

Instituto Tecnológico de Costa Rica

Ingeniería en Computadores

Arquitectura Orientada a Servicios Aplicada a Sistemas Emergentes (CE-5508)

Documentación

Smart Cupboard

Profesor:

Raúl Madrigal Acuña

Estudiantes:

Fabián Astorga Cerdas - 2014040808

David Badilla Sánchez - 2013389091

Reflexión

Temas desconocidos

El desarrollo de aplicaciones móviles mediante el framework React Native fue un objetivo completamente nuevo para la implementación de los requerimientos. Primeramente, hubo que retomar la sintaxis y la manera peculiar para programar en Javascript, HTML y CSS, después de esto, aprender más a profundidad las características y herramientas que ofrece React Native y el porqué de su gran demanda en la actualidad para construir aplicaciones.

Por otro lado, se creó un servidor restful en NodeJS con el framework HapiJS, así como también se diseñó una pequeña base de datos en MongoDB para conectar ambas partes por medio de un ODM llamado Mongoose.

Por último, mediante la implementación del hardware, se utilizó un Raspberry Pi con el fin de manejar la información capturada desde los sensores hasta la aplicación y viceversa. De esta manera se trabajó con sensores que se desconocían, además se reforzó algunas funcionalidades particulares del Raspberry Pi.

Estrategias Implementadas

Con el objetivo de satisfacer el conocimiento para lograr implementar los requerimientos del proyecto, se utilizó Internet como herramienta para la recolección de información. Asimismo, se empezó trabajando en una maqueta para tenerla como diseño físico, tal y como se muestra en la figura 2 y 3, en la sección de Anexos. Luego, se procedió a elaborar el circuito con los sensores y con un raspberry pi montado sobre la maqueta. De esta manera se concluyó con el diseño del dispositivo (véase la figura 1).

Posteriormente, se diseñaron dos pantallas de la aplicación móvil, la que pertenece al inicio de sesión y a la de configuración. Al concluir ambas pantallas, se tenía ahora una mejor noción del funcionamiento del framework React Native. Algún tiempo después, se tenían todas las pantallas de la app hechas, o al menos casi terminadas, y con un Stack Navigator (véase la figura 4) se podía interactuar con todas las pantallas según el funcionamiento establecido de la aplicación.

Finalmente, se requería diseñar un servidor restful para establecer la comunicación entre ambas etapas: aplicación y el dispositivo y sus respectivos sensores. Para ello, en clases se mostró cómo realizar un servidor en NodeJS con el framework HapiJS y cómo con esta herramienta conectarse a una base de datos hecha en MongoDB. Esto permitió que este módulo del proyecto fuese más sencillo de implementar. Una vez terminados los tres módulos, fue posible cumplir con la mayoría de los objetivos.

Utilidad

Se considera de gran valor todo el aprendizaje que se adquirió, ya que en la actualidad de las empresas, se utilizan de manera constante las herramientas que se requerían para este proyecto, por lo que genera una mayor familiarización con respecto a este tema y una mayor confianza al estudiante al momento de ejercer un puesto en una organización.

Por otro lado, se enriqueció el conocimiento en el desarrollo de software y hardware, en ámbitos como el diseño de circuitos, uso de sensores, manejo de la información, creación de aplicaciones móviles y arquitecturas cliente-servidor. Esto es justamente lo que se quiere en una ingeniería, aprender siempre algo nuevo e innovador.

Anexos

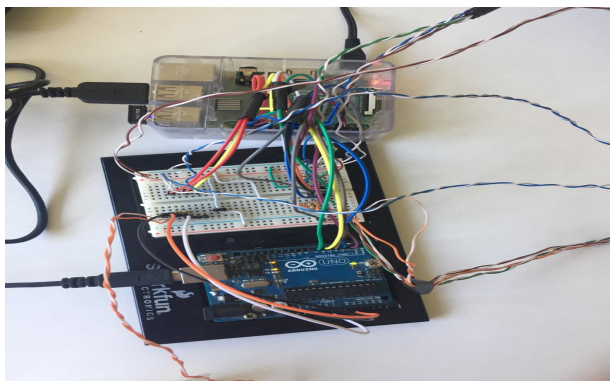


Figura 1. Implementación del circuito del proyecto



Figura 2. Parte interna de la alacena inteligente

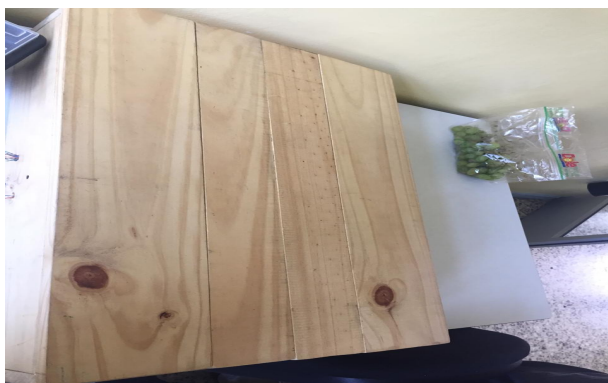


Figura 3. Parte externa de la alacena inteligente

```
1  import { createStackNavigator } from 'react-navigation';
2
3  import Login from './screens/login';
4  import MainPage from './screens/mainPage';
5  import List from './screens/shoppingList';
6  import Preview from './screens/preview';
7  import Configuration from './screens/configuration';
8  import History from './screens/history';
9  import Products from './screens/products';
10 import Names from './screens/names'
11
12 const App = createStackNavigator({
13   Login : {screen: Login},
14   Preview : {screen: Preview},
15   Configuration: {screen: Configuration},
16   MainPage : {screen: MainPage},
17   List: { screen: List },
18   History: { screen: History},
19   Products : { screen : Products},
20   Names: { screen : Names}
21 })
--
```

Figura 3. Stack Navigation para manejo de pantallas de la aplicación móvil