|  |
| --- |

**Especificación de Requisitos de Software (SRS)**

**Plan de Gestión de Proyecto (PGP)**

Proyecto: ¡Oh my dog!

Revisión: 1



|  |  |  |
| --- | --- | --- |

**Especificación de Requisitos de Software (SRS)**

**1) Introducción**

1. Propósito y alcance

Este documento tiene como propósito definir los requisitos del software, ya sean funcionales o no funcionales, para el desarrollo del proyecto ¡Oh my dog!, brindando así como producto final un sistema de organización centralizada para servicios que provee la veterinaria y otros destinados a la comunidad.

Este documento está dirigido a Pedro y Lucía, dueños de la veterinaria, y a los desarrolladores de este sistema. Se presenta como una base fundamental para el equipo de desarrollo.

El sistema recibe el nombre de ¡Oh my dog!, al igual que la veterinaria, y se encargará de:

* Administración de turnos en general.
* Promulgar y concretar la adopción de perros.
* Facilitar la cruza entre perros.
* Conectar a clientes con servicios de terceros ya sea paseadores o cuidadores.
* Promocionar campañas benéficas facilitando la donación de dinero para las mismas.

1. Definiciones, acrónimos y abreviaturas a considerar

Definiciones:

* Aplicación web: programa informático diseñado para ejecutarse en navegadores con uso de internet.
* No cliente: persona que no posee cuenta en el sistema, o en su defecto que no inició sesión en este.
* Cliente: persona a la que un empleado la registró en el sistema y goza de una cuenta.
* Empleado: persona que tiene registrada en el sistema y goza de una cuenta con permisos de gestión en el sistema.
* Minimalismo: corriente artística que utiliza elementos mínimos y básicos; tendencia a reducir a lo esencial y a despojar de elementos sobrantes.
* Encriptar: la acción de transcribir un texto en otro compuesto por letras, números, signos, etc de manera que no posee ninguna relación con la versión original.
* Base de datos: es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.
* Software: programas y aplicaciones que se ejecutan en un sistema informático, computadora o dispositivo electrónico.
* Hardware: los elementos físicos que componen un sistema informático, computadora o dispositivo electrónico.

1. Referencias

| **Nombre documento** | **Fecha creación** | **Autor** |
| --- | --- | --- |
| Entrevista 1 | 25/03/2023 | AllaDevs |
| Entrevista 2 | 28/03/2023 | AllaDevs |
| Epicas | 28/03/2023 | AllaDevs |
| Cuestionario “Sitio web: ¡Oh my dog!” | 28/03/2023 | AllaDevs |

**2) Descripción general**

1. Resumen de la idea del producto

¡Oh my dog! Es una aplicación web que permitirá a los clientes acceder a una amplia variedad de servicios de manera más cómoda y eficiente. La plataforma brindará una serie de funcionalidades, como la administración general de turnos, la adopción de perros, la facilitación de cruza entre los caninos, la conexión con servicios de terceros y la promoción de campañas benéficas, todo ello con el objetivo de mejorar la calidad de los servicios y brindar la atención que los perros y sus dueños que se merecen.

1. Perspectiva del producto

El sistema es una Aplicación web independiente, ya que no es parte de un sistema mayor.

En cuanto a conexiones con sistemas externos se vincula con los servicios de pago de Mercado Pago y con los de Stripe, necesarios para realizar pagos dentro de la aplicación. Los problemas que puedan ocurrir con estos sistemas solo afectan a la hora de realizar un pago, de manera que si se quiere realizar uno el usuario deberá esperar a que el sistema se normalice. El resto de las funcionalidades no se ven afectadas bajo ningún punto.

1. Características de los usuarios

En el sistema se encuentran los siguientes roles de usuario en conjunto con sus actividades permitidas:

* No cliente:
  + Realizar donaciones libres o á campaña de donación.
  + Ver información ante urgencias.
  + Ver las sucursales.
  + Ver listado de adopción con posibilidad de contactar al autor.
  + Ver listado de paseadores y cuidadores con posibilidad de contactarlos.
  + Adoptar.
* Cliente:
  + Iniciar sesión.
  + Cerrar sesion.
  + Actualizar credenciales de la cuenta.
  + Realizar donaciones libres o á campaña de donación.
  + Ver información ante urgencias.
  + Ver las sucursales.
  + Ver listado de adopción con posibilidad de contactar al autor.
  + Ver listado de paseadores y cuidadores con posibilidad de contactarlos.
  + Valorar paseador y/o cuidador.
  + Ver perfil propio.
  + Ver perros propios junto a su historial médico.
  + Ver listado de cruza con posibilidad de contactar al autor.
  + Registrar, actualizar y eliminar publicación de cruza.
  + Registrar, actualizar, eliminar publicación de adopción.
  + Ver listado de encuentro (perros perdidos, encontrados)
  + Registrar, actualizar y eliminar publicación de encuentro.
  + Contactar autor publicación de encuentro.
  + Solicitar turno.
* Empleado:
  + Iniciar sesión.
  + Cerrar sesion.
  + Actualizar credenciales de la cuenta.
  + Registrar y actualizar clientes.
  + Registrar, actualizar, eliminar y inhabilitar perro de un cliente.
  + Registrar y actualizar campañas de donación.
  + Registrar, actualizar y eliminar publicaciones de paseadores y cuidadores de perros.
  + Ver listado de sucursales.
  + Registrar, actualizar y eliminar sucursal.
  + Ver turnos.
  + Registrar, cancelar y efectivizar turnos.
  + Confirmar y reproponer propuestas de turnos.

1. Evolución previsible del sistema

N/A.

**3) Requisitos del Software**

Requisitos de Interfaz

1. Interfaz de Usuario

La interfaz de usuario es la de una aplicación web. El estilo sera animado y minimalista mantetiendo una estetica sobria, para lo cual se utiliza una paleta de colores: verde agua, naranja, blancos, negros y composiciones de los mismos.

El logo siempre será visible en la parte superior izquierda.

La interfaz sigue el concepto de responsive design, lo cual quiere decir que la interfaz se adapta al dispositivo en el cual se encuentre para una mejor experiencia de usuario sin limitar sus funcionalidades por factores del dispositivo.

Como comportamientos específicos se agregara:

* Énfasis en las campañas de donación dentro de la página de inicio.
* El acceso al apartado de cruza debe ser difícil.
* Énfasis en que adoptar es mejor que cruzar perros dentro de la página de cruza.

1. Interfaces de Software

El sistema interactúa con los servicios de pago de Stripe y Mercado Pago para realizar las donaciones libres o campañas benéficas de la veterinaria. La integración con ambos servicios ocurre al momento de que el usuario confirma que quiere realizar una donación ya sea de modo libre para la veterinaria o en una campaña benéfica, en este momento se le da la opción de alguno de los métodos de pagos y el servicio se encarga de la transacción del mismo.

1. Interfaces de Hardware

N/A.

Requisitos funcionales

Manejo de cuenta:

* Iniciar sesión
* Cerrar sesion
* Ver perfil
  + Ver listado de perros
    - Ver perfil perro
* Editar cuenta

Gestion de clientes:

* Registrar cliente
* Ver listado de clientes
  + Modificar perro
    - Agregar consulta
  + Agregar perro
  + Borrar perro
  + Modificar cliente
  + Ver donaciones
  + Ver historial de turnos

Servicios de atención:

* Solicitar turno
* Ver turnos
  + Confirmar propuesta de cambio
* Ver listado de turnos
  + Filtrar turnos
  + Gestionar propuesta de cambio de turno
  + Cancelar turno
  + Aceptar turno
  + Efectivizar turno
* Mostrar sucursales
* Información urgencia

Servicios de terceros:

* Ver listado de servicios de terceros
  + Filtrar servicios de terceros
  + Valorar servicio de terceros
  + Agregar servicio de terceros
  + Actualizar servicio de terceros
  + Eliminar servicio de terceros

Servicios de vinculación:

* Ver publicación de vinculación
* Ver listado de cruza
  + Filtrar cruzas
  + Agregar cruza
  + Actualizar cruza
  + Eliminar cruza
  + Contactar autor de cruza
* Ver listado de adopción
  + Filtrar adopciones
  + Adoptar perro
  + Actualizar adopción
  + Publicar adopcion
  + Resolver adopcion
* Ver listado de pérdidas/encuentros
  + Filtrar pérdidas/encuentros
  + Publicar perdido
  + Publicar encontrado
  + Actualizar pérdida/encuentro
  + Contactar autor búsqueda/encuentro

Manejo de donaciones

* Donar con tarjeta
* Donar con Mercado Pago
* Ver listado de donantes
* Ver listado de campañas donación
  + Editar campaña de donación
  + Agregar campaña de donación

Información de la veterinaria

* Mostrar inicio
* Mostar sucursales
* Ver listado sucursales
  + Agregar sucursal
  + Actualizar sucursal
  + Eliminar sucursal

Requisitos no funcionales

1. Fiabilidad

El sistema deberá funcionar normalmente en condiciones normales. Los eventos como fallos de la red, fallos de energía eléctrica, entre otros eventos de la misma naturaleza, se consideran excepciones lo cual provocará que el sistema deje de funcionar hasta recuperar las condiciones normales.

1. Mantenibilidad

El sistema recibirá mantenimiento sin costo como garantía por un periodo de 4 meses luego del lanzamiento del mismo por la empresa AllaDevs. Pasado este periodo de tiempo la empresa se compromete a establecer un acuerdo con remuneración monetaria para continuar con el mantenimiento si los dueños de la Veterinaria lo quieren.

1. Multiplataforma

El sistema al ser una aplicación web, por definición es multiplataforma dado que puede ser utilizada en cualquier dispositivo con un navegador, como lo son un ordenador, tablet, móvil, entre otros.

1. Pagos externos

El sistema maneja los pagos a través de los servicios de Stripe y Mercado Pago los cuales son usados en el mercado a nivel global para las transacciones monetarias asegurando la confiabilidad, fiabilidad, seguridad y validación de los pagos para las partes involucradas en ellos.

1. Seguridad

El sistema cuenta con un sistema de autenticación para el acceso a funcionalidades específicas a través de cuentas, en las que se encuentran dos tipos: cuenta de cliente y cuenta de empleado. Al momento de utilizar el sistema si no se inició sesión solo se puede efectuar las operaciones del rol No Cliente. Si un usuario procede a iniciar sesión y lo logra, a través de las mismas credenciales se conoce el rol Cliente o Empleado que habilitan las funcionalidades adecuadas.

Las credenciales de las cuentas se almacenan de manera encriptada en la base de datos.

El resto de los datos se almacenan en la base de datos, los cuales solo los puede acceder el mismo sistema con permisos únicos.

1. Rendimiento

El sistema debe poder manejar un gran número de usuarios en simultáneo sin degradaciones perceptibles en el tiempo de respuesta de las acciones realizadas.

Debe hacer un uso óptimo de los recursos de hardware de los usuarios que consuman la aplicación sin estresar los mismos ni afectar las demás tareas que estos puedan estar desarrollando en simultáneo y/o posterior del uso de la aplicación.

**Plan de Gestión de Proyecto (PGP)**

**1) Introducción**

1. Propósito y alcance

Este documento tiene como propósito ser el marco del plan de realización del proyecto ¡Oh my dog! Y los recursos necesarios para llevarlo a cabo. En este se especifica los plazos, roles, responsabilidades, recursos y presupuesto estimado para la realización exitosa de todos los objetivos del proyecto teniendo en cuenta como parte fundamental los riesgos presentes en este.

Este documento se encuentra dirigido a las diferentes partes interesadas en el proyecto, como lo son el equipo de desarrollo de AllaDevs y los dueños de la veterinaria Pedro y Lucia.

El proyecto ¡Oh my dog! es una aplicación web la cual servirá como servicio de gestión de turnos para la veterinaria de igual nombre, ademas este brindara la posibilidad de facilitar la comunicación y/o conexión entre diferentes individuos en contexto de posesión o no de perros, como lo es publicitar servicios de paseo y cuidado de terceros a la veterinaria, conectar perros perdidos con sus dueños, publicitación de campañas de donación, publicitación de perros en adopción y para cruza, entre otras funcionalidades más que le aportan un bien a la comunidad.

1. Definiciones, acrónimos y abreviaturas a considerar

Definiciones:

* Aplicación web: programa informático diseñado para ejecutarse en navegadores con uso de internet.
* No cliente: persona que no posee cuenta en el sistema, o en su defecto que no inició sesión en este.
* Cliente: persona a la que un empleado la registró en el sistema y goza de una cuenta.
* Empleado: persona que tiene registrada en el sistema y goza de una cuenta con permisos de gestión en el sistema.
* Minimalismo: corriente artística que utiliza elementos mínimos y básicos; tendencia a reducir a lo esencial y a despojar de elementos sobrantes.
* Encriptar: la acción de transcribir un texto en otro compuesto por letras, números, signos, etc de manera que no posee ninguna relación con la versión original.
* Base de datos: es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.
* Software: programas y aplicaciones que se ejecutan en un sistema informático, computadora o dispositivo electrónico.
* Hardware: los elementos físicos que componen un sistema informático, computadora o dispositivo electrónico.
* Scrum: metodologia agil (rapido, dinámico y efectivo) para la gestión de proyectos.
* Sprint: periodo de tiempo determinado en que se realiza trabajo determinado para cumplir metas propuestas en el proyecto con la metodología Scrum.
* Planificación Sprint: proceso de planificación de un sprint donde se determinan metas, tiempos y otros aspectos relacionados a un Sprint.
* Dominio: nombre único que identifica a una subárea de internet.
* Framework: es un conjunto de herramientas y reglas que se utilizan para desarrollar software.
* NodeJS: entorno de ejecución de código abierto para JavaScript construido en el motor JavaScript V8 de Chrome.
* JavaScript: lenguaje de programación que permite implementar características complejas e interactividad en páginas web.

1. Referencias

| **Nombre documento** | **Fecha creación** | **Autor** |
| --- | --- | --- |
| Entrevista 1 | 25/03/2023 | AllaDevs |
| Entrevista 2 | 28/03/2023 | AllaDevs |
| Epicas | 28/03/2023 | AllaDevs |
| Cuestionario “Sitio web: ¡Oh my dog!” | 28/03/2023 | AllaDevs |
| SRS | 26/04/2023 | AllaDevs |
| Full Stack Developer (argentina) | 26/04/2023 | www.salaryexpert.com |
| Registrar dominio | 26/04/2023 | www.namecheap.com |
| Servidor | 26/04/2023 | vercel.com/pricing |
| Base de datos | 26/04/2023 | www.mongodb.com/pricing |
| Cotizacion dolar | 26/04/2023 | dolarhoy.com |

**2) Planes generales**

1. Entregables del proyecto

El desarrollo de la aplicación web consta de cinco entregas de las cuales dos son de documentos y las tres restantes siendo demos del estado actual del desarrollo, resaltando que en la última demo la aplicación se encontrará finalizada.

Se detallan los entregables en la tabla siguiente.

| **Entregable** | **Fecha de entrega** |
| --- | --- |
| Entrevista 1, Entrevista 2, Cuestionario y Épicas | 31/03/2023 |
| SRS, PGP y Pila de producto | 28/04/2023 |
| Demo 1 | 02/06/2023 |
| Demo 2 | 23/06/2023 |
| Demo 3 | 14/07/2023 |

1. Calendario y resumen del presupuesto

El desarrollo se realiza en el periodo comprendido entre el 10/03/2023 y el 14/07/2023. En la tabla siguiente se detalla un calendario con las etapas de desarrollo.

| **Etapa** | **Accion/es** | **Fecha** |
| --- | --- | --- |
| Elicitación de requerimientos | Entrevista 1 | 10/03/2023 |
| Entrevista 2 | 17/03/2023 |
| Especificación de requerimientos | Entrega de Entrevistas 1 y 2, Cuestionario y Épicas | 31/03/2023 |
| Entrega de SRS, PGP y Pila de producto | 28/04/2023 |
| Sprint 1 | Planeacion Sprint 1 | 05/05/2023 |
| Scrum diario 1-1 | 12/05/2023 |
| Scrum diario 1-2 | 19/05/2023 |
| Demo 1 | 02/06/2023 |
| Sprint 2 | Planeación Sprint 2 | 02/06/2023 |
| Scrum diario 2-1 | 09/06/2023 |
| Scrum diario 2-2 | 16/06/2023 |
| Demo 2 | 23/06/2023 |
| Sprint 3 | Planeación Sprint 3 | 23/06/2023 |
| Scrum diario 3-1 | 30/06/2023 |
| Scrum diario 3-2 | 07/07/2023 |
| Demo 3 | 14/07/2023 |

El presupuesto final es de 1239000 Pesos Argentinos para el desarrollo completo.

1. Plan del personal

Para llevar a cabo el proyecto serán necesarios tres programadores con conocimiento, experiencia y determinación. Estos mismo deben ser capaces de realizar tanto la elicitación de los requerimientos, como el diseño lógico, arquitectura y gráfico producto de los requerimientos, como también la realización efectiva de los previos en código para la aplicación web. No es requerido especificación en ninguna área, solo una correcta organización de esfuerzos y tiempos mediante comunicación. Tres desarrolladores fullstack serían pertinentes en este proyecto.

El personal será contratado durante las 18 semanas, aprox. 4 meses, que se estimó el desarrollo completo.

**3) Presupuesto**

1. Principales actividades del proyecto

El proyecto consiste en el desarrollo de una aplicación web desde cero para la veterinaria ¡Oh my dog!, para lo cual se requieren las siguientes actividades:

1. Elicitar requerimientos: elaboración y realización de entrevistas y cuestionario.
2. Especificar requerimientos: elaboración de especificaciones de requerimientos de manera formal, describiendo las funcionalidades y restricciones del sistema a desarrollar. Estos documentos además tienen la importancia de ser contratos con el cliente sobre el sistema solicitado.
3. Diseño conceptual: elaboración de diseños abstractos en cuanto al manejo de la información del sistema y bocetos gráficos de la aplicación en base a los requerimientos.
4. Desarrollo: la aplicación será en un entorno pertinente como lo es la plataforma NodeJS que facilita el desarrollo de aplicaciones web, donde se usará el conjunto de tecnologías/frameworks SvelteKit, tailwindcss, Prisma, MongoDB y TypeScript ya que estas brindan la adaptabilidad, agilidad y efectividad para realizar un proyecto de esta envergadura. Para lo cual se gestionará el proceso con la metodología de desarrollo Scrum, para lo cual se gestiona el desarrollo completo en los submódulos (no se requiere la completitud del submódulo para que funciones determinados elementos del mismo):
   1. Sistema de usuarios: contiene las funcionalidades respectivas a autenticación, manejo de cuentas.
   2. Sistema de turnos: contiene las funcionalidades respectivas a la gestión de turnos por parte de los empleados y la solicitud de turnos de los clientes.
   3. Sistema de servicios de terceros: contiene las funcionalidades respectivas a la publicitación de servicio de cuidado y paseo de perros.
   4. Sistema de donaciones: contiene las funcionalidades respectivas a la gestión de campañas de donación, descuentos por donaciones y donaciones libres de los clientes y no clientes.
   5. Sistema de sucursales: contiene las funcionalidades respectivas a la gestión de sucursales de la veterinaria por parte de los empleados así como la visualización por parte de clientes y no clientes.
   6. Sistema de cruzas: contiene las funcionalidades respectivas a gestión de perros para cruza por parte de los clientes.
   7. Sistema de encuentros: contiene las funcionalidades respectivas a la publicación de perros perdidos, encontrados tanto así como para ver los casos resueltos.
   8. Sistema de adopciones: contiene las funcionalidades respectivas a la gestión de dar en adopción un perro por parte de un cliente o no cliente, como también ver los disponibles a ser adoptados por los clientes y no clientes.
5. Despliegue efectivo: se realizará el despliegue final efectivo del sistema para que pueda ser usado en la veterinaria como consumido por los clientes de la misma de manera que funcione adecuadamente y de la manera esperada.
6. Mantenimiento: se realiza el monitoreo del funcionamiento del sistema luego que se encuentre activo para verificar su correcto funcionamiento.
7. Asignación de esfuerzo

| **Actividad** | **Responsable/s (cantidad)** | **Esfuerzo unitario (hs)** | **Esfuerzo total (hs)** |
| --- | --- | --- | --- |
| Elicitar requerimientos | 3 | 5 | 15 |
| Especificar requerimientos | 3 | 8 | 24 |
| Diseño conceptual | 3 | 4 | 12 |
| Sistema de usuarios | 3 | 18 | 54 |
| Sistema de turnos | 3 | 7 | 21 |
| Sistema de servicios de terceros | 3 | 4 | 12 |
| Sistema de donaciones | 3 | 10 | 30 |
| Sistema de sucursales | 3 | 6 | 18 |
| Sistema de cruzas | 3 | 8 | 24 |
| Sistema de encuentros | 3 | 14 | 42 |
| Sistema de adopciones | 3 | 12 | 36 |
| Despliegue efectivo | 3 | 6 | 18 |
| Mantenimiento | 3 | 8 | 24 |

Total de horas entre todas las actividades: 330 hs.

1. Presupuesto final

El presupuesto se calculará en base a la suma de los presupuestos para los salarios de los desarrolladores más los costos de los recursos. El sueldo de cada desarrollador será el producto entre las horas empleadas y el salario por hora de la tabla a continuación:

| **Especialidad** | **Sueldo base** | **Extra** | **Total (salario por hora)** |
| --- | --- | --- | --- |
| Fullstack | 5 USD | 2 USD | 7 USD |

Precio total del personal = 7 USD/hs \* 330 hs = 2310 USD

Los recursos utilizados y su precio se describen en la tabla a continuación:

| **Nombre** | **Tiempo** | **Costo** |
| --- | --- | --- |
| Dominio ohmydog.vet | 1 año | 32 USD |
| Servidor | 1 mes | 20 USD |
| Base de datos | 1 mes | 5 USD |

Precio total de recursos a 1 año = 1\* 31.98 USD + 12 \* 20 USD + 12 \* 25 USD = 332 USD

Costo total en USD = 2310 USD + 332 USD = 2642 USD

Costo total en Pesos Argentinos (27/04/2023) = 2642 USD \* 469 Pesos Argentinos/USD = 1239000 Pesos Argentinos

**4) Riesgos**

Para estandarizar la medida de los riesgos se usará la métrica de impacto que describe como un riesgo impacta en el sistema. El impacto se medirá en los valores 1, 2, 3 y 4 y cada valor se describe en la tabla a continuación:

| **Valor** | **Alias** | **Interpretacion** |
| --- | --- | --- |
| 1 | Catastrofico | Cancelacion o suspension del proyecto |
| 2 | Serio | Reducción de rendimiento, retrasos en la entrega, excesos importante en costo |
| 3 | Tolerable | Reducciones mínimas de rendimiento, posibles retrasos, exceso en costo |
| 4 | Insignificante | Incidencia mínima en el desarrollo |

En el desarrollo y posterior a él se encuentran riesgos que pueden afectar en el sistema, para lo cual a continuación se detallan con un nombre, probabilidad de ocurrir, impacto en el sistema si sucede, las medidas mitigadoras para que no ocurra, estrategias de contingencia en caso de que suceda para minimizar el impacto inevitable y el miembro del equipo de desarrollo designado como responsable del riesgo.

* **Riesgo:** Menos código reusable del previsto.

**Probabilidad:** 30%.

**Impacto:** 3.

**Medidas mitigadoras:** Cumplir con buenas prácticas de programación como los criterios y heurísticas de diseño, para lograr un software modularizado y reusable.

**Plan de contingencia:** Aplicar técnicas de refactoring sobre el código.

**Responsable:** Bonoris Nicolas.

* **Riesgo:** Cambios en el equipo de desarrollo.

**Probabilidad:** 10%.

**Impacto:** 3.

**Medidas mitigadoras:** Fomentar un ambiente de trabajo agradable y alentar a los miembros del equipo a comunicarse abiertamente.

**Plan de contingencia:** : Evaluar la necesidad de traer nuevos miembros al equipo o redistribuir las responsabilidades actuales.

**Responsable:** Gallardo Lucas.

* **Riesgo:** Subestimación del tamaño del sistema.

**Probabilidad:** 40%.

**Impacto:** 2.

**Medidas mitigadoras:** Realizar la elicitación de requerimientos habiendo analizado el negocio y mercado. Analizar detalladamente la información recaudada y definir correctamente los requerimientos subyacentes. Investigar profundamente de ser necesario.

**Plan de contingencia:** Pautar con el cliente una nueva fecha de entrega del producto.

**Responsable:** Gallardo Lucas.

* **Riesgo:** Enfermedad del personal.

**Probabilidad:** 20%.

**Impacto:** 4.

**Medidas mitigadoras:** Organizar y gestionar una buena comunicación entre los miembros del equipo de desarrollo para comprender las tareas que los demás desarrolladores tienen que cumplir. Fomentar un ambiente de trabajo saludable y flexible, en el que se promueva el equilibrio entre la vida laboral y personal.

**Plan de contingencia:** Enviar un médico a domicilio, para procurar su pronta recuperación.

**Responsable:** Suarez Josue.

* **Riesgo**: Usuario no se adapta a la UI.

**Probabilidad:** 25%.

**Impacto:** 2.

**Medidas mitigadoras:** Pruebas de usabilidad, diseño y desarrollo iterativo, documentación clara de la interfaz de usuario.

**Plan de contingencia:** Capacitación adicional para el usuario, posible revisión del diseño de la UI.

**Responsable:** Bonoris Nicolas.

* **Riesgo:** Cambios de licencias de software..

**Probabilidad:** 10%.

**Impacto:** 1.

**Medidas mitigadoras:** Investigación exhaustiva de las licencias de software antes de su adopción, implementación de alternativas de software si es necesario.

**Plan de contingencia:** Cambio a alternativas de software, reasignación de recursos para adaptarse al nuevo software, posible retraso en el proyecto.

**Responsable:** Suarez Josue.

* **Riesgo:** Rendimiento del sistema no es el esperado.

**Probabilidad:** 20 %.

**Impacto:** 2.

**Medidas mitigadoras:** Pruebas de rendimiento durante desarrollo, diseño y desarrollo iterativo, revisión y optimización del código.

**Plan de contingencia:** Revisión y optimización del código, posible reasignación de recursos para mejorar el rendimiento.

**Responsable:** Gallardo Lucas.