


1 Autenticación

URL	Método	Código HTTP
/auth/login	POST	200, 401, 400
Caso de uso técnico		
Valida credenciales con Firebase Authentication. Retorna 200 con token JWT si éxito, 401 para credenciales inválidas, 400 si faltan campos requeridos		
Datos de entrada	Datos de salida (200 OK)	
<pre>{ "email": "usuario@dominio.com", "password": "*****" }</pre>	<pre>{ "status": "success", "data": { "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXbzI1NiIsImV4cCI6ODY0MD0", "role": "user", "expires_in": 86400 } }</pre>	

Table 1: Historia de usuario #1: Autenticación

Mockup 1: Pantalla de Login



LOGIN

Correo

Contraseña

ENTRAR

El boton se habilita cuando el campo correo tiene un contenido válido y el campo contraseña no está vacío

- Interacción esperada:
 - Usuario escribe email y contraseña en inputs con etiquetas claras

- Botón "Ingresar" se habilita sólo cuando ambos campos son válidos
- Enlace "¿Olvidaste tu contraseña?" bajo el formulario

- **Flujo visual y eventos:**

1. Al enviar formulario:

- Botón muestra spinner de carga y texto "Verificando..."
- Campos se deshabilitan durante la petición

2. Si éxito:

- Animación de check verde sobre el botón
- Redirección suave a dashboard con fade-out/fade-in

3. Si error:

- Sacudida horizontal en inputs inválidos
- Mensaje rojo bajo el botón: "Credenciales incorrectas"

2 Búsqueda con filtros

URL	Método	Código HTTP
/spaces/search	GET	200, 204, 500
Caso de uso técnico		
Consulta espacios en PostgreSQL usando filtros. Retorna 200 con resultados paginados, 204 si no hay coincidencias, 500 en errores de BD		
Datos de entrada (Query params)	Datos de salida (200 OK)	
?type=sala&capacity=4	<pre>{ "status": "success", "data": [{ "id": 101, "name": "Sala 3A", "capacity": 6, "slots": ["09:00-11:00"] }] }</pre>	

Table 2: Historia de usuario #2: Búsqueda con filtros

Mockup 2: Panel de Búsqueda

The mockup shows a search interface with a left sidebar for filters and a main area for results. The filters include Date (DD/MM/AAA), Time (HH:MM), and Capacity (a slider). The results area lists three sites (Sitio #1, #2, #3), each with a description, capacity, and price. Each site has a green checkmark button and a red X button. Annotations explain the functionality of these buttons.

Filtros

Fecha
DD/MM/AAA

Hora
HH:MM

Capacidad
[Slider]

Sitio #1
Descripción
Capacidad
Precio

Sitio #2
Descripción
Capacidad
Precio

Sitio #3
Descripción
Capacidad
Precio

Al darle en el botón se envía al formulario de reserva

Al darle en el botón se descarta la opción y da visibilidad a otro que cumpla con los filtros

Se muestra la información básica del lugar que cumple con los filtros

- **Interacción esperada:**

- Panel flotante izquierdo con controles deslizantes y dropdowns
- Checkboxes para equipamiento (proyector, pizarra, etc.)
- Botón "Aplicar filtros" con contador de resultados

- **Flujo visual y eventos:**

1. Al cambiar filtros:
 - Contador se actualiza en tiempo real con debounce de 500ms
 - Esqueleto de carga (skeleton screen) en grid durante búsqueda
2. Hover sobre tarjeta:
 - Elevación suave con sombra

3 Reserva de espacios

URL	Método	Código HTTP
/reservations	POST	201, 409, 400
Caso de uso técnico		
Crea reserva con transacción ACID en PostgreSQL. Retorna 201 si éxito, 409 por conflictos de horario, 400 para datos inválidos		
Datos de entrada	Datos de salida (201 Created)	
<pre>{ "space_id": 101, "start_time": "2024-03-15T09:00:00Z", "end_time": "2024-03-15T11:00:00Z" }</pre>	<pre>{ "status": "success", "data": { "reservation_id": "RES-5X8P9", "pdf_url": "https://firebase/RES-5X8P9.pdf" } }</pre>	

Table 3: Historia de usuario #3: Reserva de espacios

Mockup 3: Formulario de Reserva

Formulario de confirmación hora y fecha

Nombre del sitio

Descripción

Detalles

LUN	MAR	MIE	JUE	VIE	SAB	DO
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

7AM

11PM

ATRÁS

RESERVAR

- Interacción esperada:
 - Calendario interactivo con días disponibles en verde
 - Drag selector de rango horario en línea de tiempo
 - Previsualización dinámica del costo total
- Flujo visual y eventos:
 1. Al seleccionar fecha:
 - Animación de "onda" en día seleccionado

- Línea de tiempo se actualiza con slots disponibles
2. Durante reserva:
- Overlay semi-transparente con mensaje: "Reservando tu espacio..."
 - Progress bar indeterminada en cabecera del modal

4 Visualización tiempo real

URL /spaces/status	Método GET	Código HTTP 200, 503
Caso de uso técnico Proporciona estado actual de espacios y abre conexión WebSocket. Retorna 200 con datos iniciales, 503 si servicio no está disponible		
Datos de entrada	Datos de salida (200 OK)	
N/A	<pre>{ "status": "success", "data": { "live_updates_url": "ws://api/spaces/ws", "initial_status": [{ "space_id": 101, "status": "available" }] } }</pre>	

Table 4: Historia de usuario #4: Visualización tiempo real

Mockup 4: Mapa de Ocupación

* Insertar imagen de: Mapa interactivo tipo planta arquitectónica con iconos coloreados

- **Interacción esperada:**
 - Mapa SVG con hotspots clickeables
 - Leyenda dinámica en esquina inferior derecha
 - Selector de piso/área en cabecera
- **Flujo visual y eventos:**
 1. Al cargar mapa:
 - Iconos parpadean suavemente hasta recibir estado inicial
 - Tooltip muestra "Cargando disponibilidad..." al hover
 2. Actualización en tiempo real:
 - Transición de color suave (verde → rojo) en iconos
 - Notificación toast no intrusiva: "Sala 3A ahora ocupada"
 3. Click en espacio ocupado:
 - Modal con detalles: "Ocupado hasta 15:00 por Juan Pérez"
 - Botón "Notificarme cuando se libere" (si usuario autenticado)