**CONVENIENT CAR PARKING SYSTEM**

**LIBRARIES USED -**

*cv2, imutils, numpy, pytesseract, datetime, timedelta* and *time*

**PROJECT STRUCTURE -**

The project can be divided into three parts:

1. License Plate Detection
2. Parking Lot
3. main

**LICENSE PLATE DETECTION:**

**Modeling class:**

1. LicensePlateRecognition(*image\_path*)

* Member methods:
* get\_license\_number(*self*)

**Working:**

The following procedure is followed to obtain the license plate number from the given image. First, we convert the image to grey scale since it is easy to work on a greyscale image. Then, we blurred the greyscale image to remove noise. We then look for contours in our image. If any of all the contours found have four points in it, then it is the license plate. We then obtain that part of image and pass it to *pytesseract* library which returns the license plate number as string.

**Parking Lot:**

**Modeling class:**

1. Car (*self, license\_number*)

* Member variables
* License number
* Member methods
* get\_license\_number(*self*)

1. ParkingLot (*self, capacity, fixed\_cost\_for\_an\_hour, cost\_per\_half\_hour\_interval*)

* Member variables
  + - \_CONST\_CAPACITY
    - \_car\_count
    - \_cost\_per\_minute
    - \_car\_lot
    - \_fixed\_cost\_for\_an\_hour
    - \_cost\_per\_half\_hour\_interval
* Member methods:
  + - \_add\_car(self, car):
    - \_check\_parking\_space(self):
    - \_remove\_car(self, car):
    - \_print\_cost(self, car):
    - \_is\_present(self, car):
    - scan\_car(self, car):
    - display\_parking\_lot(self):

**Working:**

We first create a parking lot object with some defined capacity and cost for stay. Then, for each license plate obtained from the car image, we create a car object. Then that object is sent to the parking lot object *scan\_car(self, car)* method. This method scans the car and checks if its already present in the parking lot or not, if its not then its added to the parking lot else it is removed from the parking lot. If any car is removed, then a receipt displaying the total cost of stay is displayed. After addition or removal of each car, the details of the parking lot are displayed using *display\_parking\_lot(self)*.

**main:**

**Working:**

The main connects both the *LicensePlateRecognition* and *ParkingLot* class objects so that they car work together. It creates object of both *LicensePlateRecognition* and *ParkingLot.* Then, each time it sends an image to the *LicensePlateRecognition* object. The returned license number is sent to the *ParkingLot* object which takes the appropriate action.