

Proyecto final: Aplicación Web



SPRINT

ELENA CORES SMITH

RAMÓN MERCHÁN CASTILLO

GEMA PATRICIA HERRERA DE MIGUEL

MARÍA CALVO TORRES

Índice

- ▶ ¿Qué es SPRINT?
- ▶ Diagrama de Casos de uso
- ▶ Justificación de requisitos propios
- ▶ SiteMap y Wireframes
- ▶ Arquitectura
- ▶ Justificación de requisitos obligatorios

Índice

- ▶ **¿Qué es SPRINT?**
- ▶ Diagrama de Casos de uso
- ▶ Justificación de requisitos propios
- ▶ SiteMap y Wireframes
- ▶ Arquitectura
- ▶ Justificación de requisitos obligatorios

¿Qué es SPRINT?

Aplicación web que tiene como objetivo que el usuario se inicie en un deporte.

- ▶ Proporcionamos información sobre deportes para que el usuario se interese y se anime a realizar alguna actividad deportiva:
 - ▶ Información sobre deportes más conocidos
 - ▶ Eventos y noticias deportivas de actualidad
 - ▶ Catálogo de productos necesarios para practicar deporte
 - ▶ Biografías de deportistas famosos
- ▶ Espacio de opinión entre usuarios para valorar los deportes

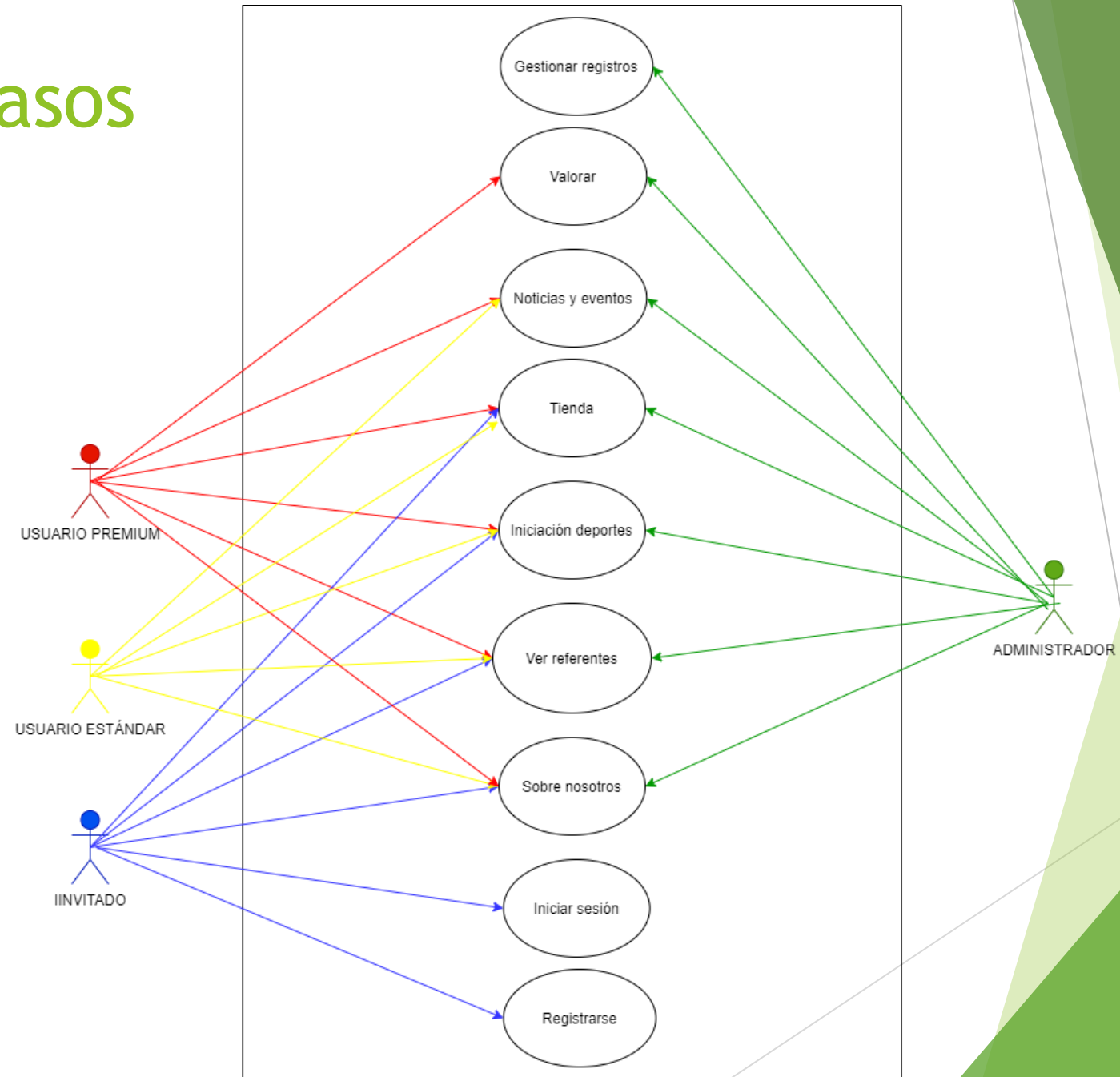
Índice

- ▶ ¿Qué es SPRINT?
- ▶ **Diagrama de Casos de uso**
- ▶ Justificación de requisitos propios
- ▶ SiteMap y Wireframes
- ▶ Arquitectura
- ▶ Justificación de requisitos obligatorios

Diagrama de casos de uso

► 4 roles:

- Invitado 
- Estándar 
- Premium 
- Administrador 



Índice

- ▶ ¿Qué es SPRINT?
- ▶ Diagrama de Casos de uso
- ▶ **Justificación de requisitos propios**
- ▶ SiteMap y Wireframes
- ▶ Arquitectura
- ▶ Justificación de requisitos obligatorios

Justificación de requisitos propios (I)

Requisitos totales: 1 9

Requisitos cumplidos: 1 5

Requisitos cumplidos parcialmente: 2

Requisitos no cumplidos: 3

Justificación de requisitos propios (II)

► Requisitos cumplidos parcialmente:

- **RF11:** La aplicación web tiene una sección “Iniciación a un deporte” que, al hacer clic sobre uno de los deportes del listado de deportes de iniciación, muestra toda la información relativa a ese deporte: breve descripción, vídeo(s) explicativo(s), material necesario, niveles de dificultad, mapa resaltando las instalaciones donde practicarlo y valoraciones de ese deporte de otros usuarios.
- **RF12:** El rol usuario *premium* podrá insertar sus propias valoraciones de un deporte, en la sección de “Iniciación”.

Justificación de requisitos propios (III)

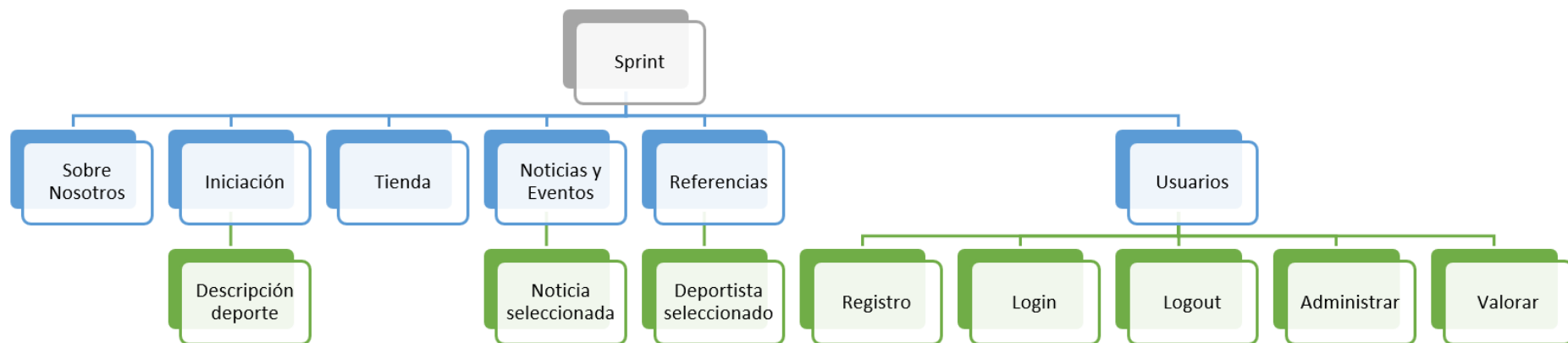
► Requisitos no cumplidos:

- **RF10:** La aplicación web tiene una sección “Iniciación a un deporte” que permite obtener resultados filtrados por el nombre de un deporte.
- **RF14:** El rol *usuario premium* podrá marcar productos en la sección “Tienda” como favoritos.
- **RF15:** La aplicación web tiene una sección “Tienda” que permite obtener resultados filtrados por el nombre de un producto.

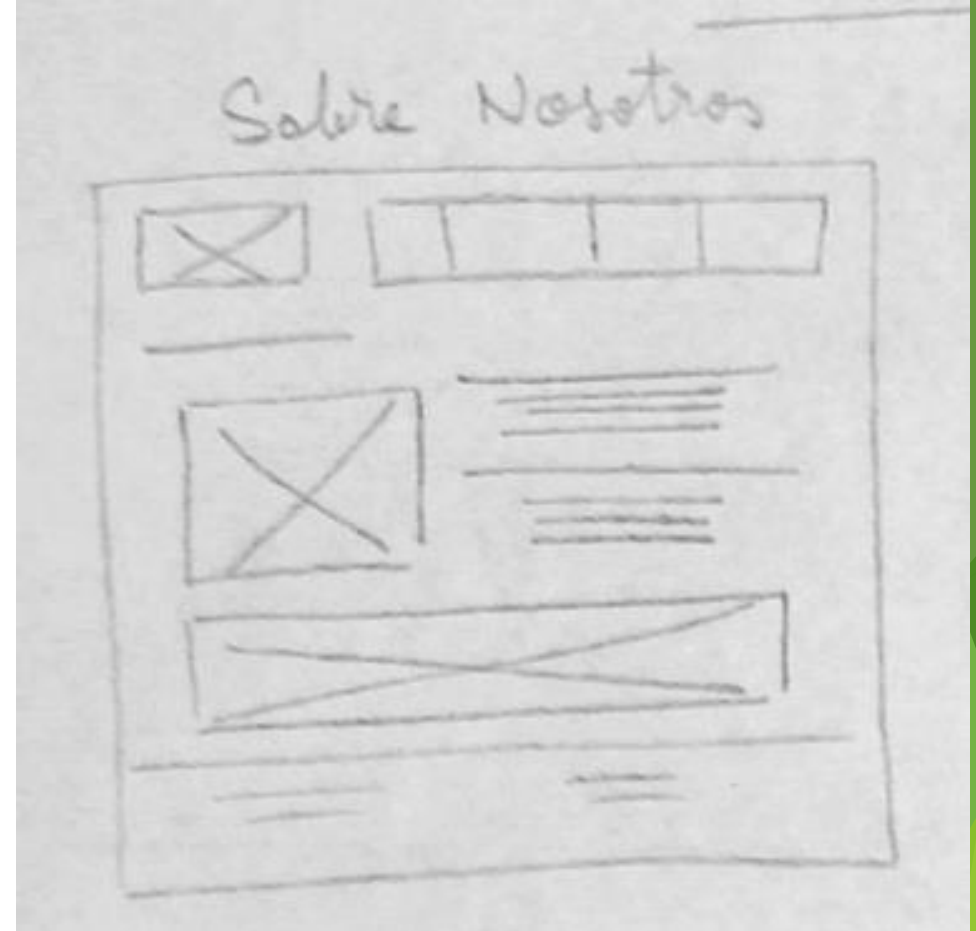
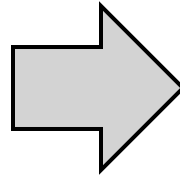
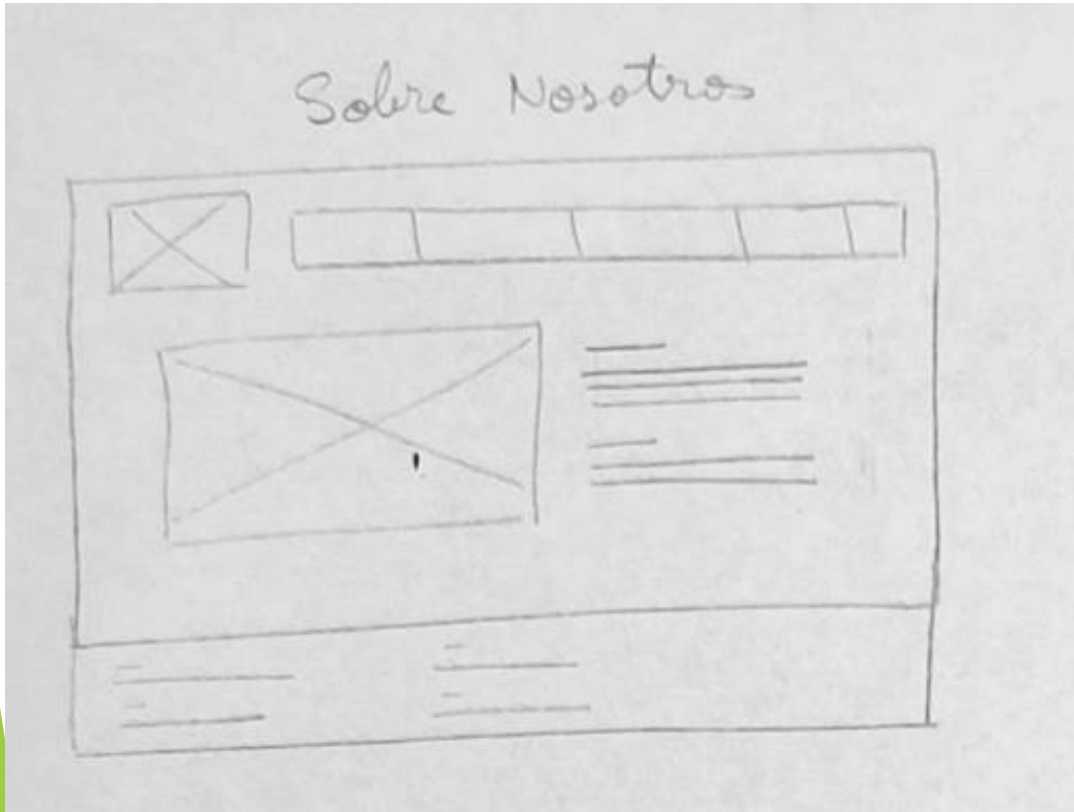
Índice

- ▶ ¿Qué es SPRINT?
- ▶ Diagrama de Casos de uso
- ▶ Justificación de requisitos propios
- ▶ **SiteMap y Wireframes**
- ▶ Arquitectura
- ▶ Justificación de requisitos obligatorios

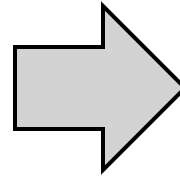
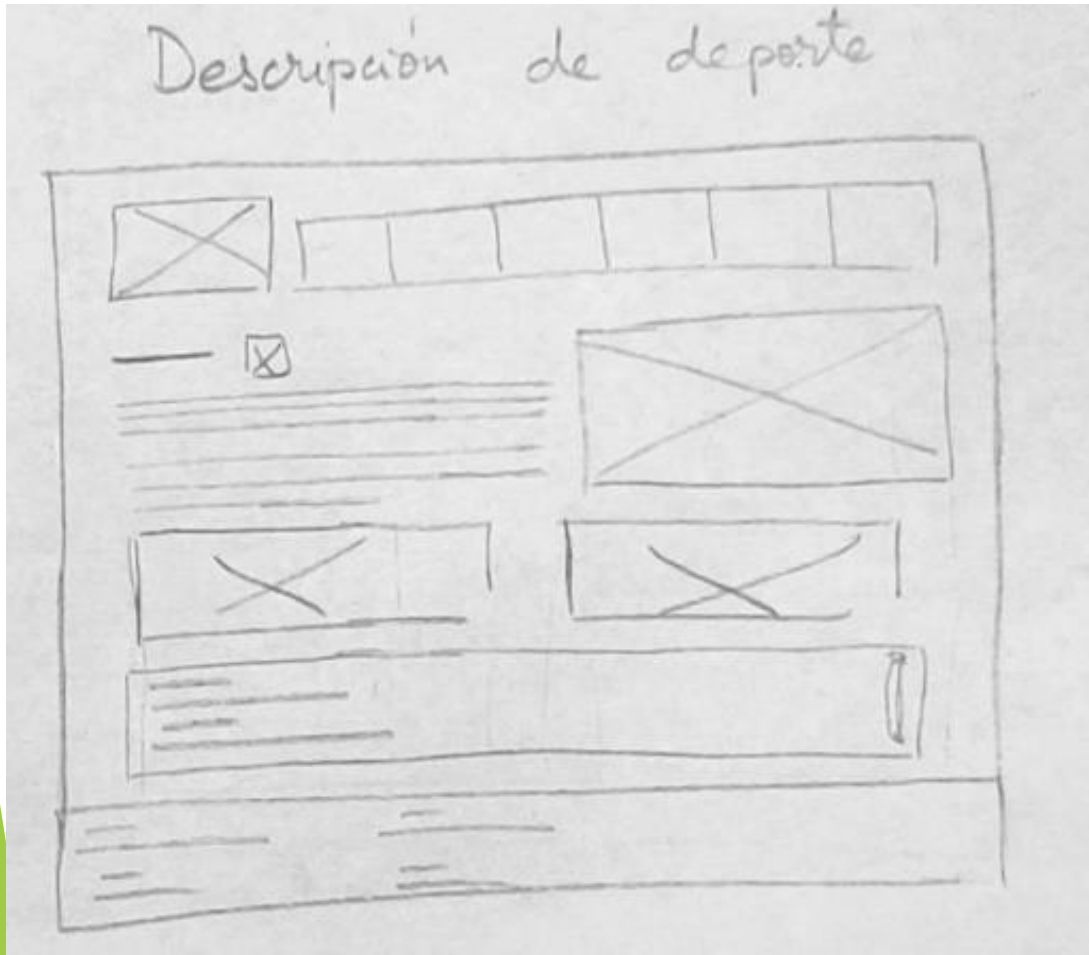
SiteMap



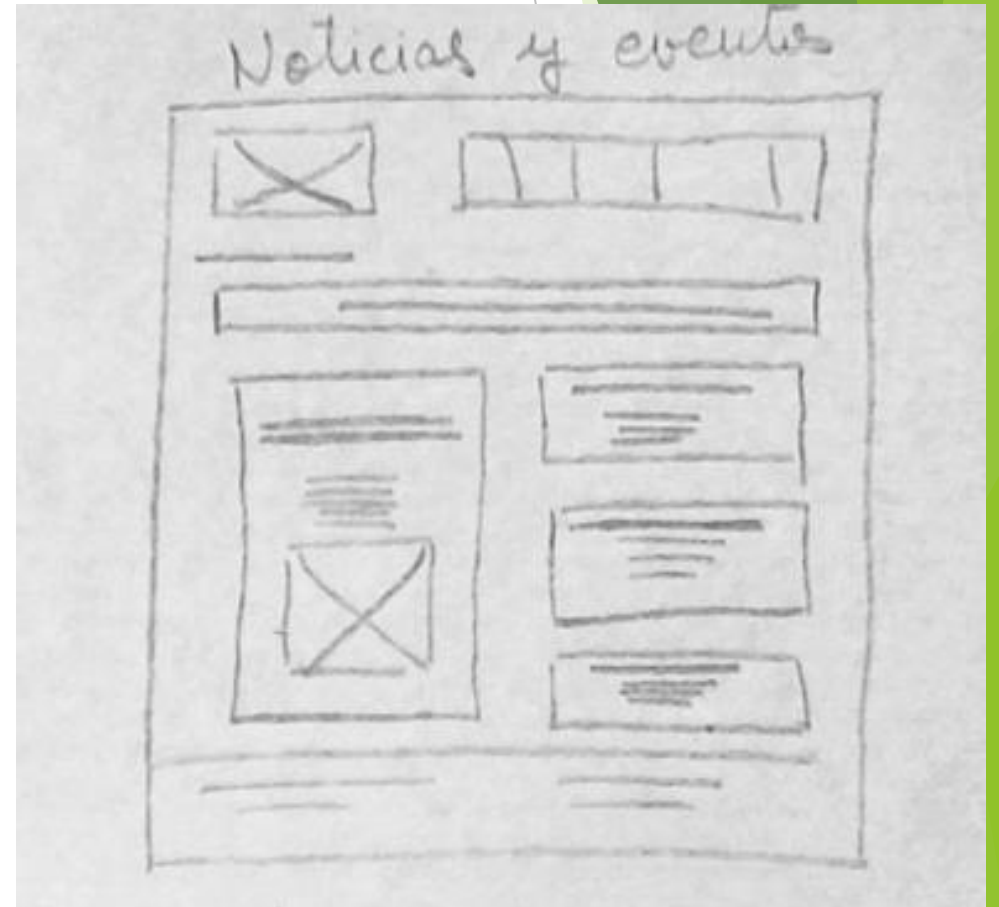
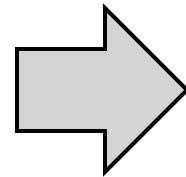
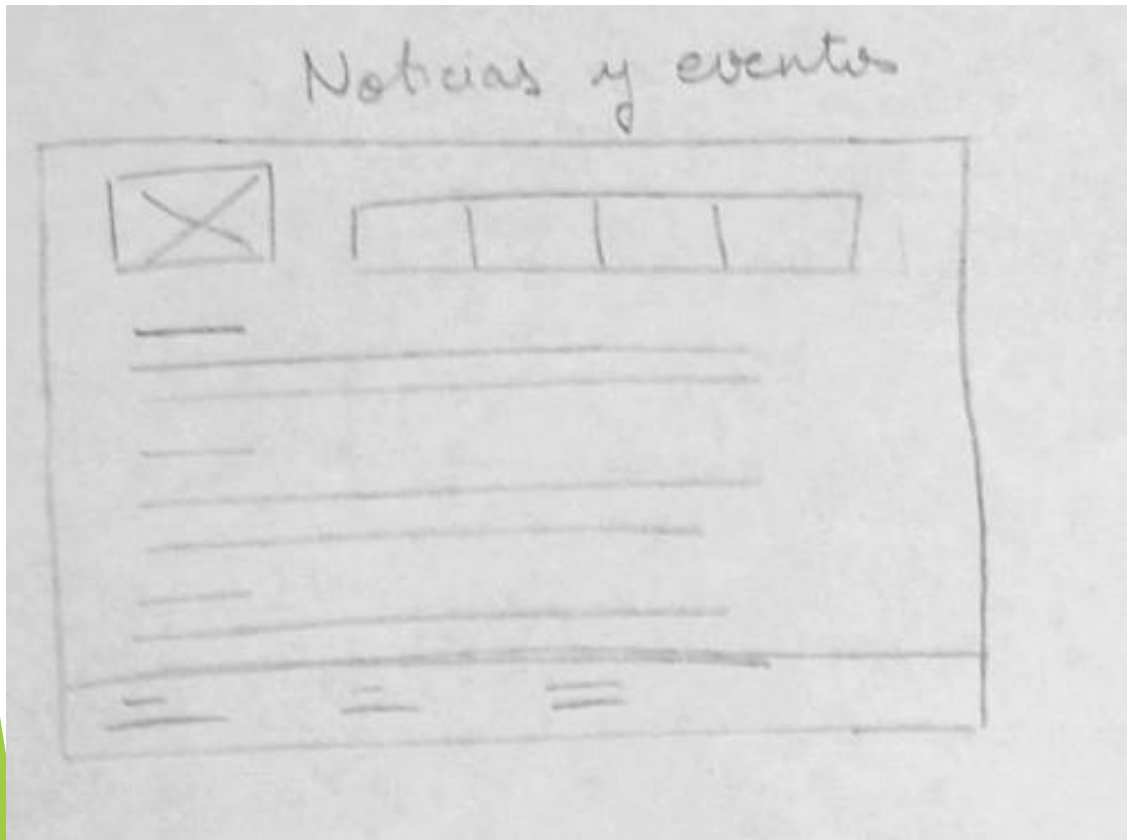
Wireframe (I)



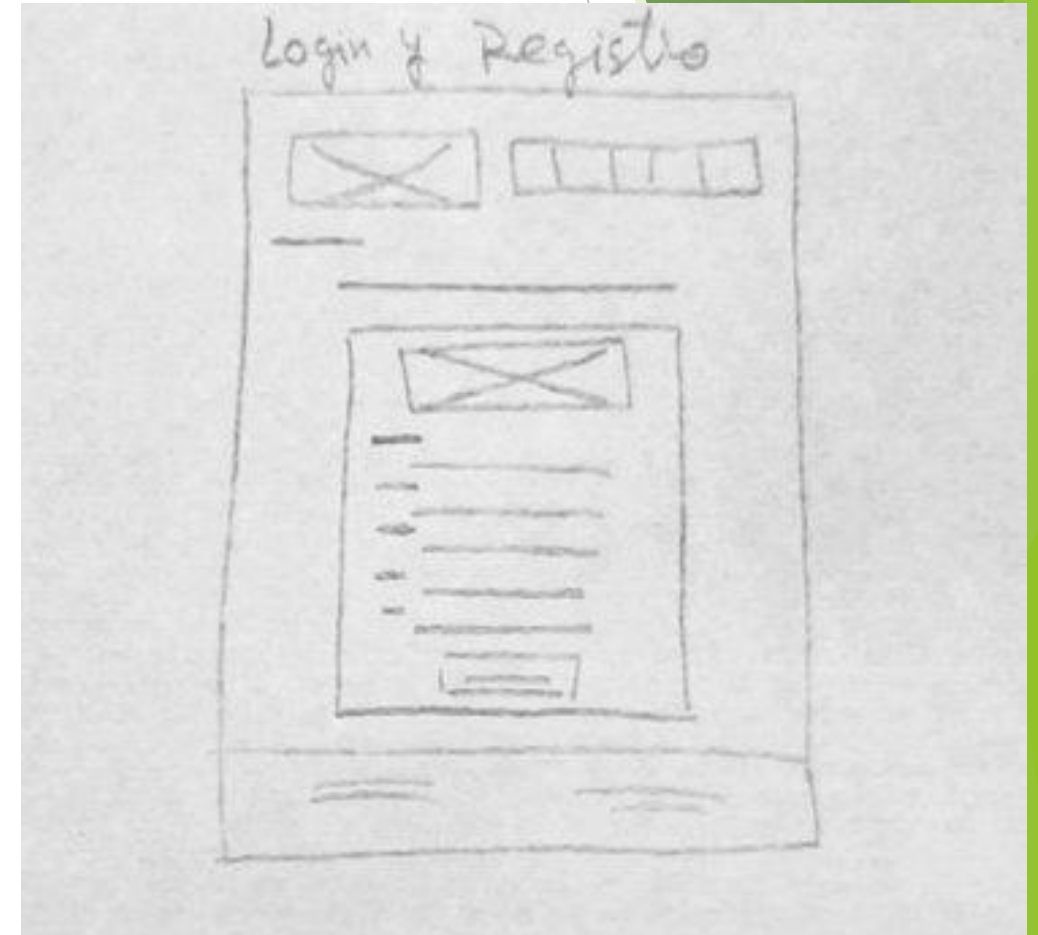
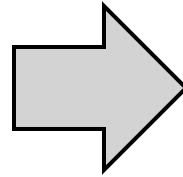
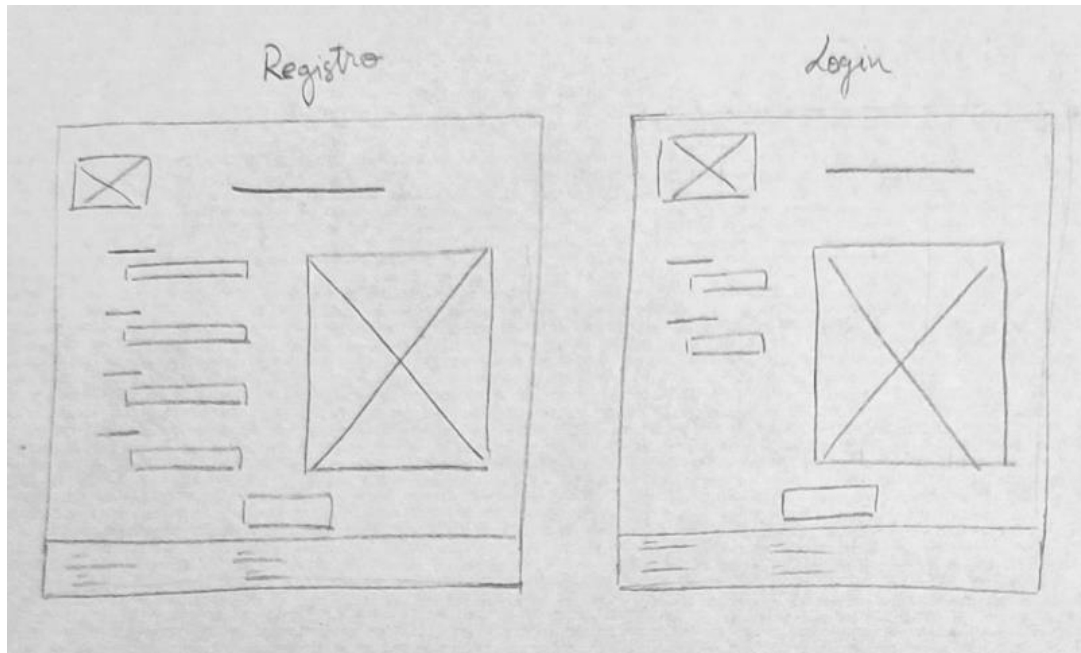
Wireframe (II)



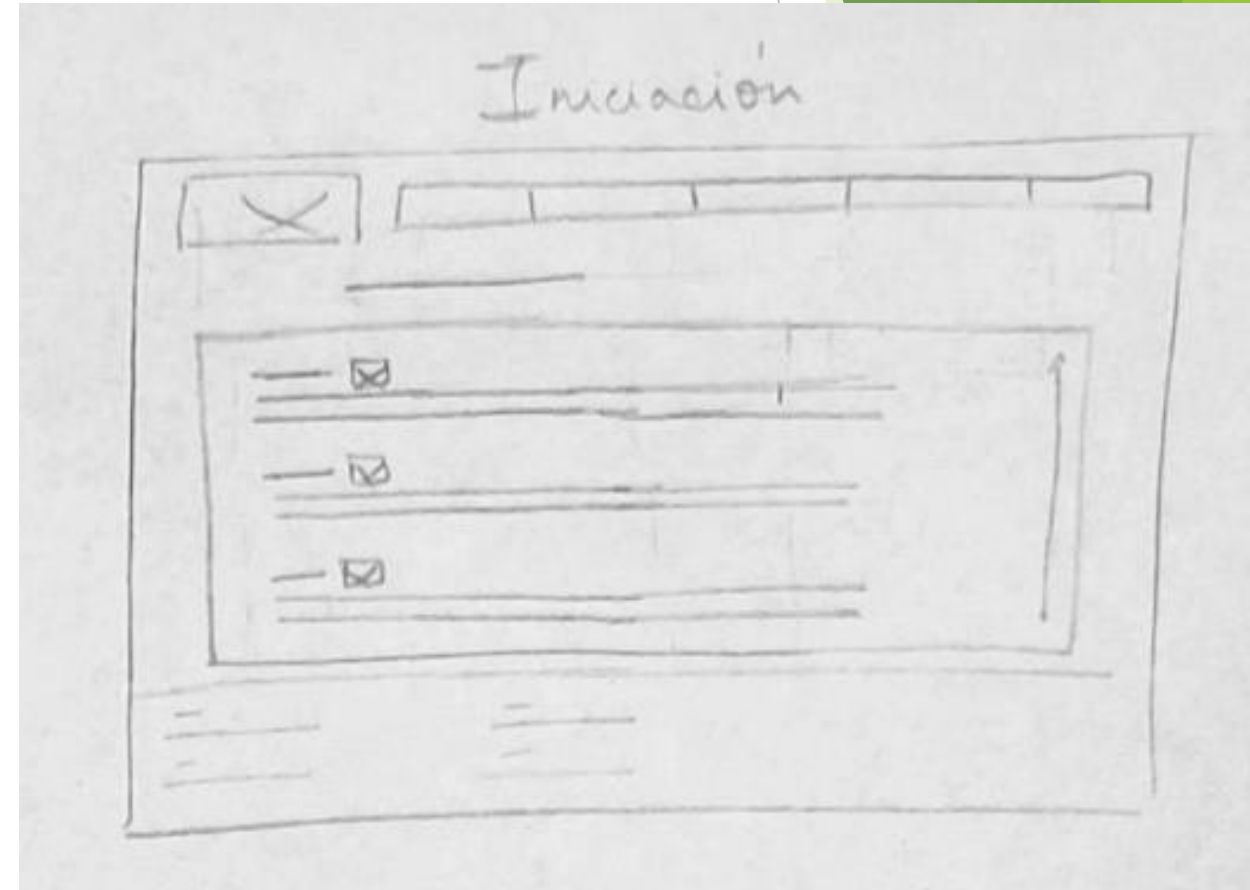
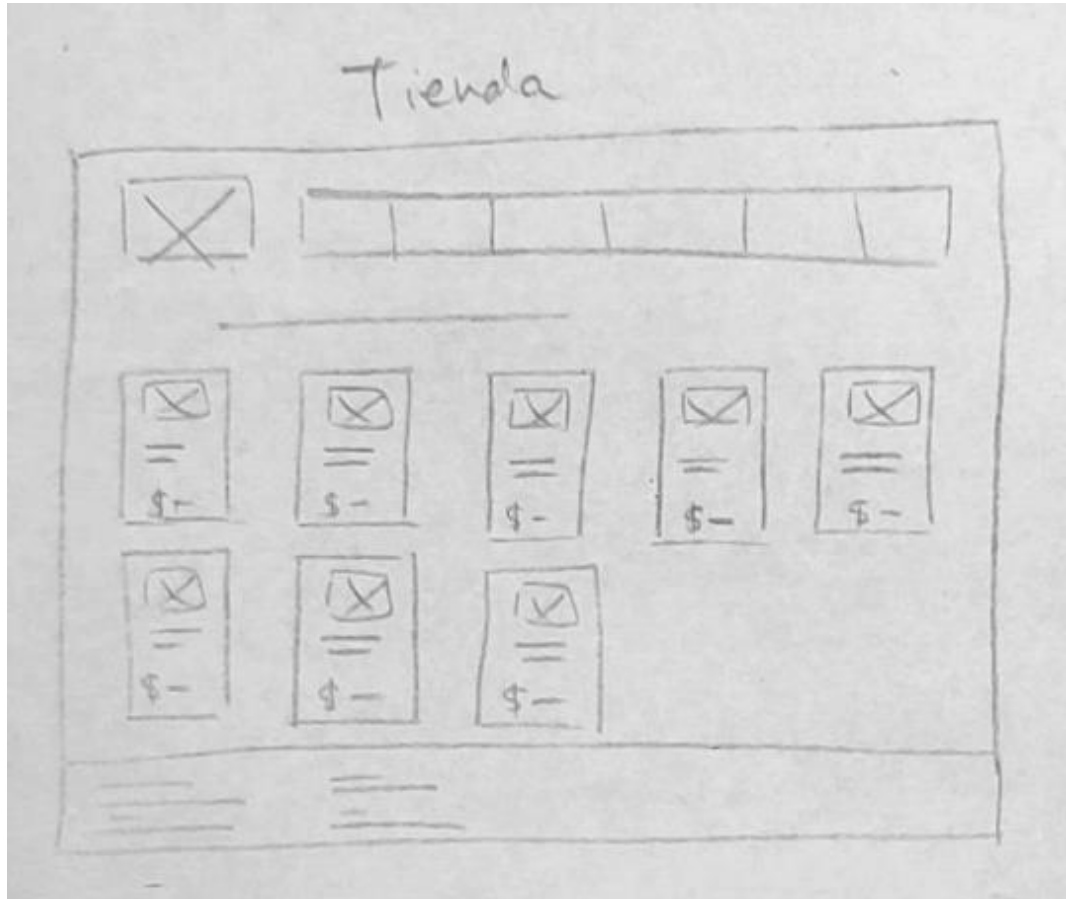
Wireframe (III)



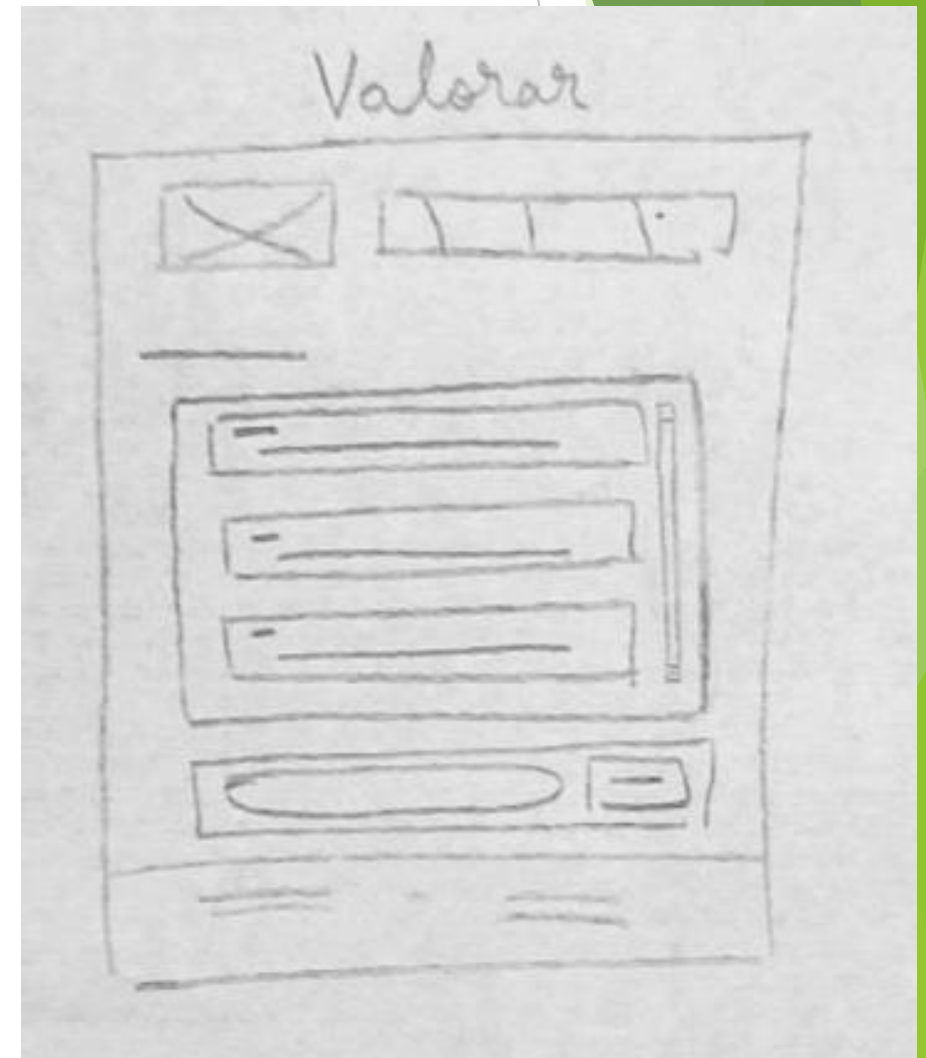
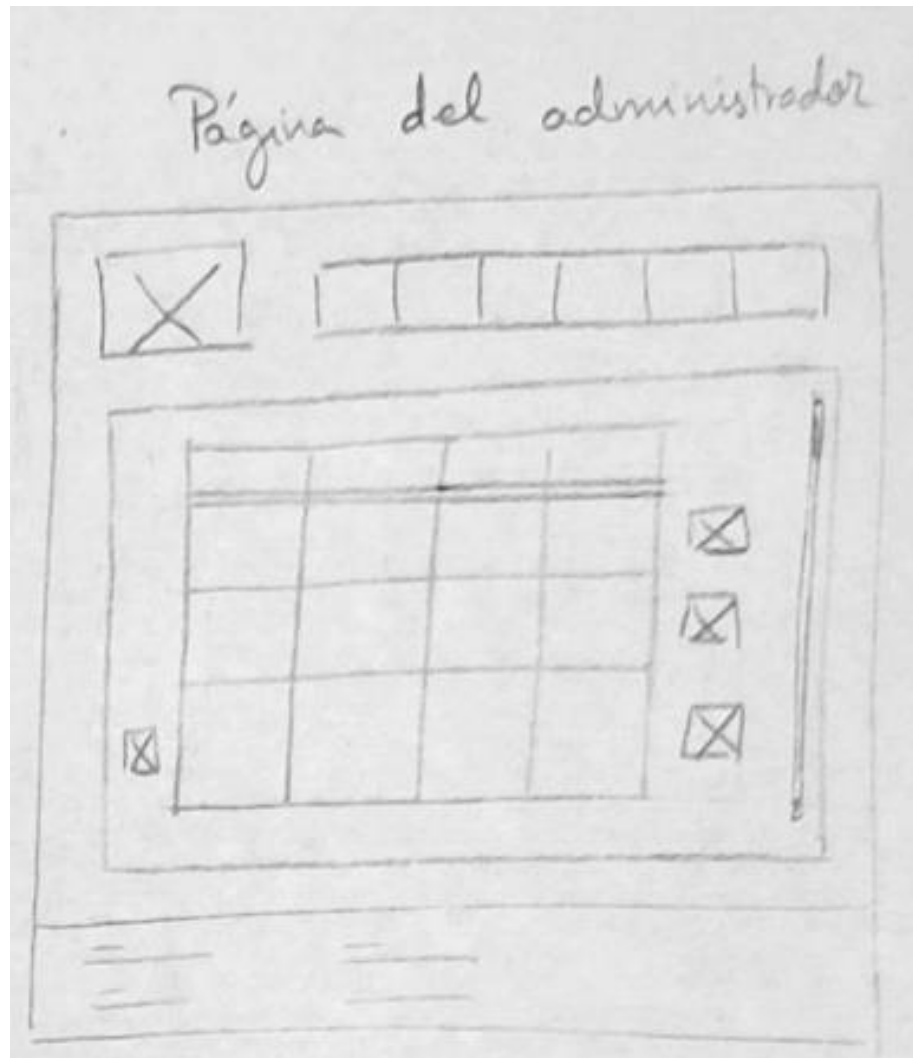
Wireframe (IV)



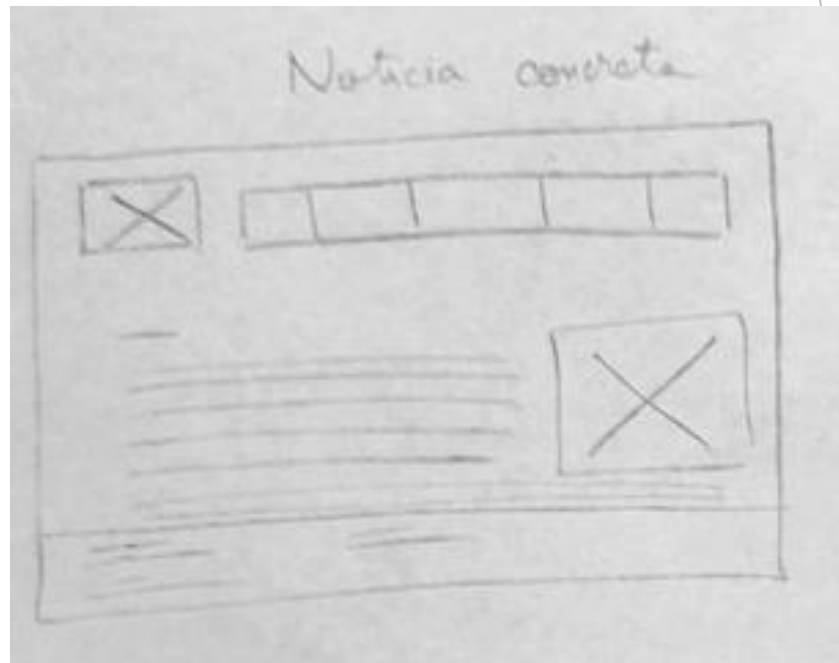
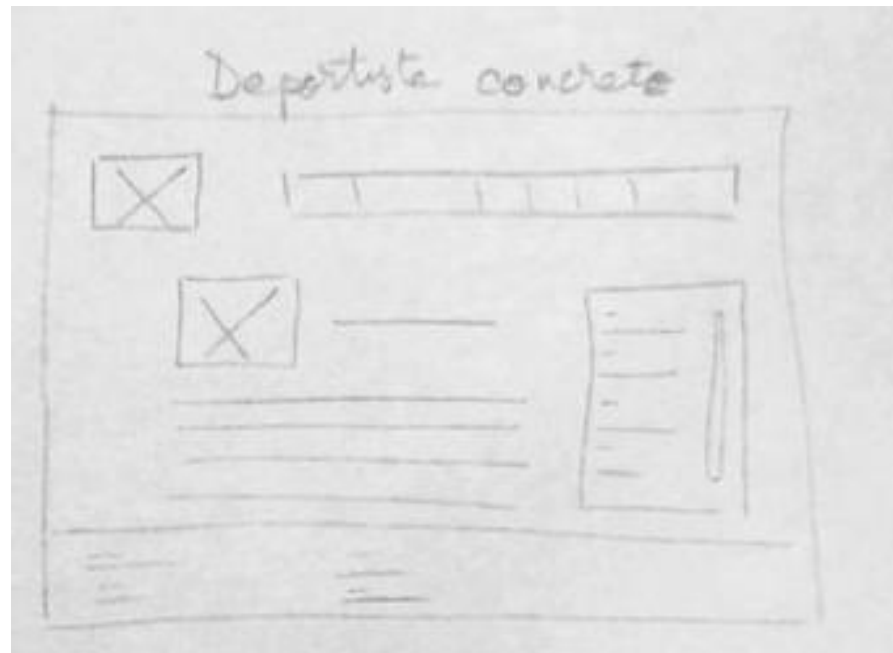
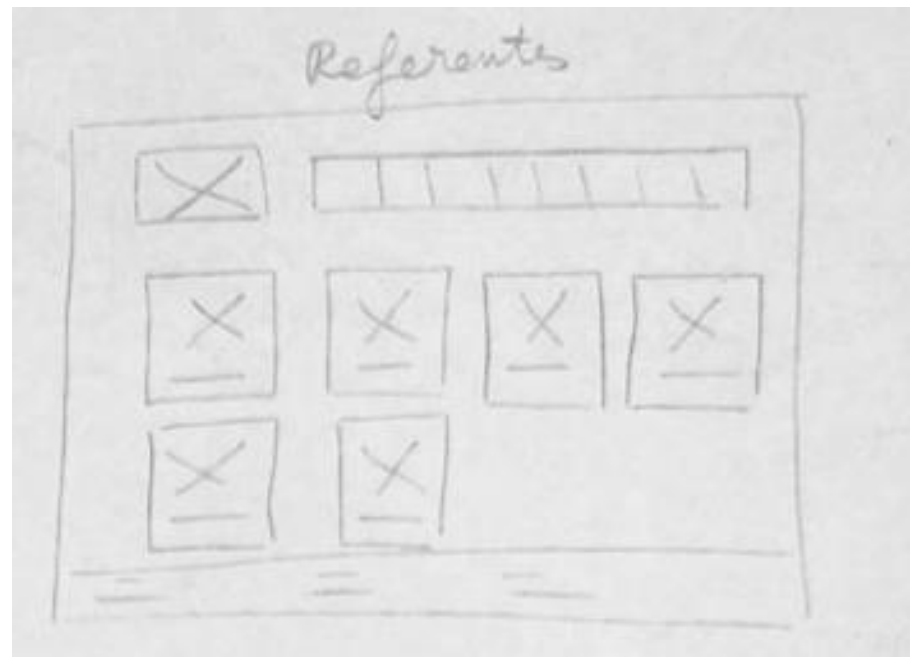
Wireframe (V)



Wireframe (VI)



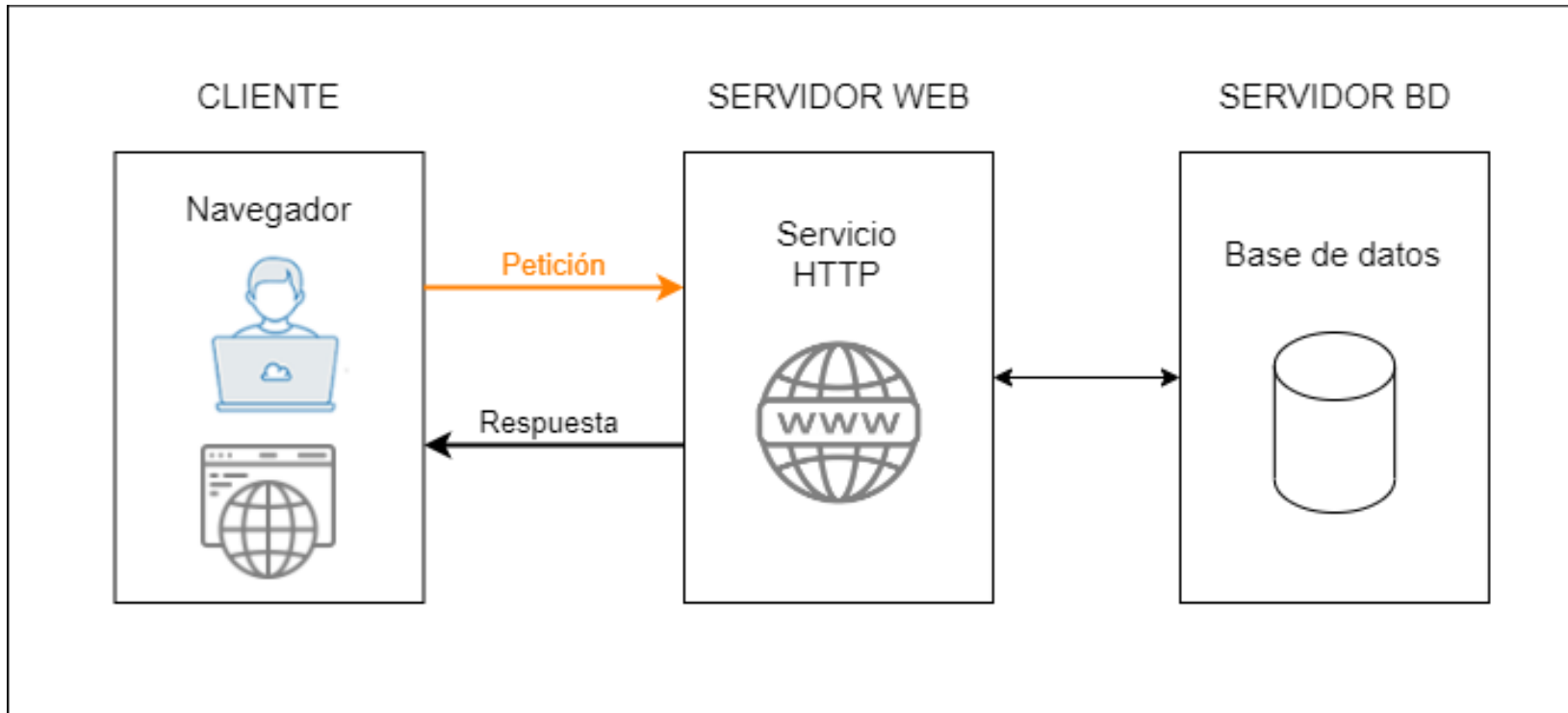
Wireframe (VII)



Índice

- ▶ ¿Qué es SPRINT?
- ▶ Diagrama de Casos de uso
- ▶ Justificación de requisitos propios
- ▶ SiteMap y Wireframes
- ▶ **Arquitectura**
- ▶ Justificación de requisitos obligatorios

Arquitectura



Índice

- ▶ ¿Qué es SPRINT?
- ▶ Diagrama de Casos de uso
- ▶ Justificación de requisitos propios
- ▶ SiteMap y Wireframes
- ▶ Arquitectura
- ▶ **Justificación de requisitos obligatorios**

Justificación de requisitos obligatorios (I)

► Front-End:

- Menú para navegar por la aplicación



Uso de CSS propio, JQuery e iconos.

Justificación de requisitos obligatorios (II)

► Back-End:

- Uso de node.js (y del framework web Express)

```
$ npm start  
> appsprint@0.0.0 start  
> node ./bin/www
```

- Hacer uso adecuado de los métodos HTTP: uso de los métodos de HTTP para el renderizado de las páginas (GET) y el envío de información a través de formularios (POST).

```
/* GET home page. */  
✓ router.get('/', function(req, res, next) {  
  | res.render('login', { title: 'Inicio de sesión', user: req.session.user, rol: req.session.  
  | });  
  
✓ router.post('/', function(req, res, next) {  
  | // se obtienen los datos del formulario  
  | let pwd = req.body.pass1;  
  | let usr = req.body.username;
```

routes/login.js

Justificación de requisitos obligatorios (III)

► Back-End:

► Login de usuarios



Nombre de usuario

Introduzca el nombre de su cuenta

Contraseña

Introduzca una contraseña

Enviar

Justificación de requisitos obligatorios (IV)

► Back-End:

► Registro de usuarios



Nombre*

Primer apellido*

Segundo apellido

Nombre de usuario*

Seleccione un perfil de usuario:

Estándar ▼

Contraseña* ⓘ

Repetir contraseña*

Enviar

Justificación de requisitos obligatorios (V)

► Back-End:

- Uso de una base de datos



```
const mysql = require("mariadb");
```

database.js

Justificación de requisitos obligatorios (VI)

► Back-End:

- Procesar al menos un formulario
- Validar en cliente los datos introducidos

`public/javascrpts/comprobaciones.js`

Notificación al usuario de errores



Nombre*

Primer apellido*

Segundo apellido

Nombre de usuario*

Seleccione un perfil de usuario:

Premium ▼

Correo Electrónico*

Error. El email ingresado no es correcto.

Contraseña* ⓘ

Repetir contraseña*

Error. La contraseña tiene que tener al menos una mayúscula.
Error. La contraseña tiene que tener al menos un número.

Enviar

Justificación de requisitos obligatorios (VII)

► Back-End:

- Procesar al menos un formulario
 - Validar en servidor los datos introducidos

routes/registro.js
+
database.js

Notificación al usuario de errores



Nombre*

Primer apellido*

Segundo apellido

Nombre de usuario*

Ya existe un usuario registrado con ese nombre de usuario

Justificación de requisitos obligatorios (VIII)

Tweets en tiempo real

► Back-End:

- Hacer uso de alguna funcionalidad en tiempo real (ej. chat, bolsa...)

Andrés Iniesta



Andrés Iniesta Luján (Fuentelabilla, Albacete, 11 de mayo de 1984), más conocido simplemente como Iniesta, es un futbolista español que juega como centrocampista en el Vissel Kobe de la J1 League de Japón.

Ha desarrollado la mayor parte de su carrera en el F. C. Barcelona y en la selección española, equipos de los que fue capitán. Es el futbolista español más laureado de la historia con un total de 39 títulos oficiales y ampliamente considerado como uno de los mejores futbolistas del mundo de su generación y uno de los mejores centrocampistas de la historia del fútbol.

Su formación como futbolista comenzó con ocho años en las categorías inferiores del Albacete Balompié. Tras una destacada actuación en el torneo nacional alevín fichó a los doce años por el F. C. Barcelona, trasladando su residencia a La Masía. En 2002 debutó en Primera División con el primer equipo azulgrana, jugando 16 temporadas (2002-2018) y conquistando 32 títulos como barcelonista: nueve de La Liga,

seis de la Copa del Rey, siete de la Supercopa de España, cuatro de Liga de Campeones de la UEFA, tres de la Supercopa de Europa y tres de la Copa Mundial de Clubes de la FIFA. Centrocampista con una técnica extraordinaria, olfato de gol y gran asistente, jugó 674 partidos oficiales con el F. C. Barcelona, anotando 57 goles y enlazando 139 asistencias de gol, convirtiéndose en uno de los jugadores más importantes de la historia del club. Es además el cuarto jugador con más partidos en la historia del Barcelona, solo por detrás de Sergio Busquets, Xavi Hernández y Lionel Messi.

Tweets from @ESPNDportes

Follow



ESPN Deportes ✓ @ESPNDportes · 19m

Sebastian Haller regresó a las canchas tras vencer el cáncer testicular que lo alejó de la actividad futbolística 🙌.

Vía: @BVB



27



Justificación de requisitos obligatorios (IX)

► Back-End:

- Capturar las excepciones de forma adecuada

```
switch (error) {  
  case 1062:  
    patron = "key ";  
    posicion = texto.search(patron)+patron.length;  
    campo = texto.substring(posicion);  
    switch (campo){  
      case ("'user_name'"):  
        mensajeError = 'Ya existe un usuario registrado con ese nombre de usuario';  
        campoError = "username";  
        break;  
      case ("'email'"):  
        mensajeError = 'Ya existe un usuario registrado con ese email';  
        campoError = "correo";  
        break;  
    }  
    break;  
}
```

database.js