**Plan de Pruebas**

**PlanAway**

***Fecha:08/11/2024***

**Tabla de contenido**

Histórico de Revisiones

| Versión | Fecha | Descripción/cambio | autor |
| --- | --- | --- | --- |
| 1.0.0 | 08/11/2024 | se comienza la creación del plan de pruebas. | Dabrian Labraña |
| 1.1.0 | 15/11/2024 | finalización del plan de pruebas | Dabrian Labraña |
|  |  |  |  |

Información del Proyecto

| Sección | CAPSTONE\_007D |
| --- | --- |
| Proyecto (Nombre) | PlanAway |
| Fecha de Inicio | 8-11-2024 |
| Fecha de Término | 23/11/2024 |

Integrantes

| Rut | Nombre | Correo |
| --- | --- | --- |
| **20068494k** | **Dabrian labraña Moreno** | **dab.labrana@alumnos.duoc.cl** |
|  |  |  |
|  |  |  |

| Alcance de las pruebas  *Definición de requisitos de S.W., módulos de Software a probar, Requisitos ambiente de pruebas y Documentación Referenciada, etc.* |
| --- |
| Un requerimiento de software se puede definir como una necesidad documentada que establece la forma en la cual debe funcionar un sistema; debe incluir los atributos y características del sistema para cumplir con las expectativas del usuario.  es por eso que a la hora de usar los testing correspondientes le dedicaremos tiempo a cada requisito entregado por el cliente, es muy importante para la satisfacción y la entrega exitosa del producto final es por eso que realizaremos diferentes tipos de pruebas tales como, testing funcional y no funcional , testing estructural de código , testing de confirmación y regresión.  podemos encontrarnos con diferentes tipos de riesgos desde el principio tales como tiempo, poca cantidad de usuarios registrados entre otros.  sabiendo algunos riesgos que podemos encontrar en el camino tenemos la ventaja de determinar qué camino seguir para evitar aquellos riesgos.  determinar el tiempo desde el principio es una gran ventaja para planificar todo el sistema es por eso que usaremos 3 días como límite para cada tipo de prueba. |

| Descripción  *Descripción general del sistema y características clave* |
| --- |
| PlanAway es una aplicación móvil enfocada en conectar arrendadores de cabañas y piscinas con potenciales arrendatarios. El sistema permite a los arrendadores crear un perfil, publicar anuncios de sus propiedades y gestionar todas las interacciones y transacciones con los usuarios, de manera similar a una red social. Esto facilita la promoción y gestión de los alquileres de propiedades recreativas. Algunas características claves de PlanAway son:   * **perfiles**: los arrendadores pueden registrarse y crear una cuenta personal. En esta cuenta pueden agregar información relevante y detalles de sus propiedades. * **Publicación de Anuncios**: Los arrendadores pueden publicar anuncios en la aplicación subiendo imágenes y descripción detallada de sus propiedades. Estos anuncios funcionan como publicaciones en redes sociales, por lo que son visibles para todos los usuarios y se pueden compartir fácilmente. * **Gestión de reservas y pagos:** Los arrendadores pueden visualizar un calendario de reservas lo que les permite gestionar fechas importantes y reservas en curso. Esto incluye la opción de recibir y aceptar solicitudes de reservas, además contiene integrado un método de pago para poder realizar el pago al instante.   Estas son solo algunas de las características que tiene la aplicación PlanAway que planea intentar darle un impulso a todos los emprendedores que tienen propiedades y no tienen el alcance suficiente. |

| Resumen de las pruebas | | |
| --- | --- | --- |
| Módulos del sistema a probar |  | |
| Objetivos de las pruebas | El objetivo de las pruebas es identificar errores de programación o estructurales.   * Identificar errores de carga o guardado en la base de datos. identificar y verificar errores en las interfaces de usuarios. * Lograr el uso correcto del diseño para el usuario sin provocar errores con alguna de las demás funciones nombradas. | |
| Detalle del orden de ejecución de los módulos | el orden de ejecución aunque sean todos independientes deben ser consecutivos:  -Proyectos  -Revisión  -Aprobación | |
| Tipos de pruebas a realizar | Los diferentes tipos de prueba que realizaremos como compañia serán los siguientes:  -Testing funcional y no funcional  -estructura de código  -testing de confirmación y regresión.  -test de mantención. | |
| Técnicas de pruebas a utilizar | Los diferentes tecnicas de pruebas a utilizar son:  -Técnica de caja negra.  -Técnica de caja blanca.  -Técnica de caja gris. | |
| Roles y responsabilidades | Rol | Responsabilidades |
| Jefe del proyecto | Matias |
| Analista QA | Dabrian Labraña |
| Tester | Dabrian Labraña |
| Desarrollar | Matias |

| Entorno y configuración de las pruebas  *Definir los requisitos de software y hardware necesarios para ejecutar las pruebas.* |
| --- |
| los requisitos de software que necesitamos para la ejecución de pruebas es de:  Software -Windows 10 64 bits, Mac Os X Jaguar -Suite Visión Empresarial  Hardware  -Disco duro ssd de 1 tera bits  -Ram 8 GB  -CPU Intel core i5-6600 / AMD Ryzen 5 2500X  -Tarjeta gráfica Nvidia GeForce GTX 660 2GB GTX 1050 2GB  -GPU AMD R9 390 8GB  -Fuente de poder de 600 watts 80 plus |

| Calendarización de las actividades de pruebas  *Listado de actividades, tareas, duración, fechas, responsables, etc.* |
| --- |
| -Testing Funcional y no funcional, el cual durará 1 días para el buen uso de los recursos y para evitar la falta de requisitos del cliente  -Estructura del código para evitar líneas de códigos innecesarias el cual estará a cargo el programador y tendrá una fecha límite de 1 días  -Testing de confirmación y regresión verificaremos el programa que ya se tenía antes para las cajas y verificaremos que nuestra mejora se implemente de una forma eficaz y sin interrumpir con otros programas pre instalados en la caja tendrá una duración de 2 días y el responsable de es el programador y el administrador de base de datos. |

| 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. Probabilidad de ocurrencia. Plan de mitigación y plan de contingencia.* |
| --- |

| Riesgo | Magnitud | Ocurrencia | Plan de Mitigación | Plan de Contingencia |
| --- | --- | --- | --- | --- |
| Restriccion de tiempo | Significativo | 50% | trabajando eficazmente cada día para entregar el producto en el tiempo correspondido. | Si no logramos entregar a tiempo el proyecto se explicará al cliente las razones y se le mostrará al cliente lo que se tiene para darle la seguridad que está realizando a la perfección. |
| Poca visibilidad por parte de los arrendadores | Alto | 60% | Se realizará publicidad a la aplicación para poder tener un alcance a todos los emprendedores con falta de visibilidad | acercarnos gradualmente a cada persona para enseñarles la app. |
| Modificación de datos sensibles | Significativo | 50% | implementar autenticación segura( firebase Authentication) | Encriptar datos sensibles. |
| Consumo excesivo de batería o mal rendimiento en dispositivos antiguos | inferior | 10% | optimizar el uso de mapas, comprimir imágenes y recursos | Crear un modo lite de la aplicación donde el usuario solamente pueda ver las funciones más focalizadas. |
| Los usuarios podrían publicar contenido inapropiado o que infrinja derechos de autor. | Significativo | 40% | Los administradores podrán revisar los datos de las publicaciones y aceptar si se suben a la página o no. | existirá un modo reporte que pueda avisarle a los administradores que la publicación que está subida a la aplicación contiene alguna sospecha. |

| Definición de artefactos  *Listar y describir los artefactos que serán administrados y entregados durante este proceso de prueba.* | |
| --- | --- |
| Artefacto | Descripción |
| Plan de pruebas | Documento que contiene toda la planificación de las actividades, requisitos, responsables del proceso de pruebas. |
| Planilla de actividad | La planilla de actividad como dice el nombre analiza y ordena las actividades a seguir a la hora del testing del software para mantener un orden y seguimiento de este. |
| Caso de prueba | Colección de todos los casos de prueba que se deben diseñar para poder verificar y validar el SW. Pueden incluir pruebas estándar, funcionales y no funcionales. Estos se pueden realizar en un documento Word, planilla Excel o utilizar alguna herramienta de gestión de pruebas. |
| Matriz de trazabilidad | La matriz de trazabilidad es una tabla que relaciona cada uno de los requerimientos con el entregable que se haya solicitado. Te permite identificar qué resultado se alcanza a través de cada requisito y, a la vez, qué requisitos son los que permiten obtener un determinado entregable. |

| 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.* |
| --- |
| Las condiciones de aceptación para el cierre del proceso son el buen uso del testing tales como, confirmar el estado de la base de datos, confirmar que los cálculos estén 100% logrados, el buen funcionamiento de todos los métodos entregados por el cliente tales como eliminar un producto, agregar un producto, buscar un producto, mostrar un producto, validar stock, imprimir las características de un producto, calcular descuento, si todas estas condiciones llegasen a cumplirce entonces el cierre del proceso de pruebas estaría completado.  Los margen de tolerancia de aceptación de defectos son de diseño del programa tales como el orden del menú o colores. |

| Glosario |
| --- |
| bd: Base de datos.  SW: Software.  SSD: Unidad de estado sólido, una alternativa de disco duro GB: gigabyte es una unidad de almacenamiento de información cuyo símbolo es el GB |