

Documentación de Diseño del Software Go Fast Go

Introducción:

1. Interfaces Graficas

Prototipos y Especificaciones de diseño de la Interfaz de Usuario

Las interfaces de usuario de Go Fast Go están diseñadas para ser intuitivas y fáciles de usar, asegurando una experiencia eficiente para administradores domiciliarios y restaurantes. Se han desarrollado prototipos en Figma que incluyen:

- **Pantalla de inicio de sesión:** permite a los usuarios autenticarse según su rol
- **Panel de Administrador:** visualización de reportes, validación de horas trabajadas y gestión de usuarios
- **Interfaz para Domiciliarios:** Registro de horas mediante escaneo de código generado por el sistema y visualización de historias

Las especificaciones de diseño siguen principios de usabilidad y accesibilidad, con una paleta de colores alineada a la identidad corporativa y elementos UI optimizados para dispositivos de escritorio y Android.

GoFastApp



Iniciar Sesión

Usuario

Contraseña

Ingresar

© 2025 Go Fast Go. Todos los derechos reservados.



Panel de Administrador

Gestionar Usuarios

Gestionar Personas

Gestionar Administradores

Iniciar Tarea

Gestionar Tiendas

Asignar Tareas

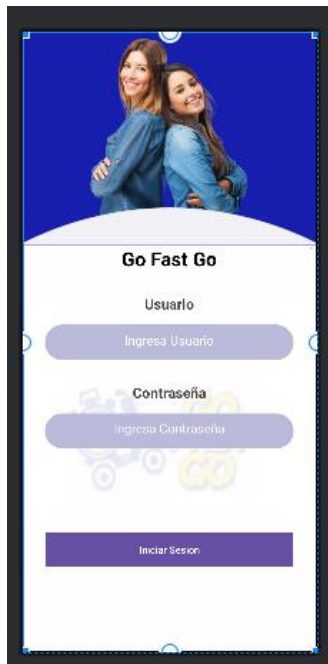
Código

Seleccione una opción para empezar

Gestión de Personas

Nombre	<input type="text"/>
Apellido	<input type="text"/>
Documento	<input type="text"/>
Dirección	<input type="text"/>
Teléfono	<input type="text"/> Completa este campo
Email	<input type="text"/>
Fecha de Nacimiento	<input type="text"/>

16/07



2. Patrones de Diseño

2.1 Garantizamos que solo haya una instancia de la conexión a la base de datos.

```
3. spring.application.name=GOAPPv2

server.port = 9000
spring.datasource.url=jdbc:mysql://localhost:3306/GOAPP
spring.datasource.username=root
spring.datasource.password=
spring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true

logging.level.org.springframework=DEBUG

spring.mvc.throw-exception-if-no-handler-found=true
spring.web.resources.add-mappings=false

# Configuración de zona horaria para Colombia (Bogotá)
spring.jackson.time-zone=America/Bogota
spring.jpa.properties.hibernate.jdbc.time_zone=America/Bogota

spring.jpa.properties.hibernate.format_sql=true
logging.level.org.hibernate.SQL=DEBUG
logging.level.org.hibernate.type.descriptor.sql.BasicBinder=TRACE
```

3.1.

4. Modelo Arquitectónico:

Es sistema se basa en una arquitectura modelo, vista, controlador, para separar la lógica del negocio, la presentación y la gestión de datos. La estructura es la siguiente:

- I. **-Modelo:** gestiona la lógica del negocio y la comunicación con la base de datos

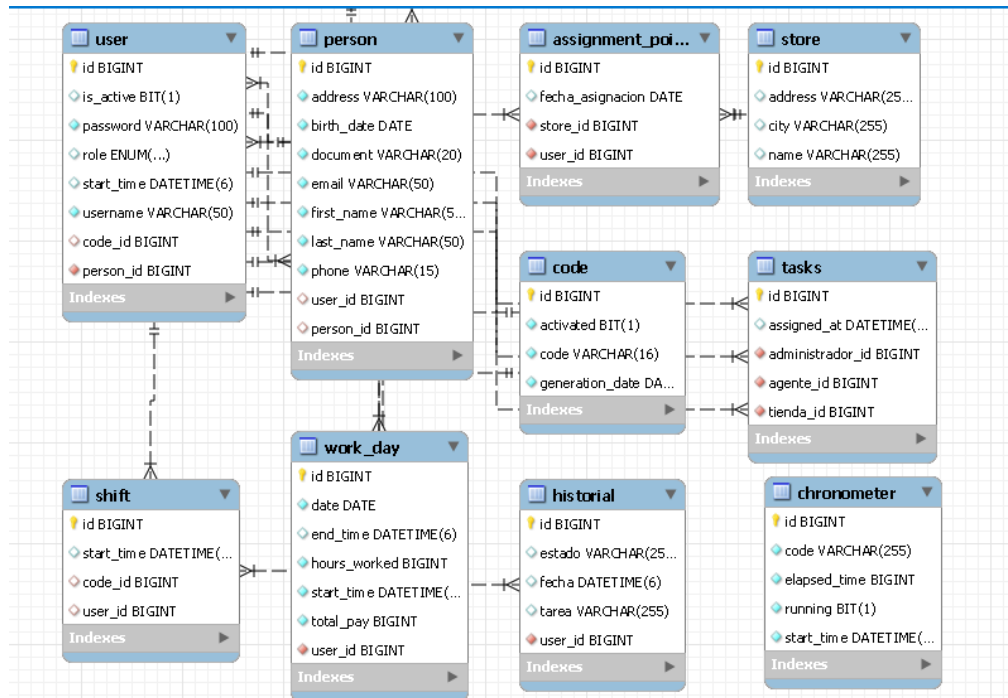
- II. **-Vista:** Representa la interfaz de usuario desarrollado con Android Java, y JavaScript
- III. **-Controlador:** maneja las solicitudes del usuario, interactúa con el modelo y actualiza la vista

El backend está implementado con java, Spring Boot para el api, mientras que la base de datos utiliza My SQL para la gestión de bases de datos relacionales.

Modelo de datos:

El sistema utiliza una base de datos relacional (My SQL) con las siguientes entidades y relaciones. Se utiliza My SQL con claves primarias y foráneas para mantener la integridad de los datos.

-
- **User**
 - **Person**
 - **Assginent_point**
 - **Store**
 - **Code**
 - **Task**
 - **Work_day**
 - **historial**
-



Conclusiones:

El diseño del sistema Go Fast Go sigue principios de modularidad, escalabilidad y facilidad de mantenimiento. Su arquitectura permite una administración eficiente de los procesos de entrega y pago.