**1 – “Happy Day” test:**

Description:

Evaluates the system under happy day conditions

Pre-conditions:

* BorrowBookCTL class exists
* memberDAO, loanDAO. bookDAO, display, cardReader, scanner, printer all exist
* BorrowBookCTL added as listener to cardReader and scanner
* BorrowBookCTL state == CREATED

Post-conditions:

* Main Menu is displayed
* All pending loans are committed and recorded
* Loan slip of committed loans printed
* cardReader is disable
* scanner is disabled
* BorrowBookCTL state == COMPLETED

Data required:

* Existing Member entity and DAO objects
* Existing Book entity and DAO objects
* Existing Loan DAO object

**2 – “Happy Day” but reject loan test**

Description:

Evaluates system behavior when pending loans are rejected

Pre-conditions:

* BorrowBookCTL class exists
* memberDAO, loanDAO. bookDAO, display, cardReader, scanner, printer all exist
* BorrowBookCTL added as listener to cardReader and scanner
* BorrowBookCTL state == CREATED

Post-conditions:

* Scanning panel of BorrowBookUI displayed
* Borrower details displayed
* Existing loan details displayed
* Pending loan list is empty
* Scan count == number of existing loans
* Cancel button enabled
* cardReader is disabled
* scanner is enabled
* BorrowBookCTL state == SCANNING\_BOOKS

Data required:

* Existing Member entity and DAO objects
* Existing Book entity and DAO objects
* Existing Loan DAO object

**3 – Overdue loans test**

Description:

Evaluates system behavior when member has overdue loans

Pre-conditions:

* BorrowBookCTL class exists
* memberDAO, loanDAO. bookDAO, display, cardReader, scanner, printer all exist
* BorrowBookCTL added as listener to cardReader and scanner
* BorrowBookCTL state == CREATED

Post-conditions:

* Restricted panel of BorrowBookUI displayed
* Cancel button enabled
* cardReader is disabled
* scanner is disabled
* Existing loans displayed
* Existing fine message displayed if relevant
* Overdue message displayed

Data required:

* Existing Member entity and DAO objects
* Existing Book entity and DAO objects
* Existing Loan DAO object

**3 – At loan limit test**

Description:

Evaluates system behavior when member is at loan limit

Pre-conditions:

* BorrowBookCTL class exists
* memberDAO, loanDAO. bookDAO, display, cardReader, scanner, printer all exist
* BorrowBookCTL added as listener to cardReader and scanner
* BorrowBookCTL state == CREATED

Post-conditions:

* Restricted panel of BorrowBookUI displayed
* Cancel button enabled
* cardReader is disabled
* scanner is disabled
* Existing loans displayed
* Existing fine message displayed if relevant
* At Loan limit message displayed

Data required:

* Existing Member entity and DAO objects
* Existing Book entity and DAO objects
* Existing Loan DAO object