Test Plan

CS 4321 - Software Engineering 1

Group Name 3
Group Member Names: Amanda Mitchell, Brandon Corbett, Thara Larkins, Zackery Weeks
11/8/17

[You will use the category partition method to generate test specs for as many features as you can. Use the format below. This document will be stored in GitHub in a folder named *testing* stored just off the root]

1. Feature: Search for Book

choice()

1. UserInput
   a. Status
       i. Valid
       ii. Invalid   [error]

ISBNInput()

1. String Content
   a. Choices
       i. Letters   [error]
       ii. Spaces   [error]
       iii. Special Characters   [error]
       iv. Numbers
       v. Negative Numbers   [error]
       vi. Blanks   [error]

KeywordInput()

1. String Content
   a. Choices
       i. Letters
       ii. Letters w/ Spaces

iii. Special Characters   [error]
          iv. Numbers
          v. Negative Numbers   [error]
          vi. Blanks   [error]

1.1 Input File for TSL Program

#choice
UserInput Status:
        Valid.
        Invalid.   [error]

ISBNInput
  String Content:
        Choices:
                Letters.   [error]
                Spaces.   [error]
                Special Characters.   [error]
                Numbers.
                Negative Numbers.   [error]
                Blanks.   [error]

KeywordInput
  String Content:
      Choices:
          Letters.
          Letters w/ Spaces.
           Special Characters.   [error]
           Numbers.
           Negative Numbers.   [error]
           Blanks.   [error]

1.2 Test Specifications from TSL Program

Test Case 1   <error>
  UserInput Status : Invalid

Test Case 2   (Key = 1.)
  UserInput Status : Valid

Test Case 1   <error>
  Choices : Letters

Test Case 2   <error>
  Choices : Spaces

Test Case 3   <error>
  Choices : Special Characters

Test Case 4   <error>
  Choices : Negative Numbers

Test Case 5   <error>
  Choices : Blanks

Test Case 6   (Key = 4.)
  Choices   : Numbers

Test Case 1   <error>
  Choices : Special Characters

Test Case 2   <error>
  Choices : Negative Numbers

Test Case 3   <error>
  Choices : Blanks

Test Case 4   (Key = 1.)
  Choices   : Letters

Test Case 5   (Key = 2.)
  Choices   : Letters w/ Spaces

Test Case 6   (Key = 4.)
  Choices   : Numbers

2.   Feature: getCheckbookAvailable

getCheckbookAvailable()

1. ISBN
    a. isbn Status:
        i.   Available.
2. TableName
    a. TableName Status:
        i.   Valid.
        ii.   Invalid.   [error]
3. TableId
    a. TableId Status:
        i.   Valid.   [property bt.checkBookExists]
        ii.   Invalid.   [error]
    b. Inventory Exists:
        i.   Available.   [if bt.checkBookExists]
        ii.   Not Available.   [if !bt.checkBookExists]
4. TableISBN
    a. TableISBN Status:
        i.   Valid.
        ii.   Invalid.   [error]
5. TablePin
    a. TablePin Status:
        i.   Valid.   [property NULL]
        ii.   Invalid.   [error]
    b. Availability:
        i.   Available.   [if NULL]
        ii.   Not Available.   [if !NULL]

2.1 Input File for TSL Program

isbn
   isbn Status:
   Available.
tablenm
 tablenm Status:
   Valid.
   Invalid.   [error]
tableid
 tableid Status:
   Valid.   [property bt.checkBookExists]
   Invalid.   [error]
 Inventory Exists:
   Available.   [if bt.checkBookExists]
   Not Available.   [if !bt.checkBookExists]
tableisbn
 tableisbn Status:
   Valid.

4

```
        Invalid.          [error]
  # tablepin
    tablepin Status:
        Valid.            [property NULL]
        Invalid.          [error]
    Availability:
        Available.      [if NULL]
        Not Available.   [if !NULL]
```

2.2 Test Specifications from TSL Program

```
Test Case 1                    <error>
  tablenm Status :  Invalid


Test Case 2                    <error>
  tableid Status :  Invalid


Test Case 3                    <error>
  tableisbn Status :  Invalid


Test Case 4                    <error>
  tablepin Status :  Invalid


Test Case 5                    (Key = 1.1.1.1.1.1.1.)
  isbn Status     :  Available
  tablenm Status  :  Valid
  tableid Status  :  Valid
  Inventory Exists :  Available
  tableisbn Status :  Valid
  tablepin Status  :  Valid
  Availability    :  Available
```

3. Feature: CheckOutBook

```
  1.  checkbookid
        a.  checkbookid Status:
              i.    Available.
              ii.   On Hold.                        [property onHold]
              iii.  Invalid.                        [error]
  2.  pinnum
        a.  pinnum Status:
              i.    Valid.
```

5

    ii. Invalid.   [error]
   b. pinnum Num Books:
     i. 0-9.
     ii. 10.   [if !onHold]

3.1 Input File for TSL Program

```
# checkbookid
  checkbookid Status:
     Available.
     On Hold.                              [property onHold]
     Invalid.                    [error]
 # pinnum
  pinnum Status:
     Valid.
     Invalid.                    [error]
  pinnum Num Books:
     0-9.
     10.                                   [if !onHold]
```

3.2 Test Specifications from TSL Program

Test Case 1   <error>
 checkbookid Status : Invalid

Test Case 2   <error>
 pinnum Status : Invalid

Test Case 3   (Key = 1.1.1.)
 checkbookid Status : Available
 pinnum Status : Valid
 pinnum Num Books : 0-9

Test Case 4   (Key = 1.1.2.)
 checkbookid Status : Available
 pinnum Status : Valid
 pinnum Num Books : 10

Test Case 5   (Key = 2.1.1.)
 checkbookid Status : On Hold
 pinnum Status : Valid
 pinnum Num Books : 0-9

6

4. Feature: CheckUserPassword

checkUserPassword(user, pass)

1. user
    a. user Status:
        i. Valid.
        ii. Invalid.   [error]
2. pass
    a. pass Status
        i. Valid.
        ii. Invalid.   [error]
    b. Num Characters:
        i. 0.   [error]
        ii. 1-6.   [error]
        iii. 7.
        iv. >7.   [error]
    c. Character Values:
        i. special characters.   [error]
        ii. all capitals.
        iii. all lowercase.
        iv. all numbers.
        v. capitals, lowercase, numbers.
        vi. spaces.   [error]

4.1 Input File for TSL Program

user
 user Status:
        Valid.
        Invalid.   [error]
pass
 pass Status:
        Valid.
        Invalid.   [error]
 Num Characters:
        0.   [error]
        1-6.   [error]
        7.
        >7.   [error]
 Character Values:
        special characters.   [error]
        all capitals.
        all lowercase.
        all numbers.
        capitals, lowercase, numbers.
        spaces.   [error]

4.2 Test Specifications from TSL Program

Test Case 1   <error>
  user Status : Invalid

Test Case 2   <error>
  pass Status : Invalid

Test Case 3   <error>
  Num Characters : 0

Test Case 4   <error>
  Num Characters : 1-6

Test Case 5   <error>
  Num Characters : >7

Test Case 6   <error>
  Character Values : special characters

Test Case 7   <error>
  Character Values : spaces

Test Case 8   (Key = 1.1.3.2.)
  user Status   : Valid
  pass Status   : Valid
  Num Characters : 7
  Character Values : all capitals

Test Case 9   (Key = 1.1.3.3.)
  user Status   : Valid
  pass Status   : Valid
  Num Characters : 7
  Character Values : all lowercase

Test Case 10   (Key = 1.1.3.4.)
  user Status   : Valid
  pass Status   : Valid
  Num Characters : 7
  Character Values : all numbers

Test Case 11   (Key = 1.1.3.5.)
  user Status   : Valid
  pass Status   : Valid
  Num Characters   : 7
  Character Values : capitals, lowercase, numbers