

Unilasallista Corporación Universitaria

Facultad de Ingenierías

Ingeniería Informática

Ingeniería del Software I

Manual Técnico de OptiStock

Alejandro Arango, Juan Pablo Sánchez, Simón Marulanda, Samuel Valencia

Feibert Alirio Guzmán

05 de Junio, 2025

Índice

1. [Arquitectura del sistema](#)
2. [Requisitos del sistema](#)
3. [Instalación y configuración](#)
4. [Estructura del proyecto](#)
5. [Funcionalidades principales](#)
6. [Mantenimiento y actualizaciones](#)
7. [Contacto y soporte](#)

1. Arquitectura del Sistema

La aplicación sigue el patrón **MVC (Modelo-Vista-Controlador)**, característico de Rails:

- **Modelos (`app/models`)**: Definen las estructuras de datos y las relaciones entre ellas.
- **Vistas (`app/views`)**: Plantillas HTML que presentan la información al usuario.
- **Controladores (`app/controllers`)**: Gestionan la lógica de la aplicación y la interacción entre modelos y vistas.

2. Requisitos del sistema

- **Ruby:** Versión especificada en `.ruby-version`.
- **Rails:** Versión especificada en `Gemfile`.
- **Base de Datos:** Configurada en `config/database.yml` (PostgreSQL 14).
- **Dependencias:** Listadas en `Gemfile`.

3. Instalación y Configuración

Clonar el repositorio:

```
Unset  
git clone https://github.com/SamuelV19/software\_gestion.git  
cd software_gestion
```

Instalar dependencias:

```
Unset  
bundle install
```

Configurar la base de datos:

```
Unset  
rails db:create  
rails db:migrate
```

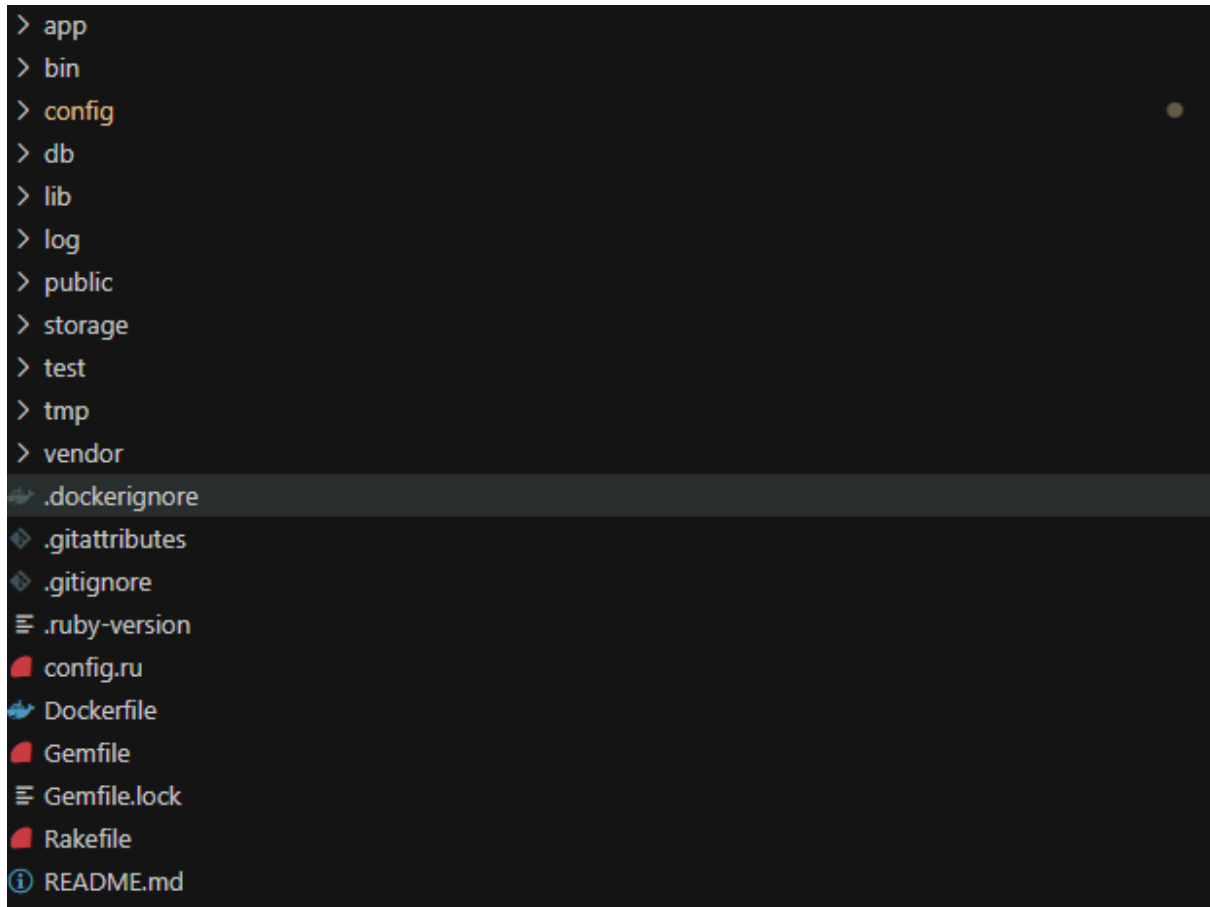
Iniciar el servidor:

```
Unset  
rails server
```

La aplicación estará disponible en <http://localhost:3000>.

4. Estructura del Proyecto

- **app/**: Contiene los modelos, vistas y controladores.
- **config/**: Archivos de configuración de la aplicación y la base de datos.
- **db/**: Migraciones y esquema de la base de datos.
- **public/**: Archivos estáticos accesibles públicamente.
- **Gemfile**: Lista de gemas (dependencias) utilizadas.
- **Rakefile**: Define tareas automatizadas para la aplicación.



A screenshot of a file explorer interface with a dark theme. The left sidebar shows a tree view of the project structure. The main pane displays the contents of the selected directory. The tree view includes folders like 'app', 'bin', 'config', 'db', 'lib', 'log', 'public', 'storage', 'test', 'tmp', and 'vendor', as well as files like '.dockerignore', '.gitattributes', '.gitignore', '.ruby-version', 'config.ru', 'Dockerfile', 'Gemfile', 'Gemfile.lock', 'Rakefile', and 'README.md'. The 'config' folder is currently selected and highlighted.

```
> app
> bin
> config
> db
> lib
> log
> public
> storage
> test
> tmp
> vendor
.dockerignore
.gitattributes
.gitignore
.ruby-version
config.ru
Dockerfile
Gemfile
Gemfile.lock
Rakefile
README.md
```

5. Funcionalidades Principales

- **Gestión de Productos:** Crear, editar y eliminar productos, incluyendo detalles como nombre, precio y stock.
- **Registro de Ventas:** Registrar ventas, actualizar el stock y generar facturas.
- **Informes:** Generar informes de ventas diarios, semanales y mensuales.
- **Autenticación:** Sistema de inicio de sesión para administradores y usuarios comunes.

6. Mantenimiento y Actualizaciones

Actualización de dependencias:

Unset

```
bundle update
```

Migraciones de base de datos:

Unset

```
rails db:migrate
```

Despliegue: Configurar el entorno de producción y utilizar herramientas como *Render* o *Heroku* para el despliegue.

7. Contacto y soporte

- **GitHub:** [SamuelV19](#)
- **Repositorio:** https://github.com/SamuelV19/software_gestion.git
- **Email:** svalencia64@unilasallista.edu.co
- **Horario de atención:** Lunes a Viernes, 9:00 AM - 6:00 PM (hora Colombia)