

Ejercicios ANGULAR

Ejercicio 1 – Maquetación en HTML5 y SCSS

Este primer ejercicio consiste en la maquetación de la pantalla Home de Ibercaja. Para ello, tendrás que crear un proyecto de Angular con SCSS, y desarrollar toda la pantalla en un componente.

Requisitos del ejercicio:

- Maquetación responsive utilizando SCSS.
- En el SCSS, separa mediante comentarios los estilos generales de los estilos específicos de cada componente.
- Uso de reset/normalizer para la homogenización entre navegadores.
- Creación y utilización de variables de color con SCSS.
- Creación de una fuente tipográfica de iconos. Se realizará con Icomoon, tal y como está explicado en la formación básica de Front.

Documentación:

- Enlace de los iconos para la creación de la fuente tipográfica:

<https://americas.internal.deloitteonline.com/sites/DxDHuesca/digitalexp/Documentos%20compartidos/Formaciones%20y%20workshops/Formaci%C3%B3n%20B%C3%A1sica/iconosEjercicioAngular.zip>

- Enlaces de los diseños:

- **Pantalla home desktop (mín: 1280px):**

<https://www.sketch.com/s/b000f19f-82ce-4bd2-bf3f-8b16020f6880/p/B7CD4589-DCA9-43B8-8AA2-871DC17DC977>

- **Pantalla home con el menú colapsado (1024px – 1279px):**

<https://www.sketch.com/s/b000f19f-82ce-4bd2-bf3f-8b16020f6880/p/DE0E3C3D-C70C-4902-A7A8-A6ECBB25ED44>

- **Pantalla home mobile (320px – 1023px):**

<https://www.sketch.com/s/b000f19f-82ce-4bd2-bf3f-8b16020f6880/p/15EAD3B5-966F-40F5-96A5-2E1FDA158A07>

Ejercicio 2 – Creación de componentes de la página principal de Ibercaja

El segundo ejercicio consiste en realizar la página principal de Ibercaja, formada por distintos componentes.

2.1. Elementos del AppComponent

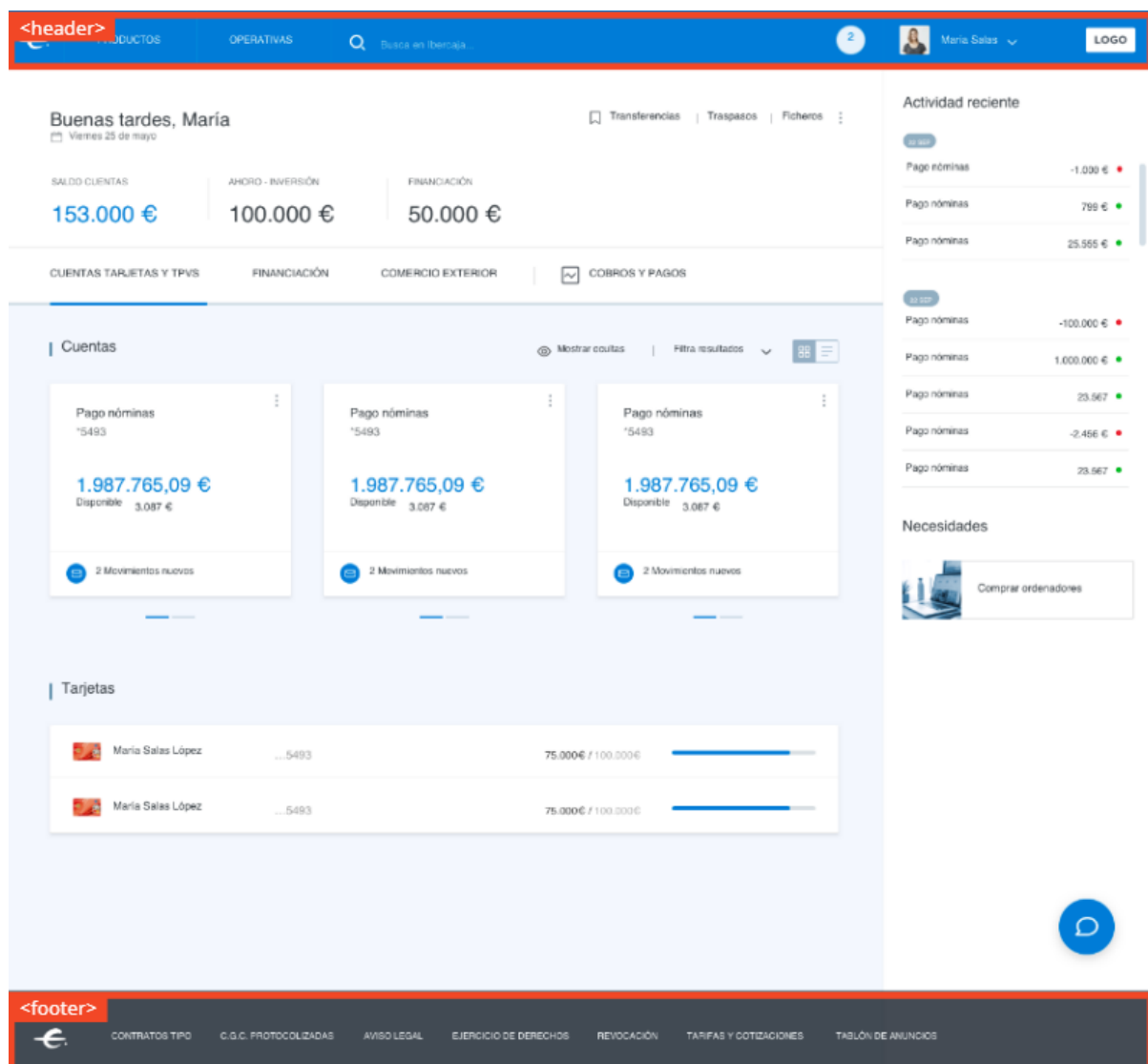
Antes de empezar a crear tus componentes, debes incluir en el AppComponent el header y el footer, ya que son elementos comunes a todo el proyecto:

<header>

Contendrá al logo de Ibercaja, 2 botones, buscador, notificaciones, usuario y avatar.

<footer>

Contendrá el logo y los enlaces.



2.2. Creación de componentes

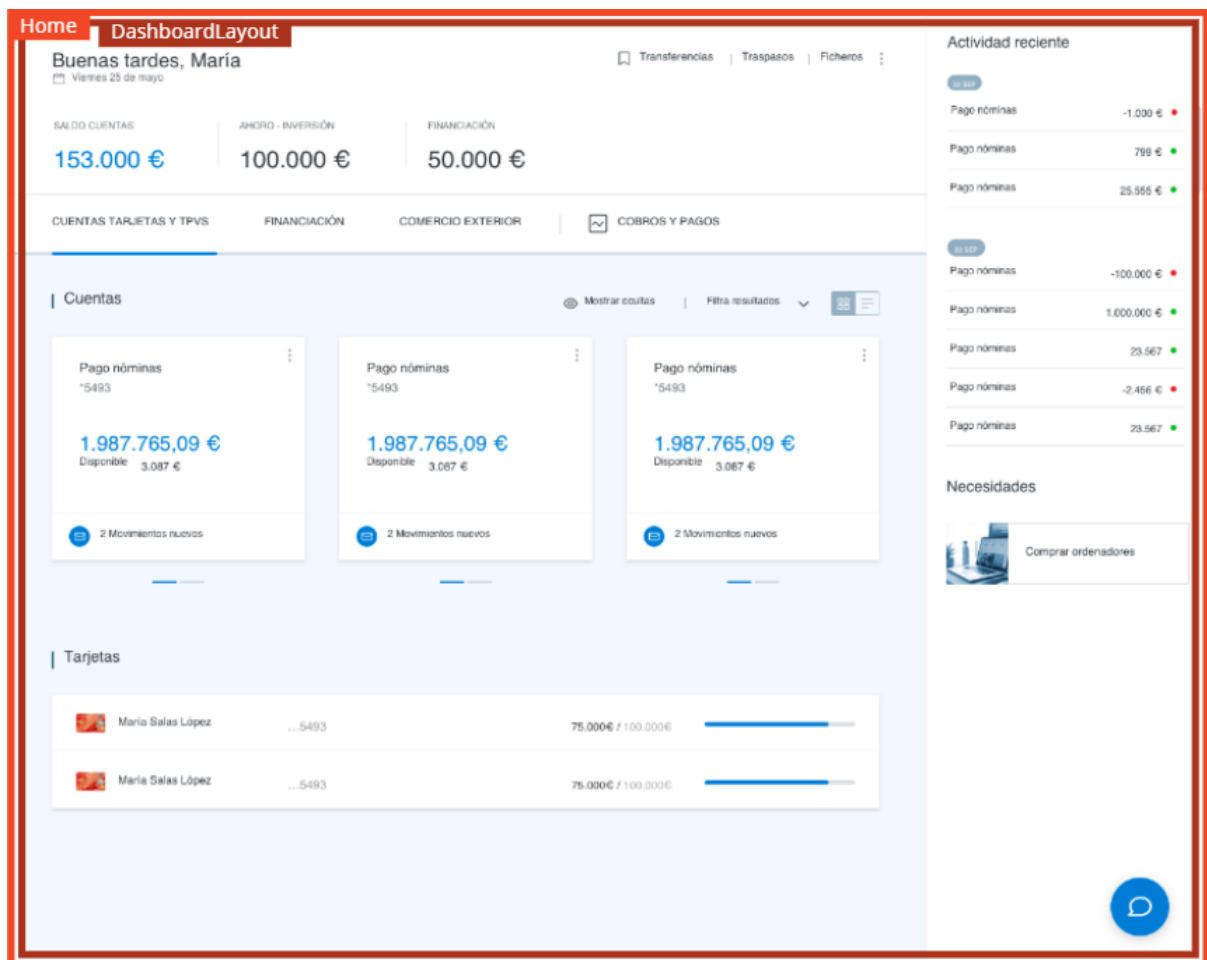
Se crearán los siguientes componentes:

Home

Componente padre de la página (se encarga de pintar los componentes hijos).

DashboardLayout

Componente que servirá como template para alojar los componentes. Se deberá poder reutilizar para todas las pantallas, independientemente si estamos en la sección de la home, en financiación, comercio exterior, etc. Será un componente que delimitará cuatro secciones (top, menu, bottom y sidebar).

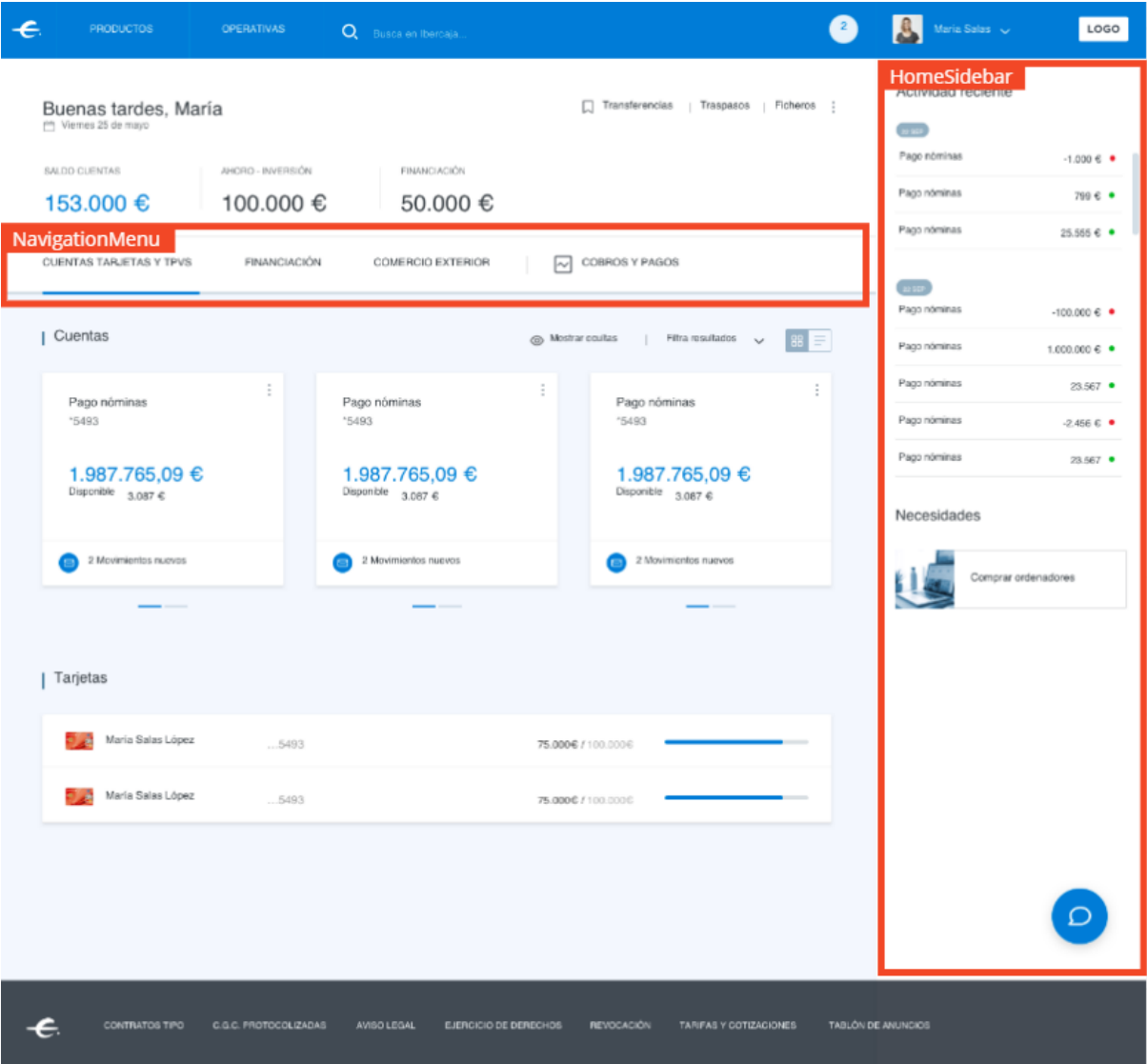


NavigationMenu

Menú de navegación horizontal.

HomeSidebar

Componente que gestiona el comportamiento (open/closed) y aloja la información de la parte del sidebar.



DashboardFavorites

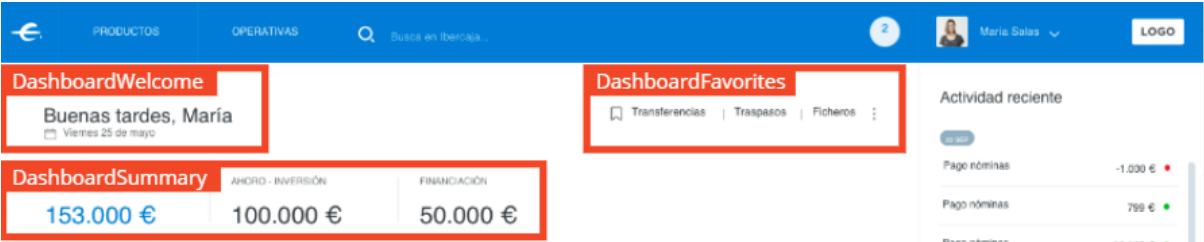
Accesos directos de la parte superior.

DashboardWelcome

Texto de bienvenida a usuario y fecha.

DashboardSummary

Los totales de las cuentas.

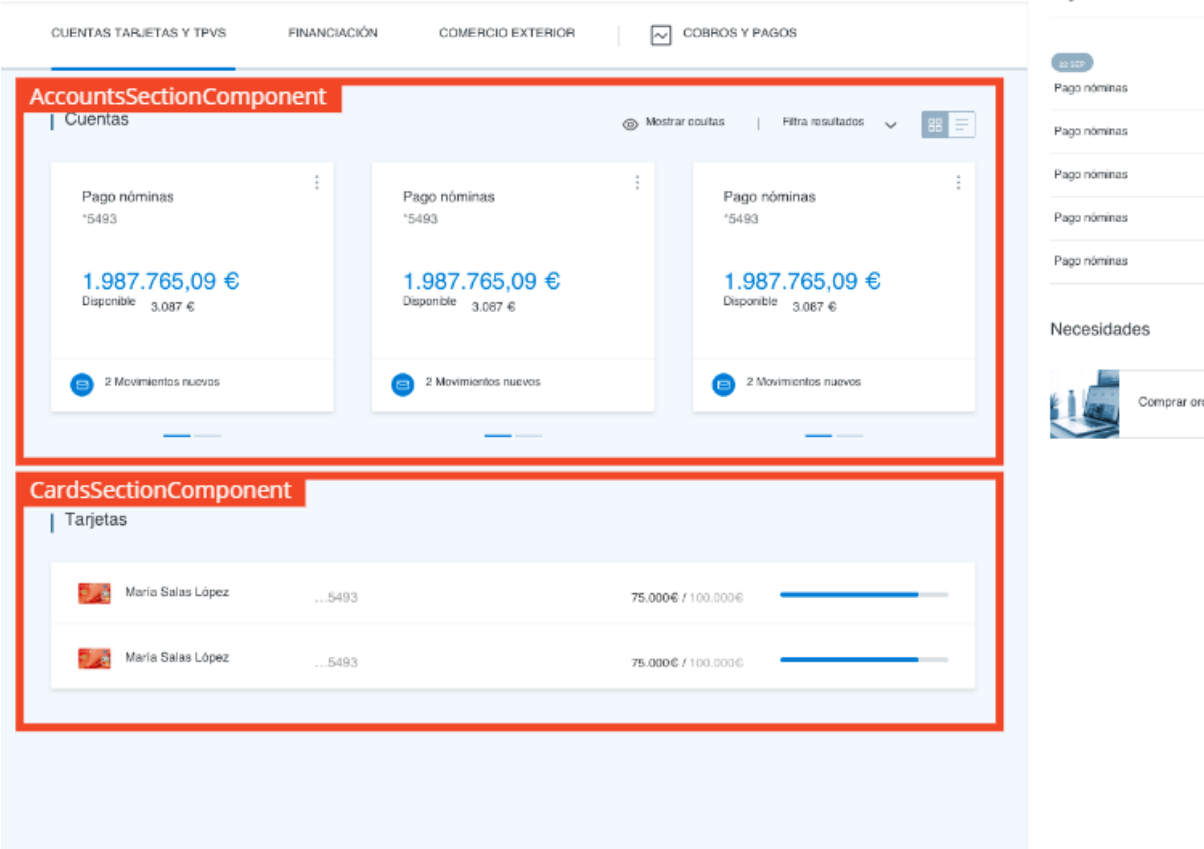


AccountsSectionComponent

Sección con los filtros y las cards de cuentas.

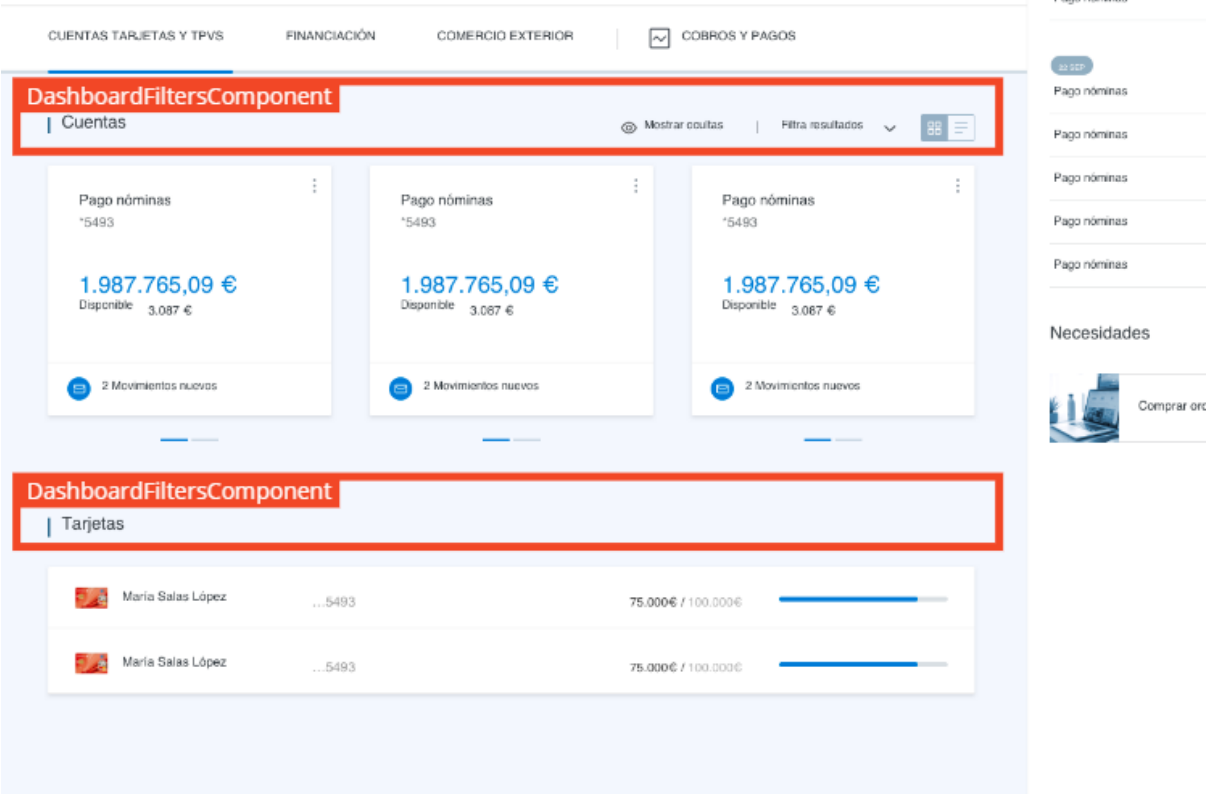
CardsSectionComponent

Sección con los filtros y la tabla de tarjetas.



DashboardFiltersComponent

Componente reutilizable con el título y la barra de acciones de cada sección de productos.



Ejercicio 3 – Creación de API y modelado de datos para la página principal de Ibercaja

Se creará un servicio que simule la petición de la información a la API. Este servicio consumirá la información de un fichero JSON que se creará en la carpeta assets.

3.1. Creación del fichero JSON

Crea un fichero JSON con un mockup de los datos que se ven en los diseños por secciones.

Ejemplo de la sección de favoritos:

```
...
"favorites": [
  {
    "label": "Transferencias",
    "path": ""
  },
  {
    "label": "Traspasos",
    "path": ""
  },
  {
    "label": "Recibos",
    "path": ""
  }
],
...
```

3.2. Creación del servicio

Crea un servicio que lea el JSON usando la librería RxJS.

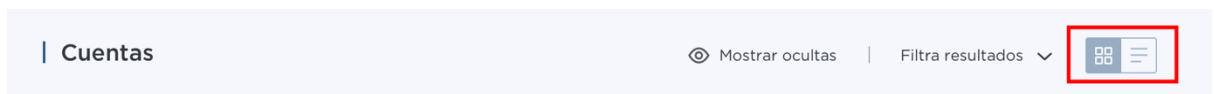
3.3. Creación de los modelos

Crear modelos de los datos devueltos por el servicio.

Ejercicio 4 – Acciones de usuario

4.1. Visualización de los componentes

- En TABLET, cuando el usuario haga clic en el sidebar, éste se cerrará si está abierto o se abrirá si está cerrado (se abre posicionado por encima de la home).
- Cuando el usuario haga clic en los siguientes iconos (tanto en cuentas como en tarjetas), se cambiará entre la vista grid y la vista table.



4.2. Menú Navegación

- Cuando el usuario haga clic en cualquier elemento del NavigationMenu se mostrará un alert (JS) con la siguiente info '*Navegamos a [nombre_menu]*'.
- Cuando el usuario haga clic en una cuenta o tarjeta aparecerá un alter (JS) con la siguiente info '*Navegamos a la cuenta/tarjeta [nombre_producto]*'.

Ejercicio 5 – Añadir lógica de navegación y modularización

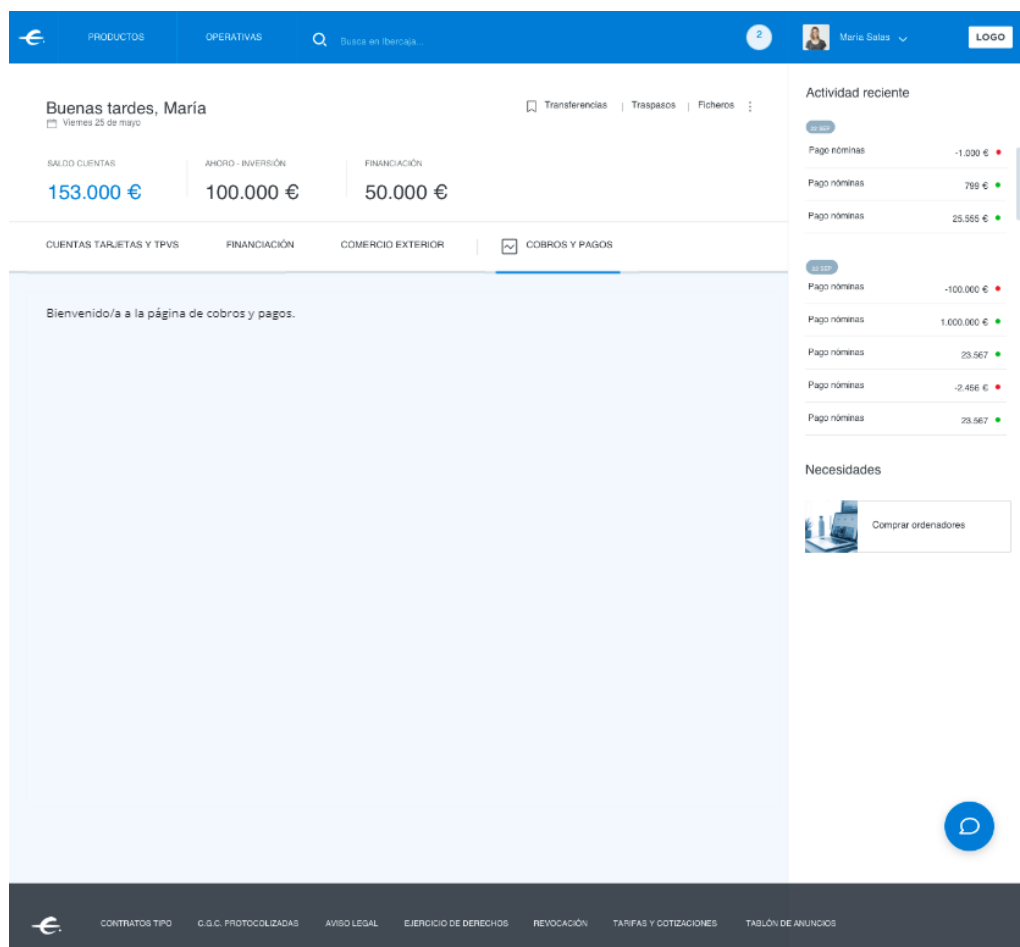
Deberás realizar la navegación a las diferentes páginas de la aplicación. Para ello, tendrás que realizar las siguientes tareas.

5.1. Modularización de los componentes

Divide la aplicación en módulos. Crea los módulos necesarios e incluye los componentes creados en los anteriores ejercicios en su correspondiente módulo. Los componentes que vayas a reutilizar podrás incluirlos en un módulo llamado shared. La página que has creado hasta el momento “Cuentas y Tarjetas”, será el módulo principal que se carga por defecto.

5.2. Navegación

Creación de las nuevas páginas de la aplicación: “Financiación”, “Comercio Exterior” y “Cobros y Pagos”. Estas páginas se deben cargar mediante LazyLoading. Las nuevas páginas, además de los componentes comunes, deberán tener el texto “Bienvenido/a a la página de [nombre_página]”, como en el siguiente ejemplo:



Ejercicio 6 (Opcional) – Creación de la nueva página de cobros y pagos

Maquetación de la página de cobros y pagos. En este ejercicio se incluye la parte del submenú y el listado con las tabs. Deberás componentizar y reutilizar todo aquello que consideréis apropiado.

Pantalla Cobros y Pagos:

<https://www.sketch.com/s/b000f19f-82ce-4bd2-bf3f-8b16020f6880/p/E287D549-A82A-4F0A-B134-FCA2468CC108>