Embedded Systems Final Project Proposal

Group No:

Group Leader:

All Members:

Student No.	Full Name
114998411	黃子奇
114023022	Milan Esser
113598059	孫文禮
113598061	古雲鄉

The information in the table should include the group leader.

Project Topic:

Real-time Parking Monitoring and Automated Access Control System

Preferred Presentation Date:

2025/12/26

Description (Max 100 words):

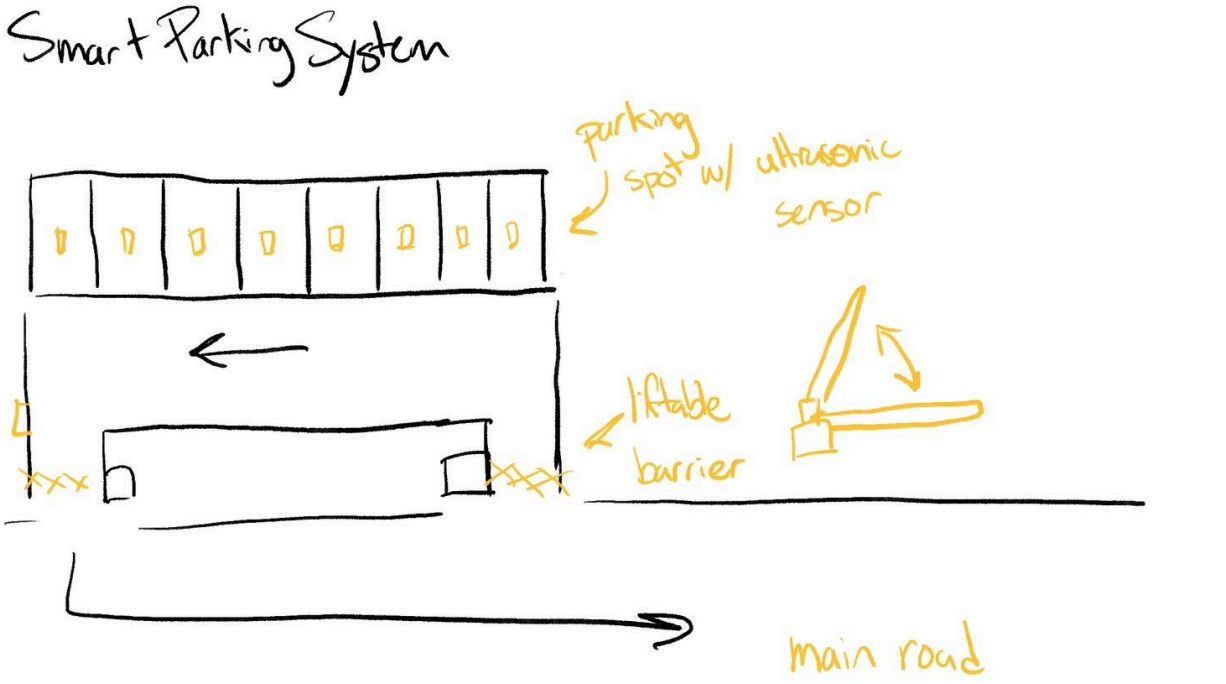
This project presents a smart parking management system designed for our Embedded Systems course. Powered by Raspberry Pi Pico 2, the system integrates multiple sensors to monitor parking slot occupancy in real-time. It continuously calculates remaining spaces and automates access control based on current capacity. Once the parking lot is full, the system immediately triggers a visual warning light and lowers the entry barrier to prevent further vehicle entry. This prototype demonstrates the practical application of hardware interfacing, sensor data processing, and automated logic control within a compact embedded environment.

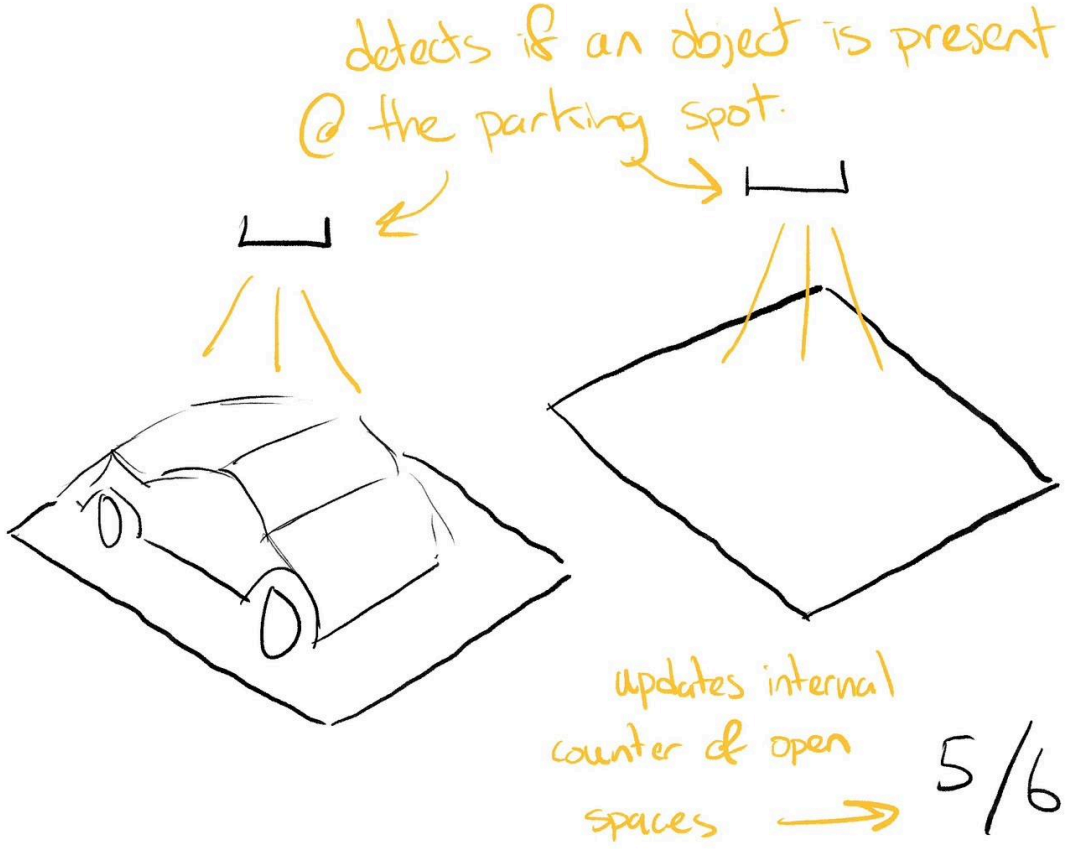
Feature List:

Feature No.	Brief Description
1	Continuously detects the presence of vehicles in individual slots using sensors to calculate and update the number of available parking spaces instantly.
2	Features a servo-driven barrier gate that automatically restricts entry when the capacity limit is reached, ensuring organized traffic flow
3	Utilizes LED indicators to provide immediate visual feedback, signaling a "Full" status to drivers when no spaces remain.
4	Use 1 pico as sensor manager, another one as server to control servo or display information

A List of features of the project you plan to implement. You can add or remove rows if needed.

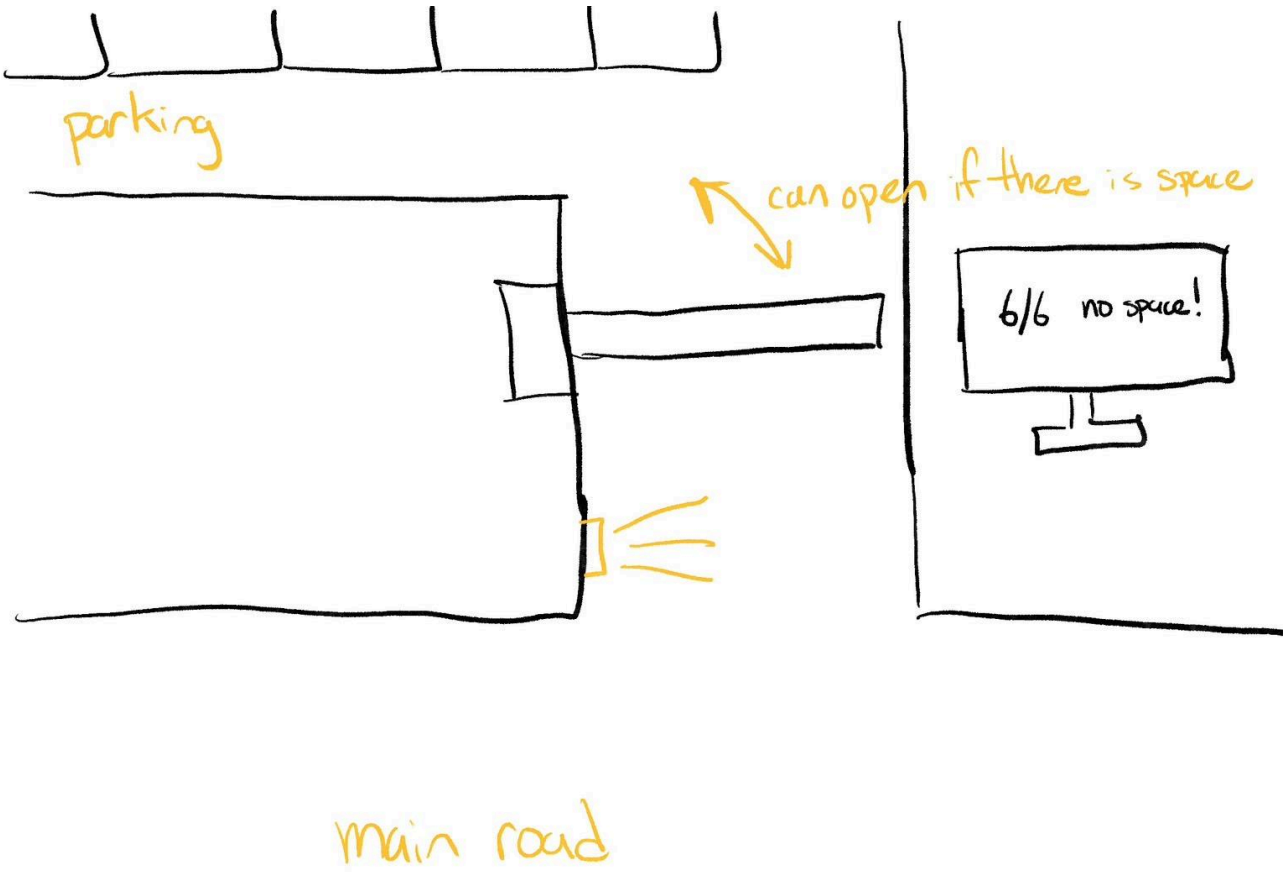
How to implement those features? (Max 300 words, explain by texts or illustrations):





we will install IR proximity sensors or ultrasonic modules at each parking slot, connected to the Pico 2's GPIO pins. The firmware will utilize polling or hardware interrupts to detect signal changes—logic HIGH or LOW—indicating a vehicle's presence. A global variable

will track the count, incrementing or decrementing based on sensor input to calculate available spaces dynamically.



a micro servo motor (e.g., SG90) will simulate the barrier gate. The servo's signal wire connects to a PWM-capable GPIO pin on the Pico. The code controls the barrier based on the occupancy variable: if available_slots > 0, the system sends a specific PWM duty cycle to rotate the servo 90 degrees (open). If the lot is full (available_slots == 0), the servo is locked at 0 degrees (closed) to physically block entry.