

RUTAS

- Generales
 - /registro → Crear una cuenta
 - /login → Iniciar sesión
- /personal
 - /perfil → Ver datos del usuario, editar foto de perfil y cambiar contraseña
 - /correo → Ver bandeja de entrada y enviar/responder
- /usuario
 - /inicio → Navegar por los tipos de instalaciones
 - /instalaciones → Para un tipo de instalación, ver sus instalaciones disponibles y reservar
 - /reservas → Ver reservas actuales y pasadas, y cancelar
- /admin
 - /inicio → Navegar por las funcionalidades de admin
 - /configuracion → Personalizar apariencia de la app
 - /usuarios → Ver y buscar usuarios, validar, banear y hacer admin
 - /instalaciones → Ver instalaciones y crear una nueva
 - /reservas → Ver reservas y filtrar
- Otros
 - error.ejs (no tiene ruta, sólo se renderiza desde servidor)
 - nav.ejs (subplantilla) → Acceder a inicio, correo y perfil
 - footer.ejs (subplantilla) → Info de la universidad

CONVENIOS CÓDIGO

- En HTML, los IDs comienzan siempre por el tipo de elemento (div, button, img...) → <elem>-...
- Idioma
 - Tablas y atributos BBDD: [ES]
 - Variables y funciones JS: [EN]
 - IDs y clases HTML: [EN]
 - Texto de la página web: [ES]
 - Rutas GET y POST: [ES]
 - Comentarios: [ES]
- Notación
 - Ficheros de cliente: snake_case
 - Ficheros de servidor: UpperCamelCase (excepto app.js)
 - Variables y funciones JS: lowerCamelCase
 - IDs y clases HTML: kebab-case
- Al definir el estilo de un selector en CSS el orden de los atributos es:
 1. Display - **display**, **flex-direction**, **align-items**...
 2. Margin & Padding
 3. Size - **width**, **height**, **font-size**...
 4. Border - **border**, **border-radius**...
 5. Color - **background-color**, **color**...
 6. Otros
- En HTML, al indicar las clases que tiene un elemento, primero se ponen las clases propias creadas por nosotros, luego se sigue el orden del punto anterior
- Orden de las etiquetas HTML:
 1. **id**
 2. Otros
 - a. <input>: **name** - **type** - **hidden** - **placeholder** - **value** - **min** - **max**
 - b. : **src** - **alt** - **width** - **height**
 - c. <a>: **href**

3. class
4. Variables data

INTERFACES Y CASOS DE USO

1- /registro

Casos de uso:

Funcionalidad	<u>Registro</u>
Entrada	<ul style="list-style-type: none"> ▪ Nombre ▪ Primer apellido ▪ Segundo apellido ▪ Correo ▪ Contraseña ▪ Facultad (combo)
Salida	<ul style="list-style-type: none"> ▪ Success: modal respuesta ▪ Error: modal error, /error
Requisitos	<ul style="list-style-type: none"> ▪ Campos no vacíos ▪ Regex correo correcto ▪ Correo válido (uno de los disponibles) ▪ Contraseña ≥ 8 caracteres y con al menos 1 número y 1 letra ▪ Facultad existe y es de su uni (está en el combo) ▪ Correo UNIQUE dentro de su uni
AJAX	<ul style="list-style-type: none"> ▪ sí

Notas:

- El link de "Inicia sesión" sólo redirecciona a /login
- **[GET AJAX]** Al cargar la página se cargan los correos disponibles
- **[GET AJAX]** Las facultades se obtienen una vez escribes el correo (depende de la uni)
- Al registrarse el mensaje que aparece dice que el usuario tendrá que esperar a ser validado

Draft:

Crea una Cuenta

1

a1

a1

@

c

f

2- /login

Casos de uso:

Funcionalidad	<u>Inicio de sesión</u>
Entrada	<ul style="list-style-type: none">▪ Correo▪ Contraseña
Salida	<ul style="list-style-type: none">▪ Success: /inicio▪ Error: /login + modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Campos no vacíos▪ Correo válido (uno de los disponibles)▪ Correo existe en esa uni▪ Contraseña correcta
AJAX	<ul style="list-style-type: none">▪ NO

Notas:

- El link de "Regístrate" sólo redirecciona a /registro
- **[GET AJAX]** Al cargar el login se cargan los correos disponibles
- Si el usuario aún no está validado se muestra en el modal error

Draft:



3- /personal/perfil

Casos de uso:

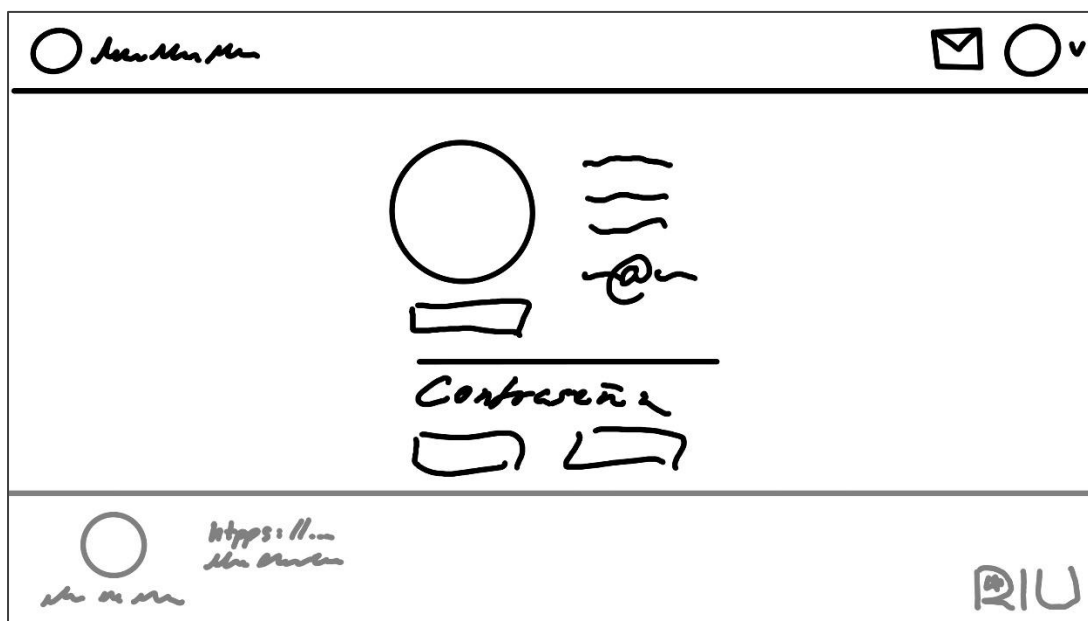
Funcionalidad	<u>Editar foto</u>
Entrada	<ul style="list-style-type: none">Foto
Salida	<ul style="list-style-type: none">Success: modal respuesta + se actualiza la foto (en /personal/perfil y en el nav) + borrado camposError: modal error, /error
Requisitos	<ul style="list-style-type: none">Formato PNGTamaño \leq 64KB
AJAX	<ul style="list-style-type: none">sí

Funcionalidad	<u>Cambiar contraseña</u>
Entrada	<ul style="list-style-type: none">Contraseña actualNueva contraseña
Salida	<ul style="list-style-type: none">Success: modal respuesta + borrado camposError: modal error, /error
Requisitos	<ul style="list-style-type: none">Contraseña actual correctaContraseña actual y nueva no coincidenRegex contraseña nueva correcto
AJAX	<ul style="list-style-type: none">sí

Notas:

- El usuario no puede editar sus datos, sólo la foto (tendría que volver a ser validado)

Draft:



4- /personal/correo

Casos de uso:

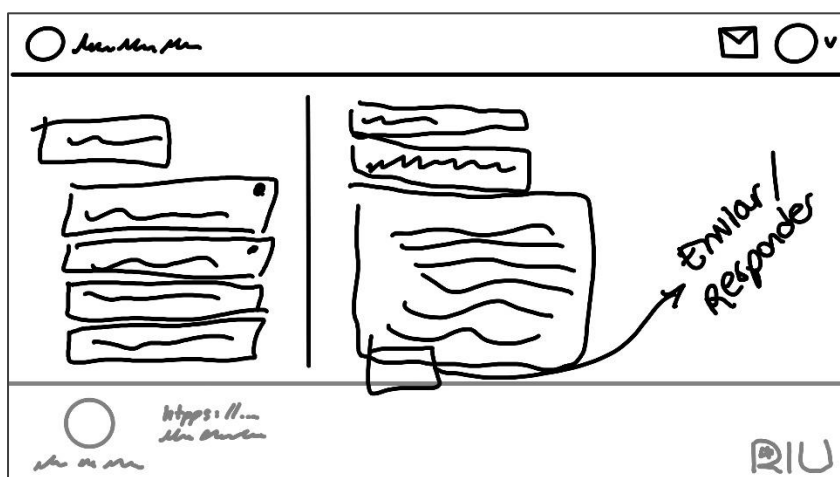
Funcionalidad	<u>Marcar como leído</u>
Entrada	-
Salida	<ul style="list-style-type: none">▪ Success: Letra deja de ser <i>bold</i> + Desaparece punto▪ Error: /error
Requisitos	<ul style="list-style-type: none">▪ No estaba ya leído
AJAX	<ul style="list-style-type: none">▪ sí

Funcionalidad	<u>Enviar correo</u>
Entrada	<ul style="list-style-type: none">▪ Correo usuario destino (sin @)▪ [admin] Facultad (combo)▪ [admin] ¿Universidad?▪ Asunto▪ Mensaje
Salida	<ul style="list-style-type: none">▪ Success: modal respuesta + borrado campos▪ Error: modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Correo existe en esa uni▪ Mensaje no vacío▪ [admin] Facultad válida▪ [admin] Sólo uno de los 3 campos (correo, facultad o universidad)
AJAX	<ul style="list-style-type: none">▪ sí

Notas:

- Los administradores pueden enviar correos a un usuario particular, a una facultad de las disponibles o a la universidad entera. Cuando escriben/marcan una de esas 3 opciones, las otras se desactivan
- Al pulsar en un correo se muestra su contenido (con los campos desactivados) y se marca como leído (si no lo estaba ya)
- Al pulsar en Redactar aparece el formulario (igual que el contenido de un correo)

Draft:

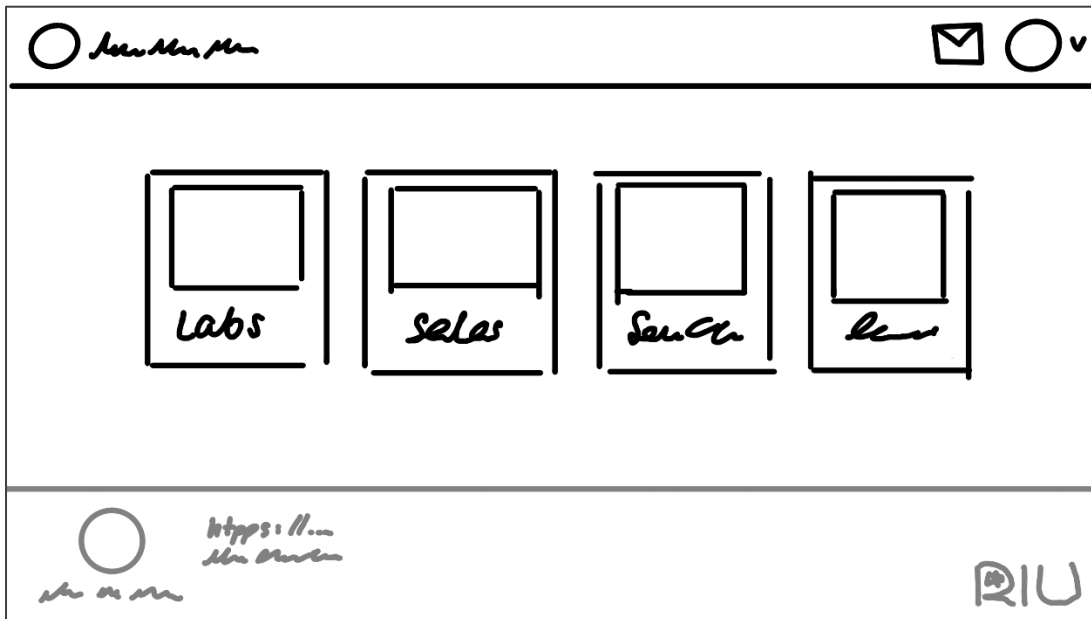


5- /usuario/inicio

Notas:

- Cada tarjeta redirige a /usuario/instalaciones/:id

Draft:



6- /usuario/instalaciones

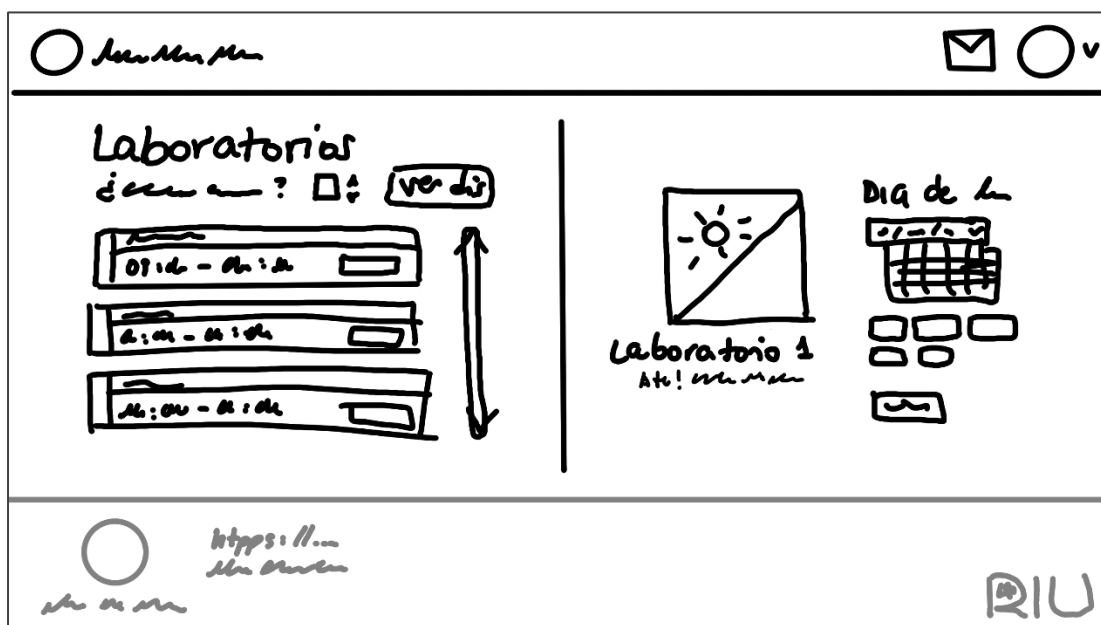
Casos de uso:

Funcionalidad	<u>Reservar</u>
Entrada	<ul style="list-style-type: none"> Nº personas Fecha Hora ID_instalación (<i>hidden</i>)
Salida	<ul style="list-style-type: none"> Success: /usuario/reservas + modal Error: /usuario/instalaciones + modal error, /error
Requisitos	<ul style="list-style-type: none"> Campos no vacíos Instalación válida (existe y es de su universidad) Nº personas es un número > 0 Nº personas ≤ Aforo Fecha+Hora posterior a hoy Hora ini ≤ Hora ≤ Hora fin Fecha != (Sábado o domingo)
AJAX	<ul style="list-style-type: none"> NO

Notas:

- Al seleccionar una instalación se muestra el formulario de la reserva
- **[CLIENTE]** Al indicar Nº personas se filtra por aforo (se dejan sólo las instalaciones con aforo \geq nº personas)
- Al pulsar botón "Terminar reserva" se abre modal de confirmación con el input otra vez de nº personas. Si había filtrado se autorellena el campo
- Horas en badges: gris = disponible, rojo = ocupada (100%, vas a la cola), azul = seleccionada
- Los días ocupados en el calendario aparecen en gris oscuro los días que no se pueden seleccionar (sábados, domingos y días anteriores a hoy), en rojo los ocupados y en blanco los normales
- En el modal de confirmación se muestra mensaje de aviso si está llena para decirte que entras en cola

Draft:



7- /usuario/reservas

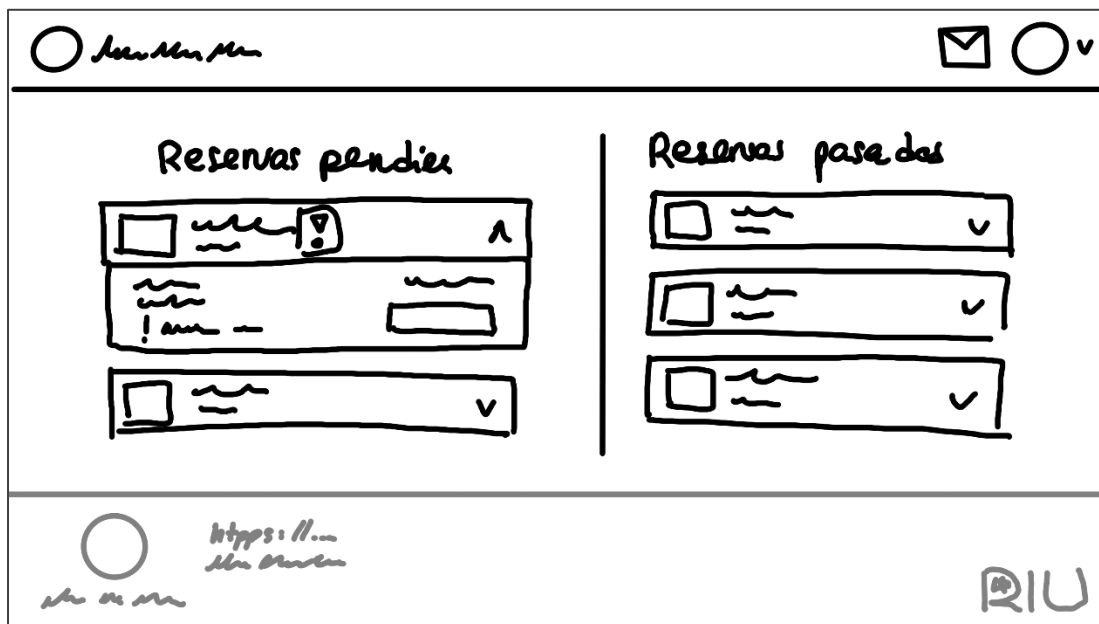
Casos de uso:

Funcionalidad	<u>Cancelar reserva</u>
Entrada	<ul style="list-style-type: none">▪ ID_reserva (<i>hidden</i>)
Salida	<ul style="list-style-type: none">▪ Success: modal respuesta + eliminar reserva de la lista▪ Error: modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Reserva válida (existe y es del usuario)▪ Reserva no es antigua▪ Reserva no estaba ya cancelada
AJAX	<ul style="list-style-type: none">▪ sí

Notas:

- Al pulsar "Cancelar" se abre modal de confirmación ("no se puede deshacer")
- Las reservas antiguas si se quedaron en la cola no se muestran (es como si no hubieran ocurrido)
- Si una reserva actual está en cola se muestra icono de aviso (!)

Draft:

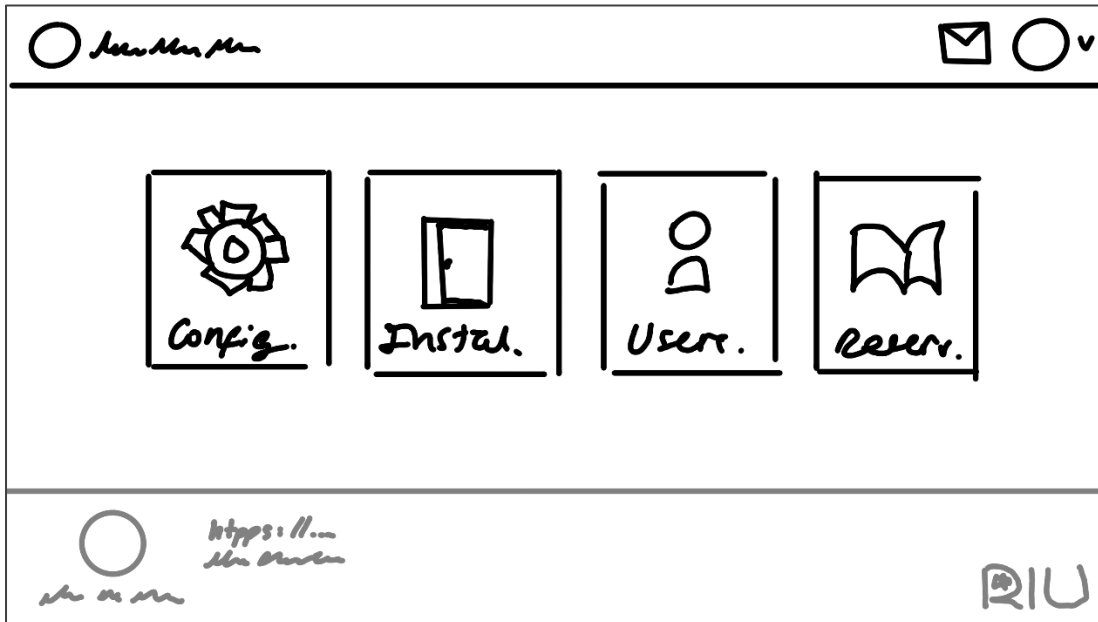


8- /admin/inicio

Notas:

- Cada tarjeta redirige a su funcionalidad por este orden: /configuracion, /instalaciones, /usuarios, /reservas
- Se debe cargar el nombre del admin al cargar la página para saludarle

Draft:



9- /admin/configuracion

Casos de uso:

Funcionalidad	Personalizar
Entrada	<ul style="list-style-type: none">▪ Nombre▪ Dirección▪ Web▪ Logo
Salida	<ul style="list-style-type: none">▪ Success: /inicio + modal respuesta▪ Error: /configuracion + modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Campos no vacíos▪ Regex web correcto▪ Logo es PNG y $\leq 64\text{KB}$▪ El usuario que realiza la acción es admin
AJAX	<ul style="list-style-type: none">▪ NO

Notas:

- El correo no se puede modificar (¿qué pasaría con los usuarios?)

Draft:

The wireframe shows a web interface for configuration. At the top, there is a header bar with a circular logo on the left and a mail icon and a circle with a checkmark on the right. The main content area is titled 'Configuración' and contains four input fields stacked vertically. To the right of these fields is a square box with a diagonal line and a sun-like icon. Below the input fields is a button labeled 'Guardar'. The footer bar contains a circular logo on the left, a URL 'https://...' in the center, and the letters 'RIU' on the right.

10- /admin/usuarios

Casos de uso:

Funcionalidad	<u>Validar</u>
Entrada	<ul style="list-style-type: none">▪ ID_usuario (<i>hidden</i>)
Salida	<ul style="list-style-type: none">▪ Success: modal respuesta + cambiar icono a tick + cambiar botón validar por banear▪ Error: modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Usuario válido (existe y es de la uni)▪ Usuario no estaba ya validado▪ Usuario no estaba baneado▪ Usuario que hace la acción es admin
AJAX	<ul style="list-style-type: none">▪ sí

Funcionalidad	<u>Banear</u>
Entrada	<ul style="list-style-type: none">▪ ID_usuario (<i>hidden</i>)
Salida	<ul style="list-style-type: none">▪ Success: modal respuesta + quitar usuario de la lista▪ Error: modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Usuario válido (existe y es de la uni)▪ Usuario no estaba ya baneado▪ Usuario que hace la acción es admin
AJAX	<ul style="list-style-type: none">▪ sí

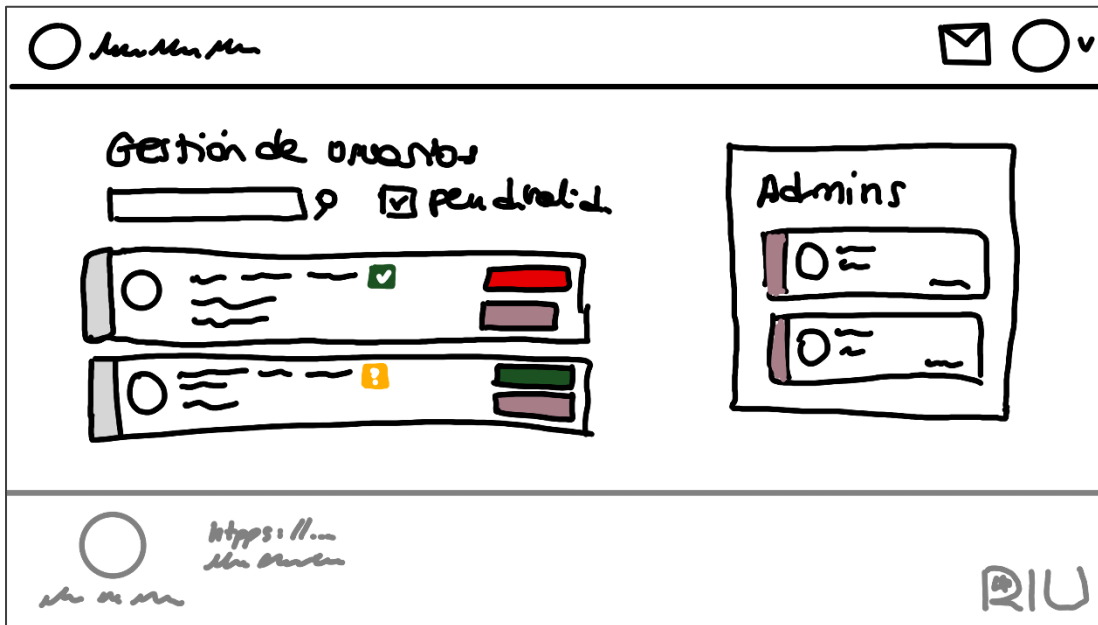
Funcionalidad	<u>Hacer admin</u>
Entrada	<ul style="list-style-type: none">▪ ID_usuario (<i>hidden</i>)
Salida	<ul style="list-style-type: none">▪ Success: modal respuesta + el usuario se mueve a la columna de admins▪ Error: modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Usuario válido (existe y es de la uni)▪ Usuario no era ya admin▪ Usuario no estaba baneado▪ Usuario que hace la acción es admin
AJAX	<ul style="list-style-type: none">▪ sí

Notas:

- Iconos usuarios: tick si validado, ? si pendiente de validar, prohibido si baneado
- Los validados pueden banearse y hacerse admin
- Los pendientes de validar pueden validarse
- Los baneados no puede hacerse nada y salen en gris
- Un admin no puede dejar de serlo (ni nadie puede hacer que deje de serlo)

- **[CLIENTE]** Búsqueda por correo del usuario, se muestran coincidentes (y se desaplica el otro filtro)
- **[CLIENTE]** Filtro no validado, se muestran los pendientes de validar (y se borra el campo de búsqueda)

Draft:



11- /admin/instalaciones

Casos de uso:

Funcionalidad	<u>Crear instalación</u>
Entrada	<ul style="list-style-type: none">▪ Nombre▪ Hora ini▪ Hora fin▪ ¿Completo?▪ Tipo reserva (Individual o Colectiva)▪ Aforo▪ Tipo (uno de los disponibles o nuevo)▪ Foto
Salida	<ul style="list-style-type: none">▪ Success: modal respuesta + se añade a la lista + se oculta el formulario▪ Error: modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Campos no vacíos▪ Nombre no repetido para ese tipo▪ Hora fin > Hora ini▪ Hora ini y hora fin son horas exactas▪ Tipo reserva válido (uno de los disponibles)▪ Aforo es un número > 0▪ Tipo válido (uno de los disponibles en esa uni)▪ Logo es PNG y $\leq 64\text{KB}$▪ El usuario que hace la acción es admin
AJAX	<ul style="list-style-type: none">▪ sí

Funcionalidad	<u>Editar instalación</u>
Entrada	<ul style="list-style-type: none">▪ Nombre▪ Foto
Salida	<ul style="list-style-type: none">▪ Success: modal respuesta + + se oculta el formulario▪ Error: modal error, /error
Requisitos	<ul style="list-style-type: none">▪ Nombre no vacío▪ Nombre no repetido para ese tipo▪ Al menos 1 de los campos se ha modificado▪ Logo es PNG y $\leq 1\text{MB}$
AJAX	<ul style="list-style-type: none">▪ sí

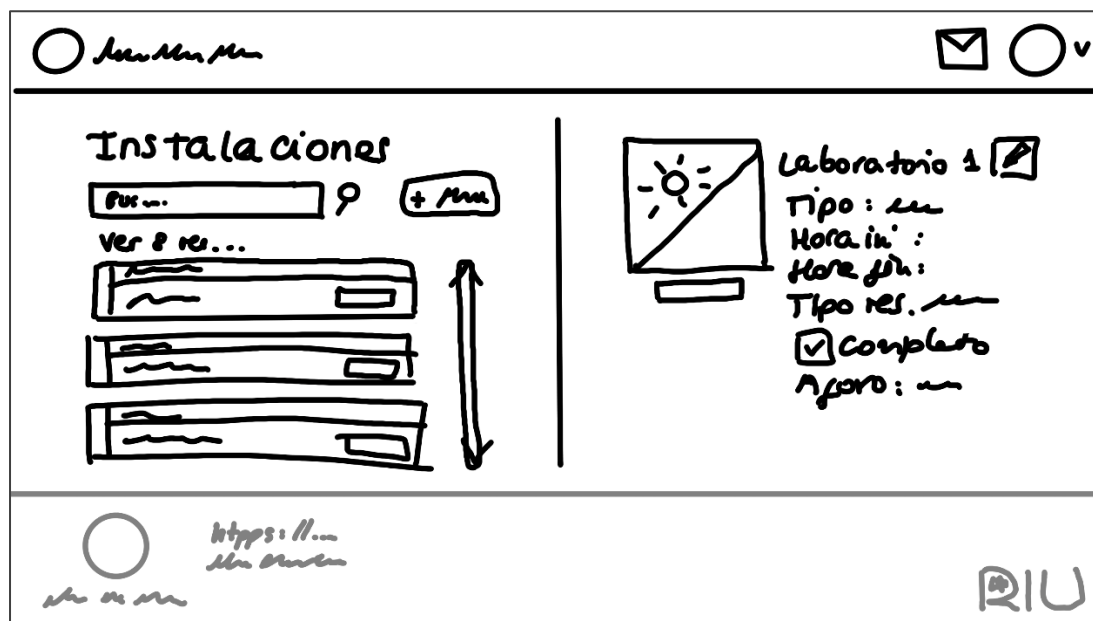
Funcionalidad	<u>Crear tipo de instalación</u>
Entrada	<ul style="list-style-type: none">▪ Nombre▪ Foto
Salida	<ul style="list-style-type: none">▪ Success: aparece opción en el formulario (seleccionada)▪ Error: modal error, /error

Requisitos	<ul style="list-style-type: none"> ▪ Nombre no vacío ▪ Nombre no repetido ▪ Logo es PNG y $\leq 64\text{KB}$
AJAX	<ul style="list-style-type: none"> ▪ sí

Notas:

- Al seleccionar una instalación se muestran sus detalles (el formulario para editar), y al pulsar en Nueva se abre el formulario
- **[CLIENTE]** Búsqueda por nombre, se quedan las coincidentes
- Sólo se puede editar nombre y foto (el resto de campos podrían afectar a reservas ya realizadas)
- Si se selecciona "Nuevo tipo" se abre el modal para crear el tipo

Draft:



12- /admin/reservas

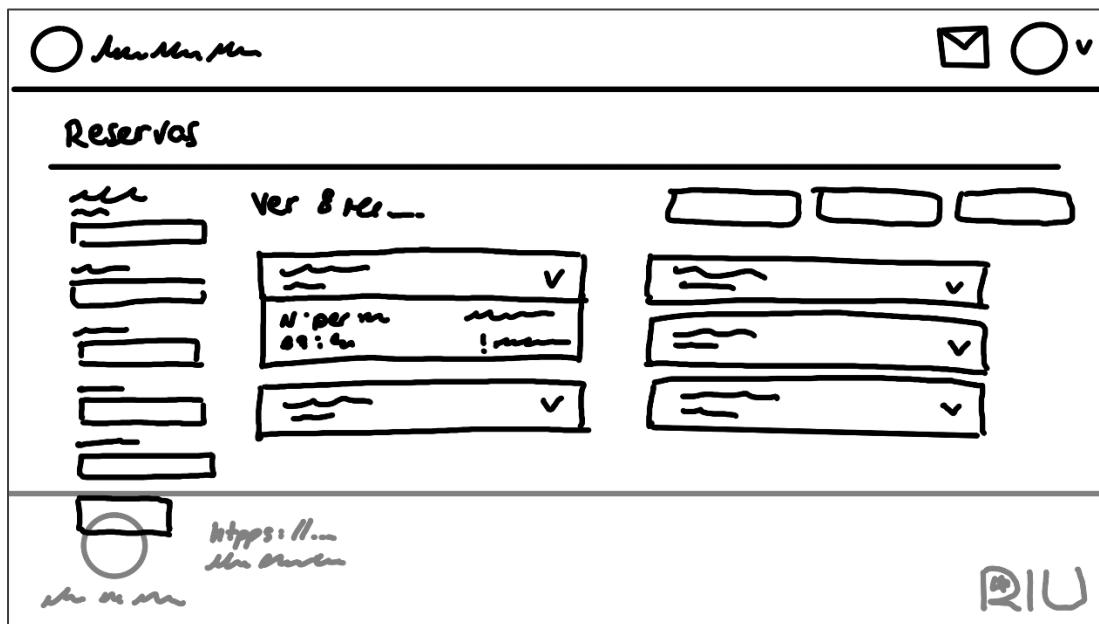
Casos de uso:

Funcionalidad	<u>Filtrar reservas</u>
Entrada	<ul style="list-style-type: none">▪ [Usuario] (nombre, apellido o correo)▪ [Fecha]▪ [Facultad]▪ [Instalación]
Salida	<ul style="list-style-type: none">▪ Success: /admin/reservas▪ Error: /admin/reservas + modal error, /error
Requisitos	-
AJAX	<ul style="list-style-type: none">▪ NO

Notas:

- Al filtrar salen los filtros que se han aplicado

Draft:



13- /error

Notas:

- El botón de atrás lleva al sitio donde estuvieras antes

Draft:

RIU

404

Not found!

all the time

← **Ante**