

Informe 2da entrega del trabajo práctico “72.36 -

Interacción Hombre Computadora (HCI)”



*De Simone, Franco - 61100
Dizenhaus, Manuel - 61101
Cornidez, Milagros - 61432
Abancens, Alberto - 62581*

Índice

Introducción	3
1) Requisitos funcionales y no funcionales implementados	4
1.1) Requisitos funcionales implementados	4
1.2) Requisitos no funcionales implementados	4
2) Capturas de pantalla del sitio web	5
3) Decisiones de usabilidad tomadas durante la etapa de implementación	12
3.1) Diferencias con prototipos	12
3.2) Temas vistos en la materia	15
3.3) Modelos de persona	16
3.4) Sugerencias/modificaciones realizadas por la cátedra	16
3.5) Diseño gráfico	17
4) Archivos necesarios para el correcto funcionamiento del sitio	18
5) Instructivo de instalación	18

Introducción

Para esta segunda entrega del trabajo práctico, tomamos los prototipos diseñados en la primera entrega y, junto a los nuevos conocimientos adquiridos durante las clases, adecuamos los mismos a un diseño de una página web funcional y responsive.

Esto se presentó como todo un desafío para el grupo, dado que ningún integrante había tenido experiencia previa con sitios web, ya sea lo que concierne el front end con html y css como con el back end, que abarcó el uso de javascript y llamado a funciones de una API otorgada por la cátedra.

Sin embargo, intentando replicar y aplicar los contenidos desarrollados en clase, sumados al estudio en profundidad de los ejemplos provistos, se logró construir un sitio web que realiza los requerimientos pedidos de manera efectiva.

1) Requisitos funcionales y no funcionales implementados

1.1) Requisitos funcionales implementados

- Registrarse e ingresar mediante un usuario y contraseña.
- Ver y editar el perfil del usuario activo.
- Crear, borrar, y editar ejercicios pertenecientes al usuario activo.
- Ver los ejercicios creados por el usuario activo.
- Crear, borrar, y editar rutinas pertenecientes al usuario activo.
- Ver las rutinas creadas por cualquier usuario

1.2) Requisitos no funcionales implementados

- Utilizar las tecnologías HTML, CSS y JavaScript y Ajax.
- Visualizarse correctamente en por lo menos dos de los siguientes navegadores: Microsoft Edge 40 (o superior), Mozilla Firefox 68 (o superior) y Google Chrome 81 (o superior).
- Visualizarse correctamente en resoluciones de pantalla entre 1280px y 1920px.
- Respetar las estrategias de separación entre la estructura y la presentación.
- Respetar las estrategias de separación entre la estructura y el comportamiento.

2) Capturas de pantalla del sitio web

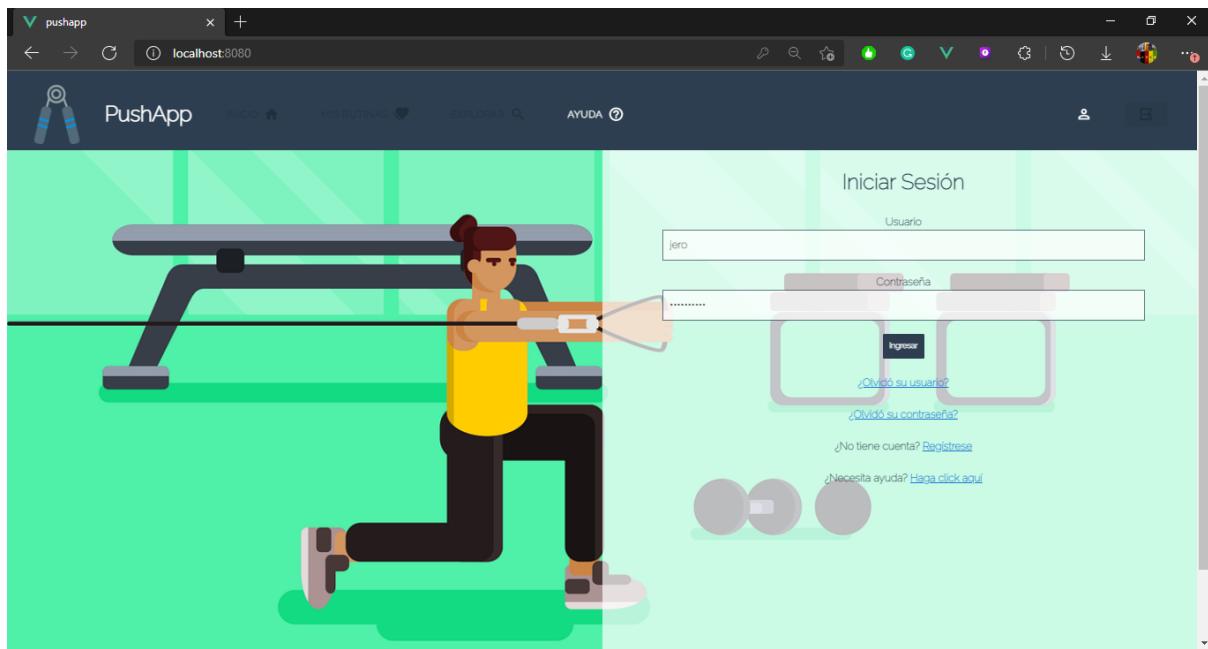


Imagen 1: Vista de “Login”

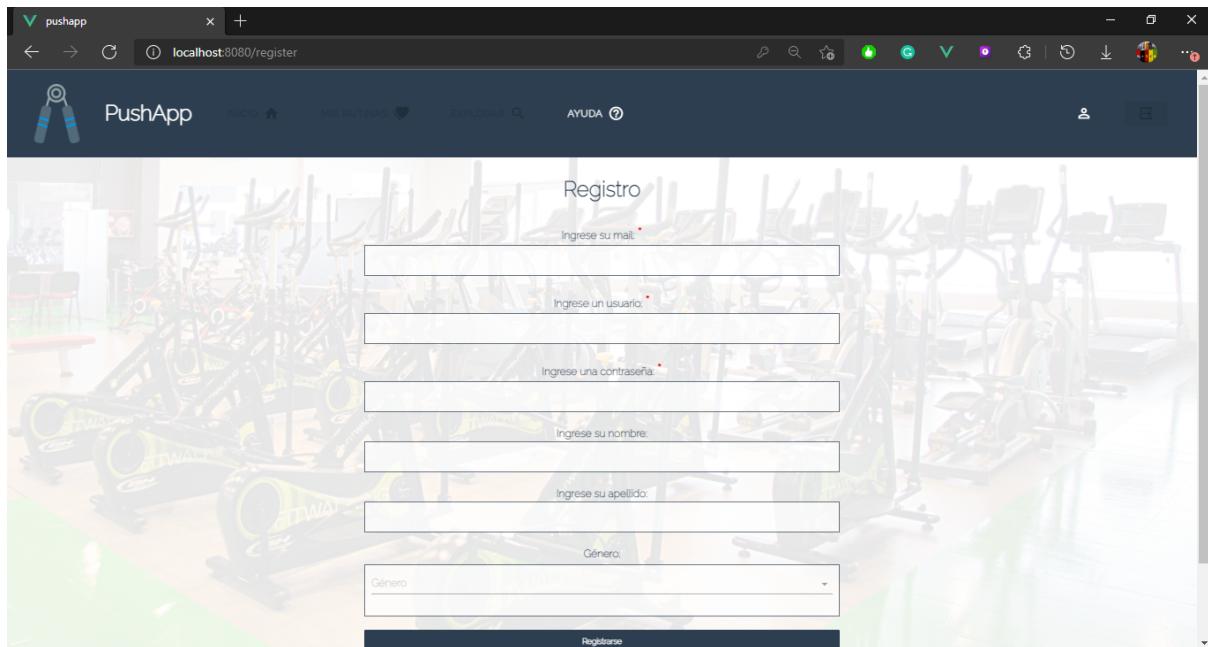


Imagen 2: Vista de “Registro”

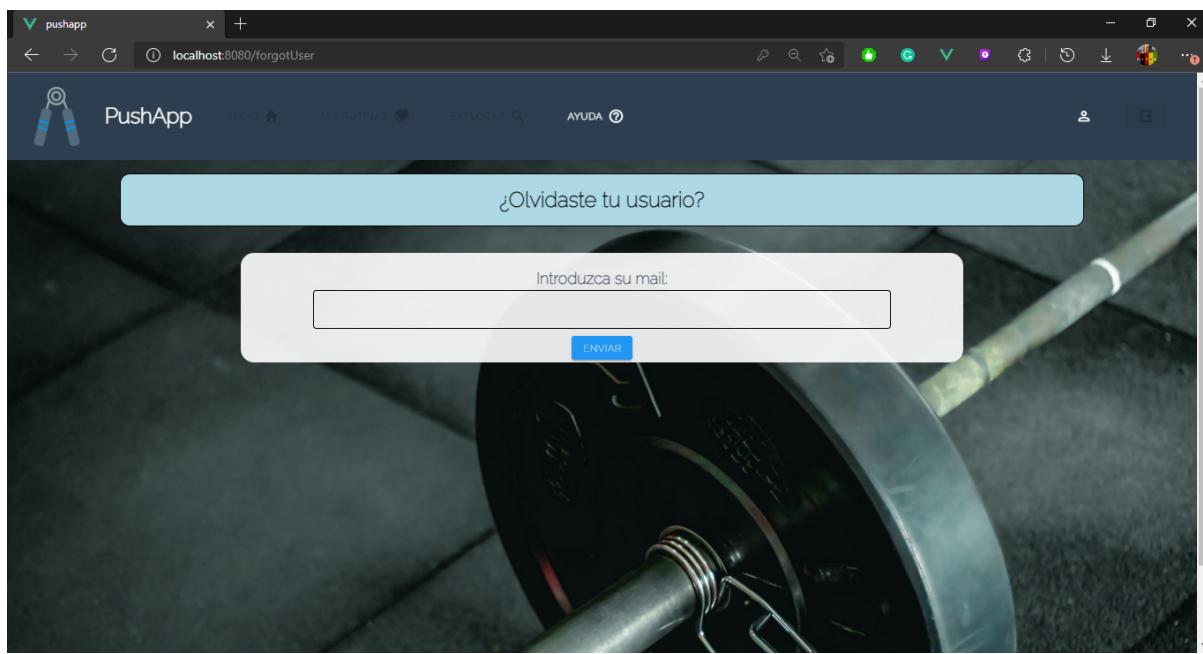


Imagen 3: Vista de “¿Olvidaste tu usuario?”

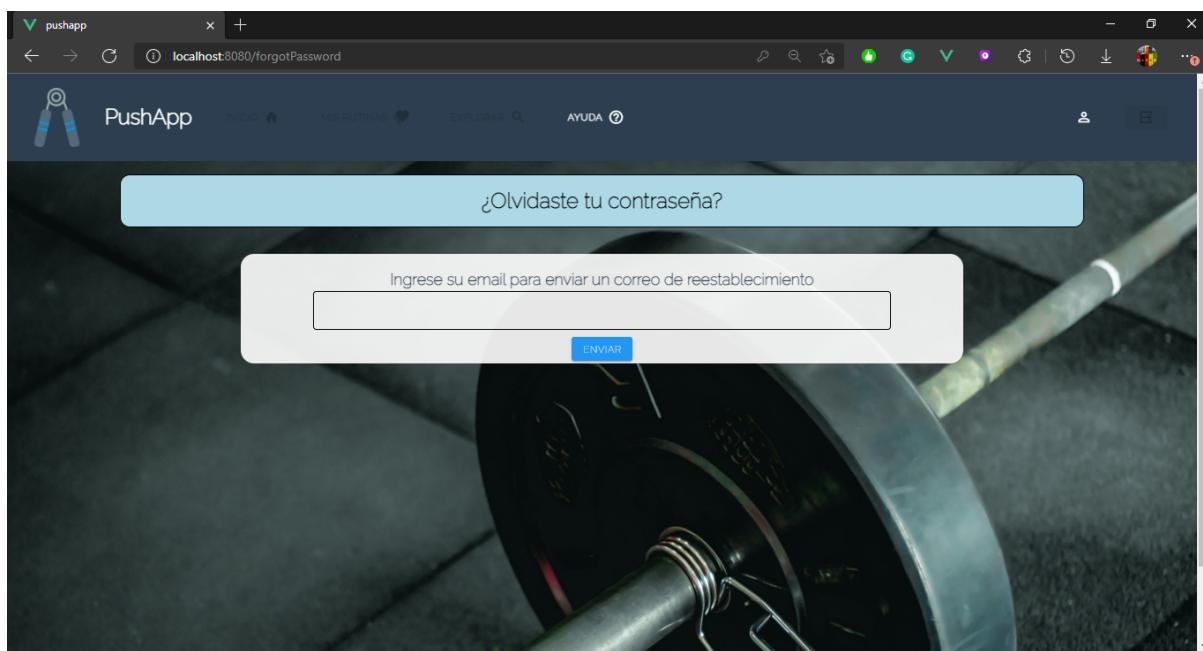


Imagen 4: Vista de “¿Olvidaste tu contraseña?”

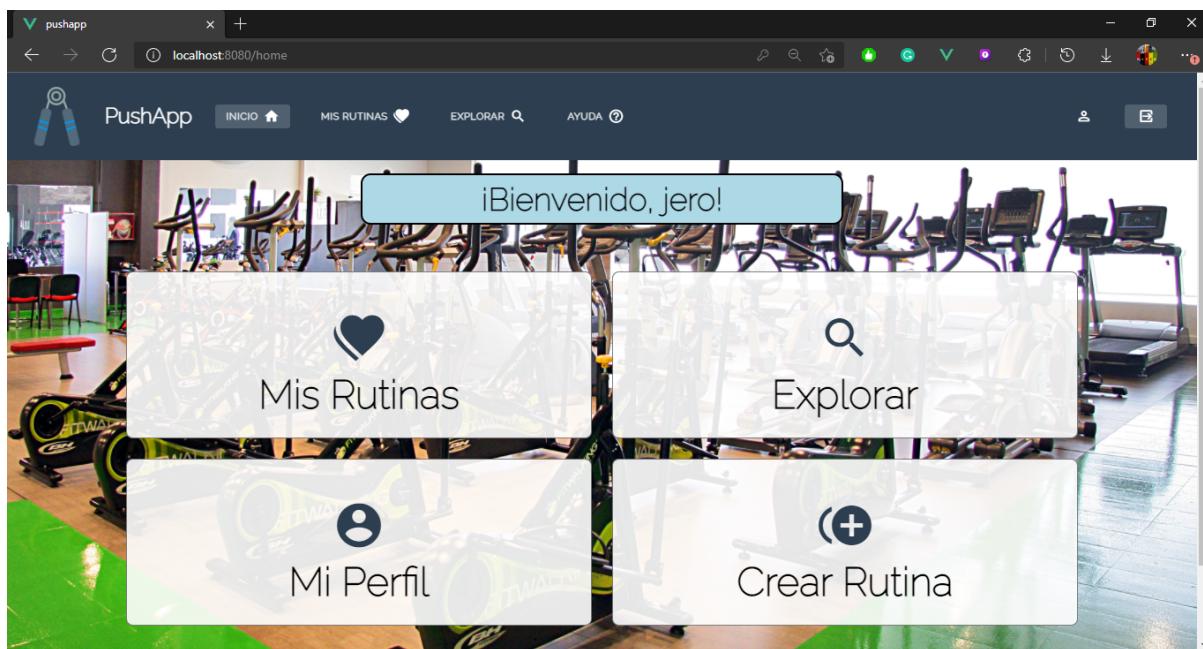


Imagen 5: Página de inicio

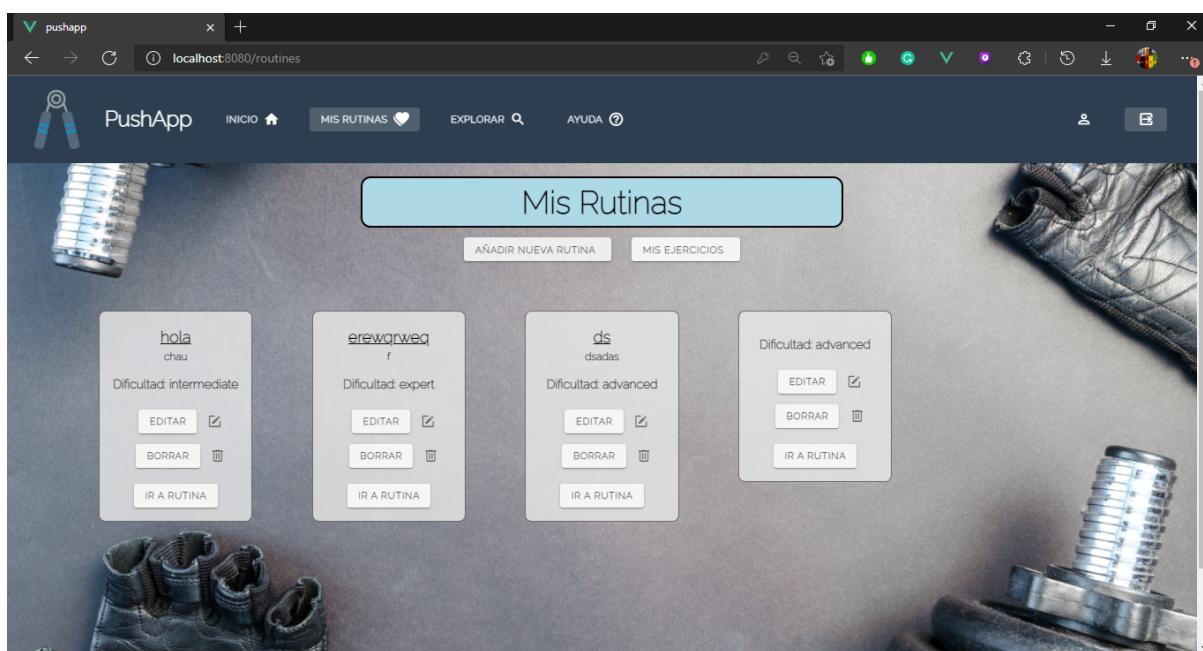


Imagen 6: Página de “Mis Rutinas”

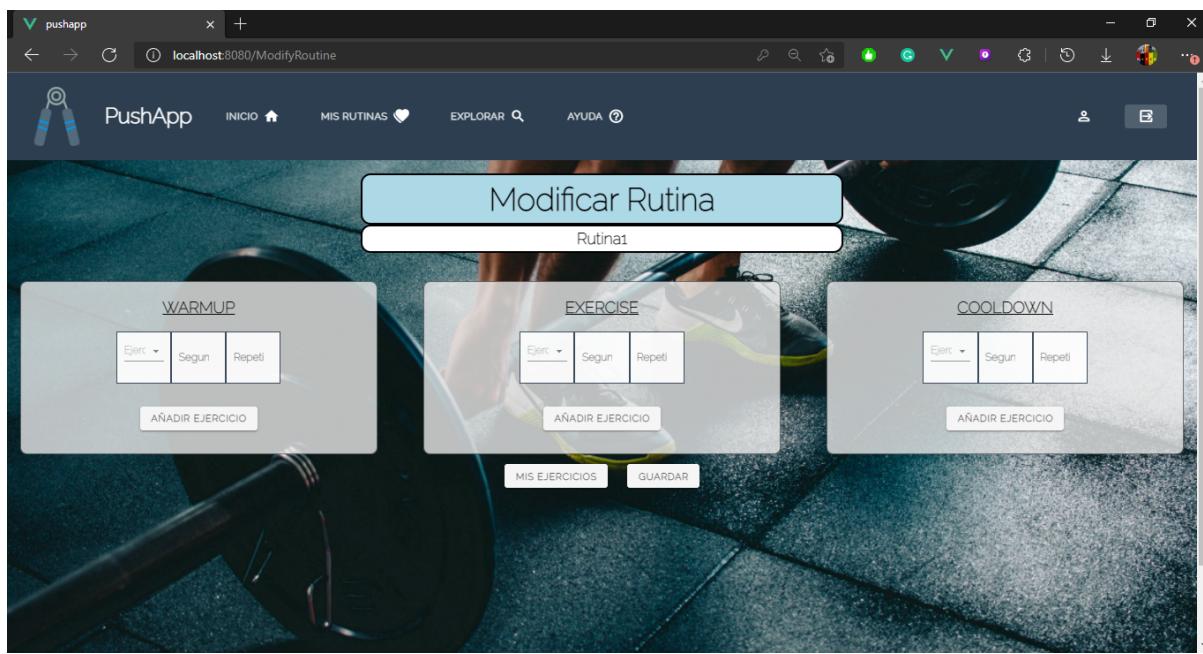


Imagen 7: Vista de edición de rutinas

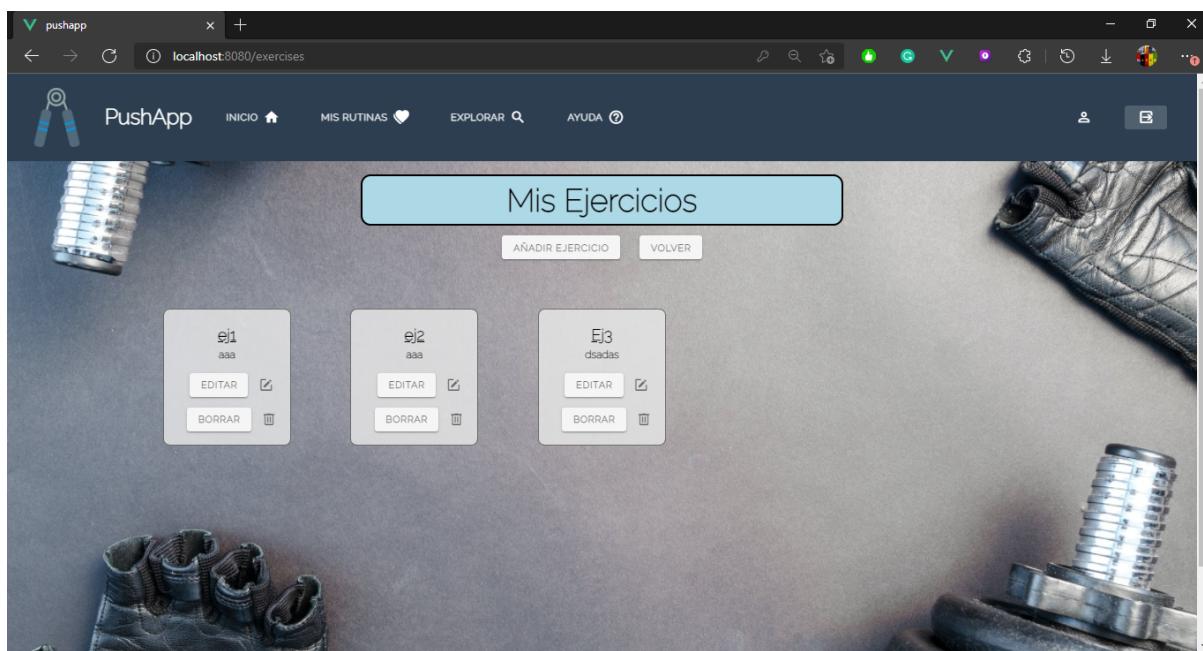


Imagen 8: Vista de “Mis ejercicios”

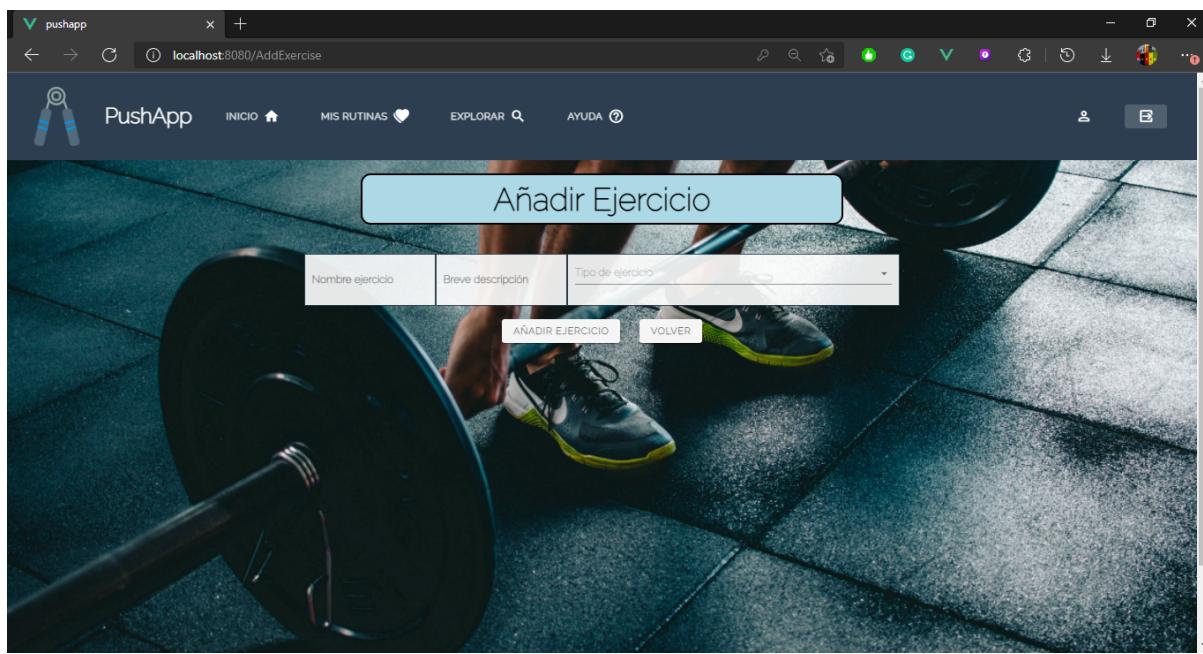


Imagen 9: Vista de “Añadir Ejercicio”

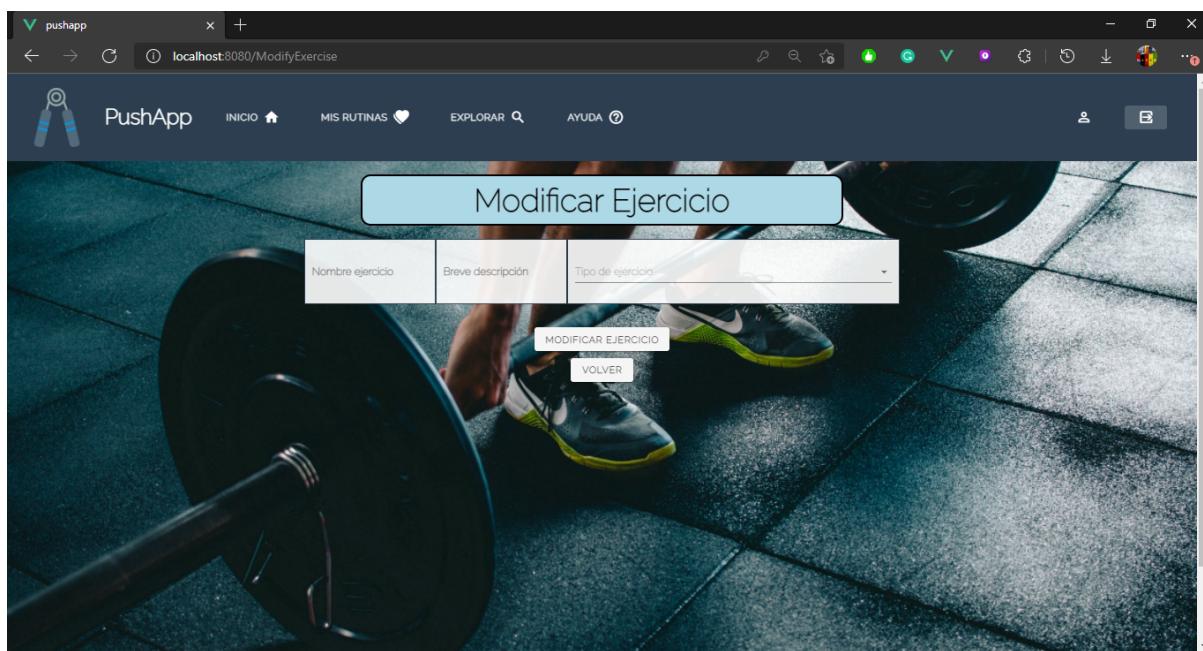


Imagen 10: Vista de edición de ejercicios

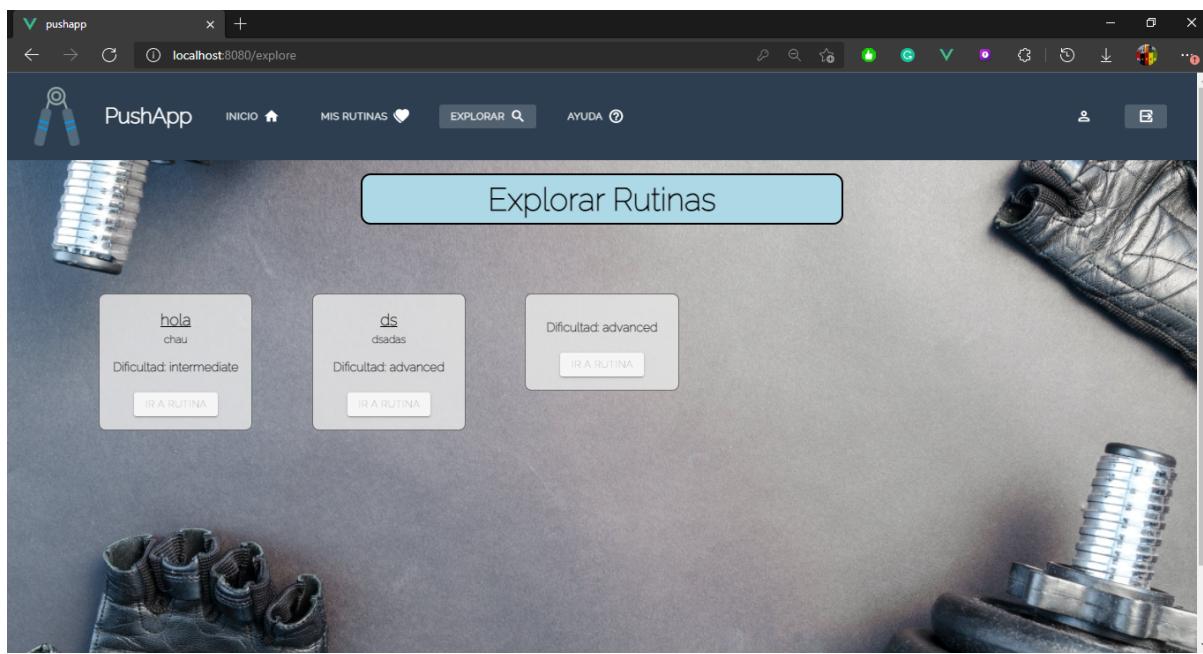


Imagen 11: Página que muestra las rutinas de todos los usuarios (que se encuentren públicas)

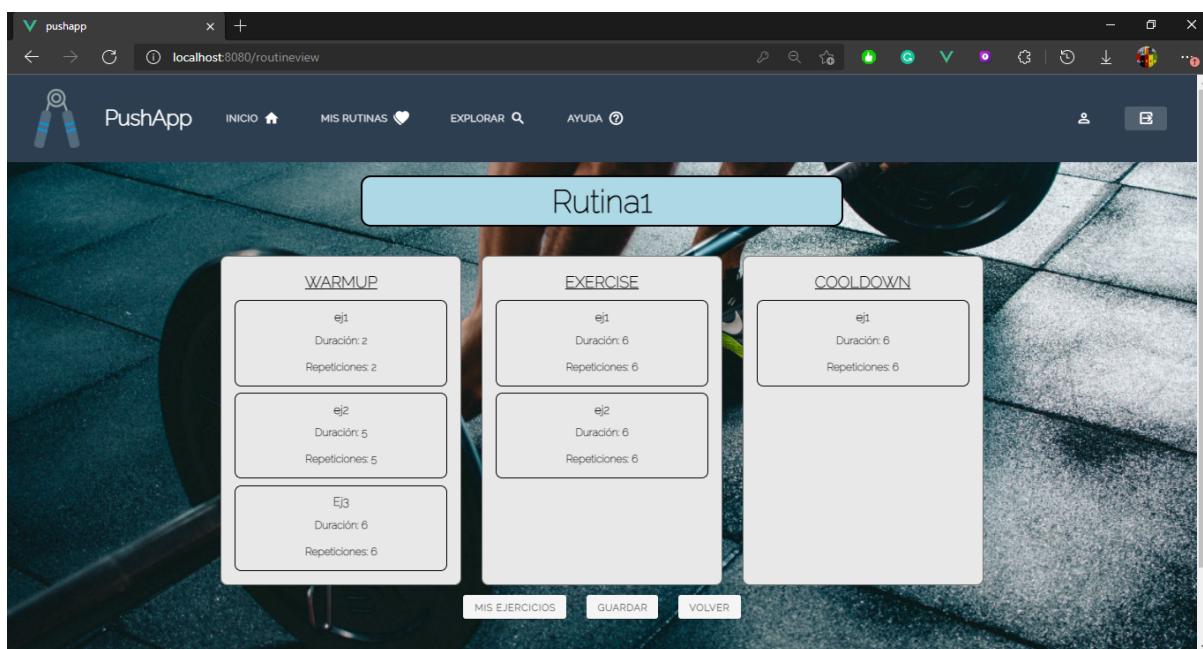


Imagen 12: Vista de rutina

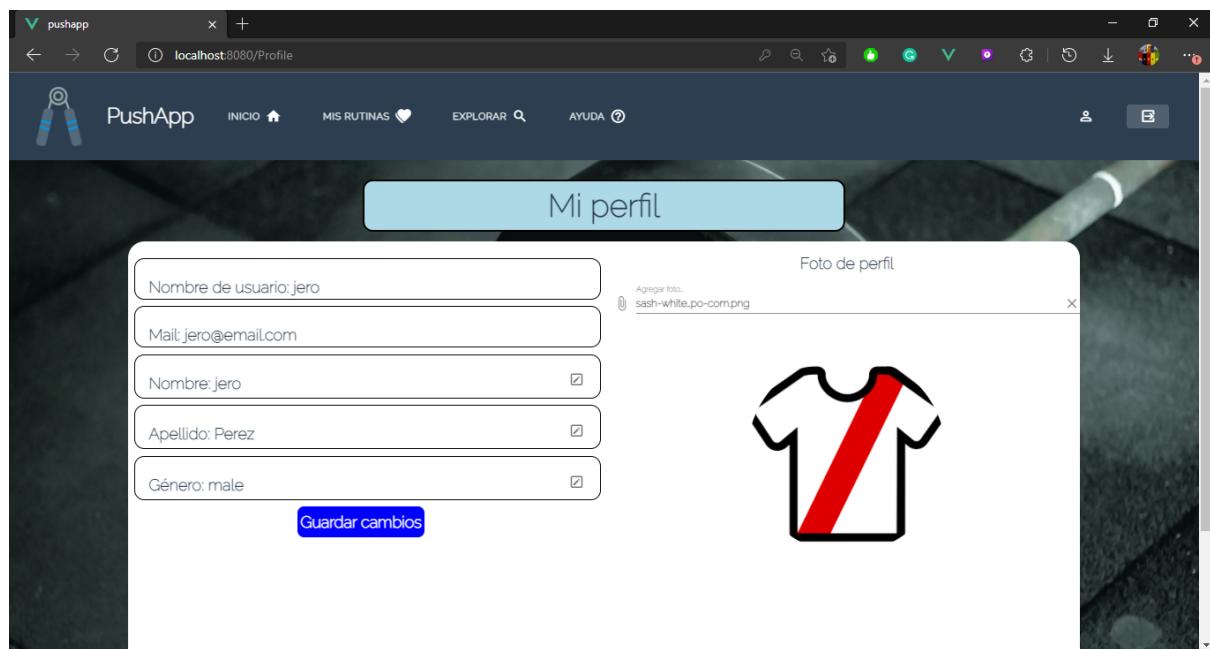


Imagen 13: Vista de “Perfil”

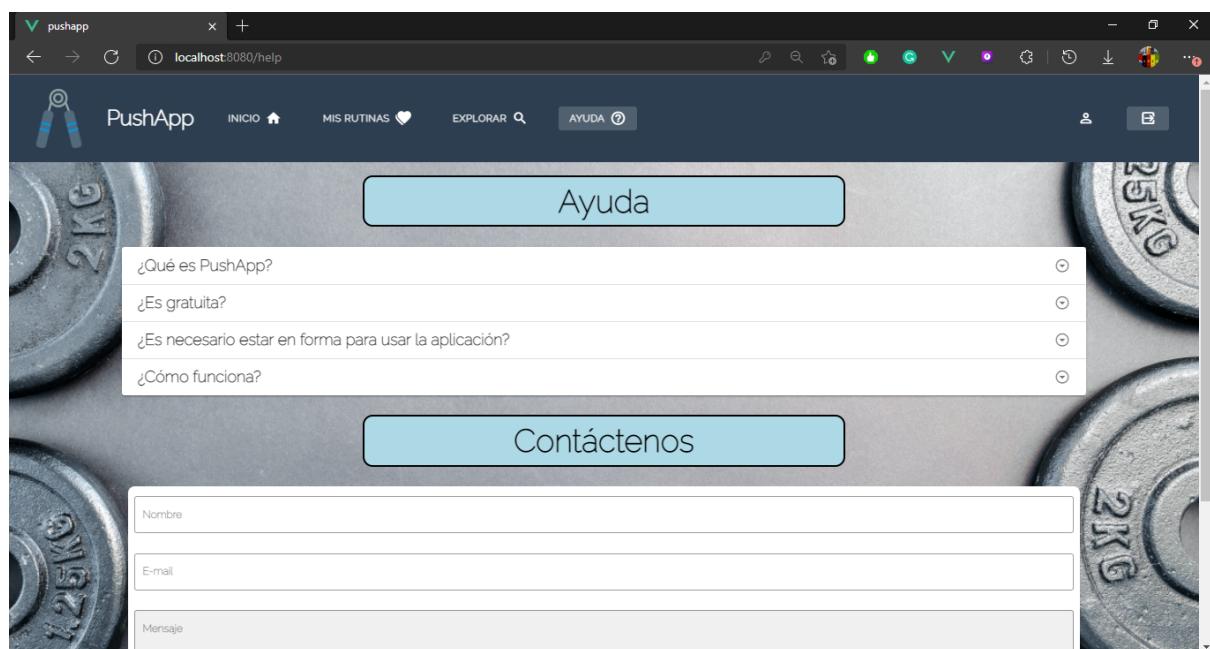


Imagen 14: Página de ayuda

3) Decisiones de usabilidad tomadas durante la etapa de implementación

En cuanto a decisiones de usabilidad, decidimos realizar modificaciones a los prototipos. Algunas fueron elecciones de diseño, ya sea por errores conceptuales realizados desde la concepción de los prototipos mismos, y otras por restricciones de diseño de html/css. Si bien la amplia mayoría de elementos pudieron ser plasmados, algunos quedaron pendientes y/o simplificados dado que requerían un nivel avanzado de construcción de sitios web. Inclusive, durante las consultas de los prototipos, se consultó acerca de la posibilidad de que ciertos elementos del sitio pudieran llegar a no ser llevados a cabo. La conclusión fue que prototipar de manera ideal siempre es preferible aunque luego no se pudieran llevar a cabo todas las implementaciones.

Sin embargo, creemos que el producto final superó nuestras expectativas, inclusive mejorando en muchos casos lo que habíamos diseñado en primera instancia. Por ejemplo, la barra de navegación tuvo varios agregados, como los íconos, el logo, el nombre de la aplicación, y un cambio de tipografía a una más atractiva. Todo esto llevó a una mejoría en cuanto a imagen, dando un reflejo de modernidad y simpleza que creemos no estaba plasmado en los prototipos.

3.1) Diferencias con prototipos

En una gran mayoría de las vistas, se mantuvo la intención original de los prototipos. Por ejemplo, en la vista de inicio, en los prototipos figuraba lo siguiente:



Imagen 15: Prototipo de vista “Inicio”

Se puede observar que se podía seleccionar como una opción “principal” a “Necesito ayuda”. Como equipo, debatimos sobre una posible mejora que tendría, en vez de “necesito ayuda”, tener un acceso rápido a un sitio de mayor relevancia, como lo es mi perfil. El resultado fue el siguiente:

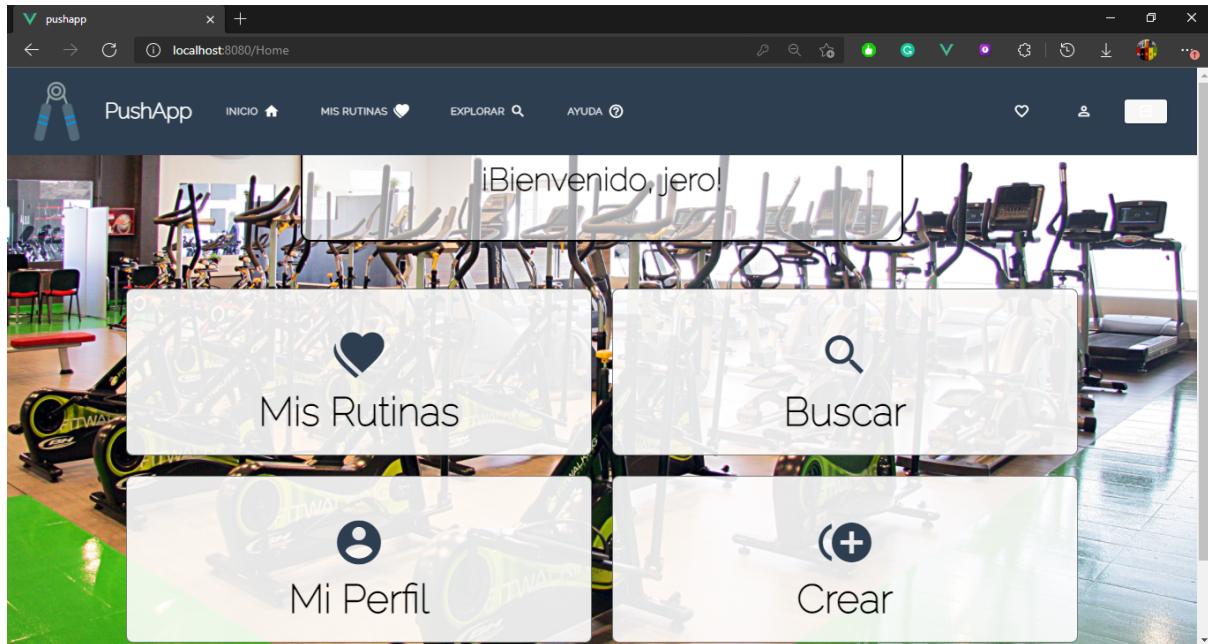


Imagen 16: Implementación de vista “Inicio”

Si bien se trata de un cambio menor, el hecho de tener visible al alcance “Mi Perfil” le permite al usuario, en caso de no lograr interpretar el símbolo estándar de ícono, una manera de acceder a su información.

Como se mencionó previamente, se pueden observar en ambas imágenes las modificaciones de la barra de navegación. En el prototipo, se colocó una palabra en referencia a cada sección, y los dos símbolos pegados. Para la implementación en el sitio web, le realizamos distintos retoques que permitieron darle una imagen moderna y adecuada. Primero, se incluyó el nombre de la aplicación en la misma. Luego, cada sección se encuentra acompañada de un símbolo asociado que le permite hacer la conexión al usuario en caso que no se encuentre familiarizado con la terminología empleada (aunque se intentó de mantener lo más simple posible). También, se incrementó la separación entre los símbolos en la parte derecha (creímos que se veían demasiado pegados) y se implementó un botón para hacer un logout de manera eficaz.

3.2) Temas vistos en la materia

Para justificar las elecciones de usabilidad del sitio web, creemos oportuno realizar la presentación a través de las famosas 8 reglas de oro:

- Consistencia

La intención fue siempre mantener una estética consistente a través de todas las vistas del sitio web, utilizando la misma paleta de colores, tipografía, botones de la colección de material icons oficial, entre otras cosas. Todo esto le da una identidad a la página que el usuario puede reconocer, y eso es un valor intangible para nosotros como desarrolladores.

- Shortcuts

La página cuenta con una barra de navegación que permite tener shortcuts a las principales secciones a las que el usuario puede querer acceder. Con esto, se permite que el usuario rápidamente tenga acceso a la sección que desea.

- Feedback

Se intentó que, a través de mensajes enviados por el sitio, el usuario sepa qué es lo que está ocurriendo cuando, por ejemplo, clicaea un botón, o agrega un ejercicio. De esta manera, el usuario tiene información en tiempo real de lo que ocurre.

- Cierre de diálogos

La página no maneja cuadros de diálogo, por lo que no es un punto a analizar.

- Simple manejo de errores

Intentamos hacer que los errores sean comunicados de la manera más directa posible para que el usuario sepa qué es lo que ocurre en determinado instante.

- Revertir fácilmente las acciones

Para esto, implementamos botones para volver atrás en caso de un click equivocado en las vistas donde resultaba relevante. Como grupo consideramos que colocar un botón de “volver” en todas las vistas era hasta invasivo, por lo que lo agregamos en vistas puntuales donde el usuario podía cometer un error apretando un botón.

- Sentir el control

El usuario en todo momento puede abortar una acción o moverse por la página con total libertad. Cuando ejecuta una acción, se lo notifica por proceso exitoso o no exitoso.

- Reducir lo que el usuario debe recordar

Este análisis va de la mano con la claridad de la presentación. Con que el usuario recuerde sus credenciales, tiene posibilidad de acceder a la funcionalidad completa del sitio web. A su vez, la presentación del sitio le permite no tener que recordar qué pasos tomar para llegar de una página a otra.

3.3) Modelos de persona

El diseño del sitio web tomó en consideración varios de los aspectos analizados para los distintos modelos de persona desarrollados en la primera entrega.

Pensando en las consideraciones que tenía Pilar a la hora de utilizar la aplicación, se realizó un especial énfasis en el diseño estético de la página, dando una imagen atractiva desde el inicio de sesión hasta la página de ayuda. También, al poder ver las rutinas diseñadas por todos los usuarios (aquellas que hayan sido diseñadas como públicas), Pilar puede cumplir su objetivo de basarse en rutinas ya diseñadas.

Para Hernán, el foco estuvo puesto en la sencillez. Buscamos que los usuarios como Hernán puedan acceder a sus rutinas de manera directa a través de la barra de navegación o desde el menú de inicio.

Luego, para usuarios como Lara, construimos una interfaz de agregado, edición, y borrado de ejercicios completamente sencilla, para que modificar sus ejercicios sea una cuestión de segundos.

Finalmente, para Marcela, se realizó hincapié en la claridad del sitio, a través del uso de íconos grandes y representativos, acompañados de una palabra/oración indicando a lo que refiere.

3.4) Sugerencias/modificaciones realizadas por la cátedra

- Evaluar la posibilidad de agregar el logo o nombre de la aplicación a la pantalla de registro y de inicio de sesión.

Esto fue tomado en consideración, y al iniciar sesión o registrarse ahora se muestra la barra de navegación, exponiendo el logo y nombre de la aplicación

- Evaluar la posibilidad de cambiar la imagen de la pantalla de registro por una más representativa.

Como grupo coincidimos en la evaluación. La imagen resultaba poco representativa, y fue reemplazada por una adecuada.

- Evaluar la posibilidad de tener una mayor consistencia respecto del diseño de los botones (por ejemplo el de "Enviar" e "Ingresar").

El diseño de los botones se intentó mantener de manera consistente para que el usuario se familiarice con los mismos y los reconozca sin hacer un gran esfuerzo.

- Evaluar la posibilidad de manejar una jerarquía de tamaños de tipografías más consistente.

Se buscó mantener los distintos elementos con sus tipografías de manera consistente.

- Evaluar la posibilidad de ubicar la opción de cerrar sesión junto a las opciones del perfil de usuario.

Se añade el botón de cerrar sesión al lado del botón de "Mi Perfil", en la barra de navegación, de modo tal que el usuario pueda visualizar claramente cómo cerrar la sesión.

3.5) Diseño gráfico

En lo que concierne al diseño gráfico, se intentó mantener todo el sitio de manera consistente en lo que abarca colores, tipografías, íconos, etc. Las imágenes utilizadas fueron extraídas todas de la misma fuente, los íconos pertenecen todos al sistema de íconos impulsado para los lineamientos del "Material Design", la paleta de colores fue definida desde un comienzo y aplicada a través de todo el sitio.

4) Archivos necesarios para el correcto funcionamiento del sitio

Para poder utilizar el sitio de manera correcta, es necesario contar con el archivo **PushApp.rar**, en el cuál se encuentran la API necesaria para el funcionamiento del sitio, y los archivos fuente del sitio.

5) Instructivo de instalación

Antes de pasar a la etapa de instalación del TP, es necesario asegurarse que la API esté correctamente seteada (e instalada). El TP utiliza la API en el puerto 8088, por lo que se requiere que si la API está seteada en otro puerto, se modifique el mismo al 8088 (ver en aclaraciones) o bien modificar el TP de la siguiente manera:

- En el archivo “api.js”, ubicado en la carpeta “PushApp/api/api.js”, hay una función llamada “baseUrl()”. Esta función establece el link para acceder a los llamados de la API. Actualmente, tiene una única línea que dice “ return <http://127.0.0.1:8088/api> ”. Si el usuario desea que la API comience desde otro puerto, por favor sustituya el valor “8088” en el string por el puerto indicado en la configuración de la API.

Ahora sí, habiendo establecido el puerto para la API, pasamos a la instalación del TP:

- 1) Abrir otra consola, y situarse esta vez en la carpeta principal (es decir, la primera carpeta que tenemos cuando descomprimimos el .rar).
- 2) Allí, ejecutar el comando “npm install”, y esperar nuevamente a que concluya
- 3) Una vez que esto termina, ejecutar el comando “npm serve” y el sitio web ya se podrá acceder desde la URL “<http://localhost:8080>”

Con estos pasos, se debería poder llegar desde PushApp.rar hasta la ejecución de la página sin ningún problema.

Algunas aclaraciones:

- Si uno desea modificar el puerto en el cuál se sitúa la API, se puede realizar de la siguiente forma:
 - 1) Modificar el archivo .env, situado en la raíz de la carpeta de la API, en la línea que dice “PORT=8088” por el puerto deseado.
 - 2) Modificar el archivo “swagger.yaml” situado dentro de la carpeta “api” en la línea “host: ‘localhost:8088’ ” nuevamente por el puerto deseado (asegurar que ambos archivos sean modificados por el mismo puerto).
- El trabajo fue testeado en los navegadores: Microsoft Edge, Google Chrome, y Mozilla Firefox, como fue requerido. Si bien debería tener compatibilidad con una gran mayoría de navegadores, algunos tienen situaciones que complican el funcionamiento, como por ejemplo “Brave”. Es por eso que recomendamos fuertemente que la ejecución se realice en uno de los navegadores mencionados.
- Se da por sentado que la instalación de la API fue realizada previamente por el usuario. Caso contrario, el trabajo práctico cuenta con una guía de instalación para la misma (pero no cuenta con los archivos fuente dado que se requirió que no sean entregados).