

User Story 1: After launching the application, log in.

Inputs:

ID: text

Password: text

if ID && Password, userAuth == true

Expected Output: userAuth == true

Actual Output: ?

Pass/Fail Criterion: userAuth == true

User Story 2: After clicking the Search Book button, type in the required book name and click the Search button to display books

Inputs:

searchBook: text

book: found

booksArray: push each found book to array

Expected Output: array length >= 1

Actual Output: ?

Pass/Fail Criterion: Actual Output == Expected Output

User Story 3: User sends message to another user

Inputs:

messageInput: text

messages: array of texts

sendButton: click to push messageInput to messages

Expected Output: Message is pushed to array

Actual Output: ?

Pass/Fail Criterion: message is pushed to array

User Story 4: User uploads book to app

Inputs:

bookTitle: text

bookAuthor: text

physicalCondition: selection

bookPrice: number

coursePrefix: selection

courseSuffix: number

booksArray

uploadButton: click to add book to list of books for course

Expected Output: book is added to array

Actual Output: ?

Pass/Fail Criterion: Actual Output == Expected Output

User Story 5: User clicks into campus management portal

Inputs:

activeUsersButton: click to display activeUsersArray

pastTransactionsButton: click to display pastTransactionsArray

trustedDropoffsButton: click to display trustedDropoffsArray

ssoButton: click to display SSO Management screen

Expected Output: each website is rendered based on button function

Actual Output: ?

Pass/Fail Criterion: Actual Output == Expected Output

User Story 6: User clicks into Student PAYment Portal

Inputs:

paymentMethods: list of payment objects containing card/bank information

paymentMethodsButton: dropdown of previously used paymentMethods

addNewPaymentButton: pushes a new payment object into payment methods

previousTransactionsButton: displays array of previousTransactionsArray containing previousTransaction objects

previousTransaction: object containing book attributes and price, seller information, buyer information, dropOffLocation, check in time and check out time

Expected Output: each button renders the correct array of information

Actual Output: ?

Pass/Fail Criterion: Actual Output == Expected Output

User Story 7: User clicks into book drop off portal

Inputs:

zipCode: number

dropOffLocationButton: if zipCode is in certain range, display locations in range

dropOffHour: dropdown of numbers 1 - 12

dropOffMinute: dropdown of numbers 00:59

submitButton: adds dropoff object to the dropOffsArray

Expected Output: dropoff object is added to the dropoff array

Actual Output: ?

Pass/Fail Criterion: Actual Output == Expected Output