

Central padel granada

Álvaro Fernández Domingo



Índice

Central padel granada.....	1
Abstract.....	3
Justificación.....	5
1. Características generales.....	5
2. Restricciones generales.....	5
3. Aspectos a cubrir.....	5
4. Comparación de prestaciones entre herramientas de la misma categoría.....	5
Justificación de la tecnología empleada.....	7
1. Front-end.....	7
2. Back-end.....	7
Requerimientos hardware y software.....	8
Hardware máquina de alojamiento del servidor y front-end.....	8
Software máquina de alojamiento del servidor y front-end.....	8
Análisis y diseño.....	9
1. Diagrama de caso de usos y de clases.....	9
2. Descripción base de datos relacional.....	10
Implementación.....	11
Hoja de estilos.....	11
Plantillas.....	11
Formularios.....	11
Funciones de envío de datos.....	11
Conexión y consultas a la base de datos.....	11
Uso de fichero de configuración.....	12
Relación detallada de cada uno de los ficheros que se incluyen.....	13
Fichero central-padel.js.....	13
Fichero device/controller.js.....	13
Fichero match/controller.js.....	13
Fichero product/controller.js.....	13
Fichero user/controller.js.....	14
Fichero userdevice/controller.js.....	14
Fichero usermatch/controller.js.....	14
Evaluación y prueba.....	15
Testeo página web.....	15
Testeo base de datos.....	15
Verificación de campos.....	15
Retroalimentación del usuario.....	15
Manual de estilos.....	16
Criterios de accesibilidad.....	16
Criterios de usabilidad.....	16
Fuentes principales y secundarias.....	16
Tamaños principales.....	17
Mapa de colores.....	18
Puntos de ruptura.....	20

Software utilizado.....	24
Mejoras posibles y aportaciones.....	25
Enlace a producción.....	25
Enlace a github.....	25

Abstract

El objetivo principal de este proyecto es la creación de una página web para la gestión de un club de padel.

En esta página hay gestión de usuarios, con sus respectivas acciones de borrar, añadir y editar, también encontramos gestión de partidos, en donde el administrador añade partidos y los usuarios pueden apuntarse a ellos, a parte el administrador puede borrar editar y añadir partidos eligiendo los usuarios que quiere que haya presentes, gestión de productos en donde un usuario puede reservar un producto para comprarlo físicamente, eligiendo la talla que desea. Encontramos tres categorías de productos: hombre, mujer y palas, esta última sin talla a elegir, además, un administrador puede borrar añadir y editar productos eligiendo las tallas y el stock de cada una.

Cuando inicias la página encuentras una landing page en la cual se da un poco de información general. En la parte superior del encabezado hay un menú hamburguesa a través del cual accedes al perfil de un usuario, en donde tienes tus reserva de productos, tus próximos partidos, tus datos disponibles para editar y tu información más relevante.

The main objective of this project is the creation of a web page for the management of a padel club.

In this page there is user management, with their respective actions to delete, add and edit, we also find match management, where the administrator adds matches and users can sign up for them, besides the administrator can delete edit and add matches choosing the users you want to be present, product management where a user can book a product to buy it physically, choosing the size you want. We find three categories of products: men, women and shovels, the latter without size to choose, in addition, an administrator can delete add and edit products by choosing the sizes and stock of each.

When you start the page you find a landing page in which some general information is given. At the top of the header there is a hamburger menu through which you access a user's profile, where you have your product stock, your next matches, your data available for editing and your most relevant information

Justificación

1. Características generales

Página dedicada a la gestión del club de padel centralpadel, con la intención de controlar las reservas de productos tanto de hombre y mujer como de palas y gestionar los partidos disponibles lanzados por un administrador.

2. Restricciones generales

El tiempo: la principal restricción que he encontrado ha sido el tiempo, ya que a parte de las 7 horas diarias en las practicas tenia que echar horas por la tarde al proyecto y no he tenido tanto tiempo como me gustaría, por lo que he tenido que recortar mucho la aplicación.

3. Aspectos a cubrir

- Gestión de usuarios con registro y logueo controlado a través de la api
- Gestión de productos con imágenes almacenadas en la base de datos y backend
- Gestión de partidos

4. Comparación de prestaciones entre herramientas de la misma categoría

Gestión de usuarios: La gestión de usuarios en esta aplicación es una gestión de usuarios de alto nivel, con seguridad utilizando el sistema de hash PBKDF2 el cual utiliza además del hash un salt para hacerlo mas seguro. Para el recordatorio de inicio de sesión se guarda un dato del navegador con el correo correspondiente al usuario para así iniciar sesión automáticamente si se ha dejado iniciada previamente.

Gestión de partidos: La gestión de partidos de este proyecto es muy parecida a la de la mayoría de la misma categoría añadiendo usuarios a los partidos y añadiendo partidos a una tabla dividida por hora día fecha y pista.

Gestión de productos: La gestión de productos es como la mayoría que se encuentra en esta categoría, reserva de productos con precios y tallas.

Justificación de la tecnología empleada

1. Front-end

Vue: He utilizado vue como framework de javascript por varios motivos:

- Separación de componentes y estilos, código y html en el mismo archivo
- Pinia como store para metodos y variables globales
- Para aprender mas vue y así tener un proyecto grande en cada framework
- Variables reactivas y v-if y v-for
- SCSS y componentes scoped con su propio scss

2. Back-end

Node: He utilizado node en el backend por los siguientes motivos:

- Herramientas como express y multer
- Organización de carpetas con controladores, configuración, middlewares y archivo principal
- Envío de respues del servidor como y sencillo
- Backend escrito en javascript

Requerimientos hardware y software

Hardware máquina de alojamiento del servidor y front-end



centralpadel

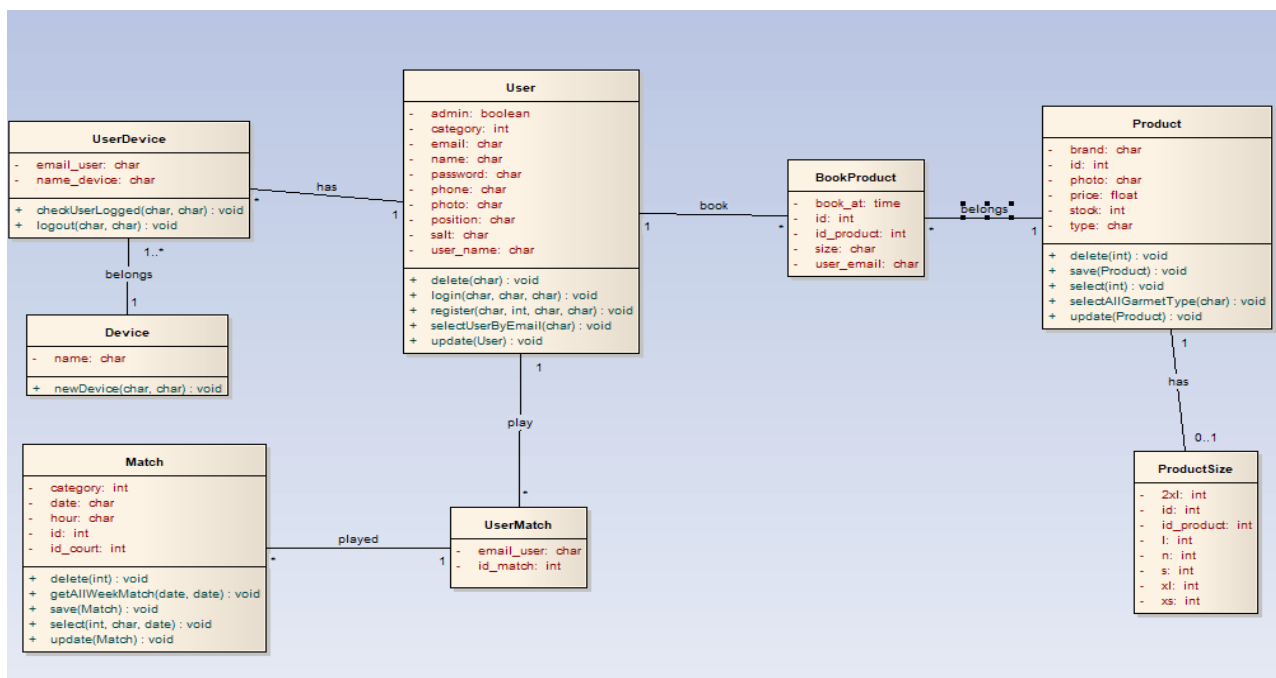
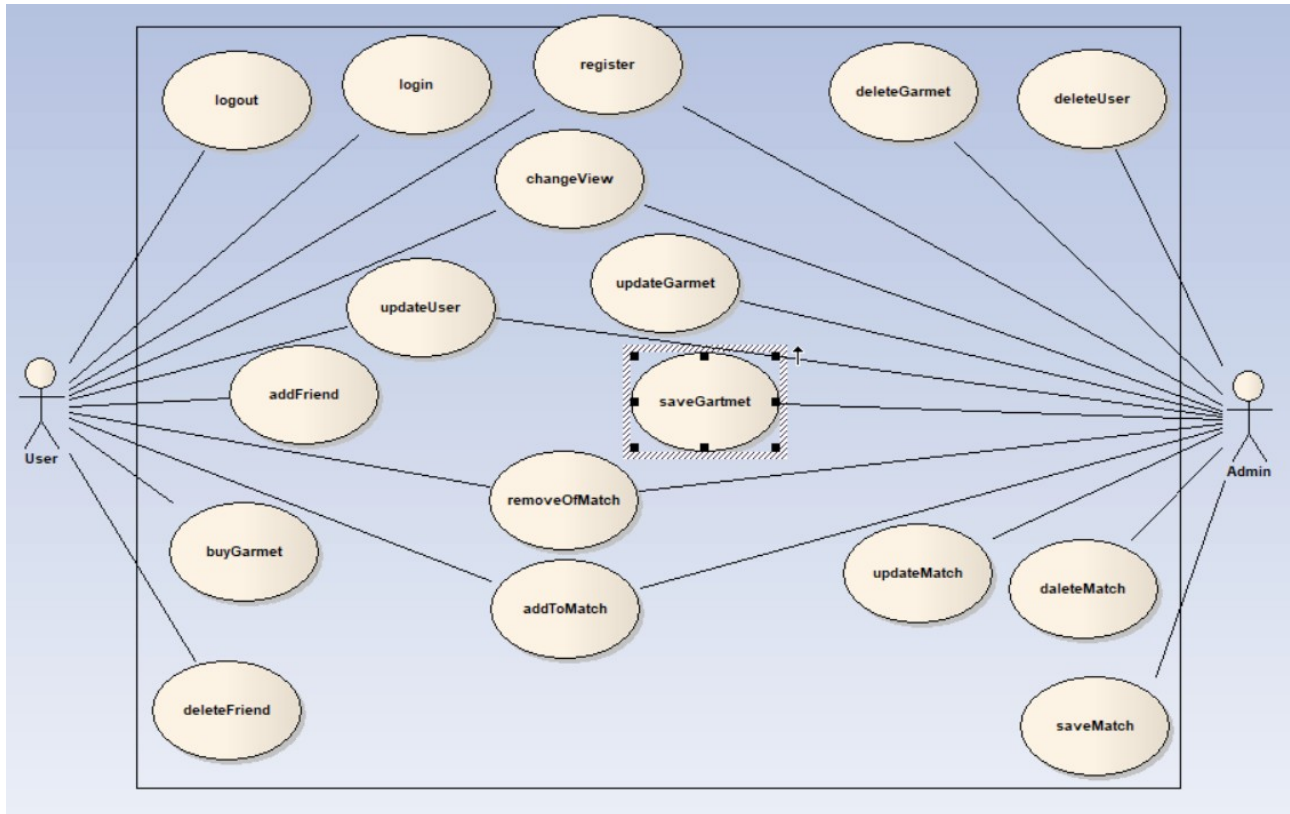
in  Central Padel / 1 GB Memory / 1 AMD vCPU / 25 GB Disk / FRA1 - Ubuntu 22.10 x64

Software máquina de alojamiento del servidor y front-end

- mysql Ver 8.0.33
- Ubuntu 22.10
- node Ver 18.7.0

Análisis y diseño

1. Diagrama de caso de usos y de clases



2. Descripción base de datos relacional

La base de datos tiene tres tablas principales: user, match y product.

La tabla user esta relacionada con la tabla device a través de userdevice, en esta se guardan los dispositivos en los que ha iniciado sesión cada usuario.

La tabla user también esta relacionada con la tabla match a través de user match, en donde se almacenan los jugadores de cada partido.

Por último la tabla user está relacionada con la tabla product a través de la tabla bookproduct, en donde se almacena el producto reservado con la talla correspondiente. Para saber que tallas tiene cada producto está la tabla prodcutsize, en donde se almacena el stock de cada talla de cada producto.

Implementación

Hoja de estilos

Uso de scss para los estilos de la página, cada componente tiene su propia parte de style.

Hay un archivo global de scss llamado style.scss en el cual se importa el archivo _variables.scss en donde están todas las variables globales que utilizo, también se importa el archivo _mixin.scss en donde se encuentran todos los mixin que utilizo, el archivo _main.scss en donde están los estilos globales de la página y por último el _base.scss que es en donde están todas las variables del root que trae por defecto scss.

Plantillas

Para las plantillas uso vue, el cual está dividido en componentes y vistas, el componente principal es el app.vue, que es donde se importan las rutas. Las vistas son las rutas principales que nos encontramos en la página, los componentes son partes de cada ruta que sirven para reutilizarlos. Cada vista y componente está dividido en tres partes: script (donde se encuentra el código js), template (donde se encuentra el html) y style (donde se encuentra el scss).

Formularios

Los formularios principales de la página son el de registro e inicio de sesión, y el editar y añadir productos, partidos y usuarios. Todos estos tienen sus respectivas expresiones regulares para validar los campos.

Funciones de envío de datos

Las funciones de envío de datos se han hecho todas en los stores de vue, hay una url principal que lleva a la raíz de la api y cada petición tiene una suburl que es a qué parte de la api se hace la petición. Las peticiones se han hecho con fetch y son recogidas en node, cada petición el método por el que se envía, el header que es donde se declara qué tipo de datos se envían y el body que es por donde se envían los datos en formato json.

Conexión y consultas a la base de datos

Una vez llega la petición al backend, este recibe los datos enviados. La petición llama a un método del backend y este hace las respectivas consultas a la base de datos con los datos

proporcionados, una vez que se realizan todas las acciones necesarias, se envía una respuesta al cliente, el cual la recibe y hace lo que corresponda en cada caso según la respuesta enviada.

Uso de fichero de configuración

Como ficheros de configuración encontramos varios.

En el frontend nos encontramos con el `package.json`, que es donde se especifican las dependencias del proyecto en cuanto al frontend, este mismo archivo se encuentra en el backend especificando las dependencias del mismo, además en el backend encontramos una carpeta llamada `config` con tres archivos: `connection.js` (donde se declara la variable de conexión a la base de datos), `database.js` (donde se especifican las diferentes configuraciones de la conexión de la base de datos y el puerto en el que se levanta el servidor) y el `handle.js` (donde se realiza la conexión a la base de datos)

Relación detallada de cada uno de los ficheros que se incluyen

Fichero central-padel.js

- **Descripción:** Es el fichero principal del backend, en este se reciben todas las peticiones a la api y es el encargado de llamar a los métodos correspondientes.
- **Acceso a base de datos:** No
- **Llamado por:** Todos los ficheros de stores del frontend
- **Redirige a:** device/controller.js, match/controller.js, product/controller.js, user/controller.js, usermatch/controller.js y userdevice/controller.js,
- **Parámetros:** res, req, connection

Fichero device/controller.js

- **Descripción:** Es el fichero encargado de los métodos de la tabla device.
- **Acceso a base de datos:** Si. Select, Insert
- **Llamado por:** central-padel.js
- **Parámetro:** res, req, db

Fichero match/controller.js

- **Descripción:** Es el fichero encargado de los métodos de la tabla match.
- **Acceso a base de datos:** Si. Select, Insert, Delete, Update
- **Llamado por:** central-padel.js
- **Parámetro:** res, req, db

Fichero product/controller.js

- **Descripción:** Es el fichero encargado de los métodos de la tabla product.
- **Acceso a base de datos:** Si. Select, Insert, Delete, Update
- **Llamado por:** central-padel.js
- **Parámetro:** res, req, db

Fichero user/controller.js

- **Descripción:** Es el fichero encargado de los métodos de la tabla user.
- **Acceso a base de datos:** Si. Select, Insert, Delete, Update
- **Llamado por:** central-padel.js
- **Parámetro:** res, req, db

Fichero userdevice/controller.js

- **Descripción:** Es el fichero encargado de los métodos de la tabla userdevice.
- **Acceso a base de datos:** Si. Select, Insert, Delete
- **Llamado por:** central-padel.js
- **Parámetro:** res, req, db

Fichero usermatch/controller.js

- **Descripción:** Es el fichero encargado de los métodos de la tabla usermatch.
- **Acceso a base de datos:** Si. Select, Insert, Delete, Update
- **Llamado por:** central-padel.js
- **Parámetro:** res, req, db

Evaluación y prueba

El testeo de toda la funcionalidad de la página ha sido de forma manual, insertando los datos a mano y comprobando las diferentes posibilidades tanto las esperadas como las de error.

Testeo página web

Para el testeo de la página web se ha probado todas las funcionalidades de la página tanto de la parte pública como de la parte privada probando que todo funcionase y forzando errores de manera que estuviesen comprobados. También se han probado casos poco comunes.

Testeo base de datos

Para el testeo de la base de datos se han llevado a cabo pruebas tanto manuales como con scripts. Las pruebas manuales han sido realizadas para controlar todas las posibilidades que ofrecen los métodos que se conectan a la base de datos, controlando errores de sintaxis o casos inesperados. Por parte de los scripts se han insertado datos perjudiciales a la base de datos a través de estos para controlar que le hacían a la base de datos y como se veían en la página.

Verificación de campos

Se han verificado y utilizado expresiones regulares tanto en el frontend como en el backend para los campos de entrada haciendo así que solo haya posibilidad de introducir en cada campo los datos que concuerden con el formato de la base de datos.

Retroalimentación del usuario

Para la retroalimentación del usuario se han empleado toast tanto de éxito como de error para dar al usuario la información necesaria sobre la acción que ha realizado.

Manual de estilos

Criterios de accesibilidad

- Accesible desde cualquier dispositivo
- Accesible para cualquier persona con el uso de atributos como alt
- Texto legible y comprensible
- Contraste de colores
- Manejo desde el teclado

Criterios de usabilidad

- Relación entre el sistema y el mundo real
- Consistencia y estándares
- Flexibilidad y eficiencia de uso
- Diseño estético y minimalista
- Facilidad de navegación

Fuentes principales y secundarias

<https://fonts.googleapis.com/css2?family=Bevan&display=swap>

<https://fonts.googleapis.com/css2?family=Ubuntu:wght@300;500&display=swap>

<https://fonts.googleapis.com/css2?family=Maitree:wght@500&display=swap>

```
/*fonts*/  
$h-f-family-title: 'Bevan', cursive;  
$h-f-family-subtitle: 'Ubuntu', sans-serif;  
$h-f-family-text: 'Maitree', serif;
```


Tamaños principales

```
$h-f-size-extra: 48px;
$h-f-size-title: 36px;
$h-f-size-subtitle: 24px;
$h-f-size-text: 16px;
$h-f-size-text-small: 13px;
```

```
h1{
  /* decoration */
  font-family: $h-f-family-title;
  font-size: $h-f-size-title;
}

h2, h3, h4{
  /* decoration */
  font-family: $h-f-family-subtitle;
}

p, a, span, li, ul, ol, button, input, textarea, select, label, option, optgroup, legend, fieldset, form, table, caption, tbody, tfoot, thead, tr, th, td, article, aside, details, figcaption,
figure, footer, header, hgroup, main, menu, nav, section, summary, time, mark, audio, video, canvas, progress, meter, output, details, dialog, menuitem, summary, datalist, keygen, command,
embed, object, param, source, track, wbr, bdi, bdo, rp, rt, ruby, iframe, noframes, picture, svg, template, applet, basefont, big, center, font, frame, frameset, isindex, strike, tt, xmp,
acronym, applet, basefont, big, center, dir, font, frame, frameset, isindex, strike, tt, xmp, acronym, dir, noframes, plaintext, rb, rtc, strike, tt, xmp, acronym, applet, basefont, big,
center, dir, font, frame, frameset, isindex, strike, tt, xmp, acronym, dir, noframes, plaintext, rb, rtc, strike, tt, xmp, acronym, applet, basefont, big, center, dir, font, frame,
frameset, isindex, strike, tt, xmp, acronym, dir, noframes, plaintext, rb, rtc, strike, tt, xmp, acronym, applet, basefont, big, center, dir, font, frame, frameset, isindex, strike,
tt, xmp, acronym, dir, noframes, plaintext, rb, rtc, strike, tt, xmp, acronym, applet, basefont, big, center, dir, font, frame, frameset, isindex, strike, tt, xmp, acronym, dir,
noframes, plaintext, rb, rtc, strike, tt, xmp, acronym, applet, basefont, big, center, dir, font, frame, frameset, isindex, strike, tt, xmp, acronym, dir,
rtc, strike, tt, xmp, acronym, applet, basefont, big, center, dir, font, frame, frameset, isindex{
  /* decoration */
  font-family: $h-f-family-text;
  font-size: $h-f-size-text;
}

/* router-link */
a {
  /* decoration */
  text-decoration: none;
  color: $h-c-black;
}
```

Mapa de colores

Blanco:

```
// white
$h-c-white: ■ #ffffff;
$h-c-white-contrast: ■ #000000;
$h-c-white-shade: ■ #c8c8c8;
$h-c-white-light: ■ rgb(244, 244, 244);
$h-c-white-tint: ■ #ffffff;
```





Negro:

```
// black
$h-c-black: ■ #000000;
$h-c-black-contrast: ■ #ffffff;
$h-c-black-shade: ■ #000000;
$h-c-black-tint: ■ #1a1a1a;
```





Rojo logo:

```
// red-logo
$h-c-red-logo: ■ #D20300;
$h-c-red-logo-contrast: ■ #ffffff;
$h-c-red-logo-shade: ■ #b90300;
$h-c-red-logo-tint: ■ #d71c1a;
```





Rojo claro:

```
// red-light
$h-c-red-ligth:  #ff4242;
$h-c-red-ligth-contrast:  #ffffff;
$h-c-red-ligth-shade:  #e03a3a;
$h-c-red-ligth-tint:  #ff5555;
```

Gris:

```
// gray
$h-c-gray:  #f0f0f0;
$h-c-gray-contrast:  #000000;
$h-c-gray-shade:  #dedede;
$h-c-gray-tint:  #fcfcfc;
```

Azul:

```
//blue
$h-c-blue:  #007bff;
$h-c-blue-contrast:  #ffffff;
$h-c-blue-shade:  #006ce0;
$h-c-blue-tint:  #1a88ff;
```

Gris oscuro:

```
//dark-gray
$h-c-dark-gray: #2c3e50;
$h-c-dark-gray-contrast: #ffffff;
$h-c-dark-gray-shade: #273746;
$h-c-dark-gray-tint: #415162;
```

Puntos de ruptura

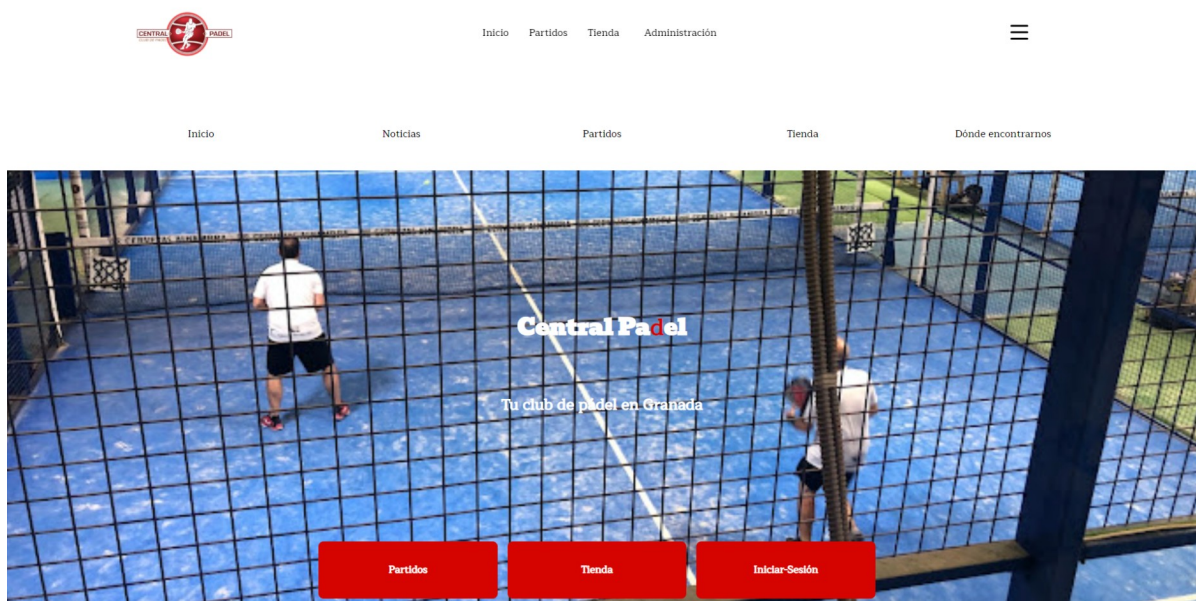
Monitores grandes: 1300px

Tablets grandes y monitores pequeños: 1000px

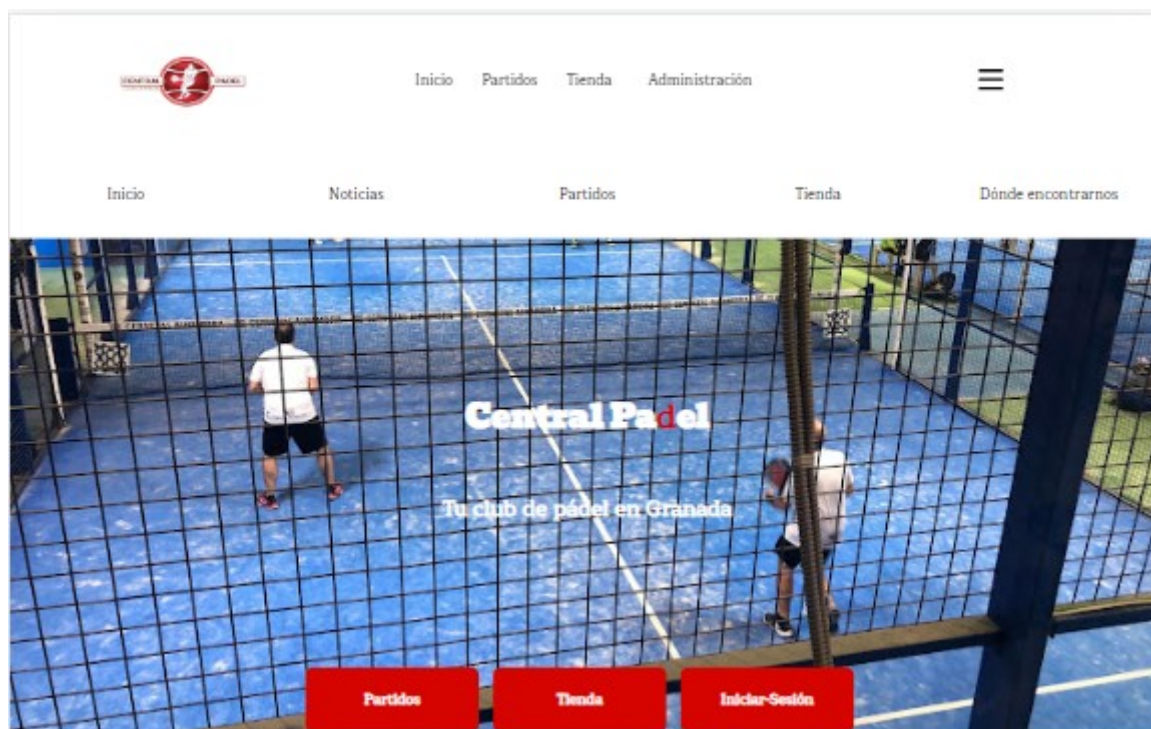
Tablets pequeñas: 800px

Móvil: 500px

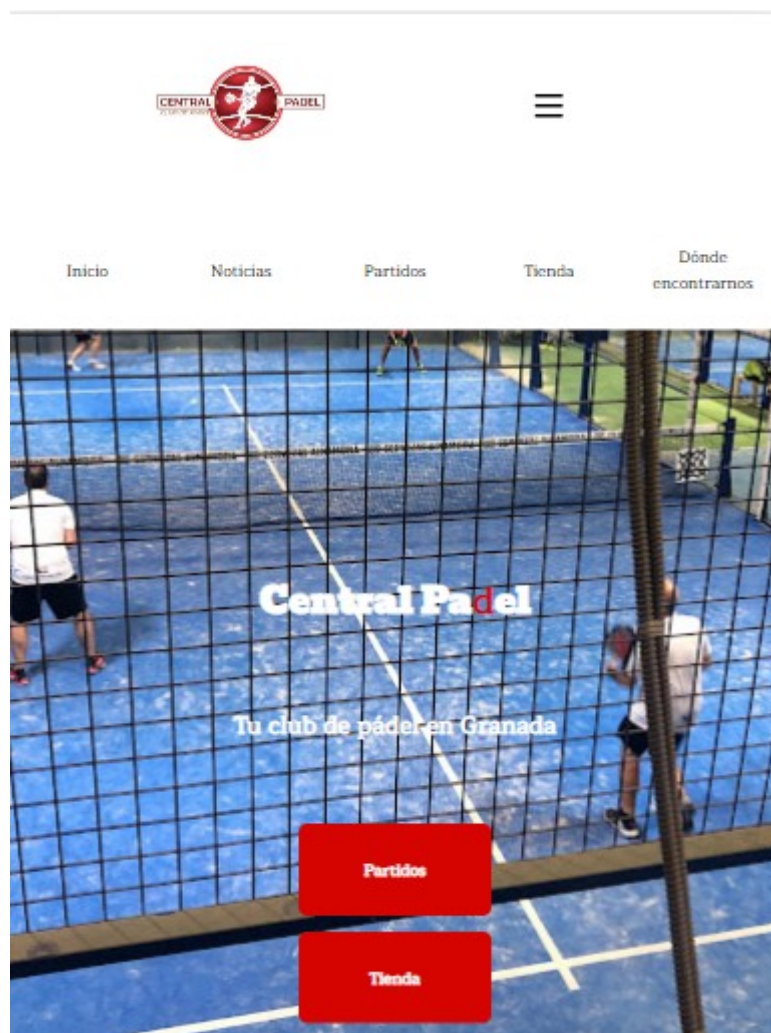
Monitor grande:



Monitor pequeño:



Tablet pequeña:



Móvil:



Software utilizado

- Vue.js:
 - Utilizado para el frontend
- Node.js
 - Utilizado para el backend
- SCSS
 - Utilizado par los estilos
- MySql
 - Utilizado para la base de datos
- DigitalOcean
 - Utilizado para el despliegue de la aplicación
- Git y GitHub
 - Utilizados para el control de versiones

Mejoras posibles y aportaciones

- Implementación de mas funcionalidades como amistades o chat
- Reutilización de código más organizada y mejor planteada
- Mejor documentación de código para mayor productividad
- Implementación de herramientas de testeo

Enlace a producción

<http://centralpadelgranada.me/>

Enlace a github

<https://github.com/alvarofernandez/Central-Padel-Server>