

# Diseñar un sistema para el control de acceso a la propiedad horizontal a través de huella (Julio de 2021)

Zaraza David  
[Jdavidzt9003@gmail.com](mailto:Jdavidzt9003@gmail.com)  
Fundación universitaria del Área Andina

**Resumen** - El presente trabajo está diseñado para el control de acceso a la propiedad horizontal por medio de sensor de huella. Se mostrará el proceso que se llevó a cabo para el diseño y desarrollo, mostrando los resultados obtenidos, fuentes consultadas. La idea principal esta basada en controlar el ingreso a espacios autorizados mediante la huella digital. Por otro lado, se pretende que la aplicación no conceda acceso y emita alerta cuando un visitante intenta acceder a un espacio no autorizado.

## Índice de Términos - Control de Acceso por Sensor de Huella

### I. INTRODUCCION

El desarrollo y programación de software permite generar solución a los problemas planteados desde los diferentes sectores de la economía mundial. Python, es un lenguaje que permite adaptar la codificación a diferentes necesidades, mediante sus frameworks y librerías que nos ayuda a minimizar código y generar un mejor performance a nuestras aplicaciones.

La combinación de varios elementos, como lo son las redes o devnet, nos permite generar soluciones a problemáticas mediante el internet de las cosas, inteligencia artificial, análisis de datos y todas estas ramas un lenguaje que nos facilita su desarrollo es Python.

Para dar una definición de lo que es Python, es un lenguaje de programación que permite trabajar más rápidamente e integrar sus sistemas de manera más efectiva [1]. Es un lenguaje interpretado, lo que quiere decir que no necesita de un compilador para ser traducido a lenguaje máquina.

El presente trabajo tiene como objetivo poner en práctica los conocimientos adquiridos durante el diplomado de grado Python-Devnet, mediante la investigación y desarrollo de MVP que muestre la utilidad de un sensor de huella para controlar el acceso a las diferentes secciones de un edificio o propiedad horizontal.

### II. PROCEDIMIENTO

#### A. Desarrollo del diplomado

El diplomado Python, desde su primer día nos introdujo en conceptos básicos a nivel de lógica de programación y en la sintaxis del lenguaje. A la explicación del instructor se le sumo los ejercicios de la plataforma <https://www.netacad.com/>, recorriendo temas tales como operadores, ciclos o bucles, variables, funciones, conexión con base de datos, creación de archivos y contenidos. Luego pasamos al tema de redes, en el cual se combino una serie de elementos tales como protoboard, led, led RGB, jumpers, esp32 Micropython, sensor de humedad y temperatura. Esta codificación se desarrolló en IDE Thonny, sugerido por el instructor para la ejecución de los scripts.

#### B. Etapa de proyecto

El proyecto fue indicado desde la primera sesión. En ese momento los sábados en sesión de asesoría el tutor, daba pautas para el desarrollo del presente trabajo. La idea planteada es la actual, el control de acceso mediante la huella dactilar de la persona.

### Manual de Usuario

El proceso es el siguiente:

1. Registrar al usuario mediante la interfaz gráfica, capturando los datos básicos como: Tipo Documento, No. Documento, Nombres, Apellidos, Correo, Celular.

Imagen 1: Pantalla index.html

Nota: Elaboración propia

- En una segunda pantalla se asignará el piso y oficina a visitar, capturando datos tales como: Tipo Documento, No. Documento, Piso y Oficina.

Imagen 2: Pantalla Register.html

Nota: Elaboración propia

Al registrar los datos de la pantalla 1 y pantalla 2, se concederá al acceso por huella digital a los lugares que fueron asociados en la pantalla 2.

Si un usuario no tiene acceso a un espacio e intenta acceder a este, se le notificará mediante correo electrónico al administrados y sistema de seguridad que este usuario intenta acceder a un lugar no permitido.

```
pip install secure-smtplib
```

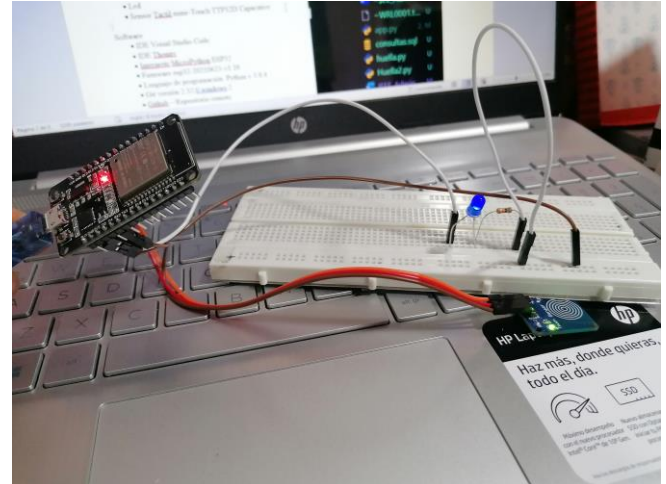
## Manual Técnico

### Hardware:

- Protoboard

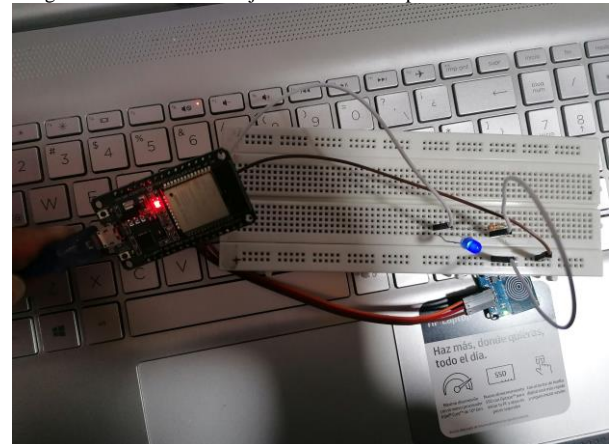
- Jumpers MM-MH-HH
- Cable MicroSD a USB
- Esp32 WROOM 32
- Led
- Sensor Tactil mini-Touch TTP32D Capacitive

Imagen 3: Estructura montaje circuito vista frontal



Nota: Elaboración propia

Imagen 4: Estructura montaje circuito vista superior



Nota: Elaboración propia

### Software

- IDE Visual Studio Code
- IDE Thonny
- Interprete MicroPython ESP32
- Firmware esp32-20210623-v1.16
- Lenguaje de programación: Python v 3.9.4
- Git versión 2.32.0. windows.2
- Github – Repositorio remoto
- XAMP

### Librerías Python:

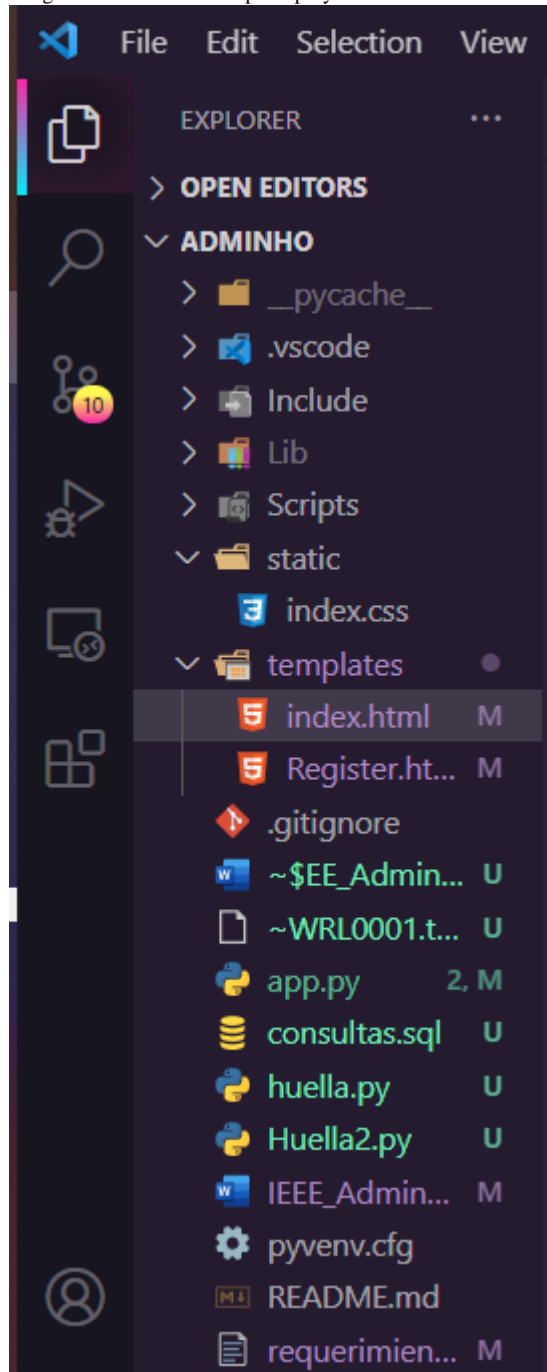
- Flask 2.0.1
- Flask-MySQLdb 0.2.0
- Mysqlclient 2.0.3
- Mysql-connector 2.2.9
- Secure-smtplib 0.1.1

Base de datos:

- Mysql → adminho

Estructura de carpetas

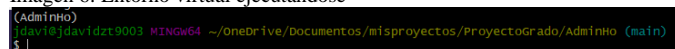
Imagen 5: Estructura de carpetas proyecto



Nota: Elaboración propia

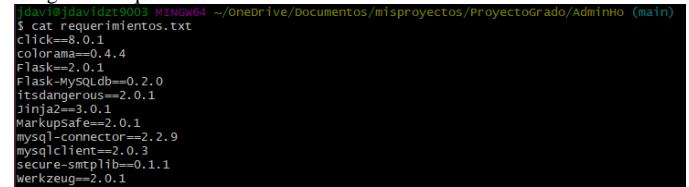
Creación del entorno virtual Python -m venv AdminHo

Imagen 6: Entorno virtual ejecutandose



Archivo de requerimientos.txt

Imagen 7: Requerimientos.txt



Nota: Elaboración propia

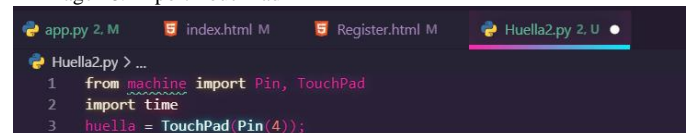
Conexión con el ESP32 y Sensor TTD32B  
Los pines son los GPI y touch sensor.

GPI04 → Touch0  
GPI02 → Touch2  
GPI015 → Touch3  
GPI013 → Touch4  
GPI012 → Touch5  
GPI014 → Touch6  
GPI027 → Touch7  
GPI033 → Touch8  
GPI032 → Touch9

El Sensor TTD32B cuenta con tres pines de conexión

SIC → conectado a un GPIO  
VCC → Conectado al pin vdd v3v ESP32  
GND → Conectado al pin GND (-)

Imagen 8: Import TouchPad



Nota: Elaboración propia

### III. CONCLUSIONES

#### A. Lecciones aprendidas

Al inicio del diplomado mi nivel en Python era un 5% en general, más en conocimientos de conceptos. Con la finalización del diplomado y el avance del entregable en podido engrosar este porcentaje al 45% en este lenguaje, considerando que para llegar un avance mayor debo seguir practicando sobre proyectos personales o laborales.

Otro avance importante fue conocer el tema de redes y circuitos, ya que no ha sido mi fuerte durante mi carrera, y considero que la práctica con los elementos electrónicos adquiridos me permitió si bien no se llega abordar todo si se consiguió un avance del 25%.

El antes y el después se puede resumir en conocimiento, fue una experiencia enriquecedora y ahora solo queda continuar trabajando en este proyecto ya iniciado.

### B. Problemas detectados

La documentación para trabajar el código para el tema de desarrollo de detección de huella no es muy amplia en la web. El proyecto no ha podido tener el avance esperado, he divagado en varios espacios de la web sin encontrar en algo concreto que me oriente al desarrollo de la solución.

### IV. PROYECCIONES A TRABAJOS FUTUROS

La idea principal era desarrollar un proyecto con tres sensores biométricos. De acuerdo con el alcance, se definió que se trabajaría con el sensor de huella, para el control de acceso. Como lo mencioné al inicio del párrafo, la idea es seguir desarrollando la aplicación y volver esto en un producto comercializable, para ello seguiré trabajando en incorporar a la solución el registro y acceso biométrico mediante reconocimiento facial y chip. Pretendo desarrollar un Api e incorporar nuevas funcionalidades cuyo objetivo sea ofrecer el servicio completo de administración en la propiedad horizontal.

En mi repositorio <https://github.com/jdavidzt9003/AdminHo>

Si llega haber un cambio de repositorio en este mismo se dejará el enlace para consulta del nuevo.

### V. COMENTARIOS

Se recomienda usar editores de texto adecuados para poder ejecutar el resultado y tengan enlace con MicroPython ESP32, tal como lo es Thonny. Un IDE que no conocía que me ha parecido muy completo para la elaboración del producto.

### RECONOCIMIENTO

Me permito dar un reconocimiento especial al Ing. Amaury Méndez, quien fue el mentor o tutor durante el avance de todo el diplomado, y al igual fue quien nos compartió sus conocimientos para el desarrollo del proyecto. Un agradecimiento a la universidad Fundación Universitaria del Área Andina, por abrir la puerta a este nuevo diplomado para enriquecer nuestra experiencia profesional.

### REFERENCES

- [1] Applications for Python. (s. f.). Python.Org. Recuperado 30 de julio de 2021, de <https://www.python.org/about/apps/>
- [2] Negi, A. (2021, 26 marzo). How to use ESP32 Touch and Hall effect Sensor with Arduino IDE. ETechnophiles. <https://www.etechnophiles.com/controlling-esp32-touch-sensor-and-hall-effect-sensor/>
- [3] Touch Sensor - ESP32 - — ESP-IDF Programming Guide latest documentation. (s. f.). Docs.Espressif.Com. Recuperado 30 de julio de 2021, de [https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/touch\\_pad.html#](https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/touch_pad.html#)
- [4] Quick reference for the ESP32 — MicroPython 1.16 documentation. (s. f.). Docs.Micropython.Org. Recuperado 30 de julio de 2021, de <https://docs.micropython.org/en/latest/esp32/quickref.html#capacitive-touch>

**Biografía Autor(es)** Estudiante de Ingeniería de Sistemas V semestre y Tecnólogo en informática años 2018. Tengo 31 años. Inicie mis labores en el área Tech alrededor de hace 5 años, desempeñándome en diferentes cargos y áreas como Base de datos (Analista), Soporte técnico nivel I y nivel II, implementador, Analista de QA y automatizador. A nivel técnico, he tenido la fortuna de manejar lenguajes como Javascript para desarrollo web, Java para automatización y ahora Python para el backend. Base de datos tales como: Oracle 19 y 12, SQL Server 2018, MySQL, PostgreSQL.