**Documento de requerimientos de software**

***GVVI (Gestión vecinal Villa Inca)***

***Fecha: 15-09-2024***

**Información del Proyecto**

| Empresa / Organización | Duoc UC |
| --- | --- |
| Proyecto | Sistema de Gestión Vecinal |
| Fecha de preparación | 01-09-2024 |
| Cliente | Junta de Vecinos Villa Inca Melipilla |
| Patrocinador principal | Dios |
| Gerente / Líder de Proyecto | Miguel Donoso |
| Gerente / Líder de Análisis de negocio y requerimientos | Diego Canales |

**Propósito**

Este documento tiene como objetivo definir los requerimientos de software para el Proyecto APT, el cual desarrollará un sistema web que permitirá la gestión eficiente de la Junta de Vecinos de la Villa Inca en Melipilla. Se describen los requisitos funcionales y no funcionales que deberá cumplir el software, así como las tecnologías y herramientas necesarias para su desarrollo e implementación.

**Alcance**

El sistema abarcará la digitalización de procesos clave de la junta de vecinos, como la inscripción de residentes, la emisión de certificados, la gestión de proyectos vecinales y la implementación de un sistema de notificaciones. Se espera que el sistema esté disponible tanto para usuarios de computadoras como dispositivos móviles y cumpla con los estándares de seguridad y accesibilidad.

**Requerimientos Funcionales**

**REQ-1: Inscripción de Vecinos**

**Descripción:** El sistema debe permitir a los vecinos inscribirse a través de un portal web.

**Prioridad:** Alta.

**Acciones del Usuario y Resultado Esperado:**

* El usuario accede al portal web.
* Completa el formulario de inscripción con su nombre, dirección, correo electrónico y número de identificación.
* El sistema valida los datos ingresados y, si son correctos, envía una solicitud a la junta de vecinos para su aprobación.
* El usuario recibe una notificación confirmando que su solicitud ha sido enviada correctamente.

**Requerimientos Funcionales:**

* El sistema debe permitir la creación de un nuevo perfil para los vecinos.
* Validación de datos duplicados para evitar múltiples registros de un mismo usuario.
* Verificación de datos antes de enviar la solicitud a la junta de vecinos.
* Notificación al usuario una vez que la solicitud ha sido enviada para su aprobación.

**REQ-2: Emisión de Certificados de Residencia**

**Descripción:** El sistema debe permitir a los vecinos solicitar y recibir certificados de residencia de manera digital.

**Prioridad:** Alta.

**Acciones del Usuario y Resultado Esperado:**

* El usuario inicia sesión en el sistema.
* Solicita un certificado de residencia a través de una opción en el menú principal.
* El sistema genera el certificado en formato PDF y lo envía al administrador para su revisión.
* El usuario recibe una notificación cuando el certificado ha sido aprobado y está disponible para su descarga.

**Requerimientos Funcionales:**

* Generación automática del certificado en formato PDF.
* Función de revisión y aprobación de la solicitud por parte de la junta de vecinos.
* Envío de notificaciones automáticas al usuario cuando el certificado esté listo para ser descargado.
* Almacenamiento seguro de los certificados emitidos para futuras consultas.

**REQ-3: Inscripción de Vecinos**

**Descripción:** El sistema debe permitir a los vecinos postular proyectos comunitarios y a la junta de vecinos revisar y aprobar dichas propuestas.

**Prioridad:** Media.

**Acciones del Usuario y Resultado Esperado:**

* El usuario accede al portal y selecciona la opción para postular un proyecto comunitario.
* Completa un formulario con la descripción, objetivos y justificación del proyecto.
* El sistema envía la postulación a la junta de vecinos para su revisión.
* El usuario recibe una notificación con la respuesta de la junta, aprobando o rechazando el proyecto.

**Requerimientos Funcionales:**

* El sistema debe permitir la creación de nuevas postulaciones de proyectos comunitarios.
* Revisión y aprobación o rechazo de los proyectos por parte de la junta de vecinos.
* Envío de notificaciones automáticas sobre el estado de la postulación.
* Almacenamiento de las postulaciones para su consulta posterior por la junta y los vecinos.

**REQ-4: Notificaciones Automáticas**

**Descripción:** El sistema debe enviar notificaciones automáticas a los vecinos por correo electrónico y WhatsApp para mantenerlos informados.

**Prioridad:** Alta.

**Acciones del Usuario y Resultado Esperado:**

* El usuario se registra o solicita un servicio dentro del sistema (por ejemplo, un certificado o una inscripción a un evento).
* El sistema envía automáticamente una notificación al correo electrónico y/o WhatsApp del usuario informando sobre el estado de su solicitud o recordando algún evento próximo.
* El usuario recibe la notificación de forma inmediata o programada según el evento.

**Requerimientos Funcionales:**

* Implementación de un sistema de notificaciones por email y WhatsApp.
* Configuración de notificaciones personalizadas por la junta de vecinos.
* Envío de notificaciones automáticas para eventos, reuniones y estados de solicitudes (certificados, inscripciones, etc.).
* Historial de notificaciones para su consulta posterior.

**REQ-5: Notificaciones Automáticas**

**Descripción:** El sistema debe permitir a la junta de vecinos publicar noticias y eventos comunitarios, y a los vecinos inscribirse en dichas actividades.

**Prioridad:** Media.

**Acciones del Usuario y Resultado Esperado:**

* La junta de vecinos publica una nueva noticia o evento en el portal.
* El usuario navega por la sección de noticias o calendario de eventos y selecciona una actividad de interés.
* El usuario se inscribe en el evento, y el sistema confirma su participación a través de una notificación.

**Requerimientos Funcionales:**

* Publicación de noticias y anuncios comunitarios por parte de la junta de vecinos.
* Gestión de un calendario de eventos con opción de inscripción para los vecinos.
* Confirmación de inscripción a los eventos mediante notificaciones.
* Visualización de noticias y eventos en una sección dedicada dentro del portal.

**Requerimientos No Funcionales**

### Usabilidad

* La interfaz debe ser intuitiva y fácil de usar, permitiendo que personas sin conocimientos técnicos puedan gestionar sus interacciones con la junta de vecinos.
* El diseño del sistema debe ser responsive, adaptándose a diferentes dispositivos, como computadoras, tablets y smartphones.

### Seguridad

* El sistema debe garantizar la autenticación segura de los usuarios mediante el uso de credenciales (usuario y contraseña).
* Los datos personales y sensibles deben estar encriptados tanto en tránsito como en reposo.
* El acceso a la información administrativa debe estar restringido a miembros autorizados de la junta de vecinos.
* Deben implementarse políticas de control de acceso basadas en roles (administradores y usuarios comunes).

### Escalabilidad

* El sistema debe ser capaz de soportar el crecimiento en el número de usuarios sin comprometer el rendimiento.
* La arquitectura debe permitir la fácil adición de nuevas funcionalidades sin afectar las actuales.

### Rendimiento

* El tiempo de carga de las páginas debe ser inferior a 3 segundos bajo condiciones normales de uso.
* El sistema debe ser capaz de manejar múltiples solicitudes simultáneamente sin degradar el servicio.

### Mantenibilidad

* El código debe estar bien documentado para facilitar futuras actualizaciones y mantenimientos.
* El sistema debe ser diseñado de manera modular para permitir la corrección de errores y la adición de nuevas funcionalidades de manera independiente.

### 

### Disponibilidad

* El sistema debe estar disponible 24/7, con un tiempo de inactividad mínimo garantizado por un servicio de alojamiento confiable.
* Se debe implementar un sistema de backup regular para garantizar la recuperación de datos en caso de fallos.

**Tecnologías utilizadas**

### Lenguajes de Programación

* HTML, CSS, JavaScript para el desarrollo frontend.
* PHP (framework Laravel) para el backend.

### Bases de Datos

* MySQL para la gestión de la base de datos.

### Servicios de Alojamiento

* El sistema se alojará en un servidor web que soporte PHP y bases de datos MySQL.

### Librerías y Herramientas

* Laravel para el desarrollo backend.
* Bootstrap o una librería CSS similar para la interfaz responsive.
* API de WhatsApp o servicio de mensajería para las notificaciones.