Project Title: Parking Lot

Author: Osváth Tamás

Group: 30421

**TASK DESCRIPTION**

Simulate partially the activity of a paid parking lot. The lot has a capacity of up to 50 cars.

You are to create a dashboard for the parking supervisor.

Cars come in through one entry which is provided with a barrier. They receive a ticket upon entry. Then the car is driven to an available lot. There is a queue of cars at the entry. Cars are admitted only if there is enough room for them. When a car attempts to leave the parking through the exit point, they are charged according to the time spent in the parking. There is a maximum time limit for parking, and charges vary depending on the time of day the car is parked. At any moment, the supervisor should know what cars are in the parking (by their license plate), how many available places are there, how much parking fees were collected so far during a day.

The simulation takes place in 1 minute steps.

You are supposed to simulate the activity the parking lot and provide a view of this.

All the relevant data (initial configuration, schedule, history data) should be stored in flat files.

The application should allow for saving the state and restoring it (serialization). A dashboard view should show the activity involved in this simulation. This should be a GUI which you should design.

**APPROACH**

First of all, we need to identify the objects which will compose the parking lot. It is straightforward that we need Car objects, we need a Car Queue for the cars, and a Parking Lot. These are the basic classes of the project.

The **CAR** class will contain information such as license plate, entry time and exit time in the parking. The cars will be added to the queue, and once they get permission, they can enter the parking, receiving the first empty slot.

The **CARQUEUE** class, as mentioned above, will gestion only the cars waiting to enter the parking.

The **PARKINGLOT** class will manage the cars, calculating their profit, letting them in, and exiting them. Also, it will display a dashboard with every slot.

**SIMULATION**

The simulation consists in creating a random number of cars, adding them to the queue, and entering them in the parking at random times. There is a timer which can be manipulated in order to get a meaningful simulation. After that, again, at random times, cars begin to exit, and profit is calculated based on that.

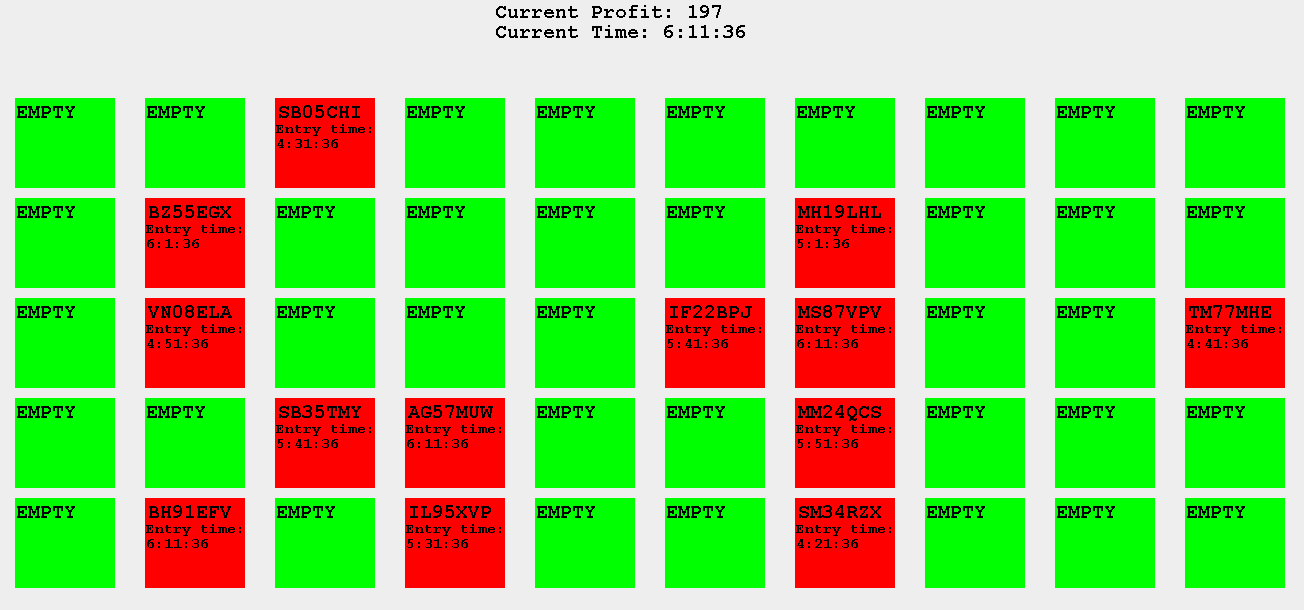
**GUI**

The GUI is pretty basic, but meaningful, since it shows all the relevant data, such as the current profit, current time and the dashboard with the cars:

A slot is green if it is empty, saying “EMPTY” on it.

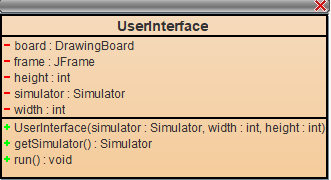
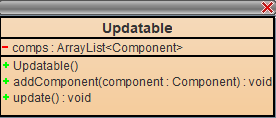
A slot is red it there is a car parked, displaying its license plate and entry time – all the necessary information about it.

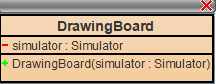
The GUI uses the Graphics2D class, in order to facilitate the drawing of rectangles, strings and so on.



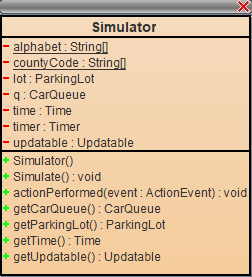
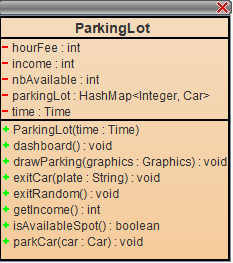
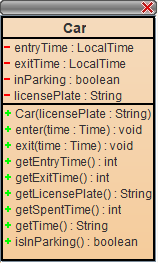
**Class Cards**

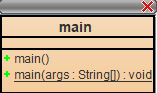
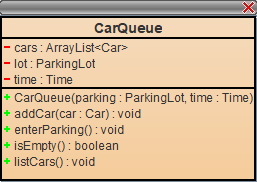
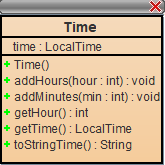
-Package GUI

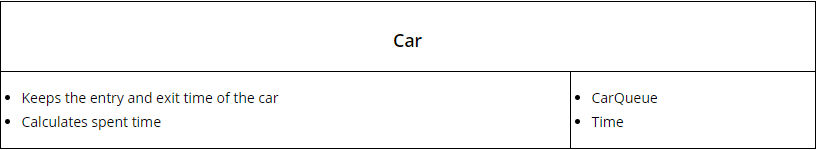


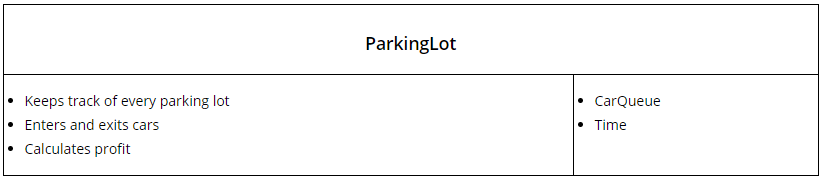
-Package parking

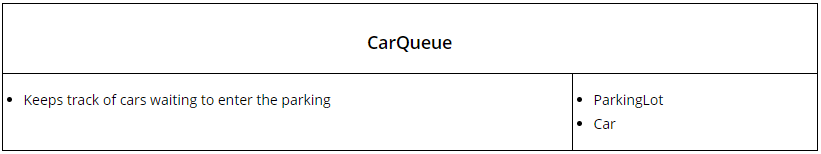
  

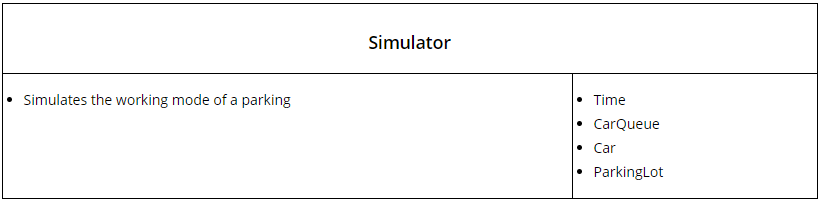
  

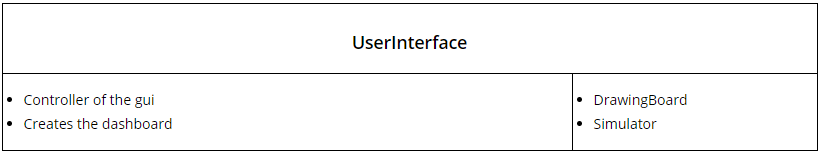
**CRC Cards**

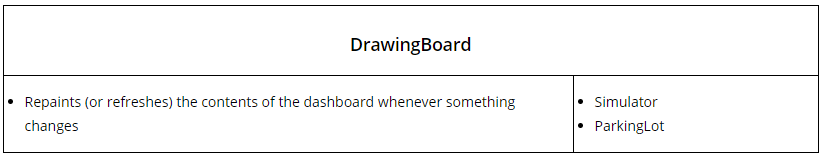












**Remarks:**

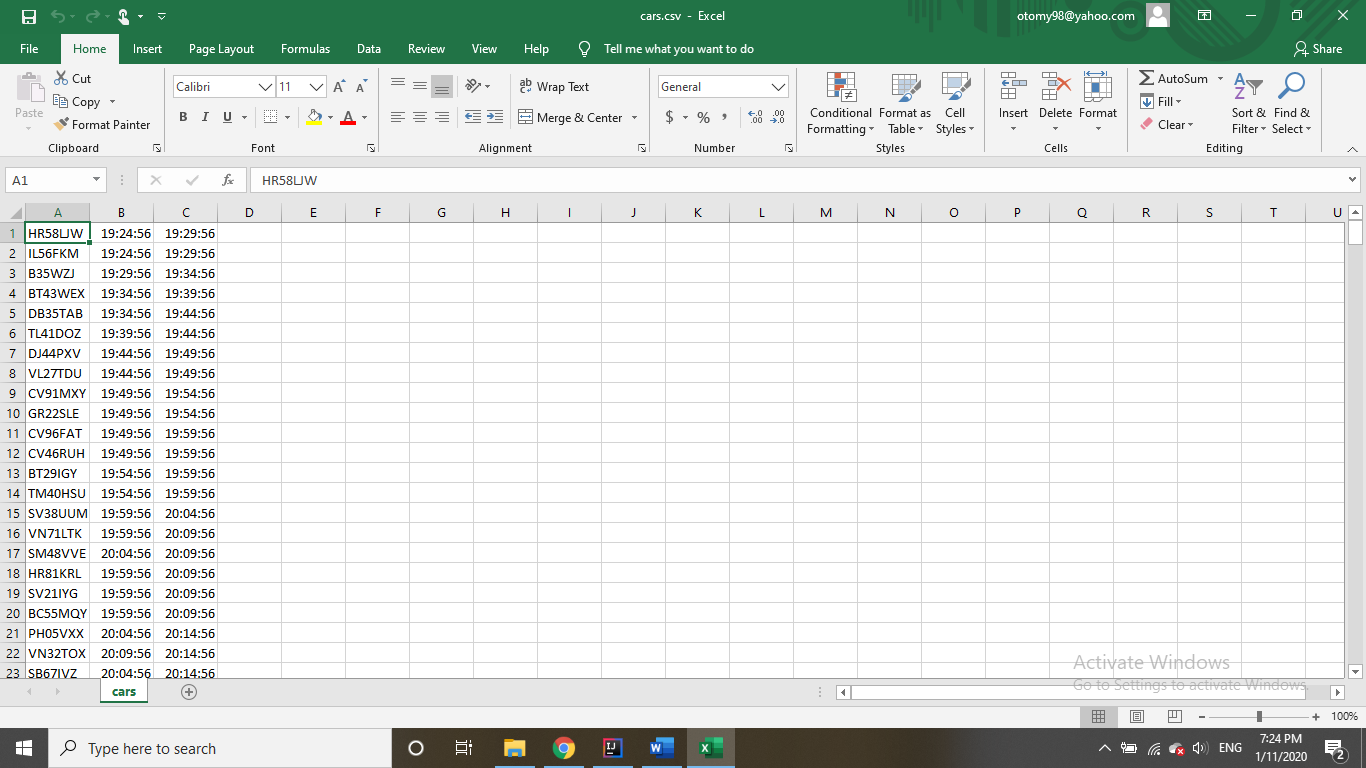
The simulation has the following particularities, which can be easily modified as wanted:

-every 5 minutes, a random number of cars (from 1 to 5) are entered and exited

-the Timer’s delay is 30 milliseconds, but can be modified in order to be able to visualize easier the simulation

-the Profit gets reset after each day (at midnight)

-the cars which entered the parking are saved in a CSV file



**Sources:**

[www.Tutorialspoint.com](http://www.Tutorialspoint.com)

[www.geeksforgeeks.com](http://www.geeksforgeeks.com)

\*to be mentioned that the classes from the gui package use functions already implemented from these sites, such as update()