

# Documentación Técnica Detallada - BusIA

## 1. Stack Tecnológico

### 1.1 Backend

- **Framework Principal:** Django 5.0
  - Framework web de alto nivel basado en Python
  - Sigue el patrón arquitectónico MVT (Model-View-Template)

### 1.2 APIs y REST Framework

- **Django REST Framework**
  - Utilizado para crear APIs RESTful
  - Incluye serialización de datos
  - Sistema de autenticación y permisos
- **JWT (JSON Web Tokens)**
  - Implementado con `rest_framework_simplejwt`
  - Utilizado para autenticación segura
  - Gestión de tokens de acceso y refresco

### 1.3 Base de Datos

- **SQLite**
  - Sistema de base de datos relacional
  - Utilizado en desarrollo
  - Ideal para aplicaciones de tamaño medio

### 1.4 Frontend

- **HTML5/CSS3**
  - Estructura y estilos base
  - Templates de Django
- **JavaScript**
  - Interactividad del lado del cliente
  - Manejo de formularios
  - Validaciones en tiempo real
- **Bootstrap**
  - Framework CSS para diseño responsivo
  - Componentes predefinidos
  - Sistema de grid

### 1.5 Seguridad

- **CORS (Cross-Origin Resource Sharing)**
  - Implementado con `django-cors-headers`
  - Control de acceso HTTP
  - Configuración de dominios permitidos
- **CSRF Protection**
  - Protección contra ataques Cross-Site Request Forgery
  - Tokens CSRF en formularios

## 2. Dependencias Principales

### 2.1 Django y Extensiones

```
django==5.0                # Framework web principal
djangorestframework        # API REST framework
djangorestframework-simplejwt # Autenticación JWT
django-cors-headers        # Manejo de CORS
```

### 2.2 Utilidades y Herramientas

```
Pillow                    # Procesamiento de imágenes
python-dateutil           # Manejo avanzado de fechas
```

## 3. Estructura de Aplicaciones

### 3.1 Módulo de Usuarios (usuarios)

- Gestión de autenticación
- Perfiles de usuario
- Registro y login
- Recuperación de contraseña

### 3.2 Módulo de Pasajes (pasajes)

- Sistema de reservas
- Gestión de tickets
- Historial de viajes
- Procesamiento de pagos

### 3.3 Módulo de Empresas (busempresas)

- Gestión de empresas de transporte
- Administración de flotas
- Rutas y horarios

### 3.4 Landing Page (landing)

- Página principal

- Información del servicio
- Contacto

## 4. APIs y Endpoints

### 4.1 API de Usuarios

```
POST /api/usuarios/registro/      # Registro de usuarios
POST /api/usuarios/login/         # Inicio de sesión
GET  /api/usuarios/perfil/        # Obtener perfil
PUT  /api/usuarios/perfil/        # Actualizar perfil
```

### 4.2 API de Pasajes

```
GET  /api/pasajes/                # Listar pasajes disponibles
POST /api/pasajes/reserva/        # Crear reserva
GET  /api/pasajes/historial/      # Historial de viajes
```

### 4.3 API de Empresas

```
GET  /api/empresas/              # Listar empresas
GET  /api/empresas/rutas/        # Obtener rutas
POST /api/empresas/horarios/     # Gestionar horarios
```

## 5. Configuraciones

### 5.1 Configuración de Correo

- Sistema de correo electrónico configurado
- Notificaciones automáticas
- Confirmaciones de reserva

### 5.2 Configuración de Logging

- Registro detallado de actividades
- Logs separados por módulos
- Niveles de logging configurables

### 5.3 Configuración de Seguridad

```
# Configuraciones JWT
SIMPLE_JWT = {
    'ACCESS_TOKEN_LIFETIME': timedelta(minutes=60),
    'REFRESH_TOKEN_LIFETIME': timedelta(days=1),
    'ROTATE_REFRESH_TOKENS': False,
}
# Configuraciones REST Framework
REST_FRAMEWORK = {
    'DEFAULT_AUTHENTICATION_CLASSES': [
```

```
        'rest_framework_simplejwt.authentication.JWTAuthentication',  
        'rest_framework.authentication.SessionAuthentication',  
    ],  
}
```

## 6. Herramientas de Desarrollo

### 6.1 Entorno Virtual

- Virtualenv para aislamiento de dependencias
- Requirements.txt para gestión de paquetes

### 6.2 Herramientas de Debug

- Django Debug Toolbar
- Logging configurado
- Console debugging

## 7. Seguridad Implementada

### 7.1 Autenticación

- Sistema JWT para APIs
- Sesiones de Django para web
- Tokens de acceso y refresco

### 7.2 Autorización

- Permisos basados en roles
- Decoradores de permisos
- Middleware de autenticación

### 7.3 Protección de Datos

- Encriptación de contraseñas
- Sanitización de datos
- Validación de formularios

## 8. Mantenimiento y Escalabilidad

### 8.1 Backup y Recuperación

- Sistema de respaldo de base de datos
- Logs de actividad
- Procedimientos de recuperación

### 8.2 Monitoreo

- Logging de errores
- Monitoreo de rendimiento
- Alertas configuradas

### 8.3 Escalabilidad

- Arquitectura modular
- Cacheo implementado
- Optimización de consultas

## 9. Integración y Despliegue

### 9.1 Control de Versiones

- Git para control de versiones
- Branches para desarrollo
- Sistema de merge requests

### 9.2 Testing

- Tests unitarios
- Tests de integración
- Cobertura de código

### 9.3 Despliegue

- Guía de despliegue
- Configuraciones de producción
- Scripts de automatización