IOT PHASE 4

SMART PARKING

DEVELOPMENT PART 2

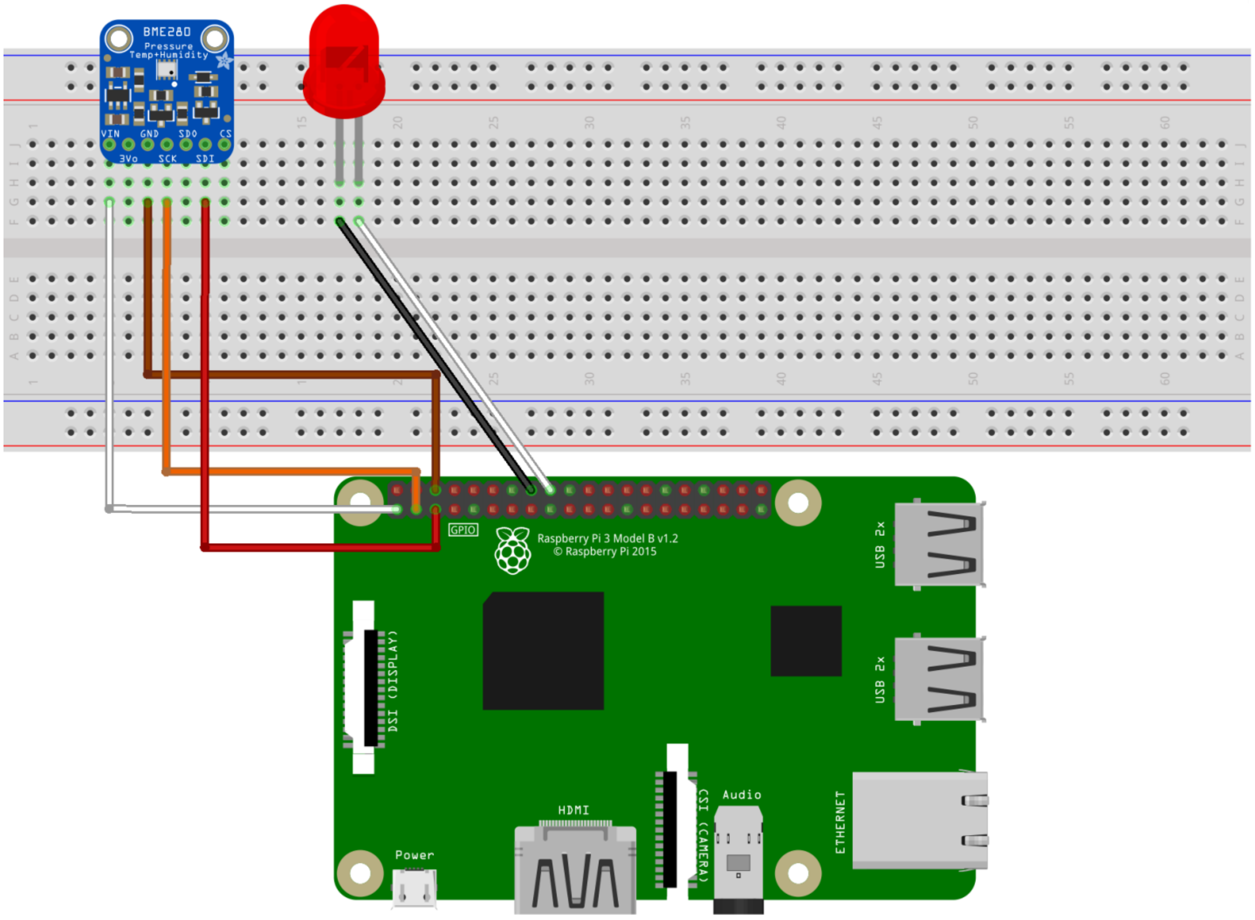
Building a camera-based parking detection system with Raspberry Pi and Microsoft Azure involves multiple steps, both in terms of hardware setup and software development. I'll break down the project into a detailed, step-by-step guide:

To check for parking space availability using camera:

Python code:

```
import picamera
import cv2
import requests
import time

# API endpoint for sending parking data
API_ENDPOINT = "https://your-server.com/api/parking"

# Initialize the camera
camera = picamera.PiCamera()

# Function to capture an image and analyze occupancy
def capture_and_analyze_image():
  # Capture an image
```

```
    timestamp = time.strftime("%Y%m%d%H%M%S")
    image_filename = f"parking_{timestamp}.jpg"
    camera.capture(image_filename)

    # Load the captured image
    image = cv2.imread(image_filename)

    # Implement image processing and analysis here (e.g., detecting car presence)

    # For demonstration, assume a simple condition for occupancy
    if True:  # Replace with your actual occupancy detection logic
        status = "Occupied"
    else:
        status = "Vacant"

    return image_filename, status

# Main loop for parking space monitoring
try:
    while True:
        image_filename, status = capture_and_analyze_image()

        # Send data to the server
        payload = {"space_id": 1, "status": status, "image_filename": image_filename}
        response = requests.post(API_ENDPOINT, json=payload)


        # Sleep for a defined interval before checking again
        time.sleep(10)


except KeyboardInterrupt:
    camera.close()
```


BME280
Pressure
Temp+Humidity
VIN
GND
SDO
CS
3Vo
SCK
SDI
GPIO
Raspberry Pi 3 Model B v1.2
© Raspberry Pi 2015
USB 2x
USB 2x
DSI (DISPLAY)
HDMI
CSI (CAMERA)
Audio
ETHERNET
Power