Phase 3 : design part 1

Introduction

In this phase we are going to design first 4 levels of the of innovation which is discuss in previous phase i.e phase 2

This level includes

- Motion sensor
- Payment
- Appilication
- Ardunio sensor

1 Motion sensor

In this we are going to user special kind of motion sensor (Displacement sensor) which is install after the gate beside the gate

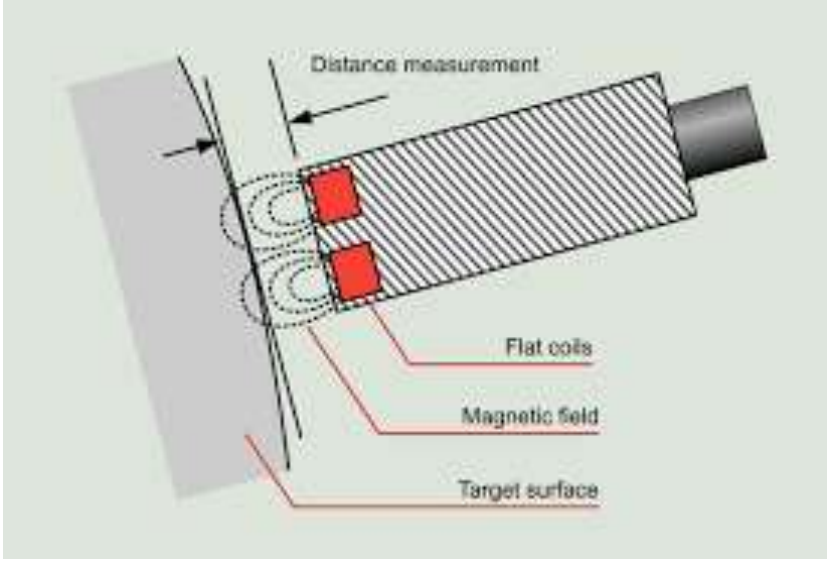
the motion sensor is connected to the control unit which monitors all activities inside the parking lot. The sensor projects a special kind of

A light which is use to detect the arrival and departure of the vehicles . The detection is done by the following process such as, normally a beam of light if vehicle/car crosses light then it is noted as 1 and the 1 is given to the control unit and the control unit is assume that as a vehicle crosses if vehicle doesn't crosses the sensor then it remains 0 if 0 means the control unit assume no vehicle crosses .

XMPP is open source protocol which is use to connect the IOT devices to the controller,where XMPP is extensible messaging present protocol,the reason for using protocol is it provides a fast connect between a devices and it updates frequently (events like arrival and departure of vehicles

A displacement sensor, also known as a position sensor or linear position sensor, is a device or transducer used to measure the displacement or change in position of an object from its initial point. Displacement sensors are commonly used in various industrial and scientific applications where precise measurement of linear or angular displacement is required. They play a critical role in automation,

robotics, automotive systems, manufacturing, and more.



2 Payment

Instant bank transfers

A services like the faster payment service in UK and similar system in other counters allow for fast ,real time bank transfers between individual and business

QR CODE PAYMENTS scanning QR CODE for payment has gained in various parts of the world customer can scan a QR CODE scan to make payment using their mobile devices

Unified Payments Interface (UPI) is a real-time payment system widely used in India for making various types of transactions. UPI enables users to link multiple bank accounts to a single mobile application and make seamless fund transfers, bill payments, online shopping, and more. Here are some key features and details about UPI payments:

Real-Time and Interoperable: UPI enables instant fund transfers between different banks. It is an interoperable system, meaning you can use any UPI-enabled app to send money to someone, regardless of which bank they use.

24/7 Availability: UPI transactions can be initiated and completed 24/7, including weekends and holidays, which makes it highly convenient for users.

Mobile Apps: To use UPI, you need a mobile app provided by your bank or a third-party UPI app like Google Pay, PhonePe, or Paytm. These apps allow you to link your bank account(s) and perform various financial transactions.

Virtual Payment Address (VPA): Instead of sharing sensitive bank account details, users can create a Virtual Payment Address (VPA) linked to their bank account. This VPA is used to send and receive money. For example, a VPA might look like "yourname@bankname."

Multiple Payment Options: UPI supports various types of transactions, including person-to-person (P2P) transfers, person-to-merchant (P2M) payments, bill payments, and recharges.

Security: UPI transactions are secure and often require multi-factor authentication, such as PIN, fingerprint, or biometric authentication, to complete a transaction. Additionally, UPI transactions use robust encryption to protect user data.

QR Code Payments: QR codes are commonly used for UPI payments. Merchants generate QR codes that customers can scan using their UPI app to make payments.

Bank Integration: Most major banks in India are part of the UPI ecosystem, allowing their customers to use UPI for various transactions.

Transaction Limits: UPI transactions are subject to limits, both daily and per transaction, as per the user's bank's policies.

It's important to note that UPI has continued to evolve since my last update in September 2021, with new features and improvements being introduced regularly. Users in India should keep their UPI apps up to date and be aware of any changes in transaction limits or additional security measures.

Application

A parking slot booking app is a mobile application that allows users to search for, book, and manage parking slot for their journeys. These apps have become increasingly popular and convenient for travelers as they streamline the slot booking process and provide a range of services and features.

The algorithm /coding of the application is as follows

```
# Define the parking spaces as a dictionary where keys are space numbers and values are availability status.parking_spaces =
{    1: "Available",
     2: "Available",
     3: "Available",
     4: "Available",
}
   def display_available_spaces():
         print("Available parking spaces:")
         for space, status in parking_spaces.items():
           if status == "Available":
                print(f"Space {space}")
def book_parking_space(space, duration):
    if space in parking_spaces and parking_spaces[space] == "Available":
      parking_spaces[space] = f"Booked for {duration} hours"
      print(f"Space {space} is booked for {duration} hours.")
```

```
else:
        print(f"Space {space} is not available or doesn't exist.")


while True:
  display_available_spaces()
        try:
     space = int(input("Enter the parking space number you want to book (1-4): "))

     if space not in parking_spaces:
          print("Invalid parking space number. Please choose a space between 1 and 4.")
          continue
  duration = int(input("Enter the duration in hours: "))
  if duration <= 0:
        print("Invalid duration. Please enter a positive number of hours.")
          continue
    book_parking_space(space, duration)
    another_booking = input("Do you want to make another booking? (yes/no): ")
     if another_booking.lower() != "yes":
                break
except ValueError:
    print("Invalid input. Please enter valid numbers.")
```

Motion sensors /ardunio sensor

The sensors has to place in two end of the parking space i,e if parking lot contains a 20 spaces in the dimention of 2X10 then pair of 14 sensors required the sensors are arrange such a way that it should cover all the parking spaces ,if the entire parking lot as no vehicles the sensors doesn't feel any distrubance i,e no activity is occurred is updated in the IOT server. If a car is parked in a specific place then the distrubance is felt by the sensors and updates in the IOT server. For Example if a car parked in the 1st vertical row of 5th horizontal column then the distrubance is felt in the sensors of 5' 5'' and A'and A'' untill the car as be taken away from the parking space till that the circuit not complete ,once the car as been taken from the parking space the circuit completed and updates the server that the parking space is free.(the sensor which is used motion sensor)

Arduino IDE: To program Arduino boards, you can use the Arduino IDE, which is a software development environment that simplifies the process of writing and uploading code to the board. The code is typically written in C/C++ and is uploaded to the Arduino board via a USB connection.

Working

The working of the ardunio sensor is based on the signal given by the motion sensors in this region the entire parking lot where the cars has been parked is fully covered with the motion sensor suppose a parking space occupie 20 spaces in the order of 2X10 then 14 sensors has to been required to cover the entire area suppose a car occupies a specific parking space then the circuit of the certain sensors has been blocked an it is updated in the ardunio mother board.

CONSTRUCTION

The construction required for the working of the ardunio sensor is changes from 1 parking lot to another parking lot The same XMPP protocol is use for communicating the IOT sensor and ardunio sensor since it provide faster communication and provide accurate information and it updates to the controller

Where all the sensors are connect in series in such a way that a small disturbances can be noted and responds to it.