**Plan de Pruebas**

**CAPSTONE**

**“Aula Virtual”**

***Fecha:[13/12/2024]***

**Tabla de contenido**

Histórico de Revisiones

| Versión | Fecha | Descripción/cambio | autor |
| --- | --- | --- | --- |
| 1.1 | 28-11-2024 | Inicio documento | Felipe Díaz |
| 1.2 | 03-12-2024 |  | Felipe Díaz |
|  |  |  |  |

Información del Proyecto

| Organización | Duoc UC. Escuela de Informática y Telecomunicaciones |
| --- | --- |
| Sección | 003 D |
| Proyecto (Nombre) | Aula Virtual |
| Fecha de Inicio | 16 agosto 2024 |
| Fecha de Término | 13 diciembre 2024 |
| Caso N° |  |
| Patrocinador principal | Duoc UC |
| Docente | Francisco Javier Juillet |

Integrantes

| Rut | Nombre | Correo |
| --- | --- | --- |
| **20.001.148-1** | **Felipe Daniel Díaz Contreras** | **felipedanield@gmail.com** |
| **21.082.200-3** | **Benjamin Enrique Antileo Rojas** | **be.antileo@duocuc.cl** |
| **21.138.978-8** | **Lucas Andres Rojas Pailahueque** | **lu.rojasp@duocuc.cl** |
| **20.848.102-9** | **Martin Vicente Rubio Zuñiga** | **mar.rubio@duocuc.cl** |
|  |  |  |

| Propósito del plan de pruebas  *Propósito, objetivo, visión que se espera de este plan de pruebas.* |
| --- |
| El propósito del plan de pruebas es garantizar tanto la calidad , funcionalidad, usabilidad y rendimiento de la página web creada en Django-python.  Con esto dicho los objetivo que se buscan son los siguientes:   * Validar las funcionalidades de la página web como lo son el login, CRUDs a los distintos mantenedores, carga de datos, etc. estén funcionando sin ninguna complicación al momento de realizarse. * Detectar problemas críticos que puedan afectar la experiencia del usuario. * Pruebas de URLS y enrutamiento para ver que no existan   Y al respecto de la visión que esperamos, es obtener una aplicación que cumpla con los requisitos definidos anteriormente en otros documentos, además de tratar de reducir o eliminar la gran mayoría de los problemas que se encuentren durante la realización de este. Haciendo esto nos permitirá que la aplicación sea recibido de manera positiva haciendo que la experiencia sea satisfactoria para los usuarios finales |

| Alcance de las pruebas  *Definición de requisitos de S.W., módulos de Software a probar, Requisitos ambiente de pruebas y Documentación Referenciada, etc.* |
| --- |
| **Definición de los Requisitos del Software (S.W.)**:   * Se probaran los siguientes funcionalidades principales:   + Funcionalidades principales de la aplicación (inicio de sesión, registro de usuarios, interacción entre profesores y estudiantes).   + Seguridad (protección de datos sensibles como contraseñas).   + Enrutamiento de páginas protegidas (Los usuarios estudiantes no deben entrar a apartados de profesor y viceversa)   **Módulos de Software a Probar**:   * Los siguientes módulos estarán a prueba   + Módulo de autenticación (login y cambio de contraseñas).   + Módulo de gestión académica (calificaciones, asignaciones, y calendario).   + Módulo de reportes o visualización de datos.   **Requisitos del Ambiente de Pruebas**:  Entorno donde se realizarán las pruebas, detallando hardware, software y configuraciones necesarias:   * + Hardware: Se necesita una computadora de escritorio o laptop con los siguientes especificaciones o similares:     - CPU: Integrada o mayores     - RAM: 8 GB de ram     - Almacenamiento: Disco duro 240 GB   + Software: Se necesitaran los siguientes componentes para una prueba exitosa: Sistemas operativos compatibles, navegadores web soportados.     - Python Instalado version 3+     - Microsoft C++ Build Tools     - Git     - Cualquier de los siguientes navegadores: Edge, Chrome, Opera     - PostgreSQL   **Documentación Referenciada**:   * Lugar donde se encuentran los documentos para verificar la existencia de realizar los casos de prueba   + Casos de usos : [Casos uso](https://drive.google.com/drive/u/1/folders/1s-XWWElX_9-ZNj3plTx_jBfK4DjmNMtw)   + ERS : [Informe ERS.docx](https://docs.google.com/document/d/1F0YKLbmNQX13XmdFFx6_KsdLW4nRbwe9/edit) |

| Definición de roles y responsabilidades  *Roles y responsabilidades de todos los participantes en el* ***proceso de pruebas*** *de SW*. | | |
| --- | --- | --- |
| Rol | Responsabilidades | Relevancia |
| Felipe Diaz | Encargado de realizar las pruebas | Importante |
| Martin Rubio | Ayudante para la realización de pruebas | Importante |
|  |  |  |
|  |  |  |
|  |  |  |

| Tipos de pruebas a realizar  *Definir el tipo de pruebas que se debe desarrollar para este proyecto, actividades y responsables.* |
| --- |
| **Tipos de Pruebas a Realizar**  1. **Pruebas Funcionales**:    * **Descripción**:      + Validar que las funcionalidades del sistema cumplen con los casos de uso.      + Asegurar que cada módulo del software realiza correctamente las tareas para las que fue diseñado.    * **Actividades**:      + Probar apartados claves, como el inicio de sesión, la creación de usuarios, y las funciones entre estudiantes y profesores.    * **Responsables**:      + Felipe Diaz y Martin Rubio    * **Objetivo**:      + Garantizar que las funcionalidades trabajen según lo planeado. 2. **Pruebas Basadas en Datos**:    * **Descripción:**      + Utilizar un conjunto variado de datos de entrada para validar el comportamiento del sistema    * **Actividades:**      + Probar la inserción de datos de manera normal como de manera masiva (Específicamente en apartado de usuarios tipo alumnos)    * **Responsables:**      + Felipe Diaz y Martin Rubio    * **Objetivo:**      + Garantizar que todas las funcionalidades trabajen según lo planeado |
|  |

| Estrategia y técnicas de pruebas a aplicar  *Definir las estrategias y técnicas de pruebas que se debe desarrollar para este proyecto, actividades y responsables.* |
| --- |
| 1. **Estrategia General de Pruebas**:    * **Objetivo**: Garantizar que el software cumpla con los requisitos definidos y proporcione una experiencia de usuario óptima.    * **Enfoque**:      + Usar datos reales simulados para replicar el entorno del usuario final.      + Probar los módulos basados en cómo lo ocuparía un usuario 2. **Estrategia de pruebas funcionales:**    * Validar que cada función del software se comporta de acuerdo a lo esperado 3. **Estrategia de pruebas de basada en datos:**    * Evaluar la interacción entre diferentes módulos del sistema. |
|  |

| Definición del proceso de testing  *Listar y describir todas las actividades a desarrollar en el proceso general de testing, responsables, artefactos, etc.* |
| --- |
| **Pruebas Funcionales**  * **Actividades**:   + Validar el flujo de inicio de sesión y recuperación de contraseñas.   + Probar la creación y gestión de usuarios (estudiantes y profesores).   + Verificar las funcionalidades de comunicación (mensajería y notificaciones en el correo). * **Responsables**: Felipe Díaz y Martín Rubio. * **Objetivo**: Garantizar que las funcionalidades trabajen según lo planeado y respondan a los requisitos funcionales especificados.  **Pruebas Basadas en Datos**  * **Actividades**:   + Probar la inserción de datos en el sistema de forma normal y masiva (específicamente en la creación de usuarios tipo "alumnos").   + Validar la correcta visualización y gestión de datos en el panel administrativo. * **Responsables**: Felipe Díaz y Martín Rubio. * **Objetivo**: Garantizar que todas las funcionalidades trabajen según lo planeado, asegurando que el sistema maneje los datos correctamente y sin errores. |

| Definición de ciclos de prueba a ejecutar  *Listar y describir cantidad de ciclos de prueba a realizar en este proyecto, las tareas y actividades para cada ciclo de prueba, responsables, artefactos, etc.* |
| --- |
| **Ciclo 1: Pruebas Unitarias**  * **Actividades**: Probar individualmente los módulos de autenticación, mensajería y dashboard académica. Validar la lógica de negocio en cada funcionalidad. * **Responsables**: Felipe Díaz y Martín Rubio. * **Duración Estimada**: 1 Día  **Ciclo 2: Pruebas Funcionales**  * **Actividades**:   + Probar la creación de usuarios.   + Validar la gestión de datos académicos como calificaciones.   + Revisar las funciones de administración, como la creación y edición masiva de usuarios. * **Responsables**: Felipe Díaz y Martín Rubio. * **Duración Estimada**: 2 Días  **Ciclo 3: Pruebas de Usabilidad**  * **Actividades**:   + Realizar sesiones con usuarios clave (estudiantes , profesores y apoderados) para probar la aplicación y verificar que las funcionalidades implementadas funcionen de manera acorde. * **Responsables**: Felipe Díaz y Martín Rubio. * **Duración Estimada**: 2 Días |

| Calendarización de las actividades de pruebas  *Listado de actividades, tareas, duración, fechas, responsables, etc.* |
| --- |
|  |
| Adjuntar carta Gantt  [Carta gantt](https://drive.google.com/file/d/1_enPb1uZVeSicK2H6ixOaGywTWLdkksJ/view?usp=drive_link) |

| Resumen de riesgos  *Listado de riesgos relacionados al proceso de pruebas de S.W. Indicar riesgo, magnitud o impacto de este riesgo por etapa en el proceso.Magnitud: Alto , Significativo , Moderado, Inferior y Baja.* | | | | | |
| --- | --- | --- | --- | --- | --- |
|  | **Fase del proceso de pruebas** | | | | **Riesgo** |
| **Planificación Y toma de requerimientos** | **Desarrollo** | **Seguimiento y control** | **Cierre de proyecto** |
| **Magnitud** |  |  |  |  |
| **Alto** | Moderado | Baja | Baja | Retraso en la entrega de requisitos |
| **No aplica** | Significativo | Alto | Moderado | Defectos criticos no detectados |
| **Significativo** | Moderado | Significativo | Bajo | Fallas en la comunicación entre equipos |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

| Clasificación de los defectos  *Definir la clasificación de los defectos según su nivel de severidad* | |
| --- | --- |
| Nivel de Severidad | Descripción |
| Alto | Defectos que afectan funcionalidades importantes, pero que tienen soluciones alternativas temporales. Ejemplo: Error en la visualización de notas. |
| Moderado | Defectos que afectan funcionalidades secundarias o la experiencia del usuario, pero no impiden el uso principal del sistema. Ejemplo: Un botón con diseño incorrecto. |
| Bajo | Defectos menores, como errores ortográficos o de formato, que no afectan el uso del sistema. Ejemplo: Títulos mal alineados. |
|  |  |
|  |  |
|  |  |
|  |  |

| Definición de artefactos  *Listar y describir los artefactos que serán administrados y entregados durante este proceso de prueba.* | |
| --- | --- |
| Artefacto | Descripción |
| Registro casos prueba | Todos las pruebas que se realizarán serán registradas en este artefacto. |
|  |  |
|  |  |
|  |  |

| Condiciones de aceptación para cierre del proceso de pruebas  *Condiciones que se deben cumplir para dar término al proceso de pruebas y margen de tolerancia de aceptación de defectos.* |
| --- |
| 90% de los casos de prueba realizados superados |