# Scenario #: Test PayStationController

## Scenario Description

* Make sure PayStationController is tested successfully

## Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version # | Date | Author | Description |
| 0.1 | 20/09/2017 | Sushila Khati | Initial Draft |

## Test Scripts

The following scripts will cover this scenario:

1. Check if the state equal STATE.IDLE
2. Check if the adhocTicket not equal to null
3. Check if ui beep then ticket rejected

## Use Case

* Pay for ticket

## Test Components/Requirements

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

* Show ticket state “IDLE”
* Show ticket state “Waiting”
* Show ticket state “Rejected”

### Testing Requirements

Test scripts covers the following specific testing requirements:

Displaying current ticket state “ IDLE“ and then displaying current ticket state “ Waiting“ and show ticket state “Rejected”

### Setup

* Create state equal state. IDLE case

State- == state. IDLE

* Create state equal state. Waiting case

State- == state. Waiting

* Create state equal state. Rejected case

State- == state. Rejected

### Teardown

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

### Script Steps

| **Step #** | **Test Action** | **Expected Results** | **Pass/ Fail** |
| --- | --- | --- | --- |
|  | Select state idle function | If equal to state display “IDLE” |  |
|  | Select state waiting function | If equal to state. Waiting display “waiting ” |  |
|  | Select state rejected function | If equal to state. rejected display “Rejected” |  |
|  | Select ticket taken function | Display the purchase ticket is taken |  |

### Test Execution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date/Time | Tester | Test ID | Test Phase | Status |
| 20/09/17 11:35 am | Sophie Taouk | Sophies1 | System Cycle 1 | Passed |
| 20/09/17 3:45 pm | Sushila Khati | Sushila2 | System Cycle 2 | passed |