE SILABO	Amarillo	Pre-impreso		
12	PD-003	AVANCE ACADÉMICO POR SESIONES	Verde	Pre-impreso		
13	PD-004	REGISTRO DE ASISTENCIA A CLASES	Blanco	Pre-impreso		
14	PD-005	EVALUACIÓN DE ENTRADA	Rosado	Pre-impreso		
15	PD-006	RESOLUCION DE EXAMEN	Celeste	Pre-impreso		
16	PD-007	FORMATO DE RÚBRICA	Blanco	Llenar e imprimir a color		
17	PD-008	INFORME DE EVALUACIÓN DE ENTRADA	Blanco	Llenar e imprimir a color		
18	PD-009	INFORME DE EVALUACIÓN DE UNIDADES DIDÁCTICAS Y OTROS	Blanco	Llenar e imprimir a color		
19	PD-010	ASIGNACIÓN DE PROYECTOS GRUPALES	Blanco	Llenar e imprimir en blanco y negro		
20	PD-011	INFORME DE AVANCE DE PROYECTOS GRUPALES	Blanco	Llenar e imprimir en blanco y negro		
21	PD-012	PLAN DE SESION DE CLASE	Blanco	Llenar e imprimir en blanco y negro		
22	PD-013	INFORME DE LA NOTA (MAXIMA, PROMEDIO, MINIMA)	Blanco	Llenar e imprimir en blanco y negro		
23	PD-014	INFORME DE MANEJO DE MATERIALES DE CLASE	Blanco	Llenar e imprimir en blanco y negro		
24	PD-015	INFORME DE NOTAS DE LABORATORIO	Blanco	Llenar e imprimir en blanco y negro		

• Anexo D: Formatos para carga academica de docente

	A	B	C	D	E	F	G	H
1	UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO						FORMATO PD-001	
2	FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA							
3	DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA							
4								
5	CARGA ACADEMICA DOCENTE							
6	Semestre 2023-1							
7							FIRMA	
8	DOCENTE:	<<nombre del docente>>						
9								
10	HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES	SÁBADO	
11	07:00 - 08:00							
12	08:00 - 09:00							
13	09:00 - 10: 00							
14	10:00 - 11:00							
15	11:00 - 12:00							
16	12:00 - 13:00							
17	13:00 - 14:00							
18	14:00 - 15:00							
19	15:00 - 16:00							
20	16:00 - 17:00							
21	17:00 - 18:00							
22	18:00 - 19:00							
23	19:00 - 20:00							
24	20:00 - 21:00							
25	21:00 - 22:00							

- Anexo E: Formato para Exposición y Entrega de Silabo

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA
DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA

FORMATO
PD-002

FECHA

FIRMA

EXPOSICIÓN Y ENTREGA DE SILABO

DOCENTE:

ASIGNATURA:

TEMA:

	AULA	HORA
Exposición y entrega de silabo		

NRO	CODIGO	APELLIDOS Y NOMBRES	PC	FIRMA
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
21				
22				
23				
24				
25				
26				
27				
28				
29				
30				
31				
32				
33				
34				
35				
36				
37				

- Anexo F: Formato para Avance Académico por Sesiones

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA
DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA

FORMATO
PD-003

AVANCE ACADÉMICO POR SESIONES

DOCENTE:				FIRMA	
ASIGNATURA:					

NRO	FECHA	TEMA	OBSERVACION
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			

• Anexo G: Formato para rúbrica de niveles de logro

	A	B	C	D	E	F	G	H	I	J	K	L
1	UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO											FORMATO
2	FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA											PD-007
3	DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA											
4												SEMESTRE
5	RÚBRICA PARA											
6												FIRMA
7	DOCENTE											
8												CODIGO
9	ASIGNATURA											FECHA
10												
11												
12	Indicadores de desempeño		Niveles de logro								PUNTAJE	
13			DESTACADO (X ptos)	ESPERADO (X ptos)		PROCESO (X ptos)		INICIO (X ptos)				
14	1											
15	2,											
16	3											
17	4											

• Anexo H: Formato para informe de conocimientos

	A	B	C	D	E	F	G	H	I	J	K	L	
1	UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO											FORMATO PD-009	
2	FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA												
3	DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA												
4												SEMESTRE	
5	INFORME DE												
6												FIRMA	
7	DOCENTE												
8												CODIGO	FECHA
9	ASIGNATURA												
10													
11													
12	Nº de alumnos:			Matriculados:			Evaluados:						
13													
14													
15			Nivel Alcanzado en %										
16	Nro.	Conocimiento o habilidad	Destacado	%	Esperado	%	Proceso	%	Inicio	%	Total %		
17	1			0.00%		0.00%		0.00%		0.00%	0.00		
18	2			0.00%		0.00%		0.00%		0.00%	0.00		
19	3			0.00%		0.00%		0.00%		0.00%	0.00		
20	4			0.00%		0.00%		0.00%		0.00%	0.00		
21	5			0.00%		0.00%		0.00%		0.00%	0.00		
22													
23	Observaciones:												
24	1												
25	2												
26	3												
27	4												

- Anexo I: Formato para asignación de proyectos

	A	B	C	D	E	F
1	UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO				FORMATO PD-010	
2	FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA					
3	DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA					
4					FECHA	
5	ASIGNACIÓN DE PROYECTOS GRUPALES					
6					FIRMA	
7	DOCENTE:					
8	ASIGNATURA:			AULA	HORA	
9	TEMA:		Asignación de proyectos grupales			
10						
11	Descripción del proyecto			Fecha de presentación		
12						
13	Grupo Nro; 01					
14	NRO	CODIGO	APELLIDOS Y NOMBRES		FIRMA	
15	1					
16	2					
17	3					
18	4					
19	5					

- Anexo J: Formato para informe de Avances de Proyectos Grupales

	A	B	C	D	E	F
1	UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO				FORMATO PD-011	
2	FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA					
3	DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA					
4					FECHA	
5	INFORME DE AVANCE DE PROYECTOS GRUPALES					
6					FIRMA	
7	DOCENTE:					
8	ASIGNATURA:			AULA	HORA	
9	TEMA:	Avance de proyectos				
10						
11						
12						
13	Grupo Nro. 01					
14	Describe el nivel de avance del proyecto			Calificación		
15						
16						
17	Grupo Nro. 02					
18	Describe el nivel de avance del proyecto			Calificación		
19						

• Anexo K: Formato para Plan de sesiones en clase

	A	B	C	D	E	F	G	H
1								
2	UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO							
3	DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA							
4	NOMBRE DE LA ASIGNATURA:							
5	PLAN DE SESIONES POR CLASE - 201-1							
6	DOCENTE:							
7	Semana	Sesión	TEMA	EJERCICIOS A RESOLVER	EJERCICIOS PROPUESTOS	Fecha	Aula	Hora
8	PRIMER PERIODO PARCIAL							
9	1	01						
10		02						
11		03						
12	2	04						
13		05						
14		06						
15	3	07						
16		08						
17		09						
18	4	10						
19		11						
20		12						
21	5	13						
22		14						
23		15						
24	6	16						
25		17						
26		18						

- Anexo L: Formato para Informe de la Nota (Máxima, Promedio, Mínima)

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO							FORMATO PD-013
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA							
DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA							
							SEMESTRE
INFORME DE LA NOTA (MAXIMA, PROMEDIO, MINIMA)							
DOCENTE							FIRMA
		CODIGO					FECHA
ASIGNATURA							
EVALUACION							
	Nro.	Resumen de Evaluación	Nivel Alcanzado en %				
			CANTIDAD	%			
	1	NRO DE ALUMNOS MATRICULADOS					
	2	NRO DE ALUMNOS EVALUADOS					
	3	NRO DE ALUMNOS APROBADOS					
	4	NRO DE ALUMNOS DESAPROBADOS					
	5	NRO DE ALUMNOS NO SE PRESENTO					
	Nro.	Resumen	ESTUDIANTE				
			CODIGO	APELLIDOS Y NOMBRES		NOTA	
	1	NOTA MAXIMA					
	2	NOTA PROMEDIO					
	3	NOTA MINIMA					
Observaciones:							
	1						
	2						
	3						
	4						

• Anexo M: Formato para Informe de manejo de materiales de clase

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO			FORMATO PD-014
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA			
DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA			
			SEMESTRE
INFORME DE MANEJO DE MATERIALES DE CLASE			
	DOCENTE		FIRMA
			CODIGO
	ASIGNATURA		
	LINK		
	Nro.	TIPO DE MATERIAL(*)	
	1		
	2		
	3		
	4		
	5		
	6		
	7		
	8		
	9		
	10		

- Anexo N: Formato para Informe de notas de laboratorio

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO										FORMATO PD-015
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA										
DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA										
										SEMESTRE
INFORME DE NOTAS DE LABORATORIO										
										FIRMA
DOCENTE										
										CODIGO
ASIGNATURA										FECHA
EVALUACION										
NRO	CODIGO	APELLIDOS Y NOMBRE	NOTA DE LAS GUIAS						PROMEDIO	
			01	02	03	04	05	06		
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										
23										
24										
25										
26										
27										
28										
29										
30										

- Anexo O: Formato para Informe de cierre de portafolio



UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA
DEPARTAMENTO ACADEMICO DE INGENIERÍA INFORMÁTICA

INFORME DE CIERRE DEL PORTAFOLIO

PARA: DR. DENNIS IVAN CANDIA OVIEDO
DIRECTOR DE DEPARTAMENTO ACADEMICO DE INGENIERIA INFORMATICA
DE: Gabriela Zuñiga Rojas
DOCENTE DE LA ASIGNATURA DE METODOLOGIAS DE DESARROLLO DE
SOFTWARE
FECHA: 22 de Agosto de 2023

Mediante la presente informo a su despacho sobre el cierre de la asignatura de METODOLOGIAS DE DESARROLLO DE SOFTWARE, impartida durante el semestre académico 2023-I

1. Cuadro resumen

INDICADOR	CANTIDAD	PORCENTAJE
Matriculados	14	100
Aprobados	12	85
Retirados	1	7
Desaprobados	1	7
Promedio de alumnos asistentes a sesiones de clase	12	85

2. Dificultades durante el semestre académico

1. Muchos feriados y tomas de local
- 2.
- 3.

3. Conclusiones:

- 1.
- 2.
- 3.
- 4.

4. Recomendaciones

1. Se requiere mas tiempo para realizar un mayor y mejor avance
- 2.
- 3.

• Anexo P: Formato para Revisión de portafolio 2023-1

	A	B	C	D	E	F	G	H	I	J	K	M	N	O	P	Q	R	S	T	U	V	W	X	Y	AC	A	
1	Calificación(No cumplió:0, Incompleto: 1 , Completo:2)						REVISIÓN DE PORTAFOLIO 2023-1																				
2	NRO	DOCENTE	CÓDIGO	CURSO	CRED	TIPO	1. PRESENTACIÓN DE PORTAFOLIO				2. CONTENIDO POR ASIGNATURA				3. ACTIV. DE	4. EVALUACIONES											
3							1.1 CARATULA (F-02)	1.2 CARGA ACADÉMICA (F-PD-001)	1.3 FICHA DEL DOCENTE (F-03)	1.4 CURRÍCULO VITAL (F-04)	2.1 SILABO (F-CC)	2.2 AVANCE ACADÉMICO (F-PD-012)	2.3 REG. ENTREGA DE SILABO (F-PD-002)	2.4 REG. DE ASISTENCIA (F-CC)		3.1 EVIDENCIA DE ACT. DE ENSEÑANZA (F-PD-014)	4.1 ENUN. Y RESOL. DE EVALUACIÓN DE ENTRADA (F-10)	4.2 INF. DE EVALUACIÓN DE ENTRADA (F-10)	4.3 ENUNCIADO Y RESOLUCIÓN DE				4.4 REG. DE ASISTENCIA A LA			4.5 REGIS. CONSOLIDADO	4.6 PROMEDIO
4																			PRIMERA EVALUACIÓN	SEGUNDA EVALUACIÓN	TERCERA EVALUACIÓN	SUSTITUTORIO	PRIMERA EVALUACIÓN	SEGUNDA EVALUACIÓN	TERCERA EVALUACIÓN		
5																											
6	1	LAURO ENCISO RODAS	IF650AIN	MODELOS PROBABILISTICOS	4	T / P	2	2	2	2	2	2	2	2	2	2	2	0	2	2			2	2		2	
7			IF456BIN	ALGORITMOS AVANZADOS	4	T / P	2	2	2	2	2	2	2	2	2	2	2	0	2	2			2	2		2	
8	2	JULIO CESAR CARBAJAL LUNA	IF653AIN	MINERIA DE DATOS	4	T / P	2	2	2	2	2	2	2	1	0	0	0	1	1			0	0		2		
9			IF664AIN	BIOINFORMATICA	4	T / P	2	2	2	2	2	2	2	1	0	0	0	1	1			0	0		2		
10	3	NILA ZONIA ACURIO USCA	IF450AIN	ABSTRACCION DE DATOS Y OBJETOS	4	T / P	2	2	2	2	2	2	2	2	2	2	2	2	2			2	2		2		
11			IF468AIN	FUNDAMENTOS DE LA PROGRAMACION	4	T / P	2	2	2	2	2	2	2	2	2	2	1	2	2	2			2	2		2	
12	4	JAVIER ARTURO ROZAS HUACHO	IF651AIN	INTELIGENCIA ARTIFICIAL	4	T / P	2	2	2	2	2	2	2	0	2	2	2	2	2			2	0		2		
13			IF466AIN	COMPILADORES	3	T / P	2	2	2	2	2	1	2	1	2	1	2	2			2			2			
14	5	LINO PRISCILIANO FLORES PACHECO	IF710AIN	INVESTIGACION I	3	T / P	2	2	2	2	2	1	0	2	2	0	0	0			0			0			
15			IF711AIN	INVESTIGACION II	3	T / P	2	2	2	2	2	1	2	2	2	0	0	0			0			0			
16	6	EDWIN CARRASCO POBLETE		ORGANIZACION Y ARQUITECTURA DEL COMPUTADOR	4	T / P																					
17			IF550AIN	COMPUTADOR			2	2	2	2	2	2	2	2	2	2	1	1	2	2			2	2			
18	7	EMILIO PALOMINO OLIVERA		FORMULACION DE PROYECTOS DE TECNOLOGIAS DE LA INFORMACION	3	T / P																					
19			IF483AIN	INFORMACION			2	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0				
20			IF553AIN	LENGUAJE ENSAMBLADOR	3	T / P	2	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0				
21			IF902AAE	TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACION	3	T																					
22	8	ENRIQUE GAMARRA SALDIVAR	IF451BIN	PROGRAMACION DIGITAL I	2	P	1	1	1	1	0	0	0	0	0	0	0	0	0			0	0				
23			IF212AEI	ANALISIS NUMERICO	3	T / P	1	1	1	1	0	0	0	0	0	0	0	0	0			0	0				
24			IF457AIN	METODOS NUMERICOS	3	T / P	1	1	1	1	0	0	0	0	0	0	0	0	0			0	0				
25	9	DENNIS IVÁN CANDIA OVIEDO		ADMINISTRACION DE TECNOLOGÍAS DE LA INFORMACIÓN	3	T / P	2	2	2	2	2	2	2	2	2	2	2	2	2			2	2				
26			IF610AIN	ANÁLISIS Y DISEÑO DE SISTEMAS DE INFORMACION	4	T / P	2	2	2	2	2	2	2	2	2	2	2	2	2			2	2				
27				TECNOLOGIAS DE LA INFORMACION Y LA	3	P	2	2	2	2	2	2	2	2	2	2	2	2	2			2	2				

• Anexo Q: Formato para Estado Final de los Portafolios

ESTADO FINAL DE LOS PORTAFOLIOS DOCENTE 2023-1 DEL DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA				
No.	NOMBRES	CAT	ESTADO	REVISOR
	DOCENTES NOMBRADOS			
1	DR. LAURO ENCISO RODAS	PR-DE	COMPLETO	KARELIA
2	MGT. JULIO CÉSAR CARBAJAL LUNA	PR-DE	COMPLETO	
3	MGT. NILA ZONIA ACURIO USCA	PR-DE	COMPLETO	
4	MGT. JAVIER ARTURO ROZAS HUACHO	PR-DE	COMPLETO	
5	MGT. LINO PRISCILIANO FLORES PACHECO	PR-TC	COMPLETO	LAURO
6	MGT. EDWIN CARRASCO POBLETE	PR-TC	COMPLETO	
7	DR. EMILIO PALOMINO OLIVERA	PR-TC	COMPLETO	
9	DR. DENNIS IVÁN CANDIA OVIEDO	PR-TC	COMPLETO	
10	DR. RONY VILLAFUERTE SERNA	PR-TC	NO CUMPLIO	WILLIAN
11	ING. GUZMÁN TICONA PARI	AS-DE	COMPLETO	
12	MGT. YESHICA ISELA ORMEÑO AYALA	AS-DE	NO CUMPLIO	
13	ING. IVÁN CÉSAR MEDRANO VALENCIA	AS-TC	COMPLETO	
14	MGT. LUIS BELTRÁN PALMA TITO	AS-TC	COMPLETO	LINO AQUILES
15	DR. ROBERT WILBERT ALZAMORA PAREDES	AS-TC	COMPLETO	
16	MGT. WALDO ELIO IBARRA ZAMBRANO	AS-TC	NO CUMPLIO	
17	MGT. KARELIA MEDINA MIRANDA	AS-TC	COMPLETO	
18	MGT. JAVIER DAVID CHÁVEZ CENTENO	AS-DE	COMPLETO	TANY
20	MGT. VANESSA MARIBEL CHOQUE SOTO	AS-TC	NO CUMPLIO	
21	ING. MANUEL AURELIO PEÑALOZA FIGUEROA	AS-TP	COMPLETO	
22	LIC. JOSÉ MAURO PILLCO QUISPE	AU-TC	COMPLETO	
23	ING. LINO AQUILES BACA CÁRDENAS	AU-TC	COMPLETO	ESTHER
24	LIC. ESTHER PACHECO VÁSQUEZ	AU-TC	COMPLETO	
25	MGT. WILLIAN ZAMALLOA PARO	AU-TC	COMPLETO	
26	MGT. HARLEY VERA OLIVERA	AU-TC	COMPLETO	
27	MGT. MARITZA KATHERINE IRPANOCA CUSIMA	AU-TC	COMPLETO	ELIDA
28	MGT. EFRAINA GLADYS CUTIPA ARAPA	AU-TC	COMPLETO	
29	DR. DARIO FRANCISCO DUEÑAS BUSTINZA	AU-TP10	COMPLETO	
30	MGT. DORIS SABINA AGUIRRE CARBAJAL	AU-TP10	COMPLETO	
31	MGT. TANY VILLALBA VILLALBA	AU-TP10	COMPLETO	CARLOS FERNANDO
	DOCENTES CONTRATADOS 2023			
32	MGT. CARLOS FERNANDO MONTOYA CUBAS	A1	COMPLETO	
33	MGT. CARLOS RAMÓN QUISPE ONOFRE	B1	COMPLETO	
34	MGT. VICTOR DARIO SOSA JAUREGUI	B1	COMPLETO	LUIS ALVARO
35	MGT. BORIS CHULLO LLAVE	B1	COMPLETO	
36	MGT. HÉCTOR UGARTE ROJAS	B1	COMPLETO	
37	MGT. LUIS ALVARO MONZON CONDORI	B1	COMPLETO	
38	MGT. RAY DUEÑAS JIMENEZ	B2	COMPLETO	LISHA
39	MGT. JISBAJ GAMARRA SALAS	B2	COMPLETO	
40	MGT. LISETH URPY SEGUNDO CARPIO	B2	COMPLETO	
41	MGT. JULIO VLADIMIR QUISPE SOTA	B3	NO CUMPLIO	
42	MGT. RAUL HUILLCA HUALLPARIMACHI	B3	COMPLETO	CARLOS RAMON
43	ING. VANESA LAVILLA ALVAREZ	JP20	COMPLETO	
	INVITADOS 2023-I			
44	MGT. ELIDA FALCON HUALLPA	B1	COMPLETO	
45	ING. STEPHAN JHOEL COSIO LOAIZA	B1	COMPLETO	RAY
46	MGT. MARÍA DEL PILAR VENEGAS VERGARA	B2	COMPLETO	
47	MGT. HENRY SAMUEL DUEÑAS DE LA CRUZ	B2	COMPLETO	
48	MGT. GERAR FRANCIS QUISPE TORRES	B3	COMPLETO	
49	ING. RAIMAR ABARCA MORA	B3	COMPLETO	RAY
50	ING. GABRIELA ZUÑIGA ROJAS	B3	COMPLETO	
51	ING. VITTALI QUISPE SURCO	B3	COMPLETO	
52	ING. AGUEDO HUAMANI HUAYHUA	B3	COMPLETO	
	INVITADOS JP 2023-I			
53	ING. LISHA SABAH DIAZ CACERES	JPTC	COMPLETO	RAY
54	ING. ALFREDO ROZAS DAVILA	JPTC	COMPLETO	
55	ING. MARCIO FERNANDO MERMA QUISPE	JP20	COMPLETO	
56	ING. OLMER CLAUDIO VILLENA LEÓN	JP10	COMPLETO	
57	ING. LUZ INDIRA TICONA FÉLIX	JP10	COMPLETO	RAY
58	ING. SHIRLEY RUTH VELAZQUE ROJAS	JP10	COMPLETO	
59	ING. JOSE GUILLERMO ANDRADE CARI	JP10	COMPLETO	

• Anexo R: Formato para Horario Docente

▲	A	B	C	D	E	F	G	H
1	AULA	Hora Inicio	Hora Fin	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
2	LAB309	7	8			<1> IN:FORMULACION DE PROYECTOS DE TECNOLOGIAS DE LA INFORMACION,	<1> CI:METODOS NUMERICOS,	
3		8	9			<1> IN:FORMULACION DE PROYECTOS DE TECNOLOGIAS DE LA INFORMACION,	<1> CI:METODOS NUMERICOS,	
4		9	10		<1> AN:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> ZO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> CI:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> CO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,
5		10	11		<1> AN:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> ZO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> CI:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> CO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,
6		11	12		<1> IN:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> CI:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> DR:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	
7		12	13		<1> IN:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> CI:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> DR:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	
8		13	14					
9		14	15				<1> AT:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	
10		15	16				<1> AT:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	
11		16	17		<1> MI:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> AO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> AO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	
12		17	18		<1> MI:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> AO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	<1> AO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	
13		18	19	<1> EO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,			<1> QM:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	
14		19	20	<1> EO:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,			<1> QM:TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN,	
15		20	21					

• Anexo S: Formato para Carga Académica 2023-2 Definitiva por docentes

	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1				UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO											
2				FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y MECÁNICA											
3				DEPARTAMENTO ACADÉMICO DE INGENIERÍA INFORMÁTICA											
4				CARGA ACADÉMICA DEFINITIVA 2023-2 POR DOCENTES											
5															
6		PR-DE	CARRERA	LAURO ENCISO RODAS	CRED.	TIPO	GPO	HT	HP	DÍA	HR-INICIO	HR-FIN	AULA	AFORO	MATRIC.
7	44	IF456AIN	INGENIERIA INFORMATICA	ALGORITMOS AVANZADOS	4	T	A	2	0	JUEVES	18	20	IN108	50	33
8	45	IF456AIN	INGENIERIA INFORMATICA	ALGORITMOS AVANZADOS	4	T	A	1	0	VIERNES	8	9	IN108	50	33
9	46	IF456AIN	INGENIERIA INFORMATICA	ALGORITMOS AVANZADOS	4	P	A	0	2	MARTES	18	20	LAB301	24	33
10				SUBTOTAL: 3 2							TOTAL HRS: 5				
11															
12		PR-DE	CARRERA	JULIO CÉSAR CARBAJAL LUNA	CRED.	TIPO	GPO	HT	HP	DÍA	HR-INICIO	HR-FIN	AULA	AFORO	MATRIC.
13	62	IF467AIN	INGENIERIA INFORMATICA	ANALISIS Y DISEÑO DE ALGORITMOS	3	T	A	2	0	MIÉRCOLES	11	13	IN201	50	21
14	63	IF467AIN	INGENIERIA INFORMATICA	ANALISIS Y DISEÑO DE ALGORITMOS	3	P	A	0	2	LUNES	11	13	LAB301	24	21
15	169	IF664AIN	INGENIERIA INFORMATICA	BIOINFORMATICA	4	T	A	2	0	MARTES	9	11	IN203	50	11
16	170	IF664AIN	INGENIERIA INFORMATICA	BIOINFORMATICA	4	T	A	1	0	VIERNES	9	10	LAB306	50	11
17	171	IF664AIN	INGENIERIA INFORMATICA	BIOINFORMATICA	4	P	A	0	2	JUEVES	9	11	LAB306	24	11
18	204	IF902AZO	ZOOTECNIA	TECNOLOGIAS DE LA INFORMACION Y LA COMUNICACIÓN	3	P	A	0	2	MIÉRCOLES	9	11	LAB309	24	22
19				SUBTOTAL: 5 6							TOTAL HRS: 11				
20															
21		PR-DE	CARRERA	JAVIER ARTURO ROZAS HUACHO	CRED.	TIPO	GPO	HT	HP	DÍA	HR-INICIO	HR-FIN	AULA	AFORO	MATRIC.
22	26	IF452AIN	INGENIERIA INFORMATICA	ALGORITMOS Y ESTRUCTURAS DE DATOS	4	T	A	2	0	MARTES	9	11	IN107	50	30
23	27	IF452AIN	INGENIERIA INFORMATICA	ALGORITMOS Y ESTRUCTURAS DE DATOS	4	T	A	1	0	VIERNES	9	10	IN107	50	30
24	28	IF452AIN	INGENIERIA INFORMATICA	ALGORITMOS Y ESTRUCTURAS DE DATOS	4	P	A	0	2	JUEVES	9	11	IN107	50	30
25	142	IF651AIN	INGENIERIA INFORMATICA	INTELIGENCIA ARTIFICIAL	4	T	A	2	0	MIÉRCOLES	9	11	IN202	50	22
26	143	IF651AIN	INGENIERIA INFORMATICA	INTELIGENCIA ARTIFICIAL	4	T	A	1	0	VIERNES	10	11	IN202	50	22
27	144	IF651AIN	INGENIERIA INFORMATICA	INTELIGENCIA ARTIFICIAL	4	P	A	0	2	LUNES	9	11	LAB301	24	22

- Anexo T: Contrato de Desarrollo de Software



UNIVERSIDAD NACIONAL SAN ANTONIO ABAD DEL CUSCO
ESCUELA DE ING. INFORMATICA Y DE SISTEMAS



CONTRATO DE DESARROLLO DE SOFTWARE

ENTRE

NOMBRE DE LA ORGANIZACION

Y

.....



UNIVERSIDAD NACIONAL SAN ANTONIO ABAD DEL CUSCO
ESCUELA DE ING. INFORMATICA Y DE SISTEMAS



**CONTRATO DE DESARROLLO DE SOFTWARE ENTRE LA NOMBRE DE LA
ORGANIZACION Y**

Cusco,dedel

COMPARECEN

De una parte, la organización denominada con RUC, debidamente representada por en calidad de Lead Developer, en virtud de las atribuciones conferidas por Resolución Rectoral 00278/2021, de fecha 12 de diciembre de 2023.

Y, de otra parte, el integrante de la respectiva organización con D.N.I., con domicilio en constituida ante Notario de, el día, quien actúa en calidad de, con poder conferido mediante escritura pública ante el Notario de Cusco, D., en fecha 13/12/2023, poder que se exhibe en este acto y se manifiesta como vigente.

MANIFIESTAN

I.- Que la organización es una entidad de desarrollo de software que tiene como objetivo

II.- Que la organización “.....” de la UNSAAC tiene experiencia en el desarrollo de software.

III.- Que ambas partes están interesadas en suscribir el presente contrato, con el objeto de que la cree una aplicación informática/software a medida de las necesidades requeridas.

Por lo que, reconociéndose mutuamente la capacidad y representación con que actúan, ambas partes otorgan el presente CONTRATO DE DESARROLLO DE SOFTWARE que se regirá por las siguientes.



UNIVERSIDAD NACIONAL SAN ANTONIO ABAD DEL CUSCO
ESCUELA DE ING. INFORMATICA Y DE SISTEMAS



CLÁUSULAS

PRIMERA.- Objeto del contrato

El presente contrato tiene por objeto el desarrollo, por parte de la, del programa de ordenador descrito en el Anexo del contrato.

SEGUNDA.- Fases de desarrollo

El desarrollo del programa objeto del presente contrato, cuenta con las siguientes fases:

- a) **Fase inicial**, que comprende las etapas de análisis y diseño del software.

En la fase de análisis.....

La fase de diseño comprende el diseño de las interfaces de usuario.....

Esta fase se iniciará a la firma del presente contrato y tendrá una duración de.....días.

- b) **Fase de pruebas**, que consistirá en la realización por parte de la UMH de todas aquellas actividades necesarias para la comprobación del buen funcionamiento del programa desarrollado.

La fase de pruebas tendrá una duración máxima de

- c) **Fase de modificaciones**: que podrá darse en caso de que, como consecuencia de la fase descrita en el apartado anterior, surja la necesidad de llevar a cabo una serie de modificaciones en el programa desarrollado.

La fase de modificaciones, siempre que sea necesaria, comenzará una vez terminada la etapa de pruebas, y tendrá una duración máxima dedías.

- d) **Fase de entrega e instalación definitiva del programa**.

En este momento la Nombre de la organización hará entrega de toda la documentación del desarrollo y uso del programa informático, incluido el manual de operaciones.

Cada una de las Fases queda desarrollada en el Anexo al presente contrato.

TERCERA.- Duración

La duración del presente contrato será de 60 días contados a partir de la fecha de la firma del mismo.

CUARTA.- Lugar de prestación del servicio.

El desarrollo del proyecto, objeto del presente contrato se efectuará en las instalaciones de la Universidad Nacional San Antonio Abad del Cusco.

En el caso de que el equipo de trabajo requiera las reuniones virtuales, ésta se compromete a facilitar el acceso puntual a las salas virtuales, suministrándole cuanta información relacionada con el desarrollo del programa en cuestión le sea requerida.

QUINTA.- Responsable del trabajo

El responsable del desarrollo del trabajo por parte de la será el adscrito en la Universidad Nacional San Antonio Abad del Cusco quien tendrá como interlocutor válido por parte de la Organización a Mg. William Zamalloa Paro.



UNIVERSIDAD NACIONAL SAN ANTONIO ABAD DEL CUSCO
ESCUELA DE ING. INFORMATICA Y DE SISTEMAS



SEXTA.- Confidencialidad

Cada una de las partes se compromete a no difundir, bajo ningún aspecto, la información científica o técnica perteneciente a la otra parte a la que haya podido tener acceso en el desarrollo del presente contrato.

Los datos e informes obtenidos durante la realización del presente contrato, así como los resultados finales, tendrán carácter confidencial.

Ambas partes se comprometen a que todo el personal participante en el desarrollo del contrato conozca y observe el compromiso de confidencialidad regulado por esta cláusula.

Las disposiciones de esta cláusula referentes a confidencialidad subsistirán durante el semestre académico 2023-II después de la terminación del presente contrato.

SEPTIMO. - Responsabilidades

El presente Integrante será responsable de desarrollar y entregar las respectivas tareas en su debido momento. Se compromete a dedicar un mínimo de 8 horas semanales al proyecto, salvo acuerdos específicos escritos entre las partes y a contribuir con sus habilidades y esfuerzos al desarrollo del software.

NOVENO. - Abandono del Proyecto

En caso de que el Integrante de la organización decida abandonar el proyecto, deberá notificar al Lead Developer con al menos 30 días de antelación por escrito, explicando las razones de su decisión.

El Lead Developer se reserva el derecho de rescindir este contrato en caso de incumplimiento sustancial por parte del Desarrollador.

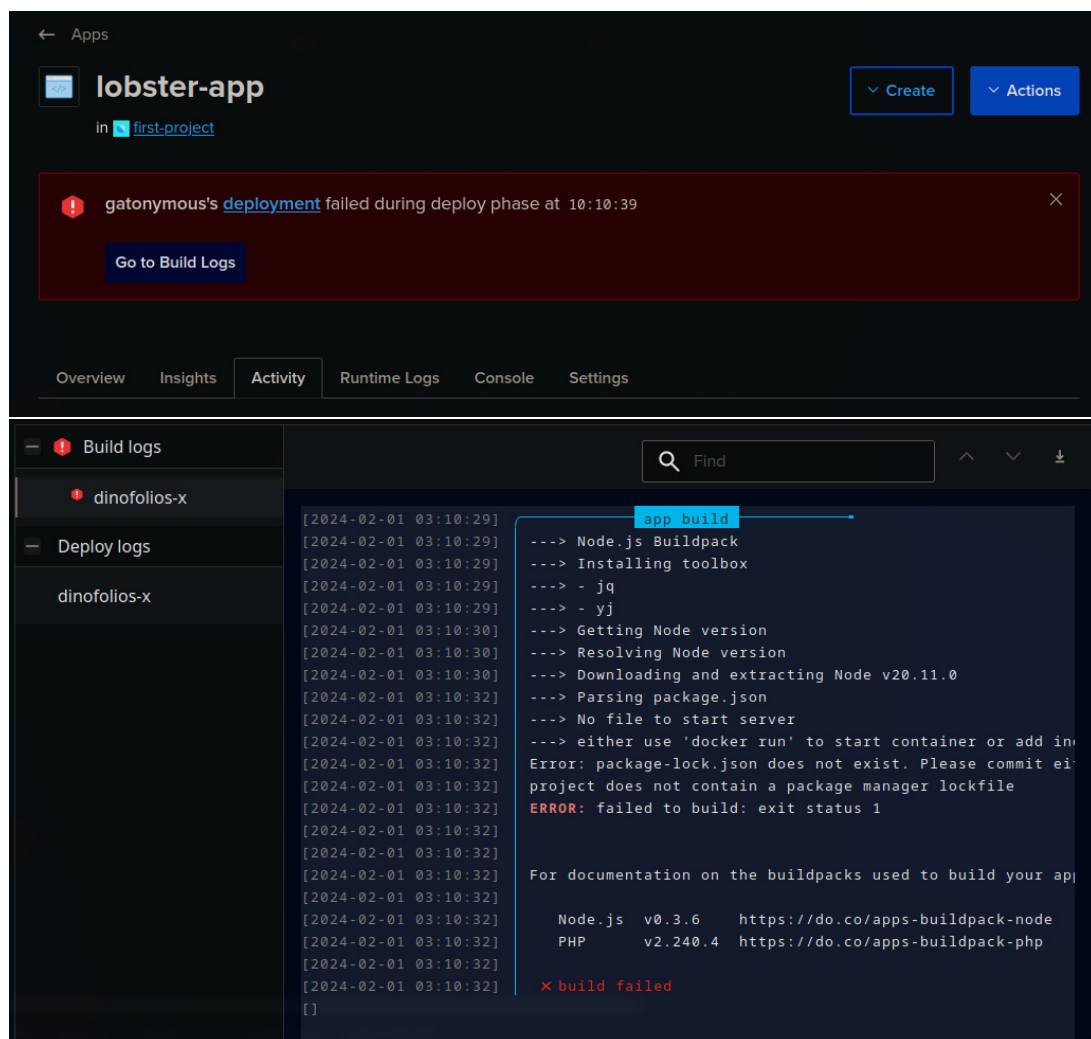
Si el Integrante no cumple con las expectativas de rendimiento, la respectiva Organización se reserva el derecho de retirar al respectivo Integrante del proyecto mediante notificación escrita con 7 días de antelación.

POR LA NOMBRE DE LA
ORGANIZACIÓN

.....
 INTEGRANTE DE LA ORGANIZACIÓN
 NOMBRE

.....
 RESPONSABLE: LEAD DEVELOPER

- Anexo U: Prueba de despliegue



- Anexo V: Manual de usuario
Manual de usuario

11. Referencias

Referencias

- [1] Sommerville. Ian, *Ingenieria de Software*, Addison Wesley, 9nd edition, 2011.
- [2] Hanmer Robert, *Pattern-Oriented-Software-Architecture*, Wiley, England, 1994.
- [3] Villalobos. Avarca *PROYECTOS DE DESARROLLO DE SOFTWARE*, Universidad de Tarapacá, Arica, Chile, 2019.
- [4] Aguirre. Jean, Sanchez. Karen, *L^AT_EX: Implementación de un sistema de portafolio digital para optimizar el proceso de evaluación de los docentes de la Universidad de Ciencias y Humanidades, Los Olivos*, Lima, 2019
- [5] Gomez. Adriana, Lopez. Maria, Migani. Silvana, otazú. Alejandra. : *UN MODELO DE ESTIMACION DE PROYECTOS DE SOFTWARE-COCOMO II*, 2018.
- [6] Cardona. Blanca, *Diseño e Implementacion de una Aplicación Web para una Escuela Profesional*, Universidad Politecnica de Madrid, Madrid España, 2022.
- [7] *PMOinformatica.com*, <https://www.pmoinformatica.com/2018/03/ejemplos-de-estimacion-de-costos-de-un-proyecto-de-software-COSMIC.html> (Accedido el 31 de enero de 2024).