# Scenario #: Car Park

## Scenario Description

* Make sure carpark give out receipt when car is detected and if the receipt (Tickets) has been given out, the gate should go up and receipt should provide price, time, car park capacity and receipt number for customers to pay for their bill when they are exiting, and also in exiting LCD display should provide those information for each different customers, and if there is limitation it should show in the LCD display for them.

## Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version # | Date | Author | Description |
| 0.1 | 24/09/2017 | Jaehoon Lee | Initial Draft |
| 1.0 | 25/09/2017 | Jaehoon Lee | Initial Version |

## Test Scripts

The following scripts will cover this scenario:

* 1.1 Car Entry, No Restrictions
* 1.2 Car Entry, Restrictions
* 1.3 Car Entry, Tickets (Receipts)
* 1.4 Limitation of Car Park ( Capacity)
* 1.5 Car Park, Invalid Receipt
* 1.6 Car Park, Invalid Payment
* 1.7 Car Park, Car Sensor Error
* 1.8 Car Park, Exceeding capacity
* 1.9 Car Park, Gate restrictions

## Use Case

* Car Park
* Carp Sensor
* Exit Sensor
* Entry Sensor

## Test Components/Requirements

This test scenario covers the following high-level test requirements (see scripts below for specific requirements covered by each test script):

* Unrestricted car, restricted, car sensor not detecting the car, giving out wrong receipt (car overstayed , too many cars in the car park, unpaid bills) users created
* Available space in car park, and different bills for different cars who stayed in the car park.
* Season tickets.
* Valid receipts for every car.
* Valid payment
* Valid time and date

### Testing Requirements

Test scripts covers the following specific testing requirements:

Carpark entry has circumstances where the customer has no restricts when car park space is available.

Carpark entry has circumstances where the customer has restricts when car park is full.

Carpark entry has circumstances where the customer has restricts when they have wrong tickets.

Carpark exit has circumstances where the customer has restricts when each tickets needs to be payed

Carpark exit has circumstances where the customer has restricts when ticket is invalid.

### Setup

* unrestricted customers
* customer 1 – with ticket and with correct payment
* Customer 2 – with correct tickets and with receipt which proofs that he/she has payed the bill.
* Customer with restrict
* Who has got wrong ticket
* Where carpark capacity is full
* Where customer who didn’t pay for their ticket
* Create available space customer
* Car park space – available,
* Car park space2 – unavailable,

### Teardown

* List all steps that should be taken after the test case is executed

1. Car sensor detects customer
2. Customer gets tickets when there is available space in the carpark
3. Gate up
4. Exit sensor detects customer’s car
5. User pays their tickets and gets receipts
6. User scans receipts in exiting stage
7. Gate up

### Script Steps

| **Step #** | **Test Action** | **Expected Results** | **Pass/ Fail** |
| --- | --- | --- | --- |
| 1 | Car Detected | LCD display Detects and if car is detected shows green | Pass |
| 2 | Customer gets valid ticket | Ticket provides customer time / date and price | Pass |
| 3 | Check for car park capacity and gate up | Gate up and turns green | Pass |
| 4 | Customer exit sensor detected | Customer goes to exit sensor | Pass |
| 5 | Customer pays the bill | Gives customer receipt if it is correct price | Pass |
| 6 | Customer gets the receipt and ticket to the exit | Customer confirms tickets | Pass |
| 7 | Gate up / customer leaves | Leaves | Pass |

### Test Execution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date/Time | Tester | Test ID | Test Phase | Status |
| 01/09/17 1:35Pm | Jaehoon Lee | Github ID = jaehoon | Car Park | Passed |
| 02/09/17 3:45 Pm | Jaehoon Lee | Github ID = jaehoon | Car Sensor | Passed |
| 04/09/17 3:45 Pm | Dylan Qiao | Github ID = JiaQiao | Gate | Passed |
| 06/09/17 3:45 Pm | Jaehoon Lee | Github ID = jaehoon | Entry Controller | Passed |
| 06/09/17 3:45 Pm | Dylan Qiao | Github ID = JiaQiao | Entry PUI | Passed |
| 10/09/17 3:45 Pm | Jaehoon Lee | Github ID = jaehoon | Pay station UI | Passed |
| 13/09/17 3:45 Pm | Dylan Qiao | Github ID = JiaQiao | pay station Controller | Passed |
| 13/09/17 3:45 Pm | Jaehoon Lee | Github ID = jaehoon | Adhoc Ticket | Passed |
| 14/09/17 3:45 Pm | Jaehoon Lee | Github ID = jaehoon | Adhoc TicketDAO | Passed |
| 15/09/17 3:45 Pm | Dylan Qiao | Github ID = JiaQiao | Adhoc TicketFactory | Passed |
| 18/09/17 3:45 Pm | Dylan Qiao | Github ID = JiaQiao | Season Ticket | Passed |
| 20/09/17 3:45 Pm | Jaehoon Lee | Github ID = jaehoon | Season TicketDAO | Passed |
| 21/09/17 3:45 Pm | Dylan Qiao | Github ID = JiaQiao | UsageRecord | Passed |
| 22/09/17 3:45 Pm | Jaehoon Lee | Github ID = jaehoon | UsageRecordFactory | Passed |