# Traceability Matrix

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Category** | **Sub Category** | **Use Case** | **System Req.** | **Pass/Fail** |
| JJBB-01a | UI | Reservation Calender | Case 1 | 1 | Pass |
| JJBB-01b | UI | Reservation Calender | Case 1 | 2 | Pass |
| JJBB-02a | UI | Guest Information - Valid | Case 1, Scenario 3,4 | 3, 4 | Pass |
| JJBB-02b | UI | Guest Information - Invalid | Case 1, Scenario 1,2 | 3, 4 | Pass |
| JJBB-03 | UI | Reports | Case 2, Scenario 1 | 6 | Pass |
| JJBB-04 | System | Verify Deposit | Case 1, Scenario 4 | 4 | Pass |
| JJBB-05 | System | Format PDF | Case 2, Scenario 1 | 6 | Pass |
| JJBB-06 | System | Send to Manager | Case 2, Scenario 1 | 6 | Pass |
| JJBB-07 | System | Check Hold-By-Date | Case 1, Scenario 3 | 5 | Pass |
| JJBB-08 | System | Calender Status | Case 1 | 1 | Pass |
| JJBB-09 | Domain | Customer Creation | Case 1, Scenario 2,3,4 | 3 | Pass |
| JJBB-10 | Domain | Reservation Creation | Case 1, Scenario 2,3,4 | 2 | Pass |
| JJBB-11 | Domain | Account Creation | Case 1, Scenario 2,3,4 | 3, 4 | Pass |
| JJBB-12 | Domain | Reservation Report | Case 2 | 6 | Pass |
| JJBB-13 | Domain | Financial Report | Case 2 | 6 | Pass |

# System Requirements

# 1. User will check calendar for reservation availability on date(s)

# 2. Select Dates and Rooms for Reservation

# 3. Enter Customer, Contact and Credit Card Information

# 4. Determine Deposit Status

# 5. Check Reservation Hold-By Dates. If overdue, Drop Reservation

# 6. Monitor Expenses and Profits

# Use Cases and User Scenarios

Use Case 1: Make New Customer Reservation

A customer selects their reservation date(s), enters their customer details, contact information and credit card information and makes their deposit selection into the JJBB system. Once the customer has entered all of their required information they will submit their request and await confirmation.

Precondition: The JJBB system needs to be configured properly and have access to its Customer, Reservation and Accounting Databases. The system needs to be ready to accept a reservation request.

Post-Condition:

1. The system makes a reservation and responds to the customer with a confirmation.
2. The system does not make a reservation and requests the customer take corrective action. No reservation due to:
   1. Incorrect customer details/contact details/credit card information.
   2. Credit Card deposit does not go through

Actor Profile: A customer who desires to make a reservation for a date or range of dates. The customer can be new to the system or a returning customer.

Sequence of Events:

1. Customer selects date or range of dates
2. Customer enters customer information, contact information, credit card information
3. Customer submits a reservation requested
4. JJBB system checks for invalid user input
5. If conflict:
   1. JJBB system prompts user to resubmit request with correct information
6. JJBB system checks credit card deposit information
7. If conflict:
   1. JJBB system prompts user to resubmit request with correct information
8. The system adds customer to Customer Database, reservation to Reservation Database, account to Account database.
9. The system responds to customer with a successful reservation.

Use Case 2: Request Report

The management requests either a Reservation Report or a Financial Report from the JJBB system. Once a request is submitted, the system will output the queried report.

Precondition: The JJBB system needs to be configured properly and have access to its Customer, Reservation and Accounting Databases. The system needs to be ready to accept a reservation request.

Post-Condition: The system prints the requested report for the management.

Actor Profile: Management personnel who desires to review a JJBB reservation or financial report. The personnel is already setup to use the system.

Sequence of Events:

1. Management submits a request for a Report
2. JJBB system queries from the databases the required information and processes it into report formalize
3. The requested report is delivered to the management.

User Scenario 1 for Case 1: Customer Requests Reservation with Invalid User Input

1. Customer enters invalid information (either customer, contact or credit card information).
2. User submits the reservation requested.
3. JJBB system prompts the user to correct the invalid data entries and resubmit the request.

User Scenario 2 for Case 1: Customer Requests Reservation with Deposit Failure

1. Customer enters all required information correctly.
2. User submits the reservation request.
3. JJBB system prompts the user saying that the deposit has failed and to resubmit the request after corrections have been made.

User Scenario 3 for Case 1: Customer Requests Reservation without Deposit

1. Customer enters all required information correctly.
2. User submits the reservation request.
3. JJBB system prompts the user that the reservation has been made without guarantee to hold the rooms.

User Scenario 4 for Case 1: Customer Requests Reservation with Successful Deposit

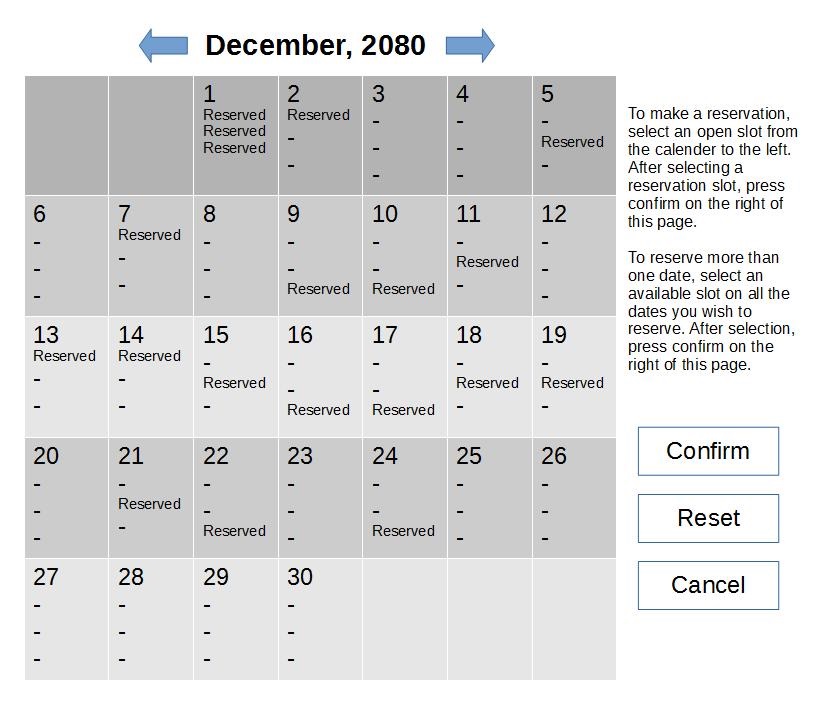
1. Customer enters all required information correctly.
2. User submits the reservation request.
3. JJBB system prompts the user that the reservation has been made.

User Scenario 1 for Case 2: Management Requests Report

1. Management submits a request for either a Reservation Report or Financial Reporting
2. JJBB system retrieves the requested data, formats it and prints it for the management.

# Test Case Specifications

## User Interface Test Cases

**Figure JJBB-01**

**Test Case ID**: JJBB-01a

**Test Type**: Black Box

**Test Objective**: The objective of this test case is to provide a test for the Reservation Calender UI module, to ensure that a reservation calender is appropriately displayed to the user and to see if month change repopulates the calender view appropriately. See Figure JJBB-01.

**Test Items**: Reservation Calender sub-module

**Input Specifications**

1. Execute the JJBB Customer system.
2. Click on the right facing arrow in the top of the calender page.
3. Click on the left facing arrow in the top of the calender page.

**Output Specifications**

1. Monthly calender is displayed correctly, with the month labeled at the top and the individual dates shown. Individual dates are populated with with “-” or “Reserved” vacancy status for each of the 3 rooms.
2. Monthly calender is incremented by 1 month. The month label at the top is updated. The individual dates are updated as with their reservation status.
3. Monthly calender is decremented by 1 month. The month label at the top is updated. The individual dates are updated as with their reservation status.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse

**Special Procedural Requirements**: None

**Inter-case Dependencies**: None

**Test Case ID**: JJBB-01b

**Test Type**: Black Box

**Test Objective**: The objective of this test case is to provide a test for the Reservation Calender UI module, to see if open reservation dates and rooms are selectable, to see if a range of dates are selectable, to see if reserved dates and rooms are not selectable, to test the “Reset”, “Confirm”, and “Cancel” buttons. See Figure JJBB-01.

**Test Items**: Reservation Calender sub-module

**Input Specifications**

1. Execute the JJBB Customer system.
2. Click on the “Cancel” button.
3. Repeat step 1.
4. Click on a single vacant room, designated with label “-”, on a single date.
5. Click on an additional vacant room on an adjacent date from the prior input step.
6. Click on the “Reset” button.
7. Click on a single reserved room, designated with label “Reserved”, on a single date.
8. Click on the “Confirm” button.
9. Repeat step 5.
10. Click on the “Confirm” button.

**Output Specifications**

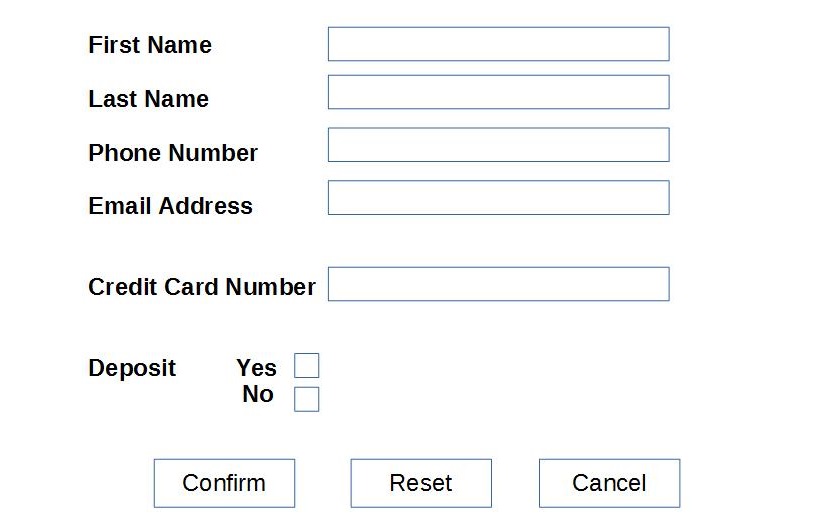
1. Monthly calender is displayed.
2. The JJBB system is terminated.
3. Monthly calender is displayed.
4. The single vacant room is highlighted.
5. The range of all vacant rooms selected are highlighted.
6. The vacant room selections are now deselected.
7. The single reserved room is not highlighted.
8. The JJBB system prompts the user with a message stating that no dates/rooms were selected.
9. The single vacant room is highlighted.
10. The JJBB system proceeds to the Guest Information UI sub-module.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse.

**Special Procedural Requirements**: None

**Inter-case Dependencies**: None

**Figure JJBB-02**

**Test Case ID**: JJBB-02a

**Test Type**: Black Box

**Test Objective**: The objective of this test case is to provide a test for the Guest Information UI module, to entering valid alphanumeric input into each of the text boxes, check the deposit tick boxes and to test the “Reset”, “Confirm”, and “Cancel” buttons. See Figure JJBB-02.

**Test Items**: Guest Information sub-module

**Input Specifications**

1. Enter “John” into the “First Name” text box.
2. Enter “Smith” into the “Last Name” text box.
3. Enter “5559876543” into the “Phone Number” text box.
4. Enter “[john.smith@customers.com](mailto:john.smith@customers.com)” into the “Email Address” text box.
5. Enter “1234123412341234” into the “Credit Card Number” text box.
6. Checkmark the “Deposit Yes” box.
7. Click on the “Reset” button.
8. Repeat step 1-6.
9. Click on the “Confirm” button.
10. Repeat step 1-5.
11. Checkmark the “Deposit No” box.
12. Click on the “Confirm” button.
13. Click on the “Cancel” button.

**Output Specifications**

1. Entering text into the “First Name”, “Last Name”, “Phone Number”, “Email Address” text boxes will display the entered text.
2. Clicking on the “Yes” Deposit tick box will check that box.
3. Clicking on the “No” Deposit tick box will check that box and uncheck the “Yes” Deposit tick box.
4. Clicking on the “Reset” button will erase all the entered text in the text boxes and untick the Deposit checkbox.
5. Clicking on the “Confirm” button with the “Deposit Yes” checkbox marked will prompt the user of a successful reservation.
6. Clicking on the “Confirm” button with the “Deposit No” checkbox marked will prompt the user of a successful reservation, with an additional message informing the customer that the reservation made is not guaranteed and the reservation will be dropped on the first day of the reservation OR will be overwritten if another customer with a deposit reserves that date and room.
7. Clicking on the “Cancel” button will send the user to the “Calender Reservation” module.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse and Keyboard.

**Special Procedural Requirements**: None

**Inter-case Dependencies**: JJBB-01b

**Test Case ID**: JJBB-02b

**Test Type**: Black Box

**Test Objective**: The objective of this test case is to provide a test for the Guest Information UI module, to entering invalid alphanumeric input into each of the text boxes. See Figure JJBB-02.

**Test Items**: Guest Information sub-module

**Input Specifications**

1. Click on the “Confirm” button.
2. Enter “John” into the “First Name” text box.
3. Enter “Smith” into the “Last Name” text box.
4. Enter “$%^#@%^” into the “Phone Number” text box.
5. Enter “”:)(\*&^%[@customers.com](mailto:john.smith@customers.com)” into the “Email Address” text box.
6. Enter “!@#$%^&\*()(\*&^%$” into the “Credit Card Number” text box.
7. Checkmark the “Deposit Yes” box.
8. Click on the “Confirm” button.

**Output Specifications**

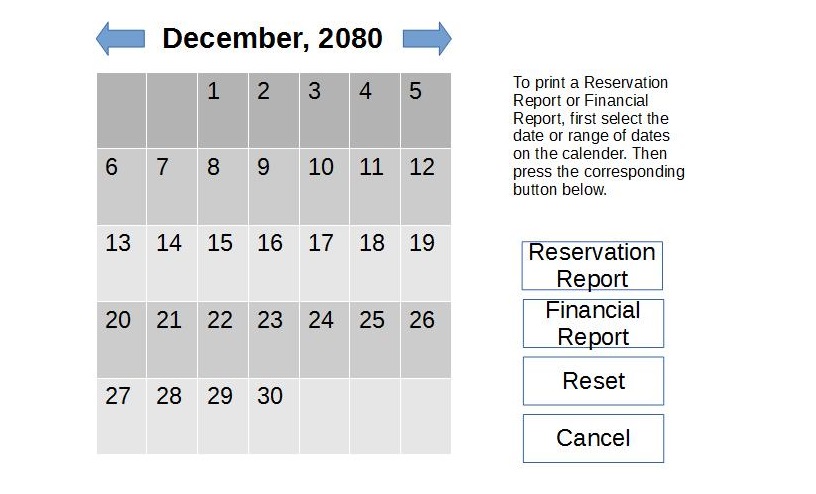
1. Clicking on the “Confirm” button with missing entries will prompt the user with a message stating which entries are missing.
2. Clicking on the “Confirm” button with invalid entries will prompt the user with a message stating which entries are invalid.
3. Clicking on the “Confirm” button with the “Deposit Yes” box checkmarked and if the deposit verification will result in a prompt saying that the deposit has failed and to correct the financial information.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse and Keyboard.

**Special Procedural Requirements**: None

**Inter-case Dependencies**: JJBB-01b

**Figure JJBB-03**

**Test Case ID**: JJBB-03

**Test Type**: Black Box

**Test Objective**: The objective of this test case is to provide a test for the Report UI module, to test the Calender selection for the month, date and range of dates, to test the “Reservation Report”, “Financial Report”, “Reset” and the “Cancel” buttons. See Figure JJBB-03.

**Test Items**: Report sub-module

**Input Specifications**

1. Execute the JJBB Manager system.
2. Click on the right facing arrow in the top of the calender page.
3. Click on the left facing arrow in the top of the calender page.
4. Click on a date.
5. Click on an adjacent date.
6. Click on the “Reservation Report” button.
7. Click on the “Financial Report” button.
8. Click on the “Reset” button.
9. Click on the “Cancel” button.

**Output Specifications**

1. Monthly calender is displayed correctly, with the month labeled at the top and the individual dates shown.
2. Monthly calender is incremented by 1 month. The month label at the top is updated. The individual dates are updated.
3. Monthly calender is decremented by 1 month. The month label at the top is updated. The individual dates are updated.
4. The single date is highlighted.
5. The range of all dates selected are highlighted.
6. The Reservation Report is now printed. This shows the reservation information for the range of dates selected.
7. The Financial Report is now printed. This shows the financial information for the range of dates selected.
8. The date selections are now deselected.
9. The JJBB system is terminated.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse.
3. Output devices: Printer.

**Special Procedural Requirements**: None

**Inter-case Dependencies**: None

**Test Case ID**: JJBB-04

**Test Type**: White Box

**Test Objective**: The objective of this test case is to provide a test for the Verify Deposit System Service module. It is used to determine, if the customer has selected to make a deposit, if the deposit will go through or not.

**Test Items**: Verify Deposit sub-module

**Input Specifications**

1. Execute the JJBB Customer system.
2. Select a single vacant room on a single date.
3. Click on the "Confirm" button.
4. Enter valid user information for all fields.
5. Checkmark the "Deposit Yes" box.
6. Click on the "Confirm" button.

**Output Specifications**

1. returnStatus() will pass a boolean value depending on a successful or unsuccessful reservation.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse and Keyboard

**Special Procedural Requirements**: This test case can only be tested indirectly from the JJBB-02a test case. There is no interface which allows direct verification of the deposit, as it is coupled together with the Guest Information UI module.

**Inter-case Dependencies**: JJBB-01b and JJBB-02a

**Test Case ID**: JJBB-05

**Test Type**: White Box

**Test Objective**: The objective of this test case is to provide a test for the Format PDF System Service module. It is used to determine if the information gathered by this service correctly produces a report in PDF form.

**Test Items**: Format PDF sub-module

**Input Specifications**

1. Execute the JJBB Manager system.
2. Click on a date.
3. Click on the "Reservation Report" button.

**Output Specifications**

1. formatPDF() will pull the requested information from the databases.
2. The pdf class will be instantiated and populated.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse.

**Special Procedural Requirements**: This test case can only be tested indirectly from the JJBB-03 test case. There is no interface which allows direct PDF formatting, as it is coupled together with the Reports UI module and the Send-To-Manager service module.

**Inter-case Dependencies**: JJBB-03

**Test Case ID**: JJBB-06

**Test Type**: White Box

**Test Objective**: The objective of this test case is to provide a test for the Send-To-Manager System Service module. It is used to determine if the PDF passed from the Format PDF sub-module is correctly sent to the management.

**Test Items**: Send-To-Manager sub-module

**Input Specifications**

1. Execute the JJBB Manager system.
2. Click on a date.
3. Click on the "Reservation Report" button.

**Output Specifications**

1. sendToUI() will pass the instantiated PDF object created in the Format PDF sub-module.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse.

**Special Procedural Requirements**: This test case can only be tested indirectly from the JJBB-05 test case, which is also dependent on the JJBB-03 test case. These three modules are tightly coupled together to form a seamless service for the management.

**Inter-case Dependencies**: JJBB-03, JJBB-05

**Test Case ID**: JJBB-07

**Test Type**: White Box

**Test Objective**: The objective of this test case is to provide a test for the Check Hold-By-Date System Service module. It is used to determine if a no-guarantee reservation will actually drop according to the System Specification.

**Test Items**: Check Hold-By-Date sub-module

**Input Specifications**

1. Execute the JJBB Customer system.
2. Select a single vacant room on a single date.
3. Click on the "Confirm" button.
4. Enter valid user information for all fields.
5. Checkmark the "Deposit No" box.
6. Click on the "Confirm" button.
7. Click on the "Cancel" button to exit.
8. Change current date forward to the reservation date on computer system with JJBB system installed.
9. Execute the JJBB Manager system.
10. Click on the reservation date that the computer system was changed to.
11. Click on the "Reservation Report" button.

**Output Specifications**

1. checkReservations() will call the deleteThis() function of the evaluated reservation object.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse and Keyboard.

**Special Procedural Requirements**: This test case can only be tested indirectly from JJBB-02a test case and also by JJBB-03 test case, by creating a no-despoit reservation and waiting until the day of the reservation, then requesting a Reservation Report of that day. These sub-modules are tightly coupled in ithis test case. The System Specification states that a reservation with no deposit is held without guarantee up until a specified date. For this implementation of this requirement, the reservation is held until the day of the reservation, and if it is still has no deposit, will be dropped.

**Inter-case Dependencies**: JJBB-02a and JJBB-03

**Test Case ID**: JJBB-08

**Test Type**: White Box

**Test Objective**: The objective of this test case is to provide a test for the Calender status System Service module. It is used to determine if a calender status check correctly shows a monthly calender with each date populated with the reservation statuses for each room on that date.

**Test Items**: Calender Status sub-module

**Input Specifications**

1. Execute the JJBB Customer system.

**Output Specifications**

1. sendCalender() function will send the reservation date data to the UI.
2. The calander[] array is the data that is passed to the Reservation Calender UI module.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse.

**Special Procedural Requirements**: This test case can only be tested indirectly from JJBB-01a or JJBB-01b test cases. Upon execution of the program, the Reservation Calender UI module will automatically call the Calender Status System Service module.

**Inter-case Dependencies**: JJBB-01a or JJBB-01b

**Test Case ID**: JJBB-09 through JJBB-11

**Test Type**: White Box

**Test Objective**: The objective of this test case is to provide a test for the Customer Creation, Reservation Creation, and Account Creation Domain Object modules. This is to simply verify that on reservation creation, the customer, reservation and account objects are properly instantiated and stored in the database.

**Test Items**: Customer Creation sub-module, Reservation Creation sub-module, Account Creation sub-module

**Input Specifications**

1. Execute the JJBB Customer system.
2. Select a single vacant room on a single date.
3. Click on the "Confirm" button.
4. Enter valid user information for all fields.
5. Checkmark the "Deposit Yes" box.
6. Click on the "Confirm" button.

**Output Specifications**

JJBB-09

1. getCustomerNo() called on that customer instance will return a valid customer number.

JJBB-10

1. getReservationNo() called on that customer instance will return a valid reservation number.

JJBB-11

1. getAccountNo() called on that customer instance will return a valid account number.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse and Keyboard.

**Special Procedural Requirements**: This test case can only be tested indirectly from JJBB-02a test case. There is no interface which allows direct verification of the these classes being instantiated.

**Inter-case Dependencies**: JJBB-01b and JJBB-02a

**Test Case ID**: JJBB-12 and JJBB-13

**Test Type**: White Box

**Test Objective**: The objective of this test case is to provide a test for the Reservation Report and Financial Report Domain Object modules. This is to simply verify that on Report Request, reservation and financial report objects are properly instantiated.

**Test Items**: Reservation Report sub-module, Financial Report sub-module

**Input Specifications**

1. Execute the JJBB Manager system.
2. Click on a date.
3. Click on the “Reservation Report” button.
4. Click on the “Financial Report” button.

**Output Specifications**

1. sendToUI() from the Send-To-Manager System Service module will pass the instantiated PDF object created in the Format PDF System Service module to the Report UI module.
   * This will be true for both Reservation Report and the Financial Report, test cases JJBB-12 and JJBB-13 respectively.

**Environmental Needs**

1. Computer system with JJBB system installed.
2. User input devices: Mouse.

**Special Procedural Requirements**: This test case can only be tested indirectly from the JJBB-03 test case. There is no interface which allows direct verification of the reservation and financial report classes being instantiated.

**Inter-case Dependencies**: JJBB-03, JJBB-05 and JJBB-06