



LBB Exit Detection System | 2024

LASER-BASED BED EXIT DETECTION SYSTEM

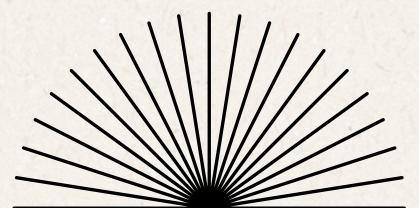
Non-invasive care for every need

DATE:

Dec. 9th 2024

PRESENTED BY:

Tammy Dahl

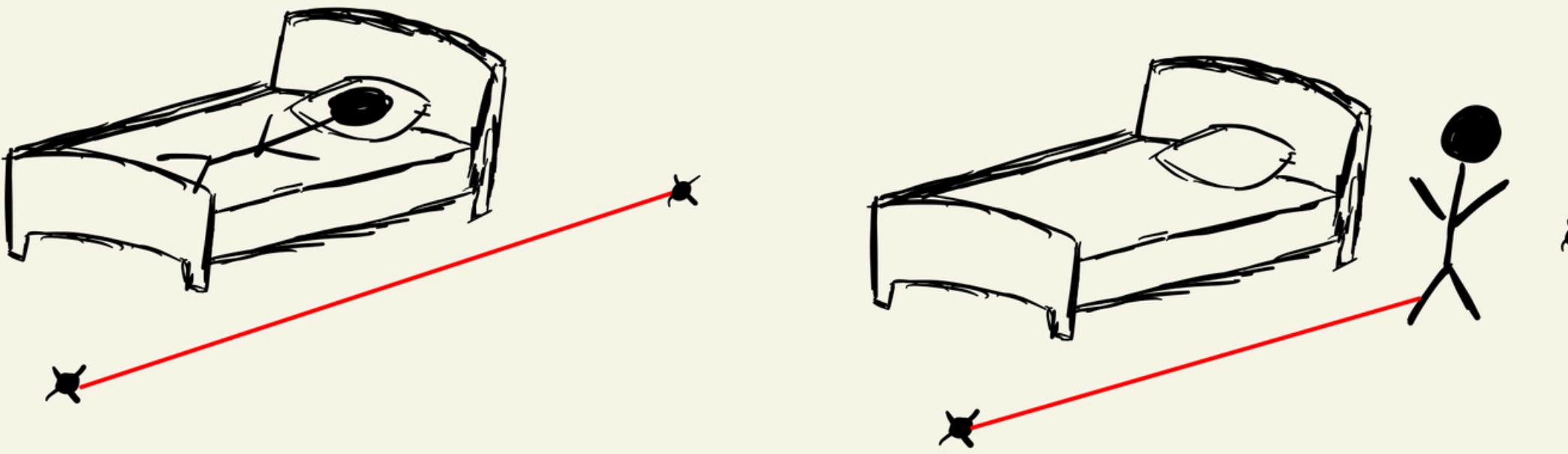


Project Overview

Reminder

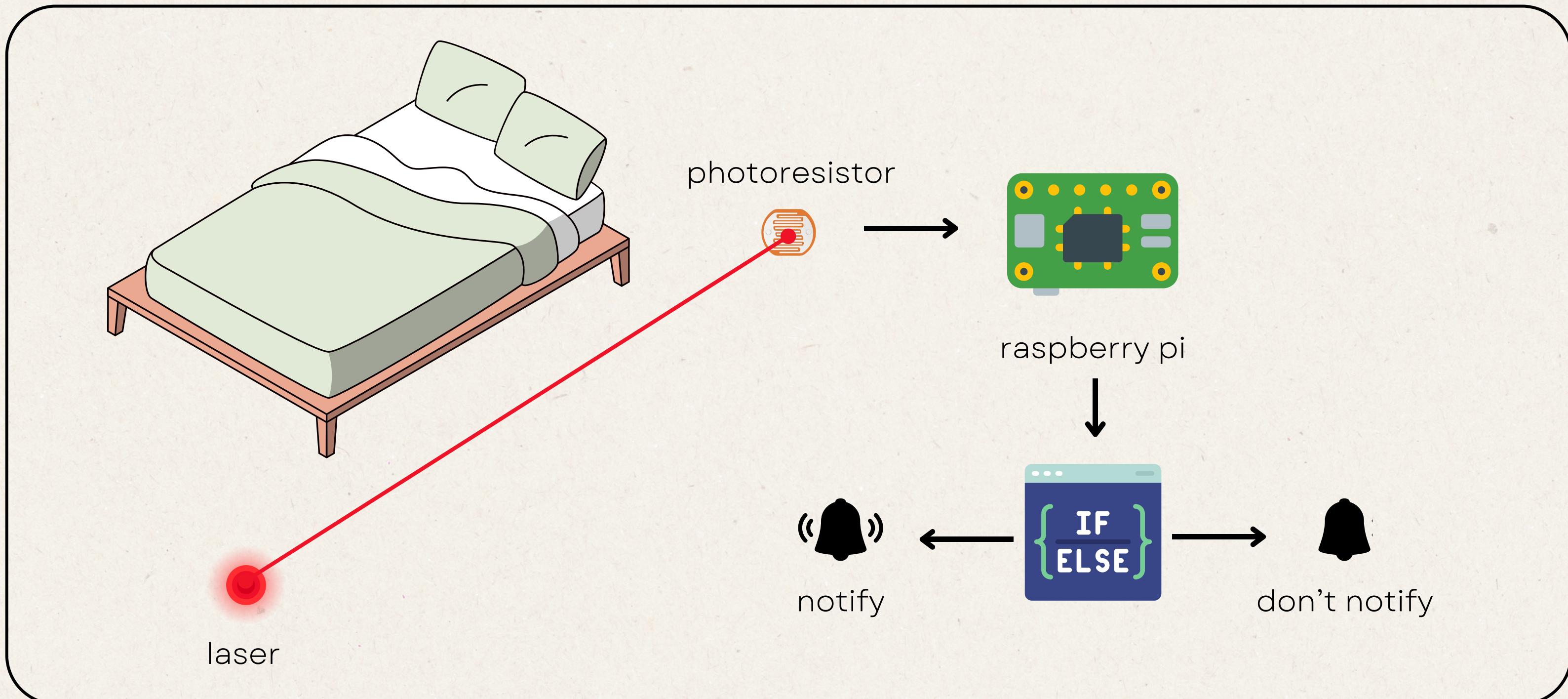
Our project aims to develop a **laser-based bed exit detection system** specifically designed for individuals who **cannot make coherent decisions**, such as patients in psychiatric wards, elderly individuals with cognitive impairments, or young children.

The system uses a laser beam projected across the bed and a photoresistor to detect when the beam is interrupted, indicating that the person has left the bed. The system is controlled by a Raspberry Pi, which sends an immediate alert to caregivers via IFTTT. This non-invasive, automated solution provides a proactive approach to ensure the safety of vulnerable individuals without relying on their ability to call for help.



Technical Overview

SYSTEM DESIGN



Our Implementation

```
import RPi.GPIO as GPIO
import time
import requests

LDR_PIN = 4 # GPIO pin connected to the LDR
IFTTT_URL = "https://maker.ifttt.com/trigger/motion_detected/with/key/QGj2DX350j4usMahLU0BE"

GPIO.setmode(GPIO.BCM)
GPIO.setup(LDR_PIN, GPIO.IN)

no_signal_count = 0 # Tracks no-signal duration
max_no_signal_time = 5 # Max allowed no-signal time (seconds)
last_notification_time = 0 # Timestamp of last notification
notification_interval = 10 # Minimum time between notifications (seconds)

try:
    print("Starting LDR monitoring...")

    while True:
        pin_state = GPIO.input(LDR_PIN) # Read LDR state

        if pin_state == GPIO.HIGH: # Light detected
            print("LDR is detecting light!")
            no_signal_count = 0
        elif pin_state == GPIO.LOW: # Darkness detected
            print("LDR is in the dark!")
            no_signal_count = 0

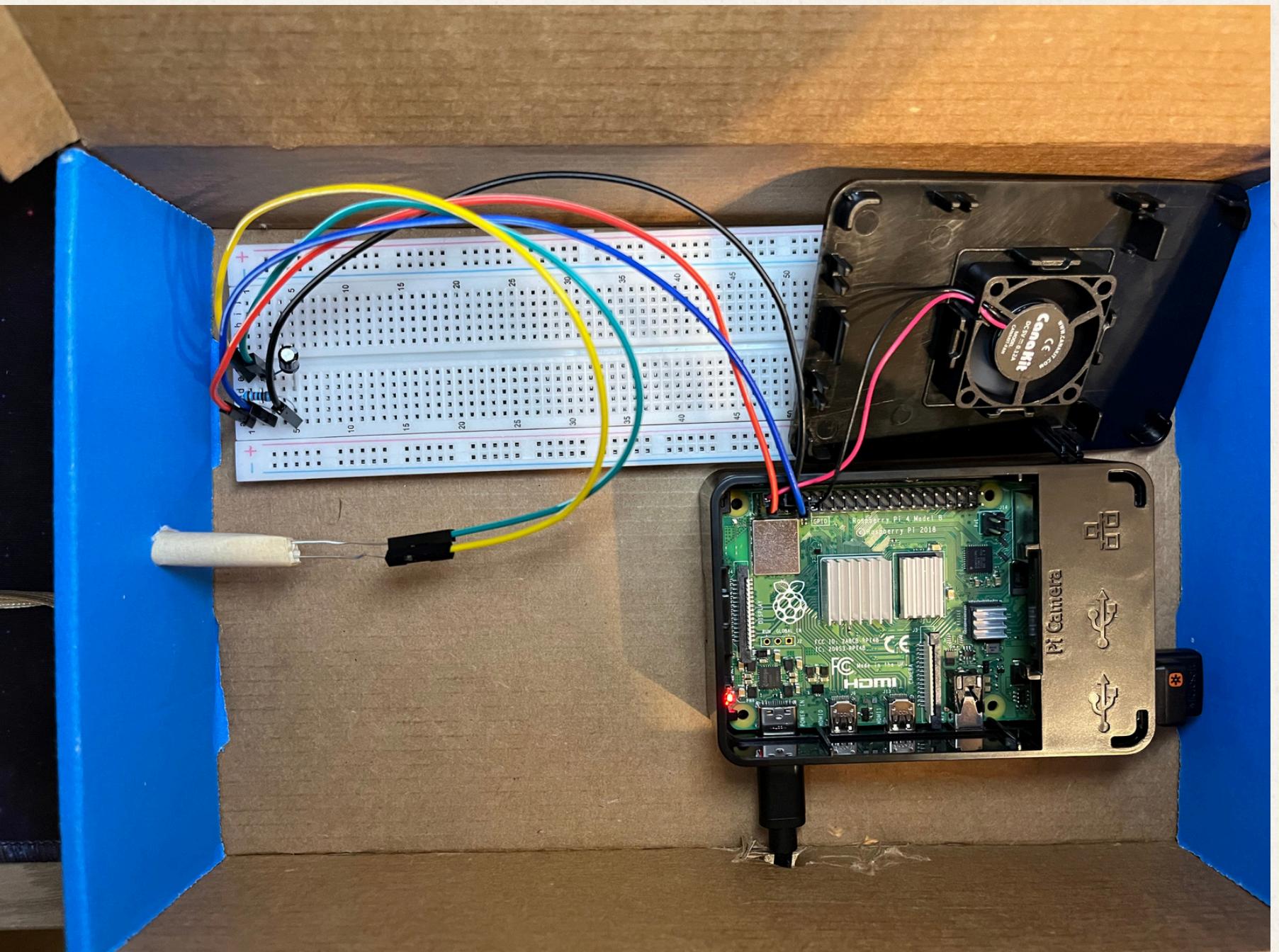
            # Send notification if interval has elapsed
            if time.time() - last_notification_time > notification_interval:
                print("Triggering IFTTT Webhook...")
                response = requests.post(IFFT_URL)
                if response.status_code == 200:
                    print("Trigger sent successfully!")
                else:
                    print(f"Failed to send notification: {response.status_code}")
                last_notification_time = time.time()

        else:
            print("Unknown GPIO state!") # Handle unexpected state

        no_signal_count += 1
        # Alert if no signal exceeds max allowed time
        if no_signal_count > (max_no_signal_time / 0.1):
            print("WARNING: No input detected for over 5 seconds!")
            no_signal_count = 0

        time.sleep(0.1) # Monitoring frequency delay

except KeyboardInterrupt:
    print("Program stopped by user.")
finally:
    GPIO.cleanup() # Clean up GPIO resources
```



Our Results: Demo

