

Manual Técnico del Sistema de Votación

Introducción:

Este manual técnico tiene como objetivo proporcionar una guía detallada para aquellos involucrados en el desarrollo, implementación y mantenimiento del sistema de votación. El sistema de votación se ha diseñado para automatizar y simplificar el proceso de votación en una variedad de contextos, permitiendo a los usuarios registrar sus votos de manera eficiente y segura. A continuación, se establecen los objetivos técnicos y el alcance del sistema.

Objetivos Técnicos:

Los objetivos técnicos del sistema de votación son los siguientes:

Desarrollo Funcional: Desarrollar un sistema funcional que permita a los usuarios registrarse, iniciar sesión, votar y ver los resultados de la votación de manera efectiva y eficiente.

Seguridad: Implementar medidas de seguridad sólidas para proteger la integridad de los datos de votación y garantizar la autenticidad de los usuarios. Esto incluye la gestión segura de contraseñas y la prevención de votos duplicados o fraudulentos.

Escalabilidad: Diseñar el sistema para que pueda manejar un número variable de votantes y candidatos. El sistema debe ser capaz de escalar a medida que se incremente la participación en la votación.

Usabilidad: Garantizar que el sistema sea fácil de usar para los votantes, administradores y auditores. Esto incluye una interfaz de usuario intuitiva y una experiencia de votación sencilla.

Transparencia: Proporcionar una forma transparente de verificar y auditar los resultados de la votación. Los auditores deben poder revisar el proceso de votación y los resultados de manera efectiva.

Integridad de Datos: Garantizar que los datos de votación se almacenen y transmitan de manera segura y que no se vean comprometidos en ningún punto del proceso.

Rendimiento: Optimizar el rendimiento del sistema para manejar un alto volumen de votantes sin degradación del servicio.

Disponibilidad: Asegurar que el sistema esté disponible durante el período de votación, evitando interrupciones no planificadas.

Auditoría y Registro de Eventos: Registrar eventos importantes del sistema, como la creación de usuarios, votos emitidos y cambios en la configuración del sistema, para facilitar la auditoría y el seguimiento.

Respaldo y Recuperación de Datos: Implementar un sistema de respaldo y recuperación de datos para garantizar que los datos críticos no se pierdan en caso de fallos o problemas técnicos.

Documentación: Proporcionar documentación técnica completa que describa la arquitectura del sistema, las tecnologías utilizadas y los procedimientos de mantenimiento.

Cumplimiento Normativo: Asegurarse de que el sistema cumple con todas las regulaciones y requisitos legales aplicables en el contexto en el que se utiliza.

Alcance del Sistema:

El alcance del sistema de votación abarca:

Registro de Usuarios: Permite a los votantes registrarse en el sistema proporcionando su información personal y la creación de una cuenta con contraseña.

Inicio de Sesión: Los usuarios pueden iniciar sesión en el sistema utilizando su nombre de usuario y contraseña.

Emisión de Votos: Los votantes pueden emitir sus votos por candidatos específicos después de iniciar sesión.

Conteo de Votos: El sistema cuenta y almacena los votos de manera segura, evitando la duplicación y asegurando la confidencialidad.

Resultados de Votación: Proporciona una interfaz para que los usuarios, incluyendo auditores, vean los resultados de la votación en tiempo real.

Medidas de Seguridad: Implementa medidas de seguridad como cifrado de contraseñas, prevención de votos duplicados y auditorías para garantizar la integridad y autenticidad del proceso de votación.

Este manual técnico proporcionará detalles técnicos específicos sobre la implementación del sistema, su arquitectura y sus componentes. Además, servirá como referencia para el mantenimiento continuo y las mejoras futuras del sistema de votación.

Diagrama de casos de uso:

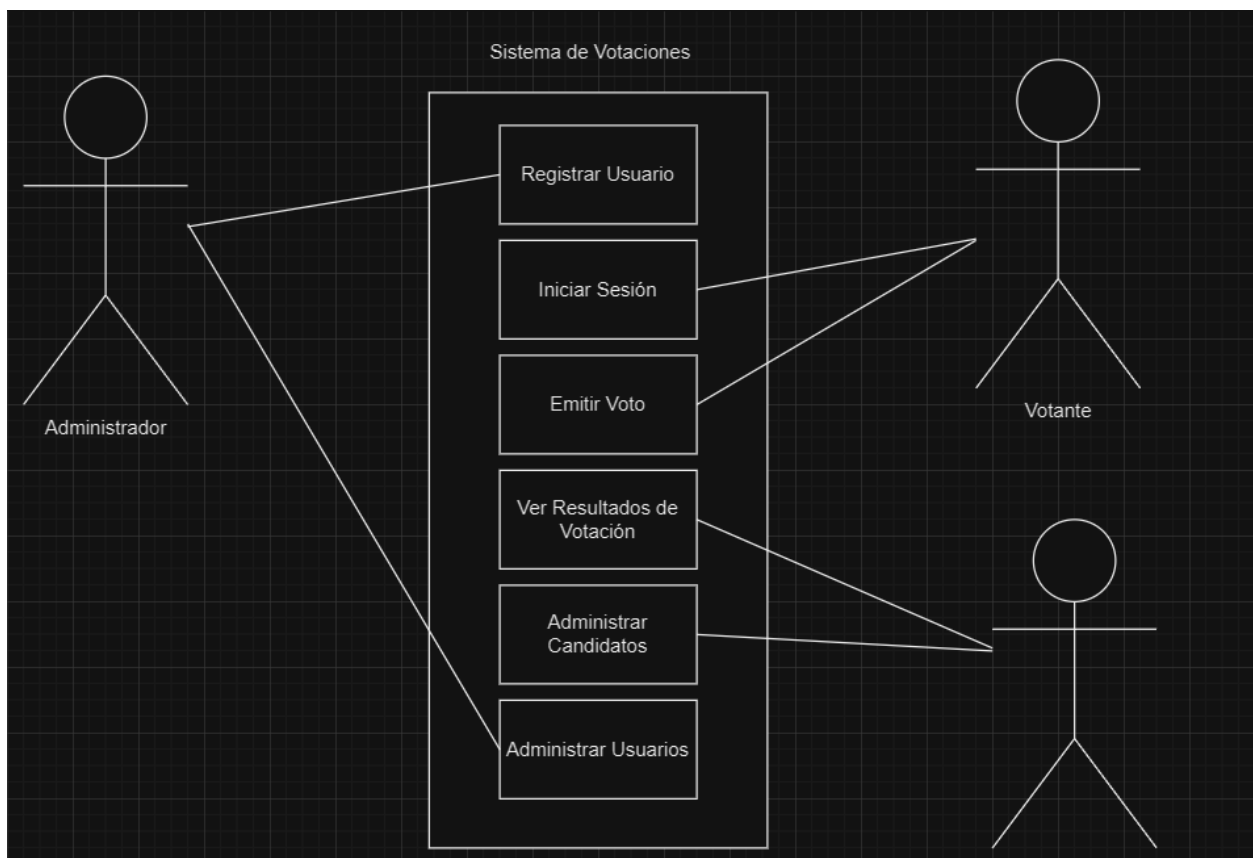


Diagrama de clases:

Algoritmo:

Iniciar el programa.

Crear una matriz para almacenar usuarios con un límite máximo de usuarios.

Inicializar un arreglo de candidatos y un arreglo para contar los votos.

Mostrar un menú con las siguientes opciones:

- a. Registrar usuario.
- b. Iniciar sesión y votar.
- c. Ver resultados de votación.
- d. Salir del programa.

Leer la opción seleccionada por el usuario.

Realizar acciones basadas en la opción seleccionada:

- a. Si la opción es "Registrar usuario":
 - i. Solicitar al usuario que ingrese su nombre, tipo de usuario y contraseña.
 - ii. Almacenar la información del usuario en la matriz de usuarios.
 - iii. Incrementar el contador de usuarios registrados.
 - iv. Mostrar un mensaje de éxito.
- b. Si la opción es "Iniciar sesión y votar":
 - i. Solicitar al usuario que ingrese su nombre de usuario y contraseña.
 - ii. Verificar si el usuario existe y si es un votante.
 - iii. Si es un votante, mostrar la lista de candidatos y permitir al usuario emitir un voto.
 - iv. Registrar el voto y mostrar un mensaje de confirmación.
 - v. Si el usuario no es válido o no tiene permiso para votar, mostrar un mensaje de error.
- c. Si la opción es "Ver resultados de votación":
 - i. Mostrar los resultados de la votación, incluyendo el número de votos para cada candidato.
- d. Si la opción es "Salir del programa":
 - i. Cerrar el programa y liberar recursos.

Repetir el proceso desde el paso 4 hasta que el usuario decida salir del programa.

Diagrama de flujo:

Tecnología Utilizadas:

Java: El lenguaje de programación en sí, que es utilizado para desarrollar la lógica y la funcionalidad del programa.

Scanner: Una clase de Java que se utiliza para obtener la entrada del usuario desde la consola. En este caso, se utiliza para leer la entrada del usuario, como nombres de usuarios, contraseñas y opciones de menú.

Matrices (Arrays): Se utilizan matrices para almacenar información sobre usuarios registrados y para realizar un seguimiento de los votos para cada candidato. Esto involucra matrices de cadenas (String) y matrices de enteros (int).

Ciclo While: El ciclo while (true) se utiliza para crear un bucle infinito que permite que el programa continúe ejecutándose hasta que el usuario elija la opción "4. Salir".

Switch-Case: Se utiliza una estructura de control switch-case para gestionar diferentes opciones seleccionadas por el usuario. Cada caso maneja una funcionalidad específica, como registrar un usuario, permitir a un votante votar, mostrar los resultados de la votación, o salir del programa.

En resumen, el código está escrito en Java, utiliza la clase Scanner para interactuar con el usuario a través de la consola y se apoya en estructuras de control como bucles y switch-case para gestionar las opciones del programa. Además, hace uso de matrices para almacenar y gestionar datos relacionados con los usuarios y la votación.

Instrucciones de Configuración del entorno de trabajo:

Instalar Java Development Kit (JDK):

Asegúrate de tener el Java Development Kit (JDK) instalado en tu sistema. Puedes descargarlo desde el sitio web oficial de Oracle o utilizar una distribución OpenJDK. Asegúrate de que la versión de Java sea al menos Java 8.

Instalar un Entorno de Desarrollo Integrado (IDE) (Opcional):

Aunque no es obligatorio, se recomienda utilizar un IDE para desarrollar y ejecutar código Java de manera más eficiente. Algunas opciones populares incluyen Eclipse, IntelliJ IDEA o NetBeans. Puedes descargar e instalar uno de estos IDEs de acuerdo a tus preferencias.

Editor de Texto:

Si no deseas utilizar un IDE, puedes escribir y editar tu código en cualquier editor de texto, como Notepad (en Windows) o Visual Studio Code. Guarda el archivo con la extensión ".java".

Compilar y Ejecutar:

Una vez que tengas tu código Java listo, puedes compilarlo y ejecutarlo de la siguiente manera:

Abre una terminal o línea de comandos en tu sistema.

Navega al directorio donde se encuentra tu archivo Java utilizando el comando `cd` (por ejemplo, `cd /ruta/al/directorio`).

Compila el código Java utilizando el comando `javac` seguido del nombre del archivo Java (sin la extensión `".java"`). Por ejemplo: `javac SistemaDeVotacion`.

Ejecuta el programa compilado con el comando `java` seguido del nombre del archivo (sin la extensión `".class"`). Por ejemplo: `java SistemaDeVotacion`.

Interactúa con el Programa:

Después de ejecutar el programa, seguirá ejecutándose y te mostrará un menú en la consola. Puedes interactuar con el programa siguiendo las instrucciones que muestra en la pantalla.

Proceso de compilación del proyecto

Asegúrate de tener Java instalado en tu sistema. Puedes verificar si Java está instalado ejecutando el comando `java -version` en una ventana de terminal. Si no tienes Java instalado, puedes descargarlo desde el sitio web oficial de Oracle o utilizar una distribución de OpenJDK.

Abre un editor de texto o un entorno de desarrollo integrado (IDE) como Eclipse, IntelliJ IDEA o NetBeans.

Copia y pega el código Java proporcionado en un nuevo archivo en tu editor de texto o IDE.

Guarda el archivo con una extensión `".java"`, por ejemplo, `"SistemaDeVotacion.java"`.

Abre una ventana de terminal o línea de comandos y navega al directorio donde se encuentra el archivo `".java"` que acabas de crear.

Compila el código Java escribiendo el siguiente comando en la terminal y presionando Enter:

```
javac SistemaDeVotacion.java
```

Esto generará un archivo `".class"` que contiene el bytecode compilado.

Ejecución:

7. Una vez compilado con éxito, puedes ejecutar el programa escribiendo el siguiente comando en la terminal y presionando Enter:

```
java SistemaDeVotacion
```

El programa se ejecutará y comenzará a interactuar con el usuario a través de la consola.

Sigue las instrucciones en la consola para registrar usuarios, permitir que los votantes emitan sus votos, ver los resultados de la votación o salir del programa.

Recuerda que para compilar y ejecutar código Java, es importante tener el entorno de desarrollo adecuado y Java correctamente configurado en tu sistema.

Contacto

Correo:betogar14@gmail.com