



Segunda entrega
HCI

Implementación del sitio Web



Grupo 2

- Zornio Juan - Legajo 55039
- Dorado Tomás - Legajo 56594
- Battilana Joaquín - Legajo 57683
- Pouthier Florian - Legajo 61337

Índice

Introducción	2
Decisiones previas a la implementación	3
Páginas del sitio web	3
Instructivo de instalación	3
Navegadores soportados	4
Conclusión	4

Introducción

El objetivo de este trabajo es presentar el sitio web Smart Home, el cual permite controlar y realizar acciones en forma remota sobre dispositivos inteligentes que se encuentren presentes en un hogar.

Por lo tanto, a lo largo del siguiente informe, se expondrán las distintas páginas del sitio web, resaltando las diferencias entre dichas páginas y los prototipos de las mismas durante la etapa de implementación, así como también las decisiones de usabilidad y diseño tomadas para la construcción del sitio.

Y por último, también se proporcionará un instructivo de instalación, con la secuencia de pasos que debe realizarse para instalar y configurar el sitio Web en un servidor Web.

Decisiones previas a la implementación

Al comenzar el desarrollo no se había tomado la decisión si de realizar la web en React o en Vue, pero finalmente se decidió por React, que algunos miembros del grupo ya tenían conocimientos, para facilitar el desarrollo.

Se utilizaron varias librerías, como axios para la conexión con la api, react-redux para hacer un estado común entre los componentes, redux-forms para el manejo de los forms y react router para el manejo del enrutamiento de URLs.

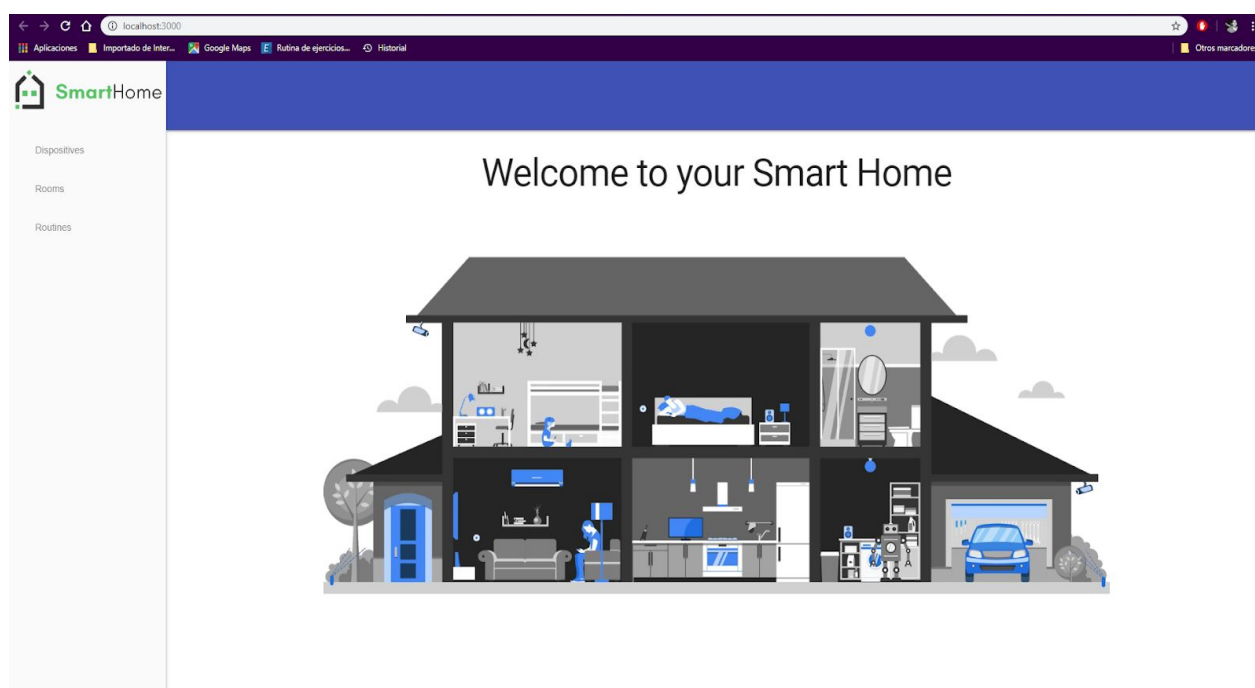
Para la parte de estilado se decidió usar Material Design Lite, la implementación oficial de material design de Google para web. Se usaron solo clases de estilo, ningún componente ya hecho. También se usó toastify para el muestreo de feedback con toasts.

Páginas del sitio web

A continuación podrán verse las capturas de pantalla de las distintas páginas del sitio web.

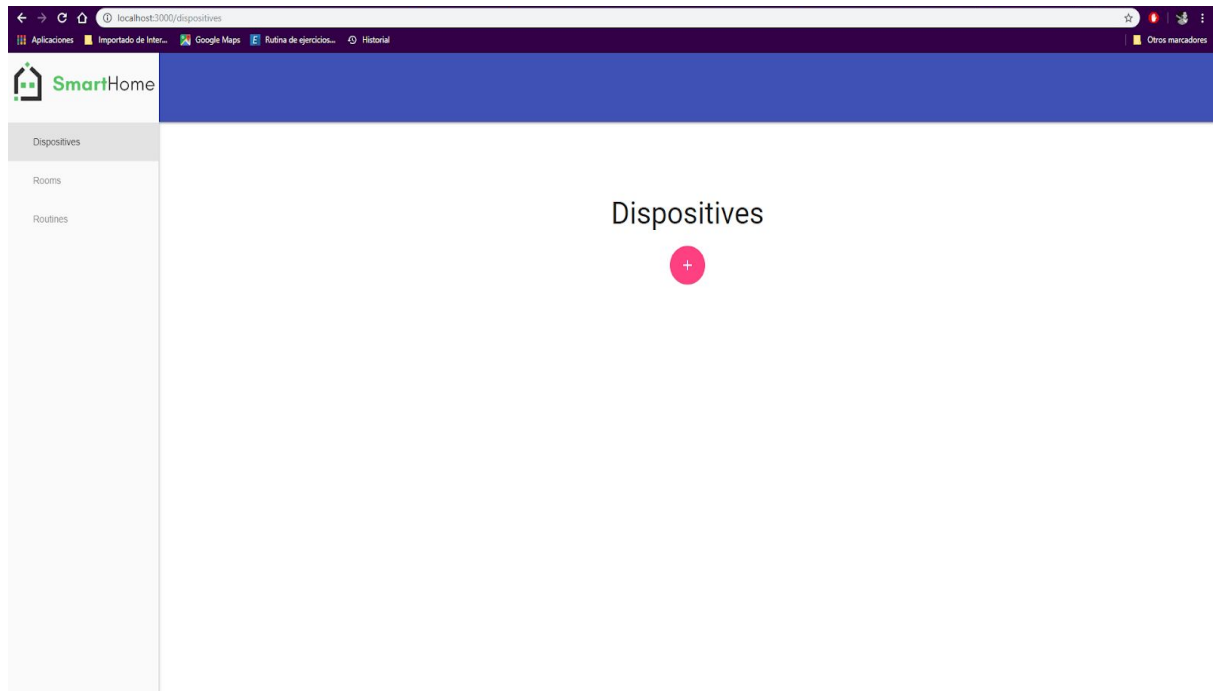
El primer cambio que se decidió hacer fue el de eliminar la pantalla de bienvenida, dado que tener que ver eso cada vez que se entra y tener que clicar en el botón Ingresar para poder entrar al sitio, genera ruido visual innecesario.

Entonces cuando el usuario ingresa al sitio, va a ver la siguiente pantalla con un mensaje de bienvenida:

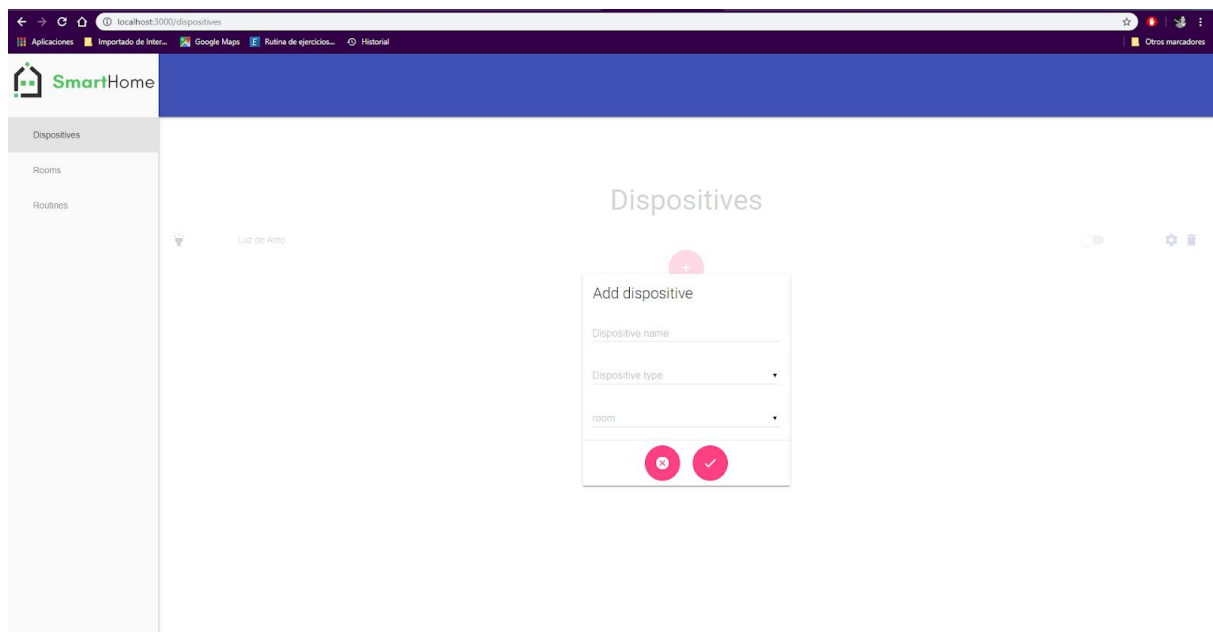


Además, se decidió cambiar el icono y el logo, por un tema de diseño estético.

Luego al hacer clic, en el tab de *Dispositivos*, verá lo siguiente:

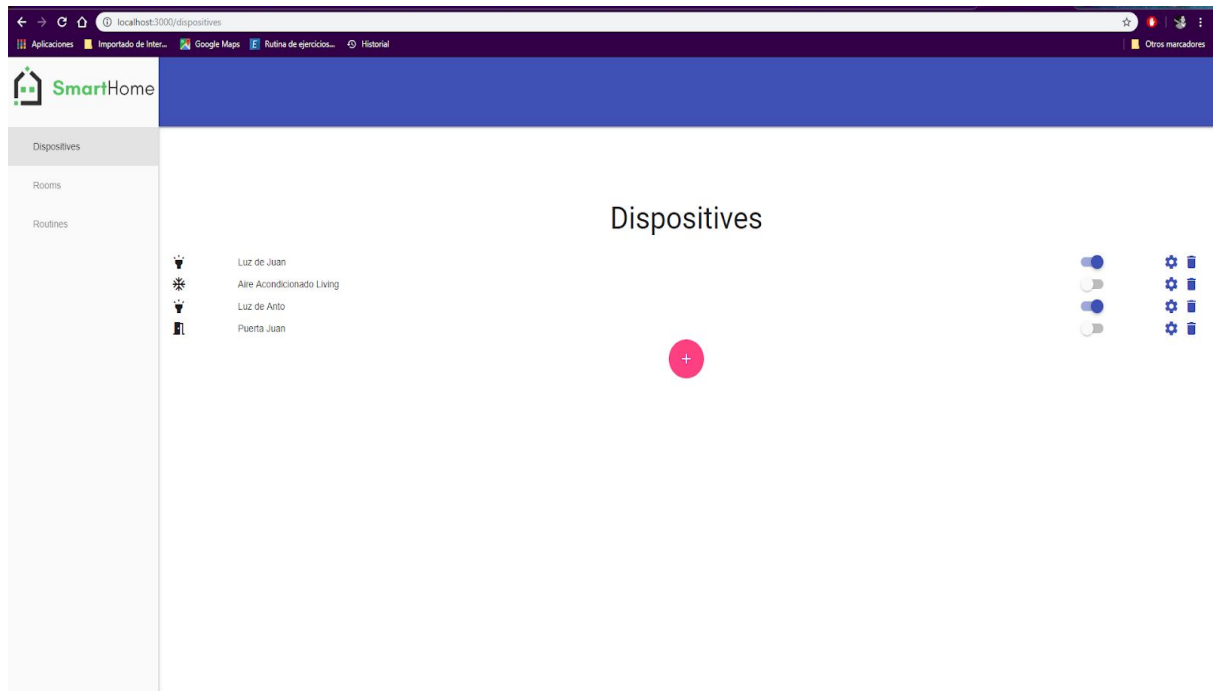


Nótese que aparece vacío dado que no hay ninguno dispositivo agregado. Si se desea agregar uno, se abre la siguiente ventana, en donde se elige el Nombre para el dispositivo (a elección del usuario), así como también el Tipo de dispositivo (aparece una lista de los dispositivos soportados) y si va estar asociado a alguna habitación o suelto:



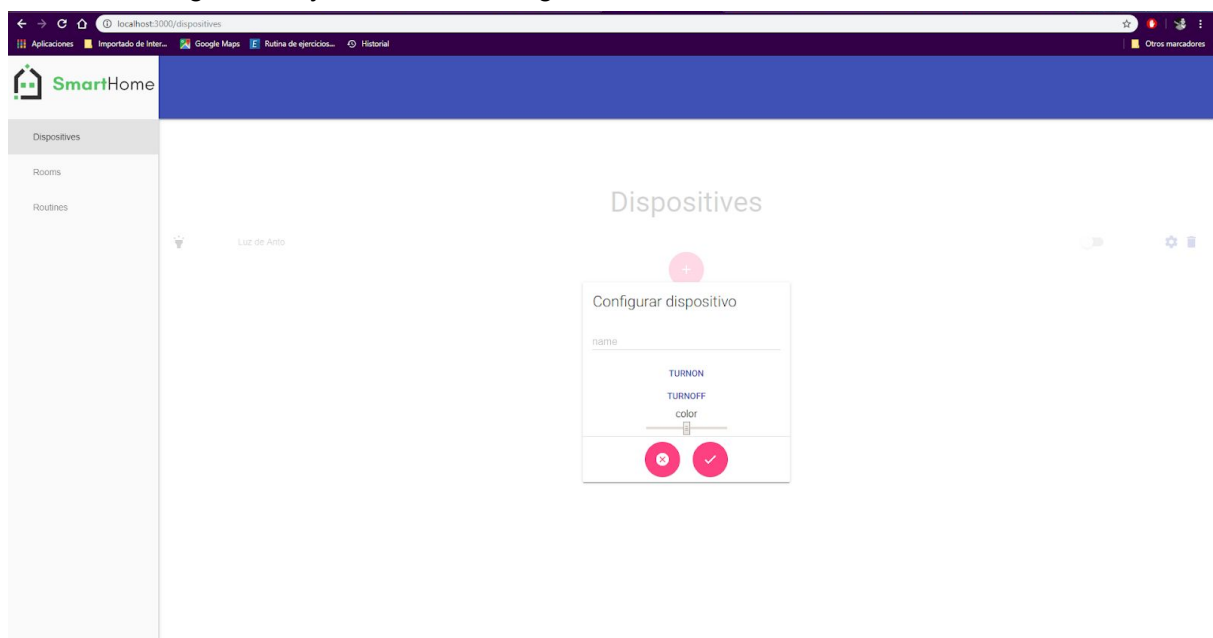
luego, aparece un mensaje de que se agregó correctamente

Luego de agregar varios dispositivos, dentro del tab *Dispositivos* se observa:

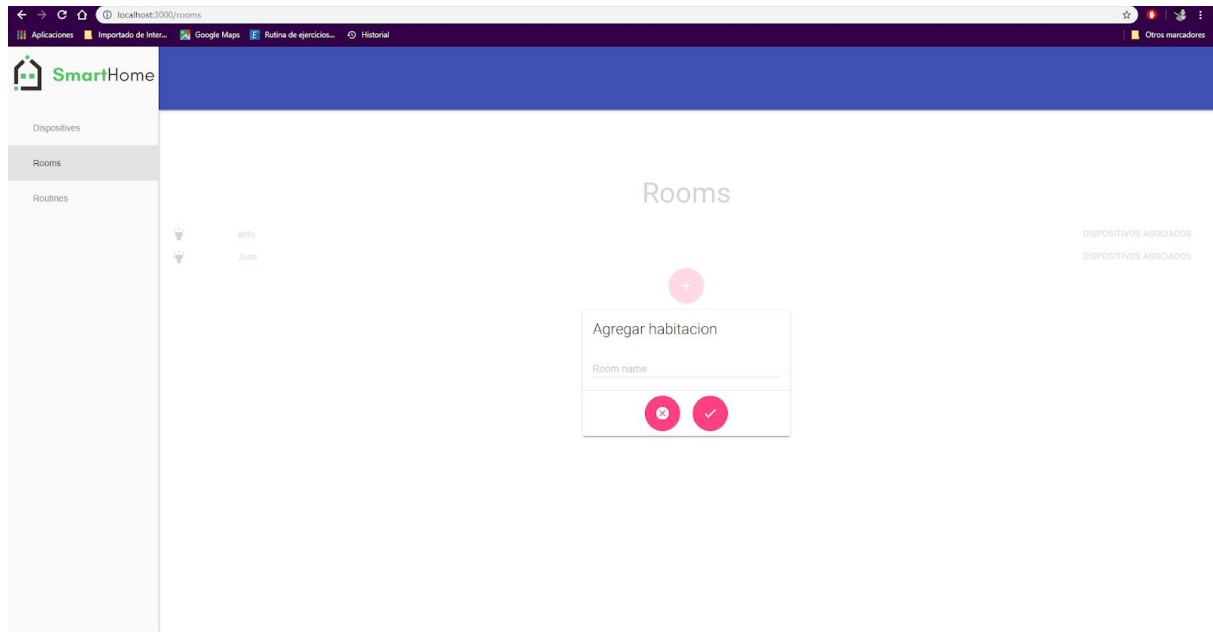


Si se desea eliminar un dispositivo, se hace click en el botón de tachito de basura, el dispositivo se elimina y aparece un mensaje diciendo que el dispositivo fue eliminado.

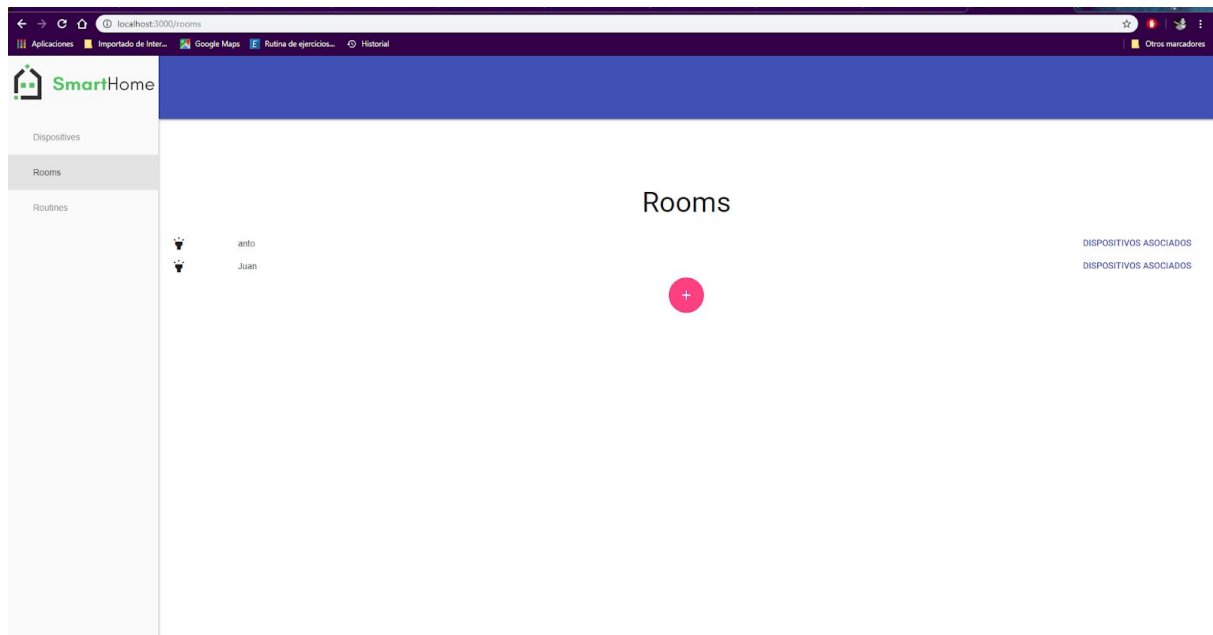
Ahora sí se quiere ver un dispositivo en detalle o si se lo quiere editar se hace clic sobre el botón de configuración y se muestra lo siguiente:



Ahora, sí se selecciona el tab *Rooms* y se quiere agregar una nueva habitación, se abrirá la siguiente ventana ,donde el usuario puede escoger el nombre que desee :

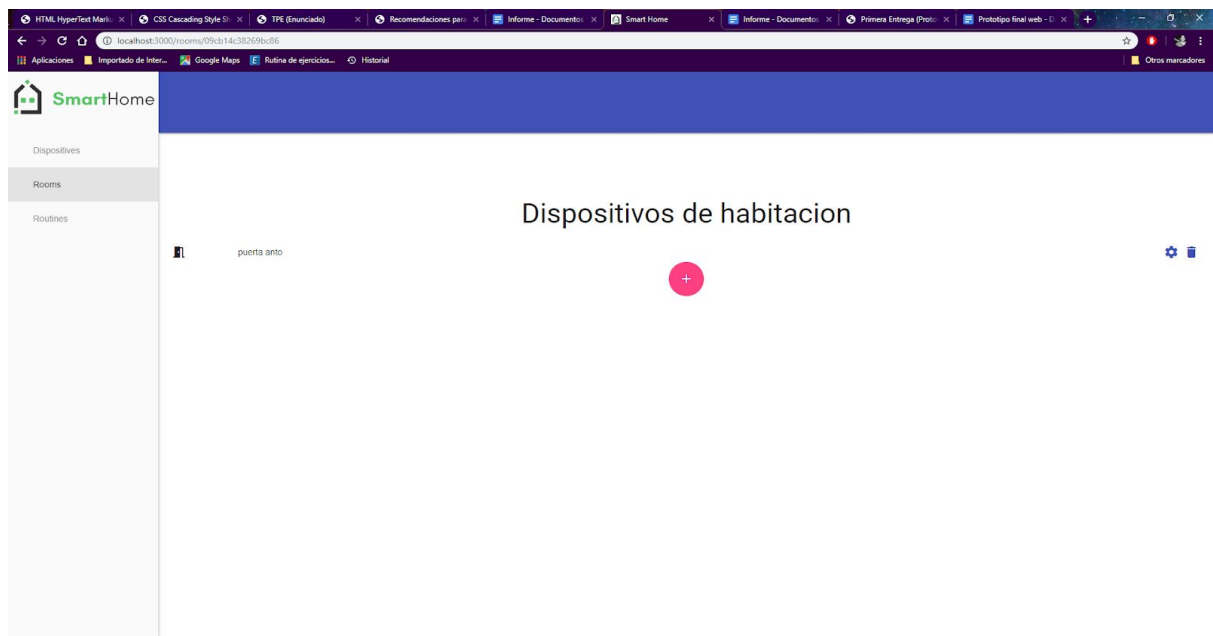


Luego de agregar varias habitaciones, luce de la siguiente manera:

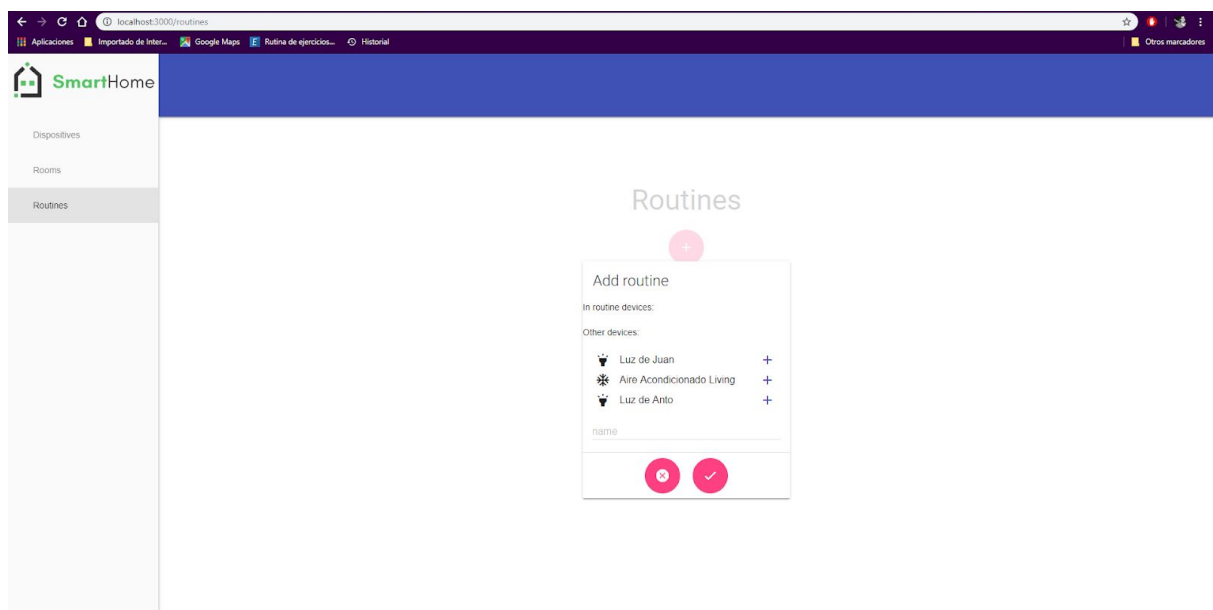


Las habitaciones no se pueden editar ni eliminar, dado que decidimos no implementarlo

Si se quiere ver qué dispositivos están asociados a una habitación, se hace clic en la opción:



Ahora, sí se va a la sección de *Routines* y se desea agregar una rutina nueva, se abre la siguiente ventana:



Se modificó la forma en que se ejecuta una rutina. En el prototipo final del informe anterior, la forma de utilizar rutinas era haciendo clic en el botón switch para ejecutarlas, lo cual hacía que la rutina siguiese activada hasta que se haga clic nuevamente para apagarla. Se modificó ese comportamiento sustituyendo el switch por un botón de Ejecutar, dado que la rutina tiene que ser de ejecución instantánea y no deben permanecer “prendidas”. Además se eliminó el temporizador de la rutina puesto no es necesario tener que planificar su acción futura sino que se activa manualmente al momento de querer utilizarla. Y también, dado que no eran requisitos funcionales obligatorios se decidió no poder editar ni eliminar rutinas.

Instructivo de instalación

A continuación, se detallará la secuencia de pasos que deben realizarse para instalar y configurar el sitio Web en un servidor Web.

1. Instalar node js en al menos la versión 10.15.3 (<https://nodejs.org/en/>)
2. Clonar el siguiente repositorio de git: <https://github.com/JoaquinBattilana/hci-web>
3. Desde una consola en el directorio hci-web/api ejecutar los siguientes comandos:
 - 3.1. npm install
 - 3.2. npm start
4. Luego la api ya estará ejecutándose.
5. Desde una consola en el directorio hci-web ejecutar los siguientes comandos:
 - 5.1. npm install
 - 5.2. npm start
6. Luego se abrirá automáticamente la página web sobre el explorador predeterminado, o abrir la web ingresando a <http://localhost:3000/>
7. Si se quiere cambiar la ruta de la API se puede simplemente entrando a src/config/api y modificando la URL.

Navegadores soportados

Se le ha otorgado mucha importancia a la compatibilidad para que funcione en la mayor cantidad de navegadores posibles. Dentro de los navegadores soportados en los que funciona se encuentran:

- Google Chrome 64 (o superior)
- Mozilla Firefox 60 (o superior)

Errores conocidos/ Implementaciones que faltaron:

Hubo muchas cosas que no pudimos implementar por falta de conocimiento/tiempo, principalmente: cuando uno va a editar un dispositivo, no se carga el estado actual del dispositivo.

Se eliminó la opción de ordenar por determinada categoría.

En las rutinas cuando vamos a usar una de las acciones que son booleanas no andan y tiran la aplicación, esto es porque no sabíamos cómo implementar esos casos en las rutinas.

Nos faltó auto actualizar las cosas cada ciertos ms para que sea responsive, actualmente cada vez que se agrega o borra algo hay que actualizar con f5 para que se muestre lo nuevo.

Nos faltó muchos catch en el caso de la conexión con la API y validaciones de front en caso de que nos desactiven JS.

Así mismo nos faltó agregar muchos toast o notificaciones para avisar de los cambios importantes que van sucediendo mientras el usuario usa la app, si bien los más importantes están, faltaron algunos para seguir la consistencia.

Conclusión

En este informe se expuso el sistema Smart Home mostrando las páginas del sitio, su forma de uso y marcando las diferencias entre dichas páginas y los prototipos, donde pueden notar varios cambios realizados, algunos debido a decisiones de usabilidad y otros a correcciones y falta de tiempo.

El trabajo práctico, que fue extenso y complejo, ayudó para el aprendizaje de diversos conocimientos de desarrollo y programación web, como lenguajes, herramientas, frameworks y además ayudó a aplicar dichos conocimientos a la construcción de un sitio web.