

INFORME

Aplicación mobile



Della Sala, Rocío 56507

Giorgi, M. Florencia 56239

Santoflaminio, Alejandro 57042

Interacción Hombre-Computadora

1er cuatrimestre 2018

Grupo 1

Indice

Introducción	2
Aplicación vs Prototipo	2
Diseño final	6
Devices	7
Notifications	12
Routines	13
Settings	14
Help	15
Instructivo de instalación	15
Conclusiones	16
Anexo	16

Introducción

En el siguiente informe se detalla el diseño de las pantallas de la aplicación mobile *TechHaus* y su funcionamiento.

En primer lugar, se mostrarán las modificaciones que sufrió la app tanto frente al prototipo final del informe anterior, como también a lo largo de su implementación. Las similitudes, en la mayoría de los casos, no serán mencionadas para no extendernos. También mencionaremos algunas diferencias en cuanto a las funcionalidades del sitio web.

En segundo lugar, se incluirán capturas que muestren las pantallas de la versión final de la aplicación.

Asimismo, durante el desarrollo del informe se explicitan las decisiones de usabilidad tomadas durante la etapa de implementación. Las mismas quedarán sustentadas por la teoría y/o por las conclusiones de las entrevistas, observaciones y evaluaciones incluidas en el informe anterior.

Por último, se incluye una conclusión del trabajo realizado junto con el instructivo de instalación.

Aplicación vs Prototipo

En esta sección se detallan los cambios en la disposición y diseño de los elementos de la app frente al prototipo final del informe anterior. Además, se detallan los cambios que fue sufriendo la misma a lo largo de su implementación hasta llegar a su versión final.

En cuanto a los colores de la aplicación, se optó por continuar utilizando los que habíamos elegido para el sitio web (azul petróleo para los fondos y los iconos y verde agua para los detalles como switches y sliders) debido a que su combinación fue satisfactoriamente aprobada por la cátedra.

En primer lugar, comenzaremos mencionando los cambios presentes en la pantalla de “Home”, la cual se corresponde con la pantalla de “Devices”.

En cuanto a lo que respecta en cambios únicamente sobre “Devices” se puede ver que algunos iconos cambiaron, como por ejemplo el de las cortinas y los aires acondicionados. Además, se achicó su tamaño, ya que en la teoría se recomienda que las imágenes que se usan sean pequeñas y estén separadas por un mínimo de 8 DP. Por último, se juntó el nombre de cada tipo de dispositivo con el icono (ver Figuras 1 y 2).

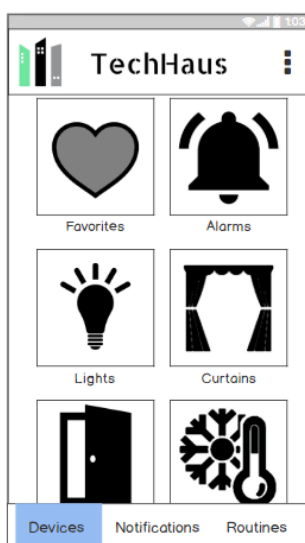


Figura 1: Pantalla “Devices” del prototipo.

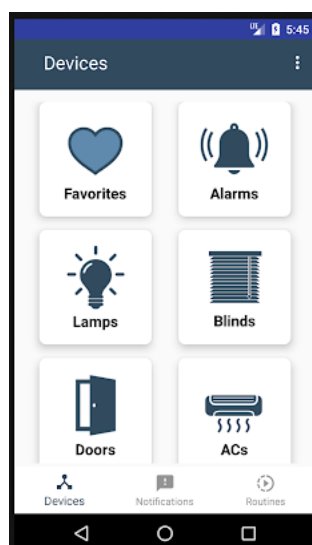


Figura 2: Pantalla “Devices” final.

A diferencia del sitio web, la app no presenta la categoría “All”. La elección de no agregarla se debió a que consideramos que resulta incómodo tener una lista demasiado extensa en un dispositivo chico como es un smartphone.

Los otros dos cambios que mencionaremos estuvieron relacionados con los elementos de navegación, los cuales no solo aparecen únicamente en esta pantalla sino también en el resto de la app.

El primero fue la modificación el diseño de la barra de acción principal. La misma contenía el logo de *TechHaus* para todas las pantallas, pero la cátedra recomendó que era preferible mostrarle al usuario el nombre de la pantalla en la cual el usuario estaba ubicado (llamado vista de control). Esto tenía mayor concordancia con el diseño de aplicaciones para Android. En cuanto a lo que respecta a los botones de acción u overflow de los mismos, eso permaneció sin modificaciones (ver Figuras 3 y 4).

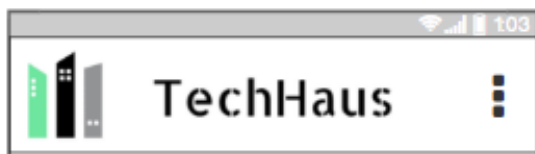


Figura 3: Barra de acción principal del prototipo.



Figura 4: Barra de acción principal final.

El segundo fue la modificación del diseño de la barra de acción secundaria. Acompañamos cada sección con un icono, ya que esto ayuda a facilitar la interacción con el usuario (ver Figura 5 y 7). Por otro lado, sabemos que existen iconos predefinidos para la barra de acción, pero al no tratarse de iconos estándar de Android, es mejor acompañarlos con una breve descripción de su utilidad para no confundir al usuario.

El color de la barra de abajo sufrió modificaciones ya que en un primer momento indicábamos con verde agua en la pantalla que nos encontrábamos y con azul en las que no. Optamos por cambiar esto a azul petróleo y gris respectivamente, por dos cuestiones. En primer lugar, el color verde agua escogido es brillante y no es recomendable utilizar colores de este tipo en texto “común” (solo para hacer énfasis y este no era el caso). En segundo lugar, coincidimos que el color gris se representa mejor con la inactividad que color azul petróleo (ver Figura 6).



Figura 5: Barra de acción secundaria del prototipo.

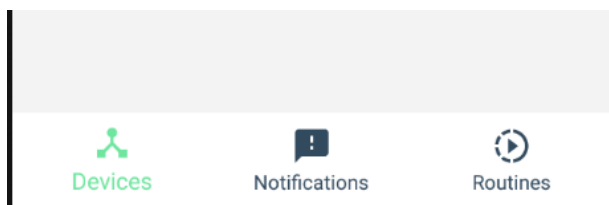


Figura 6: Barra de acción secundaria primera versión.



Figura 7: Barra de acción secundaria final.

Por otro lado, realizamos un cambio que corresponde con la interacción del usuario con la aplicación. En un principio, si el usuario scrolleaba para abajo, la barra de acción superior se movía hacia arriba y “desaparecía”. En el diseño final la misma queda fija, aunque el usuario scrollee.

En cuanto a la pantalla donde se listan los diferentes dispositivos según su categoría hicimos dos modificaciones en la estructura de la barra de cada dispositivo.

En primer lugar, se eliminó la flecha para navegar hacia el dispositivo ya que no estaba en concordancia con el material de diseño de Android, sino que correspondía más con iOS. En su lugar, se accede al mismo clickeando en la barra (ver Figuras 8 y 10). Esto se aplicó también para la lista de rutinas que veremos más adelante.

En segundo lugar, la barra contenía un acceso rápido a la funcionalidad principal del dispositivo. Llenar la barra de funcionalidades cargaba mucho a la interfaz, lo que además hacía difícil su utilización por el poco espacio que contienen las pantallas de los celulares y/o tablets en comparación con las computadoras. No podemos considerar un dispositivo móvil como un pequeño desktop, por ende, no podemos migrar el diseño de la página web hacia la aplicación (ver Figuras 8 y 10).

Además, debido al escaso espacio que tenemos en estas pantallas, al usuario le podría pasar que termine presionando en la funcionalidad de un dispositivo que no era el que quería modificar. Esto lo cambiamos por el icono del corazón para poder marcar y/o desmarcar como favorito sin tener que entrar al dispositivo.

En un primer diseño, el corazón que marcaba/desmarcaba como favorito estaba ubicado a la izquierda del nombre del dispositivo (ver Figura 9). Al quitar la flecha, corrimos dicha funcionalidad a la derecha debido a que los usuarios suelen agarrar el celular con una mano y utilizar el dedo pulgar para manejarlo. Esto facilita su utilización ya que el dedo se encuentra más cerca de la parte de abajo y la parte derecha de la pantalla.

En cuanto a la barra de acción, como sacamos el logo de TechHaus, también subimos la flecha “up arrow” que nos permite volver a la pantalla anterior (ver Figura 10). Esta flecha solo aparece en algunas pantallas, por ejemplo, ya vimos que en “Devices” no estaba, ya que no tiene sentido porque no se puede volver hacia atrás en dicha pantalla.

Esta flecha no se corresponde con la flecha que trae el celular en la parte inferior izquierda, ya que las mismas tienen diferentes funcionalidades. La que agregamos nosotros, sirve para movernos entre pantallas que tengan un orden jerárquico, es decir, si el usuario se encontraba ubicado en los detalles de una puerta, al presionar la flecha volverá a la lista de puertas. Si en cambio, el usuario al entrar a los detalles de una puerta presiona las rutinas, al presionar la flecha que trae el celular, en teoría debería volver a los detalles del dispositivo, sin importar que estas pantallas no están relacionadas en la app.

De todos modos, siempre que se vuelve a una de las categorías principales (las que están ubicadas en la barra de acción secundaria), el stack se limpia. De esta forma, si quiero salir de la aplicación bastará con posicionarme en una de estas pantallas y apretar la flecha inferior.

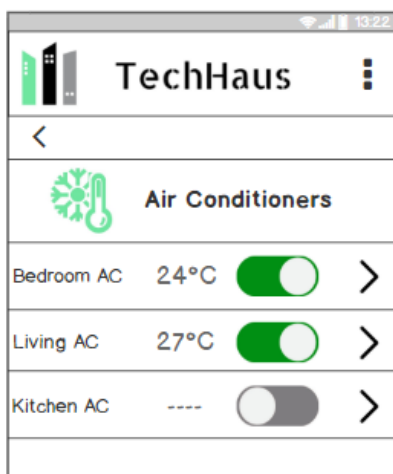


Figura 8: Pantalla “Air Conditioners” del prototipo.

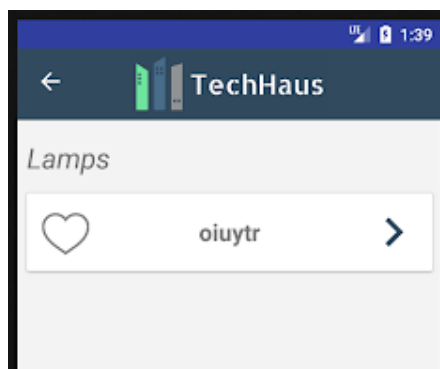


Figura 9: Pantalla “Lamps” primera versión.

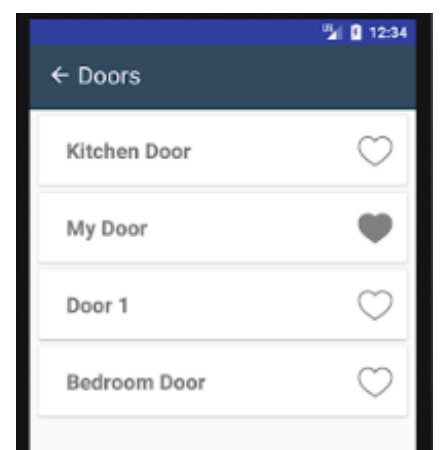


Figura 10: Pantalla “Doors” final.

Al acceder a las características de cada dispositivo, no hay cambios respecto al diseño del prototipo. Las pantallas se adjuntarán más adelante cuando se detalle el diseño del prototipo final.

En cuanto a la pantalla de “Notifications”, la misma también sufrió cambios respecto al prototipo. La misma era en forma de lista, pero no contenía los botones para refrescar (no se contempló esta problemática en el diseño) ni tampoco para borrarlas todas, sino que había que seleccionirlas una por una (ver Figuras 11 y 12). Además, contenía una imagen en el extremo izquierdo que las clasificaba según su relevancia. Por otro lado, algunos mensajes mostraban cuestiones más complejas como que la temperatura del exterior bajó y por ende, el aire acondicionado debería apagarse. Claramente, debido a su complejidad, esto cambió quedando una pantalla de notificaciones mucho más sencilla.

Para la versión final, se agregaron dos botones; “Refresh” y “Clear All”. Además, cada notificación tiene una breve descripción de que dispositivo se modificó, su nombre y la acción realizada (ver Figura 13).

Para eliminar una notificación, no basta con presionar el tachito correspondiente (para visualizar el cambio en pantalla), sino que será necesario refrescar la pantalla mediante el botón “Refresh”. Debido a esto, al borrar una notificación aparecerá un toast notificando que la acción se realizó correctamente. También se pueden borrar todas las notificaciones mediante el botón “Clear All”.

Originalmente la idea era que las notificaciones listadas, pudieran ser borradas cuando se mantenían apretadas mediante la aparición de una opción para realizar tal acción. Por problemas de implementación se terminó realizando la alternativa mencionada.

Por otra parte, si aparece una notificación en el momento en el que estamos en la pantalla “Notifications” entonces para visualizarla en esa misma pantalla es necesario presionar “Refresh”.

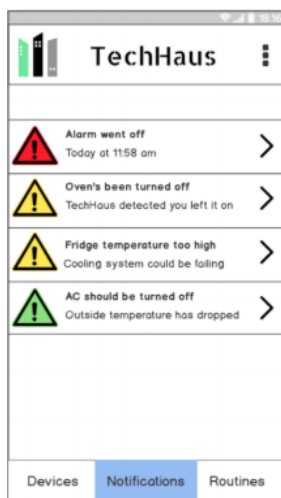


Figura 11: Pantalla “Notifications” del prototipo.

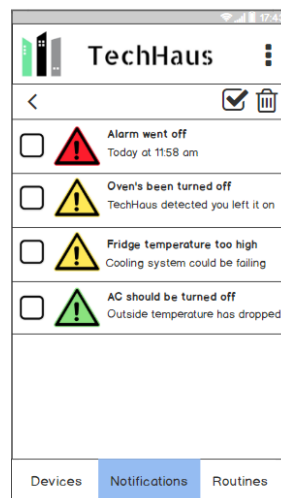


Figura 12: Pantalla “Notifications” del prototipo.

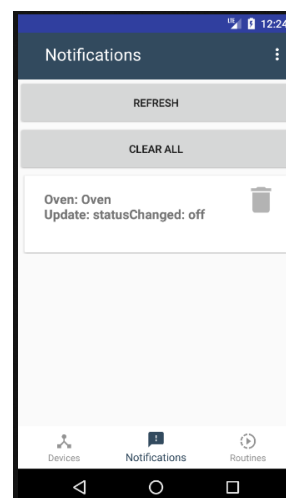


Figura 13: Pantalla “Notifications” final.

Por último, en cuanto a la pantalla “Routines” se produjo un cambio importante en el diseño de la barra que lista cada una de las mismas.

Al igual que sucedía para los dispositivos, en un primer diseño teníamos una flecha que nos llevaba a la siguiente pantalla (ver Figura 14). Como esto fue eliminado, el botón de play se corrió hacia el lado derecho de la pantalla (misma justificación utilizada para los dispositivos y el botón para favear) (ver Figura 15).

Otra aclaración importante es que en la teoría vimos que es recomendable dejar espacio entre los elementos de la aplicación (8 DP) y cómo podemos ver, la barra que detalla a cada dispositivo ahora aparece separada del resto por un “margen”.

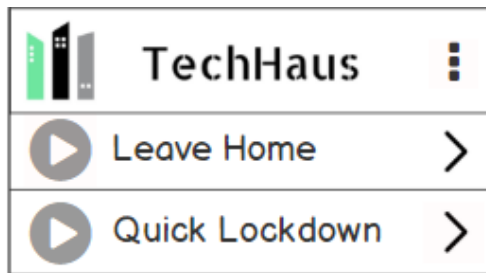


Figura 14: Pantalla “Routines” del prototipo.

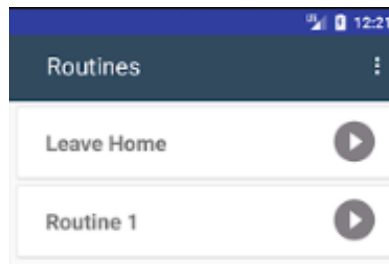


Figura 15: Pantalla “Routines” final.

Nos parece importante hacer notar que no tenemos una barra de búsqueda en la aplicación como sí sucedía en la página web. En principio, esto se debe a que en las interfaces mobile, nos manejamos a través del touch de la pantalla y, como vimos en la teoría, hay que evitar siempre que podamos los campos para ingresar información, ya que los mismos implican tipear (que es una tarea más difícil). Además, al no tener una sección de cuartos ni la categoría “All” como sucedía en la versión web, no creemos que el usuario tenga una lista demasiado extensa que implique usar la barra para buscar. Si puede pasar que, en algún caso particular, quede una lista extensa, como la sección de puertas o de lámparas, los cuales son dispositivos que existen en gran cantidad en un único hogar. En fin, no nos parecía razonable agregar una barra de búsqueda porque implicaba destinar píxeles de pantalla a una funcionalidad que va a ser poco usada.

Con respecto al detalle de cada una de las rutinas, el botón de play se subió a la barra de acción principal, ya que Android establece que los botones de acción de una sección deben estar ubicados en la barra de acción. Por otro lado, como sacamos el logo de TechHaus, también subimos la flecha “up arrow” que nos permite volver a la pantalla anterior (ver Figuras 16 y 17).

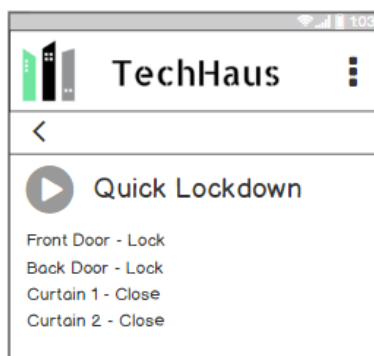


Figura 16: Pantalla “Quick Lockdown” del prototipo.

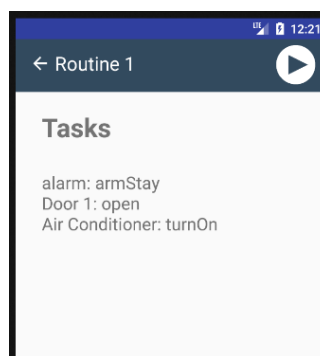


Figura 17: Pantalla “Routine 1” final.

Diseño final

Antes de comenzar mostrando las capturas finales de la aplicación, vamos a desarrollar sobre algunas características generales de la misma que se repiten a lo largo de las pantallas. Intentamos cumplir con todas las pautas establecidas en el diseño Android para desarrollar nuestra aplicación.

La aplicación está comprendida por 3 pantallas principales. La primera es la de “Home” (o más bien “Devices”), “Notifications” y “Routines”. Para acceder a las mismas, dispusimos de una barra de acción secundaria en la parte inferior de la pantalla, donde el dedo pulgar tiene un acceso mucho más rápido y

fácil. El fin es facilitar la navegación y el acceso a las acciones más utilizadas. Esta barra aparece en todas las pantallas salvo en la de “Help” (y sus secciones) y “Settings” (esto mismo sucede en aplicaciones como Spotify, Youtube o Gmail).

También tenemos una barra superior, la cual nos muestra el nombre de la pantalla en la cual estamos parados, es decir, “Devices” o quizá el nombre del dispositivo del cual estamos viendo los detalles. En esta barra, para algunas pantallas se agregaron funcionalidades, como, por ejemplo, para ejecutar una rutina en los detalles de esta. También tenemos la flecha “up arrow” en el extremo izquierdo de la misma para volver a la pantalla anterior.

La app funciona en 2 idiomas, español e inglés, para lo cual se realizaron las traducciones necesarias. Para el caso que la aplicación se encuentre en español, aquellas funcionalidades que son propias de la API no fueron traducidas (por ejemplo, *armAway* para el caso de las alarmas).

Se realizaron las pantallas para celular y tablet en ambas orientaciones, pero no se agregó ninguna nueva funcionalidad que sea distintiva en cada una. Se adjuntarán algunas imágenes representativas de las tablets y de la posición horizontal, pero el trabajo se mostrará en su totalidad con la pantalla de un celular de forma vertical.

Finalmente, podemos agregar que el dispositivo fue probado en la versión de Android Nougat 7.0 en un “Pixel 2 XL” y en un “Nexus 5” (API Level 24).

Devices

En primer lugar, tenemos la pantalla de “Devices”, la cual nos muestra en el área de contenido las distintas categorías donde podemos encontrar a los dispositivos. Cada categoría está diseñada a través de un icono representativo de la misma y debajo de él se coloca el nombre de la categoría por si algún usuario no identifica la imagen con la categoría que corresponde. También se utilizaron dos barras; una barra de acción principal y una barra de acción secundaria que ya mencionamos anteriormente (ver Figuras 18, 19, 20 y 21).

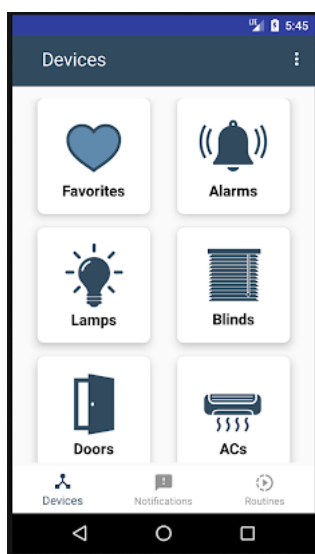


Figura 18: Pantalla “Devices” final.

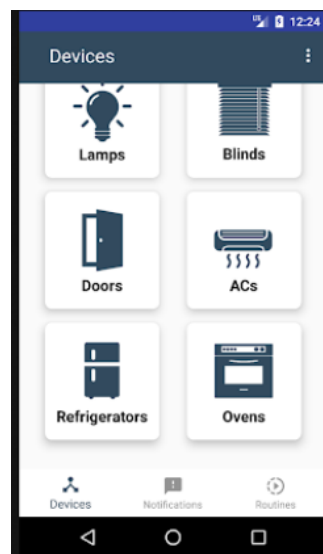


Figura 19: Pantalla “Devices” final.

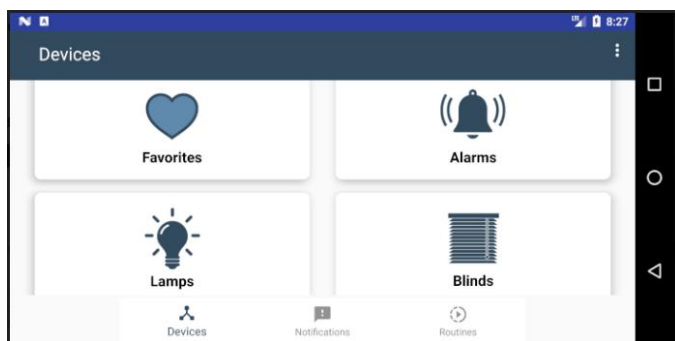


Figura 20: Pantalla “Devices” final, vista horizontal.



Figura 21: Pantalla “Dispositivos” final, tablet, español.

Como podemos ver en total se soportaron 7 tipos de dispositivos distintos: alarma, lámpara, cortina, puerta, aire acondicionado, refrigerador y horno. Vale la pena destacar que también se respetó el mismo orden usado en el sitio web para las categorías, ordenándolas según los resultados que habíamos obtenido en las encuestas.

Al presionar sobre el botón de overflow se despliega un menú con dos opciones: “Help” y “Settings” (ver Figura 22). La descripción de cada una de estas secciones las veremos luego.

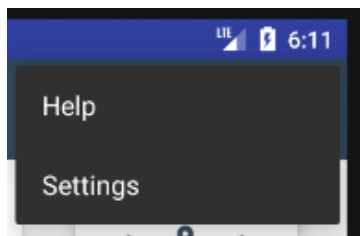


Figura 22: Overflow desplegado.

Al presionar sobre una sección, se cargará una nueva pantalla que contendrá el nombre de la sección en la barra de acción superior y una lista de todos los dispositivos registrados de esa categoría en el área de contenido.

En una primera interacción con la app, en la cual no tenemos dispositivos agregados de una categoría en particular, al acceder por ejemplo a “Doors”, nos encontraremos con una página vacía, pero con un mensaje que le detalla al usuario que no hay dispositivos registrados de ese tipo (ver Figura 23).

Si en cambio tenemos dispositivos agregados, la lista estará formada por múltiples barras que detallan el nombre del dispositivo junto con el icono del corazón para marcarlo y desmarcarlo como favorito (ver Figura 24 y 25).

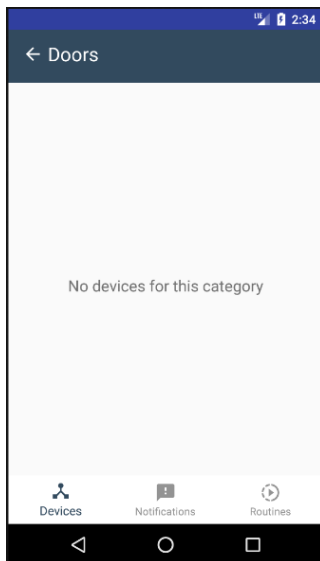


Figura 23: Pantalla “Doors” final, sin dispositivos agregados.

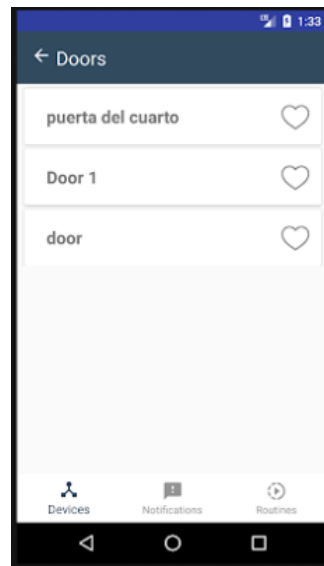


Figura 24: Pantalla “Doors” final, con dispositivos agregados.

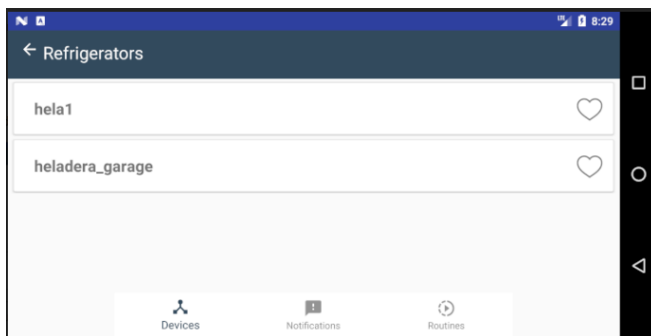


Figura 25: Pantalla “Refrigerators” final, vista horizontal.

Una vez que presionamos en un dispositivo, pasamos a ver las funcionalidades del mismo. En la nueva pantalla, lo que cambia es el área de contenido, en donde tenemos el icono representativo del dispositivo y debajo una lista de las funcionalidades que posee, junto con switches, botones, sliders para cambiar su estado. El título de la funcionalidad se coloca a la izquierda y a la derecha el estado, el cual se puede modificar presionándolo (ver Figuras 26, 28, 30, 31, 32, 33, 34, 35 y 36).

Las funcionalidades que nos permiten seleccionar modos, opciones, colores o requieren un input abrirán un cuadro de diálogo para llevarlas a cabo (ver Figura 29). Otras poseen switches y sliders y se ejecutarán de forma directa. Aquellas que no se realizan de forma directa, (que el usuario no ve inmediatamente el cambio en pantalla) al realizar el cambio aparecerá un toast en la parte central inferior de la pantalla que le confirmará la acción realizada al usuario. Las pantallas de los dispositivos de la misma categoría serán idénticas, lo único que cambiará es el estado del mismo.

Para el caso de las lámparas, es necesario aclarar cuestiones relacionadas al diseño del color picker. Nos hubiera gustado más hacerlo únicamente con círculos de colores para escoger, y que cada uno se ubique siguiendo al otro y no en forma de lista como finalmente quedó (ver Figura 27). Esto se debe a que, de la

primera forma, nos entrarían muchísimos más colores y no sería necesario scrollear en la pantalla. Por cuestiones de complejidad en la implementación, quedó una lista de los colores junto con el nombre (lo cual nos parece redundante pero así estaba implementado en el modelo sobre el cual nos basamos).

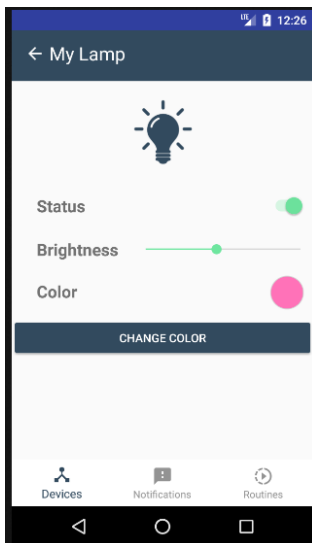


Figura 26: Pantalla "My Lamp" final.

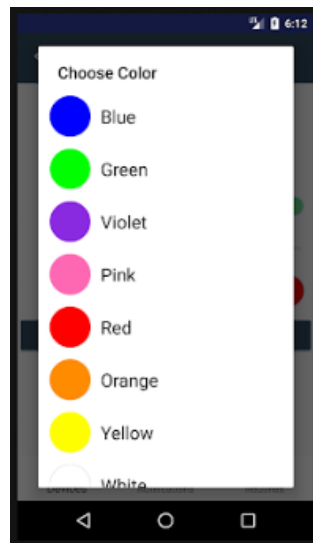


Figura 27: Pantalla "Choose Color" final.

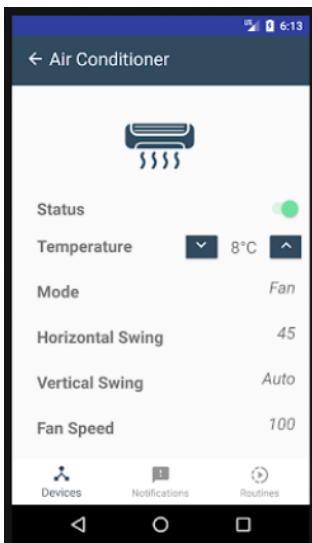


Figura 28: Pantalla "Air Conditioner" final.

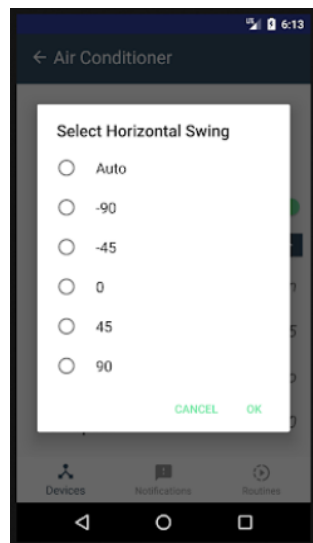


Figura 29: Pantalla "Select Horizontal Swing" final.

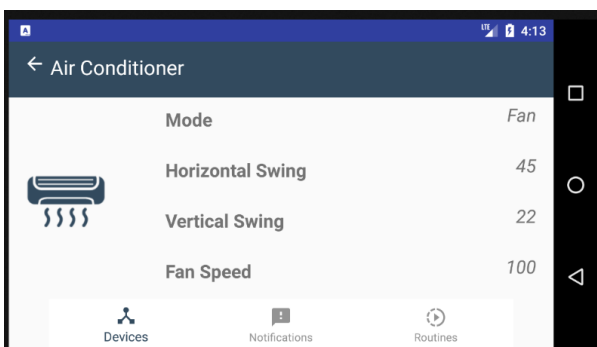


Figura 30: Pantalla "Air Conditioner" final, horizontal.

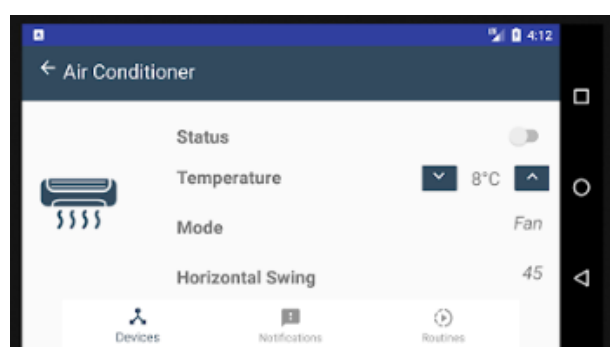


Figura 31: Pantalla "Air Conditioner" final, horizontal.

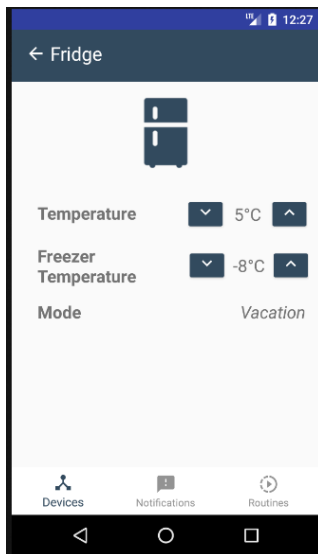


Figura 32: Pantalla “Fridge” final.

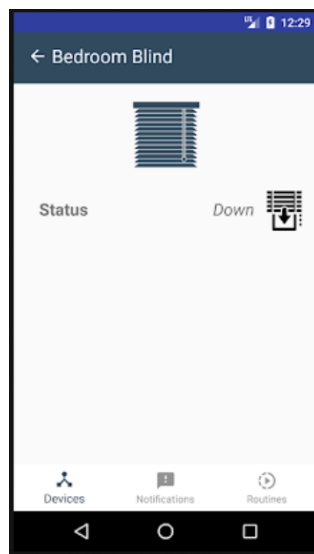


Figura 33: Pantalla “Bedroom Blind” final.

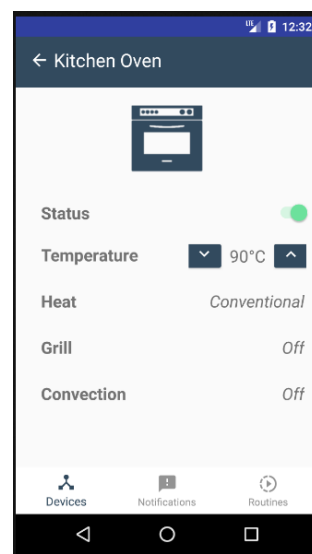


Figura 34: Pantalla “Kitchen Oven” final.

Para las puertas, arreglamos el error que tuvimos en el sitio web. Ya no es posible abrir una puerta si se encuentra “locked” ni tampoco trazarla si se encuentra abierta.

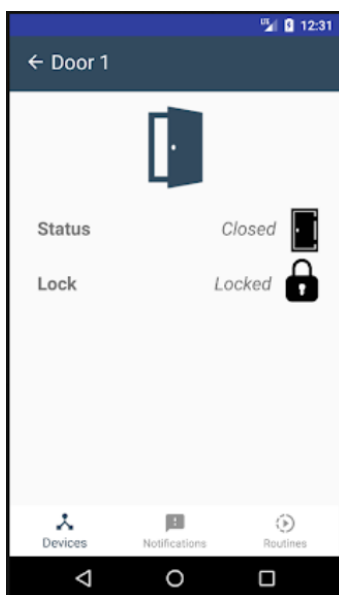


Figura 35: Pantalla “Door 1” final.

Para las alarmas, hay varias cosas que aclarar respecto a su funcionamiento. En primer lugar, si la alarma se encuentra en un estado en particular y el usuario presiona el mismo estado, la app le mostrará un toast en el cual le avisará que el estado no puede ser cambiado (justamente porque es el mismo). En segundo lugar, si la funcionalidad que el usuario quiere ejecutar le pide el código para confirmar y el ingresado es incorrecto, se le mostrará el error también a través de un toast. Por último, para los casos en los cuales el estado es satisfactorio (se cambió de estado o se ingresó el código correcto), también se le notificará al usuario a través de este método.

Por otro lado, se realizó programación defensiva en el caso que el usuario quiera cambiar el código de la alarma. El usuario debe ingresar el código nuevo en formato correcto, sino el código nuevo no será aceptado.

Además, debe ingresar el código viejo para llevar a cabo esta funcionalidad (si es incorrecto se mostrará el toast correspondiente).

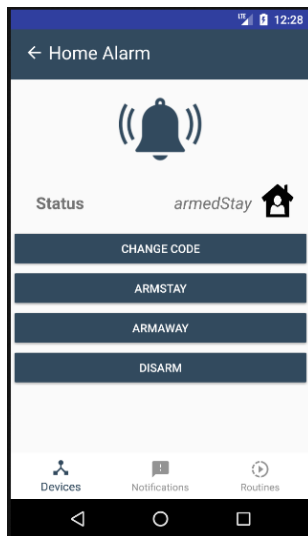


Figura 36: Pantalla "Home Alarm" final.

Notifications

En segundo lugar, tenemos la pantalla "*Notifications*". La misma puede estar vacía (ver Figura 37), es decir, sin notificaciones o con al menos una. Las notificaciones se dispondrán en la pantalla en forma de lista y por encima de la misma se ubicarán dos botones; "*Refresh*" y "*Clear All*" (ver Figuras 38 y 40).

Cada barra que conforme una notificación estará constituida por una descripción en el extremo izquierdo y un tachito en el extremo derecho. La descripción está formada por el tipo de dispositivo modificado, su nombre y la modificación que se le realizó. Para borrar cada una, bastará con presionar el icono del tachito, el cual las borrará directamente sin pedir confirmación, ya que no se trata de una acción importante que requiera que el usuario este seguro. Además, creemos que agregar una confirmación para una acción que no es relevante no es amigable para el usuario, porque lo terminará agobiando.

Para visualizar la eliminación en pantalla será necesario refrescar como fue mencionado anteriormente. De todos modos, aparecerá un toast para mostrarle al usuario que se efectivamente se eliminó, aunque no se vea en pantalla.

Por fuera de la aplicación, cada vez que llegue una notificación correspondiente a TechHaus, la misma aparecerá en el panel de notificaciones ubicado en la parte superior de la pantalla del celular/tablet (ver Figura 39).

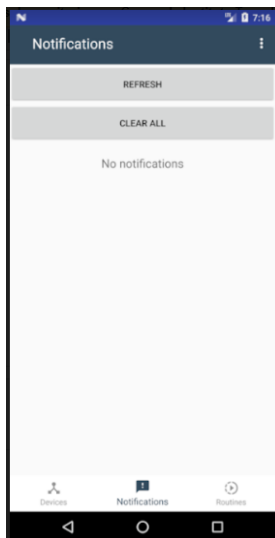


Figura 37: Pantalla “Notifications” vacía final.

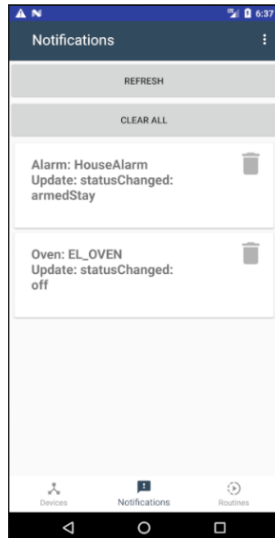


Figura 38: Pantalla “Notifications” con notificaciones final.

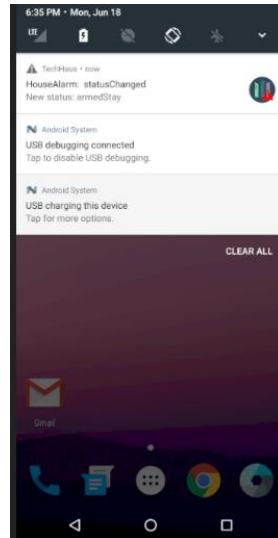


Figura 39: Vista de las notificaciones por fuera de la app.

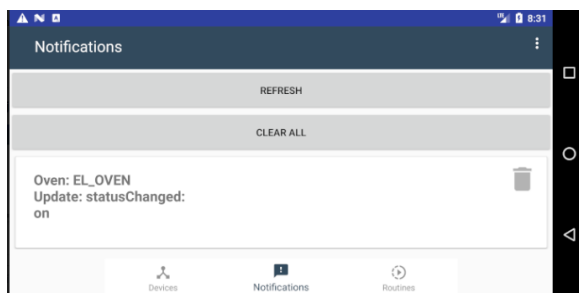


Figura 40: Pantalla “Notifications” final, horizontal.

El chequeo de notificaciones se inicia cuando se bootea el dispositivo. Por eso, en la API, para testearlo ejecutábamos “Cold boot now”, con un teléfono emulado que ya tenía la aplicación instalada. Esto produce que se inicie (sin abrir la aplicación) y comienza a chequear si hubo algún evento que tiene que notificar. Si algo es modificado, se envía una notificación, la cual puede ser abierta, redirigiendo al usuario a la pantalla de “Notifications”.

Routines

En tercer lugar, tenemos la pantalla “Routines”. De la misma forma que los dispositivos, se listan todas las rutinas que están registradas. La misma puede estar vacía y en dicho caso, aparecerá un mensaje que nos avisará que no hay rutinas en el sistema.

En el caso en el cual sí existen rutinas, se mostrarán en forma de lista una debajo de la otra. Podremos ejecutarlas directamente desde esta pantalla o ingresar en los detalles de la misma (para ver acción hace cada dispositivo involucrado).

Al ejecutarlas se aparecerá en la parte inferior de la pantalla un toast indicando que la acción se realizó correctamente.

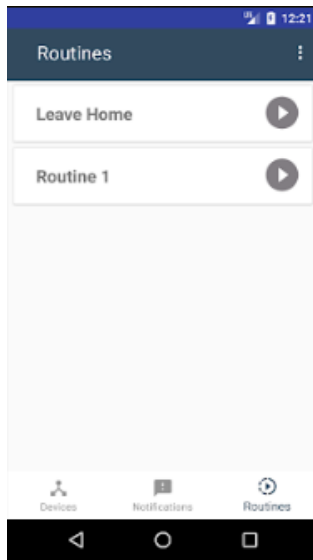


Figura 41: Pantalla “Routines” final.

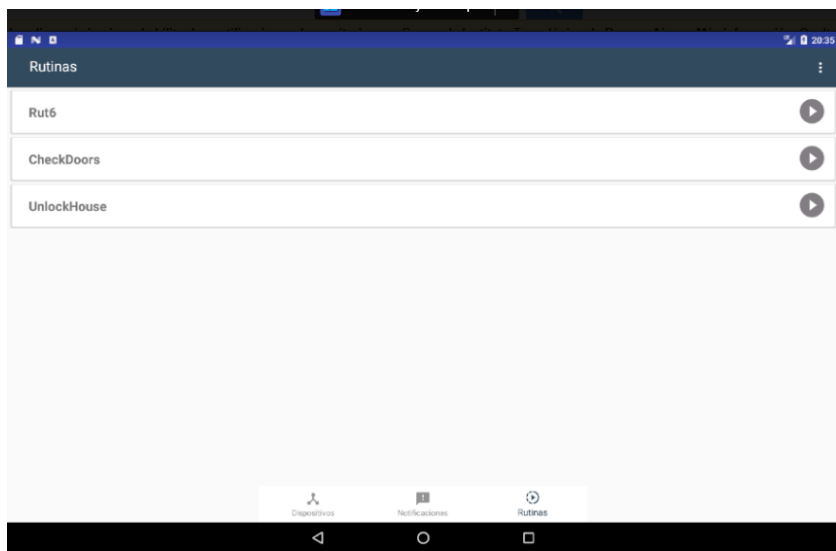


Figura 42: Pantalla “Rutinas” final, tablet, español.

Help

En cuarto lugar, tenemos la pantalla “*Help*”, la cual vamos a poder acceder desde “*Devices*”, “*Notifications*” y “*Routines*”. Nos mostrará una lista de barras de las distintas secciones de ayuda (ver Figura 43). Cada barra contiene una breve descripción sobre lo que trata la sección. El usuario deberá presionar en la barra para acceder a la sección correspondiente. Podemos ver que la barra de acción secundaria desaparece. Por ende, para volver a la pantalla anterior será necesario utilizar la “up arrow” que se encuentra localizada en el extremo superior izquierdo de la pantalla en la barra de acción principal.

El usuario podrá acceder a cada sección donde se encontrará un manual que explica mediante imágenes y texto como utilizar la aplicación (ver Figura 44).

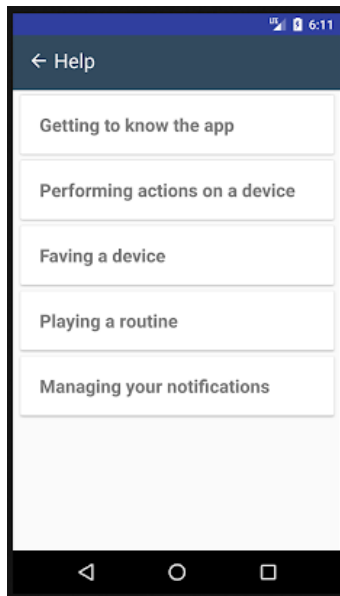


Figura 43: Pantalla “Help” final.

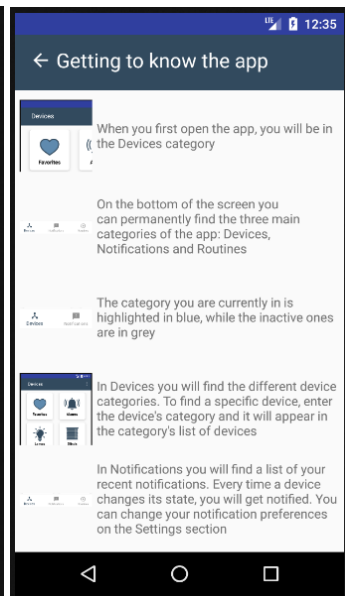


Figura 44: Pantalla “Getting to know..” final.

Settings

Por último, tenemos la pantalla “Settings” donde podemos permitirle al usuario personalizar sobre qué tipo de dispositivo queremos recibir notificaciones (ver Figuras 45 y 46). Por default, las notificaciones están seteadas únicamente para las alarmas.

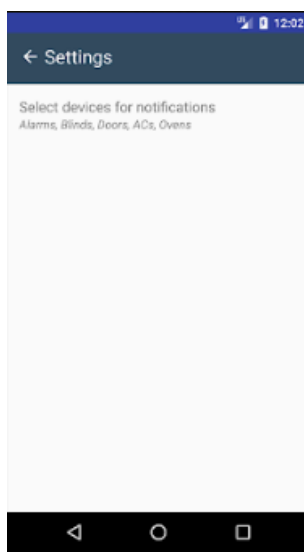


Figura 45: Pantalla “Settings” final.

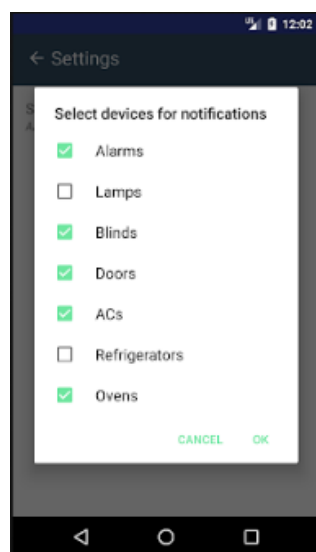


Figura 46: Pantalla “Select devices..” final.

Instructivo de instalación

- 1) Previamente, para acceder a las funcionalidades completas de la app es necesario tener la API provista por la cátedra inicializada.
- 2) Abrir la carpeta del proyecto con Android Studio.
- 3) Si es necesario cambiar el puerto de la URL de la API hacerlo en la clase API.java en el string baseUrl.
- 4) Hacer un build del proyecto.
- 5) Para una funcionalidad completa (especialmente con el tema de las notificaciones) se requiere que el emulador o dispositivo sea booteado con la app ya instalada. Por lo tanto, se sugiere instalar la

app con el botón de play y luego reiniciarlo haciendo un reboot, el cual puede hacerse por ejemplo mediante el uso de "cold boot now" en Android Studio (desde el AVD Manager) (ver Figuras 47 y 48).

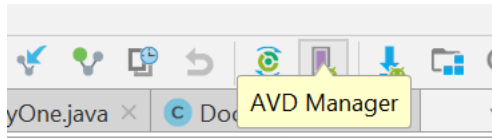


Figura 47: ADV Manager desde Android Studio.

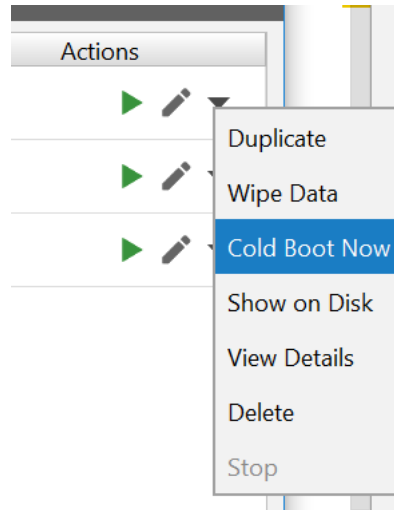


Figura 48: Cool Boot Now desde ADV Manager.

Conclusiones

En resumen, a partir del desarrollo de los tres trabajos entregados durante la materia se lograron implementar conocimientos teóricos respecto a la interacción entre las personas y las computadoras en dos desarrollos importantes (web y móvil).

Cada uno de ellos, tenía diferencias debido a que los usuarios utilizan ciertas plataformas y dispositivos para realizar acciones más complejas y otros, como su smartphone, para acciones más rápidas y simples. En ambos desarrollos siempre se buscó darle la mayor simplicidad y facilidad de uso al usuario, pero en el proyecto para dispositivos móviles se buscó darle mayor comodidad sacrificando incluso algunos accesos rápidos o funcionalidades que podría llegar a resultar molestas en una pantalla pequeña.

Ambos aplicativos pueden utilizarse como complementos, realizando acciones en uno y verificando su realización en el otro. Se intentó siempre respetar la estética entre ambos, sobre todo para darle mayor facilidad de uso al usuario que ya había probado o experimentado con la versión web.

Por último, hubiéramos deseado que las notificaciones sean borradas de otra manera (desde la pestaña "Notifications"), implementar fragmentos para no inicializar un nuevo "Intent" en cada pestaña, así como también mejorar la comprobación de errores de ciertas acciones en los dispositivos. Consideramos que son arreglos que podrían ser realizados en caso de que la aplicación fuera a ser actualizada continuamente, como si sucede con las aplicaciones que se encuentran en el mercado.

Anexo

Requerimientos

En **negrita** se marcan los requerimientos que fueron logrados.

La implementación de la aplicación móvil deberá cumplir obligatoriamente con los siguientes requisitos funcionales:

- **RF 1:** Soportar al menos 5 de los tipos de dispositivos disponibles en el API Web (lámpara, puerta, cortina, alarma, heladera, horno, aire acondicionado, temporizador, etc.).
- **RF 2:** Consultar el estado de los dispositivos registrados.
- **RF 3:** Ejecutar acciones sobre los dispositivos registrados.
- **RF 4:** Alertar mediante notificaciones en la barra de estado del dispositivo las novedades que presenten determinados dispositivos registrados que sean de interés.
- **RF 5:** Ejecutar las rutinas registradas.

La implementación del RF 5 es opcional para los grupos conformados por tres integrantes.

Opcionalmente, la implementación de la aplicación móvil también podrá cumplir con los siguientes requisitos funcionales:

- **RFOP 1:** Ejecutar una acción en un determinado dispositivo registrado a partir de un evento generado por la ejecución de una acción en el mismo u otro de los dispositivos registrados.
- **RFOP 2:** Consultar el registro de acciones realizadas sobre un dispositivo registrado.
- **RFOP 3:** Planificar la ejecución futura de acciones sobre dispositivos registrados.

La implementación de la aplicación móvil deberá cumplir obligatoriamente con los siguientes requisitos no funcionales:

- **RNF 1:** Utilizar el idioma establecido en la configuración regional del dispositivo, reaccionar ante cualquier cambio en el mismo y estar localizada por lo menos a los idiomas español e inglés.
- **RNF 2:** Utilizar la barra de acción para facilitar la navegación y el acceso a las acciones más utilizadas.
- **RNF 3:** Permitir la personalización de aspectos relacionados con el funcionamiento de la aplicación.
- **RNF 4:** Brindar una experiencia de usuario **distinta** dependiendo del factor de forma (teléfonos y tabletas) del dispositivo.
- **RNF 5:** Brindar una experiencia de usuario **distinta** dependiendo de la orientación (vertical y horizontal) del dispositivo.
- **RNF 6:** Funcionar en dispositivos *Marshmallow 6.0 (API Level 23)* y *Nougat 7.0 (API Level 24)*.

Opcionalmente, la implementación de la aplicación móvil también podrá cumplir con los siguientes requisitos no funcionales:

- **RNFOP 1:** Capturar con la cámara de fotos códigos QR como mecanismo alternativo de búsqueda de dispositivos registrados.
- **RNFOP 2:** Permitir interactuar en forma automática sobre determinados dispositivos cuando el dispositivo móvil detecte, a partir de sus coordenadas GPS, la cercanía de los mismos. Por ejemplo: que se encienda la luz de la habitación cuando se ingresa a la misma o que se pueda escuchar la misma música al desplazarse por las diferentes habitaciones del hogar.
- **RNFOP 3:** Soportar la conversión de voz a texto (*speech to text*) para facilitar la interacción con los dispositivos registrados.

