

Proyecto Final ASIR

Página Web: Work Hard



15 de junio de 2020

IES Cánovas del castillo

Málaga

**Introducción2**

Objetivo2

Herramientas utilizadas2

**Base de datos5**

**Aspectos Técnicos9**

**Manual de usuario17**

**INTRODUCCIÓN**

Objetivo:

Este proyecto tiene como objetivo crear una forma rápida y sencilla para el usuario a la hora de darse de alta, reservar actividades, darse de baja, …

De esta forma intentamos que el usuario pueda usar la página rápidamente para la reserva de las actividades proporcionadas por el gimnasio. Miramos por la facilidad del usuario y de los trabajadores, el proyecto es escalable, puede añadirse opciones que se sean necesarias.

Herramientas utilizadas:

La página web ha sido realizada con:

Bootstrap 4: es uno de los **frameworks** más populares utilizado para crear sitios web y aplicaciones responsivas y fiables. Inicialmente desarrollado por Twitter, en los últimos años ha recibido muchas modificaciones, cada una mejorándolo y aumentando su popularidad. Una de las características que mejor puede definir a Bootstrap es que resulta bastante sencillo de manejar, ya que **hace uso de un sistema de parrilla o grid con 12 columnas** en las que se puede insertar el contenido. Ésta es la clave para que los sitios web realizados con Bootstrap tengan un **diseño responsive**, ya que los desarrolladores pueden gestionar todas esas columnas según cuales sean sus necesidades **para adaptarse a los tamaños de los distintos dispositivos**.

**HTML**: Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros. Es un estándar a cargo del *World Wide Web Consortium* (W3C) o Consorcio WWW, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación. HTML se considera el lenguaje web más importante siendo su invención crucial en la aparición, desarrollo y expansión de la World Wide Web (WWW). Es el estándar que se ha impuesto en la visualización de páginas web y es el que todos los navegadores actuales han adoptado.

**CSS**: es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado.​ Es muy usado para establecer el diseño visual de los documentos web, e interfaces de usuario escritas en HTML o XHTML; el lenguaje puede ser aplicado a cualquier documento XML, incluyendo XHTML, SVG, XUL, RSS, etcétera. Te puede ayudar a crear tu propio sitio web. Junto con HTML y JavaScript, CSS es una tecnología usada por muchos sitios web para crear páginas visualmente atractivas, interfaces de usuario para aplicaciones web y GUIs para muchas aplicaciones móviles (como Firefox OS).

CSS está diseñado principalmente para marcar la separación del contenido del documento y la forma de presentación de este, características tales como las capas o *layouts*, los colores y las fuentes. Esta separación busca mejorar la accesibilidad del documento, proveer más flexibilidad y control en la especificación de características presentacionales, permitir que varios documentos HTML compartan un mismo estilo usando una sola hoja de estilos separada en un archivo .css, y reducir la complejidad y la repetición de código en la estructura del documento.

**JAVASCRIPT**: Se utiliza principalmente del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas y JavaScript del lado del servidor (*Server-side JavaScript* o *SSJS*). Su uso en aplicaciones externas a la web, por ejemplo, en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar a C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo, Java y JavaScript tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. Actualmente es ampliamente utilizado para enviar y recibir información del servidor junto con ayuda de otras tecnologías como AJAX. JavaScript se interpreta en el agente de usuario al mismo tiempo que las sentencias van descargándose junto con el código HTML.

PHPMYADMIN: es una herramienta gratuita y de código abierto escrita en PHP destinada a manejar la administración de MySQL con el uso de un navegador web. Puede realizar diversas tareas como crear, modificar o eliminar bases de datos, tablas, campos o filas; ejecutar sentencias SQL o gestionar usuarios y permisos. El software, que está disponible en 78 idiomas,​ es mantenido por *The phpMyAdmin Project*.

Puede importar datos de CSV y SQL, y transformar los datos almacenados en cualquier formato utilizando un conjunto de funciones predefinidas, como la visualización de datos BLOB como imágenes o enlaces de descarga.

MySQL es muy utilizado en aplicaciones web, como Joomla, Wordpress, Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación.

**MAILGUN**: El producto está construido para desarrolladores que necesitan una simple integración de la API de correo electrónico para enviar, recibir y rastrear correo electrónico con su aplicación, ya sea que utilicen Python, Ruby, PHP, C#, node.js o Java.

Un conjunto de potentes API que te permiten enviar, recibir y realizar un seguimiento del correo electrónico desde tu aplicación sin esfuerzo, aún si utilizas Python, Ruby, PHP, C#, node.js o Java.

La fácil integración de SMTP y una sencilla API RESTful abstrae los detalles desordenados de enviar correo electrónico transaccional o masivo. Escala rápidamente, ya sea que necesite enviar 10 o 10 millones de correos electrónicos.

Con registros indexados para búsqueda siempre sabrás lo que está sucediendo con tu correo electrónico, mientras que las etiquetas facilitan la comprobación A/B e informan de tus datos, todo a través de los ganchos web.

IONOS: combina dominios, correo electrónico y páginas web con hosting de primera clase, soluciones cloud escalables y asesor personal.

IONOS es el aliado en soluciones de hosting y cloud para las pymes. Ofrecemos nuestros servicios a autónomos que dan sus primeros pasos online, a dueños de negocios crecientes, a grandes empresas y a socios que les estén ayudando a alcanzar sus metas. No importa si estás construyendo tu presencia online comprando un dominio, si estás creando tu página web, trasladando procesos de back-office a la cloud u organizando un cluster de contenedores, nuestro portafolio de productos te ofrece aplicaciones, servidores y servicios de primera.

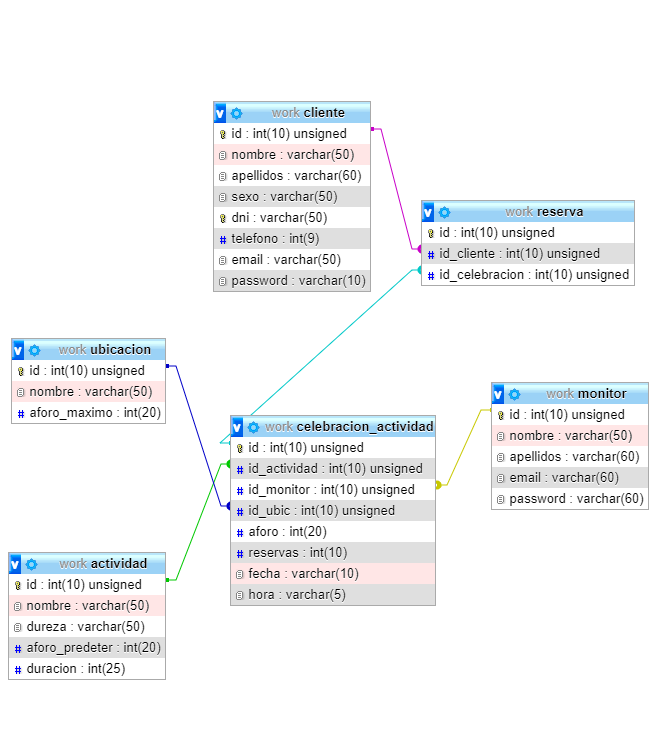
XAMPP: **e**s un paquete de software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl. El nombre es en realidad un acrónimo: **X** (para cualquiera de los diferentes sistemas operativos), **A**pache, **M**ariaDB/MySQL, **P**HP, **P**erl. A partir de la versión 5.6.15, XAMPP cambió la base de datos MySQL por MariaDB, un fork de MySQL con licencia GPL.

El programa se distribuye con la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. A esta fecha, XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris y Mac OS X.

*BASE DE DATOS*

La base de datos consta de las siguientes tablas:

* Cliente
* Actividad
* Monitor
* Celebracion\_actividad
* Reserva
* Ubicación



La tabla monitor está relacionada con celebracion\_actividad, esta relación no tienen ninguna configuración como delete en cascada porque si se borra un monitor la clase se mantiene y se cambia el monitor por otro que esté disponible.

La tabla cliente se relaciona con la tabla reserva de forma que si un cliente se da de baja todas las reservas se eliminan.

La tabla celebracion\_actividad está relacionada con la tabla reserva para que al eliminar una clase todas las reservas de dicha clase se eliminen.

La tabla actividad está relacionada con la tabla celebracion\_actividad para que al eliminar un actividad se eliminen las clases de dicha actividad.

La tabla UBICACION está relacionada con la tabla celebracion\_actividad para que, al borrar una ubicación, si hay clases que se llevarían a cabo allí estas se eliminaran.

La base de datos está creada en PHPMyAdmin.

El archivo para crear la base de datos con las foreign keys es:

DROP DATABASE IF EXISTS work;

CREATE DATABASE work CHARACTER SET utf8mb4;

USE work;

CREATE TABLE cliente (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

apellidos VARCHAR(60) NOT NULL,

sexo VARCHAR(50) NOT NULL,

dni VARCHAR(50) UNIQUE,

telefono INT (9) NOT NULL,

email VARCHAR (50) NOT NULL

);

CREATE TABLE actividad (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

dureza VARCHAR(50) NOT NULL,

aforo\_predeter INT(20),

duracion INT(25) NOT NULL

);

CREATE TABLE monitor (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

apellidos VARCHAR(60) NOT NULL,

email VARCHAR(60) NOT NULL,

password VARCHAR(60) NOT NULL

);

CREATE TABLE ubicacion (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

aforo\_maximo INT(20) NOT NULL

);

CREATE TABLE celebracion\_actividad (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

id\_actividad INT UNSIGNED,

id\_monitor INT UNSIGNED,

id\_ubic INT UNSIGNED,

aforo INT(20) NOT NULL,

reserva INT(20) NOT NULL,

fecha VARCHAR (10) NOT NULL,

hora VARCHAR (5) NOT NULL

FOREIGN KEY (id\_actividad) REFERENCES actividad(id) ON DELETE CASCADE,

FOREIGN KEY (id\_monitor) REFERENCES monitor(id),

FOREIGN KEY (id\_ubic) REFERENCES ubicacion(id) ON DELETE CASCADE

);

CREATE TABLE reserva (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

id\_cliente INT UNSIGNED,

id\_celebracion INT UNSIGNED,

FOREIGN KEY (id\_cliente) REFERENCES cliente(id) ON DELETE CASCADE,

FOREIGN KEY (id\_celebracion) REFERENCES celebracion\_actividad(id) ON DELETE CASCADE

);

Las tablas están relacionadas con una eliminación en cascada de forma que:

Si un cliente se da de baja y tiene reservada alguna actividad, esta reserva desaparece y el contador de reservas restaría 1 para que no exista ningún conflicto cuando se vuelva a reservar.

Si se elimina una actividad se borrará la celebración correspondiente en la tabla celebracion\_actividad y todas las reservas de esa actividad.

Si se elimina la ubicación se eliminará las celebraciones de las actividades que se desarrollen en esa ubicación porque no podrán realizarse.

*Aspectos técnicos*

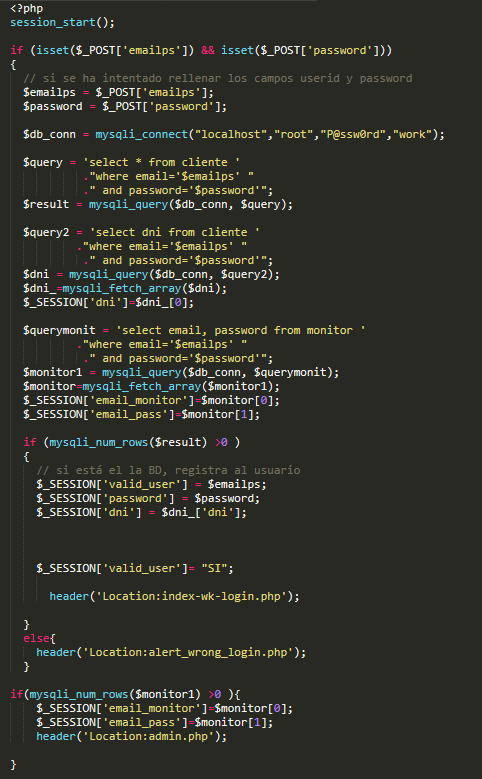
Empezando desde el principio tenemos el login:

el login es el mismo para el cliente y para los técnicos,

el cliente accede con su email que proporciona en el registro y el técnico accede con el email que le otorgan al entrar a trabajar en el gimnasio.

Los email de los técnicos se caracterizan porque todos tienen @workhard.es

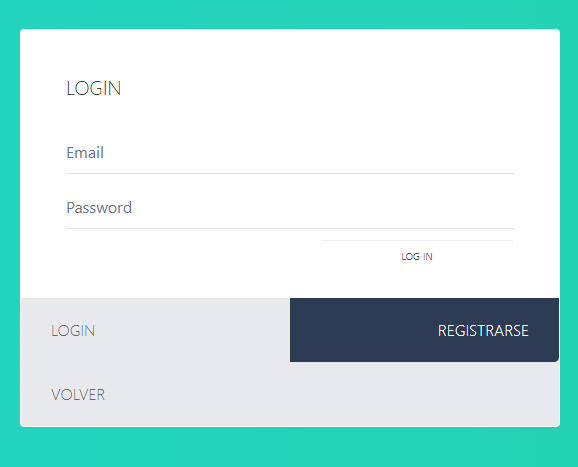
En el código se hace un select en la tabla “monitor” para que puedan acceder a su página correspondiente.



código login

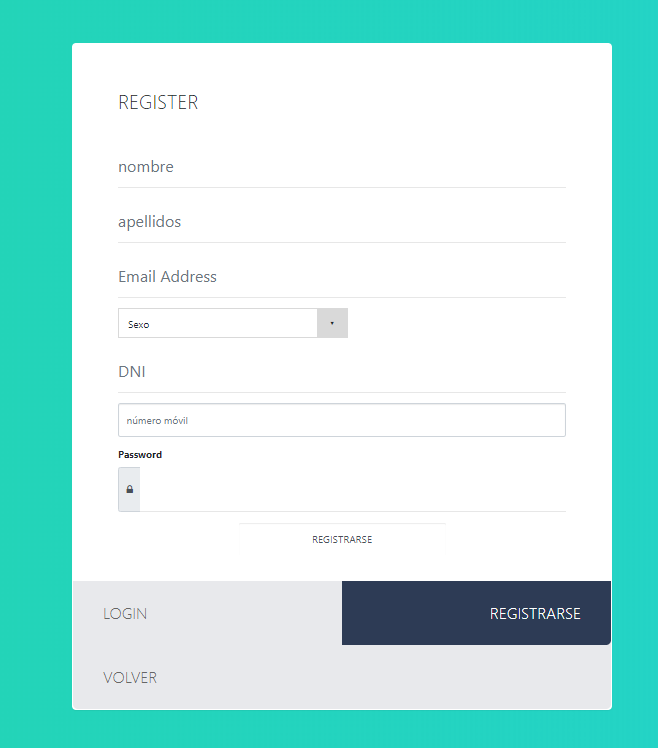
A continuación, vemos la pantalla de login:

Aquí es donde el usuario o el técnico acceden a sus páginas correspondientes



login

Si se pulsa el botón “Registrarse” veremos el panel de registro donde el usuario pondrá sus datos para poder darse de alta y poder acceder a la página.

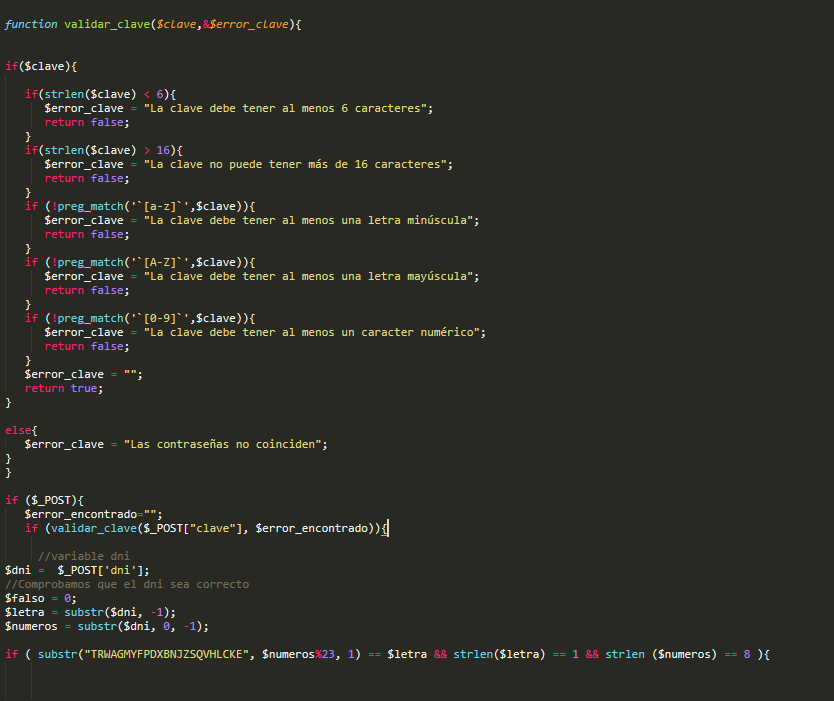


registro

El registro tiene la seguridad para que la contraseña tenga obligatoriamente una letra mayúscula, una letra minúscula, un número y mínimo de 6 caracteres.

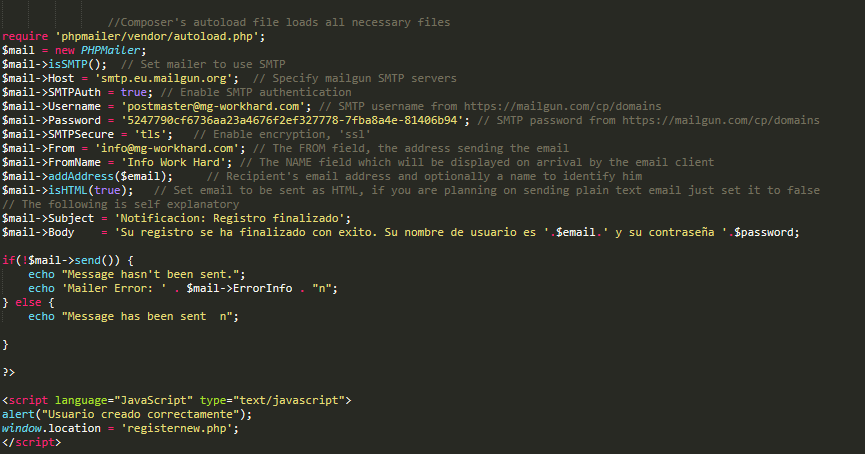
También está integrado un validador de DNI para que se use un dni correcto y no sea inventado. De esta forma nos aseguramos que sean datos totalmente correctos y válidos.

Una vez un usuario esté dado de alta, ese email y DNI no podrán repetirse, de esta forma evitamos duplicar datos que consideramos esenciales para cada usuario.



código validación dni y contraseña

Una vez el usuario esté dado de alta recibirá un correo con su usuario y contraseña, cuando se ha dado de alta aparece un alert de javascript avisando que se ha creado el usuario correctamente, si ocurre algo aparecerán alerts diciendo si el problema es un DNI erróneo o ya ha sido dado de alta o el email ya se está usando:



envío email y alert

Una vez esté dado de alta el usuario podrá ver las clases que hay disponible esa semana y reservar la que le interese, el usuario también podrá ver todas las clases que ha reservado y eliminar alguna si no desea participar.

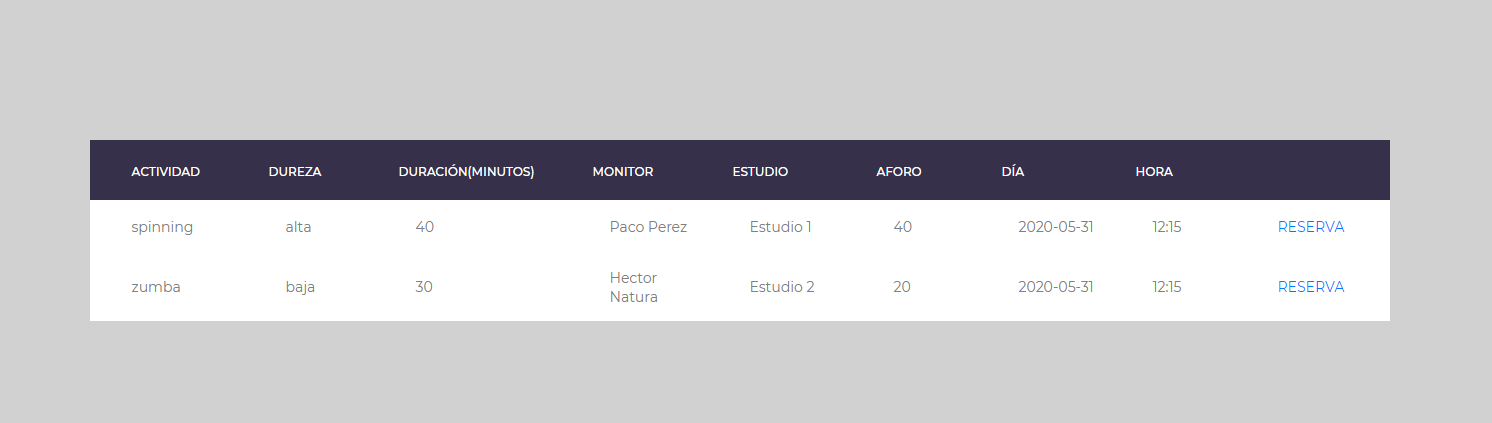


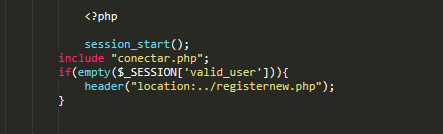
imagen reserva actividades



reservas hechas por el usuarios

El usuario no logueado tendrá impedimento para acceder a las reservas, necesitará loguearse o darse de alta si no lo ha hecho.

Con este código se revisa si hay valor en la variable de sesión “valid\_user”, si está vacía se redirige a la página para login/registro.



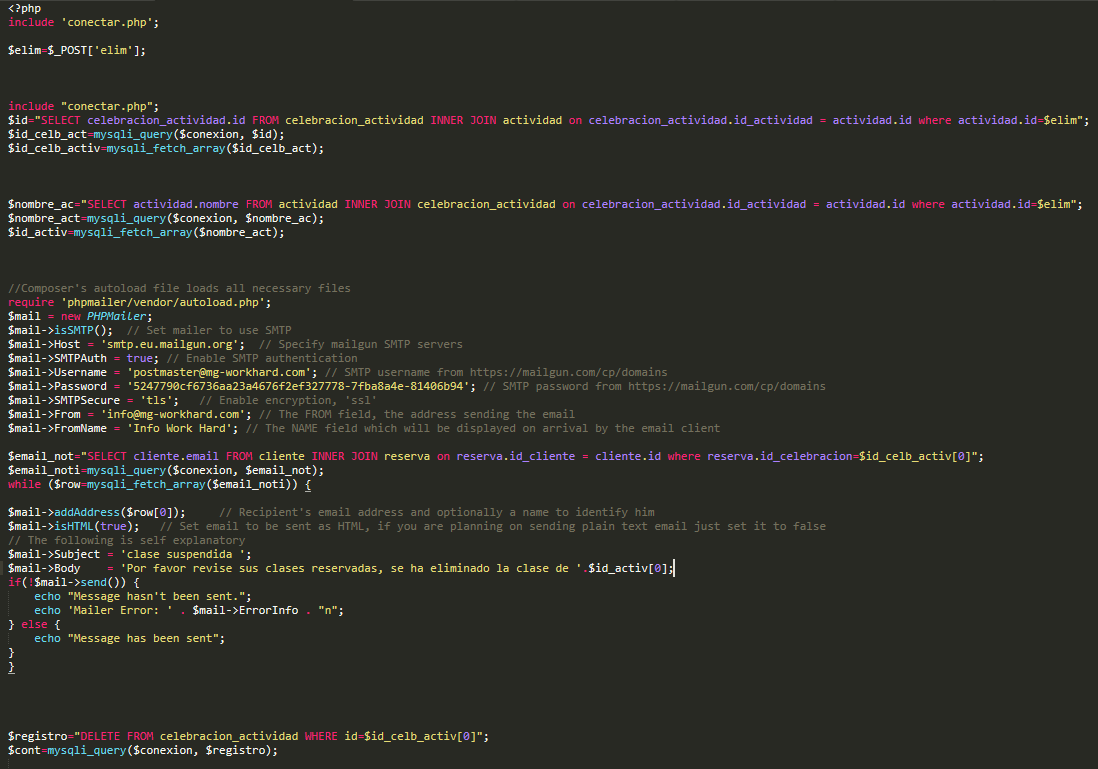
Código comprobar login

En la parte de administrador, los técnicos entraran con su cuenta de correo y su contraseña.

Una vez accedan verán la página donde podrán crear actividades nuevas, eliminar actividades, eliminar clases concretas, cambiar el monitor de una clase, eliminar algún monitor y crear uno nuevo, …

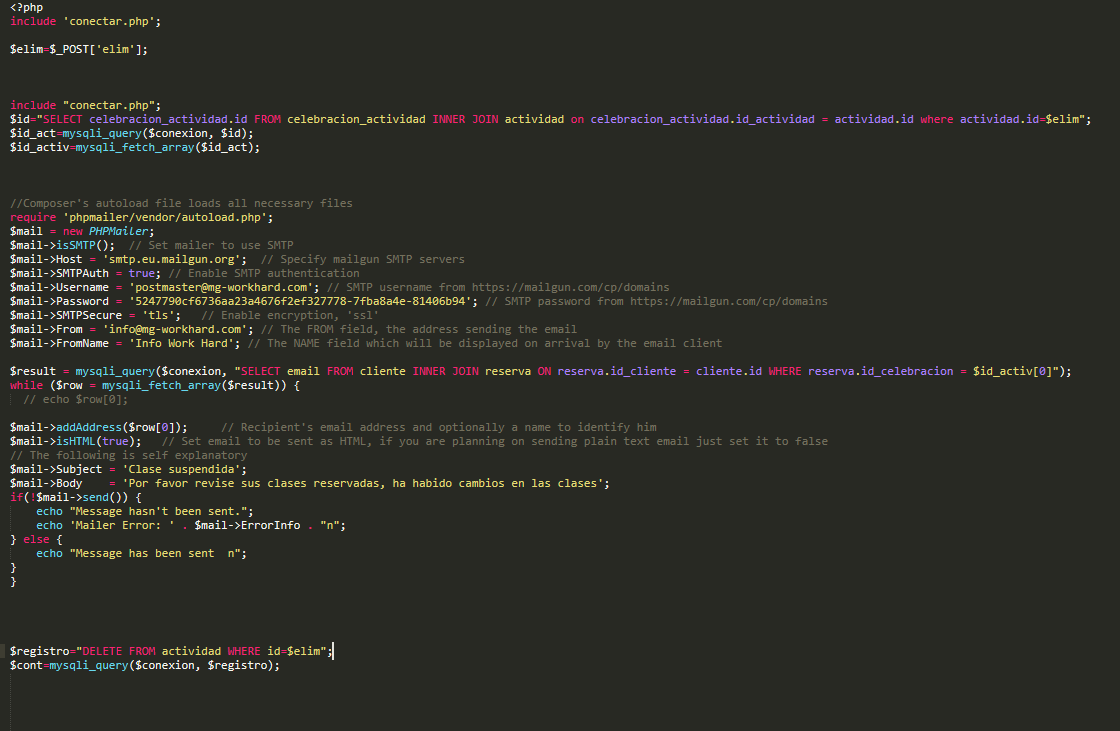
Siempre que las clases se vean afectadas y por tanto esto afecte a las reservas, todos los clientes que han reservado la clase reciben un correo notificando que la clase ha sido suspendida.

Vemos el código tanto al eliminar una actividad como al eliminar una clase:



clase eliminada

En la captura anterior vemos el código para eliminar una clase, aquí hacemos un bucle para recorrer todas las reservas y enviar el correo a todos los clientes que han reservado la clase que acaba de eliminarse.



actividad eliminada

En la captura anterior vemos el código para eliminar una actividad, al eliminarse la actividad se elimina por cascada las clases creadas y las reservas, por lo tanto, se hace un bucle para enviar el correo a toda la gente que ha reservada la actividad.

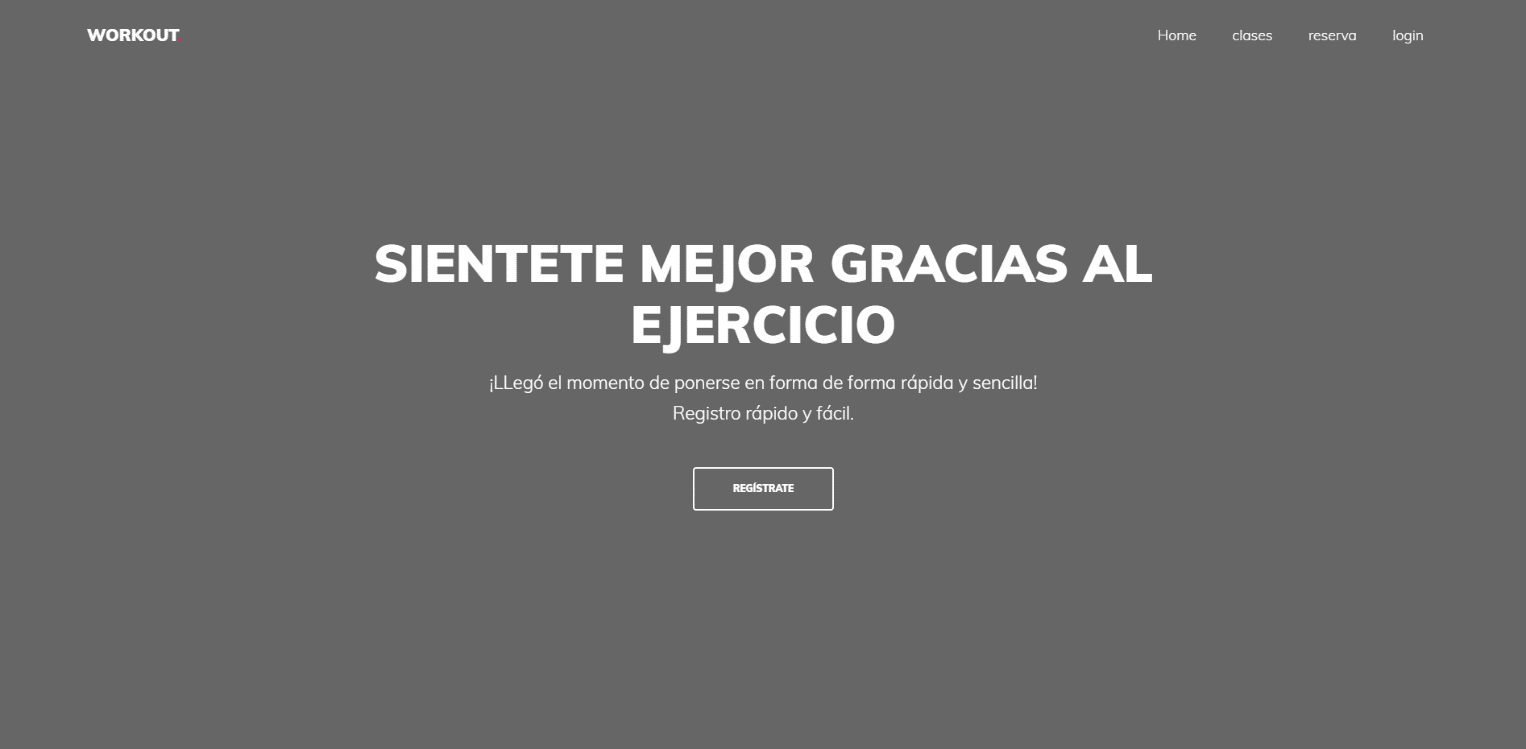
*Manual del usuario*

el usuario se encontrará con una pagina donde puede ver el nombre del club, se cuenta un poco sobre la ética de trabajo del gimnasio, un botón que redirecciona a la pagina de login y registro. Si baja verá diferentes recomendaciones y consejos que obtendrá en el club de la mano de los técnicos.

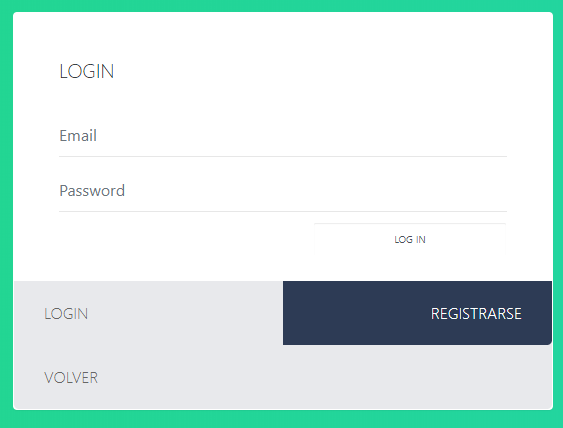
Verá un apartado con las actividades que se dan en el gimnasio y una breve introducción para que el usuario conozca de qué se trata.

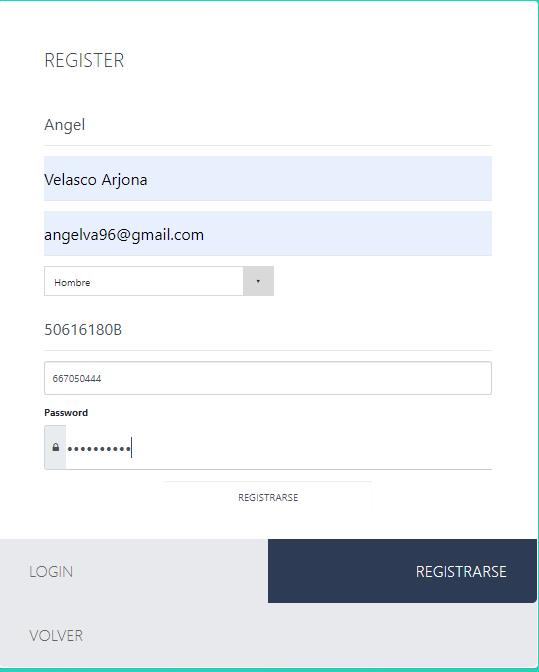
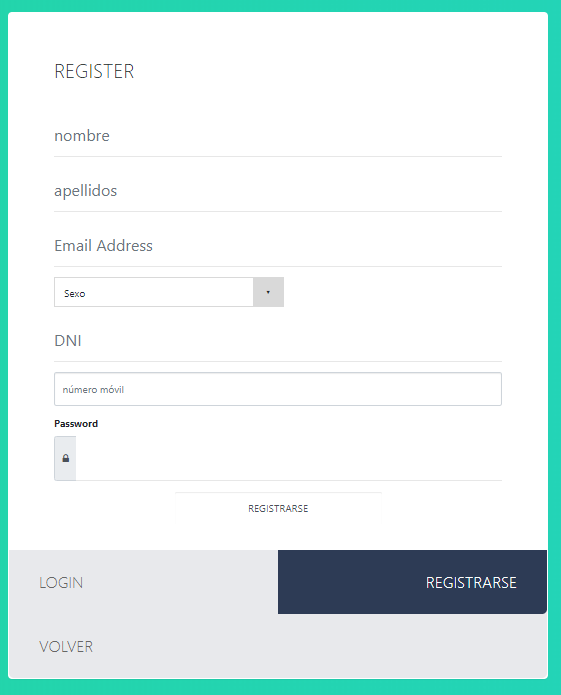
En estas recomendaciones el cliente ve un botón que le llevará a la página de reserva de las actividades.

En la parte superior tendrá una serie de botones que le permitirá moverse rápidamente a las clases, a la reserva y al login

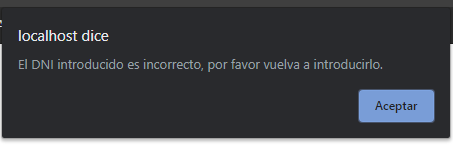


El usuario se registrará completando los campos correspondientes:

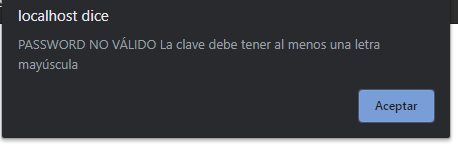




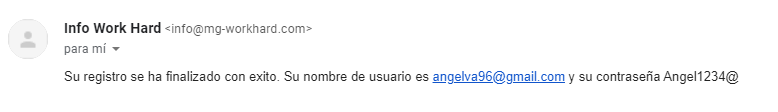
Si pone mal el DNI aparece este mensaje:



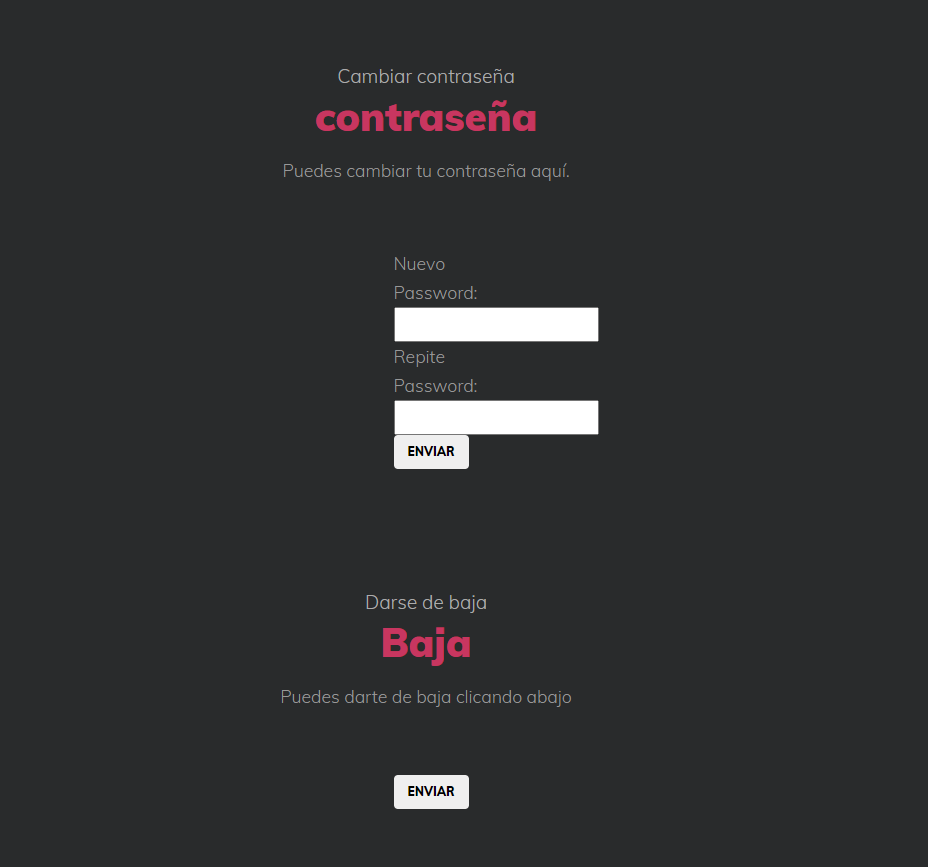
Si la contraseña no cumple los requisitos aparecerá esto:



Una vez complete el registro recibirá un correo con su usuario y contraseña:



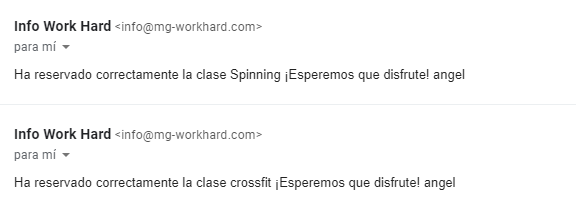
un vez dentro como usuario verá la misma página, pero en la barra superior tendrá un botón de Ajustes que sirve para cambiar su contraseña o para darse de baja.



Si el usuario va a la opción “reservas” verá las clases disponibles a las que puede apuntarse si así lo desea, verá el nombre de la actividad, la dureza de esta, quién realiza la clase, fecha y hora , …



Cuando reserva una clase recibe un email con la verificación:

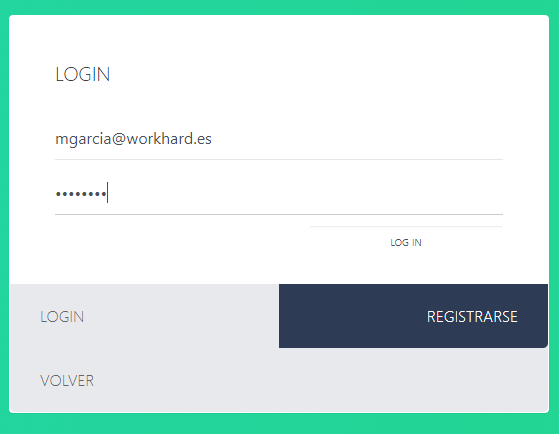


El usuario puede ver las reservas que tiene pulsando en “mis reservas” donde puede eliminar dicha reserva si así lo desea, se verá el nombre y apellido del usuario, la clase reservada



Desde la parte del administrador vemos lo siguiente:

al loguearse un técnico:

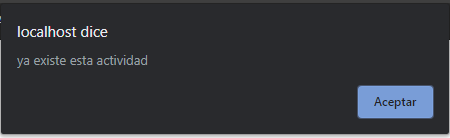


puede añadirse una actividad nueva:

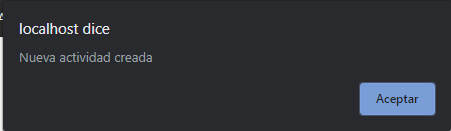




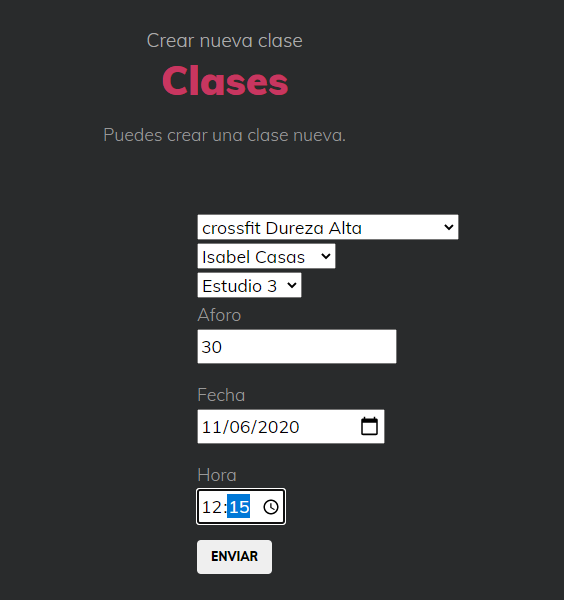
Si la actividad ya existe aparecerá este mensaje:



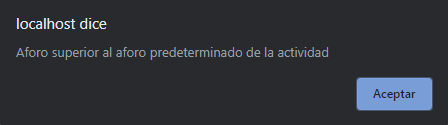
Y si no existe se creará sin problemas:



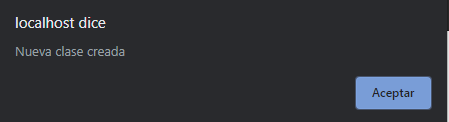
puede crear una clase nueva:



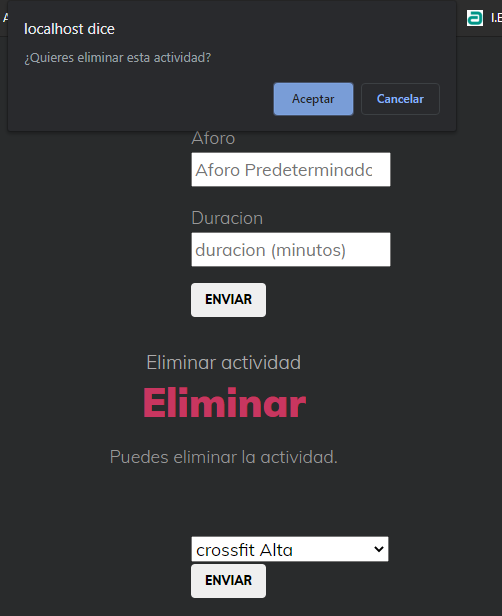
Si el aforo que ponemos cuando creemos la clase es superior al que pusimos al crear la actividad saltará este aviso:

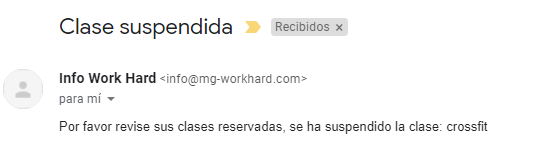


Si todo está correcto:



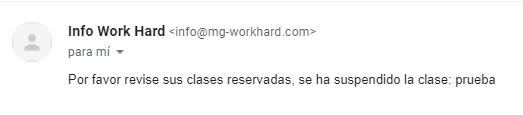
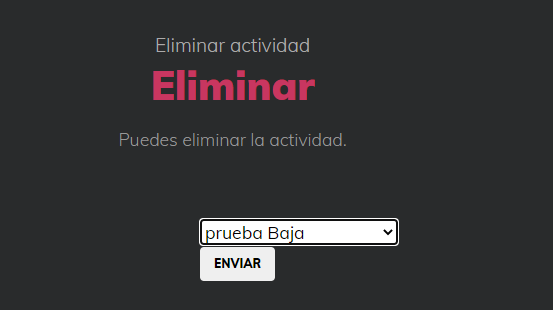
Eliminar una actividad, lo que lleva a que se elimine las clases creadas y por lo tanto se eliminan las reservas que existan de esa actividad, al hacerlo se envía un correo que llega a todos los usuarios que han reservado:



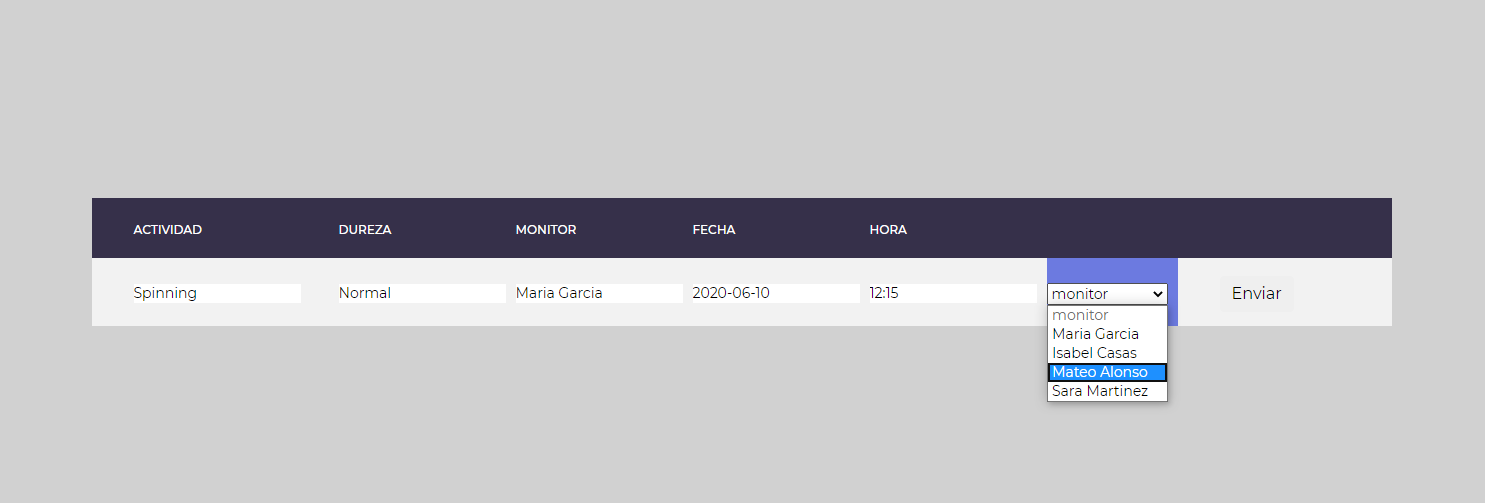


Puede eliminar una clase, ocurre lo mismo que lo anterior:

se envía un correo a todos los usuarios que han reservado esa clase:



Los técnicos podrán cambiar el técnico que realice una actividad por otro si es necesario o si el técnico ya no trabajase más en el gimnasio:





Los técnicos pueden añadir nuevos técnicos creando su correo, se comparan los correos ya existentes para que no se creen correos iguales que dará lugar a problemas, la contraseña se somete al mismo proceso de comprobación que hemos visto anteriormente:

