**01 – Member Clean Accept No Books:**

Description: This test case will run the user through a perfect scenario with a user that has no fines or no loans and accept the loan agreement but without scanning a book

Pre-conditions: Test data is created within the program

Post-conditions: The user was informed that no books where scanned

Data required: member ID 1

**02 – Member Clean Accept Minimum Books:**

Description: This test case will run the user through a perfect scenario with a user that has no fines or no loans and accept the loan agreement

Pre-conditions: Test data is created within the program

Post-conditions: The user was able accept the loan of 1 book

Data required: member ID 1, Book ID(s) 10

**03 – Member Clean Accept Maximum Books:**

Description: This test case will run the user through a perfect scenario with a user that has no fines or no loans and accept the loan agreement

Pre-conditions: Test data is created within the program

Post-conditions: The user was able accept the loans of 5 books

Data required: member ID 1 Book ID(s) 10,11,12,13,14

**04 – Member Clean Cancel at Swipe:**

Description: This test case will run the user through a perfect scenario with a user that has no fines or no loans but cancel at the swipe screen

Pre-conditions: Test data is created within the program

Post-conditions: The user was able to cancel once swiped

Data required: member ID 1

**05 – Member Clean Cancel at Borrow:**

Description: This test case will run the user through a perfect scenario with a user that has no fines or no loans but cancel at the borrow screen

Pre-conditions: Test data is created within the program

Post-conditions: The user was able to cancel at the borrow screen

Data required: member ID 1

**06 – Member Clean Cancel After Scan:**

Description: This test case will run the user through a perfect scenario with a user that has no fines or no loans but cancels after a scan

Pre-conditions: Test data is created within the program

Post-conditions: The user was able to cancel after a scan with no residue data in the system

Data required: member ID 1, Book ID(s) 10

**07 – Member Clean Reject and Cancel:**

Description: This test case will run the user through a perfect scenario with a user that has no fines or no loans

Pre-conditions: Test data is created within the program

Post-conditions: The user was able to reject and cancel a loan with no residue data in the system

Data required: member ID 1 Book ID(s) 10

**08 – Book on Loan, Not Found, In List:**

Description: This test case will run the user through a perfect scenario with a user that has no fines or no loans and will test that a book is on loan, not found or already in the pending loan list.

Pre-conditions: Test data is created within the program

Post-conditions: The user attempted to borrow a book on loan, could not Find a book that does not exist and could not borrow a book twice

Data required: member ID 1, Book ID(s) 1, 16, 10, 10

**09 – Member Overdue:**

Description: This test case will ensure that a user that has overdue books cannot borrow

Pre-conditions: Test data is created within the program

Post-conditions: The user could not borrow a book

Data required: Member ID 2

**10 – Member Maxed Fines:**

Description: This test case will ensure that a user that has Maxed fines cannot borrow

Pre-conditions: Test data is created within the program

Post-conditions: The user could not borrow a book

Data required: Member ID 3

**11 – Member Maxed Loans:**

Description: This test case will ensure that a user that has reached the loan limit cannot borrow

Pre-conditions: Test data is created within the program

Post-conditions: The user could not borrow a book

Data required: Member ID 4

**12 – Member Acceptable Fines:**

Description: This test case will ensure that a user that has fines but has not reached the fine limit can still borrow

Pre-conditions: Test data is created within the program

Post-conditions: The user can borrow a book

Data required: Member ID 5, Book ID 10

**13 – Member Acceptable Loans:**

Description: This test case will ensure that a user that has current loans but has not reached the loan limit can still borrow

Pre-conditions: Test data is created within the program

Post-conditions: The user can borrow a book

Data required: Member ID 6, Book ID 10

**14 – Member Not Found:**

Description: This test case will check that a user that does not exist cannot borrow a book

Pre-conditions: Test data is created within the program

Post-conditions: The user could not access the system

Data required: member ID 9