|  |
| --- |
| DEPARTAMENTO DE INFORMÁTICA  Ciclo Formativo: DESARROLLO DE APLICACIONES WEB Curso: 2  Módulo: DISEÑO DE INTERFACE WEB |

|  |  |  |  |
| --- | --- | --- | --- |
| **PR0304** | **Proyecto entrenamiento** | | **2º**  **DAW** |
| **ALUMNO/S:** | |
| **FECHA:** | **NOMBRE EQUIPO:** |

Objetivo

El objetivo es la creación de una **aplicación web para el seguimiento de actividades deportivas**. Este tipo de aplicaciones se ha popularizado enormemente en los últimos años aprovechando la disponibilidad de múltiples dispositivos para el trazado de rutas, bien mediante smartphones con GPS o bien mediante relojes diseñados específicamente para el seguimiento de rutas geoposicionadas.

Ejemplos de estas aplicaciones son *Strava*, *Endomondo*, *Runkeeper*, *Runtastic* o *Sports Tracker* y, aunque cada una tiene sus propias particularidades, todas comparten un funcionamiento general común.

Algunas de las características compartidas de que disponen, o lo que podíamos definir como la funcionalidad mínima, son:

* El usuario se registra en la página y dispone de un espacio personal donde almacenar todos sus entrenamientos.
* Los dispositivos suelen utilizar el formato GPX para almacenar los *tracks* o rutas, por lo que las páginas permiten subir los entrenamientos mediante estos ficheros.
* El usuario puede ver cronológicamente todos los entrenamientos que ha subido.
* También puede acceder en detalle cada entrenamiento con información como distancia, tiempo empleado, velocidad media,… Junto a esta información se muestra un mapa con la ruta.

## Desarrollo del proyecto

El sitio Web que vamos a diseñar tiene que tener las siguientes páginas:

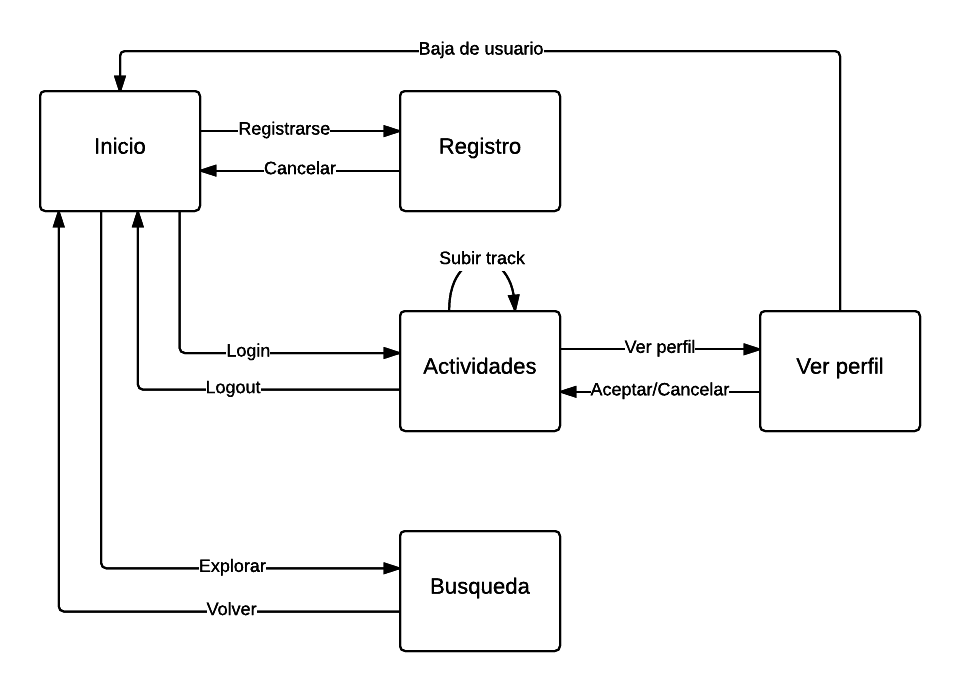
* Inicio
* Registro
* Visualización de actividades
* Búsqueda anónima de actividades
* Edición de perfil

### Aspectos comunes a todo el sitio

#### Mapa de navegación

En el siguiente esquema se muestran las diferentes páginas que componen el proyecto así como los flujos de navegación entre ellas.

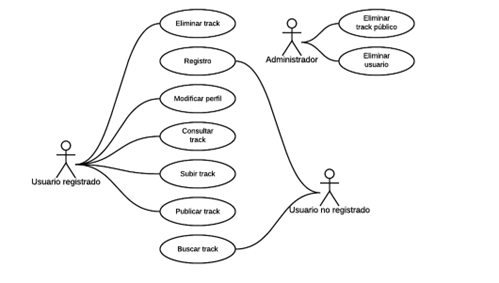
Este mapa es orientativo y el alumno podrá realizar las mejoras o cambios que considere oportunas, siempre y cuando respecte la funcionalidad general de la aplicación.



#### Funcionalidad offline

Vamos a implementar la posibilidad de utilizar esta página offline (por lo menos en una funcionalidad básica). Para ello almacenaremos los entrenamientos que descarguemos en el *LocalStorage* como veremos más adelante y además crearemos un **fichero de manifiesto** que permita guardar en la caché del navegador las páginas.

#### Casos de uso



### Página de inicio

Es la página de bienvenida al sitio y por tanto debe tener un fuerte impacto visual y capacidad de atraer al usuario. Algunas características que debe de tener:

* En cuanto a funcionalidad solo debe de tener la opción de **Registrarse** y de **Iniciar sesión**. La primera opción debe llevar a la página de registro mientras que la segunda debe dar la posibilidad al usuario de introducir su nombre y contraseña. Ten en cuenta que estos datos se pueden pedir en otra página, mediante un *popup* o directamente en la página.
* En el cuadro de inicio de sesión debe haber una opción que permita recordar los datos de usuario para el próximo inicio de sesión. En caso afirmativo los datos se guardarán en el *LocalStorage*.
* También debe permitir el acceso anónimo a la aplicación llevando este acceso a la búsqueda anónima de actividades.
* El resto del contenido de la página es a tu criterio. Puede incluir elementos como:
  + Resumen de funcionalidades de la aplicación.
  + Capturas de la página interna de la web
  + El típico mensaje de usuario contento con la aplicación.
* Debe disponer de acceso web a la documentación y al código de la aplicación.

### Página de registro

En la página de registro se solicitarán los datos del usuario. Se le tienen que pedir los siguientes datos:

* Nombre y apellidos
* Nombre de usuario
* Correo electrónico
* Contraseña
* Estatura
* Peso
* Fecha de nacimiento
* Zonas de frecuencia cardíaca. Hay 5 zonas que podemos llamar *sprint*, *velocidad*, *cardio*, *fitness* y *sencillo*. La idea es que el usuario introduzca su rango de pulsaciones para cada una de estas zonas.
* Tipo de corredor. Aquí habrá varios tipos y según el tipo que elija el usuario se le preguntarán unas cosas y otras:
  + Corredor casual
  + Corredor de medio fondo: se le preguntará por su mejor marca en 5 km, 10 km y media maratón.
  + Corredor de fondo: se le pedirá mejor marca de media maratón y maratón.
  + Corredor de trail: se le pedirá carrera más larga que ha corrido con nombre, distancia, desnivel y tiempo empleado.

En este formulario se realizarán las siguientes acciones:

* En todo el formulario se realizarán las validaciones típicas, que el nombre no tenga dígitos ni caracteres extraños, que el email sea válido, estatura y peso dentro de unos límites,…
* Nombre de usuario: debe ser único, para ello se realizará una solicitud AJAX contra el servidor que nos indicará si ese usuario ya existe o no.
* Los datos de frecuencia cardiaca se precargarán automáticamente con una estimación basada en la edad del usuario. Para ello se tomará el valor de *220 - edad* como frecuencia cardiaca máxima, a partir de este valor se obtendrán las zonas tal tomando como referencia los valores indicados en la página <https://goo.gl/2OncY7>.

### Página de búsqueda/acceso anónimo

Esta página permitirá a usuarios no registrados realizar búsquedas sobre los entrenamientos de otros usuarios.

La estructura de esta página será muy similar a la de visualización de actividades, pero en lugar de tener un calendario tendrá un formulario con diferentes opciones de búsqueda (p.e. usuario, fecha, distancia,…). Los resultados de la búsqueda se mostrarán en una lista y hacer *click* en un elemento de esta lista realizará una solicitud Ajax al servidor para obtener el entrenamiento de forma análoga a como haremos en la página de visualización de actividades.

### Página de perfil

A esta página accederá el usuario desde la página de actividades, una vez que haya iniciado sesión. En ella se le mostrará la información introducida al registrarse y tendrá la posibilidad de **modificar los datos**.

Igualmente desde aquí deberá poder darse de baja, lo que implicará eliminar todos los datos relativos al usuario del servidor.

### Página de visualización de actividades

Esta es la página más importante de nuestro sitio. En ella el usuario podrá monitorizar todos los entrenamientos que haya subido al servidor de forma cronológica así como ver información detallada de cada entrenamiento.

La página se divide en dos partes:

* Entrenamientos ordenados cronológicamente
* Información detallada de cada entrenamiento

La idea es que cuando el usuario haga click en un entrenamiento se realice una solicitud Ajax que cargue la información detallada de dicho entrenamiento.

#### Parte 1: entrenamientos

En esta parte se tienen que ver los entrenamientos ordenados cronológicamente. Lo ideal es añadir algún mecanismo para ver los entrenamientos de meses pasados.

#### Parte 2: Información de cada entrenamiento

De cada entrenamiento se tiene que ver lo siguiente:

* Botones para compartirlo en redes sociales. Por ejemplo Twitter y Facebook.
* Información general del mismo: distancia, duración, ritmo medio (min/km), ritmo máximo (min/km), calorías, altitud máxima y mínima, desnivel, frecuencia cardiaca media y máxima.
* Mapa con la ruta.
* Gráfica con la velocidad media en cada kilómetro. Hay muchas formas de hacer esto y puedes elegir la que quieras. En las siguientes imágenes puedes ver algunos ejemplos de cómo se muestra en diversas páginas.

|  |  |
| --- | --- |
|  |  |
|  |

Esta página se pretende que sea interactiva, por lo que será necesario que utilices tecnologías de todas las asignaturas que has visto en el curso.

El objetivo es que cuando el usuario seleccione un entrenamiento se muestren sus datos sin necesidad de recargar la página. Indudablemente para esto necesitarás Ajax.

El proceso a seguir es el siguiente:

1. El usuario hace *click* en un entrenamiento.
2. Desde JavaScript se intercepta ese *click* y se construye una solicitud Ajax que pide al servidor los datos de ese entrenamiento1.
3. El servidor localiza el fichero GPX que corresponde al entrenamiento.
4. El servidor analiza el fichero GPX para extraer los datos que debe devolver2.
5. El servidor construye un JSON con los datos que ha generado. Este fichero tendrá más o menos la siguiente estructura:



Se puede ver una simulación de la respuesta en la dirección apirest.victorjgonzalez.com/training/544 (el número final simularía el código de entrenamiento. En este ejemplo se puede poner cualquiera porque siempre devuelve lo mismo). Aunque se puede poner directamente la URL en el navegador es aconsejable utilizar alguna herramienta específica para probar APIs REST. A mí personalmente me gusta la extensión de Chrome **Restlet Client – DHC3**.

1. El cliente recibe el JSON por lo que ahora debe construir la parte de la página que corresponde al detalle del entrenamiento a partir de los datos obtenidos. Dentro del campo *info* del JSON los datos genéricos del entrenamiento y dentro del campo *vueltas* los datos para dibujar el diagrama en el *canvas* con el perfil de entrenamiento.
2. Finalmente guardaremos los datos del entrenamiento en el **LocalStorage** para que estén disponibles en caso de que no haya conectividad. Solamente se guardarán la información del campo *info* del JSON.
3. ¿Qué pasa cuando no haya conectividad? En ese caso la llamada Ajax nos devolverá error y será el momento de acceder al LocalStorage para buscar a ver si los datos del entrenamiento están guardados ahí. En caso afirmativo se utilizarán para mostrar los datos en pantalla. Si no estuvieran ya se mostrará un erro indicando que no se pueden obtener los datos.