

urbanraspberry-ui

Cali, Colombia, 2015.

Interfaz gráfica que se debe alojar en una Raspberry, en la que el usuario puede configurar los dispositivos y sensores que quiere utilizar para enviar datos a UrbanEyes.

Este código hace parte de un proyecto mayor, que apunta a conectar una Raspberry con un Arduino para recopilar datos sensados en este último y posteriormente enviarlos al API de UrbanEyes. La arquitectura de dicho proyecto es la siguiente:

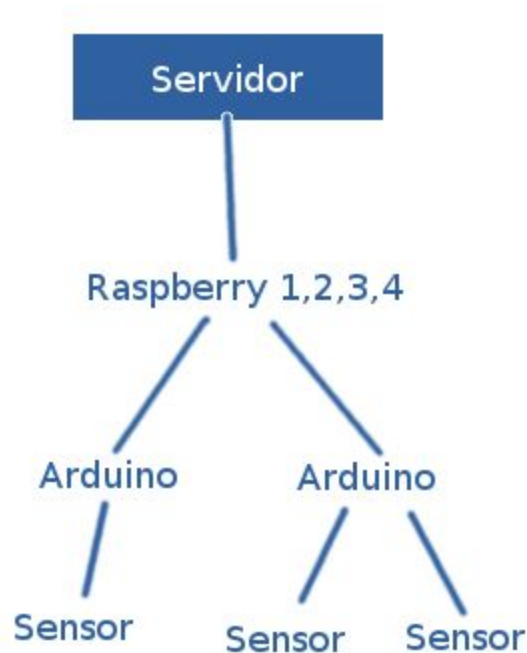


Figura 1. Arquitectura del proyecto para enviar datos a UrbanEyes

En este caso, el presente repositorio se aloja en las Raspberries 1,2,3,4..n

Arquitectura del repositorio

Se tienen las siguientes carpetas:

- *api/*: Carpeta donde se encuentra un servicio web que se encarga de manejar las acciones de la interfaz y se conecta con el servicio web desarrollado en el repositorio "urbanraspberry" para agregar, editar y borrar equipos y variables.
- *templates/*: Carpeta donde se encuentran las plantillas HTML que se muestran en la aplicación
- *lib/*: Carpeta de librerías que se utilizaron
- *js/*: Archivos javascript que definen el comportamiento de la aplicación
- *css/*: Los archivos css propios de la aplicación

Librerías utilizadas

- Bootstrap (css y js): Se encarga de la estética de la aplicación
- JQuery: Añadido como dependencia de Bootstrap
- Angular.js: Librería MVC de Google para hacer aplicaciones web
- UI-Bootstrap: Se encarga de parte de la estética de la aplicación, conectándose con Angular.js

Versiones

v0.0.1

- Contenido sólo en Español
- Se pueden obtener y visualizar los equipos añadidos localmente
- Se pueden crear equipos nuevos con un id especificado
- Se pueden añadir variables a los equipos y establecer los pines en los cuales se quiere que se lea la información
- Se puede cambiar la contraseña
- Se puede hacer login/logout
- Se pueden filtrar los equipos agregados por nombre, nombre de variable, id, pin

Créditos

Desarrollado por:

- Nicolás Escobar Cruz
- Juan David Orejuela
- Juan Camilo Guarín P - twitter: @jcguarinp