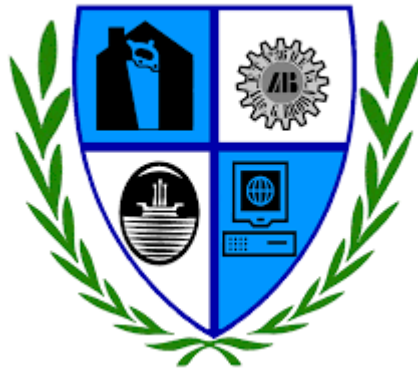


**E.T.N°36 D.E 15 “Alte. Guillermo Brown.”
Galván 3700**



CUE: 20072400

YAKU



Un “ecosistema” automatizado para tu huerta.

Docente Referente: Franco Rodriguez DNI: 43819192

Alumnos: Jeremias Davison DNI: 45234487, Federico Garcia DNI: 45747650

14/09/22

Índice

Resumen	2
Introducción	2
Desarrollo	3
Resultados obtenidos	4
Discusión	7
Conclusiones	7
Bibliografía consultada	7
Agradecimientos	8

Resumen

Yaku es un proyecto, que plantea la idea de realizar una huerta totalmente automatizada capaz de activar el riego en forma automática en función de la humedad en tierra y la cantidad de luz solar, permitiendo establecer los valores mínimos y máximos aceptables para comenzar el riego.

A su vez tiene un depósito donde recolecta agua de lluvia para hacer un uso eficiente de este recurso indispensable. Ya que en caso de que el depósito de agua de lluvia se encuentre vacío, el sistema automáticamente empezará a utilizar agua de la red.

Además cuenta con paneles solares permitiéndole reducir su consumo energético de la red.

Los datos obtenidos por los sensores, son reportados a un servidor en la nube permitiendo un monitoreo del estado actual del sistema y además consultar los valores históricos con el fin de obtener conclusiones sobre el cultivo. Dichos valores, son almacenados en una base de datos utilizando la API que nos ofrece el servicio Rainmaker, que es ofrecido por Espressif, empresa desarrolladora de la placa que utilizamos para el dispositivo.

Cuenta con una página web en la que se puede ver a detalle todo el contenido que abarca el proyecto, y revisar una galería de fotos del progreso de armado y evolución.

Tiene también, una aplicación de celular (Android), asociada a ESP Rainmaker en la que se consulta el estado actual o más reciente de cada uno de los sensores que contiene el dispositivo, también se puede ver el historial de los datos recabados de temperaturas, humedad, mediante consultas por lapsos de tiempo, permite habilitar / deshabilitar el riego de forma manual y establecer nuevas condiciones para el riego.

Además, cuenta con una pantalla OLED que nos indica una versión reducida de los datos, para verlo desde el mismo dispositivo, en caso de no contar con la aplicación.

Aprovechando la central con la que cuenta para recabar cierta cantidad de datos, contiene una estación meteorológica, la cual proporciona valores de temperatura, altura y presión atmosférica.

Es OpenSource, lo que significa que está libre para que cualquier persona pueda utilizar el código y adecuarlo al uso que prefiera, haciéndole modificaciones/mejoras, además de una muy completa documentación para una más sencilla implementación.

Introducción

Nuestra problemática fue el hecho de lo demandante que es tener que estar pendiente de los bienestar de una huerta, y más si uno no puede estar presente en su espacio y de sus necesidades. Analizando nuestro nuestro polo educativo vimos que nuestro proyecto se adecuaba mucho en nuestro colegio vecino "Escuela de Discapacitados Motores N° 1 D.E 15", teniendo en cuenta que a nosotros se nos hacía una dificultad mantener una huerta en condiciones, pensamos que para los chicos que concurren a ese establecimiento iba a ser aún más efectivo la utilización de nuestro dispositivo ideado. También teniendo en cuenta el poco cuidado que se tiene en épocas de receso escolar (debido a la ausencia de personas en el colegio), YAKU, permitiría tener un control de la huerta remotamente, y al ser automático, no tendría que acercarse nadie.

Utilizando las horas de las materias Proyecto Informático I, Proyecto Informático II y Prácticas Profesionalizantes, aprovechamos que la metodología de estudio era libre por lo

que pudimos abarcar varias tecnologías y formas de trabajo acordes a las necesidades que requería este proyecto. Se comenzó con una placa Arduino y una página web sin hojas de estilo CSS, y más adelante, se rediseñó la idea, pensando aún más en grande, y es por eso que decidimos concurrir a varios cursos para plantear nuestra idea y poder contar con más personas e ideas para llevar a cabo el trabajo, no solo por la necesidad de implementarlo sino por las ganas de que otros alumnos puedan conocer nuevas tecnologías y trabajar en equipo. Nos dividimos en equipos de trabajo para ocupar cada área y nos conectamos por medio de Trello para organizar cada sección y sus correspondientes tareas.

Desarrollo

Nuestra idea del proyecto fue dividirlo en varios grupos los cuales no fueron capacitados sino que fueron incentivados a un aprendizaje autodidacta, estos grupos tenían un referente que se reunía todas las semanas con profesores y alumnos para dar un resumen de los avances de cada etapa y de las tareas en las cuales estaban estancados para darles una guía y feedback de los resultados obtenidos.

Además utilizamos el software Trello para organizarnos en nuestras tareas y etapas del proyecto, allí se plantearon periódicamente las tareas de cada grupo, se fueron marcando como por hacer, en progreso, pendiente de revisión, terminado, según correspondiese. Esto nos ayudó para tener un historial del proceso, y como herramienta para compartir recursos.

Sumado a eso, utilizamos GitHub para compartir los archivos entre los miembros de los grupos y mantener el proyecto actualizado en cada modificación ya que sino se generaba una acumulación de programas y archivos en distintas computadoras y no era efectivo para la organización de los equipos.

Otro método que utilizamos fue un grupo de WhatsApp en el que solo podían enviar mensajes los administradores (constituidos por el grupo de organización del proyecto), donde mantenían al tanto a todos de las próximas reuniones, encuentros, informaciones, etc.

Finalmente como método de comunicación se creó un servidor de Discord en el cual cada uno podía interactuar más verbalmente con los miembros de distintos grupos y mediante también se envió información.

Los grupos en los cuales fue dividido el proyecto fueron:

Bot de Rainmaker - Python: Acceder a los datos que provee Rainmaker mediante su API para obtener los datos de cada sensor/dispositivo en ese momento y almacenarlo en una base de datos MongoDB para que después pueda ser consultada y armar las métricas del estado de la huerta. El bot se va a colocar en algún servicio como AWS para que corra en la nube de forma constante.

Aplicación de Android - Java: El objetivo es hacer una aplicación interactiva que nos provea los datos de Rainmaker para controlarlos y leerlos desde nuestros dispositivos. Utilizando el programa Android Studio se diseña la aplicación con sus botones e items.

Desarrollo del firmware ESP32 - C: Desarrolla el programa (software) que controla el hardware de ese dispositivo. Este equipo tiene como objetivo desarrollar el programa que irá en nuestra placa ESP32 para leer datos de sensores, y accionar una bomba de agua para el riego de la huerta.

Instalación y puesta en marcha de la huerta: Conexión de sensores y actuadores en la placa ESP32. Ayudan a los que desarrollen el firmware de la ESP32 para que conecten correctamente los dispositivos a la placa. Diagrama en bloques del proyecto y lista de materiales.

Diseño gráfico - Lightroom: El objetivo principal de este grupo es realizar diagramas para el proyecto, tanto para su interfaz web y de la aplicación como también para documentar el proyecto y diseñar el logo con su paleta de colores. Los diagramas e interfaces fueron diseñados en Figma y Lightroom siguiendo normativas correctas de UI/UX.

Página web - HTML5/CSS: Se realiza una página interactiva para que pueda verse información del proyecto, del dispositivo y métricas sobre su estado actual. Esto se realizará utilizando los datos recopilados con el bot que se encuentra en la nube.

Traducción: Traducción la app a español, gran parte de la documentación que es común a todos los equipos Rainmaker Docs, API está en inglés. El objetivo de este grupo es servir de soporte respondiendo dudas sobre la interpretación de la documentación que pueda necesitar un miembro del equipo.

Materiales: Se utilizaron para el desarrollo del proyecto paneles solares, placa ESP32, sensores de humedad en tierra, sensor de luminosidad, bomba de agua, sensor de humedad/temperatura. Teniendo en cuenta que nuestro colegio técnico, tiene una especialidad de construcciones, por eso vamos a trabajar en conjunto para que se haga un bancal de madera para que se aloje la huerta y nuestro dispositivo.

Resultados Obtenidos:

Decidimos subir los archivos realizados en un repositorio de GitHub en el cual se ve todo lo hecho hasta el momento categorizado por grupos (teniendo en cuenta que hay cosas que no pueden dejarse en evidencia, ej. armado físico con las placas, reuniones, etc.), los grupos que pudieron subir algún tipo de contenido son los siguientes: Bot de Python, Diseño Gráfico, Traducción y Android. También contamos con bocetos y diagramas de flujos de la instalación de la huerta.

Tareas Realizadas:

- **Bot de Python:** Se consiguieron los nodos sobre los cuales hay que realizar la consulta de la API de ESP. Se subieron los datos a la Base de Datos no relacional. Se enviaron a la página web en Flask.
- **Diseño:** Logo de Yaku, paleta de colores, diseño de la página web, armado del fondo de pantalla para la aplicación de celular. Como también se realizó un diagrama de cómo debería funcionar el proyecto y su implementación.
- **Traducción:** Se tradujo toda la documentación de ESP. Se tradujeron variables de tipo string que venían con la aplicación base proveída por Espressif
- **Aplicación Android:** Se hicieron los respectivos logos de la app, se descargaron las fuentes de ESP Rainmaker, se hizo una investigación del código realizado para Android.
- **Organización:** Se encargó de nuestro correcto funcionamiento, organizando reuniones en las que se pudo discutir inconvenientes, sacar dudas, sugerir nuevas ideas o modificaciones, compartir contenidos y progresos, etc.

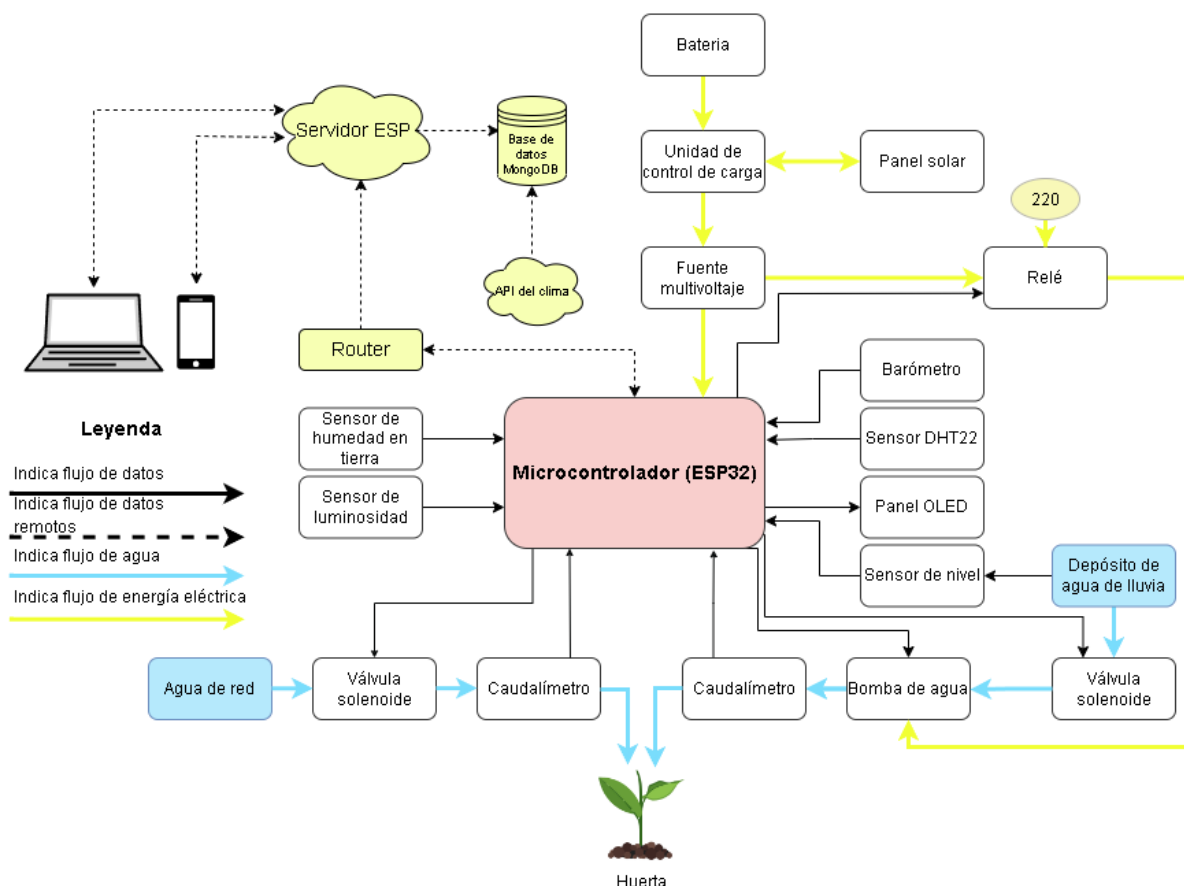
- **Firmware ESP32:** Se tomaron datos de todos los sensores necesarios, se subieron a ESP Rainmaker, y se vieron reflejados en la aplicación los valores de los mismos. Se realizó una estación meteorológica.
- **Instalación y puesta en marcha:** Se conectaron a modo de prueba todos los sensores. Se hicieron las conexiones a modo de prueba de los paneles solares. Se comenzaron a realizar las soldaduras correspondientes.
- **Página web:** Se completó toda la página web por completo.

En proceso de la realización y finalización:

- **Instalación y Conexión de la Placa - ESP32:** Estamos en proceso de realizar las debidas conexiones de la placa en una plaqueta electrónica y las conexiones eléctricas. Seguido a eso se realizará la instalación en el colegio vecino, colocando las bombas de agua (las cuales ya cuentan con el código hecho) y los paneles solares (los cuales ya fueron testeados).

Adjuntamos imágenes y link:

Diagrama de bloques del funcionamiento de Yaku:



Datos subidos a la Base de Datos de Mongo DB (son tomados de la API de Rainmaker):

```

_id: ObjectId("63227db1b32d7e9fd8c98f0b")
Soil Moisture: 2863
LDR: 2824
Pressure: 1023.62097
Humidity: 40.5
Temperature: "22.2°C"
Height: "86mts"
Relay: true
Wind: "3.6m/s"
Clouds: "0%"
Time: "14 Sep 2022, 22:19"
Timestamp: 1663204783.790964

```

Diseño del Logo y su paleta de colores:



Link: <https://github.com/jeremiasdavison/Yaku-ET36>

Discusión

Creemos que en comparación a otros proyectos parecidos que se hicieron en el colegio o fuera de él, nosotros implementamos varias secciones con distintas tecnologías para hacer en gran escala esta idea e inculcar un desarrollo saludable y ecológico.

También traducimos la documentación para su entendimiento en el inglés. Podemos recalcar que nuestro proyecto está implementado en los servidores de AWS y Rainmaker por lo tanto tenemos un alojamiento las 24 hs y podemos ver su estado en el momento que queramos desde una aplicación o página web.

Es importante remarcar que el fin de éste es ayudar a nuestro colegio vecino, cuyos alumnos tienen dificultades motrices, y que les ayudará en gran escala.

Reutiliza el agua de lluvia, filtrándose y haciendo que no haya desperdicios de la misma.

Tiene la particularidad de que los parámetros que evalúan si se debe regar o no, son modificables por el usuario, lo que hace que sea adaptable a cualquier planta/huerta y no sea restringida al caso de uso que nosotros le estamos dando.

La ventaja de poder monitorear el regado desde una aplicación de celular, logra que no haya dificultades a la hora del cuidado de una huerta en épocas de vacaciones, sea de invierno o verano.

Conclusiones

Se comenzó con una idea básica para reducir el tiempo de cuidado necesario para una huerta, que más tarde se enfocó para ayudar a los chicos del colegio vecino de nuestro polo educativo, y terminamos desarrollando todo un ecosistema, que incluye una página web, una aplicación, una base de datos, entre muchas otras cosas ya mencionadas.

Se realizó el aprendizaje de todos los contenidos necesarios para estas áreas, que comenzaron siendo nulos, y también se fomentó totalmente el trabajo en equipo, la comunicación y el aprendizaje autodidacta.

Se fueron implementando constantemente nuevas tecnologías e ideas, no solo para mejorar el proyecto, sino también para poder incluir a nuevos alumnos a que se sumen a la experiencia del trabajo y desarrollo en equipo. Además podemos sumar la incorporación de distintas formas de organizarse con páginas y herramientas que se utilizan en el mundo laboral real.

Bibliografía consultada

Se consultaron distintas páginas web con el contenido necesario, documentación, etc.

Documentación ESP Rainmaker.

<http://rainmaker.espressif.com/docs/get-started.html>

Repositorio de GitHub del cual se comienza a realizar el desarrollo de la aplicación de Android.

<https://github.com/espressif/esp-rainmaker-android>

Video sobre la estación meteorológica que se utilizó:

https://www.youtube.com/watch?v=LMTtIC2jKUg&ab_channel=IngenieroenCasa

Video utilizado para subir los datos a ESP Rainmaker

<https://www.youtube.com/watch?v=651EoGQHWck>

Documentación de MongoDB.

<https://www.mongodb.com/docs/>

Documentación de la API de ESP.

<https://swaggerapis.rainmaker.espressif.com/>

Documentación de Trello.

<https://blog.trello.com/es/como-usar-trello>

Manual del usuario

Dirección General de Administración y Gestión Financiera Sistema de Transferencia de Recursos Educativos (SITRARED) Ejecución y Rendición Escuelas.

Agradecimientos

Este proyecto tuvo muchas etapas y procesos por los cual estuvieron involucrados muchas personas, entre ellas encontramos a:

Sebastian Schachtner - Profesor de Proyecto Informatico I y Proyecto Informatico II.

Franco Rodriguez - Alumno precursor del proyecto, el cual se convirtió en profesor este último año y brindó el espacio de la materia Prácticas Profesionalizantes.

Mariano Zalazar - Alumno de 6to año que guió y organizó partes importantes de la estructura del proyecto, y siendo responsable del grupo de Python en una etapa.

Maximiliano Lopez y Daniel Farias - Mentores de Intel.

Martín Oliva - Profesor del colegio vecino.

Asociación cooperadora ET 36

Cabe mencionar que muchos alumnos estuvieron involucrados, y que éste proyecto no hubiera podido llevarse a cabo sin ellos.

Federico Garcia - Parte del equipo de organización del proyecto, del desarrollo en Python, desarrollo ESP32, instalación y puesta en marcha, ideación de nuevas mejoras.

Jeremias Davison - Parte del equipo de organización del proyecto, del desarrollo en Python, desarrollo ESP32, instalación y puesta en marcha, ideación de nuevas mejoras.

Tomás Fernandez - Responsable del equipo de ESP32, realizando gran parte de las conexiones y código.

Nicolas García Rial y Julian Ippolito - Miembros del equipo de ESP32

Candelaria Clariá - Parte del equipo de organización por un periodo de tiempo, responsable del equipo de Traducción.

Macarena Fernandez y Daj Lautaro - Responsables del diseño gráfico.

Lihuel Troncoso - Responsable de la aplicación móvil.

Ailén Pestaña - Responsable del equipo de Desarrollo Web.

Federico Gutierrez, Tomas Enriquez y Pedro Bueno Pereyra - Equipo de Desarrollo Web primer grupo.

Lucas Barreto - Responsable final de realización de la página web.

