**Proyecto Sistema Unidad Territorial**

**GESTIÓN DE RIESGOS**

Alumnos:

Marcelo Madriaga

Alejandro Villa Villavicencio

Matías General

Profesor Guia:

Willy Francisco Bascunan Silva

**Contenido**

[Nuestra empresa. 3](#_heading=h.2s8eyo1)

[Beneficios Clave del Proyecto 3](#_heading=h.2np1h8d9tkzp)

[Metodología 3](#_heading=h.dd8md5mi3ueg)

[Impacto en la Comunidad 3](#_heading=h.u1p5cynuymt0)

[Versión del Plan 4](#_heading=h.30j0zll)

[Objetivo de la Versión 4](#_heading=h.1fob9te)

[Propósito del Plan 4](#_heading=h.3znysh7)

[Historial de Revisiones: 4](#_heading=h.2et92p0)

[Introducción 5](#_heading=h.tyjcwt)

[Contexto actual 5](#_heading=h.3dy6vkm)

[Resumen ejecutivo 5](#_heading=h.1t3h5sf)

[Propósito del proyecto 5](#_heading=h.4d34og8)

[Metodología a Implementar 6](#_heading=h.17dp8vu)

[Estructura del Equipo Scrum 6](#_heading=h.3rdcrjn)

[Desarrollo 7](#_heading=h.26in1rg)

[Mapa de Procesos 7](#_heading=h.35nkun2)

[Diagrama AS-IS 8](#_heading=h.1ksv4uv)

[8](#_heading=h.899qq76v9fmr)

[Diagrama TO-BE 8](#_heading=h.ozadowc85r7)

[8](#_heading=h.uabya6w0awj9)

[Objetivo del Proyecto 9](#_heading=h.vp81nec9nhyo)

[Alcance del Proyecto 9](#_heading=h.f5zsdteuux9b)

[Entregables del Proyecto 10](#_heading=h.3hbiq29bw4s9)

[Stakeholders Involucrados 10](#_heading=h.hqg9ab9c892r)

[Restricciones y Limitaciones 10](#_heading=h.lv10zsgw17m9)

[Cronograma de Desarrollo del Software 11](#_heading=h.44sinio)

[Sprint 1 11](#_heading=h.29yienl5auaf)

[Sprint 2 11](#_heading=h.hx8lsdamenh6)

[Sprint 3 12](#_heading=h.wxwweipt5rs1)

[Sprint 4 12](#_heading=h.58bn3xzg18i8)

[Sprint 5 12](#_heading=h.j4boj8wx9g8l)

[Sprint 6 Final 12](#_heading=h.44kgluryy00)

[Diagrama EDT 14](#_heading=h.wnye8j481yf8)

[Diagrama RBS 15](#_heading=h.gkcapg251pv7)

[Áreas involucradas: 16](#_heading=h.8m4fy2q7k5av)

[Riesgo asociado a cada área 18](#_heading=h.wss0hb6c4zfh)

[**Desarrollo de Software** 18](#_heading=h.fzwr738a0932)

[2. Gestión de Proyectos 18](#_heading=h.uofkfz538w9a)

[3. Calidad y Pruebas 19](#_heading=h.h81f3zf32qfp)

[4. Soporte Técnico 19](#_heading=h.vb34ws406vkv)

[Actores y responsables 20](#_heading=h.2jxsxqh)

[**1. Product Owner (PO)** 20](#_heading=h.vfidltx59la6)

[**2. Scrum Master** 20](#_heading=h.mb8gfy5s5i14)

[**3. Equipo de Desarrollo** 21](#_heading=h.mq7qr4otqd2o)

[**4. Equipo de Pruebas (Testers)** 21](#_heading=h.p44zqk6juz5p)

[**5. Vecinos y Usuarios Finales** 22](#_heading=h.33vwehxxgb3m)

[**6. Equipo de Soporte Técnico** 22](#_heading=h.h2o03wl5sphd)

[**7. Líder de la Junta de Vecinos (Stakeholder Principal)** 23](#_heading=h.jam3zd2j765)

[Identificación de los riesgos 24](#_heading=h.z337ya)

[Costos del proyecto 28](#_heading=h.3j2qqm3)

[**5. Riesgos** 31](#_heading=h.4i7ojhp)

[Riesgos 32](#_heading=h.2xcytpi)

[Total 32](#_heading=h.1ci93xb)

[Conclusión 33](#_heading=h.3whwml4)

# **Nuestra empresa.**

**Sistema Unidad Territorial.**

El Sistema de Unidad Territorial para la Junta de Vecinos de San Bernardo es un proyecto diseñado para modernizar la gestión comunitaria mediante la implementación de una solución tecnológica integral. La propuesta busca abordar la falta de digitalización en las juntas de vecinos, que actualmente dependen de procesos manuales o poco centralizados, lo que dificulta la eficiencia y accesibilidad en la gestión de actividades, proyectos, y recursos de la comunidad.

El proyecto tiene como objetivo principal desarrollar una plataforma digital que permita a la junta de vecinos organizar de manera eficiente actividades comunitarias, gestionar proyectos vecinales, emitir certificados de residencia, y agilizar la comunicación entre los vecinos y el directorio. La plataforma estará disponible tanto en versión web como móvil, garantizando accesibilidad y usabilidad para todos los miembros de la comunidad mayores de 14 años.

### **Beneficios Clave del Proyecto**

1. Mejora en la eficiencia organizativa: La digitalización de los procesos permitirá reducir tiempos y costos en la gestión de documentos, espacios comunes y actividades vecinales.
2. Participación ciudadana: Al facilitar el acceso y la interacción de los vecinos con la junta, el sistema fomentará una mayor participación y colaboración dentro de la comunidad.
3. Transparencia y responsabilidad: La plataforma ofrecerá mayor visibilidad de las actividades y decisiones del directorio, aumentando la confianza y el compromiso de los vecinos.
4. Accesibilidad tecnológica: El sistema permitirá a cualquier junta de vecinos de Chile adaptarse a la tecnología, independientemente de sus recursos, promoviendo la inclusión digital.

### **Metodología**

El desarrollo del proyecto se llevará a cabo utilizando la metodología Scrum, que permitirá una gestión ágil y eficiente a través de ciclos de trabajo cortos (sprints). Cada sprint se enfocará en aspectos específicos del desarrollo, desde la planificación hasta la implementación y pruebas del sistema, asegurando la entrega continua de valor.

### **Impacto en la Comunidad**

El sistema propuesto beneficiará directamente a la junta de vecinos de San Bernardo y será un modelo replicable para otras comunidades en Chile. Facilitará la toma de decisiones, mejorará la comunicación y gestionará recursos de manera más eficiente, impactando positivamente en la calidad de vida de los vecinos.

# **Versión del Plan**

Nombre del Proyecto:

Versión del Plan:2.3

Fecha de la Versión: 27 de agosto de 2024

Autores: Alejandro Villa, Marcelo Madriaga, Matías General

# **Objetivo de la Versión**

Esta versión inicial del plan está diseñada para establecer un marco claro para la estructuración y digitalización de la junta de vecinos de la comuna de San Bernardo, asegurando que todas las partes interesadas comprendan el proceso y las expectativas.

# **Propósito del Plan**

El propósito de este plan es proporcionar una guía detallada para el desarrollo, implementación y gestión del proyecto de solución informática a la junta de vecinos de San Bernardo. Define los objetivos del proyecto, los roles y responsabilidades de los miembros del equipo, las metodologías a utilizar, el alcance del proyecto, el cronograma, el presupuesto y otros aspectos clave necesarios para el éxito del proyecto.

El propósito del plan es proporcionar una guía detallada para el proyecto de la junta de vecinos, incluyendo la identificación de áreas clave de mejora, la automatización de los procesos y la implementación de mejores prácticas para la gestión comunitaria.

# **Historial de Revisiones:**

| **Versión** | **Fecha de Revisión** | **Descripción de los Cambios** | **Autor de los Cambios** |
| --- | --- | --- | --- |
| 1.0 | 23/08/2024 | Versión Inicial del Plan de Proyecto | Alejandro Villa |
| 2.0 | 26/08/2024 | Nueva versión del documento con errores corregidos | Marcelo Madriaga |
| 2.3 | 28/08/2024 | Actualización de documento con nuevas directrices | Matías General |

# 

# 

# **Introducción**

## **Contexto actual**

**Sistema Unidad Territorial**

La junta de vecinos ha identificado la necesidad de modernizar sus procesos y mejorar la eficiencia en la gestión comunitaria. Actualmente, las actividades de la junta se manejan de manera descentralizada, lo que ha generado ineficiencias y falta de claridad en la toma de decisiones, el proyecto tiene como propósito centralizar y optimizar los procesos de la junta de vecinos, facilitando la toma de decisiones, la participación comunitaria y la transparencia en la gestión.

# **Resumen ejecutivo**

El Sistema de Unidad Territorial para la Junta de Vecinos de San Bernardo es un proyecto enfocado en digitalizar la gestión comunitaria. A través de una plataforma web y móvil, la junta podrá gestionar actividades, proyectos, y recursos, como la emisión de certificados de residencia y la organización de espacios comunes. El objetivo principal es mejorar la eficiencia operativa, facilitar la participación ciudadana y promover la transparencia en la toma de decisiones.

Este proyecto emplea la metodología Scrum, permitiendo un desarrollo ágil y colaborativo, asegurando entregas continuas y adaptaciones a los requisitos de la comunidad. El sistema busca ser un modelo replicable para otras juntas de vecinos, impactando positivamente en la gestión comunitaria y fortaleciendo la cohesión social.

# **Propósito del proyecto**

El propósito del Sistema de Unidad Territorial es digitalizar y optimizar la gestión de la Junta de Vecinos de San Bernardo, facilitando la organización de actividades, la administración de recursos y la emisión de documentos. La plataforma busca mejorar la eficiencia operativa, fomentar la participación ciudadana y promover la transparencia en la toma de decisiones, proporcionando una solución accesible y adaptable a cualquier junta de vecinos en Chile.

## 

# **Metodología a Implementar**

Se utilizará la metodología de Scrum. El proyecto tiene una duración total de 66 días hábiles, por lo que podemos planificar Sprints de 2 semanas para permitir iteraciones suficientes y abordar las diversas fases y actividades.

## 

# **Estructura del Equipo Scrum**

Product Owner (PO): Gestiona el backlog y maximiza el valor del producto.

Scrum Master: Facilita el proceso Scrum y elimina impedimentos.

Equipo de Desarrollo: Incluye desarrolladores, diseñadores, testers y especialistas en seguridad.

# **Desarrollo**

## **Mapa de Procesos**

En este mapa de procesos se detalla el flujo para el desarrollo de una solución digital, con la cual automatizamos los diferentes procedimientos de la junta de vecinos. Con este plan de desarrollo, organizamos las tareas que se asignan a los responsables y se llevarán a cabo, ajustando las actividades y responsabilidades a los sprints programados. En estas fases se analizarán posibles retrasos o mejoras mediante reuniones programadas, donde participarán todos los interesados en el proyecto. A través de esta guía, se busca cumplir con los plazos establecidos y se evaluarán las decisiones orientadas a cumplir con los requerimientos del proyecto. Lo llamamos: ¡El camino del éxito!



























# 

# **Diagrama AS-IS**

***Planificación de Riesgos***

## 

# **Diagrama TO-BE**

## 

## 

## 

# **Objetivo del Proyecto**

El propósito del proyecto "Sistema Unidad Territorial" es desarrollar una solución tecnológica que mejore la gestión de las juntas de vecinos, con especial énfasis en la organización de actividades, manejo de recursos y comunicaciones. Este sistema está orientado a resolver las necesidades de las juntas de vecinos, permitiendo una gestión más eficiente y efectiva en sus operaciones diarias.

# **Alcance del Proyecto**

Desarrollar una solución tecnológica integral que permita a las juntas de vecinos gestionar de manera eficiente sus actividades, recursos y la comunicación a través de una aplicación web y móvil, mejorando así la organización y participación dentro de la comunidad.

Diseñar la arquitectura del sistema, la cual incluye el backend, frontend y la aplicación móvil, que asegure la escalabilidad y la seguridad de la información.

El proyecto consistirá en el desarrollo de un sistema utilizando NodeJS como backend y Firebase para la seguridad de inicio de sesión y el almacenamiento de datos. El frontend web será desarrollado con Ionic y Angular. El sistema permitirá la gestión de eventos, la administración de presupuestos, la comunicación interna, y la difusión de noticias relevantes para la comunidad. La app móvil también será desarrollada en Ionic y Angular, proporcionando una experiencia fluida y multiplataforma.

Crear una aplicación móvil utilizando Ionic y Angular, compatible con dispositivos Android 10 en adelante, que permita a los usuarios gestionar actividades y recibir notificaciones en tiempo real.

Realizar pruebas de usabilidad y funcionalidad en diferentes versiones de Android y navegadores web para asegurar la compatibilidad y el buen desempeño del sistema en todos los entornos.

Documentar cada fase del desarrollo del proyecto, incluyendo la toma de requisitos, diseño, implementación, pruebas y entrega final, para garantizar la trazabilidad y transparencia del proceso.

# 

# 

# **Entregables del Proyecto**

* Documentación del Product Backlog
* Sprint Backlogs
* Incrementos del Producto
* Documentación Técnica y Manuales de Usuario
* Pruebas y Resultados de Evaluación
* Informe Final del Proyecto

# **Stakeholders Involucrados**

* Junta de Vecinos
* Residentes
* Desarrolladores
* Scrum Master
* Product Owner

# **Restricciones y Limitaciones**

* Tiempo: El proyecto debe completarse en un período específico.
* Recursos: Disponibilidad de recursos tecnológicos y humanos.
* Presupuesto: Limitaciones financieras que afectan el alcance del proyecto.

# 

# 

# 

# 

# 

# 

# **Cronograma de Desarrollo del Software**

***Implementación de Scrum en el Proyecto***

### 

### **Sprint 1**

Duración y Fechas: 22-08-24 a 05-09-24 (10 días hábiles)

* Actividades:
  + Definición del Product Backlog
  + Identificación y Priorización de Historias de Usuario
  + Estimación de Esfuerzos
  + Planificación del Sprint (Sprint Planning)
  + Asignación de Tareas para el Sprint
  + Daily Scrum Meetings

### **Sprint 2**

Duración y Fechas: 06-09-24 a 10-10-24 (23 días hábiles)

* Actividades:
  + Desarrollo Iterativo del Producto
  + Implementación de Funcionalidades
  + Revisión de los Criterios de Aceptación de Historias de Usuario
  + Revisión de Código y Colaboración en Pares
  + Daily Scrum Meetings
  + Identificación y Evaluación de Riesgos (continuo)

### 

### 

### 

### **Sprint 3**

Duración y Fechas: 11-10-24 a 17-10-24 (05 días hábiles)

* Actividades:
  + Ejecución de Pruebas Unitarias e Integración
  + Revisión de Pruebas de Aceptación (User Acceptance Testing - UAT)
  + Ajustes basados en el Feedback de las Pruebas
  + Actualización del Backlog con Mejoras y Correcciones
  + Daily Scrum Meetings

### **Sprint 4**

Duración y Fechas: 18-10-24 a 24-10-24 (05 días hábiles)

* Actividades:
  + Despliegue del Incremento en un Entorno de Pruebas o Producción
  + Capacitación a Usuarios Finales
  + Documentación del Uso del Sistema
  + Revisión de la Retroalimentación del Usuario (Feedback Loop)
  + Daily Scrum Meetings

### **Sprint 5**

Duración y Fechas: 25-10-24 a 07-11-24 (07 días hábiles)

* Actividades:
  + Monitoreo del Rendimiento y Uso del Sistema
  + Identificación de Áreas de Mejora
  + Incorporación de Mejoras en el Product Backlog
  + Daily Scrum Meetings

### **Sprint 6 Final**

Duración y Fechas: 08-11-24 a 26-11-24 (13 días hábiles)

* Actividades:
  + Revisión Final del Proyecto y Confirmación de Entrega (Sprint Review)
  + Documentación Final del Proyecto
  + Evaluación Retrospectiva del Proyecto (Sprint Retrospective)
  + Lecciones Aprendidas y Cierre Formal del Proyecto
  + Daily Scrum Meetings

Este cronograma detalla las actividades de cada sprint, y las fechas clave del desarrollo del software, asegurando una gestión clara y organizada del proyecto según la metodología Scrum.

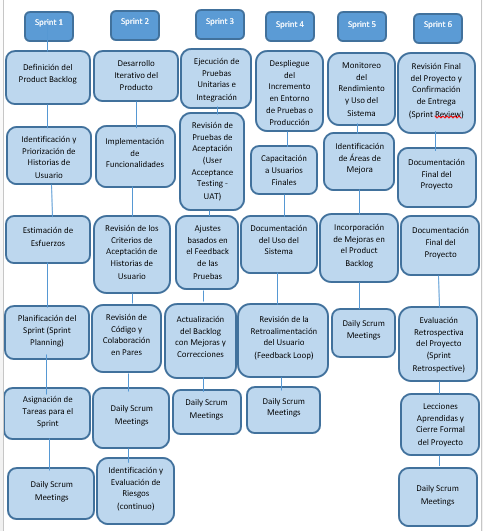
# 

# 

# 

# **Diagrama EDT**

Este diagrama representa un EDT (Estructura de Desglose de Trabajo) para el proyecto Sistema Unidad Territorial. El EDT está organizado en seis fases principales, cada una con sus actividades específicas detalladas. A continuación, se proporciona una explicación breve de cada fase.



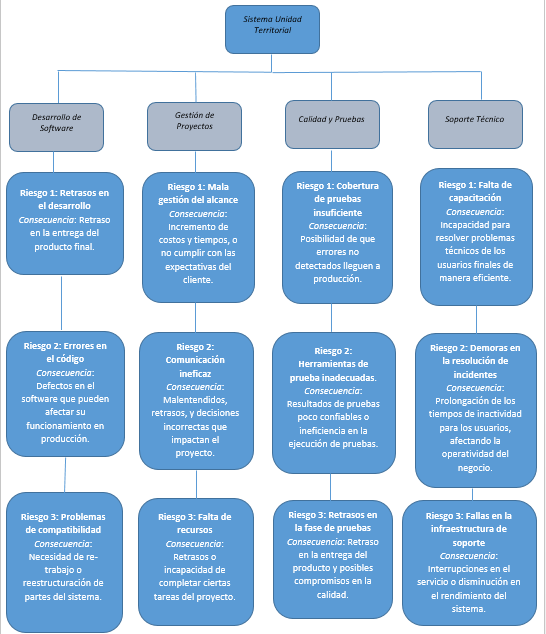
# 

# 

# 

# **Diagrama RBS**

El siguiente listado de riesgos ha sido identificado y clasificado según el Registro de Riesgos (RBS) para el proyecto de Sistema Unidad Territorial en la comuna de San Bernardo. Estos riesgos se representarán en el siguiente diagrama para facilitar su visualización y gestión. A continuación, se proporciona una explicación breve de cada área y riesgo asociado a cada una de estas.



# 

# 

# **Áreas involucradas:**

**1. Desarrollo de Software:**

El área de Desarrollo de Software se encarga de la construcción, programación y codificación del sistema que está siendo desarrollado. Esto incluye desde la planificación técnica hasta la implementación de funcionalidades específicas, utilizando diversos lenguajes de programación, frameworks y herramientas tecnológicas. Los desarrolladores trabajan en la creación de módulos, interfaces y bases de datos, asegurándose de que el software funcione correctamente de acuerdo con los requerimientos y estándares definidos.

***Responsabilidades principales:***

1. Implementación de las funcionalidades del software.
2. Integración con otros sistemas o servicios externos.
3. Corrección de errores y optimización del código.
4. Actualización y mantenimiento del software.

**2. Gestión de Proyectos:**

El área de Gestión de Proyectos se encarga de planificar, coordinar y supervisar el progreso del proyecto desde su inicio hasta su cierre. Los gestores de proyectos trabajan para asegurar que todas las fases del proyecto se ejecuten dentro del presupuesto, el alcance y los plazos establecidos. La gestión de proyectos incluye la administración de recursos, la gestión de riesgos, la comunicación con stakeholders y la gestión del equipo para cumplir con los objetivos del proyecto.

***Responsabilidades principales:***

1. Planificación y organización del cronograma del proyecto.
2. Asignación de recursos humanos, materiales y financieros.
3. Supervisión del avance del proyecto y mitigación de riesgos.
4. Comunicación con las partes interesadas y aseguramiento de la entrega de valor.

**3. Calidad y Pruebas:**

El área de Calidad y Pruebas tiene como objetivo garantizar que el software desarrollado cumpla con los estándares de calidad requeridos antes de ser lanzado a producción. Esto se logra a través de la ejecución de pruebas, incluyendo pruebas unitarias, de integración, de rendimiento, de usabilidad y pruebas de aceptación por parte del usuario. Los equipos de pruebas también identifican y documentan defectos o errores en el software, asegurándose de que estos se solucionen antes de la entrega.

***Responsabilidades principales:***

1. Diseñar e implementar planes de prueba.
2. Realizar pruebas exhaustivas en cada iteración del software.
3. Identificar, reportar y gestionar la corrección de errores.
4. Asegurar que el software cumpla con los requisitos funcionales y no funcionales.

**4. Soporte Técnico:**

El área de Soporte Técnico se encarga de brindar asistencia a los usuarios y equipos en el uso y mantenimiento del software una vez que ha sido implementado. Este equipo ofrece soluciones rápidas a problemas técnicos, asegura que el sistema funcione de manera óptima y gestiona las solicitudes de ayuda de los usuarios. Además, el soporte técnico puede encargarse del mantenimiento de la infraestructura que soporta el software, como servidores y redes.

***Responsabilidades principales:***

1. Brindar soporte post-implementación a usuarios y equipos internos.
2. Resolver problemas técnicos y asegurar el funcionamiento continuo del software.
3. Monitorear el rendimiento del sistema y gestionar la infraestructura.
4. Proveer capacitaciones a usuarios y documentación técnica relevante.

### 

### 

# **Riesgo asociado a cada área**

### **1. Desarrollo de Software**

Riesgo 1: Retrasos en el desarrollo  
Causa: Subestimación del tiempo necesario para implementar funcionalidades complejas o integración de sistemas.  
Consecuencia: Retraso en la entrega del producto final.

Riesgo 2: Errores en el código  
Causa: Falta de pruebas unitarias o de integración adecuadas durante el desarrollo.  
Consecuencia: Defectos en el software que pueden afectar su funcionamiento en producción.

Riesgo 3: Problemas de compatibilidad  
Causa: Incompatibilidad entre componentes del software o tecnologías utilizadas.  
Consecuencia: Necesidad de re-trabajo o reestructuración de partes del sistema.

### 

### 

### **2. Gestión de Proyectos**

Riesgo 1: Mala gestión del alcance  
Causa: Cambios frecuentes en los requerimientos del proyecto sin una adecuada gestión del alcance.  
Consecuencia: Incremento de costos y tiempos, o no cumplir con las expectativas del cliente.

Riesgo 2: Comunicación ineficaz  
Causa: Falta de comunicación clara entre los miembros del equipo y con los stakeholders.  
Consecuencia: Malentendidos, retrasos, y decisiones incorrectas que impactan el proyecto.

Riesgo 3: Falta de recursos  
Causa: Mala planificación de la asignación de recursos o falta de disponibilidad de recursos clave.  
Consecuencia: Retrasos o incapacidad de completar ciertas tareas del proyecto.

### **3. Calidad y Pruebas**

Riesgo 1: Cobertura de pruebas insuficiente  
Causa: Falta de tiempo o recursos para ejecutar pruebas exhaustivas.  
Consecuencia: Posibilidad de que errores no detectados lleguen a producción.

Riesgo 2: Herramientas de prueba inadecuadas  
Causa: Uso de herramientas de pruebas que no se ajustan a las necesidades del proyecto.  
Consecuencia: Resultados de pruebas poco confiables o ineficiencia en la ejecución de pruebas.

Riesgo 3: Retrasos en la fase de pruebas  
Causa: Subestimación del tiempo necesario para pruebas o problemas inesperados durante las mismas.  
Consecuencia: Retraso en la entrega del producto y posibles compromisos en la calidad.

### **4. Soporte Técnico**

Riesgo 1: Falta de capacitación  
Causa: Los equipos de soporte no están adecuadamente capacitados en el nuevo software.  
Consecuencia: Incapacidad para resolver problemas técnicos de los usuarios finales de manera eficiente.

Riesgo 2: Demoras en la resolución de incidentes  
Causa: Recursos insuficientes o falta de procedimientos de escalamiento claros.  
Consecuencia: Prolongación de los tiempos de inactividad para los usuarios, afectando la operatividad del negocio.

Riesgo 3: Fallas en la infraestructura de soporte  
Causa: Problemas en la infraestructura tecnológica que soporta el software, como servidores o redes.  
Consecuencia: Interrupciones en el servicio o disminución en el rendimiento del sistema.

# **Actores y responsables**

### **1. Product Owner (PO)**

El Product Owner es el representante de los intereses de la Junta de Vecinos. Este rol se encarga de gestionar y priorizar el Product Backlog, asegurándose de que el equipo esté trabajando en las tareas de mayor valor para los usuarios.

Responsabilidades:

1. Definir y priorizar las historias de usuario según las necesidades de la Junta de Vecinos.
2. Maximizar el valor del producto desarrollado.
3. Asegurar que el equipo tenga clara la visión del producto y los requisitos.
4. Aprobar o rechazar las funcionalidades entregadas en cada sprint.
5. Recoger feedback de los vecinos y otras partes interesadas.

### **2. Scrum Master**

Descripción: El Scrum Master es el facilitador del equipo, asegurando que se sigan los principios de Scrum y eliminando los impedimentos que puedan afectar al progreso del proyecto.

Responsabilidades:

1. Facilitar las reuniones de Scrum (Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective).
2. Eliminar impedimentos que afecten el progreso del equipo.
3. Asegurarse de que el equipo de desarrollo trabaje de manera eficiente y enfocada.
4. Promover la mejora continua y buenas prácticas dentro del equipo.
5. Asegurar que el equipo siga los principios ágiles.

### 

### **3. Equipo de Desarrollo**

Descripción: Este equipo está compuesto por desarrolladores, diseñadores, testers y especialistas en seguridad. Son los responsables de entregar el producto en cada sprint.

Responsabilidades:

1. Desarrollar las funcionalidades y tareas según lo definido en el Sprint Backlog.
2. Diseñar, programar y probar el software que se implementará.
3. Colaborar para entregar un incremento del producto al final de cada sprint.
4. Realizar pruebas unitarias y de integración para asegurar la calidad del producto.
5. Asegurar que las funcionalidades cumplan con los criterios de aceptación definidos por el Product Owner.

### **4. Equipo de Pruebas (Testers)**

Descripción: Parte del equipo de desarrollo, los testers son responsables de garantizar que las funcionalidades cumplan con los requisitos establecidos y sean de alta calidad antes de su entrega.

Responsabilidades:

1. Realizar pruebas funcionales, unitarias, de integración y de usuario para detectar errores.
2. Asegurar que el producto cumple con los criterios de aceptación.
3. Reportar errores o problemas en el software y colaborar en su resolución.
4. Asegurar que el producto sea seguro y estable para su uso por parte de los vecinos.

### 

### 

### **5. Vecinos y Usuarios Finales**

Descripción: Son los principales beneficiarios del proyecto. Si bien no forman parte del equipo directamente, juegan un papel clave al proporcionar feedback y utilizar el sistema que se desarrollará.

Responsabilidades:

1. Proporcionar feedback sobre las funcionalidades desarrolladas.
2. Participar en la validación del producto durante la fase de pruebas de usuario (User Acceptance Testing - UAT).
3. Utilizar el sistema una vez implementado.

### **6. Equipo de Soporte Técnico**

Descripción: Este equipo se encarga de dar soporte una vez que el producto está en producción. Aseguran el correcto funcionamiento del sistema y resuelven problemas técnicos que puedan surgir.

Responsabilidades:

1. Proveer asistencia técnica en el despliegue del sistema.
2. Monitorear el rendimiento y solucionar problemas que puedan surgir.
3. Mantener el sistema actualizado y asegurar su disponibilidad para los vecinos.
4. Gestionar el mantenimiento preventivo y correctivo.

### 

### **7. Líder de la Junta de Vecinos (Stakeholder Principal)**

**Descripción:** El líder de la Junta de Vecinos actúa como un vínculo entre la comunidad y el equipo del proyecto. Asegurar que las expectativas y necesidades de los vecinos se transmitan correctamente.

Responsabilidades:

1. Representar los intereses de la comunidad durante el desarrollo del proyecto.
2. Aprobar las fases del proyecto, especialmente en términos de funcionalidad y cumplimiento de requisitos.
3. Colaborar con el Product Owner para definir las prioridades del proyecto.
4. Facilitar la comunicación entre los vecinos y el equipo de desarrollo.

# **Identificación de los riesgos**

Veremos los riesgos de una manera organizada del plan de gestión de riesgos de la junta de vecinos de la comuna de San Bernardo. Estos son algunos de los campos que se ven en la planilla de identificación de riesgos.

**1. Riesgo/Evento de Riesgo:** Sobrecarga del servidor web

**Descripción y Comentarios:** El servidor puede no soportar la carga de usuarios simultáneos, lo que podría llevar a caídas o interferencias.

**Mitigación o Plan de Contingencia**: Aumentar la capacidad de un solo servidor, por ejemplo, añadiendo más memoria RAM, CPUs más potentes, o mayor almacenamiento y disposición del servicio.

**2. Riesgo/Evento de Riesgo:** Costos adicionales por licencias de software

**Descripción y Comentarios:** Se podrían requerir licencias adicionales, lo que impactaría en el presupuesto.

**Mitigación o Plan de Contingencia:** Revisar y ajustar el presupuesto para incluir posibles costos adicionales

**3. Riesgo/Evento de Riesgo:** Fallo en la integración de la base de datos

**Descripción y Comentarios:** Se podrían requerir licencias adicionales, lo que impactaría en el presupuesto.

**Mitigación o Plan de Contingencia**: Realizar pruebas de integración previas

**4. Riesgo/Evento de Riesgo:** Pérdida de datos sensibles

**Descripción y Comentarios:** La solución podría ser vulnerable a ataques que resulten en la pérdida de datos.

**Mitigación o Plan de Contingencia**: Implementar protocolos de seguridad robustos y pruebas de penetración regulares

**5. Riesgo/Evento de Riesgo:** Desacuerdos entre las partes interesadas

**Descripción y Comentarios:** Las partes interesadas podrían no ponerse de acuerdo en los requisitos del proyecto.

**Mitigación o Plan de Contingencia:** Facilitar talleres de alineación y establecer un mediador

**6. Riesgo/Evento de Riesgo:** Cambios en los requerimientos del proyecto

**Descripción y Comentarios:** Las solicitudes adicionales de funcionalidades pueden desviar el proyecto.

**Mitigación o Plan de Contingencia:** Implementar un proceso riguroso de control de cambios

**7. Riesgo/Evento de Riesgo:** Falta de capacitación del personal

**Descripción y Comentarios:** El personal no capacitado podría cometer errores que afecten la operatividad.

**Mitigación o Plan de Contingencia:** Planificar y ejecutar un programa de capacitación extensivo.

**8. Riesgo/Evento de Riesgo:** Baja adopción del sistema por parte de los vecinos

**Descripción y Comentarios:** Los vecinos pueden resistirse a usar la nueva plataforma, afectando su éxito.

**Mitigación o Plan de Contingencia**: Realizar campañas de sensibilización y talleres de formación para los vecinos.

**9. Riesgo/Evento de Riesgo:** Interrupciones en el servicio debido a mantenimientos programados

**Descripción y Comentarios**: Mantenimientos programados pueden interrumpir el acceso al sistema, afectando la continuidad de operaciones.

**Mitigación o Plan de Contingencia:** Planificar mantenimientos fuera de horas pico y comunicar con antelación.

**10. Riesgo/Evento de Riesgo:** Falta de retorno de inversión

**Descripción y Comentarios:** El costo del sistema podría no justificarse si no se obtiene el retorno esperado.

**Mitigación o Plan de Contingencia:** Realizar un análisis exhaustivo de costo-beneficio antes de la implementación

**11. Riesgo/Evento de Riesgo:** Pérdida de datos debido a fallas del sistema

**Descripción y Comentarios:** La falta de copias de seguridad adecuadas podría resultar en la pérdida de datos importantes.

**Mitigación o Plan de Contingencia:** Implementar sistemas de respaldo automáticos y realizar pruebas de recuperación

**12. Riesgo/Evento de Riesgo:** Errores en la generación de certificados de residencia

**Descripción y Comentarios:** Revisar y probar exhaustivamente la lógica

**Mitigación o Plan de Contingencia:** Implementar pruebas unitarias y de integración

**13. Riesgo/Evento de Riesgo:** Fallos de seguridad en la autenticación de usuarios

**Descripción y Comentarios:** Implementar doble autenticación y encriptación robusta

**Mitigación o Plan de Contingencia:** Realizar auditorías de seguridad periódicas.

**14. Riesgo/Evento de Riesgo**: Desmotivación del equipo de trabajo

**Descripción y Comentarios:** Monitorear la carga de trabajo y ajustar

**Mitigación o Plan de Contingencia**: Organizar actividades de team-building para que los miembros interactúen en grupo con el fin de mejorar sus capacidades.

**15. Riesgo/Evento de Riesgo:** Problemas en la gestión de la calidad

**Descripción y Comentarios:** Implementar un sistema de gestión de calidad durante el desarrollo

**Mitigación o Plan de Contingencia**: Realizar revisiones de calidad regulares

**16. Riesgo/Evento de Riesgo:** Falta de feedback oportuno por parte del cliente

**Descripción y Comentarios:** Establecer expectativas claras sobre tiempos de respuesta

**Mitigación o Plan de Contingencia:** Implementar checkpoints regulares con el cliente

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# **Costos del proyecto**

Los gastos están organizados según los valores del mercado y las necesidades del proyecto a continuación desglosamos cada uno de los gastos de manera detallada para transparentar los costos del proyecto.

El personal es el componente más significativo del presupuesto. A continuación se detalla cada cargo de los especialistas capacitados para el desarrollo de sistema de unidad territorial y su sueldo mensual estimado

Dado que el proyecto es sin fines de lucro, no se incluirán márgenes de ganancia no obstante se mantiene un 20% adicional para la mitigación de riesgos.

A continuación, se presenta el desglose de los costos asociados al proyecto considerando los salarios, herramientas, y otros gastos.

**1. Costos del Equipo (96 días)**

| **Cargo** | **Costo proyecto 96 Días** | **Total** |
| --- | --- | --- |
| Product Owner/PM/Equipo de desarrollo | CLP $ 5.440.000 |  |
| Equipo de desarrollo | CLP $ 3.840.000 |  |
| Scrum Master/Equipo de desarrollo | CLP $ 5.120.000 |  |
|  |  | CLP $ 14.400.000 |

**2. Costos de Herramientas**

| **Herramienta** | **Valor** |
| --- | --- |
| Computadoras y Kits (3 unidades): | CLP $2.400.000 |
| Software y Herramientas de Desarrollo (licencias, servicios en la nube)  (para herramientas de desarrollo como IDE, Firebase, y gestión de proyectos) | CLP $500.000 |

**3. Gastos Básicos**

| **Gastos básicos** | **valor** |
| --- | --- |
| Conexión a Internet, electricidad y otros gastos operativos:  Estimación mensual por 3.2 meses para un equipo de trabajo remoto: | CLP $ 300.000 |

**4. Capacitación**

| **Capacitación** | **Valor** |
| --- | --- |
| Cursos o talleres sobre tecnologías específicas (Ionic, Firebase, etc.): | CLP $300.000 |

**5. Mitigación de Riesgos (20%)**

| **Cargo** | **Valor** | **Total** |
| --- | --- | --- |
| Salarios del Equipo | 14.400.000 |  |
| Herramientas (computadores) | 2.400.000 |  |
| Software y herramientas | 500.000 |  |
| Gastos Básicos | 300.000 |  |
| Capacitación | 300.000 |  |
| Mitigación de Riesgos (20%) |  | CLP $3380.000 |
| Costo total del proyecto |  | CLP$**$ 21.280.000** |

**Resumen de Costos Totales**

**Conceptos:**

* Salarios del Equipo

$14.400.000

* Herramientas (computadores)

$2.400.000

* Software y herramientas

$500.000

* Gastos Básicos

$300.000

* Capacitación

$300.000

* Mitigación de Riesgos (20%)

$3.380.000

* Total del Proyecto

**$21.280.000**

# **Conclusión**

El Plan de Gestión de Riesgos para el desarrollo del Sistema de Unidad Territorial destinado a la Junta de Vecinos de San Bernardo es una herramienta fundamental para garantizar el éxito del proyecto, minimizando posibles impactos negativos que puedan comprometer su calidad, tiempo de entrega o costos.

Tras el análisis de los riesgos identificados, hemos establecido un plan robusto que abarca tanto riesgos cuantitativos como cualitativos, los cuales se organizan por su probabilidad e impacto en el proyecto. Las áreas críticas incluyen la sobrecarga del servidor, el manejo de costos adicionales por licencias, la pérdida de datos sensibles, y los fallos en la integración de la base de datos, entre otros. Además, se han propuesto estrategias de mitigación para cada uno de los riesgos, basadas en la implementación de protocolos de seguridad, mejoras en la infraestructura tecnológica, y un proceso ágil de control de cambios.

El proyecto ha sido diseñado para ser viable y sostenible dentro del presupuesto asignado de CLP $21.280.00 lo que incluye una planificación financiera ajustada para la gestión de riesgos y la ejecución efectiva de las tareas de desarrollo, capacitación y soporte técnico. Los costos asociados a la implementación de medidas preventivas y correctivas han sido optimizados, priorizando la seguridad de la información, la estabilidad del sistema y la satisfacción de los usuarios finales.

Con este plan de riesgos en marcha, se espera que el proyecto no solo logre cumplir con los objetivos planteados, sino que también se convierta en un modelo de eficiencia y transparencia en la administración de los recursos de la comunidad. Al adoptar una metodología ágil como Scrum, el equipo de desarrollo podrá responder de manera rápida y efectiva a los imprevistos, asegurando entregas continuas de valor y una solución tecnológica adaptable a las necesidades de la junta de vecinos.

En resumen, el Sistema de Unidad Territorial no solo digitalizará la gestión comunitaria de la junta de vecinos, sino que también fortalecerá la transparencia, la participación ciudadana y el fortalecimiento del tejido social en la comuna de San Bernardo, garantizando un impacto positivo y sostenible en la comunidad.