

Centro Universitario de Occidente
Análisis y Diseño de Sistemas 1
Segundo Semestre 2024



Manual Técnico QuickAppointment

201830498 - Oscar Antonio de León Urizar

201832069 - Marcos Andrés Aguiare Bravo

Índice

Índice.....	2
Manual Técnico.....	4
Requisitos del Sistema.....	4
Descripción del Proyecto.....	4
Características Principales.....	4
Tecnologías Utilizadas.....	5
Funcionalidades Clave.....	6
Estructura del Proyecto.....	7
Backend.....	7
- Controllers:.....	7
- Entities.....	10
- Repositories.....	10
Frontend.....	10
Components.....	11
Services.....	11
Base de Datos.....	12

Manual Técnico

Requisitos del Sistema

El software puede ser utilizado en cualquier sistema operativo, ya que es un proyecto web. Está desarrollado utilizando TypeScript para el frontend, y en el backend se utilizó Java. A continuación, se listan los requisitos principales:

- Frontend: Angular 17.x+
- Backend: Java 21 con SpringBoot
- Base de datos: MySQL
- Herramientas de desarrollo: IntelliJ, Visual Studio Code, DataGrip
- Navegadores soportados: Google Chrome, Mozilla Firefox, Microsoft Edge (últimas

versiones)

Descripción del Proyecto

El proyecto consiste en desarrollar una aplicación web para la gestión de citas y reservas orientada a negocios como clínicas dentales, salones de belleza, y alquiler de canchas deportivas, permitiendo a los clientes organizar sus citas de acuerdo con la disponibilidad del negocio. La aplicación permitirá personalizar elementos como el nombre y logo del negocio, integrándose en la interfaz y en los comprobantes generados. Además, gestionará roles y permisos para garantizar la seguridad de la información.

Características Principales

1. **Gestión de Citas y Calendario:** Los clientes pueden visualizar un horario con la disponibilidad del negocio para agendar citas. Los administradores pueden configurar los horarios de atención, duración de los servicios, y la cantidad de empleados o recursos disponibles simultáneamente. Se ofrece la opción de asignación aleatoria o selección de personal específico, permitiendo un balance de la carga de trabajo.

2. **Personalización del Negocio:** Los administradores pueden configurar el nombre, logo y horarios del negocio, que se verán reflejados en la interfaz y los comprobantes. Además, pueden definir el número de empleados o recursos disponibles en cada horario.
3. **Gestión de Roles y Permisos:** Dos roles principales, Administrador y Cliente, controlan las acciones dentro del sistema. Los administradores pueden también definir roles personalizados, como ayudantes, con permisos específicos para la gestión de citas, empleados y reportes.
4. **Seguridad y autenticación:** Implementación de encriptación de contraseñas y autenticación segura de usuarios mediante correo electrónico. También se asegura la unicidad de los datos clave (NIT, CUI, teléfono y correo), y se incluye un proceso de recuperación de contraseña por correo electrónico.
5. **Estadísticas y Reportes:** Un dashboard de estadísticas permite a los administradores visualizar el rendimiento del negocio en diferentes periodos. Además, se pueden generar reportes personalizados sobre las ventas, clientes frecuentes, servicios más solicitados y disponibilidad de recursos, con la opción de exportarlos en PDF, Excel o imagen.
6. **Configuración de horarios:** El administrador puede definir los días laborables y el tiempo estimado de cada cita. Los calendarios de empleados y recursos estarán sincronizados en tiempo real, optimizando la organización diaria.

Tecnologías Utilizadas

- **Backend:** Spring Boot
- **Base de Datos:** MySQL
- **Frontend:** Angular
- **API REST:** Para la comunicación entre frontend y backend
- **Testing:** JUnit y Mockito (con una cobertura de código del 80%, validada mediante JaCoCo)
- **CI/CD:** Jenkins, para integrar e implementar la aplicación automáticamente.

Funcionalidades Clave

1. **Interfaz de Calendario Interactivo:** Los clientes pueden agendar citas y los administradores configurar horarios de forma visual e intuitiva.
2. **Gestión de Empleados y Recursos:** Personalizable según el tipo de negocio, permitiendo varias citas en paralelo.
3. **Autenticación Segura y Recuperación de contraseñas:** Seguridad mediante encriptación y verificación por correo.
4. **Dashboard de Estadísticas y Reportes Exportables:** Visualización del desempeño del negocio con capacidad de exportar los datos.
5. **Personalización del Negocio:** Configuración del nombre y logo visibles en la aplicación y en los comprobantes.

Estructura del Proyecto

Backend

- Controllers:
 - **AppointmentController**: Controla lo relacionado a las citas.

```
// Guarda una cita en base de datos
createAppointment(@RequestBody AppointmentDto appointmentDto)

// Obtiene las citas en base al id de un usuario.
getAppointmentsByUser(@PathVariable Integer fkUser)

// Obtiene las citas en base a una fecha
getAppointmentsByDate(@PathVariable Date date)
// Obtiene las citas próximas de un usuario
getMyAppointments()
```

- **CompanyController**: Controla lo relacionado a los negocios y sus horarios.

```
// Guarda un negocio en la base de datos
createCompany(@RequestBody CompanyDto companyDto)

// Devuelve todas los negocios disponibles
getAllCompanies()

// Devuelve una compañía en base a su id
getCompanyById(@PathVariable Long id)

// Actualiza campos de un negocio
updateCompany(@PathVariable Long id, @RequestBody
UpdateNameDescCompanyDto updateCompanyDto)

// Actualiza la imagen del negocio
updateCompanyLogo(@PathVariable Long id, @RequestBody
UpdateCompanyLogoDto logoDto)
// Elimina un negocio de la base de datos.
```

```

deleteCompanyById(@PathVariable Long id)

// Devuelve el horario de la compañía
getCompanySchedule(@PathVariable Integer fkCompany)

// Guarda el horario de un negocio en la base de datos
updateCompanySchedule(
    @PathVariable Integer fkCompany,
    @RequestBody List<ScheduleDto> newSchedules)

```

- **EmployeeController:** Controla lo relacionado a los empleados.

```

// Guarda un empleado en la base de datos
createEmployee(@RequestBody EmployeeDto employeeDto)

// Obtiene los empleados en base a su negocio
getEmployeesByCompany(@PathVariable Integer fkCompany)

// Devuelve el empleado en base a su id
getEmployeeById(@PathVariable Integer id)

// Actualiza los datos del empleado
updateEmployee(@PathVariable Integer id, @RequestBody EmployeeDto
employeeDto)

// Elimina un empleado de la base de datos
deleteEmployee(@PathVariable Integer id)

```

- **FileUploadController:** Controla el guardado o eliminación de imágenes.

```

// Guarda una imagen en la base de datos
uploadImage(@RequestParam("file") MultipartFile file)

```

- **PlaceController:** Controla lo relacionado a los recursos de cada negocio.

```

// Guarda un recurso en la base de datos
createPlace(@RequestBody PlaceDto placeDto)

// Devuelve los recursos de un negocio
getPlacesByCompany(@PathVariable Integer fkCompany)

// Devuelve un recurso en base a un id
getPlaceById(@PathVariable Integer id)

// Actualiza un recurso
updatePlace(@PathVariable Integer id, @RequestBody PlaceDto placeDto)

// Elimina un recurso de la base de datos
deletePlace(@PathVariable Integer id)

```

- **TypeAppointmentController:** Controla lo relacionado a los tipos de servicios.

```

// Guarda un servicio en la base de datos
createTypeAppointment(@RequestBody TypeAppointmentDto typeAppointmentDto)

// Devuelve los servicios de un negocio
getTypeAppointmentsByCompany(@PathVariable Integer fkCompany)

// Devuelve un servicio en base a su id
getTypeAppointmentById(@PathVariable Integer id)

// Actualiza los datos de un servicio
updateTypeAppointment(
    @PathVariable Integer id,
    @RequestBody TypeAppointmentDto typeAppointmentDto)

// Elimina un servicio de la base de datos
deleteTypeAppointment(@PathVariable Integer id)

```

- **UserController:** Controla lo relacionado a los usuarios.


```
// Registra un usuario en la base de datos
registerUser(@RequestBody UserDto userDto)

// Comprueba que un usuario este en base de datos
loginUser(@RequestBody LoginDto loginDto)

// Obtiene las paginas que puede obtener un usuario en base a sus roles
getPagesByUserId(@PathVariable("userId") Integer userId)
```

- Entities

- Appointment
- Company
- Company
- CompanySchedule
- Employee
- Place
- TypeAppointment
- User
- UserHasRole

- Repositories

- AppointmentRepository
- CompanyRepository
- CompanyScheduleRepository
- EmployeeRepository
- PlaceRepository
- TypeAppointmentRepository
- UserHasRoleRepository
- UserRepository

Frontend

Components

- **admin/**
 - add-company
 - admin-home
 - config-schedule
 - dashboard
 - edit-company-settings
 - edit-schedule-modal
 - employee-form
 - home-company
 - place-form
 - role
 - role-form
 - services-form
 - users-form
 - users
 - view-employees
 - view-places
 - view-services
- **client/**
 - client-home
 - company
 - court-reservation
 - my-reservation
 - reservation
- **commons/**
 - banner
 - forgot-password
 - home
 - login
 - navbar
 - navbar-guest

- not-found
- register-modal

Services

- admin.service.ts
- client.service.ts
- local-storage.service
- user.service

Base de Datos

Tablas Principales

company

- **Descripción:** Almacena la información sobre las compañías registradas en el sistema.
- **Campos principales:**
 - **id:** Identificador único de la compañía.
 - **name:** Nombre de la compañía (máximo 100 caracteres).
 - **description:** Descripción detallada de la compañía.
 - **logo:** Ruta de la imagen del logo de la compañía.
 - **court_rental:** Indica si la compañía ofrece alquiler de canchas (1 para sí, 0 para no).
 - **is_available:** Estado de disponibilidad de la compañía (1 para disponible, 0 para no disponible).
 - **created_at:** Fecha y hora de creación del registro.

week_day

- **Descripción:** Almacena los días de la semana.
- **Campos principales:**
 - **id:** Identificador único del día de la semana.
 - **day_name:** Nombre del día de la semana (máximo 15 caracteres).

type_appointment

- **Descripción:** Almacena los tipos de citas disponibles en el sistema.
- **Campos principales:**
 - **id:** Identificador único del tipo de cita.

- **name:** Nombre del tipo de cita (máximo 100 caracteres).
- **description:** Descripción detallada del tipo de cita.
- **duration:** Duración de la cita en minutos.
- **price:** Precio del servicio de la cita.
- **is_available:** Estado de disponibilidad del tipo de cita (1 para disponible, 0 para no disponible).
- **created_at:** Fecha y hora de creación del registro.
- **FK_Company:** Clave foránea que relaciona el tipo de cita con una compañía específica.
- **Relaciones:**
 - Relacionado con la entidad **company** para identificar la empresa a la que pertenece el tipo de cita.

employee

- **Descripción:** Almacena la información de los empleados de las compañías.
- **Campos principales:**
 - **id:** Identificador único del empleado.
 - **first_name:** Nombre del empleado (máximo 100 caracteres).
 - **last_name:** Apellido del empleado (máximo 100 caracteres).
 - **dpi:** Documento Personal de Identificación (máximo 15 caracteres).
 - **date_birth:** Fecha de nacimiento del empleado.
 - **created_at:** Fecha y hora de creación del registro.
 - **is_available:** Estado de disponibilidad del empleado (1 para disponible, 0 para no disponible).
 - **FK_Company:** Clave foránea que relaciona al empleado con una compañía específica.
- **Relaciones:**
 - Relacionado con la entidad **company** para identificar la empresa a la que pertenece el empleado.

module

- **Descripción:** Almacena la información de los módulos disponibles en el sistema.
- **Campos principales:**
 - **id:** Identificador único del módulo.
 - **name:** Nombre del módulo (máximo 100 caracteres).
 - **direction:** Dirección o ruta asociada al módulo (máximo 100 caracteres).

role

- **Descripción:** Almacena los roles de usuarios en el sistema.
- **Campos principales:**
 - **id:** Identificador único del rol.
 - **name:** Nombre del rol (máximo 100 caracteres).
 - **description:** Descripción del rol, detallando sus funciones y permisos.

page

- **Descripción:** Almacena la información de las páginas disponibles en el sistema.
- **Campos principales:**
 - **id:** Identificador único de la página.
 - **name:** Nombre de la página (máximo 100 caracteres).
 - **direction:** Dirección o ruta asociada a la página (máximo 100 caracteres).
 - **isAvailable:** Estado de disponibilidad de la página (1 para disponible, 0 para no disponible).
 - **FK_Module:** Clave foránea que relaciona la página con un módulo específico.
- **Relaciones:**
 - Relacionado con la entidad **module** para especificar el módulo al que pertenece la página.

user

- **Descripción:** Almacena la información de los usuarios registrados en el sistema.
- **Campos principales:**
 - **id:** Identificador único del usuario.
 - **email:** Correo electrónico del usuario (máximo 100 caracteres).
 - **username:** Nombre de usuario (máximo 100 caracteres).
 - **password:** Contraseña encriptada del usuario (máximo 200 caracteres).
 - **auth_token:** Token de autenticación para el acceso seguro.
 - **is_verified:** Estado de verificación del usuario (1 para verificado, 0 para no verificado).
 - **created_at:** Fecha y hora de creación del registro.

place

- **Descripción:** Almacena la información de los lugares asociados a las compañías en el sistema.
- **Campos principales:**

- **id**: Identificador único del lugar.
- **name**: Nombre del lugar (máximo 100 caracteres).
- **place**: Ubicación o dirección específica del lugar (máximo 100 caracteres).
- **is_available**: Estado de disponibilidad del lugar (1 para disponible, 0 para no disponible).
- **created_at**: Fecha y hora de creación del registro.
- **FK_Company**: Clave foránea que relaciona el lugar con una compañía específica.
- **Relaciones**:
 - Relacionado con la entidad **company** para identificar la empresa a la que pertenece el lugar.

company_schedule

- **Descripción**: Almacena los horarios de apertura y cierre de las compañías en días específicos.
- **Campos principales**:
 - **id**: Identificador único del horario de la compañía.
 - **FK_Company**: Clave foránea que relaciona el horario con una compañía específica.
 - **FK_Day**: Clave foránea que especifica el día de la semana.
 - **opening_time**: Hora de apertura de la compañía en el día especificado.
 - **closing_time**: Hora de cierre de la compañía en el día especificado.
- **Relaciones**:
 - Relacionado con las entidades **company** y **week_day** para definir el horario de cada empresa en días específicos.

user_has_role

- **Descripción**: Almacena la relación entre los usuarios y los roles asignados en el sistema.
- **Campos principales**:
 - **id**: Identificador único de la relación entre usuario y rol.
 - **FK_User**: Clave foránea que identifica al usuario.
 - **FK_Role**: Clave foránea que identifica al rol asignado al usuario.
- **Relaciones**:
 - Relacionado con las entidades **user** y **role** para gestionar los permisos y niveles de acceso de cada usuario en el sistema.

role_has_page

- **Descripción**: Almacena la relación entre los roles y las páginas, definiendo los permisos de cada rol sobre las páginas del sistema.

- **Campos principales:**
 - **id:** Identificador único de la relación entre rol y página.
 - **FK_Role:** Clave foránea que identifica al rol.
 - **FK_Page:** Clave foránea que identifica la página.
 - **allowCreate:** Permiso para crear contenido en la página (1 para permitir, 0 para denegar).
 - **allowEdit:** Permiso para editar contenido en la página (1 para permitir, 0 para denegar).
 - **allowDelete:** Permiso para eliminar contenido en la página (1 para permitir, 0 para denegar).
- **Relaciones:**
 - Relacionado con las entidades **role** y **page** para gestionar los permisos de acceso y edición de cada rol en las páginas del sistema.

appointment

- **Descripción:** Almacena la información de las citas agendadas por los usuarios en el sistema.
- **Campos principales:**
 - **id:** Identificador único de la cita.
 - **FK_User:** Clave foránea que identifica al usuario que agendó la cita.
 - **date:** Fecha de la cita.
 - **hour:** Hora de la cita.
 - **is_confirmed:** Estado de confirmación de la cita (1 para confirmada, 0 para no confirmada).
 - **is_canceled:** Estado de cancelación de la cita (1 para cancelada, 0 para no cancelada).
 - **FK_Employee:** Clave foránea que identifica al empleado asignado a la cita.
 - **FK_Type:** Clave foránea que identifica el tipo de cita.
 - **FK_Place:** Clave foránea que identifica el lugar donde se realizará la cita.
- **Relaciones:**
 - Relacionado con las entidades **user**, **employee**, **type_appointment** y **place** para gestionar la información completa de cada cita, incluyendo quién la solicitó, quién la atenderá, su tipo y ubicación.