

25/2/25

Proyecto control de luces LED con API REST

* Cosas a Hacer (Primera org.)

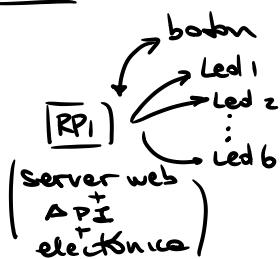
- Frontend web (html, css, js) + mobile
- Backend API REST
- control LEDs
- Documentación, instalación

* Descripción features

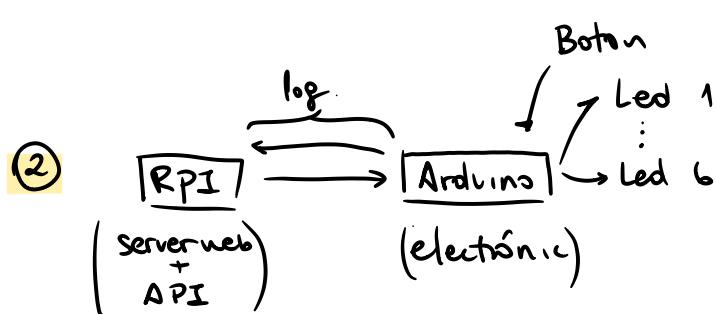
- Controlar los efectos de LED
- Controlar velocidad de efectos
- Encender / Apagar todos efectos (LEDs)
- DB Logs de acciones

Ideas

①



②



① Raspi 4

- ↳ usar threading
- ↳ reescribir códigos
- ↳ producto final pequeño + menor voltaje

② Raspi 4 + Arduino

- ↳ Reutilización de códigos (Arduino LEDs)
- ↳ Implementación IDEA ORIGINAL
- ↳ producto final grande y usa más voltaje
- ↳ Separación de servidor y controlador

* ¿Comunicación entre Raspberry y Arduino?

- ↳ PINES → Pines de entrada/salida (normales)
- ↳ Señal → Cable USB o pines TX/RX

} cableados físicos

- ↓
tiene que usar pins normales para simular entrada de un botón en pull off
- ↳ RPi → pin.HIGH (OUTPUT)
 - ↳ Arduino → pin.INPUT

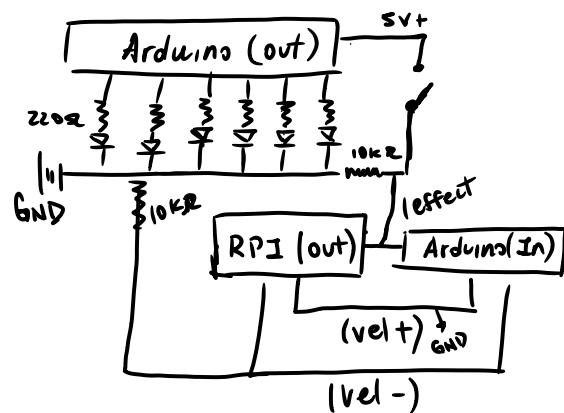
↓
añadir resistores

enviar una señal cada vez que se quiere cambiar de efecto

(BOTON VIRTUAL)

ELECTRÓNICA

Diseño de Circuito



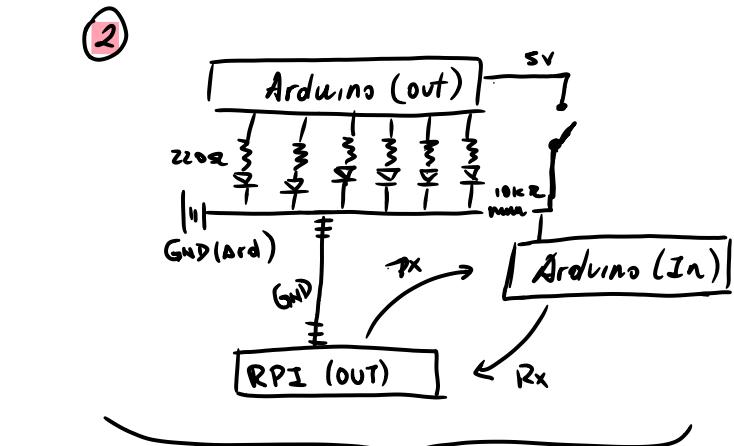
* Pines a Utilizar

| | | |
|--------------|-------------|-------------|
| Led 1 - p 11 | Led 4 - p 6 | Boton - p 2 |
| Led 2 - p 10 | Led 5 - p 5 | +vel - p 4 |
| Led 3 - p 9 | Led 6 - p 3 | -vel - p 7 |

(out)

Cosas a hacer (26/2/25)

- Hacer diagrama ✓
- Armar circuito físico ✓ emplazar C'd
- Limpiar código (reutilizar) and ✓
- Actualizar código ⇒ Add effectVel Count ✓
- Agregar función Cambio Vel ✓
- Conectar debug and + Rpi ✓

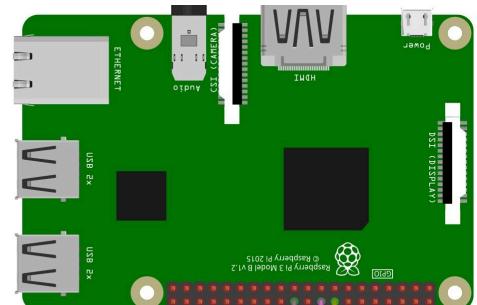


Problema = muy complicado la comunicación

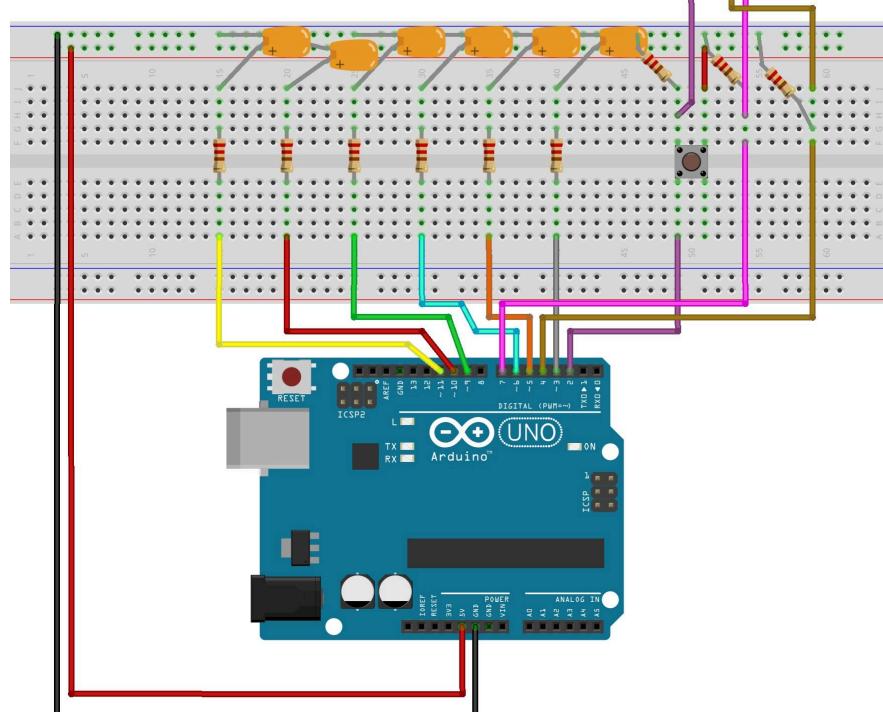
Senal → experimentar pinos

Rpi (out)

- Control efecto GPIO 25
- Control +vel GPIO 24
- control -vel GPIO 23



+ pinout diagram



* Especificaciones software 27/2/23

- Rango velocidades → empezar por la velocidad del medio
↓
→ 10 (9) velocidades en total (niveles)

- + vel → - debounce
- vel → + debounce (effect)
- cooldown 10-30 ms (debounce BOTON)
- + vel (p4) / - vel (p7)
- pin SpeedUpButton / pin SpeedDownButton
- check Speed Buttons State() → Recibe una sola señal (arduino)
- La cant de señal recibida la define Rasp Pi
- calcular cant señal a enviar

* Raspberry Pi → (server) [server API | server web]

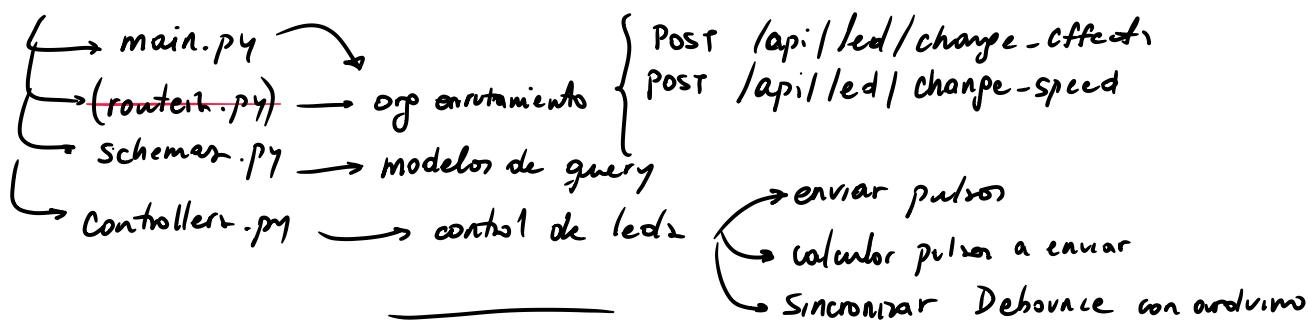
Estructura de carpetas

raspberry-pi /

- : web /] → HTML + JS + CSS (generar con IA)
- : API /] → control led + API routing → Fast API y Python
- Scriptz / → Scripts auxiliares de prsche

* API REST

api /



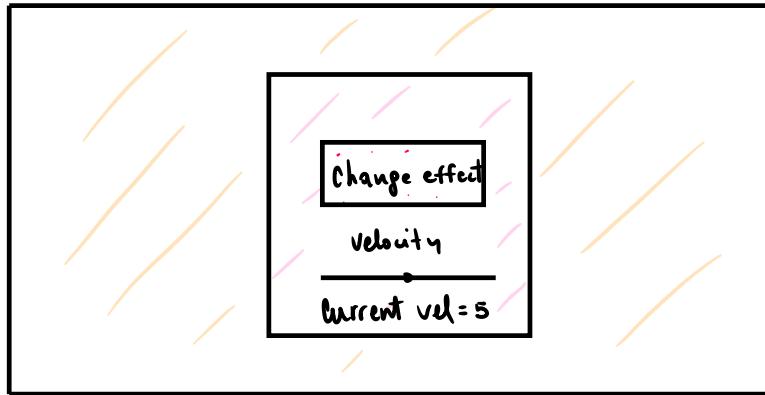
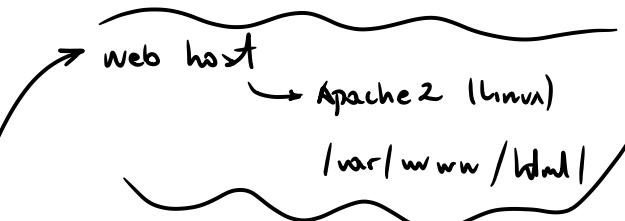
POST Body examples

```
{  
    value: 100 ms.  
    (int)  
} ] Led Speed  
      ⇒ response {  
        current_speed: int  
    }
```

28/2/2025

* Diseño Frontend (idea)

↓ Proporcionar descripción y modificar con IA



Descripción general

- página simple de botón y barra para hacer un POST a un API, para controlar luces LED realizada
-
- Características / features
 - Ambos botón y barra con cooldown (crear animación) ✓
 - Animaciones de Botón al tener mouse arriba y en box también ✓
 - Barra tiene nivel de velocidad 1 al 9, comienza en 5 \Rightarrow No dejar espacios en medio
 - Fade en el fondo afuera del Bot + color pastel ✓
 - Devolver en pantalla velocidad actual ✓ → hacer un fetch de velocidad al inicio.
 - Adaptación dispositivos móviles
- cuidado si se tiene en conexión varios dispositivos "todos comienzan en 5 se desync"

3/3/2025

Conveniones \Rightarrow Respetar convenciones POST

- nombre convencional URL API
- POST con Body
- return de POST en JSON

Cosas a hacer

- corregir convenciones URL ✓
- Agregar comentarios + chequear códigos
- Añadir función Barra deslizadora función Arreste AL SOLTAR ✓
- Agregar ICON + mejora usuario \Rightarrow + CSS
- Hacer scripts de deploy, undeploy ✓

- 4/3 * pendientes ① Cerrar issues ✓
- Ordenar cód JS + comments ✓
 - Eliminar códigos debug ✓
 - archivar setup + start
 - Documentación
 - + Features

→ setup.

↳ setup web

- instalar (actualizar) apache 2
- mover archivos web a /var/www/html

↳ setup API → inst o actualizar python3

- entrar carpeta ./api
- Crear virtual environment → (entorno)

- instalar paquetes python
 - ↳ fastapi
 - ↳ RPi.GPIO

- Cerrar virtual environment

↳ setup ./start.sh → cambiar permisos de ejecución

start

↳ entrar venv

↳ start web apache2

↳ start API server

↳ cerrar web apache2

↳ salir venv

* DOCUMENTACIÓN

↳ Readme general

- Descripción general
- Estructura proyecto
- Materiales
- Requisitos
- Pasos de ejecución del proyecto
- Más información (mentionar planificador)

*fotos
Demos video*

↳ Readme Arduino

- Descripción y rel Arduino
- Cómo cargar código a microcontrolador
- Cómo encastrar componentes físicos
- Líneas de cambio de código

explicar cambio de velocidad

↳ Readme raspberry-pi

- Descripción general (qué hace = Servers)
- Instrucciones instalación
- Conexión Hardware
- Estructura proyecto
 - ↳ web/
 - ↳ api/
- Ejecución

mentionar que use IP LOCAL

*↓
Cambiar IP local url API*