

**UNIVERSIDAD DON BOSCO**



**FACULTAD DE INGENIERÍA**  
**ESCUELA DE COMPUTACIÓN**

**Manual de Técnico**

**Presentado por:**

Magaña Martínez, Samuel Eduardo MM220035

Padilla Ramírez, Alexandra Guadalupe NR221019

Pineda Fuentes, Geovany Arturo PF211251

Rodríguez Sánchez, Enrique Ernesto RS132134

Trujillo Osorio, Erick Geovanni TO220989

Villalobos Eguizábal, Oscar Alejandro VE220589

**Catedrático:**

Ing. Karens Medrano

**Materia:**

Lenguajes Interpretados en el Cliente 01T

**Soyapango, 1 de septiembre del 2024**

## Contenido

Introducción .....	3
Objetivos .....	3
Requerimientos Técnicos.....	3
Herramientas utilizadas para el desarrollo.....	3
React .....	3
Express.js .....	4
MySQL .....	4
Casos de Uso .....	10
Diagrama de Clases .....	14
Diagrama de Secuencia .....	15

# Introducción

Este manual describe los pasos necesarios para cualquier persona que tenga ciertas bases de sistemas pueda realizar la instalación del Sistema para Web para Cooperativas. Es importante tener en cuenta que en el presente manual se hace mención a las especificaciones mínimas de hardware y software para la correcta instalación del sistema.

## Objetivos

Brindar la información necesaria para poder realizar la instalación y configuración del sistema.

Específicos:

- Representar la funcionalidad técnica de la estructura, diseño y definición del aplicativo.
- Definir claramente el procedimiento de instalación del aplicativo.
- Detallar la especificación de los requerimientos de Hardware y Software necesarios para la instalación de la aplicación.
- Describir las herramientas utilizadas para el diseño y desarrollo del software

## Requerimientos Técnicos

- **Sistema Operativo:** Windows 10 o superior, MacOS 10.12 o superior, Linux (Ubuntu, Fedora, etc.) con soporte para navegadores.
- **Procesador:** Un procesador moderno de **1 GHz** o superior.
- **Memoria RAM:** Mínimo de **2 GB** de RAM.
- **Almacenamiento:** Al menos **10 GB** de espacio libre en disco (para la instalación de navegadores y otras herramientas relacionadas).
- **Pantalla:** Resolución mínima de **1024x768** (aunque una resolución más alta es recomendada para una mejor experiencia).
- **Conexión a Internet:** Conexión de banda ancha con al menos **2 Mbps** de velocidad de descarga para la navegación web fluida.

## Herramientas utilizadas para el desarrollo

### React

React es una biblioteca de JavaScript creada por Facebook que se utiliza para construir interfaces de usuario interactivas y dinámicas. Su enfoque principal es el desarrollo de aplicaciones web mediante componentes reutilizables que manejan su propio estado y lógica. React emplea un "Virtual DOM", una representación en memoria de la interfaz, que optimiza las actualizaciones y renderizados, lo que mejora el rendimiento de la aplicación. Además, React sigue un enfoque declarativo, lo que permite a los desarrolladores describir cómo debe lucir la interfaz en diferentes estados, mientras React se encarga de actualizar la vista de manera eficiente cuando los datos cambian.

## Express.js

Express.js es un framework web minimalista y flexible para Node.js, diseñado para crear aplicaciones web y APIs de manera sencilla y eficiente. Proporciona un conjunto robusto de herramientas y características para gestionar peticiones HTTP, rutas, middleware y respuestas. Aunque es minimalista, Express.js es altamente personalizable, permitiendo a los desarrolladores agregar funcionalidades a través de módulos y paquetes adicionales.

## MySQL

Es un manejador de Bases de Datos, el cual permite múltiples hilos y múltiples usuarios, fue desarrollado como software libre. Aunque se puede usar sobre varias plataformas es muy utilizado sobre LINUX. Es libre para uso en Servidores WEB. Ofrece ventajas tales como fácil adaptación a diferentes entornos de desarrollo, Interacción con Lenguajes de Programación como PHP, Java Script y fácil Integración con distintos sistemas operativos.

## Importación de la base de datos en MySQL

SET

```
@OLD_UNIQUE_CHECKS = @@UNIQUE_CHECKS,  
  
UNIQUE_CHECKS = 0;
```

SET

```
@OLD_FOREIGN_KEY_CHECKS = @@FOREIGN_KEY_CHECKS,  
  
FOREIGN_KEY_CHECKS = 0;
```

SET

```
@OLD_SQL_MODE = @@SQL_MODE,  
  
SQL_MODE =  
'ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FO  
R_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
```

```
DROP SCHEMA IF EXISTS `cooperativa_db`;
```

```
CREATE SCHEMA IF NOT EXISTS `cooperativa_db` DEFAULT CHARACTER SET utf8;
```

```
USE `cooperativa_db`;
```

```
CREATE TABLE IF NOT EXISTS `cooperativa_db`.`estados` (  
  `idEstado` INT NOT NULL AUTO_INCREMENT,  
  `estado` VARCHAR(12) NOT NULL,  
  `descripcion_estado` MEDIUMTEXT NULL,  
  PRIMARY KEY (`idEstado`)  
) ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS `cooperativa_db`.`tipos_empleados` (  
  `idTipo_empleado` INT NOT NULL AUTO_INCREMENT,  
  `tipo_empleado` VARCHAR(30) NOT NULL,  
  `descripcion_tipo_empleado` MEDIUMTEXT NULL,  
  PRIMARY KEY (`idTipo_empleado`)  
) ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS `cooperativa_db`.`creditos` (  
  `idCredito` INT NOT NULL AUTO_INCREMENT,  
  `nombre_credito` VARCHAR(50) NOT NULL,  
  `interes` DECIMAL(8, 2) NOT NULL,  
  `descripcion_credito` MEDIUMTEXT NOT NULL,  
  PRIMARY KEY (`idCredito`)  
) ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS `cooperativa_db`.`empleados` (  
  `idEmpleado` INT NOT NULL AUTO_INCREMENT,  
  `nombre_empleado` VARCHAR(50) NOT NULL,  
  `apellido_empleado` VARCHAR(50) NOT NULL,  
  `direccion_empleado` MEDIUMTEXT NOT NULL,
```

```

`username` VARCHAR(30) NOT NULL,
`password` VARCHAR(25) NOT NULL,
`correo_empleado` VARCHAR(50) NOT NULL,
`telefono_empleado` VARCHAR(8) NOT NULL,
`tipo_empleado_id` INT NULL,
`estado_id` INT NULL,
PRIMARY KEY (`idEmpleado`),
CONSTRAINT `fk_estado_id_empleados` FOREIGN KEY (`estado_id`) REFERENCES
`cooperativa_db`.`estados` (`idEstado`) ON DELETE CASCADE ON UPDATE CASCADE,
CONSTRAINT `fk_tipo_empleado_id` FOREIGN KEY (`tipo_empleado_id`) REFERENCES
`cooperativa_db`.`tipos_empleados` (`idTipo_empleado`) ON DELETE CASCADE ON UPDATE
CASCADE
) ENGINE = InnoDB;

```

```

CREATE UNIQUE INDEX `username_UNIQUE` ON `cooperativa_db`.`empleados` (`username`
ASC) VISIBLE;

```

```

CREATE UNIQUE INDEX `correo_empleado_UNIQUE` ON `cooperativa_db`.`empleados`
(`correo_empleado` ASC) VISIBLE;

```

```

CREATE INDEX `tipo_empleado_id_idx` ON `cooperativa_db`.`empleados` (`tipo_empleado_id`
ASC) VISIBLE;

```

```

CREATE INDEX `estado_id_idx` ON `cooperativa_db`.`empleados` (`estado_id` ASC) VISIBLE;

```

```

CREATE TABLE IF NOT EXISTS `cooperativa_db`.`clientes` (

```

```

`idCliente` INT NOT NULL AUTO_INCREMENT,
`nombre_cliente` VARCHAR(50) NOT NULL,
`apellido_cliente` VARCHAR(50) NOT NULL,
`direccion_cliente` MEDIUMTEXT NOT NULL,
`dui` VARCHAR(10) NOT NULL,

```

```

`salario` DECIMAL(8, 2) NOT NULL,
`correo_cliente` VARCHAR(50) NOT NULL,
`telefono_cliente` VARCHAR(8) NOT NULL,
`documento1` MEDIUMTEXT NULL,
`documento2` MEDIUMTEXT NULL,
`documento3` MEDIUMTEXT NULL,
`estado_id` INT NOT NULL,
PRIMARY KEY (`idCliente`),
CONSTRAINT `fk_estado_id_clientes` FOREIGN KEY (`estado_id`) REFERENCES
`cooperativa_db`.`estados` (`idEstado`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE = InnoDB;

```

```

CREATE UNIQUE INDEX `dui_UNIQUE` ON `cooperativa_db`.`clientes` (`dui` ASC) VISIBLE;

```

```

CREATE UNIQUE INDEX `correo_cliente_UNIQUE` ON `cooperativa_db`.`clientes`
(`correo_cliente` ASC) VISIBLE;

```

```

CREATE INDEX `estado_id_idx` ON `cooperativa_db`.`clientes` (`estado_id` ASC) VISIBLE;

```

```

CREATE TABLE IF NOT EXISTS `cooperativa_db`.`solicitudes` (
  `idSolicitud` INT NOT NULL AUTO_INCREMENT,
  `cliente_id` INT NOT NULL,
  `empleado_id` INT NULL,
  `estado_id` INT NOT NULL,
  `credito_id` INT NOT NULL,
  PRIMARY KEY (`idSolicitud`),
  CONSTRAINT `fk_cliente_id` FOREIGN KEY (`cliente_id`) REFERENCES
`cooperativa_db`.`clientes` (`idCliente`) ON DELETE CASCADE ON UPDATE CASCADE,
  CONSTRAINT `fk_empleado_id` FOREIGN KEY (`empleado_id`) REFERENCES
`cooperativa_db`.`empleados` (`idEmpleado`) ON DELETE CASCADE ON UPDATE CASCADE,

```

```
CONSTRAINT `fk_estado_id_solicitudes` FOREIGN KEY (`estado_id`) REFERENCES  
`cooperativa_db`.`estados` (`idEstado`) ON DELETE CASCADE ON UPDATE CASCADE,  
  
CONSTRAINT `fk_credito_id` FOREIGN KEY (`credito_id`) REFERENCES  
`cooperativa_db`.`creditos` (`idCredito`) ON DELETE CASCADE ON UPDATE CASCADE  
  
) ENGINE = InnoDB;
```

```
CREATE INDEX `cliente_id_idx` ON `cooperativa_db`.`solicitudes` (`cliente_id` ASC) VISIBLE;
```

```
CREATE INDEX `empleado_id_idx` ON `cooperativa_db`.`solicitudes` (`empleado_id` ASC)  
VISIBLE;
```

```
CREATE INDEX `estado_id_idx` ON `cooperativa_db`.`solicitudes` (`estado_id` ASC) VISIBLE;
```

```
CREATE INDEX `credito_id_idx` ON `cooperativa_db`.`solicitudes` (`credito_id` ASC) VISIBLE;
```

```
SET  
  
SQL_MODE = @OLD_SQL_MODE;
```

```
SET  
  
FOREIGN_KEY_CHECKS = @OLD_FOREIGN_KEY_CHECKS;
```

```
SET  
  
UNIQUE_CHECKS = @OLD_UNIQUE_CHECKS;
```

```
INSERT INTO estados VALUES  
  
(1, 'activo', 'esta activo'),  
  
(2, 'inactivo', 'no esta activo'),  
  
(3, 'Pendiente', 'La solicitud no ha sido asignada a un empleado'),  
  
(4, 'En Proceso', 'La solicitud ha sido asignada y se comenzo la gestion'),  
  
(5, 'Finalizada', 'La gestion ha llegado a una conclusion');
```



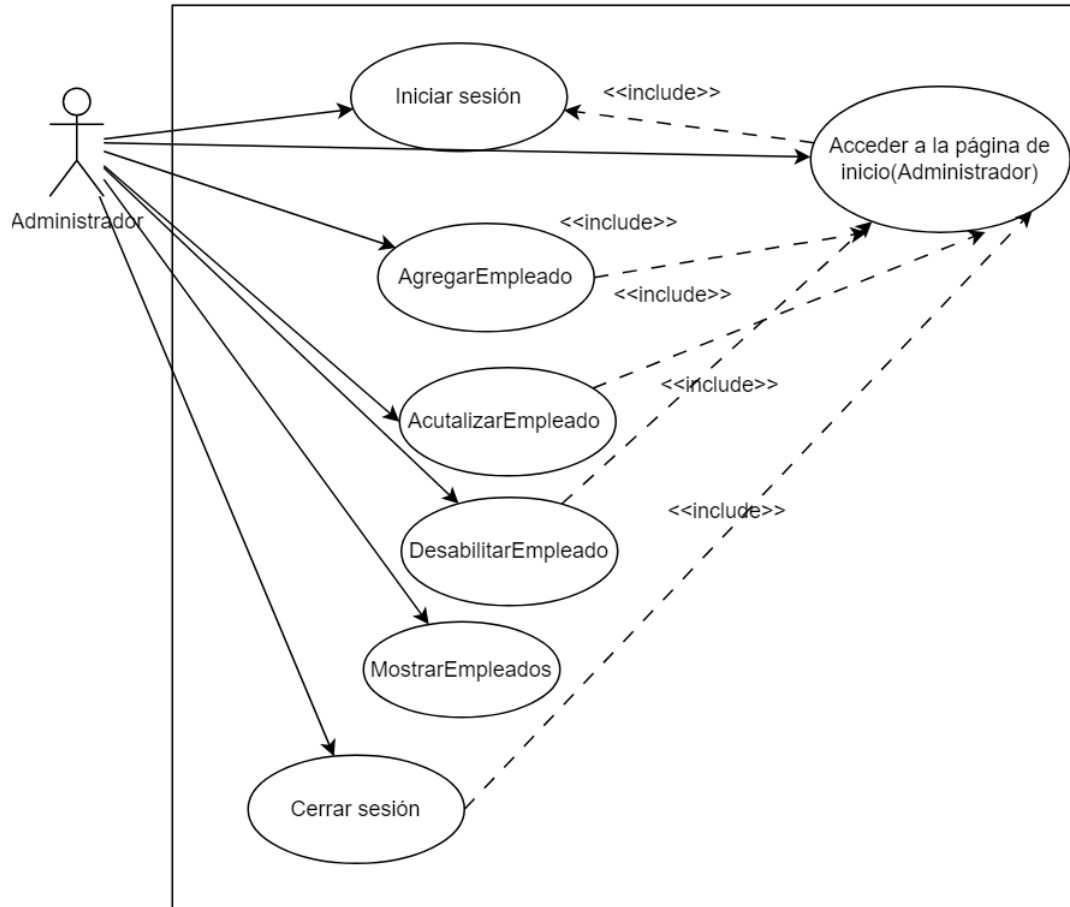
```
INSERT INTO tipos_empleados VALUES
```

```
(1, 'administrador', 'el admin'),
```

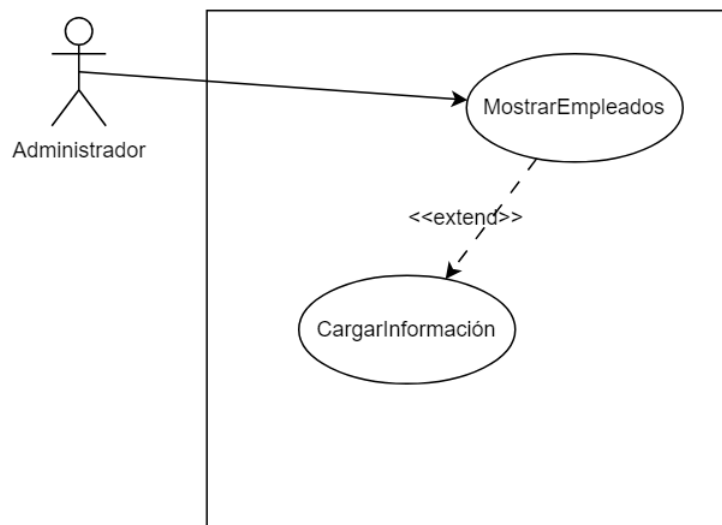
```
(2, 'empleado', 'no admin');
```

# Casos de Uso

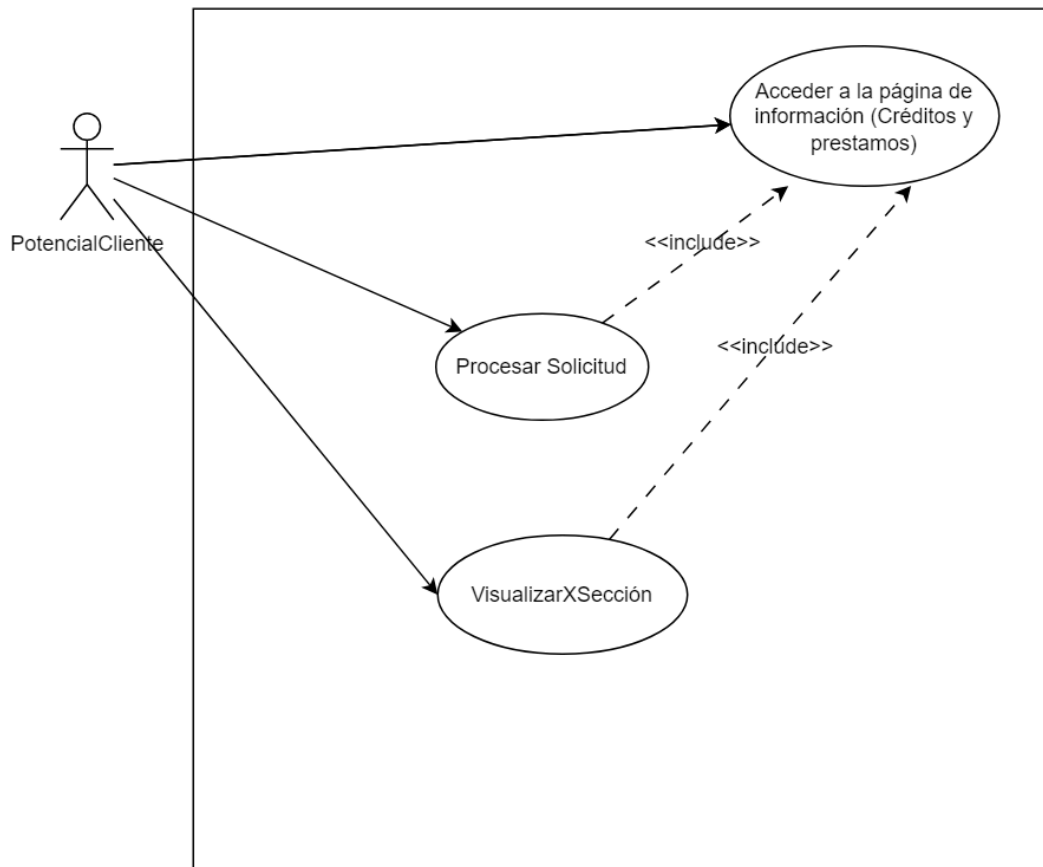
## Gestión de empleados Nivel 1



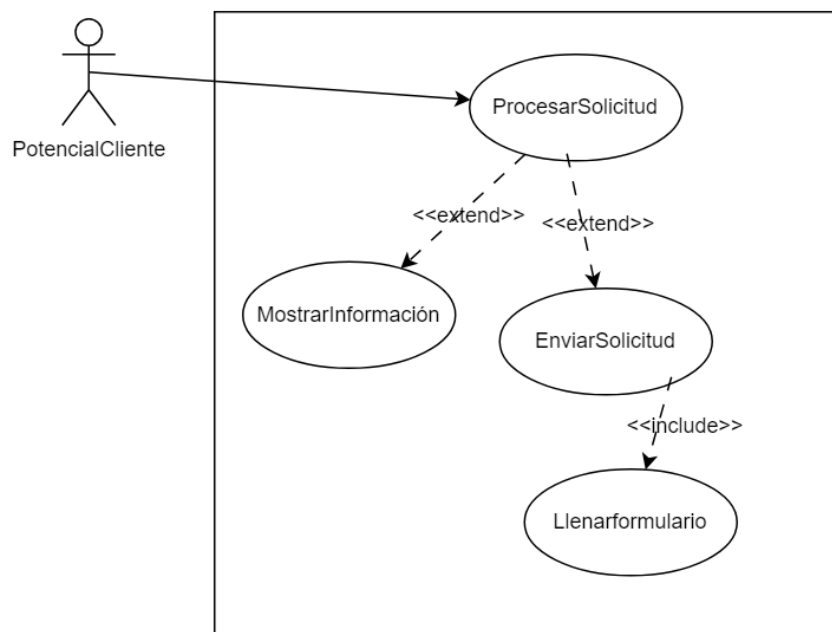
## Gestión de empleados Nivel 2

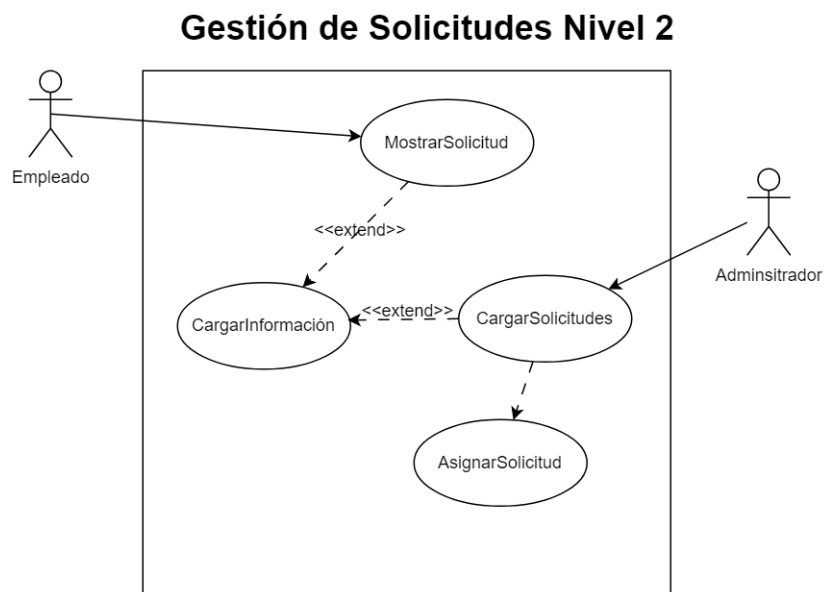
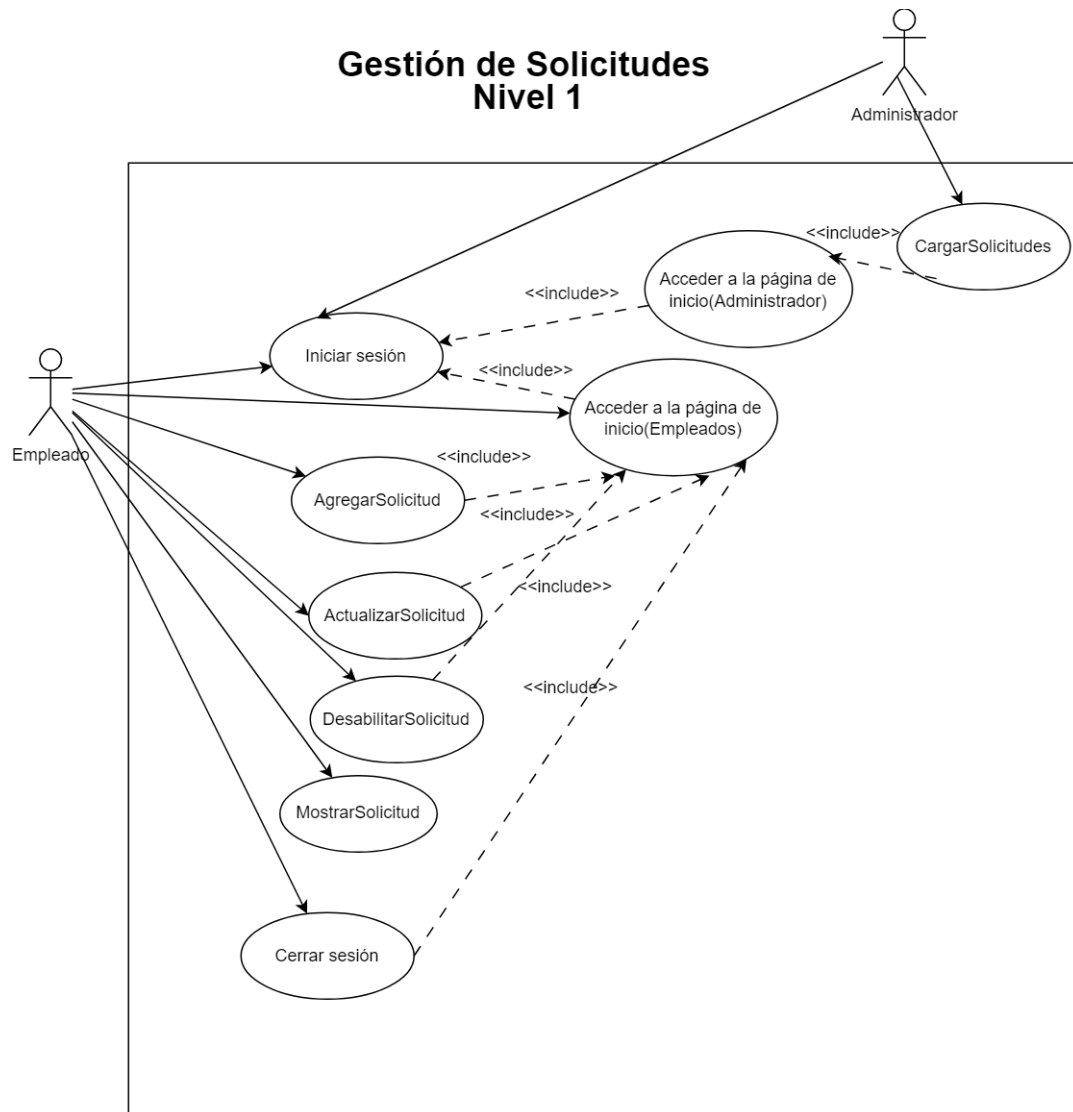


## Generar Solicitud Nivel 1

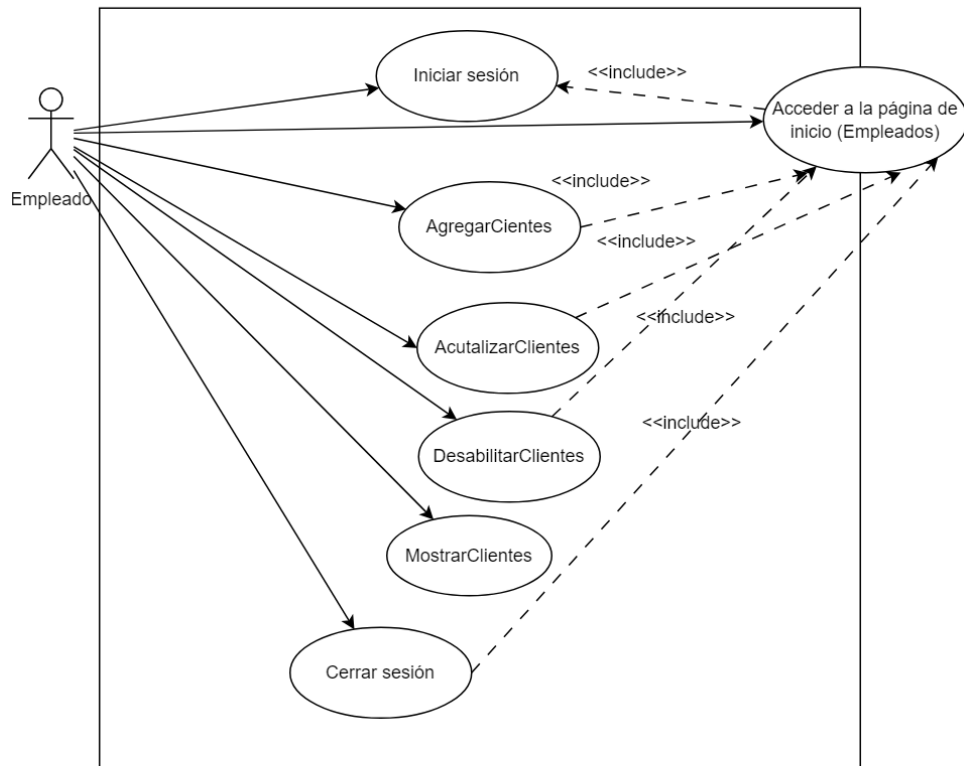


## Procesar solicitud Nivel 2

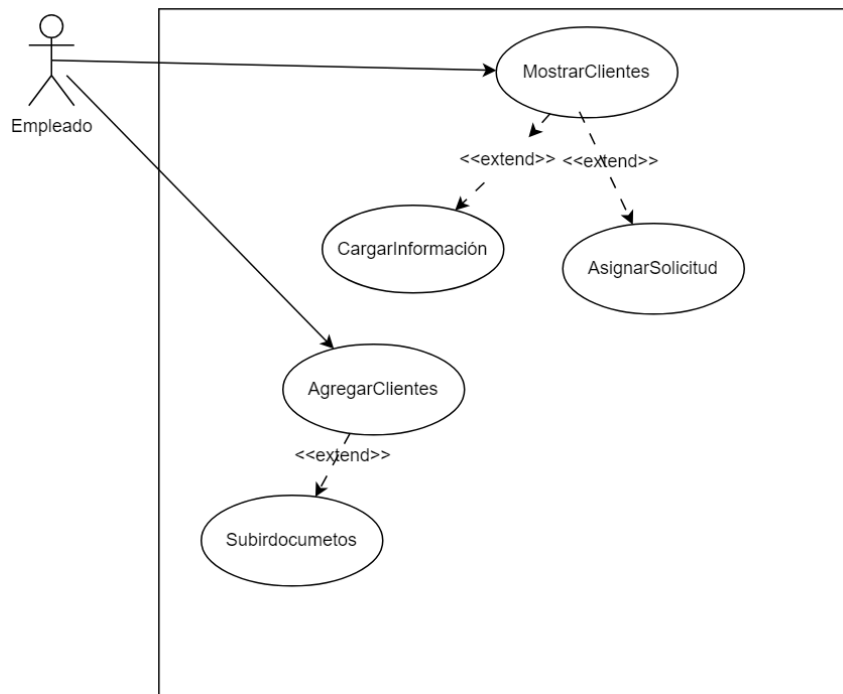




## Gestión de clientes Nivel 1

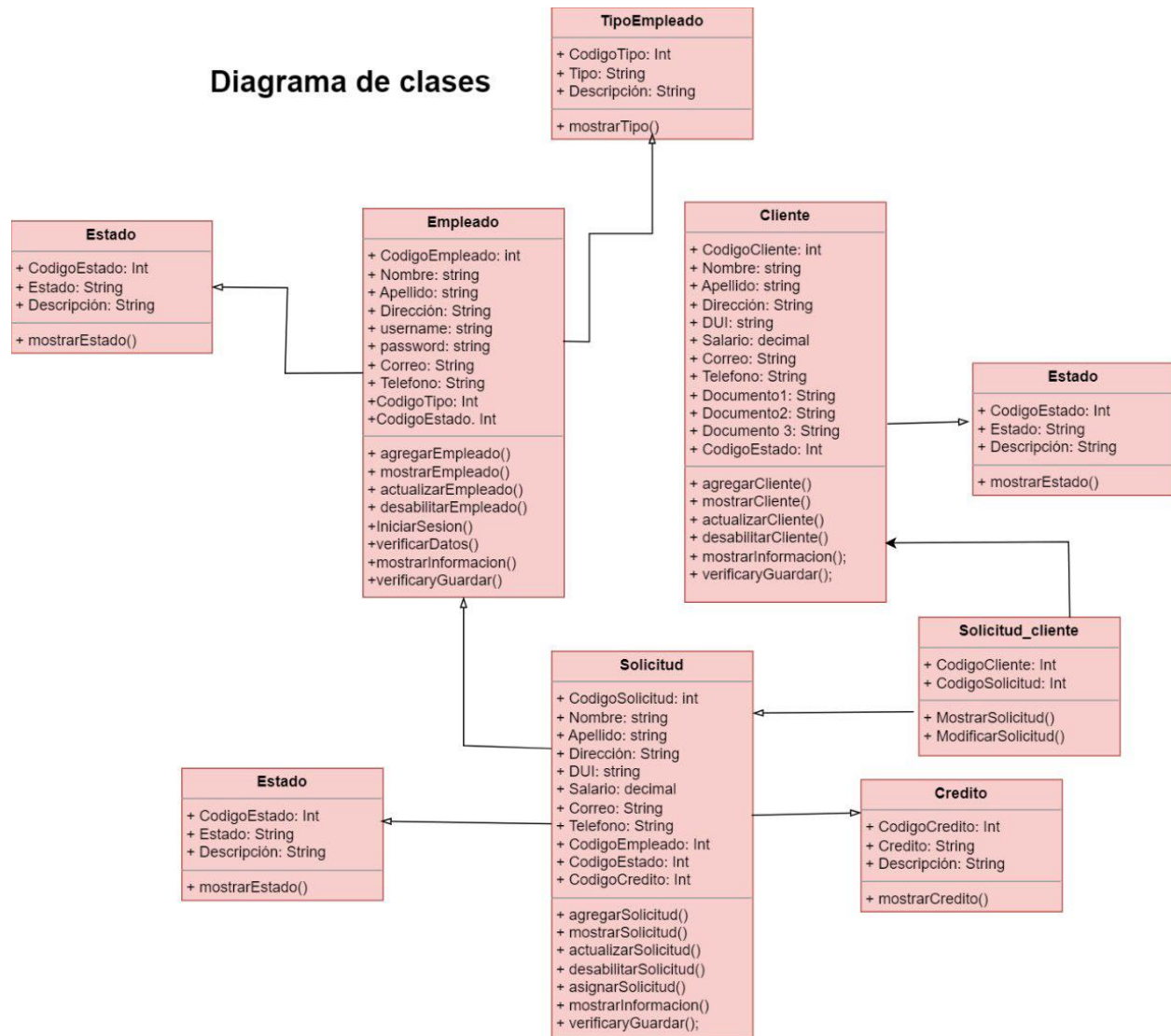


## Gestión de clientes Nivel 2



# Diagrama de Clases

Diagrama de clases



# Diagrama de Secuencia

