

Introducción

El presente proyecto tiene como objetivo el desarrollo de un sistema de Ticket que pueda crear ticket, borrar ticket y modificar ticket, agregar usuarios, modificar usuarios y eliminar usuarios también Asignar áreas de trabajo. Este sistema, implementado en **Java y con interfaz grafica de javaFX**, está diseñado para proporcionar una visualización clara y detallada de una gestión de ticket y áreas de trabajo de una empresa, mediante una interfaz gráfica que nos muestra que es lo que ya está guardado y lo que podemos realizar dependiendo el roll. El objetivo es hacer un sistema funcional que se de utilidad para una empresa.

Objetivos Técnicos:

- Configurar Parámetros del Sistema
- Gestionar Roles y Permisos
- Registrar Departamentos o Áreas de Trabajo
- Registrar Usuarios Iniciales
- Definir Estados de los Tickets
- Configurar Flujos de Trabajo de los Tickets
- Gestionar Tickets
- Consultar Lista de Solicitudes Pendientes
- Cambiar Estado de un Ticket
- Agregar Nota a un Ticket

Algoritmo de toda la solución:

Menú:

Hay 10 botones

- 9 son de los distintos puntos
- 1 es para salir.

Configurar Parámetros del Sistema

En la configuración encontraremos:

1. Un apartado para cambiar el nombre de la empresa
2. Un botón para cargar un logo
3. Un apartado para seleccionar un tiempo de vencimiento
4. Un apartado para escribir los niveles de prioridad.
5. 3 botones

- Para Guardar
- Para regresar al menú
- Para Cancelar

6. 2 apartados para seleccionar

- Idiomas
- Zona Horaria

Gestionar Roles y Permisos

En gestión de roles y permisos se encuentra

1. 4 apartados para ingresar:

- Nombre Completo
- Correo Electrónico
- Nombre de Usuario

- Contraseña
2. 2 apartados para seleccionar
 1. Rol Asignado
 2. Departamento
 3. 3 botones
- Para Guardar
 - Para regresar al menú
 - Para Cancelar

Registrar Departamentos o Áreas de Trabajo

En Departamentos o Áreas de Trabajo se encuentra:

1. 3 apartados para ingresar:
 - Departamentos
 - Nombre de Departamentos
 - Descripción
2. Un apartado para seleccionar
3. Técnicos Asignados
3. 3 botones
 - Para Guardar
 - Para regresar al menú

- Para Cancelar

Registrar Usuarios Iniciales

1. 4 apartados para ingresar:

- Roles
- Nombre de Rol
- Descripción de Rol
- Permisos

2. 3 botones

- Para Guardar
- Para regresar al menú
- Para Cancelar

Definir Estados de los Tickets

En estados de los tickets se encuentra:

1. 2 apartados para ingresar:

- Nombre de los Estados
 - Descripción
2. Un apartado para seleccionar el estado final con si o no
 3. Una lista de estados
 4. 3 botones
 - Para Guardar
 - Para regresar al menú
 - Para Cancelar

Configurar Flujos de Trabajo de los Tickets

En la configuración de flujo de trabajo se encuentra:

1. Un apartado para ingresar el flujo de trabajo y 3 botones los cuales son:
 - Agregar
 - Modificar
 - Eliminar
2. Un apartado para ingresar el nombre de flujo de trabajo y un botón de Guardar

3. Un botón para regresar al menú

Gestionar Tickets

En Gestión de tickets se encuentra:

1. 3 apartados para ingresar:
 - Titulo
 - Lista
 - Descripción
2. 2 apartados para seleccionar
 - Departamentos Asignados
 - Nivel de Prioridad
3. 4 botones
 - Crear
 - Actualizar
 - Cerrar

- Menú

Consultar Lista de Solicitudes Pendientes

En la Consulata de listas de solicitudes se encuentra:

1. Un apartado para ingresar Numeros de tickets
2. Un apartado para seleccionar la descripción y un botón para aplicar un filtro
3. Una tabla de tickets

Cambiar Estado de un Ticket

En el estado de tickets se encuentra:

1. Una tabla del número de estado
2. Un apartado para ingresar un comentario

3. Un apartado para seleccionar un nuevo estado

4. 3 botones

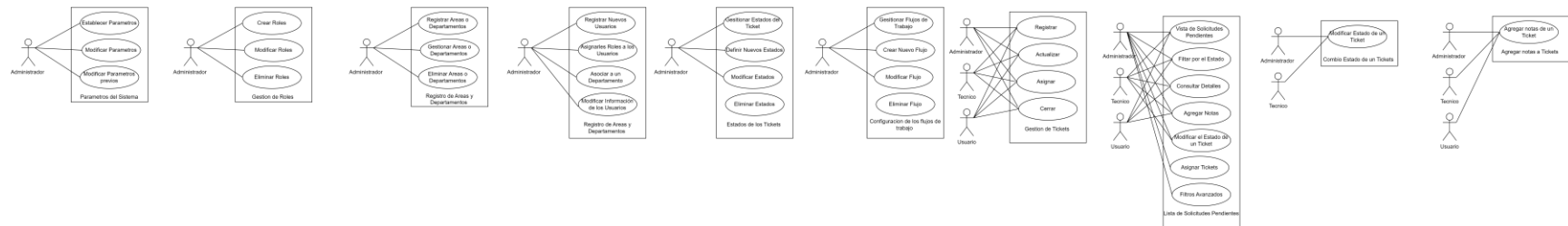
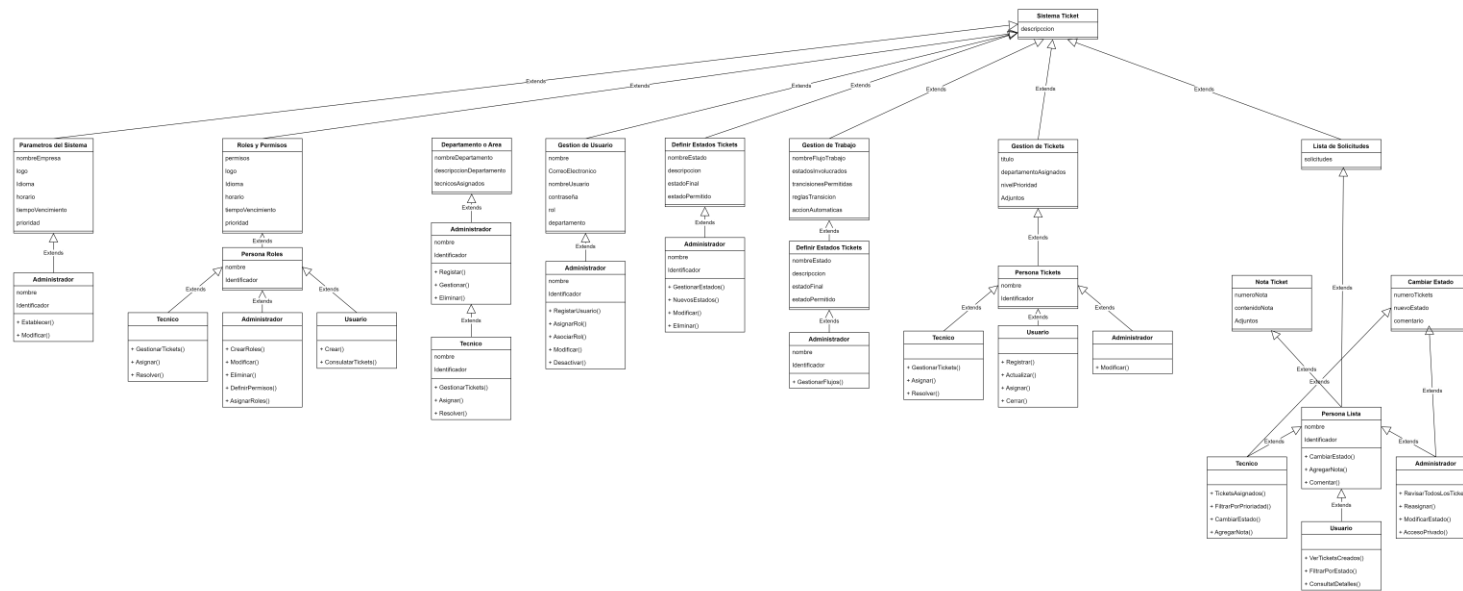
- Para Guardar
- Para regresar al menú
- Para Cancelar

Agregar Nota a un Ticket

En agregar nota de ticket se encuentra:

1. Una lista de tickets
2. Un apartado apartado para ingresar una Nota
3. 3 botones
 - Para Guardar
 - Para regresar al menú
 - Para Cancelar

Diagramas de Clases:



Instrucciones de configuración del entorno de trabajo:

1. Instalación de Herramientas Necesarias

- Asegúrate de tener instalado el JDK. Puedes descargarlo desde [Oracle](#) o usar un gestor de paquetes como apt en Ubuntu

```
sudo apt update
sudo apt install openjdk-17-jdk
```

- Verifica la instalación:

```
java -version
```

- Si tu proyecto utiliza **Maven** para la gestión de dependencias, asegúrate de instalarlo:

```
sudo apt install maven
```

2. Estructura del Proyecto

- Asegúrate de que tu proyecto Java tenga la siguiente estructura:

```
/sistema-ventas
  /src
    /main
      /java
        /com/miempresa/sistemaventas
          Main.java
          VentasController.java
          ProductosController.java
      /resources
        productos.csv
        ventas.csv
    /libs
  /target
  pom.xml (si usas Maven)
```

3. Configuración del Proyecto en el IDE

a. Configuración del JDK

- Abre tu IDE y configura la ruta del JDK en las preferencias del proyecto:
 - En **IntelliJ**, ve a File -> Project Structure -> Project, y selecciona la versión del JDK.

b. Agregar Dependencias

- Si no usas Maven, asegúrate de agregar manualmente los archivos .jar (como JFreeChart o JavaFX) a tu proyecto. En **Eclipse**, ve a:
 - Project -> Properties -> Java Build Path -> Libraries -> Add External JARs.

4. Conexión a Base de Datos (Opcional)

Si deseas usar una base de datos para almacenar información de productos, ventas, etc., puedes configurar un motor de base de datos como **MySQL** o **SQLite**.

- Instala MySQL en tu sistema:

```
sudo apt install mysql-server
```

5. Ejecución del Proyecto

Para ejecutar tu proyecto:

a. Desde el IDE

- Selecciona la clase Main.java y ejecuta el proyecto desde el IDE.

b. Desde la Línea de Comandos

Si usas Maven:

```
mvn clean install  
mvn exec:java -Dexec.mainClass="com.miempresa.sistemaventas.Main"
```

6. Pruebas y Depuración

Asegúrate de realizar pruebas unitarias para comprobar el correcto funcionamiento de cada parte del sistema, usando frameworks como **JUnit**:

```
<dependency>  
  <groupId>junit</groupId>  
  <artifactId>junit</artifactId>  
  <version>4.13.2</version>  
  <scope>test</scope>  
</dependency>
```


Tecnologías utilizadas:

1. Lenguaje de programación: Java.
2. JavaFX
3. GitHub
4. SceneBuilder

Sección de solución de problemas comunes:

Error: `ClassNotFoundException` al intentar ejecutar el proyecto

asegúrate de que los archivos `.jar` están en la ruta de construcción (build path).

Error de compilación: `incompatible types`

Verifica que el tipo de datos que intentas asignar es compatible con el tipo de la variable. Por ejemplo, si una variable es de tipo `int`, asegúrate de que no estás asignando un valor de tipo `double` sin realizar una conversión.

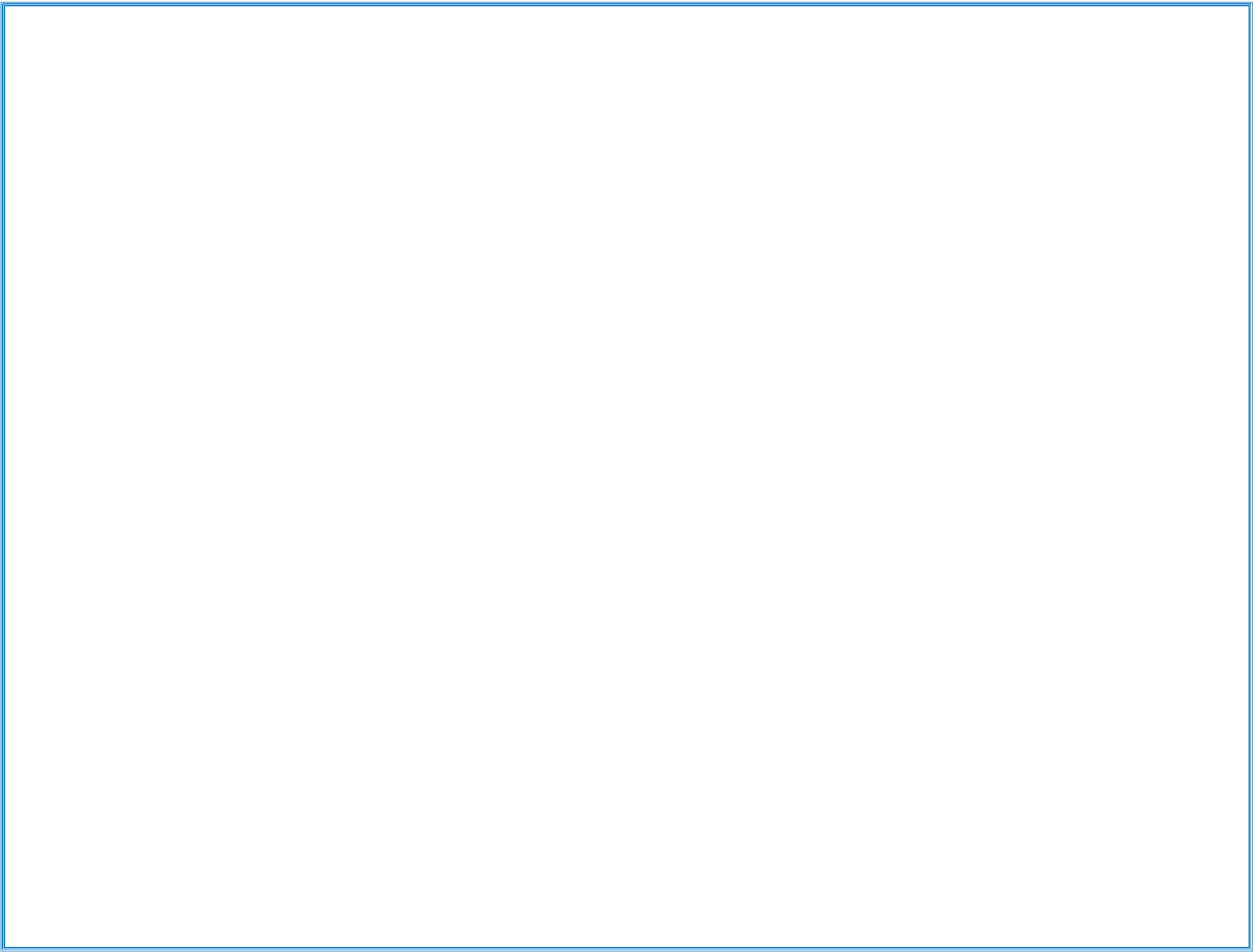
Problemas con el control de versiones (Git)

Si enfrentas un conflicto en Git, puedes resolverlo manualmente editando los archivos en conflicto, y luego ejecutar:

```
git add <archivo_resuelto>
git commit
```

Aplicación Java lenta o con problemas de rendimiento

- Optimiza las consultas a la base de datos, por ejemplo, utilizando índices en las tablas.
- Revisa tu código para evitar el uso innecesario de bucles anidados o procesamiento innecesario de grandes volúmenes de datos en el lado de la aplicación.



Información de contacto:

Nombre:

Luis Alejandro Berreondo Florián

Número telefónico:

3399-9914

Correo electrónico:

luisalejandroberreondoflorian@gmail.com

archivo tipo jar