

**PROYECTO FINAL DESARROLLO DE SOFTWARE 1**  
**ASAMBLEA DE VOTACIONES**

**INTEGRANTES**

**JHON ANDERSON LUNA**  
**JHONATAN ANDRÉS CUERO**  
**JUAN ESTEBAN PEREIRA**  
**ALEJANDRO GONZALEZ SILVA**  
**MIGUEL ÁNGEL PINEDA**

**DOCENTE**

**GUSTAVO ADOLFO OSORIO**

**TECNOLOGÍA EN DESARROLLO DE SOFTWARE**  
**UNIVERSIDAD DEL VALLE**  
**2024**

## Contenido

Introducción .....	3
Antecedentes .....	4
Justificación.....	5
Alcance del proyecto .....	6
Visión del proyecto .....	8
Marco Tecnológico .....	9
Backend .....	9
Frontend .....	9
Bases de datos .....	9
Servidores y Almacenamiento .....	9
Modelado.....	10
Casos de uso .....	10
Diagrama de casos de uso.....	19
Diagrama de flujo .....	19
Diagramas de actividad.....	20
Diseño de la base de datos.....	20
Modelo conceptual .....	20
Modelo lógico .....	21
Esquema físico.....	21
Conclusiones .....	22
Uso de metodologías ágiles .....	22
Dificultades encontradas .....	23
Tecnológicas .....	23
Metodológicas .....	23
Del equipo de trabajo.....	24

# Introducción

Este proyecto tiene como objetivo desarrollar una plataforma digital que permita la gestión de asambleas y la realización de votaciones de manera eficiente, segura y accesible para todos los participantes. En un entorno donde la toma de decisiones colectivas es esencial para el funcionamiento óptimo de cualquier organización, este sistema está diseñado para modernizar y optimizar el proceso tradicional de votación.

A través del uso de tecnologías avanzadas, el sistema proporciona funcionalidades clave, como el registro de usuarios, la creación y votación de propuestas, y la visualización de resultados en tiempo real. Este enfoque no solo mejora la seguridad y la confiabilidad del proceso de votación, sino que también facilita una mayor participación. De esta manera, se adapta a las necesidades de organizaciones con miembros dispersos o que enfrentan restricciones para reunirse físicamente, asegurando que todos los participantes puedan contribuir efectivamente a las decisiones colectivas.

## **Antecedentes**

En el contexto de las organizaciones modernas, la toma de decisiones colectivas es una práctica esencial. Las asambleas de votación se han utilizado históricamente para decidir sobre asuntos importantes, permitiendo a los miembros expresar sus opiniones y participar activamente en la dirección de la organización.

En la actualidad, muchos sistemas de votación tradicionales enfrentan varios desafíos. Los métodos manuales de votación, como el uso de papeletas en papel, son susceptibles a errores humanos y fraudes. Además, estos métodos suelen ser lentos y costosos, requiriendo una cantidad significativa de recursos para contar y verificar los votos.

Con el avance de la tecnología, han surgido soluciones electrónicas o virtuales para la votación. Estos sistemas de votación prometen mejorar la eficiencia del proceso de votación. Sin embargo, muchos de estos sistemas también enfrentan críticas y desafíos relacionados con la seguridad y la privacidad.

## Justificación

Se decidió realizar este proyecto debido a que se consideró un reto alcanzable según los conocimientos previos. Se consideraba necesario y posible realizar un sistema que gestione asambleas y permitiera las votaciones en esta, ya que existe una notable diferencia entre las votaciones de una asamblea de forma presencial a unas votaciones de forma virtual. Entre las ventajas que presenta realizar las votaciones haciendo uso de la tecnología tenemos:

**Mejor Seguridad:** Los sistemas de votación tradicionales enfrentan desafíos de seguridad, con riesgos de fraudes y errores humanos. Un sistema virtual ofrece mayor seguridad al incorporar encriptación de datos y otras medidas avanzadas para garantizar la integridad de los votos y la precisión de los resultados.

**Confiabilidad:** La confianza en el sistema de votación es crucial para la legitimidad de una asamblea. Un sistema de votación virtual puede aumentar la transparencia al permitir auditorías y verificaciones en tiempo real, lo que no solo aumenta la confianza de los votantes, sino que también facilita la detección y corrección rápida de problemas.

**Accesibilidad y Participación:** En organizaciones grandes o dispersas por todo el mundo, reunir a todos los miembros en un solo lugar para votar puede ser muy complicado. Un sistema de votación virtual permite a los usuarios participar desde cualquier lugar con acceso a internet, lo que puede aumentar significativamente la participación y asegurar que todos sean escuchados.

**Situaciones de Emergencia:** La pandemia de COVID-19 ha demostrado la importancia de estar preparados para situaciones de emergencia que pueden impedir las reuniones presenciales. Un sistema de votación en línea permite a las organizaciones continuar con sus procesos decisionales esenciales, incluso en circunstancias adversas, asegurando la continuidad operativa.

## Alcance del proyecto

**Objetivo del Proyecto:** Desarrollar un sistema de votaciones completo para asambleas, que permita a los usuarios registrarse, participar en votaciones de diferentes propuestas, visualizar los resultados de las votaciones realizadas, y buscar propuestas para votar. El sistema debe proporcionar todas las funcionalidades necesarias para gestionar las votaciones de manera eficiente y segura.

### Funcionalidades Principales:

**Registro de Usuarios:** Los usuarios podrán registrarse en el sistema proporcionando información básica como nombre, apellido, correo electrónico y contraseña. Se verificará la validez del correo electrónico durante el proceso de registro.

**Autenticación y Sesiones de Usuario:** El sistema permitirá a los usuarios iniciar sesión utilizando su correo electrónico y contraseña. Se implementará un sistema de sesiones para mantener la sesión del usuario activa durante su interacción con el sistema.

**Votación en Propuestas:** Los usuarios podrán votar en diferentes propuestas presentadas durante las asambleas. Se proporcionará una interfaz intuitiva para que los usuarios elijan entre las opciones de votación disponibles. Se deben implementar mecanismos de seguridad para garantizar que cada usuario pueda emitir un solo voto por propuesta.

**Visualización de Resultados:** Después de votar, los usuarios podrán ver los resultados de la votación en tiempo real o en un momento posterior. Se presentarán los resultados de manera clara y comprensible para los usuarios.

**CRUD de Propuestas de Votación:** Los usuarios del sistema podrán crear, leer, actualizar y eliminar (CRUD) propuestas para votación dentro de un tema de una asamblea.

Se incluirá un panel de administración donde los administradores podrán gestionar las asambleas.

**Conexión a Base de Datos:** Se establecerá una conexión a una base de datos para almacenar la información de las votaciones y los usuarios.

Se implementará un diseño de base de datos eficiente y escalable para garantizar un rendimiento óptimo del sistema.

**Funcionalidad de Búsqueda:** Se debe incorporar un buscador que permita a los usuarios encontrar rápidamente las propuestas de votación en base a ciertos criterios de búsqueda.

El buscador debe ser fácil de usar y proporcionar resultados relevantes y precisos.

El sistema estará diseñado para manejar un número variable de usuarios y propuestas de votación, con la capacidad de escalar según sea necesario.

**Entregables:**

- Sistema funcional de votaciones para asambleas con todas las características mencionadas.
- Documentación detallada del sistema, incluyendo diagramas de arquitectura (casos de uso y diagrama de casos de uso, diagrama de flujo de datos y diagramas de actividad) diseño de base de datos (modelo conceptual, modelo lógico, esquema físico).
- Código fuente del sistema, debidamente comentado y organizado.

**Cronograma:**

Para el proyecto se utilizó la metodología ágil Scrum y se asignaron roles a cada miembro del equipo de trabajo: 1 Producto Owner (Jhon Anderson Luna), 1 Scrum Master (Jhonatan Andrés Cuero) y 3 Developers (Miguel Ángel Pineda, Juan Esteban Pereira, Alejandro Gonzalez Silva).

Cada Sprint Planning contaba con una duración de una semana. Se realizaron un total de 4 Sprint Planning en donde se realizaban Daily Sprints (Reuniones diarias pequeñas donde se contaban los avances del proyecto), al finalizar cada semana, se registraban los avances, lo que no se completo y lo que se debe adaptar del proyecto en el Sprint Review, en el Sprint Retrospective se registraba lo que salió bien, lo que salió mal y lo que se podía mejorar.

## **Visión del proyecto**

Nuestro principal objetivo con este proyecto es desarrollar una plataforma de asamblea de votaciones que disponga de un sistema seguro y eficiente para facilitar la toma de decisiones en diferentes contextos organizativos. La visión a largo plazo es transformar la manera en que se llevan a cabo las votaciones, promoviendo la transparencia, la participación y la confianza en el proceso de votación.

### **Objetivos:**

**Desarrollar una Plataforma Intuitiva:** Crear una interfaz de usuario intuitiva y fácil de usar que permita a los usuarios de la asamblea participar en las votaciones de manera efectiva, independientemente de su nivel de habilidad técnica.

**Facilitar la Verificación de Resultados:** Diseñar un sistema que permita a los participantes verificar los resultados de las votaciones en todo momento.

**Optimizar la Eficiencia del Proceso:** Minimizar el tiempo y los recursos requeridos para organizar y llevar a cabo votaciones, automatizando tareas repetitivas y simplificando la gestión de datos.

### **Público Objetivo:**

Miembros de la asamblea, incluidos directivos, empleados, accionistas y otros interesados.

Administradores y organizadores de asambleas, responsables de coordinar y gestionar el proceso de votación.



## Marco Tecnológico

### Backend

Se utilizó el lenguaje de programación **PHP** el cual es un lenguaje de programación de propósito general especialmente adecuado para el desarrollo web.

### Frontend

Se utilizó **HTML** y **CSS**, los cuales son un lenguaje de marcado para crear páginas web y un lenguaje para establecer el estilo de las páginas web; respectivamente. Además de pequeños scripts de Javascript para algunas funciones específicas.

### Bases de datos

Se utilizó un sistema de base de datos relacional **MySQL**, gestionada en el servidor con phpMyAdmin, proporcionado por XAMPP.

### Servidores y Almacenamiento

Se utilizó la herramienta de desarrollo XAMPP el cual es un paquete de software gratuito y de código abierto que facilita la creación y administración de servidores web locales. Para el servidor web local XAMPP provee un servidor Apache, el cual es de código abierto, sumamente conocido y práctico para este tipo de proyectos. Todo el proyecto fue manejado a través de la herramienta GitHub que ayudó significativamente con la gestión del proyecto y el historial de versiones, así cada miembro pudo estar al día con las actualizaciones que se realizaron.

# Modelado

## Casos de uso

Campo	Descripción
Nombre	Registro de Usuarios
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de registro de nuevos usuarios en el sistema.
Actores	Usuario
Precondiciones	Ninguna
Flujo Normal	1. El usuario accede a la interfaz de registro. 2. El usuario ingresa sus datos personales: id, nombres, apellidos, edad, cargo, correo, usuario y contraseña. 3. El sistema valida la unicidad del ID y del usuario en la base de datos. 4. El sistema guarda los datos del usuario en la base de datos.
Flujo Alternativo	Ninguno
Poscondiciones	El usuario queda registrado en el sistema.

Nombre	Inicio de Sesión
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de inicio de sesión de un usuario registrado en el sistema.
Actores	Usuario
Precondiciones	El usuario debe estar registrado en el sistema.
Flujo Normal	1. El usuario accede a la interfaz de inicio de sesión. 2. El usuario ingresa su nombre de usuario y contraseña. 3. El sistema verifica la existencia y validez de las credenciales en la base de datos. 4. El sistema permite el acceso al programa si las credenciales son válidas.
Flujo Alternativo	Ninguno
Poscondiciones	El usuario inicia sesión en el sistema.

Nombre	Creación de Asambleas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de creación de una nueva asamblea por parte de un usuario con permisos de administrador.
Actores	Usuario con permisos de administrador
Precondiciones	1. El usuario debe haber iniciado sesión en el sistema.
	2. El usuario debe tener permisos de administrador
	3. El usuario debe tener acceso a la interfaz de creación de asambleas del sistema.
Flujo Normal	1. El usuario accede a la interfaz de creación de asambleas.
	2. El usuario ingresa los datos requeridos para la asamblea: <ul style="list-style-type: none"> <li>• Nombre</li> <li>• Fecha</li> <li>• Hora</li> <li>• Descripción</li> </ul>
	3. El sistema valida los datos ingresados por el usuario: <ul style="list-style-type: none"> <li>• Verifica que se hayan ingresado todos los campos obligatorios.</li> <li>• Valida el formato de los datos (por ejemplo, fecha, hora).</li> </ul>
	4. Si los datos son válidos: <ul style="list-style-type: none"> <li>• El sistema crea la asamblea en la base de datos.</li> <li>• Muestra un mensaje de confirmación al usuario.</li> </ul>
	5. La asamblea queda activa y disponible para que los usuarios puedan crear temas y propuestas.
Flujo Alternativo	1. Si el usuario no ingresa todos los datos requeridos: <ul style="list-style-type: none"> <li>• El sistema muestra un mensaje indicando que faltan campos por completar.</li> <li>• El usuario debe ingresar los datos faltantes y volver a intentar.</li> </ul>
	2. Si el usuario ingresa datos inválidos: <ul style="list-style-type: none"> <li>• El sistema muestra un mensaje indicando el error correspondiente.</li> <li>• El usuario debe corregir los datos e intentar nuevamente.</li> </ul>
Poscondiciones	La asamblea se crea exitosamente en el sistema y queda activa para su uso.

Nombre	Edición y Eliminación de Asambleas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de edición y eliminación de una asamblea por parte de un usuario con permisos de administrador.
Actores	Usuario con permisos de administrador
Precondiciones	1. El usuario debe haber iniciado sesión en el sistema.
	2. El usuario debe tener permisos de administrador.
	3. El usuario debe tener acceso a la interfaz de edición y eliminación de asambleas del sistema.
	4. La asamblea que se desea editar o eliminar debe existir en el sistema.
Flujo Normal	1. El usuario accede a la lista de asambleas del sistema.
	2. El usuario selecciona la asamblea que desea editar.
	3. El sistema muestra la interfaz de edición de la asamblea con los datos actuales de la asamblea.
	4. El usuario modifica los datos de la asamblea según sea necesario (nombre, fecha, hora, descripción, etc.).
	5. El sistema valida los datos modificados por el usuario.
	6. Si los datos son válidos, el sistema actualiza la información de la asamblea en la base de datos y muestra un mensaje de confirmación al usuario.
	7. La asamblea queda actualizada con los nuevos datos.
Flujo Alternativo	Si el usuario cancela la eliminación, el sistema no elimina la asamblea y vuelve a la lista de asambleas.
Poscondiciones	La asamblea se edita o elimina exitosamente en el sistema, según la acción realizada por el usuario.

Nombre	Creación de Temas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de creación de nuevos temas dentro de una asamblea activa.
Actores	Usuario
Precondiciones	El usuario debe haber iniciado sesión y tener los permisos necesarios para crear un tema.
Flujo Normal	1. El usuario accede a la funcionalidad de crear un nuevo tema.
	2. El usuario ingresa los detalles del tema, como título, descripción, etc.
	3. El sistema valida los permisos del usuario para crear un tema.
	4. El sistema guarda los datos del tema en la base de datos.
Flujo Alternativo	Ninguno
Poscondiciones	Se crea un nuevo tema dentro de la asamblea activa.

Nombre	Creación de Propuestas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de creación de nuevas propuestas dentro de una asamblea para un tema específico.
Actores	Usuario
Precondiciones	El usuario debe haber iniciado sesión y tener los permisos necesarios para crear una propuesta.
Flujo Normal	1. El usuario accede a la funcionalidad de crear una nueva propuesta para un tema específico.
	2. El usuario ingresa los detalles de la propuesta, como título y descripción.
	3. El sistema valida los permisos del usuario para crear una propuesta.
	4. El sistema guarda los datos de la propuesta en la base de datos asociada al tema y usuario correspondientes.
Flujo Alternativo	Ninguno
Poscondiciones	Se crea una nueva propuesta dentro del tema especificado.

Nombre	Visualización de Propuestas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de visualización de propuestas creadas por un usuario dentro de una asamblea.
Actores	Usuario
Precondiciones	El usuario debe haber iniciado sesión y tener propuestas creadas en el sistema.
Flujo Normal	1. El usuario accede a la funcionalidad de visualizar sus propuestas.
	2. El sistema recupera las propuestas del usuario desde la base de datos.
	3. El sistema muestra las propuestas del usuario en una lista ordenada en la interfaz.
Flujo Alternativo	Ninguno
Poscondiciones	El usuario visualiza sus propuestas dentro del sistema.

Nombre	Edición y Eliminación de Propuestas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de edición y eliminación de propuestas creadas por un usuario dentro de una asamblea.
Actores	Usuario
Precondiciones	El usuario debe haber iniciado sesión y tener propuestas creadas en el sistema.
Flujo Normal	1. El usuario accede a la funcionalidad de editar una de sus propuestas.
	2. El usuario modifica el título y/o la descripción de la propuesta.
	3. El sistema valida que la propuesta no tenga votos para permitir la modificación.
	* Si la propuesta tiene votos, el sistema muestra un mensaje indicando que no se puede editar y la razón.
	4. El sistema guarda los cambios en la base de datos.
	5. El usuario accede a la funcionalidad de eliminar una de sus propuestas.
	6. El sistema elimina la propuesta de la base de datos.
Flujo Alternativo	Ninguno
Poscondiciones	La propuesta se edita o elimina según la acción realizada por el usuario.

Nombre	Listado de Propuestas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de listar todas las propuestas creadas en una asamblea, mostrando detalles como título, descripción, usuario creador y número de votos.
Actores	Usuario
Precondiciones	Ninguna
Flujo Normal	1. El usuario accede a la funcionalidad de listar todas las propuestas.
	2. El sistema recupera todas las propuestas almacenadas en la base de datos para la asamblea actual.
	3. El sistema muestra las propuestas en una lista ordenada en la interfaz, incluyendo:
	* Título de la propuesta
	* Descripción de la propuesta
	* Nombre del usuario creador
	* Número de votos recibidos
Flujo Alternativo	Ninguno
Poscondiciones	El usuario visualiza todas las propuestas existentes en la asamblea actual.

Nombre	Votación de Propuestas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de votación por parte de un usuario en las propuestas presentadas dentro de una asamblea.
Actores	Usuario
Precondiciones	El usuario debe haber iniciado sesión y las propuestas deben estar disponibles para votar.
Flujo Normal	1. El usuario accede a la funcionalidad de votar por una propuesta.
	2. El usuario selecciona una de las propuestas disponibles para votar.
	3. El sistema verifica que el usuario no esté votando por sus propias propuestas. Si el usuario intenta votar por su propia propuesta, el sistema muestra un mensaje indicando que no está permitido.
	4. El sistema registra el voto del usuario en la base de datos.
	5. El sistema actualiza el número de votos de la propuesta seleccionada.
Flujo Alternativo	Ninguno
Poscondiciones	El voto del usuario se registra y se actualiza el conteo de votos para la propuesta seleccionada.

Nombre	Visualización de Resultados de Votación
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de visualización de los resultados de las votaciones para cada propuesta dentro de una asamblea.
Actores	Usuario
Precondiciones	Deben existir propuestas con votos en la asamblea.
Flujo Normal	1. El usuario accede a la funcionalidad de visualizar los resultados de votación.
	2. El sistema recupera los resultados de las votaciones desde la base de datos para la asamblea actual.
	3. El sistema muestra los resultados de las votaciones en una tabla clara y comprensible para el usuario.
	4. El sistema ordena la tabla por número de votos, de mayor a menor.
	5. El sistema destaca la propuesta ganadora con un formato visual diferente.
Flujo Alternativo	Ninguno
Poscondiciones	El usuario visualiza los resultados de las votaciones para cada propuesta en la asamblea actual, incluyendo la propuesta ganadora.

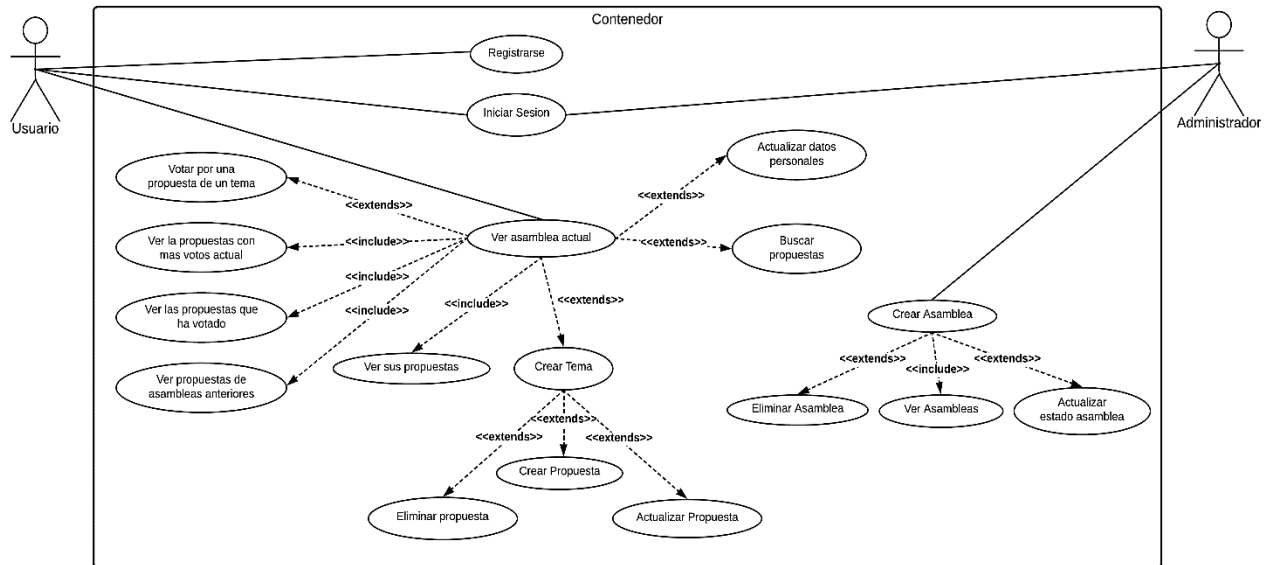
Nombre	Finalización de Asambleas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de finalización de una asamblea, mostrando el número total de votos en las propuestas y la propuesta ganadora.
Actores	Usuario
Precondiciones	La asamblea debe estar en estado de finalización.
Flujo Normal	1. El usuario accede a la funcionalidad de finalizar la asamblea.
	2. El sistema verifica que la asamblea esté en estado de finalización.
	* Si la asamblea no está en estado de finalización, el sistema muestra un mensaje indicando que no se puede finalizar y la razón.
	3. El sistema calcula el total de votos en todas las propuestas.
	4. El sistema determina la propuesta ganadora basada en el número de votos.
	5. El sistema muestra un mensaje emergente o una ventana modal al usuario indicando que la asamblea ha finalizado.
	6. El sistema muestra en la interfaz el número total de votos y la propuesta ganadora.
	7. El sistema cambia el estado de la asamblea a "Finalizada" en la base de datos.
Flujo Alternativo	Ninguno
Poscondiciones	La asamblea se finaliza, se muestran los resultados finales y el estado de la asamblea se actualiza en la base de datos.



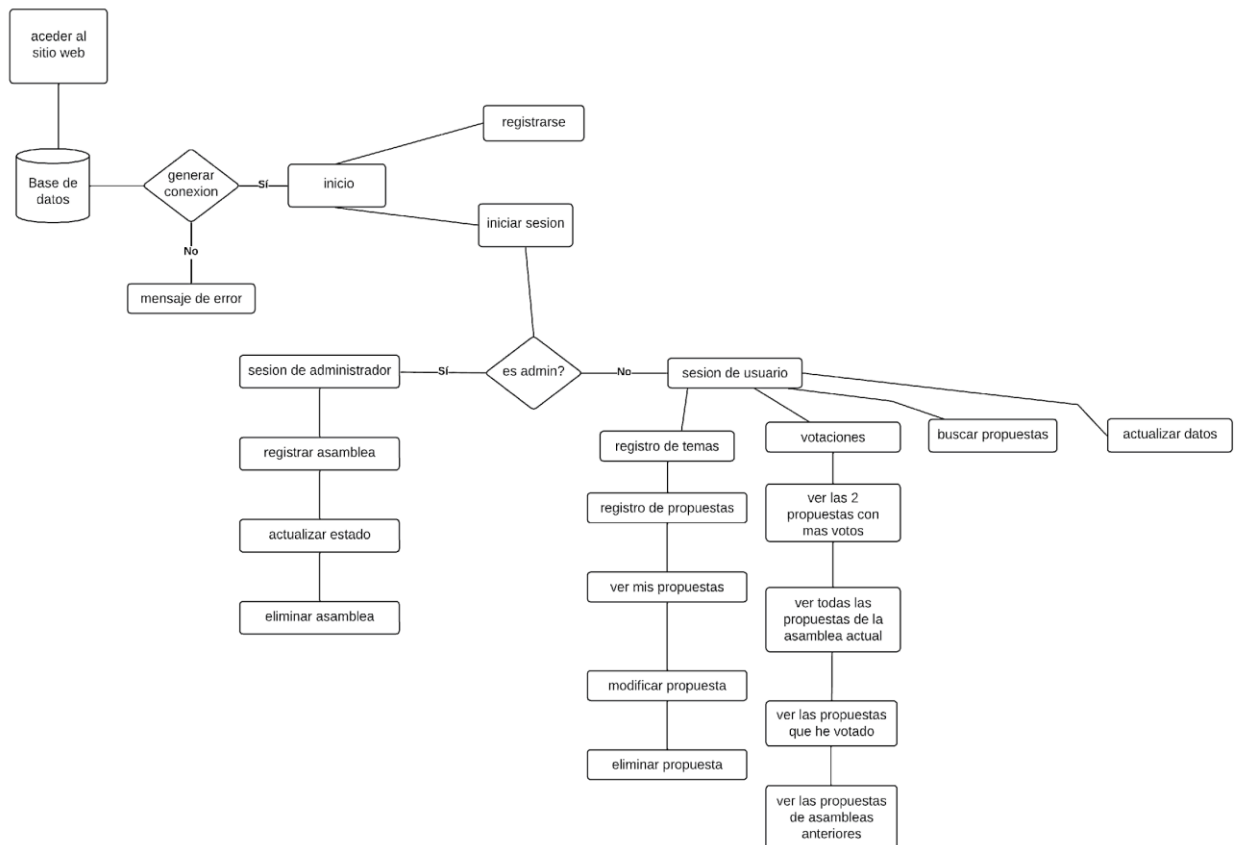
Nombre	Búsqueda de Propuestas
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de búsqueda de propuestas por título, descripción o nombre del creador de la propuesta.
Actores	Usuario
Precondiciones	Deben existir propuestas en la asamblea.
Flujo Normal	1. El usuario accede a la funcionalidad de búsqueda de propuestas.
	2. El usuario ingresa un término de búsqueda en el campo correspondiente.
	3. El sistema realiza la búsqueda en la base de datos para la asamblea actual, considerando los siguientes criterios:
	* Coincidencia total o parcial con el título de la propuesta.
	* Coincidencia total o parcial con la descripción de la propuesta.
	* Coincidencia total o parcial con el nombre del usuario creador de la propuesta.
	4. El sistema muestra los resultados de la búsqueda en una lista ordenada por relevancia (mayor coincidencia a menor coincidencia).
	5. El sistema muestra para cada resultado:
	* Título de la propuesta.
	* Descripción de la propuesta (resumida).
	* Nombre del usuario creador de la propuesta.
	6. El usuario puede seleccionar un resultado de la lista para visualizar la propuesta en detalle.
Flujo Alternativo	1. Si el usuario no ingresa ningún término de búsqueda, el sistema muestra un mensaje indicando que debe ingresar un término para realizar la búsqueda.
	2. Si no se encuentran propuestas que coincidan con el término de búsqueda, el sistema muestra un mensaje indicando que no se encontraron resultados.
Poscondiciones	El usuario visualiza los resultados de la búsqueda de propuestas o un mensaje informativo según corresponda.

Nombre	Gestión de Datos Personales
Autor	Equipo de Desarrollo de la Aplicación
Fecha	3 de junio de 2024
Descripción	Este caso de uso describe el proceso de actualización de los datos personales de un usuario registrado en el sistema.
Actores	Usuario
Precondiciones	El usuario debe haber iniciado sesión.
Flujo Normal	1. El usuario accede a la funcionalidad de gestión de datos personales.
	2. El sistema muestra al usuario sus datos personales actuales almacenados en la base de datos.
	3. El usuario puede modificar los siguientes campos:
	* Nombre completo.
	* Correo electrónico.
	* Contraseña.
	4. El usuario ingresa los nuevos datos en los campos correspondientes.
	5. El sistema valida la nueva información ingresada (por ejemplo, formato de correo electrónico, longitud de contraseña).
	* Si la información no es válida, el sistema muestra un mensaje indicando el error al usuario.
	* El usuario debe corregir la información e intentar nuevamente.
	6. Si la información es válida, el sistema actualiza los datos personales del usuario en la base de datos.
	7. El sistema muestra un mensaje al usuario indicando que sus datos personales se han actualizado correctamente.
Flujo Alternativo	Ninguno
Poscondiciones	Los datos personales del usuario se actualizan en la base de datos y se muestra un mensaje informativo al usuario.

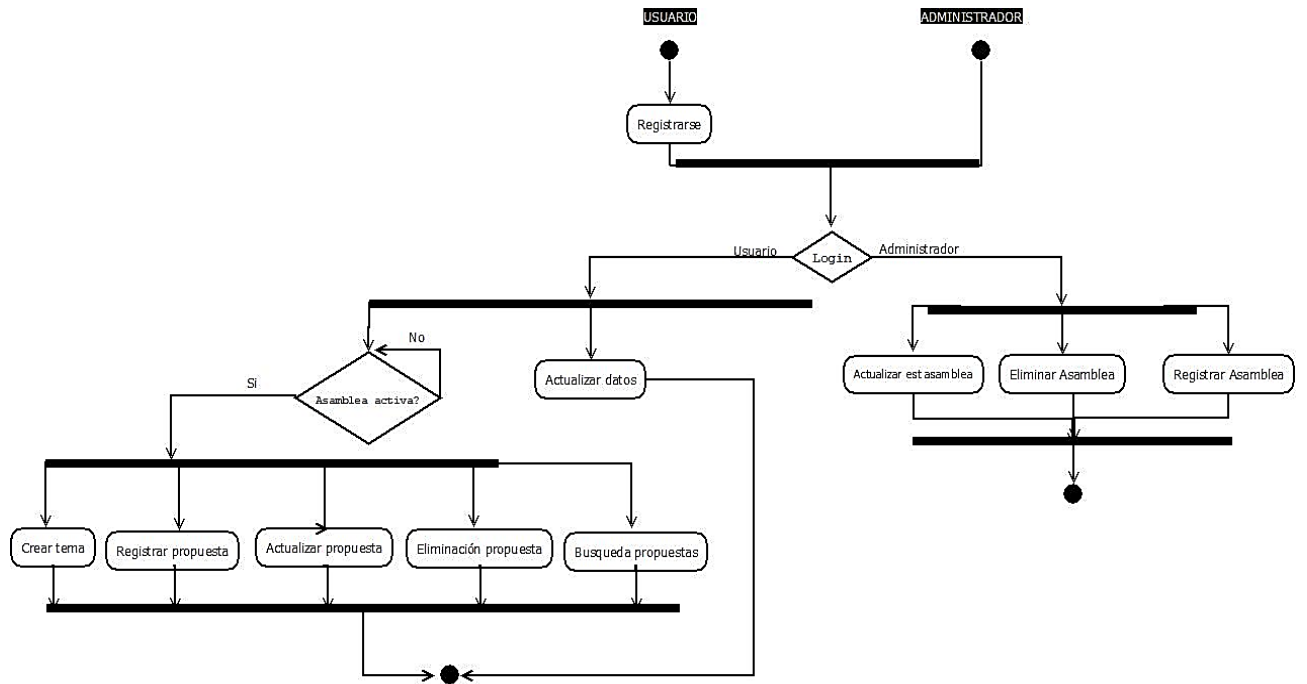
## Diagrama de casos de uso



## Diagrama de flujo

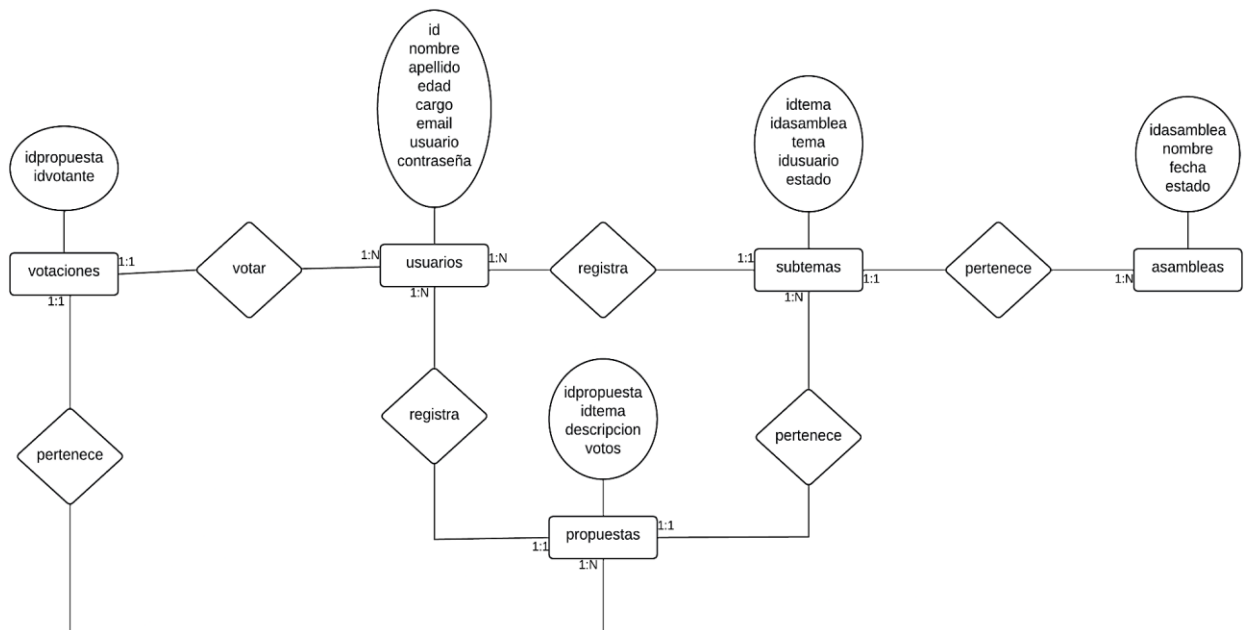


## Diagramas de actividad

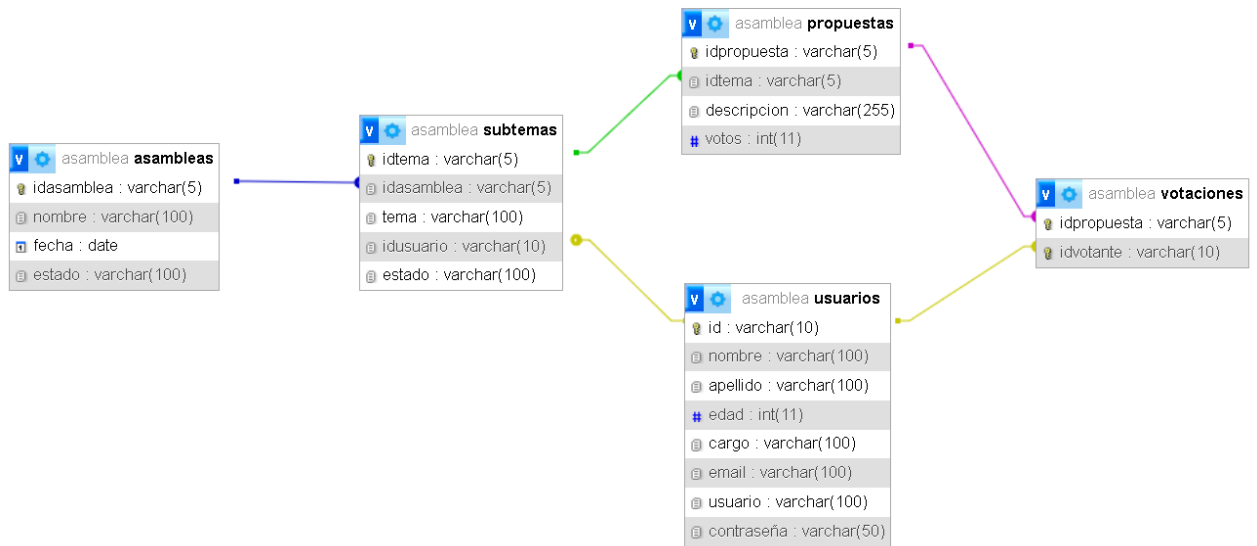


## Diseño de la base de datos

### Modelo conceptual



## Modelo lógico



## Esquema físico

- usuarios (id(PK), nombre, apellido, edad, cargo, email ,usuario, contraseña)
- asambleas (idasamblea(PK),nombre,fecha,estado)
- subtemas (idtema(PK),idasamblea(FK),tema,idusuario(FK),estado)
- propuestas (idpropuesta(PK),idtema(FK),descripcion,votos)
- votaciones (idpropuesta(FK),idvotante(FK))

# Conclusiones

## Uso de metodologías ágiles

Se concluye que las metodologías ágiles traen beneficios como la flexibilidad y adaptabilidad, las cuales permiten al equipo adaptarse o cambiar de acuerdo a los requisitos del cliente.

Permite también que se realice un trabajo mejor direccionado en base a las prioridades, dando como consecuencia una mejora constante de la calidad de los proyectos a realizar. La mejora constante se apoya en una revisión reiterativa y corrección de errores que logran una mayor satisfacción en el resultado final o del cliente, al cual se dan entregas frecuentes e información actualizada sobre el estado de su producto. Consiguiendo que tenga un conocimiento del progreso y que permita una incorporación de cambios según sean necesarios o requeridos por este último.

Lo cual no sería posible sin una colaboración y comunicación constante entre los miembros y participantes del proyecto, mejorando la experiencia con las reuniones retrospectivas que facilitan las mejoras continuas que se mencionan en el párrafo anterior. Esto le permite al equipo incrementar su eficiencia, gestionando el trabajo de una buena manera en relación al tiempo mediante los sprints, evitando los retrasos en el trabajo.

Por otra parte, las metodologías ágiles dejan cosas por aprender con sus respectivos desafíos, en los que podemos encontrar que se deben hacer cambios en nuestra forma de pensamiento y acercarnos más a la que nos plantean las metodologías ágiles, que fueron en su momento creadas por ingenieros y trabajadores de compañías asiáticas. Esto exige una transición a un entorno más ágil y estar dispuestos a la colaboración.

Compromiso en el que lleva someterse a una metodología, ya que estas a pesar de que su nombre lo dice son ágiles, contienen una estructura que definen los pasos que se deben hacer, así como que todos sus miembros estén dispuestos y en sincronía para poder trabajar. Así como también se debe gestionar de buena manera las reuniones para que estas sean únicamente retrospectivas y no como las típicas reuniones empresariales que duran demasiado.

Por último, estas metodologías permiten una organización eficiente para proyectos en entornos definidos como grupos no tan numerosos en los que se hay una mayor cohesión y colaboración que permitan una mayor calidad que se puede observar mediante la entrega continua. El éxito o no de la metodología radica principalmente en la disposición de los participantes para un cambio en algunos aspectos de su pensamiento y tiempo, con compromiso de buenas prácticas para el éxito del proyecto evitando el exceso de trabajo y reuniones innecesarias.

## **Dificultades encontradas**

### **Tecnológicas**

En la parte tecnológica del proyecto se pudieron encontrar algunas dificultades principalmente en la inexperiencia de los participantes en usar nuevas tecnologías o entornos en un proyecto. Esto debido a que nos encontrábamos de alguna manera en una zona de confort por trabajar en proyectos de aplicativos orientados a escritorio. A esto le podemos agregar a tener dificultades en un principio en donde alojar el proyecto web, que servidor usar y aprender nuevos lenguajes de programación, etc. Aunque al final, este último ítem no fue un problema porque tuvo una curva de aprendizaje relativamente baja por lo que no fue un impedimento o retraso para el proyecto. Con respecto a los otros componentes nos vimos obligados a probar con algunas herramientas hasta al final pudimos trabajar con una que fue Xampp para gestionar la base de datos y utilizar el servidor Apache para el proyecto.

### **Metodológicas**

En este apartado como todo lo nuevo hubo un pequeño tiempo de adaptabilidad, principalmente con respecto a la gestión de los documentos que pedía la metodología, entre los que se pudo encontrar las historias de usuario, por ejemplo. Ligado a este punto hubo dificultades a la hora de poder llevar a cabo los Sprints Retrospective y sincronizarnos debido a los distintos horarios, trabajo y demás actividades que tenían los participantes, también esto nos llevó a tener que disminuir de alguna manera la carga entre Sprints puesto que íbamos a un ritmo muy acelerado.

## **Del equipo de trabajo**

En las dificultades del equipo de trabajo pudimos evidenciar en algún momento que íbamos a un ritmo muy acelerado porque teníamos el temor de no acabar a tiempo o tener luego inconvenientes con otras asignaturas o actividades. El ritmo radica en que cada uno tenía una forma de trabajar distinta a algunos participantes, deducimos que puede ser a que no habíamos trabajado en grupos en un proyecto como este, puesto que en anteriores semestres se realizaban de forma individual o máximo en parejas. Por último, a veces la falta de sincronía, porque a pesar de realizar todas las tareas con éxito hacíamos las cosas en diferentes horarios o espacios, que no es necesariamente algo negativo, que fue uno de las cosas mencionadas en el punto anterior, pero fue una buena experiencia que nos deja un muy pequeño abrebocas al mundo laboral.