Design choice:

We understand there are various components in the parking lot system. We also went through some real life implementation of an ordinary parking lot and tried abstracting the ideas on the OOPs domain.

There is the customer, his vehicle, the building itself, the floors in it, the parking spot, payment portal and so on.

We listed down properties of these various components, and decided the functions on them.

We also decided that all input for the various element will be taken in the main function and the respective constructors will be called to initialize the object elements.

We also understood that number of parking spot and floors are static i.e. in the real case, these values do not change. Once set, they cannot be altered. Hence, these had to be static arrays.

The customers and their vehicles are variable i.e. the number of these objects vary from time to time. This collection had to be dynamic and hence, we chose arraylist to represent customer, bike and truck.

We needed a common element in customer and their vehicle element in order to uniquely identify and link them with one another. We chose an integer value that is same in both customer and their vehicle. This allows us to retrieve their vehicle information and its parking spot, once we have the customer’s unique id.

We choose the parking space for the customer. This done to maximize efficiency in parking. The place in the next floor is allotted only after the previous floor is filled.

We also understand the need for a supervisor to monitor the customer data and manually make changes their presence in the system, in case of any error. So there is provision for an admin which can only be accessed with a password.

Work division:

The classes we had were :

* Parking\_lot (Main class)
* Vehicle (Has child classes – car, bike, truck)
* Entrance (Interface on entry for taking in values)
* Exit (Interface on exit for payment)
* Admin (For supervision of information and manual adjustments on customer)
* Floor (Defines the type of parking spots available in that particular floor)
* Parking spot (Defines the properties of the parking spot

We worked on Google meet with a person sharing the screen.

Srujan, Yashwanth and Gagan took up the task for creating the classes vehicle, entrance and exit.

Sowmya and Pavan took up the task for designing the function and members of floor and parking spot.

The main class “Parking\_Lot” had its basic outline by Pavan. The whole team worked together in coding the parking lot because each person knew how the functions had been implemented in their respective classes.

Debugging was also done with everybody putting in different ideas for the cause of errors.

No one was slacking in the task assigned to them and were eager to help one another.