# Scenario 1.1: Enter Carpark

## Scenario Description

## When a customer wants to enter the car park, they interact with the entrance control pillar, so that they can enter the car park and park their car.

## Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version # | Date | Author | Description |
| 0.1 | 22/09/2017 | Ryan Smith | Initial Draft |
| 1.0 | 22/09/2017 | Ryan Smith | Initial Version |

## Test Scripts

The following scripts will cover this scenario:

* 1.1 Successful entry – Adhoc ticket
* 1.2 Successful entry – Adhoc ticket, after waiting for a car to leave the full carpark.
* 1.3 Successful entry – Season ticket
* 1.4 Failed entry – Invalid Season ticket holder
* 1.5 Failed entry – Blocked entrance
* 1.6 Failed entry – User backs out after partially entering

## Use Case

* Enter Carpark

## Test Components/Requirements

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

* A record of the ticket has been created or updated and stored.
* The customer has been allowed or denied entry.
* Number of available car spaces for adhoc ticket customers is decremented
* The season ticket is recorded as in use.
* Entry Controller UI displays appropriate messages

## User Groups

* Adhoc Ticket customer
* Season Ticket holder

## Script 1.1: Successful entry – Adhoc ticket

### Script Description

* The user attempts to enter the carpark without being a season ticket holder.

### Testing Requirements

This test script covers the following specific testing requirements:

* A record of the ticket has been created and stored
* The customer has been allowed entry
* The number of available spaces for adhoc ticket users has been decremented
* Entry controller UI displays appropriate messages.

### Setup

* Carpark system is initialized. It will be allowed to hold 20 cars, 2 of which would be season ticket holders during business hours.

### Teardown

* No teardown is necessary after this script.
* No teardown is necessary between this script and Script 1.2, as that script will require this script be run a specified number of times as setup.

### Script Steps

| **Step #** | **Test Action** | **Expected Results** | **Pass/ Fail** |
| --- | --- | --- | --- |
| 1 | Detection of a car on outside sensor | Entry pillar UI (herein “UI”) displays “Push Button” | Pass |
| 2 | “Issue Adhoc Ticket” button pushed | Ticket is created and displayed by printer, UI displays “Take Ticket” | Pass |
| 3 | Ticket barcode copy-pasted to test results text file | File Script 1.1 Results.txt contains barcode of ticket. | Pass |
| 4 | “Take Ticket” button pushed | Gate is open, UI reads “Ticket Taken”. | Pass |
| 5 | Detection of a car on inside sensor | UI displays “Entering” | Pass |
| 6 | Detection of no car on outside sensor | UI displays “Entered” | Pass |
| 7 | Detection of no car on inside sensor | Gate is closed, UI displays “Idle”, ticket has been recorded in system | Pass |

### Test Execution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date/Time | Tester | Test ID | Test Phase | Status |
| 22/09/2017 11:47 am | Ryan Smith | Rsmith1 | System Cycle 1 | Passed x18 |

## Script 1.2: Successful entry – Adhoc ticket, after waiting for a car to leave the full carpark

### Script Description

* The user attempts to enter the carpark without being a season ticket holder, but the carpark is full and the user must wait for another car to leave.

### Testing Requirements

This test script covers the following specific testing requirements:

* A record of the ticket has been created and stored
* The customer has been denied entry, then allowed entry after another car left the carpark
* The number of available spaces for adhoc ticket users has been decremented
* Entry controller UI displays appropriate messages.

### Setup

* Carpark system is initialized. It will be allowed to hold 20 cars, 2 of which would be season ticket holders during business hours.
* Script 1.1 has been run 18 times during business hours, or 20 times while outside of them. The test data for those tests should be accessible from Script 1.1 Results.txt – it will be necessary to use that data.

### Teardown

* No teardown is necessary after this script.

### Script Steps

| **Step #** | **Test Action** | **Expected Results** | **Pass/ Fail** |
| --- | --- | --- | --- |
| 1 | Detection of a car on outside sensor | Entry pillar UI (herein “UI”) displays “Push Button” | Pass |
| 2 | “Issue Adhoc Ticket” button pushed | UI displays “Carpark Full” | Pass |
| 3 | Copy-pasted any barcode from Script 1.1 results.txt into a new file Script 1.2 results.txt. | Script 1.2 results.txt contains a barcode from an iteration of the previous script. | Pass |
| 4 | Pasted that barcode into the paystation controller, Read Ticket button pushed | Paystation UI displays “Pay (some amount)” | Pass |
| 5 | “Pay” button pushed, then “Take Ticket” button pushed | Ticket is paid and reprinted in the Paystation Ticket Printer. | Pass |
| 6 | Detection of a car on inside exit sensor | Exit controller displays “Insert Ticket” | Pass |
| 6 | Pasted that barcode into the exit controller, Read Ticket” button pushed. | Entry Controller UI displays “Take Processed ticket | Passs |
| 7 | “Take Ticket” button pushed on Exit Controller | Exit gate open, UI reads “Ticket Taken”. | Pass |
| 8 | Detection of a car on outside exit sensor | UI displays “Exiting” | Pass |
| 9 | Detection of no car on inside exit sensor | UI displays “Exited” | Pass |
| 10 | Detection of no car on outside exit sensor | Exit gate is closed, UI displays “Idle”, ticket has been recorded as exited in system, Entry UI displays Push Button | Pass |
| 11 | “Issue Adhoc Ticket” button pushed | Ticket is created and displayed by printer, UI displays “Take Ticket” | Pass |
| 12 | Ticket barcode copy-pasted to test results text file | File Script 1.2 Results.txt contains barcode of ticket. | Pass |
| 13 | “Take Ticket” button pushed | Gate is open, UI reads “Ticket Taken”. | Pass |
| 14 | Detection of a car on inside sensor | UI displays “Entering” | Pass |
| 15 | Detection of no car on outside sensor | UI displays “Entered” | Pass |
| 16 | Detection of no car on inside sensor | Gate is closed, UI displays “Idle”, ticket has been recorded in system | Pass |

### Test Execution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date/Time | Tester | Test ID | Test Phase | Status |
| 22/09/2017 12:20 pm | Ryan Smith | Rsmith1 | System Cycle 1 | Passed x2 |

## Script 1.3: Successful entry – Season Ticket

### Script Description

* Season ticket holder attempts to enter the carpark.

### Testing Requirements

This test script covers the following specific testing requirements:

* A record of the season ticket has been updated and stored.
* The customer has been allowed entry.
* The season ticket is recorded as in use.
* Entry Controller UI displays appropriate messages

### Setup

* Carpark system is initialized. It will be allowed to hold 20 cars, 2 of which would be season ticket holders during business hours.
* Two season tickets should be created, with the codes “S1111” and “S2222”. One code will be used each time the test is run
* Test should be executed during business hours, ideally while the carpark is full after script 1.2 has been executed, to demonstrate the reservation of spaces for season ticket holders.

### Teardown

* No teardown is necessary after this script.
* No teardown is necessary between this script and Script 1.4, as that script will require this script be run a specified number of times as setup.

### Script Steps

| **Step #** | **Test Action** | **Expected Results** | **Pass/ Fail** |
| --- | --- | --- | --- |
| 1 | Detection of a car on outside sensor | Entry pillar UI (herein “UI”) displays “Push Button” | Pass |
| 2 | Season ticket code has been entered into the Season Ticket Reader, and the “Validate Season Ticket” button has been pushed. | UI reads “Ticket Validated”. | Pass |
| 3 | “Take Ticket” button pushed | Gate is open, UI reads “Ticket Taken”. | Pass |
| 4 | Detection of a car on inside sensor | UI displays “Entering” | Pass |
| 5 | Detection of no car on outside sensor | UI displays “Entered” | Pass |
| 6 | Detection of no car on inside sensor | Gate is closed, UI displays “Idle”, ticket has been recorded in system | Pass |

### Test Execution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date/Time | Tester | Test ID | Test Phase | Status |
| 22/09/2017 12:35 pm | Ryan Smith | Rsmith1 | System Cycle 1 | Passed x2 |

## Script 1.4: Failed entry – Invalid Season ticket holder

### Script Description

* A season ticket holder attempted to enter the carpark, but

### Testing Requirements

This test script covers the following specific testing requirements:

* The customer has been denied entry.
* Entry Controller UI displays appropriate messages

### Setup

* Carpark system is initialized. It will be allowed to hold 20 cars, 2 of which would be season ticket holders during business hours.
* Two season tickets should be created, with the codes “S1111” and “S2222”. Script 1.3 should be run twice to set both these season tickets as “in use”. The data for this test will be one ticket already in use (“S1111”) and one made up ticket with the code “S1234”
* Test should be executed during business hours, ideally while the carpark is full after script 1.2 has been executed, to demonstrate the reservation of spaces for season ticket holders.

### Teardown

* No teardown is necessary after this script.

### Script Steps

| **Step #** | **Test Action** | **Expected Results** | **Pass/ Fail** |
| --- | --- | --- | --- |
| 1 | Detection of a car on outside sensor | Entry pillar UI (herein “UI”) displays “Push Button” | Pass |
| 2 | Season ticket code has been entered into the Season Ticket Reader, and the “Validate Season Ticket” button has been pushed. | UI will beep, and reads “Invalid Ticket” and rejects the ticket. | Fail (no message displayed) |

### Test Execution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date/Time | Tester | Test ID | Test Phase | Status |
| 22/09/2017 12:47 pm | Ryan Smith | Rsmith1 | System Cycle 1 | Failed x2 |

## Script 1.5: Failed entry – Entrance blocked

### Script Description

* A user attempted to enter the carpark, but the gate was blocked by another car or an obstruction.

### Testing Requirements

This test script covers the following specific testing requirements:

* The customer has been denied entry.
* Entry Controller UI displays appropriate messages

### Setup

* Carpark system is initialized. It will be allowed to hold 20 cars, 2 of which would be season ticket holders during business hours.

### Teardown

* No teardown is necessary after this script.

### Script Steps

| **Step #** | **Test Action** | **Expected Results** | **Pass/ Fail** |
| --- | --- | --- | --- |
| 1 | Detection of a car on inside sensor | Entry pillar UI (herein “UI”) displays “Blocked” | Pass |
| 2 | Detection of a car on outside sensor | UI continues to display “Blocked” | Pass |
| 3 | “Issue Adhoc Ticket” button pushed | UI will beep, and continue to display “Blocked”. | Pass |
| 4 | Detection of no car on inside sensor | UI displays “Push Button” | Pass |

### Test Execution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date/Time | Tester | Test ID | Test Phase | Status |
| 22/09/2017 12:56 pm | Ryan Smith | Rsmith1 | System Cycle 1 | Passed |

## Script 1.6: Failed entry – User backed out

### Script Description

* A user attempted to enter the carpark, but backed out before completely passing the gate, or because the carpark was full.

### Testing Requirements

This test script covers the following specific testing requirements:

* The customer has been denied or allowed entry.
* The customer has chosen not to enter the carpark after interacting with the control pillar.
* Entry Controller UI displays appropriate messages.
* Entry Controller resets correctly.
* A record of the ticket was not kept.

### Setup

* Carpark system is initialized. It will be allowed to hold 20 cars, 2 of which would be season ticket holders during business hours.
* Carpark may be full from executing script 1.2. If so, reset the carpark by closing and reopening it, or have a car exit by following script 1.2: steps 3 to 10. Record the ticket used in a new text file “Script 1.6 test data.txt”.

### Teardown

* No teardown is necessary after this script.

### Script Steps

| **Step #** | **Test Action** | **Expected Results** | **Pass/ Fail** |
| --- | --- | --- | --- |
| 1 | Detection of a car on outside sensor | Entry pillar UI (herein “UI”) displays “Push Button” | Pass |
| 2 | “Issue Adhoc Ticket” button pushed | Ticket is created and displayed by printer, UI displays “Take Ticket” | Pass |
| 3 | Ticket barcode copy-pasted to test results text file | File Script 1.6 Test Data.txt contains barcode of ticket. | Pass |
| 4 | “Take Ticket” button pushed | Gate is open, UI reads “Ticket Taken”. | Pass |
| 5 | Detection of a car on inside sensor | UI displays “Entering” | Pass |
| 6 | Detection of no car on outside sensor | UI displays “Entered” | Pass |
| 7 | Detection of a car on outside sensor | UI displays “Entering” | Pass |
| 8 | Detection of no car on inside sensor | UI displays “Take Ticket” | Pass |
| 9 | Detection of no car on outside sensor | Gate is closed, UI displays “Idle” | Pass |
| 7 | Ticket barcode pasted into Paystation ticket reader, Read Ticket button pressed | Paystaion UI displays “Take Rejected Ticket” | Fail |

### Test Execution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date/Time | Tester | Test ID | Test Phase | Status |
| 22/09/2017 1:10 pm | Ryan Smith | Rsmith1 | System Cycle 1 | Failed |