Test Plan

Student Multi-Tool

2.25.2022

Team Marvel
Albert Toscano
Audrey Brio
Bradley Nickle
Devarsh Patel
Jacob Delgado
Joseph Cutri
Michael Kriesel (Team Leader)
SzeMan Tang

Version History

Date	Changes	Approval Date
10/18/2021	Initial Draft	
12/15/2021	Core Requirements + Updates	
2/4/2022	Updating for new features	
2/25/2022	Removing old Features	

Testing Schedule

Feature	Core/ Application	Start Test	End Test	Passed	Est Time (hours)
Logging/Archiving	Core	12/2/2021	12/7/2021	yes	4
User Management	Core	12/2/2021	12/7/2021	yes	4
Login	Core	3/1/2022	3/5/2022	n/a	4
Logout	Core	3/1/2022	3/5/2022	n/a	4
Error Handling	Core	3/1/2022	3/5/2022	n/a	4
Registration	Core	3/1/2022	3/5/2022	n/a	4
User Access Control	Core	3/1/2022	3/5/2022	n/a	4
Usage Analysis Dashboard	Core	3/1/2022	3/5/2022	n/a	4
Schedule of Classes	Application	3/1/2022	3/5/2022	n/a	4
Book Selling	Application	3/1/2022	3/5/2022	n/a	4
Schedule Builder	Application	3/1/2022	3/5/2022	n/a	4

Automated Moderating	Application	3/1/2022	3/5/2022	n/a	4
Team Code Review Testing	Both	3/1/2022	3/5/2022	n/a	40
Schedule Comparison	Application	4/10/2022	4/16/2022	n/a	6
Course Review	Application	4/10/2022	4/16/2022	n/a	6
Messaging	Application	4/10/2022	4/16/2022	n/a	6
Matching	Application	4/10/2022	4/16/2022	n/a	6
Event Planning	Application	4/10/2022	4/16/2022	n/a	6
Recipe Sharing	Application	4/10/2022	4/16/2022	n/a	6
Student Discounts	Application	4/10/2022	4/16/2022	n/a	6
Aid Eligibility	Application	4/10/2022	4/16/2022	n/a	6
Individual Code Review Testing	Both	4/10/2022	4/16/2022	n/a	56
Application Testing	Application	4/24/2022	4/27/2022	n/a	6
All System Test	Both	2/25/2022	4/27/2022	n/a	65

Testing														
User Management	Audrey													
Logging/Archiving	Albert													
Login	Audrey													
Logout	Audrey													
Error Handling	Joseph													
Registration	Albert													
User Access Control	Devarsh													
Usage Analysis Dashboard	Sze													
Book Selling	Jacob													
Schedule Builder	Brad													
Automated Moderating	Joseph													
Schedule of Classes	Michael													
Student Discounts	Albert													
Matching	Audrey													
Schedule Comparisons	Brad													
Recipe Sharing	Devarsh													
Messaging	Jacob													
Aid Eligibility	Joseph													
Course Rating	Michael													
Event Planning	Sze													
Deployment														
Set up Web Server	Michael													
Set up DataBase	Audrey													
Security Configuration	Albert													
Deployment	Brad													

Progression and Regression Testing

This section discusses when progression and regression testing will be done on our system

- Progression Testing
    - Test created as features are implemented
- Regression Testing
    - As features are implemented
    - Dedicated testing sprint near the end of the term whenever that is
    - After deployment, continuously

Testing Environment

Before deployment, our testing environment will consist of our personal computers. We will try to simulate the network detailed in our network diagram:

- For our datastore, we will use SQL Server 2019 and MariaDB instead of AWS Relational Database Service (AWS RDS).
- We will run our server software on our own devices instead of on AWS Elastic Compute Cloud (AWS EC2).
- We will run our client software locally in Google Chrome.
- We will be excluding the AWS Elastic Load Balancer (AWS ELB).

After deployment, our testing environment will shift to AWS. See our network diagram for more information.

Unit Testing will be performed on individual modules as they are created using Visual Studios. Inputs and Outputs will be validated

For failure cases, only one case needs to be true for the test to fail.

Test Cases

The following sections heavily reference the “Functional and NonFunctional Requirements” section of the Business Requirements Document (BRD).
https://github.com/mriheel/SeniorProject/blob/main/Milestone%201/CECS491A-BRD.pdf
The success and failure cases are listed in an exhaustive form, and will all be tested for if the test data and execution steps are followed to completion and verified properly.

Collaborative Schedule Builder Test Cases

1. The user creates a new schedule
    a. Requirements Tested
        i. SB01, SB02, SB03, SB04, SB05, SB06, SB07, SB08
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. Time data in this document will be written in 24-hour format, regardless of what is shown to the user
    d. Test Data
        i. Input
            1. One user account
            2. The following schedule items (SI’s):
                a. SI A
                    i. Type: Course
                    ii. Title: “Software Development with Frameworks”
                    iii. Start time: 14:00
                    iv. End time: 15:15
                    v. Days: Monday, Wednesday
                    vi. Category: “CECS”, Course number: “475”
                    vii. Section number: “01”
                b. SI B
                    i. Type: Commute
                    ii. Title: “Automatic Commute: to CECS475”
                    iii. Start time: 13:00
                    iv. End time: 13:30
                    v. Days: Monday, Wednesday

c. SI C
    i. Type: Commute
    ii. Title: “Automatic Commute: CECS475”
    iii. Start time: 15:30
    iv. End time: 16:00
    v. Days: Monday, Wednesday
d. SI D
    i. Type: Misc
    ii. Title: “Test: Studying”
    iii. Start time: 12:00
    iv. End time: 13:00
    v. Days: Monday, Tuesday, Wednesday, Friday, Saturday, Sunday

ii. Output
    1. True - all information is on schedule

e. Test Execution Steps
    i. User is on home page
    ii. User selects Build Schedules
    iii. The user clicks the new schedule button
    iv. The user adds SI A, SI B, SI C, and SI D to the schedule
    v. The user names the schedule
    vi. The user exits the edit view/page
    vii. The user reopens the schedule
    viii. The user examines each schedule item for consistency
f. Success Cases
    i. Upon reopening the schedule, all schedule items are the same as they were when first added
g. Failure Cases
    i. A schedule that the user did not select is opened for editing
    ii. Any schedule item cannot be added to the schedule
    iii. Upon reopening the schedule, any schedule item is missing, or has incorrect data

2. The user views a schedule
    a. Requirements Tested
        i. SB01, SB02
    b. Dependencies
        i. User must be logged in

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. One user account that owns a copy of both schedule A and schedule B
    ii. Output
        1. Schedule A displayed
        2. Schedule B displayed
e. Test Execution Steps
    i. User is on home page
    ii. User selects builds schedules
    iii. The user opens schedule A for editing
    iv. The user returns to the selection screen
    v. The user opens schedule B for editing
f. Success Cases
    i. Schedule A is opened first, then schedule B is opened
g. Failure Cases
    i. Schedule B is opened when schedule A is supposed to be opened, and vice versa

3. The user edits an existing schedule
    a. Requirements Tested
        i. SB01, SB02
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. The user should be able to create, update, and delete schedule items (referred to as SI's)
        iv. The user should be able to create, update, and delete inactive hours
        v. Schedule items can overlap with inactive hours
        vi. Time data in this document will be written in 24-hour format, regardless of what is shown to the user
    d. Test Data
        i. Input

1. One user account that owns schedule A
2. Test update string: " - updated"
3. The following schedule items (SI's):
    a. SI A
        i. Type: Course
        ii. Title: "Intro to Testing"
        iii. Start time: 09:00
        iv. End time: 10:15
        v. Days: Monday, Wednesday
        vi. Category: "CECS"
        vii. Course number: "123"
        viii. Section number: "01"
    b. SI B
        i. Type: Commute
        ii. Title: "Automatic Commute: to Intro to Testing"
        iii. Start time: 08:30
        iv. End time: 08:45
        v. Days: Monday, Wednesday
    c. SI C
        i. Type: Commute
        ii. Title: "Automatic Commute: from Intro to Testing"
        iii. Start time: 10:30
        iv. End time: 10:45
        v. Days: Monday, Wednesday
    d. SI D
        i. Type: Commute
        ii. Title: "Manual Commute: Test Drive"
        iii. Start time: 15:00
        iv. End time: 16:00
        v. Days: Tuesday
    e. SI E
        i. Type: Misc
        ii. Title: "Test: Studying"
        iii. Start time: 12:00
        iv. End time: 13:00
        v. Days: Monday, Tuesday, Wednesday, Friday,

Saturday, Sunday

ii. Output
    1. Schedule A with the following changes
        a. SI A
            i. Type: Course
            ii. Title: “Intro to Testing”
            iii. Start time: 08:30
            iv. End time: 09:45
            v. Days: Wednesday, Friday
            vi. Category: “MATH”
            vii. Course number: “123”
            viii. Section number: “02”
        b. SI B
            i. Type: Commute
            ii. Title: “Automatic Commute: to Intro to Testing”
            iii. Start time: 08:00
            iv. End time: 08:15
            v. Days: Wednesday, Friday
        c. SI C
            i. Type: Commute
            ii. Title: “Automatic Commute: from Intro to Testing”
            iii. Start time: 10:00
            iv. End time: 10:15
            v. Days: Wednesday, Friday
        d. SI D
            i. Type: Commute
            ii. Title: “Manual Commute: Test Drive”
            iii. Start time: 14:30
            iv. End time: 15:30
            v. Days: Thursday
        e. SI E
            i. Type: Misc
            ii. Title: “Test: Studying”
            iii. Start time: 11:30
            iv. End time: 12:30
            v. Days: Monday, Wednesday, Thursday,Friday,

Saturday, Sunday

e. Test Execution Steps
    i. User is on home page
    ii. User selects builds schedules
    iii. User opens schedule A for editing
    iv. Delete all of schedule A's original SI's
    v. Add SI A, SI B, SI C, SI D and SI E to the schedule
    vi. For SI A, update the category to "MATH" and update the section number to "02"
    vii. For all SIs:
        1. Append the test update string to the title
        2. Shift the start and end time forward by 1 hour
        3. Shift the start and end time back by 0.5 hours
    viii. For SI A, SI B, SI C:
        1. Remove Monday from the list of days
        2. Add Friday to the list of days
    ix. For SI D and SI E:
        1. Add Thursday to the list of days
        2. Remove Tuesday from the list of days
f. Success Cases
    i. All SI data match the specified output data
g. Failure Cases
    i. An error is raised, preventing any SI from being added to the schedule
    ii. Any SI is added to the schedule with erroneous data

4. Multiple users edit the same schedules at the same time
    a. Requirements Tested
        i. SB01, SB02, SB11, SB12, SB13
    b. Dependencies
        i. There must be at least two users to perform this test
        ii. All users must be logged in
        iii. All users used in this test must have permission to edit the same schedules, regardless of who owns the schedules
    c. Assumptions
        i. Users are connected to internet
        ii. Users do not lose connection to internet
        iii. Schedules used in this test will be owned by different users, although there can be multiple schedules owned by the same user

iv. Time data in this document will be written in 24-hour format, regardless of what is shown to the user

d. Test Data
    i. Input
        1. Two users, who will be referred to as “Alice” and “Bob”
        2. A copy of schedule A, which Alice owns and Bob can edit
        3. A copy of schedule B, which Bob owns and Alice can edit
        4. The following schedule items (SIs):
            a. SI A
                i. Type: Misc
                ii. Title: “Test: Studying”
                iii. Start time: 12:00
                iv. End time: 13:00
                v. Days: Tuesday
            b. SI B
                i. Type: Misc
                ii. Title: “Test: Practicing”
                iii. Start time: 14:00
                iv. End time: 15:30
                v. Days: Thursday
    ii. Output
        1. True - Correct information is changed and visible to both users
e. Test Execution Steps
    i. Alice and Bob are on the home page
    ii. Alice and Bob select builds schedules
    iii. Alice and Bob each select both schedules to be edited
    iv. Alice adds SI A to schedule B
    v. Bob adds SI B to schedule A
    vi. Alice updates the title of SI B
    vii. Bob moves the time of SI A forward by 1 hour
    viii. Alice deletes an SI (not SI A or SI B) from schedule B
    ix. Bob deletes an SI (not SI A or SI B) from schedule A
    x. Alice and Bob exit the schedule builder
    xi. Alice and Bob reopen schedule A to check that their changes have been made
    xii. Alice and Bob reopen schedule B to check that their changes have been made

f. Success Cases
    i. Upon reopening the schedules in the final step, all changes made during the test should still be visible to Alice and Bob
g. Failure Cases
    i. Either schedule fails to open or cannot be viewed by Alice and Bob simultaneously
    ii. Alice cannot see that Bob is also editing the schedules, or vice versa
    iii. Alice or Bob is prevented from making any change to either schedule
    iv. Upon reopening the schedules in the final step, Alice cannot see a change that Bob has made to the schedule, or vice versa

5. Give Permissions
    a. Requirements Tested
        i. SB09
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Test 1
            1. Input
                a. User Alice
                b. User Bob
                c. Permission - edit
                d. Schedule A
            2. Output
                a. True - Bob can edit schedule A
        ii. Test 2
            1. Input
                a. User Alice
                b. User Bob
                c. Permission - view
                d. Schedule B
            2. Output
                a. True - Bob can view schedule B
    e. Test Execution Steps
        i. Test 1

1. User is on home page
2. User selects builds schedules
3. Alice selects schedule A
4. Alice gives permissions to Bob to edit schedule A

ii. Test 2

1. User is on home page
2. User selects builds schedules
3. Alice selects schedule A
4. Alice gives permissions to Bob to view schedule B

f. Success Cases
   i. Bob can edit and view schedule A
   ii. Bob can view schedule B
g. Failure Cases
   i. Bob can not edit or view schedule A
   ii. Bob can not view Schedule B

6. Remove Permissions
   a. Requirements Tested
      i. SB10
   b. Dependencies
      i. User must be logged in
      ii. User must have granted permissions to another user
   c. Assumptions
      i. User is connected to internet
      ii. User does not lose connection to internet
   d. Test Data
      i. User Alice
      ii. User Bob
      iii. Schedule A
   e. Test Execution Steps
      i. User is on home page
      ii. User selects builds schedules
      iii. ALice selects schedule A
      iv. Alice removes permissions for Bob on schedule A
   f. Success Cases
      i. Bob can no longer view or edit schedule A
   g. Failure Cases
      i. Bob can still view or edit schedule A

Schedule Comparison Test Cases

1. The user selects two to five schedules to be compared
    a. Requirements Tested
        i. SC01, SC03
    b. Dependencies
        i. User must be logged in
        ii. User needs access to at least six schedules to fully perform this test
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Schedules A, B, C, D, and E
            2. A duplicate of schedule A
        ii. Output
            1. True or false, depending on whether or not each comparison is executed
    e. Test Execution Steps
        i. User is on home page
        ii. User selects compare schedules
        iii. For each integer n between 0 and 6(inclusive):
            1. The user selects n schedule(s)
            2. The user clicks the compare button
            3. The comparison is either executed or not
    f. Success Cases
        i. The user selects 2, 3, 4, or 5 schedules and the comparison is executed
        ii. The user selects 0, 1, or 6 schedule(s) and the comparison is not executed
    g. Failure Cases
        i. The user only selects 0, 1, or 6 schedule(s) and the comparison is executed
        ii. The user tries to select 2, 3, 4, or 5 schedules and the comparison is not executed
2. Two to five schedules are compared
    a. Requirements Tested
        i. SC01, SC03, SC04, SC05, SC06

b. Dependencies
    i. User must be logged in
    ii. User needs access to at least 5 schedules to fully perform this test
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
    iii. The number of schedules compared will only ever be between 2 and 5 (inclusive)
    iv. The user owns or has permission to edit schedules A, B, C, D, and E
    v. The scope of this test is only to test that the comparison logic works correctly
d. Test Data
    i. Input
        1. The following sets of schedules: {A, B}, {A, B, C}, {A, B, C, D}, and {A, B, C, D, E}
        2. The following lists of schedules: (A, A), (B, B, B), (C, C, C, C), and (D, D, D, D, D)
    ii. Output
        1. Compared schedules for {A,B}, {A,B,C}, {A,B,C,D}, {A,B,C,D,E}
        2. Compared schedules for (A, A), (B, B, B), (C, C, C, C), and (D, D, D, D, D)
e. Test Execution Steps
    i. User is on home page
    ii. User selects compare schedules
    iii. Each set or list of schedules is compared
    iv. Each result is checked for correct free time
f. Success Cases
    i. The resultant free time never overlaps with schedule items and inactive hours from any of the compared schedules
g. Failure Cases
    i. The resultant free time overlaps with any schedule item or inactive hours

Course Reviews Test Cases

1. Add a new review
    a. Requirements Tested
        i. CR02, CR03
    b. Dependencies
        i. User must be logged in
        ii. The schedule of classes for the school is up to date
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. The school is the one the user listed in their account
        iv. The classes and professors being entered are accurate
    d. Test Data
        i. Review A:
            1. Input:
                a. Major: Computer Science;
                b. Course: CECS228;
                c. Term: Fall 2021,
                d. Professor: Mehrnia,
                e. Comments: “This was a fun class”;
                f. Difficulty: 3/5
            2. Output: Review published successfully. Overall difficulty = 3/5
        ii. Review B:
            1. Input:
                a. Major: Computer Science;
                b. Course: CECS228;
                c. term: Fall 2021,
                d. professor: Mehrnia,
                e. comments: “This class was hard”;
                f. Difficulty: 5/5
            2. Output: Review published successfully. Overall difficulty = 4/5
        iii. Review C:
            1. Input:
                a. Major: Computer Science;
                b. Course: CECS228;
                c. term: Fall 2021,
                d. professor: Mehrnia,

            e. comments: “This class was easy”
        2. Output: Review failed to publish, no difficulty
    e. Test Execution Steps
        i. The user starts on the page to add a new review
        ii. The user selects the following information that comes from the corresponding test data above:
            1. User selects a major
            2. The user searches for and selects the class
            3. The user selects the professor
            4. The user adds comments about the class & professor
            5. The user inputs the desired difficulty rating
            6. The user publishes the review
        iii. This is repeated for all reviews
    f. Success Cases
        i. The user successfully publishes a new review, and it is able to be viewed from that courses page
        ii. The user successfully publishes a new review, and it is able to be viewed from that professors page
        iii. The overall difficulty average is adjusted appropriately according to the output listed in the review
    g. Failure Cases
        i. The user doesn’t input all required data
        ii. The class entered does not exist
2. Add suggested class pairing
    a. Requirements Tested
        i. CR04
    b. Dependencies
        i. The user must be logged in
        ii. The schedule of classes for the school is up to date
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. The school is the one listed in the user’s account information
    d. Test Data
        i. Pairing 1 (2 classes):
            1. Input:
                a. Major: Computer Science

b. CECS 326 - Xu - Spring 2021
c. CECS 327 - Luti - Fall 2021
d. Comments: 327 Builds off of 326

ii. Pairing 2 (3 classes):
1. Input:
a. Major: Computer Science
b. CECS 323 - Brown - Fall 2020
c. CECS 328 - Sharifian - Fall 2020
d. CECS343 - Hoffman - Spring 2021
e. Comments: 323 and 328 are useful for 343

iii. Pairing 3 (4 classes):
1. Input:
a. Major: Computer Science
b. CECS 451 - Moon - Fall 2020
c. CECS 465 - Whenlu - Spring 2021
d. CECS 453 - Nguyen - Spring 2021
e. CECS 478 - Giacalone - Fall 2020
f. Comments: These electives were all very interesting

iv. Pairing 4 (5 classes):
1. Input:
a. Major: Computer Science
b. CECS 174 - Nachawati - Fall 2018
c. CECS 274 - Ghazarian - Spring 2019
d. CECS 277 - Cleary - Fall 2019
e. CECS 282 - Gold - Fall 2019
f. CECS 228 - Mehrnia - Spring 2020
g. Comments: These classes are about the fundamentals of programming

v. Pairing 5 (Only one class):
1. Input:
a. Major: Computer Science
b. CECS 328 - Goldstein - Fall 2020
c. Comments: This class was very difficult
2. Output:
a. False - Review did not post, only one class

vi. Pairing 6 (6 classes):
1. Input:

a. Major: Computer Science
b. CECS 323 - Monge - Fall 2020
c. CECS 225 - Chelian - Spring 2020
d. CECS 282 - Gold - Spring 2020
e. CECS 326 - Xu - Fall 2020
f. MATH 122 - Johnson - Fall 2021
g. PHYS 151 - Klaehn - Fall 2019
h. Comments: These classes are not related at all

2. Output:
    a. False - Review did not post, too many classes

vii. Pairing 7 (same 3 classes):

1. Input:
    a. Major: Computer Science
    b. CECS 323 - Brown - Fall 2020
    c. CECS 328 - Sharifian - Fall 2020
    d. CECS343 - Hoffman - Spring 2021
    e. Comments: Taking 343 after 323 was very hard
2. Output:
    a. Linked to Pairing 2

viii. Pairing 8 (same 4 classes):

1. Input:
    a. Major: Computer Science
    b. CECS 451 - Moon - Fall 2020
    c. CECS 465 - Whenlu - Spring 2021
    d. CECS 453 - Nguyen - Spring 2021
    e. CECS 478 - Giacalone - Fall 2020
    f. Comments: These classes were fun
2. Output:
    a. Linked to Pairing 3

ix. Pairing 9 (3 classes):

1. Input:
    a. Major: Computer Science
    b. CECS 323 - Brown - Fall 2020
    c. CECS 328 - Sharifian - Fall 2020
    d. CECS343 - Hoffman - Spring 2021
    e. Comments:
2. Output:

a. False - Review did not post, missing comment

e. Test Execution Steps
    i. User is on home page
    ii. User selects Course Review
    iii. The user navigates to the ‘add suggested pairing’ page
    iv. The user selects the major (for all tests, Computer Science)
    v. The user searches for and adds all classes for this test
    vi. The user adds the comments
    vii. The user publishes the suggested pairing
    viii. Steps i-v are repeated for all test data
f. Success Cases
    i. The suggested pairing is published
    ii. A suggested pairing that already exists is linked to the existing one
g. Failure Case
    i. The user only selected one class
    ii. The user selected 6 or more classes
    iii. The user is missing information
    iv. One or more classes do not exist

3. Search for a review
    a. Requirements Tested
        i. CR01, CR06
    b. Dependencies
        i. User is logged in
        ii. The schedule of classes for the school is up to date
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. The school is the one in the user profile
        iv. All of the test data listed in Course Reviews Test Case 1 is already loaded into the database
    d. Test Data
        i. Test 1:
            1. Input: CECS 228
            2. Output: True - This class has a review
        ii. Test 2:
            1. Input: CECS 328
            2. Output: False - This class has no review

iii. Test 3:
    1. Input: CECS 333
    2. Output: False - This class does not exist
iv. Test 4:
    1. Input: Mehrnia
    2. Output: True - This professor has a review
v. Test 5:
    1. Input: Goldstein
    2. Output: False - This professor has no review
vi. Test 6:
    1. Input: Kriesel
    2. Output: False - This professor does not exist

e. Test Execution Steps
    i. User is on home page
    ii. User selects Course Review
    iii. User goes to search bar
    iv. The user enters major
    v. The user searches for the class or professor, determined by test case listed above
    vi. Validate that the desired class could be found
    vii. Repeat for all testing data
f. Success Cases
    i. The desired class is found
    ii. The desired professor is found
g. Failure Cases
    i. The class could not be found
    ii. No reviews exist for the class
    iii. The professor could not be found
    iv. No reviews exist for the professor

4. Search for class pairings
    a. Requirements Tested
        i. CR04
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet

iii. The database already contains class pairings from course review test case 2

d. Test Data
    i. Test 1 - Search for major:
        1. Input: Computer science
        2. Output: All Test 2 Test data
    ii. Test 2 - Search for a class:
        1. Input: CECS323
        2. Output: Test 2 Pairing 2
    iii. Test 3 - Search for class name
        1. Input: Database
        2. Output: Test 2 Pairing 2
    iv. Test 3 - Search for major:
        1. Input: Physics
        2. Output: No pairings found
    v. Test 4 - Search for a class:
        1. Input: CECS 100
        2. Output: No pairings found
e. Test Execution Steps
    i. User is on home page
    ii. User selects Course Review
    iii. The user searches for the major or class based on test data listed above
    iv. The user views the pairings
    v. Verify the pairings are the ones desired
    vi. Tests are repeated until all are done
f. Success Cases
    i. The user finds desired class pairings
g. Failure Cases
    i. No pairings are available

5. View Rating
    a. Requirements Tested
        i. CR02, CR06, CR07
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet

ii. User does not lose connection to internet
iii. School is same as listed in user settings
iv. The test data from Course Reviews test 1 (add a new review) is already loaded into the database

d. Test Data
    i. Test 1:
        1. Input: CECS 228
        2. Output:
            a. Overall difficulty - 4/5
            b. Comments: “This class was hard”, “This was a fun class”
    ii. Test 2:
        1. Input: Mehrnia
        2. Output:
            a. Overall difficulty - 4/5
            b. Comments: “This class was hard”, “This was a fun class”
e. Test Execution Steps
    i. User is on home page
    ii. User selects Course Review
    iii. The user navigates to the ‘find a review” page
    iv. The user enters their major
    v. The user searches for the class
    vi. The user selects the class
f. Success Cases
    i. The desired class is viewed with accurate information
    ii. The desired professor is viewed with accurate information
g. Failure Cases
    i. The class could not be found
    ii. The professor could not be found
    iii. The information in the review is inaccurate or not up to date

6. View Suggested Class Pairings
    a. Requirements Tested
        i. CR04
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet

iii. School is same as listed in setting
iv. The test data from test 2 is already loaded into the database

d. Test Data
    i. Test 1 - Search for major:
        1. Input: Computer science
        2. Output: All Test 2 Test data
    ii. Test 2 - Search for a class:
        1. Input: CECS323
        2. Output: Test 2 Pairing 2
e. Test Execution Steps
    i. The user navigates to the 'find suggested pairing' page
    ii. The user searches for and selects the desired pairing
    iii. Verify the information
f. Success Cases
    i. The desired pairing is viewed with accurate information
g. Failure Cases
    i. The pairing does not exist
    ii. The information in the suggested pairing is inaccurate

7. Like/Dislike
    a. Requirements Tested
        i. CR05
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. The school is the one in the users profile
        iv. School is same as listed in user settings
        v. The test data from course review Course Review test case 1 (add a new review) is already loaded into the database
    d. Test Data
        i. Test 1:
            1. Input: Like review a
            2. Output: Review A has 1 like added
        ii. Tset 2:
            1. Input: Dislike review b
            2. Output: Review B has 1 dislike added

e. Test Execution Steps
    i. The user searches for and selects a review
    ii. The user likes or dislikes it (depending on the test)
    iii. Verify the like/dislike counter matches the current test's output
f. Success Cases
    i. The class's like/dislike counter is adjusted appropriately
    ii. A review with a higher ratio of likes to dislikes is more relevant
g. Failure Cases
    i. The class's like/dislike counter is not adjusted
    ii. A review with a lower ratio of likes to dislikes is more relevant

Matching Test Cases

1. Activity Profile - Correct Information
    a. Requirements Tested
        i. MA01, MA02
    b. Dependencies
        i. User must be logged in
        ii. User is required to have at least one schedule created
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Activities selected: studying, get food (on/off campus), hang out
            2. Schedule A (from test schedules)
        ii. Output
            1. True - profile successfully created
    e. Test Execution Steps
        i. User is on home page
        ii. User selects matching profile
        iii. User selects schedule A
        iv. User selects create/edit activity profile
        v. User selects studying, get food (on/off campus), hang out as activities
        vi. User clicks save
    f. Success Cases
        i. User successfully created a matching profile for activities and the system can begin matching
        ii. User receives no matches if a possible match does not exist
    g. Failure Cases
        i. User profile was not created
2. Activity Profile - Incorrect Activity Information
    a. Requirements Tested
        i. MA01 and MA02
    b. Dependencies
        i. User must be logged in
        ii. User is required to have at least one schedule created
    c. Assumptions
        i. User is connected to internet

        ii. User does not lose connection to internet
    d. Test Data
        i. Test 1
            1. Input
                a. Activities selected: studying, get food (on/off campus), hang out, exercising, go to event, other
                b. Schedule A (from test schedules)
            2. Output
                a. False - profile not successfully created
        ii. Test 2
            1. Input
                a. Activities selected: none selected
                b. Schedule A (from test schedules)
            2. Output
                a. False - profile not successfully created
    e. Test Execution Steps
        i. Test 1
            1. User is on home page
            2. User selects matching profile
            3. User selects schedule A
            4. User selects create/edit activity profile
            5. User selects studying, get food (on/off campus), hang out, exercising, go to event, other as activities
            6. User clicks save
        ii. Test 2
            1. User is on home page
            2. User selects matching profile
            3. User selects schedule A
            4. User selects create/edit activity profile
            5. User does not select any activities
            6. User clicks save
    f. Success Cases
        i. User profile is not created
    g. Failure Cases
        i. User selects no activity or user selects more than 5 activities
        ii. A profile was created
3. Tutoring Profile - Correct Information

a. Requirements Tested
    i. MA03 and MA04
b. Dependencies
    i. User must be logged in
    ii. User is required to have at least one schedule created
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Test 1
        1. Input
            a. Schedule B
            b. Courses: CECS 328, MATH 122
            c. Group/Individual Setting: Individual
            d. Offers/Requires Tutoring: Requiring
        2. Output
            a. True - tutoring profile successfully created
    ii. Test 2
        1. Input
            a. Schedule B
            b. Courses: CECS 328, MATH 122
            c. Group/Individual Setting: Group
            d. Offers/Requires Tutoring: Requiring
        2. Output
            a. True - tutoring profile successfully created
    iii. Test 3
        1. Input
            a. Schedule B
            b. Courses: CECS 328, MATH 122
            c. Group/Individual Setting: Individual
            d. Offers/Requires Tutoring: Offering
        2. Output
            a. True - tutoring profile successfully created
    iv. Test 4
        1. Input
            a. Schedule B
            b. Courses: CECS 328, MATH 122

c. Group/Individual Setting: Group
d. Offers/Requires Tutoring: Offering

2. Output
    a. True - tutoring profile successfully created

e. Test Execution Steps
    i. Test 1
        1. User is on home page
        2. User selects matching profile
        3. User selects schedule B
        4. User selects create/edit tutoring profile
        5. User creates matching profile for tutoring
            a. Tutoring in courses CECS 328, MATH 122
            b. User requires tutoring
            c. User prefers individual tutoring
        6. User clicks save
    ii. Test 2
        1. User is on home page
        2. User selects matching profile
        3. User selects schedule B
        4. User selects create/edit tutoring profile
        5. User creates matching profile for tutoring
            a. Tutoring in courses CECS 328, MATH 122
            b. User requires tutoring
            c. User prefers group tutoring
        6. User clicks save
    iii. Test 3
        1. User is on home page
        2. User selects matching profile
        3. User selects schedule B
        4. User selects create/edit tutoring profile
        5. User creates matching profile for tutoring
            a. Tutoring in courses CECS 328, MATH 122
            b. User offers tutoring
            c. User prefers individual tutoring
        6. User clicks save
    iv. Test 4
        1. User is on home page

2. User selects matching profile
3. User selects schedule B
4. User selects create/edit tutoring profile
5. User creates matching profile for tutoring
    a. Tutoring in courses CECS 328, MATH 122
    b. User offers tutoring
    c. User prefers group tutoring
6. User clicks save

f. Success Cases
    i. User successfully created a matching profile for tutoring and the system can begin matching
    ii. A user can receive no matches if there does not exist another user who meets a match criteria
g. Failure Cases
    i. Profile not created

4. Tutoring Profile - Incorrect Course Information
    a. Requirements Tested
        i. MA03 and MA04
    b. Dependencies
        i. User must be logged in
        ii. User is required to have at least one schedule created
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Test 1
            1. Input
                a. Schedule B
                b. Courses: CECS 328, MATH 122, CECS 323, CECS 277, MATH 323, MATH 123, MUSC 468
                c. Group/Individual Setting: Individual
                d. Offers/Requires Tutoring: Requiring
            2. Output
                a. False - tutoring profile not successfully created
        ii. Test 2
            1. Input
                a. Schedule B

b. Courses: none entered
c. Group/Individual Setting: Group
d. Offers/Requires Tutoring: Offering

2. Output
    a. False - tutoring profile not successfully created

e. Test Execution Steps
    i. Test 1
        1. User is on home page
        2. User selects matching profile
        3. User selects schedule B
        4. User selects create/edit tutoring profile
        5. User creates matching profile for tutoring
            a. Tutoring in courses CECS 328, MATH 122, CECS 323, CECS 277, MATH 323, MATH 123, MUSC 468
            b. User requires tutoring
            c. User prefers individual tutoring
        6. User clicks save
    ii. Test 2
        1. User is on home page
        2. User selects matching profile
        3. User selects schedule B
        4. User selects create/edit tutoring profile
        5. User creates matching profile for tutoring
            a. User enters no course
            b. User offers tutoring
            c. User prefers group tutoring
        6. User clicks save
f. Success Cases
    i. User profile is not created
g. Failure Cases
    i. User does not enter any course or enters more than 6 courses
    ii. User profile is created

5. Tutoring Profile - Incorrect Requiring/Offering Information
    a. Requirements Tested
        i. MA03 and MA04
    b. Dependencies
        i. User must be logged in

ii. User is required to have at least one schedule created

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. Schedule B
        2. Courses: CECS 328, MATH 122
        3. Group/Individual Setting: Individual
        4. Offers/Requires Tutoring: no selection
    ii. Output
        1. False - tutoring profile not successfully created
e. Test Execution Steps
    i. User is on home page
    ii. User selects matching profile
    iii. User selects schedule B
    iv. User selects create/edit tutoring profile
    v. User creates matching profile for tutoring
        1. Tutoring in courses CECS 328, MATH 122
        2. User does not select if they require/offer tutoring
        3. User prefers individual tutoring
    vi. User clicks save
f. Success Cases
    i. User profile is not created
g. Failure Cases
    i. User does not select if they require or offer tutoring
    ii. User profile is created

6. Tutoring Profile - Incorrect Group/Individual Preference Information
    a. Requirements Tested
        i. MA03 and MA04
    b. Dependencies
        i. User must be logged in
        ii. User is required to have at least one schedule created
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data

        i. Input
            1. Schedule B
            2. Courses: CECS 328, MATH 122
            3. Group/Individual Setting: no selection
            4. Offers/Requires Tutoring: Requiring
        ii. Output
            1. False - tutoring profile not successfully created
    e. Test Execution Steps
        i. User is on home page
        ii. User selects matching profile
        iii. User selects schedule B
        iv. User selects create/edit tutoring profile
        v. User creates matching profile for tutoring
            1. Tutoring in courses CECS 328, MATH 122
            2. User does not select their preference for group or individual tyroing
            3. User requires tutoring
        vi. User clicks save
    f. Success Cases
        i. User profile is not created
    g. Failure Cases
        i. User does not select if they prefer group or individual tutoring
        ii. User profile is created
7. Profile - No Schedule Information
    a. Requirements Tested
        i. MA05
    b. Dependencies
        i. User must be logged in
        ii. User is required to have at least one schedule created
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. No schedule selected
        ii. Output
            1. False - cannot create or edit a profile

e. Test Execution Steps
    i. User is on home page
    ii. User selects matching profile
    iii. User does not select a schedule
    iv. User selects create/edit activity or tutoring profile
f. Success Cases
    i. User is not taken to page to create or edit profile
g. Failure Cases
    i. User does not enter a schedule
    ii. User cannot make or edit a profile

8. Select Match from Matching List
    a. Requirements Tested
        i. MA07, MA11, MA12
    b. Dependencies
        i. User must be logged in
        ii. User needs to have matching profile created
        iii. User needs to be opted in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. User is matched with at least one other user who has not blocked them
    d. Test Data
        i. Input
            1. List of matches: Alice, Bob, Claire, Dale, Emma
        ii. Output
            1. Reason for match - Studying, Exercising
            2. Schedule showcasing overlapping freetime - Schedule F
            3. Direct message option
    e. Test Execution Steps
        i. User is on home page
        ii. User opens matching list
        iii. User clicks on match - Alice
    f. Success Cases
        i. Correct reasons for matching and overlapping freetime schedule shows up after selecting match
    g. Failure Cases

        i. Incorrect information on why users were matched appears or incorrect information on overlapping freetime appears
        ii. No information on why users were matched appears or no information on overlapping freetime appears
        iii. Message button not displayed
9. Message Match
    a. Requirements Tested
        i. MA07
    b. Dependencies
        i. User must be logged in
        ii. User needs to have at least one match
        iii. User needs to be opted in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. User is matched with at least one other user who has not blocked them
    d. Test Data
        i. Input
            1. Match - Alice
            2. True - Message
        ii. Output
            1. Direct message with Alice
    e. Test Execution Steps
        i. User is on home page
        ii. User opens matching list
        iii. User clicks on match - Alice
        iv. User clicks on message
    f. Success Cases
        i. User is taken to direct message with Alice
    g. Failure Cases
        i. Clicking message button does not take user to direct message with Alice
        ii. User is taken to direct message with a different person other than Alice
10. Opt In
    a. Requirements Tested

i. MA09

b. Dependencies
    i. User must be logged in
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. Yes = true
        2. No = false
    ii. Output
        1. True - opted in
e. Test Execution Steps
    i. User is on home page
    ii. User opens matching profile
    iii. User clicks opt in
    iv. User clicks yes or
    v. User selects no instead
f. Success Cases
    i. If user selected yes, user is successfully included in matching again
    ii. If user selected no, nothing changes
g. Failure Cases
    i. Yes/no buttons do nothing
    ii. Information is not saved and the user still does not receive match information
    iii. Even if click no, information is changed

11. Opt Out
    a. Requirements Tested
        i. MA09
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Yes = true

2. No = false

ii. Output

1. True - opted out

e. Test Execution Steps

i. User is on home page

ii. User opens matching profile

iii. User clicks opt out

iv. User clicks yes or

v. User selects no instead

f. Success Cases

i. If user selected yes, user is successfully not included in matching process

ii. If user selected no, nothing changes

g. Failure Cases

i. Yes/no buttons do nothing

ii. Information is not saved and the user still receives match information

iii. Even if click no, information is changed

12. Filter Matches

a. Requirements Tested

i. MA08

b. Dependencies

i. User must be logged in

ii. User has received matches

c. Assumptions

i. User is connected to internet

ii. User does not lose connection to internet

iii. User has at least two matches

d. Test Data

i. Test 1

1. Input

a. List of matches - Alice, Bob, Claire, Dale, Emma, Francis, Greg, Hayley

b. Filter by date

2. Output

a. List of matches - Bob, Claire, Alice, Hayley, Greg, Francis, Dale, Emma

ii. Test 2

1. Input
    a. List of matches - Alice, Bob, Claire, Dale, Emma, Francis, Greg, Hayley
    b. Filter by activity matched
2. Output
    a. List of matches - Hayley, Emma, Dale, Greg, Alice, Francis, Claire, Bob

e. Test Execution Steps
    i. Test 1
        1. User is on home page
        2. User opens matching list
        3. User selects filter by date
        4. User selects save
    ii. Test 2
        1. User is on home page
        2. User opens matching list
        3. User selects filter by activity matched
        4. User selects save
f. Success Cases
    i. Matches are displayed in new way, order determined by filter chosen
g. Failure Cases
    i. Filter buttons do nothing
    ii. Matches are not displayed in new way
    iii. Matches are filtered in different way then chosen by users

13. Display List Of Matches
    a. Requirements Tested
        i. MA11, MA12
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. User has at least 2 matches
    d. Test Data
        i. Input
            1. List of Matches: Alice, Bob, Claire, Dale, Emma
        ii. Output

1. List displayed in no more than 10 seconds

e. Test Execution Steps
   i. User is on home page
   ii. User opens matching list
f. Success Cases
   i. List is displayed in less than 10 seconds
g. Failure Cases
   i. List took longer than 10 seconds to display

14. Update Profile
    a. Requirements Tested
       i. MA14
    b. Dependencies
       i. User must be logged in
       ii. User has already created a profile, they are editing it
    c. Assumptions
       i. User is connected to internet
       ii. User does not lose connection to internet
    d. Test Data
       i. Test 1
          1. Input
             a. Activities selected: exercising, get food (dining hall), hang out
             b. Schedule A (from test schedules)
          2. Output
             a. True - profile successfully updated in less than 1 second
       ii. Test 2
          1. Input
             a. Course: MATH 123, CECS 343
             b. Prefer group/Individual tutoring: Individual
             c. Offer/Requires Tutoring: Offers
             d. Schedule B
          2. Output
             a. True - profile successfully updated in less than 1 second
    e. Test Execution Steps
       i. Test 1
          1. User is on home page
          2. User opens matching profile

3. User selects schedule A
4. User selects create/edit activity profile
5. User edits activity profile by selecting new activities: exercising, get food (dining hall), hang out and deselecting previous activities
6. User selects save

ii. Test 2

1. User is on home page
2. User selects matching profile
3. User selects schedule B
4. User selects create/edit tutoring profile
5. User edits matching profile for tutoring
    a. Tutoring in courses CECS 328, MATH 122
    b. User requires tutoring
    c. User prefers individual tutoring
6. User clicks save

f. Success Cases
    i. Profiles are successfully updated to database in less than one second
g. Failure Cases
    i. Profiles take longer than one second to successfully update to the database

Student Discounts Test Cases

1. Post Student Discounts
    a. Requirements Tested
        i. DI01, DI06
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. The user will post a valid student discount
        ii. The user has the required student discount info
        iii. User is connected to internet
        iv. User does not lose connection to internet
    d. Test Data
        i. Post Establishment Discount A:
            1. Input:
                a. Name: Office Depot
                b. Address: 123 Main st. Long Beach, CA
                c. Discount description: 0.20 per copy for students
            2. Output:
                a. Display message: Discount posted successfully
                b. If there is already a record of name and address in the database, discount will be appended to that record
        ii. Post Establishment Discount B:
            1. Input:
                a. Name: Office Depot
                b. Address: “” (empty string)
                c. Discount description: 0.20 per copy for students
            2. Output:
                a. Display message: Discount failed to be posted. No address provided
        iii. Post Establishment Discount C:
            1. Input:
                a. Name: “” (empty string)
                b. Address: 123 Main st. Long Beach, CA
                c. Discount description: 0.20 per copy for students
            2. Output:
                a. Display message: Discount failed to be posted. No name provided

iv. Post Website Discount A:
    1. Input:
        a. Web address: www.officedepot.com
        b. Discount description: 0.20 per copy for students
    2. Output:
        a. Display message: Discount posted successfully
v. Post Website Discount B:
    1. Input:
        a. Web address: “” (empty string)
        b. Discount description: 0.20 per copy for students
    2. Output:
        a. Display message: Discount failed to be posted. No web address not provided

e. Test Execution Steps
    i. From the user home page, the user must click on student discounts button
    ii. User selects Post student Discounts
    iii. User selects either post establishment or post website
    iv. User enters the required info in the input fields
    v. User clicks Post
f. Success Cases
    i. The student discount posted by the user is added to our database
    ii. Users will be able to see their posted discounts in the search discount option
g. Failure Cases
    i. If the user does not enter input fields (name, address, and description for physical establishments or URL and description for websites), a popup error message will be displayed suggesting the user to enter the required field
    ii. If the user enters wrong information about a establishment or website discount, the record will be added to the database, but it will be incorrect

2. Search Student Discounts
    a. Requirements Tested
        i. Testing DI02 from BRD
        ii. Testing DI04 from BRD
        iii. Testing DI05 from BRD

iv. Testing DI09 from BRD

b. Dependencies
    i. User must be logged in
    ii. The database has records of student discounts
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
    iii. All records of student discounts are correct
d. Test Data
    i. Search establishment discount A:
        1. Input:
            a. User selects: Office Depot from the list
            b. Or User selects: Office Depot from the map
        2. Output:
            a. Display message: 0.20 per copy for students
            b. Get directions on the map
    ii. Search establishment discount B:
        1. Input:
            a. User selects: Office Depot from the list
            b. Or User selects: Office Depot from the map
        2. Output:
            a. Display message: “Error loading map for directions”
    iii. Search websites discount A:
        1. Input:
            a. User selects: www.officedepot.com from the list
        2. Output:
            a. Display message: 0.20 per copy for students
            b. Displays and Enables: “Go to website” button
            c. Displays and Enables: “votes” buttons
    iv. Search websites discount B:
        1. Input:
            a. User selects: www.officedepot.com from the list
        2. Output:
            a. Display message: “Error loading discount information”
e. Test Execution Steps
    i. From the user home page, the user must click on student discounts button

ii. Select search student discounts
iii. User selects either search establishment or search website
iv. If the user searches for establishments, there will be a list of establishments and a map marking where they are.
v. If the user searches for websites, a list of websites will be displayed.
vi. If user clicks on a item in the lists or in the map (for establishments),

f. Success Cases
    i. A list of establishments or websites (and a map for establishments) is displayed.
    ii. The items in the list of discounts will be displayed in order from most votes to least votes
g. Failure Case
    i. There are no records of student discounts in the database, therefore, no students discounts will be displayed

3. Vote for student discount
    a. Requirements Tested
        i. Testing DI03 from BRD
        ii. Testing DI07 from BRD
        iii. Testing DI08 from BRD
    b. Dependencies
        i. User must be logged in
        ii. The database must store all discounts' votes info
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. User's vote is correct
    d. Test Data
        i. Establishment/website vote A:
            1. Input:
                a. User clicks: Positive vote
            2. Output:
                a. Display message: positive vote selected
                b. Discount's positive votes increments by 1
        ii. Establishment/website vote B:
            1. Input:
                a. User clicks: Negative vote
            2. Output:

a. Display message: Negative vote selected
b. Discount's negative votes increments by 1

iii.

e. Test Execution Steps
    i. User is on home page
    ii. User selects student discounts
    iii. User searches for a discount
    iv. User selects a discount
    v. User votes for or against the student discount
f. Success Cases
    i. Votes from different users will be cumulative and they will be displayed on their respective student discounts
g. Failure Cases
    i. User tries to vote for the same student discounts before the 30 days period

4. Map of establishments
    a. Requirements Tested
        i. Testing DI04 from BRD
    b. Dependencies
        i. User must be logged in
        ii. User must search for establishments discounts
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. Establishments addresses are accurate
    d. Test Data
        i. Establishments' address A:
            1. Input:
                a. User clicks: Office Depot on map
            2. Output:
                a. Display message: 0.20 per copy for students
                b. Displays and Enables: "Get directions" button
                c. Displays and Enables: "votes" buttons
        ii. Establishments' address B:
            1. Input:
                a. User clicks: Office Depot on map
            2. Output:

a. Display message: “Error loading map”

e. Test Execution Steps
   i. User is on home page
   ii. User selects student discounts
   iii. User selects search student discounts
   iv. User selects search establishments
f. Success Cases
   i. Map of establishments is displayed on the right side of the web page
g. Failure Cases
   i. Map of establishments is not loaded
   ii. Establishments are not displayed on the map

5. List and description of student discounts
   a. Requirements Tested
      i. Testing DI02 from BRD
   b. Dependencies
      i. User must be logged in
      ii. User must search for establishments or websites discounts
   c. Assumptions
      i. User is connected to internet
      ii. User does not lose connection to internet
      iii. Establishments and websites information is correct
   d. Test Data
      i. List of student discounts A:
         1. Input:
            a. User clicks: Office Depot in the list or on the map of student discounts
         2. Output:
            a. Display message: Office Depot
               123 Main st. Long Beach, CA
               0.20 per copy for students
      ii. List of student discounts B:
         1. Input:
            a. User clicks: Office Depot in the list or on the map of student discounts
         2. Output:
            a. Display message: “Error loading discount information”

e. Test Execution Steps
    i. User is on home page
    ii. User selects student discounts
    iii. User selects search student discounts
    iv. User selects search establishments or websites
    v. User selects a establishment or website
f. Success Cases
    i. List of student discounts is displayed and the description of each item in the list is accurate
g. Failure Cases
    i. List of student discounts is not displayed
    ii. The description of student discounts is not accurate or does not match

Aid Eligibility Test Cases

1. View Aid Eligibility Matches
    a. Requirements Tested
        i. Testing AE05 from the BRD
    b. Dependencies
        i. User must be logged in
        ii. User must be connected to the internet
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to the internet
        iii. User has a valid .edu email address
        iv. User enters a valid email address and password
    d. Test Data
        i. Email of user A:
            1. Input:
                a. User's email: firstname.lastname@student.csulb.edu
                b. Password: valid password
            2. Output:
                a. Displays Aid Eligibility page
        ii. Email of user B:
            1. Input:
                a. User's email: firstname.lastname@yahoo.com
                b. Password: valid password
            2. Output:
                a. Displays error message: not valid email address
    e. Test Execution Steps
        i. User on home page
        ii. Open Aid Eligibility
        iii. Enter Username
        iv. Enter Password
        v. Click on Authenticate button
    f. Success Cases
        i. User enters correct ID and Password information
    g. Failure Cases
        i. User enter wrong ID and/or Password information
2. Enter Student's Information for Aid Eligibility Matches
    a. Requirements Tested

i. Testing AE01 from the BRD
ii. Testing AE02 from the BRD
iii. Testing AE03 from the BRD
iv. Testing AE04 from the BRD

b. Dependencies
    i. User must be logged in
    ii. User must be connected to the internet
c. Assumptions
    i. User is connected to the internet
    ii. User does not lose connection to the internet
    iii. User has valid outlook email (to prove they are registered)
d. Test Data
    i. User's A information:
        1. Input:
            a. Major: Computer Science
            b. Current GPA: 3.5
            c. Current income: 20000
            d. Veteran Status: Active
            e. Current semester units enrolled in: 12
            f. Level of Education/Number of degrees: Undergraduate
            g. User's age: 23
            h. Disability Status: No disability
            i. User is a first generation college student: Yes
            j. User is associated with other programs/organizations: Yes
            k. Residency: In-state
        2. Output:
            a. Displays: list of financial aid resources based on user's personal information
    ii. User's A information:
        1. Input:
            a. Major: Computer Science
            b. Current GPA: " " (No input)
            c. Current income: " " (No input)
            d. Veteran Status: " " (No input)
            e. Current semester units enrolled in: " " (No input)
            f. Level of Education/Number of degrees: " " (No input)

g. User's age: " " (No input)
h. Disability Status: " " (No input)
i. User is a first generation college student: " " (No input)
j. User is associated with other programs/organizations: " " (No input)
k. Residency: " " (No input)

2. Output:
    a. Displays: list of financial aid resources only where a requirement to qualify is having a major in Computer Science

e. Test execution steps
    i. User is on home page
    ii. User selects aid eligibility
    iii. Open Aid Eligibility
    iv. Enter Username
    v. Enter Password
    vi. Click on Authenticate button
f. Success Cases
    i. User Enters all information in the appropriate fields and nothing is left blank
    ii. After the user enters their required information they are shown their Aid Eligibility Matches
g. Failure Cases
    i. The user entered the wrong username or password
    ii. The user forgot to enter their username or password
    iii. Incorrect User information
    iv. The user isn't shown any aid, when aid items for them exist

3. Apply to Aid Eligibility Matches
    a. Requirements Tested
        i. Testing AE06 from the BRD
        ii. Testing AE07 from the BRD
        iii. Testing AE08 from the BRD
        iv. Testing AE09 from the BRD
        v. Testing AE10 from the BRD
    b. Dependencies
        i. User must be logged in
        ii. User must be connected to the internet

c. Assumptions
    i. User is connected to the internet
    ii. User does not lose connection to internet
d. Test Data
    i. User ID/Name and Password for the account is valid
    ii. Match A:
        1. Input:
            a. User selects: So Cal Scholarship
        2. Output:
            a. Displays So Cal Scholarship details
            b. Displays and enables the Apply button to be redirected to the website where the user can apply
    iii. Match B:
        1. Input:
            a. User selects: So Cal Scholarship
        2. Output:
            a. So Cal Scholarship information is not loaded
e. Test Execution Steps
    i. User is on home page
    ii. User selects aid eligibility
    iii. User clicks on matches after entering their information
    iv. User can view their Aid Eligibility matches
    v. User can sort their Aid Eligibility matches
    vi. User clicks on a match that he wants to read more about or apply to.
f. Success Cases
    i. Aid Eligibility matches have been populated for the user
    ii. User can apply to a selected match
g. Failure Cases
    i. No Aid Eligibility matches are visible

4. Aid Eligibility Filtering Results
    a. Requirements Tested
        i. Testing AE06 from the BRD
        ii. Testing AE07 from the BRD
        iii. Testing AE08 from the BRD
        iv. Testing AE09 from the BRD
        v. Testing AE10 from the BRD
    b. Dependencies

i. User must be logged in
ii. User must be connected to internet

c. Assumptions
    i. User is connected to the internet
    ii. User does not lose connection to the internet
d. Test Data
    i. Match List A:
        1. Input:
            a. User selects: sort by largest scholarship amount
        2. Output:
            a. List A will be sorted by largest scholarship amount to lowest scholarship amounts
            b. List A will be displayed
    ii. Match List B:
        1. Input:
            a. User selects: sort by lowest scholarship amount
        2. Output:
            a. List B will be sorted by lowest scholarship amount to largest scholarship amounts
            b. List B will be displayed
    iii. Match List C:
        1. Input:
            a. User selects: sort by nearest scholarship due date
        2. Output:
            a. List C will be sorted by nearest scholarship due date to farthest scholarship due date
            b. List C will be displayed
    iv. Match List D:
        1. Input:
            a. User selects: sort by farthest scholarship due date
        2. Output:
            a. List D will be sorted by farthest scholarship due date to nearest scholarship due date
            b. List D will be displayed
e. Test Execution Steps
    i. User is on home page
    ii. User selects aid eligibility

iii. User clicks on sorting their Aid matches in the increasing amount given
iv. User clicks on sorting their Aid matches in the decreasing amount given
v. User clicks on sorting by the decreasing due date of their Aid matches
vi. User clicks on sorting by the increasing due date of their Aid matches
vii. User clicks on sorting by special requirements (additional information needed)

f. Success Cases
    i. User is able to view the sorting of the increasing amount Aid matches
    ii. User is able to view the sorting of the decreasing amount Aid matches
    iii. User is able to view the sorting by decreasing due date of their Aid matches
    iv. User is able to view the sorting by increasing due date of their Aid matches
    v. User is able to view the sorting by special requirements (additional information needed)
g. Failure Cases
    i. User cannot view the proper sorting of the increasing amount of their Aid matches
    ii. User cannot view the proper sorting of the decreasing amount of their Aid matches
    iii. User cannot view the proper sorting of the decreasing due date of their Aid matches
    iv. User cannot view the proper sorting of the increasing due date of their Aid matches
    v. User is able to view the proper sorting by special requirements (additional information needed)

5. Aid Eligibility Removal of Expired Matches
    a. Requirements Tested
        i. Testing AE06 from the BRD
        ii. Testing AE07 from the BRD
        iii. Testing AE08 from the BRD
        iv. Testing AE09 from the BRD
        v. Testing AE10 from the BRD
    b. Dependencies
        i. User must be logged in

        ii. User must be connected to internet
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to the internet
    d. Test Data
        i. Once the due date of an Aid match has passed it will be deleted from the user's matches
        ii. No matches that are past the due date should be visible
    e. Test Execution Steps
        i. User is on home page
        ii. User selects aid eligibility
        iii. User should not be able to see expired Aid matches
    f. Success Cases
        i. User's expired Aid matches are automatically gone and are no longer viewed within their Aid matches
    g. Failure Cases
        i. User still has expired Aid matches visible
6. Aid Eligibility Updates of Matches
    a. Requirements Tested
        i. Testing AE06 from the BRD
        ii. Testing AE07 from the BRD
        iii. Testing AE08 from the BRD
        iv. Testing AE09 from the BRD
        v. Testing AE10 from the BRD
    b. Dependencies
        i. User must be logged in
        ii. User must be connected to internet
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to the internet
    d. Test Data
        i. There should never be no Aid Eligibility matches
    e. Test Execution Steps
        i. User is on home page
        ii. User selects aid eligibility
        iii. Ensure that there is at least one Aid Eligibility match
    f. Success Cases

        i. User should be able to view new matches on a monthly basis at least
    g. Failure Cases
        i. User does not see any new matches on a by monthly basis
7. Aid Eligibility information opens within 5 secs
    a. Requirements Tested
        i. Testing AE09 from the BRD
    b. Dependencies
        i. User must be logged in
        ii. User must be connected to internet
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to the internet
    d. Test Data
        i. Aid Eligibility matches should open within 5 secs
    e. Test Execution Steps
        i. User is on home page
        ii. User selects aid eligibility
        iii. Ensure that each page is loaded within a minimum of 5 secs
    f. Success Cases
        i. Every navigation page within the Aid Eligibility matches is loaded within a 5 sec time frame
    g. Failure Cases
        i. One or more pages take longer than 5 secs to load
8. Aid Eligibility populates to view within 10 secs
    a. Requirements Tested
        i. Testing AE10 from the BRD
    b. Dependencies
        i. User must be logged in
        ii. User must be connected to internet
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to the internet
    d. Test Data
        i. Data on each page of the Aid Eligibility needs to populate within a 10 secs time frame
        ii. The matches page should be able to populate within a 10 sec time frame

e. Test Execution Steps
    i. User is on home page
    ii. User selects aid eligibility
    iii. Click on each page to ensure that the 10 sec time frame is met to populate each page
f. Success Cases
    i. Every page clicked is loaded and populated with the data in a 10 sec time frame
g. Failure Cases
    i. One or more pages takes longer than 10 secs to load and populate the data

Messaging Test Cases

9. Direct message
    a. Requirements Tested
        i. ME01 and ME03
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. User Alice
            2. User Bob
        ii. Output
            1. Alice can send and receive messages to Bob
            2. Bob can send and receive messages to Alice
    e. Test Execution Steps
        i. User is on home page
        ii. User opens messages
        iii. Alice selects messages with Bob or starts a new direct message with Bob
        iv. Alice sends message to Bob
        v. Bob sends message to Alice
    f. Success Cases
        i. Both users can send and receive messages from each other
        ii. Messages are only visible to users in the direct message
    g. Failure Cases
        i. Past messages do not display
        ii. Message is not visible to both users
        iii. Message is visible to another user not included in the original two users
        iv. New message option won't display
10. Group Message
    a. Requirements Tested
        i. ME02
    b. Dependencies
        i. User must be logged in

c. Assumptions
    i. User is connected to the internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. User Alice
        2. User Bob
        3. User Claire
    ii. Output
        1. Alice can send and receive messages from Bob and Claire
        2. Bob can send and receive messages from Alice and Claire
        3. Claire can send and receive messages from Alice and Bob
e. Test Execution Steps
    i. User is on home page
    ii. User opens messages
    iii. Alice selects group message with Bob and Claire or starts a new direct message with Bob and Claire
    iv. Alice sends message to group
    v. Bob sends message to group
    vi. Claire sends message to group
f. Success Cases
    i. All users can send and receive messages from each other
    ii. Messages are only visible to users in the group message
g. Failure Cases
    i. Past messages do not display
    ii. Message is not visible to all users
    iii. Message is visible to another user not included in the group
    iv. New message option won't display

11. Group Message - Too many Users
    a. Requirements Tested
        i. ME02
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to internet
    d. Test Data

        i. Input
            1. User Alice
            2. User Bob
            3. User Claire
            4. User Dale
            5. User Emma
        ii. Output
            1. False - Group message could not be created
    e. Test Execution Steps
        i. User is on home page
        ii. User opens messages
        iii. Alice tries to start a new direct message with Bob, Claire, Dale, and Emma
    f. Success Cases
        i. Group message is not created
    g. Failure Cases
        i. Number of users exceeds limit to how many users allowed in group chat
12. Add Contacts List
    a. Requirements Tested
        i. ME05
    b. Dependencies
        i. User must be logged in
        ii. Knowledge of username
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Username - bobdylan
        ii. Output
            1. True - contact added
    e. Test Execution Steps
        i. User is on home page
        ii. User opens messages
        iii. User selects add new contact
        iv. User enters contact information - bobdylan

v. User selects save

f. Success Cases
    i. New contact is successfully saved

g. Failure Cases
    i. New page does not pop up
    ii. Information cannot be entered
    iii. Information is not saved

13. Remove Contacts List
    a. Requirements Tested
        i. ME05
    b. Dependencies
        i. User must be logged in
        ii. Knowledge of username
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Username - bobdylan
        ii. Output
            1. True - contact removed
    e. Steps
        i. User is on home page
        ii. User opens messages
        iii. User opens contacts list
        iv. User selects contact bobdylan
        v. User selects remove contact
        vi. User selects save
    f. Success Cases
        i. Contact is successfully deleted
    g. Failure Cases
        i. Contact page does not pop up
        ii. User can not click button
        iii. Information is not saved
14. Block Users
    a. Requirements Tested
        i. ME06

b. Dependencies
    i. User must be logged in
c. Assumptions
    i. User is connected to the internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. Username - bobdylan
    ii. Output
        1. True - contact blocked
e. Test Execution Steps
    i. User is on home page
    ii. User opens messages
    iii. User clicks on username - bobdylan
    iv. User selects block user
    v. User selects save
f. Success Cases
    i. User is successfully blocked
    ii. New messages are no longer available to see with newly blocked user
g. Failure Cases
    i. No window allowing user to block shows up
    ii. User cannot click button
    iii. Information is not saved
    iv. Can still see new messages with blocked user after they blocked the user

15. Unblock Users
    a. Requirements Tested
        i. ME06
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Username - bobdylan
        ii. Output

1. True - contact unblocked

e. Test Execution Steps
   i. User is on home page
   ii. User opens settings
   iii. User goes to blocked list
   iv. User selects username - bobdylan
   v. User selects unblock user
   vi. User selects save
f. Success Cases
   i. User is successfully unblocked
g. Failure Cases
   i. No window allowing user to unblock shows up
   ii. User cannot click button
   iii. Information is not saved

16. Leave Group
    a. Requirements Tested
       i. ME02
    b. Dependencies
       i. User must be logged in
       ii. User needs to be in at least one group message
    c. Assumptions
       i. User is connected to the internet
       ii. User does not lose connection to internet
    d. Test Data
       i. Input
          1. Group message with Alice, Bob, Claire, and Dale
       ii. Output
          1. Group message with Alice, Bob, and Claire
    e. Test Execution Steps
       i. User is on home page
       ii. User opens messages
       iii. User clicks on group message with Alice, Bob, Claire, and Dale
       iv. Dale selects option to leave group
       v. User selects yes
    f. Success Cases
       i. User successfully leaves the group and does not receive messages from that group anymore

g. Failure Cases
    i. Group message does not pop up
    ii. Users cannot select button
    iii. Information not saved
    iv. User still receives messages from group

17. Search
    a. Requirements Tested
        i. ME04
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Username - bobdylan
        ii. Output
            1. List of search matches - bobdylan
    e. Test Execution Steps
        i. User is on home page
        ii. User opens messages
        iii. User goes to search option
        iv. User enters username bobdylan
        v. User selects search
    f. Success Cases
        i. Desired username pops up
    g. Failure Cases
        i. Information cannot be entered
        ii. No username pops up
18. Character Length
    a. Requirements Tested
        i. ME07
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to the internet
        ii. User does not lose connection to internet

d. Test Data
    i. Test 1
        1. Input
            a. Message to send - “According to all known laws of aviation, there is no way a bee should be able to fly. Its wings are too small to get its fat little body off the ground.”
        2. Output
            a. True - message under 1000 characters, message sends
    ii. Test 2
        1. Input
            a. Message to send - “Once upon a time there was a lovely princess. But she had an enchantment upon her of a fearful sort which could only be broken by love's first kiss. She was locked away in a castle guarded by a terrible fire-breathing dragon. Many brave knights had attempted to free her from this dreadful prison, but none prevailed. She waited in the dragon's keep in the highest room of the tallest tower for her true love and true love's first kiss. (Laughs, tears out a page of the book) Like that's ever gonna happen. What a load of - (toilet flush). Shrek exits an outhouse and goes about his day in the swamp. In a nearby village, an angry mob gather up to go after Shrek. At night, the villagers head into the swamp and wait outside Shrek's home. Think it's in there? All right. Let's get it! Whoa. Hold on. Do you know what that thing can do to you? Yeah, it'll grind your bones for it's bread. Shrek sneaks up behind them and laughs. Yes, well, actually, that would be a giant. Now, ogres, oh they're much worse. They'll make a suit from your freshly peeled skin...”
        2. Output
            a. False - message over 1000 characters, does not send
e. Test Execution Steps
    i. Test 1
        1. User is on home page
        2. User opens messages

3. User goes to send message to Bob
4. User types message from test data test 1 input
5. User presses send

ii. Test 2

1. User is on home page
2. User opens messages
3. User goes to send message to Bob
4. User types message from test data test 2 input
5. User presses send

f. Success Cases
   i. Messages under 1000 characters are sent
g. Failure Cases
   i. Messages over 1000 characters do not get sent

19. Encrypted messaging
    a. Requirements Tested
       i. ME08
    b. Dependencies
       i. User must have chatted with someone
    c. Assumptions
       i. User has already messaged a different user
    d. Test Data
       i. Test 1
          1. Input
             a. Query database for specific message
          2. Output
             a. Illegible text should appear
    e. Test Execution Steps
       i. Test 1
          1. User queries message database
    f. Success Cases
       i. Messages are shown as random text
    g. Failure Cases
       i. Messages are able to be read
20. Auto reconnect
    a. Requirements Tested
       i. ME09
    b. Dependencies

i. User must be logged in

c. Assumptions
    i. User is connected to the internet
    ii. User drops connection to internet, but goes back online
d. Test Data
    i. Test 1
        1. Input
            a. Message to send - “According to all known laws of aviation, there is no way a bee should be able to fly. Its wings are too small to get its fat little body off the ground.”
        2. Output
            a. True - user reconnects, message sends
    ii. Test 2
        1. Input
            a. Message to send - “Once upon a time there was a lovely princess. But she had an enchantment upon her of a fearful sort which could only be broken by love's first kiss. She was locked away in a castle guarded by a terrible fire-breathing dragon. Many brave knights had attempted to free her from this dreadful prison, but none prevailed. She waited in the dragon's keep in the highest room of the tallest tower for her true love and true love's first kiss. (Laughs, tears out a page of the book) Like that's ever gonna happen. What a load of - (toilet flush). Shrek exits an outhouse and goes about his day in the swamp. In a nearby village, an angry mob gather up to go after Shrek. At night, the villagers head into the swamp and wait outside Shrek's home. Think it's in there? All right. Let's get it! Whoa. Hold on. Do you know what that thing can do to you? Yeah, it'll grind your bones for it's bread. Shrek sneaks up behind them and laughs. Yes, well, actually, that would be a giant. Now, ogres, oh they're much worse. They'll make a suit from your freshly peeled skin...”
        2. Output
            a. False - user reconnects, message does not send

e. Test Execution Steps
    i. Test 1
        1. User is on home page
        2. User opens messages
        3. User goes to send message to Bob
        4. User types message from test data test 1 input
        5. User disconnects
        6. User presses send
    ii. Test 2
        1. User is on home page
        2. User opens messages
        3. User goes to send message to Bob
        4. User types message from test data test 2 input
        5. User disconnects
        6. User presses send
f. Success Cases
    i. Messages are sent after reconnecting
g. Failure Cases
    i. Messages do not get sent after reconnecting

Schedule of Classes Test Cases

21. Obtaining course information from school databases
    a. Requirements Tested
        i. CI01, CI02, CI06, CI07
    b. Dependencies
        i. IPs or URLs to obtain the course information from
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. Assumes that we have direct access to query the school's database for courses
        iv. When testing is ready to begin, course sections used as test data will be manually checked for changes and new sections will be added to the test data as needed
        v. The test data will be updated to include course sections from summer 2022 and/or fall 2022 as they become available
    d. Test Data
        i. The following courses from the spring 2022 catalogue:
            1. CECS 328 "Algorithms", Section 01, MW, 09:30-10:45, "Staff"
            2. ENG 100A "Composition I", Section 07, TuTh, 08:00-09:15, "Staff"
            3. MATH 247 "Introduction Linear Algebra", Section 03, MW, 12:30-13:45, "Wu B"
            4. PHYS 151 "Mechanics and Heat", Section 10, Tu, 10:00-12:45, "Luna C"
    e. Test Execution Steps
        i. Send the necessary request to obtain a copy of the data and save it to our database
        ii. Check for the presence of the test courses in
    f. Success Cases
        i. All test data course sections are present in the database
    g. Failure Cases
        i. Any test data course section is missing from the database
22. Searching for a particular course
    a. Requirements Tested
        i. CI03, CI04

b. Dependencies
    i. User must be logged in
    ii. Courses from the user's school must already have been obtained from the school
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
    iii. This test case's steps can be repeated to test the search feature of the schedule builder and the course difficulty ratings features
    iv. When testing is ready to begin, course sections used as test data will be manually checked for changes and new sections will be added to the test data as needed
    v. The test data will be updated to include course sections from summer 2022 and/or fall 2022 as they become available
d. Test Data
    i. The following courses from the spring 2022 catalogue:
        1. CECS 328 "Algorithms", Section 01, MW, 09:30-10:45, "Staff"
        2. ENG 100A "Composition I", Section 07, TuTh, 08:00-09:15, "Staff"
        3. MATH 247 "Introduction Linear Algebra", Section 03, MW, 12:30-13:45, "Wu B"
        4. PHYS 151 "Mechanics and Heat", Section 10, Tu, 10:00-12:45, "Luna C"
    ii. The following course, which does not exist: CECS 000, "Does Not Exist", Section 00, MW, 00:00-01:15, "Staff"
e. Test Execution Steps
    i. The user navigates to the schedule builder and searches for each test course
    ii. The user searches for CECS 000
    iii. The user navigates to the course difficulty rating feature and searches for both test courses
    iv. The user searches for CECS 000
f. Success Cases
    i. Searching for each course yields that course as a search, excluding CECS 000
g. Failure Case
    i. Searching for either test course yields no results

ii. CECS 000 is present in search results

Automated Moderating Test Cases

23. Automated Moderating - Blacklisted term is found in a message
    a. Requirements Tested
        i. AM01, AM04, AM05, AM06, AM07, AM08
    b. Dependencies
        i. Messaging
    c. Assumptions
        i. There is at least one blacklisted term in the tested message.
    d. Test Data
        i. Input
            1. “Hello world! Fuck”
            2. “Hello world! Fuck you”
            3. “Hello world! Fuck y0u”
        ii. Output
            1. “Hello world! ****”
            2. “Hello world! ********”
            3. “Hello world! ********”
    e. Test Execution Steps
        i. For each input:
            1. User types the message
            2. User sends the message
    f. Success Cases
        i. The user’s message is censored
    g. Failure Cases
        i. The user’s message is not censored
        ii. Any term not on the blacklist is found but not censored
        iii. The blacklisted term is found but not censored
        iv. The blacklisted term is not found
        v. The system goes offline
24. Automated Moderating - Blacklisted term is not found in a message
    a. Requirements Tested
        i. AM01, AM04, AM05, AM06, AM07, AM08
    b. Dependencies
        i. Messaging
    c. Assumptions
        i. There are no blacklisted terms in the tested message.
    d. Test Data

i. Input
    1. “How’s it going?”
ii. Output
    1. “How’s it going?”

e. Test Execution Steps
    i. For each input:
        1. User types the message
        2. User sends the message
f. Success Cases
    i. The user’s message is not censored
g. Failure Cases
    i. Any part of the user’s message is censored

25. Automated Moderating - Warning is displayed when the user tries to create a post - Success
    a. Requirements Tested
        i. AM01, AM02, AM03, AM05, AM06, AM07, AM08
    b. Dependencies
        i. Recipe Sharing, Book Selling, or Course Rating (at least one)
    c. Assumptions
        i. There is at least one blacklisted term in the tested post.
    d. Test Data
        i. Input
            1. “Hello world! Fuck y0u”
        ii. Output
            1. A message that says “You are not allowed to create a post containing the following word(s) or phrase(s): ‘Fuck y0u’. Please update your post accordingly”
    e. Test Execution Steps
        i. The user types their post
        ii. The user tries to submit their post
        iii. If the user was warned about their post’s content, the user closes the message. If the user was not warned, the test ends (failure)
        iv. The user removes all blacklisted terms from their post
        v. The user tries to submit their post
        vi. If there were no blacklisted terms in their post, the submission is accepted
    f. Success Cases

    i. When the user tries to submit their post, they are shown a warning that tells them to remove blacklisted terms. They are unable to submit their post until they do so.

g. Failure Cases
    i. The blacklisted term is not detected.
    ii. The user is not warned.
    iii. The user is not told which term in their post is not allowed.
    iv. The user is able to create their post regardless of whether or not they were warned.

Recipe Sharing Test Cases

1. Upload a Recipe
    h. Requirements Tested
        i. RE01
    i. Dependencies
        i. Student must be logged in
    j. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    k. Test Data
        i. Input
            1. Name: Pasta
            2. Category: Dinner
            3. Caloric value: 100
            4. Overall price: $10
            5. Date posted: 2/6/2022
            6. Meals it makes (for one): 3
            7. Step-by-step instructions: Boil water. Add pasta to water. Cook for the amount of time as said on the box. Drain pasta. Use sauce of your choosing. Eat.
        ii. Output
            1. True - post made
    l. Test Execution Steps
        i. User is on home page
        ii. User selects Recipe Sharing
        iii. User selects add new post
        iv. User inputs above data for said category
        v. User selects post (ADD)
    m. Success Cases
        i. User's post is successfully uploaded
    n. Failure Cases
        i. User exceeds more than 5000 characters in step by step instructions
        ii. Users post in not uploaded successfully
2. Search a Recipe
    a. Requirements Tested
        i. RE02, RE05
    b. Dependencies

i. Student must be logged in
ii. A post of a recipe must already be made

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Test 1
        1. Input
            a. Search Name: Pasta
        2. Output
            a. True - posts are filtered
    ii. Test 2
        1. Input
            a. Search Overall Price: $10
        2. Output
            a. True - posts are filtered
    iii. Test 3
        1. Input
            a. Search Caloric Value: 100
        2. Output
            a. True - posts are filtered
    iv. Test 4
        1. Input
            a. Search Category: Dinner
        2. Output
            a. True - posts are filtered
e. Test Execution Steps
    i. User is on home page
    ii. User selects Recipe Sharing
    iii. User selects search bar
    iv. User inputs above test data to filter results
    v. User selects enter
    vi. User repeats until all tests are completed
f. Success Cases
    i. User's results are successfully filtered
    ii. By default users results are ranking in order of caloric value
g. Failure Cases

        i. User's results are not successfully filtered

3. Filter by Daily Budget
    a. Requirements Tested
        i. RE03, RE06
    b. Dependencies
        i. Student must be logged in
        ii. Post must be made
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Daily Budget: $15
            2. Meals per day: 1
        ii. Output
            1. True - posts filtered
    e. Test Execution Steps
        i. User is on home page
        ii. User selects Recipe Sharing
        iii. User selects filter
        iv. User inputs above data for budget and meals per day
        v. User selects enter
    f. Success Cases
        i. User's post are successfully filtered
    g. Failure Cases
        i. User's post are not successfully filtered and can

Book Selling Test Cases

1. Making a Posting
    a. Requirements Tested
        i. BKS05/BKS01
    b. Dependencies
        i. Student must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Book Title: Artificial Intelligence A Modern Approach
            2. Edition: 4th
            3. ISBN-10: 0134610997
            4. Condition: Used, Like New
        ii. Output
            1. User confirmed to have taken class previously? Yes.
            2. Book Title: Artificial Intelligence A Modern Approach
            3. Edition: 4th
            4. ISBN-10: 0134610997
            5. Condition: Used, Like New
    e. Test Execution Steps
        i. User navigates to book selling page.
        ii. User Clicks on “Post book to sell”.
        iii. User fills out criteria for the book.
        iv. User clicks “post”.
    f. Success Cases
        i. The book is successfully added to the selling list and is viewable.
    g. Failure Cases
        i. The book is successfully added to the selling list and is not viewable.
        ii. The book is unsuccessfully added to the selling list.
        iii. The book is posted with the incorrect information.
2. Updating listing
    a. Requirements Tested
        i. BKS02
    b. Dependencies
        i. Student must be logged in

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. Book Title: Artificial Intelligence A Modern Approach
        2. ISBN-10: 0134610997
        3. Price sold: $80
        4. Sold to: Brad
    ii. Output
        1. Post updated to “Sold”
e. Test Execution Steps
    i. User navigates to the book selling page.
    ii. User Clicks on their own post.
    iii. User fills out criteria for the sale.
    iv. User clicks “update listing”.
    v. Seller clicks “confirm”
f. Success Cases
    i. Post is now updated with new information given by the user and redirected to their updated listing.
g. Failure Cases
    i. Post is not updated.
    ii. User is redirected to a separate unrelated page.

3. Searching for books
    i. Requirements Tested
        1. BKS03
    ii. Dependencies
        1. Student must be logged in
    iii. Assumptions
        1. User is connected to internet
        2. User does not lose connection to internet
    iv. Test Data
        1. Input
            a. Book Title: Artificial Intelligence A Modern Approach
        2. Output
            a. Found x Results for “Book Title: Artificial Intelligence A Modern Approach”
    v. Test Execution Steps

1. User navigates to the book selling page.
2. User enters the book title or ISBN-10 number of the book
3. User clicks on a book from the search

vi. Success Cases

1. Users are able to correctly view all available listings for their book.

vii. Failure Cases

1. User is redirected to a page that isn't their search results
2. User is unable to filter search results
3. User is not notified that their are no results for their search

Event Planning Test Cases

1. Post an Event
    a. Requirements Tested
        i. EP0, EP06, EP07
    b. Dependencies
        i. Student must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Name of event: Summer Bash
            2. Time of event: 7pm
            3. Date of event: 5/12/2022
            4. Location of event: The beach
            5. Optional description of event: Schools out time to party
        ii. Output
            1. True - post successfully made
    e. Test Execution Steps
        i. User is on home page
        ii. User selects Event Planning
        iii. User selects add new post
        iv. User inputs above data for said category
        v. User selects post
    f. Success Cases
        i. User's post is successfully made within 5 seconds
    g. Failure Cases
        i. User's post is not successfully made
        ii. Users description exceeds 1000 characters
2. View an Event
    a. Requirements Tested
        i. EP01, EP05
    b. Dependencies
        i. Student must be logged in
        ii. Event must be made
    c. Assumptions
        i. User is connected to internet

ii. User does not lose connection to internet

d. Test Data
    i. Input
        1. Name of event: Summer Bash
    ii. Output
        1. Time of event: 7pm
        2. Date of event: 5/12/2022
        3. Location of event: The beach
        4. Optional description of event: Schools out time to party
e. Test Execution Steps
    i. User is on home page
    ii. User selects Event Planning
    iii. User selects an event
f. Success Cases
    i. User can view an event and see all the information
    ii. Events are sorted by closest to farthest distance by default
g. Failure Cases
    i. User cannot view an event
    ii. Information is not showing up or is incorrect

3. Update an Event
    a. Requirements Tested
        i. EP02, EP06, EP07
    b. Dependencies
        i. Student must be logged in
        ii. User must have made a post for an event
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Name of event: Summer Bash
            2. Update Date: 5/20/2022
            3. Update Time: 6 pm
        ii. Output
            1. True - post successfully updated
    e. Test Execution Steps
        i. User is on home page

ii. User selects Event Planning
iii. User selects their post
iv. User selects update post
v. User updates post with above test criteria
vi. User selects post

f. Success Cases
    i. User's post is successfully updated
    ii. Users who added it to their calendar will be notified of the change
g. Failure Cases
    i. User's update is not successfully made within 5 seconds
    ii. Users post still contains incorrect information

4. On Campus Events Prioritized
    a. Requirements Tested
        i. EP03
    b. Dependencies
        i. Student must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Test 1
            1. Input
                a. Name of event: Summer Bash
                b. Time of event: 7pm
                c. Date of event: 5/12/2022
                d. Location of event: The beach
                e. Optional description of event: Schools out time to party
            2. Output
                a. True - post successfully made
        ii. Test 2
            1. Input
                a. Name of event: Study Party
                b. Time of event: 7pm
                c. Date of event: 4/12/2022
                d. Location of event: CSULB Library
                e. Optional description of event: Let's have a study party and not actually study

2. Output
    a. True - post successfully made

e. Test Execution Steps
    i. User is on home page
    ii. User selects Event Planning
    iii. User selects add new post
    iv. User inputs above data for said category
    v. User selects post
    vi. User repeats steps for both tests
f. Success Cases
    i. User's posts are successfully updated
    ii. On campus event is prioritized over the off campus event
g. Failure Cases
    i. User's update is not successfully made within 5 seconds
    ii. On campus event is not prioritized over the off campus event

5. Add event to calendar
    a. Requirements Tested
        i. EP04
    b. Dependencies
        i. Student must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Name of event: Summer Bash
        ii. Output
            1. True - event added to calendar
    e. Test Execution Steps
        i. User is on home page
        ii. User selects Event Planning
        iii. User selects a post
        iv. User selects add to calendar
    f. Success Cases
        i. Event is successfully added to users calendar
    g. Failure Cases
        i. Event is not added to users calendar

Registration Test Cases

1. Register An Account - Success
    a. Requirements Tested
        i. RE01, RE02, RE03, RE04, RE05, RE06, RE07, RE09, RE10, RE11
    a. Dependencies
        i. The user must not be logged in
    b. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    c. Test Data
        i. Input
            1. Username - abrio
            2. Pass phrase - hello world
            3. School - CSULB
            4. Email - audrey.brio@student.csulb.edu
        ii. Output
            1. True - user registered
    d. Test Execution Steps
        i. User goes to http://student-multi-tool.com
        ii. User selects create an account
        iii. User enters above information
        iv. User selects enter
    e. Success Cases
        i. User registers with a valid email and passphrase.
        ii. System message displays “email confirmation pending”
        iii. User receives confirmation email within 15 seconds of message displayed
        iv. User complete email confirmation within 24 hours
        v. Message displayed, “Account created successfully
        vi. Login information is saved to the datastore
    f. Failure Cases
        i. User does not get confirmation email
        ii. User does not complete email confirmation within 24 hours
        iii. System messages are not displayed
        iv. System goes offline
2. Register An Account -Fail - Invalid passphrase

a. Requirements Tested
    ii. RE01, RE02, RE04, RE07, RE09, RE10, RE11
b. Dependencies
    i. The user must not be logged in
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Test 1
        1. Input
            a. Username - abrio
            b. Pass phrase - hello
            c. School - CSULB
            d. Email - audrey.brio@student.csulb.edu
        2. Output
            a. False - user not registered
    ii. Test 2
        1. Input
            a. Username - abrio
            b. Pass phrase - pa$$
            c. School - CSULB
            d. Email - audrey.brio@student.csulb.edu
        2. Output
            a. False - user not registered
e. Test Execution Steps
    i. User goes to http://student-multi-tool.com
    ii. User selects create an account
    iii. User enters above information from test 1
    iv. User selects enter
    v. User retries using information from test 2
    vi. User selects enter
f. Success Cases
    i. User is not registered
    ii. User does not get email confirmation
    iii. Message displayed saying “Email or pass phrase error”
g. Failure Cases
    i. User received confirmation email

ii. User is registered
iii. System goes offline

3. Register An Account -Fail - Invalid Username
    a. Requirements Tested
        iii. RE01, RE02, RE05, RE07, RE09, RE10, RE11
    b. Dependencies
        i. The user must not be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Test 1
            1. Input
                a. Username - ab
                b. Pass phrase - hello world
                c. School - CSULB
                d. Email - audrey.brio@student.csulb.edu
            2. Output
                a. False - user not registered
        ii. Test 2
            1. Input
                a. Username - ABRIO
                b. Pass phrase - hello world
                c. School - CSULB
                d. Email - audrey.brio@student.csulb.edu
            2. Output
                a. False - user not registered
    e. Test Execution Steps
        i. User goes to http://student-multi-tool.com
        ii. User selects create an account
        iii. User enters above information from test 1
        iv. User selects enter
        v. User retries using information from test 2
        vi. User selects enter
    f. Success Cases
        i. User is not registered
        ii. User does not get email confirmation

iii. Message displayed saying “username error”

g. Failure Cases
   i. User received confirmation email
   ii. User is registered
   iii. System goes offline

4. Register An Account -Fail - Invalid Email
   a. Requirements Tested
      iv. RE01, RE02, RE07, RE09, RE10, RE11
   b. Dependencies
      i. The user must not be logged in
   c. Assumptions
      i. User is connected to internet
      ii. User does not lose connection to internet
   d. Test Data
      i. Input
         1. Username - abrio
         2. Pass phrase - hello world
         3. School - CSULB
         4. Email - abrio@student.csun.edu
      ii. Output
         1. False - user not registered
   e. Test Execution Steps
      i. User goes to http://student-multi-tool.com
      ii. User selects create an account
      iii. User enters above information
      iv. User selects enter
   f. Success Cases
      i. User is not registered
      ii. User does not get email confirmation
      iii. Message displayed saying “Email or pass phrase error”
   g. Failure Cases
      i. User received confirmation email
      ii. User is registered
      iii. System goes offline
5. Register An Account -Fail - User Already Exists
   a. Requirements Tested
      v. RE01, RE03, RE07, RE09, RE10, RE11

b. Dependencies
    i. The user must not be logged in
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. Username - mkriesel
        2. Pass phrase - hello world
        3. School - CSULB
        4. Email - michael.kriesel@student.csulb.edu
    ii. Output
        1. False - user not registered
e. Test Execution Steps
    i. User goes to http://student-multi-tool.com
    ii. User selects create an account
    iii. User enters above information
    iv. User selects enter
f. Success Cases
    i. User is not registered
    ii. User does not get email confirmation
    iii. Message displayed saying “Error User Already Exisits”
g. Failure Cases
    i. User received confirmation email
    ii. User is registered
    iii. System goes offline

6. Register An Account -Fail - Create Admin Account
    a. Requirements Tested
        vi. RE01, RE07, RE08, RE09, RE10, RE11
    b. Dependencies
        i. The user must not be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Username - abrio

2. Pass phrase - hello world
3. School - CSULB
4. Email - abrio@student.csulb.edu
5. Role - admin

ii. Output

1. False - user not registered

e. Test Execution Steps
   i. User goes to http://student-multi-tool.com
   ii. User selects create an account
   iii. User enters above information
   iv. User selects enter
f. Success Cases
   i. User is not registered
   ii. User does not get email confirmation
   iii. Message displayed saying “Error cannot create admins”
g. Failure Cases
   i. User received confirmation email
   ii. User is registered
   iii. System goes offline

Login Test Cases

1. Login User - Success
    a. Requirements Tested
        i. LI01, L102, LI07, LI08
    b. Dependencies
        i. User must have a registered account
        ii. The user must not be logged in already
        iii. User must be on login view
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Student email: audrey.brio@student.csulb.edu
            2. Pass code - hello world
            3. Username - abrio
            4. One time password: 123456
        ii. Output
            1. True - user has successfully logged into their account
    e. Test Execution Steps
        i. Navigate to site website https;//studentmulti-tool.com
        ii. Enter student email audrey.brio@student.csulb.edu and pass code hello world for current test
        iii. Enter username abrio and one time password (that would have been sent to the email) 123456
        iv. Select login
    f. Success Cases
        i. User is authenticate and taken to the student multi tool's home page
    g. Failure Cases
        i. User is logged in, but not taken to the student multi-tool's home page
        ii. User is logged in, but no automation navigation takes place
2. Login User - Incorrect Email or Pass code
    a. Requirements Tested
        i. LI01, LI07, LI08
    b. Dependencies
        i. User must have a registered account
        ii. The user must not be logged in already

iii. User must be on login view

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Test 1
        1. Input
            a. Student email: audrey.brio@student.csulb.edu
            b. Pass code - no
        2. Output
            a. False - pass code incorrect
    ii. Test 2
        1. Input
            a. Student email: abrio@student.csulb.edu
            b. Pass code - hello world
        2. Output
            a. False - email incorrect
e. Test Execution Steps
    i. Navigate to site website https;//studentmulti-tool.com
    ii. Enter student email audrey.brio@student.csulb.edu and pass code no
    iii. Select enter
    iv. Try again using student email abrio@student.csulb.edu and pass code hello world
    v. Select enter
f. Success Cases
    i. User is not logged in
    ii. A system message displays “Invalid pass code or email provided. Retry again or contact system admin”.
g. Failure Cases
    i. The user is taken to enter username and password

3. Login User - Incorrect username or one time password
    a. Requirements Tested
        i. LI01, L102, LI07, LI08
    b. Dependencies
        i. User must have a registered account
        ii. The user must not be logged in already
        iii. User must be on login view

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Test 1:
        1. Input
            a. Student email: michael.kriesel@student.csulb.edu
            b. Pass code - super man
            c. Username - mk
            d. One time password: Password!
        2. Output
            a. False - the username is incorrect
    ii. Test 2:
        1. Student email: michael.kriesel@student.csulb.edu
        2. Pass code - super man
        3. Username - mkiesel
        4. One time password: Password1
        5. Output
            a. False - The password is incorrect
e. Test Execution Steps
    i. Navigate to site website https;//studentmulti-tool.com
    ii. Enter student email michael.kriesel@student.csulb.edu and pass code super man
    iii. Enter username mk and one time password (that would have been sent to the email) Password!
    iv. Select login
    v. Retry with entering same email and passcode and mkriesel for username and Password1 for password
    vi. Select login
f. Success Cases
    i. User is not logged in
    ii. A system message displays “Invalid username or password provided. Retry again or contact system administrator”.
g. Failure Cases
    i. The username is incorrect (does not exist in the database)
    ii. The one time password is incorrect
    iii. The users email/pass code is incorrect

iv. User is logged in, but not taken to the student multi-tool's home page
v. User is logged in, but no automation navigation takes place

4. Login User - Trying to log into a Disabled Account
    a. Requirements Tested
        i. LI01, L102, LI07, LI08
    b. Dependencies
        i. User must have a registered account
        ii. The user must not be logged in already
        iii. User must be on login view
        iv. User must have a disabled account
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Student email: bradley.nickle@student.csulb.edu
            2. Pass code - marvel fan
            3. Username - bnickle
            4. One time password: 987654
        ii. Output
            1. False - User account is disabled
    e. Test Execution Steps
        i. Navigate to site website https;//studentmulti-tool.com
        ii. Enter student email bradley.nickle@student.csulb.edu and pass code marvel fan for current test
        iii. Enter username bnickle and one time password (that would have been sent to the email) 987654
        iv. Select login
    f. Success Cases
        i. User is not logged in and the system displays the message "Account disabled. Perform account recovery or contact system admin".
        ii. The failure attempt is recorded accurately.
    g. Failure Cases
        i. User is logged in
        ii. The failure attempt is not recorded accurately and the system attempts to log that the failure attempt did not complete successfully
5. Login User - More than 5 incorrect attempts

a. Requirements Tested
    i. LI04, LI06, LI07, LI08
b. Dependencies
    i. User must have a registered account
    ii. The user must not be logged in already
    iii. User must be on login view
    iv. All attempts are within a 24 hour period
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Test 1:
        1. Input
            a. Student email: audrey.brio@student.csulb.edu
            b. Pass code - hello world
            c. Username - audreybrio
            d. One time password: 12345634
        2. Output
            a. False - the username is incorrect
    ii. Test 2:
        1. Student email: audrey.brio@student.csulb.edu
        2. Pass code - hello world
        3. Username - abrio
        4. One time password: 222222
        5. Output
            a. False - The password is incorrect
    iii. Test 3:
        1. Input
            a. Student email: audrey.brio@student.csulb.edu
            b. Pass code - hello world
            c. Username - audreybrio
            d. One time password: 333333
        2. Output
            a. False - the username is incorrect
    iv. Test 4:
        1. Student email: audrey.brio@student.csulb.edu
        2. Pass code - hello world

3. Username - abrio
4. One time password: 555555
5. Output
    a. False - The password is incorrect

v. Test 5:
1. Input
    a. Student email: audrey.brio@student.csulb.edu
    b. Pass code - hello world
    c. Username - audreyb
    d. One time password: 999999
2. Output
    a. False - the username and password is incorrect

e. Test Execution Steps
i. Navigate to site website https;//studentmulti-tool.com
ii. Enter student email audrey.brio@student.csulb.edu and pass code hello world
iii. Enter username audreybrio and one time password (that would have been sent to the email) 12345634
iv. Select login
v. Retry with entering same email and passcode and abrio for username and 222222 for password
vi. Select login
vii. Retry with entering same email and passcode and audreybrio for username and 333333 for password
viii. Select login
ix. Retry with entering same email and passcode and abrio for username and 555555 for password
x. Select login
xi. Retry with entering same email and passcode and audrey for username and 999999 for password
xii. Select login

f. Success Cases
i. User is not logged in
ii. User account is disabled
iii. IP address of each attempt is recorded

g. Failure Cases
i. User is logged in

ii. User account is not disabled

6. Login User - 2 incorrect attempts, then correct attempt
    a. Requirements Tested
        i. LI04, LI05, LI06, LI07, LI08
    b. Dependencies
        i. User must have a registered account
        ii. The user must not be logged in already
        iii. User must be on login view
        iv. All attempts are within a 24 hour period
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Test 1:
            1. Input
                a. Student email: audrey.brio@student.csulb.edu
                b. Pass code - hello world
                c. Username - audreybrio
                d. One time password: 123456
            2. Output
                a. False - the username is incorrect
        ii. Test 2:
            1. Student email: audrey.brio@student.csulb.edu
            2. Pass code - hello world
            3. Username - abrio
            4. One time password: 222222
            5. Output
                a. False - The password is incorrect
        iii. Test 3:
            1. Input
                a. Student email: audrey.brio@student.csulb.edu
                b. Pass code - hello world
                c. Username - abrio
                d. One time password: 123456
            2. Output
                a. True - user is logged into system
    e. Test Execution Steps

i. Navigate to site website https;//studentmulti-tool.com
ii. Enter student email audrey.brio@student.csulb.edu and pass code hello world
iii. Enter username audreybrio and one time password (that would have been sent to the email) 123456
iv. Select login
v. Retry with entering same email and passcode and abrio for username and 222222 for password
vi. Select login
vii. Retry with entering same email and passcode and abrio for username and 123456 for password
viii. Select login

f. Success Cases
   i. User is logged in after correct attempt (test 3)
   ii. User account is not disabled
   iii. IP address of failed each attempt is recorded
   iv. Fail count reset to 0
g. Failure Cases
   i. User is not logged in after correct attempt
   ii. User account is disabled
   iii. Fail count is not reset to

Logout Test Cases

1. Logout User
    a. Requirements Tested
        i. LO01, LO02, LO03
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Any valid user account that is logged in
    e. Test Execution Steps
        i. User selects logout button
        ii. User selects yes
    f. Success Cases
        i. User is successfully logged out and returned to the login view within 5 seconds of logging out. The active session ends, and the user is unauthenticated.
    g. Failure Case
        i. User is still logged in
        ii. Logging out takes longer than 5 seconds to complete
2. User Closes the Browser
    a. Requirements Tested
        i. LO01
    b. Dependencies
        i. User must be logged in
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Valid user account that is logged in
    e. Test Execution Steps
        i. User closes the browser
        ii. User reopens the browser and visits the home view
    f. Success Cases
        i. User is successfully logged out. The active session ends, and the user is unauthenticated.

g. Failure Case
    i. User is still logged in
    ii. Logging out takes longer than 5 seconds to complete

User Management Test Cases

1. Creating a User
    a. Requirements Tested
        i. UM01, UM03, UM05, UM07, UM08, UM10
    b. Dependencies
        i. Triggered by a user registering an account
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. User creating a new user is an admin
    d. Test Data
        i. Input
            1. Student email - michael.kriesel@student.csulb.edu
            2. Username - mkriesel
            3. Password - Pas$word1
            4. University - CSULB
        ii. Output
            1. True - information is in database
    e. Test Execution Steps
        i. Admin user is on homepage
        ii. Admin user selects settings
        iii. Admin user selects admin - user management
        iv. Admin user selects create user(s)
        v. Admin user creates an account for mkriesel
        vi. mkriesel can login
    f. Success Cases
        i. Mkriesel is now in the database with the same information entered at registration (information above)
        ii. System performs operation within 5 seconds
        iii. System display message, “UM operation was successful”
    g. Failure Cases
        i. Mkriesel is not in the database
        ii. System takes more than 5 seconds to perform the operation
        iii. System does not display message, “UM operation was successful”
        iv. System goes offline
2. Deleting a User
    a. Requirements Tested

i. UM01, UM02, UM03, UM05, UM07, UM08, UM10

b. Dependencies
    i. User must have an account
c. Assumptions
    i. User is connected to internet
    ii. User is logged in
    iii. User does not lose connection to internet
    iv. User deleted another existing user is an admin
d. Test Data
    i. Input
        1. Username: mkriesel
        2. Password - Pas$word1
    ii. Output
        1. True - user is deleted from database
e. Test Execution Steps
    i. Admin user is on home page
    ii. Admin user selects settings
    iii. Admin user selects admin - user management
    iv. Amid user selects delete account(s)
    v. User selects mkrisel
    vi. User selects yes
f. Success Cases
    i. Mkriesel is successfully deleted from the database
    ii. Mkriesel can no longer use username and password to log in
    iii. System performs operation within 5 seconds
    iv. System display message, “UM operation was successful”
g. Failure Cases
    i. mkriesel is not deleted from the database
    ii. mkriesel can still log into the system
    iii. System takes more than 5 seconds to perform the operation
    iv. System does not display message, “UM operation was successful”

3. Updating a User - Student to Admin
    a. Requirements Tested
        i. UM01, UM03, UM05, UM07, UM08, UM10
    b. Dependencies
        i. User must be logged in
        ii. User updating status is an admin

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. Student User: mkriesel
        2. Admin User: abrio
    ii. Output
        1. True - status updated
e. Test Execution Steps
    i. abrio is on home page
    ii. abrio selects settings
    iii. abrio selects admin - user management
    iv. abrio selects add a user
    v. Abrio adds mkriesel as an admin
f. Success Cases
    i. mkriesel now has admin access
    ii. Database updates mkriesel status to an admin
    iii. System performs operation within 5 seconds
    iv. System display message, “UM operation was successful”
g. Failure Cases
    i. mkriesel does not have admin access
    ii. System takes more than 5 seconds to perform the operation
    iii. System does not display message, “UM operation was successful”

4. Updating a User - Admin to Student - Different users account
    a. Requirements Tested
        i. UM01, UM03, UM05, UM07, UM08, UM10
    b. Dependencies
        i. User must be logged in
        ii. User updating status is an admin
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Admin User: mkriesel
            2. Admin User: abrio

        ii. Output
            1. True - status updated
    e. Test Execution Steps
        i. abrio is on home page
        ii. abrio selects settings
        iii. abrio selects admin - user management
        iv. abrio selects update status
        v. Abrio removes mkriesel as an admin
    f. Success Cases
        i. mkriesel now has student access
        ii. Database updates mkriesel status to a student
        iii. System performs operation within 5 seconds
        iv. System display message, “UM operation was successful”
    g. Failure Cases
        i. mkriesel has admin access still
        ii. System takes more than 5 seconds to perform the operation
        iii. System does not display message, “UM operation was successful”
5. Updating a User - Admin to Student - Own Account
    a. Requirements Tested
        i. UM01, UM03, UM05, UM07, UM08, UM10
    b. Dependencies
        i. User must be logged in
        ii. User updating status is an admin
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Admin User: abrio
            2. Admin User: abrio
        ii. Output
            1. False - status not updated
    e. Test Execution Steps
        i. abrio is on home page
        ii. abrio selects settings
        iii. abrio selects admin - user management
        iv. abrio selects update status

v. Abrio removes abrio as an admin

f. Success Cases
   i. Abrio is still an admin
   ii. Database shows abrio has admin access level
   iii. System performs operation within 5 seconds
   iv. System display message, “UM operation was successful”

g. Failure Cases
   i. Abrio has student access
   ii. System takes more than 5 seconds to perform the operation
   iii. System does not display message, “UM operation was successful”

6. Enable a User
   a. Requirements Tested
      i. UM01, UM03, UM05, UM07, UM08, UM10
   b. Dependencies
      i. User must be logged in
      ii. User updating status is an admin
      iii. User must be updating user who is disabled
   c. Assumptions
      i. User is connected to internet
      ii. User does not lose connection to internet
   d. Test Data
      i. Input
         1. Student User: mkriesel
         2. Admin User: abrio
      ii. Output
         1. True - user enabled
   e. Test Execution Steps
      i. abrio is on home page
      ii. abrio selects settings
      iii. Abrio selects admin - user management
      iv. abrio selects users
      v. Abrio enables mkriesel
   f. Success Cases
      i. mkriesel now has access to system
      ii. Database updates mkriesel status to enabled
      iii. System performs operation within 5 seconds
      iv. System display message, “UM operation was successful”

g. Failure Cases
    i. mkriesel does not have access to system
    ii. System takes more than 5 seconds to perform the operation
    iii. System does not display message, “UM operation was successful”

7. Disable a User
    a. Requirements Tested
        i. UM01, UM03, UM05, UM07, UM08, UM10
    b. Dependencies
        i. User must be logged in
        ii. User updating status is an admin
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Student User: mkriesel
            2. Admin User: abrio
        ii. Output
            1. True - user disabled
    e. Test Execution Steps
        i. abrio is on home page
        ii. abrio selects settings
        iii. Abrio selects admin - user management
        iv. abrio selects users
        v. Abrio disables mkriesel
    f. Success Cases
        i. mkriesel now does not have access to system
        ii. Database updates mkriesel status to diasabled
        iii. System performs operation within 5 seconds
        iv. System display message, “UM operation was successful”
    g. Failure Cases
        i. mkriesel still has access to system
        ii. System takes more than 5 seconds to perform the operation
        iii. System does not display message, “UM operation was successful”
8. Creating a group of Users
    a. Requirements Tested
        i. UM01, UM04, UM05, UM06, UM07, UM10

b. Dependencies
    i. User is logged in
    ii. User creating new users is an admin
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. User 1
            a. Student email - michael.kriesel@student.csulb.edu
            b. Username - mkriesel
            c. Password - Pas$word1
            d. University - CSULB
        2. User 2
            a. Student email - audrey.brio@student.csulb.edu
            b. Username - abrio
            c. Password - Pas$word2
            d. University - CSULB
        3. User 3
            a. Student email - bradly.nickle@student.csulb.edu
            b. Username - bnickle
            c. Password - Pas$word3
            d. University - CSULB
        4. User 4
            a. Student email - joseph.cutri@student.csulb.edu
            b. Username - jcutri
            c. Password - Pas$word4
            d. University - CSULB
        5. User 5
            a. Student email - albert.toscano@student.csulb.edu
            b. Username - atoscano
            c. Password - Pas$word5
            d. University - CSULB
    ii. Output
        1. True - information is in database
e. Test Execution Steps
    i. Admin user is on home page

ii. Admin user clicks on settings
iii. Admin user selects admin - user management
iv. Admin user selects create new user(s)
v. Admin creates accounts for mkriesel, abrio, bnickle, jcutri, and atoscano

f. Success Cases
i. mkriesel, abrio, bnickle, jcutri, and atoscano are now in the database with the same information entered at registration (information above)
ii. mkriesel, abrio, bnickle, jcutri, and atoscano are processed one at time in the order they came in, one after the other
iii. System performs operation within 60 seconds
iv. System display message, “Bulk UM operation was successful”

g. Failure Cases
i. mkriesel, abrio, bnickle, jcutri, or atoscano are not in the database
ii. System takes more than 60 seconds to perform the operation
iii. System does not display message, “Bulk UM operation was successful”

9. Deleting a group of Users
a. Requirements Tested
i. UM01, UM02, UM04, UM05, UM06, UM07, UM10

b. Dependencies
i. User must have an account

c. Assumptions
i. User is connected to internet
ii. User is logged in
iii. User is an admin
iv. User does not lose connection to internet

d. Test Data
i. Input
1. User 1
a. Username - mkriesel
b. Password - Pas$word1
2. User 2
a. Username - abrio
b. Password - Pas$word2
3. User 3
a. Username - bnickle
b. Password - Pas$word3

4. User 4
    a. Username - jcutri
    b. Password - Pas$word4
5. User 5
    a. Username - atoscano
    b. Password - Pas$word5

ii. Output
1. True - user is deleted from database

e. Test Execution Steps
i. User is on home page
ii. User selects settings
iii. User selects admin - user management
iv. User selects delete account(s)
v. User selects User 1, User 2, User 3, User 4, User 5
vi. User selects yes

f. Success Cases
i. User successfully deleted User 1, User 2, User 3, User 4, User 5 from from the database
ii. User 1, User 2, User 3, User 4, and User 5 can no longer use username and password to log in
iii. System performs operation within 60 seconds
iv. System display message, “Bulk UM operation was successful”

g. Failure Cases
i. User 1, User 2, User 3, User 4, or User 5 is not deleted from the database
ii. User 1, User 2, User 3, User 4, or User 5 can still log into the system
iii. System takes more than 60 seconds to perform the operation
iv. System does not display message, “Bulk UM operation was successful”

10. Updating a Group of Users - Students to Admins
a. Requirements Tested
i. UM01, UM04, UM05, UM06, UM07, UM10

b. Dependencies
i. User must be logged in
ii. User updating status is an admin

c. Assumptions
i. User is connected to internet
ii. User does not lose connection to internet

d. Test Data
    i. Input
        1. Student User: mkriesel
        2. Student User: bnickle
        3. Student User: jcutri
        4. Student User: atoscano
        5. Admin User: abrio
    ii. Output
        1. True - status updated
e. Test Execution Steps
    i. abrio is on home page
    ii. abrio selects settings
    iii. abrio selects admin - user management
    iv. abrio selects add a user
    v. Abrio adds mkriesel as an admin
    vi. Abrio adds bnickle as an admin
    vii. Abrio adds jcutri as an admin
    viii. Abrio adds atoscano as an admin
f. Success Cases
    i. Mkriesel, bnickle, jcutri, and atoscano now has admin access
    ii. Database is updated with Mkriesel, bnickle, jcutri, and atoscano status as an admin
    iii. System processes request one after the other
    iv. System performs operation within 60 seconds
    v. System display message, “Bulk UM operation was successful”
g. Failure Cases
    i. Mkriesel, bnickle, jcutri, and atoscano do not have admin access
    ii. System takes more than 60 seconds to perform the operation
    iii. System does not display message, “Bulk UM operation was successful”

11. Updating a Group of Users - Admin to Student
    a. Requirements Tested
        i. UM01, UM04, UM05, UM06, UM07, UM10
    b. Dependencies
        i. User must be logged in
        ii. Both users are admins
    c. Assumptions
        i. User is connected to internet

ii. User does not lose connection to internet

d. Test Data
    i. Input
        1. Admin User: mkriesel
        2. Admin User: bnickle
        3. Admin User: jcutri
        4. Admin User: atoscano
        5. Admin User: abrio
    ii. Output
        1. True - status updated
e. Test Execution Steps
    i. abrio is on home page
    ii. abrio selects settings
    iii. abrio selects admin - user management
    iv. abrio selects update status
    v. Abrio removes mkriesel as an admin
    vi. Abrio removes bnickle as an admin
    vii. Abrio removes jcutri as an admin
    viii. Abrio removes atoscano as an admin
f. Success Cases
    i. Mkriesel, bnickle, jcutri, and atoscano now have student access
    ii. Database is updated with Mkriesel, bnickle, jcutri, and atoscano status as a student
    iii. System processes request one after the other
    iv. System performs operation within 60 seconds
    v. System display message, “Bulk UM operation was successful”
g. Failure Cases
    i. mkriesel, bnickle, jcutri, and atoscano have admin access still
    ii. System takes more than 60 seconds to perform the operation
    iii. System does not display message, “Bulk UM operation was successful”

12. Enabling a Group of Users
    a. Requirements Tested
        i. UM01, UM04, UM05, UM06, UM07, UM10
    b. Dependencies
        i. User must be logged in
        ii. User must be updating user who is disabled
        iii. User updating status is an admin

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. Student User: mkriesel
        2. Student User: bnickle
        3. Student User: jcutri
        4. Student User: atoscano
        5. Admin User: abrio
    ii. Output
        1. True - user enabled
e. Test Execution Steps
    i. abrio is on home page
    ii. abrio selects settings
    iii. abrio selects admin - user management
    iv. abrio selects users
    v. Abrio enables mkriesel
    vi. Abrio enables bnickle
    vii. Abrio enables jcutri
    viii. Abrio enables atoscano
f. Success Cases
    i. Mkriesel, bnickle, jcutri, and atoscano now have access to system
    ii. Database is updated with Mkriesel, bnickle, jcutri, and atoscano status enabled
    iii. System performs operation within 60 seconds
    iv. System display message, “Bulk UM operation was successful”
g. Failure Cases
    i. Mkriesel, bnickle, jcutri, and atoscano does not have access to system
    ii. System takes more than 60 seconds to perform the operation

13. Disabling a Group of Users
    a. Requirements Tested
        i. UM01, UM04, UM05, UM06, UM07, UM10
    b. Dependencies
        i. User must be logged in
        ii. User updating status is an admin
    c. Assumptions

i. User is connected to internet
ii. User does not lose connection to internet

d. Test Data
    i. Input
        1. Student User: mkriesel
        2. Student User: bnickle
        3. Student User: jcutri
        4. Student User: atoscano
        5. Admin User: abrio
    ii. Output
        1. True - user disabled

e. Test Execution Steps
    i. abrio is on home page
    ii. abrio selects settings
    iii. abrio selects admin - user management
    iv. abrio selects users
    v. Abrio disables mkriesel
    vi. Abrio disables bnickle
    vii. Abrio disables jcutri
    viii. Abrio disabled atoscano

f. Success Cases
    i. Mkriesel, bnickle, jcutri, and atoscano now does not have access to system
    ii. Database is updated with Mkriesel, bnickle, jcutri, and atoscano status disabled
    iii. System performs operation within 60 seconds
    iv. System display message, “Bulk UM operation was successful”

g. Failure Cases
    i. Mkriesel, bnickle, jcutri, and atoscano still have access to system
    ii. System takes more than 60 seconds to perform the operation
    iii. System does not display message, “Bulk UM operation was successful”

14. Bulk Operations - Upload File Extract
    a. Requirements Tested
        i. UM01, UM04, UM05, UM06, UM07, UM10
    b. Dependencies
        i. User must be logged in
        ii. User status is an admin

c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. File - bulkops
    ii. Output
        1. True - all operations completed
e. Test Execution Steps
    i. user is on home page
    ii. user selects settings
    iii. user selects admin - user management
    iv. user selects file
    v. User uploads bulkops
f. Success Cases
    i. System successfully performs all operations listed in file within 60 seconds
    ii. System display message, “Bulk UM operation was successful”
g. Failure Cases
    i. System takes more than 60 seconds to perform the operation
    ii. File size is larger than 2 GB
    iii. System does not display message, “Bulk UM operation was successful”

Logging Test Cases

1. Log entries are saved to a persistent data store
    a. Requirements Tested
        i. LG02, LG03, LG05
    b. Dependencies
        i. The system logs all system events
        ii. Persistent data store is active and accessible by the system
        iii. Persistent data store has storage capacity for log entry
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. User sign-up to the website
                a. Student email - michael.kriesel@student.csulb.edu
                b. Username - mkriesel
                c. Password - Pas$word1
                d. University - CSULB
        ii. Output
            1. True - information is stored in database
    e. Test Execution Steps
        i. User registers an account successfully with above information
        ii. User can access the home page
    f. Success Cases
        i. User creates an account and the system logs the event to a persistent data store
            1. Logging entry: Info - Sat, 10 Nov 2021, 04:27:25 GMT - mkriesel - Data Store - New user account
        ii. User attempts to create an account, but account was not created due to the email address belonging to an existing account, and the system logs the event to a persistent data store
            1. Logging entry: Info - Sat, 10 Nov 2021, 04:27:25 GMT - Null - Data Store - New user account unsuccessfully
    g. Failure Cases
        i. New user account was created, but the system did not log the event to a persistent data store
        ii. User attempts to create an account, but account was not created and

the system did not logs the event to a persistent data store

2. All log entries are immutable
    a. Requirements Tested
        i. LG01, LG02, LG03, LG05
    b. Dependencies
        i. The system logs all system events
        ii. Persistent data store is active and accessible by the system
        iii. Persistent data store has storage capacity for log entry
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. mkriesel is an Admin and sign-in
        ii. Input
            1. Student email - michael.kriesel@student.csulb.edu
            2. Username - mkriesel
            3. Password - Pas$word1
        iii. Output
            1. True - information is stored in database
    e. Test Execution Steps
        i. User registers an account successfully with above information
        ii. User can access the home page
    f. Success Cases
        i. mkriesel creates an account and the system logs the event
            1. Logging entry: Info - Sat, 10 Nov 2021, 04:27:25 GMT - mkriesel - Data Store - Admin sign-in
            2. Mkriesel attempts to modify the logging timestamp, but he is unable to access the logging data to modify it
    g. Failure Cases
        i. mkriesel creates a setter(e.g. logging.setTimeStamp(new_date, new_time)) to modify logging entries' data
            1. mkriesel calls the setter: logging.setTimeStamp("07/30/2021", " 8:30:02")
            2. Modified logging entry: Info - Fri, 30 Jul 2021, 08:30:02 GMT - mkriesel - Data Store - Admin sign-in

3. The logging process does not block any user from performing any interaction with the system
    a. Requirements Tested
        i. LG02, LG03, LG04, LG05
    b. Dependencies
        i. User must be logged in
        ii. The schedule of classes for the school is up to date
        iii. The system logs all system events
        iv. Persistent data store is active and accessible by the system
        v. Persistent data store has storage capacity for log entry
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
        iii. The school is the one the user listed in their account
        iv. The classes and professors being entered are accurate
    d. Test Data
        i. User selects to add a new course review
        ii. Input
            1. Major: Computer Science;
            2. Course: CECS228;
            3. Term: Fall 2021,
            4. Professor: Mehrnia,
            5. Comments: “This was a fun class”;
            6. Difficulty: 3/5
        iii. Output
            1. Review published successfully. Overall difficulty = 3/5
    e. Test Execution Steps
        i. The user starts on the page to add a new review
        ii. The user selects the following information that comes from the corresponding test data above:
            1. User selects a major
            2. The user searches for and selects the class
            3. The user selects the professor
            4. The user adds comments about the class & professor
            5. The user inputs the desired difficulty rating
            6. The user publishes the review
    f. Success Cases

i. User adds a new class review and the system logs the event
    1. The system keeps running normally without any blocked process
    2. Logging entry: Info - 11/10/2021, 4:27:25 - mkriesel - Data Store - User added new review
ii. User cannot add a new class review, but the system logs the event
    1. The system keeps running normally without any blocked process
    2. Logging entry: Error - 11/10/2021, 4:27:25 - mkriesel - Data Store - User added new review unsuccessfully

g. Failure Cases
    i. User adds a new class review, the system logs the event, but the logging process blocks the other system processes
    ii. User cannot add a new class review, the system logs the event, but the logging process blocks the other system processes

4. The logging process is complete within 5 seconds upon invocation
    a. Requirements Tested
        i. LG02, LG03, LG05, LG07
    b. Dependencies
        i. User must be logged in
        ii. User needs access to at least six schedules to fully perform this test
        iii. The system logs all system events
        iv. Persistent data store is active and accessible by the system
        v. Persistent data store has storage capacity for log entry
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. The user selects two to five schedules to be compared
        ii. Input
            1. Schedules A, B, C, D, and E
            2. A duplicate of schedule A
        iii. Output
            1. True or false, depending on whether or not each comparison is executed
    e. Test Execution Steps
        i. mkriesel is an student and sign-in

ii. mkriesel is on home page
iii. User selects compare schedules
iv. For each integer n between 0 and 6(inclusive):
1. mkriesel selects n schedule(s)
2. mkriesel clicks the compare button
3. The comparison is either executed or not

f. Success Cases
i. mkriesel selects 2, 3, 4, or 5 schedules, the comparison is executed, and the system logs the event within 5 seconds
1. Logging entry: Info - 11/10/2021, 4:27:25 - mkriesel - View - mkriesel compared schedules
ii. mkriesel selects 2, 3, 4, or 5 schedules, the comparison is not executed, but the system logs the event within 5 seconds
1. Logging entry: Error - 11/10/2021, 4:27:25 - mkriesel - View - mkriesel compared schedules unsuccessfully

g. Failure Cases
i. mkriesel selects 2, 3, 4, or 5 schedules, the comparison is executed, but the system takes longer than 5 seconds to log the event
ii. mkriesel selects 2, 3, 4, or 5 schedules, the comparison is not executed, but the system takes longer than 5 seconds to log the event

5. System failures from this feature does not result in the system going offline
a. Requirements Tested
i. LG02, LG03, LG05, LG06
b. Dependencies
i. User must be logged in
ii. The system logs all system events
iii. Persistent data store is active and accessible by the system
iv. Persistent data store has storage capacity for log entry
c. Assumptions
i. User is connected to internet
ii. User does not lose connection to internet
iii. School is same as listed in user settings
iv. The test data from Course Reviews test 1 (add a new review) is already loaded into the database
d. Test Data
i. The user selects course review
ii. Test 1:

1. Input: CECS 228
2. Output:
    a. Overall difficulty - 4/5
    b. Comments: “This class was hard”, “This was a fun class”

iii. Test 2:
1. Input: Mehrnia
2. Output:
    a. Overall difficulty - 4/5
    b. Comments: “This class was hard”, “This was a fun class”

e. Test Execution Steps
    i. mkriesel is an student and sign-in
    ii. mkriesel is on home page
    iii. mkriesel selects Course Review
    iv. mkriesel navigates to the ‘find a review” page
    v. mkriesel enters their major
    vi. mkriesel searches for the class
    vii. mkriesel selects the class
f. Success Cases
    i. Class CECS 228 could not be found, but the system remains online
        1. Logging entry: Info - 11/10/2021, 4:27:25 - mkriesel - Data store - The system could not found the class CECS 228
    ii. The professor could not be found, but the system remains online
        1. Logging entry: Info - 11/10/2021, 4:27:25 - mkriesel - Data store - The system could not found the professor
    iii. The information in the review is inaccurate or not up to date, but the system remains online
        1. Logging entry: Info - 11/10/2021, 4:27:25 - mkriesel - Data store - The review information is not accurate
g. Failure Cases
    i. Class CECS 228 could not be found and the system goes offline
    ii. The professor could not be found and the system goes offline
    iii. The review information is not accurate and the system goes offline

Archiving Test Cases

1. Archival process activates
    a. Requirements Tested
        i. AR01, AR02, AR03, AR04, AR05, AR06, AR07
    b. Dependencies
        i. Persistent data store must be active and accessible by the system
        ii. Archival destination location must have storage capacity
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Test 1:
            1. Input
                a. System local time set to: 00:00:00 AM, January 1st, 2021
                b. Log set A
            2. Output
                a. Compressed logs
                b. True - the logs were successfully stored in the archival location
        ii. Test 2:
            1. Input
                a. System local time set to: 00:00:00 AM, January 5th, 2021
                b. Log set A
            2. Output
                a. False - No archival process took place
        iii. Test 3:
            1. Input
                a. System local time set to: 02:00:00 AM, January 1st, 2021
                b. Log set A
            2. Output
                a. False - No archival process took place
    e. Test Execution Steps
        i. The logs listed in the input are manually placed into the data store system
        ii. The internal clock of the system is set to the time in the input
        iii. After 60 seconds, verify the archival process has been completed
    f. Success Cases

i. Archival process executes at 00:00:00AM on the 1st of the month. All log entries older than 30 days are compressed and relocated to the archival location, then removed from the data store.

g. Failure Cases
    i. Archival process did not start at 00:00:00AM local time
    ii. Archival process started at a time other than 00:00:00AM local time
    iii. Archival process started, but not on the 1st of the month
    iv. Archival process did not archive all log entries older than 30 days
    v. Some or all logs less than 30 days old were archived
    vi. Archival process did not consolidate log entries
    vii. Archival process did not compress all log entries
    viii. Archival process did not relocate consolidated and compressed logs to another location
    ix. Archival process did not remove the logs after successfully archiving them
    x. Archival process took longer than 60 seconds to complete

Usage Analysis Dashboard Test Cases

1. Navigate to View - Success
    a. Requirements Tested
        i. AS01, AS02, AS03, AS04, AS05, AS06, AS07
    b. Dependencies
        i. User must be an admin
        ii. User must be on usage analysis dashboard view
        iii. User must have authenticated session
        iv. Relational database is active and accessible by system
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. n/a
        ii. Output
            1. True - navigated to page
    e. Test Execution Steps
        i. User is on home page
        ii. User selects settings
        iii. User selects admin
        iv. User selects usage analysis dashboard
    f. Success Cases
        i. Usage analysis dashboard view loads within 15 seconds
        ii. All graph are presented in correct manner
        iii. All key performance indicator data automatically refreshes data within 60 seconds
    g. Failure Cases
        i. User is unable to navigate to page despite being admin
        ii. Usage analysis dashboard view does not load within 15 seconds
        iii. No key performance indicator data is refreshed
        iv. Some key performance indicator data is refreshed but not all
        v. The key performance indicator data takes longer than 60 seconds to refresh
2. Navigate to View - Failure to navigate to page
    a. Requirements Tested
        i. AS01, AS02, AS03, AS04, AS05, AS06, AS07

b. Dependencies
    i. User is a student
    ii. User must be on usage analysis dashboard view
    iii. User must have authenticated session
    iv. Relational database is active and accessible by system
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. n/a
    ii. Output
        1. False - unable to navigate to page
e. Test Execution Steps
    i. User is on home page
    ii. User selects settings
    iii. User selects admin
    iv. User selects usage analysis dashboard
f. Success Cases
    i. User is not able to access usage analysis dashboard as they do not have correct access level
g. Failure Cases
    i. User is able to access dashboard

3. Graphs - Correct Data
    a. Requirements Tested
        i. AS02, AS03
    b. Dependencies
        i. User must be an admin
        ii. User must be on usage analysis dashboard view
        iii. User must have authenticated session
        iv. Relational database is active and accessible by system
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. n/a

        ii. Output
            1. True - graphs are correct
    e. Test Execution Steps
        i. User is on home page
        ii. User selects settings
        iii. User selects admin
        iv. User selects usage analysis dashboard
    f. Success Cases
        i. Top five most visited views are displayed in a bar chart
        ii. Top 5 average duration per view of all time displayed in bar chart
        iii. The number of logins per day within the span of 3 months displayed in trend chart
        iv. The number of registrations per day within the span of 3 month displayed in trend chart
        v. App specific displayed in
        vi. App specific displayed in
    g. Failure Cases
        i. Any graph is displayed in incorrect manner
        ii. Any graph fails to display on dashboard
4. Refresh Graphs
    a. Requirements Tested
        i. AS02, AS03, AS05, AS07
    b. Dependencies
        i. User must be an admin
        ii. User must be on usage analysis dashboard view
        iii. User must have authenticated session
        iv. Relational database is active and accessible by system
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. n/a
        ii. Output
            1. True - graphs automatically refresh
    e. Test Execution Steps
        i. User is on home page
        ii. User selects settings

iii. User selects admin
iv. User selects usage analysis dashboard

f. Success Cases
    i. All key performance indicator graphs automatically refreshes data within 60 seconds
g. Failure Cases
    i. No key performance indicator graphs automatically refreshes data within 60 seconds
    ii. Some key performance indicator graphs automatically refreshes data within 60 seconds, but not all
    iii. All key performance indicator graphs automatically refreshes data, but takes longer than 60 seconds

Authorization Test Cases

1. Authorize User to Access UM - Requires Admin Role - Success
    a. Requirements Tested
        vii. AU01, AU06, AU07
    b. Dependencies
        i. The user must be logged in already
        ii. User must be active
        iii. User must be an admin
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Username: abrio
            2. Role: admin
        ii. Output
            1. True - access granted
    e. Test Execution Steps
        i. User is on homepage
        ii. User selects settings
        iii. User selects admin
        iv. User is taken to admin settings
    f. Success Cases
        i. User is taken to admin settings view and can perform read and write operations
    g. Failure Cases
        i. User is denied access to admin settings view
        ii. System goes offline
2. Authorize User to Access UM - Requires Admin Role - Failure
    a. Requirements Tested
        i. AU01, AU03, AU04, AU05, AU07
    b. Dependencies
        i. The user must be logged in already
        ii. User must be active
        iii. User must be an student
    c. Assumptions
        i. User is connected to internet

ii. User does not lose connection to internet

d. Test Data
    i. Input
        1. Username: mkriesel
        2. Role: student
    ii. Output
        1. False - access denied

e. Test Execution Steps
    i. User is on homepage
    ii. User selects settings
    iii. User selects admin
    iv. User is not taken to admin settings - access denied

f. Success Cases
    i. User is denied access with message saying, “Unauthorized Access”
    ii. Unauthorized access is recorded by system
    iii. User cannot access admin settings view with message saying, “Unauthorized Access to View”
    iv. User cannot access any data privileged to admins and message displays saying, “Unauthorized access to data”

g. Failure Cases
    i. User is granted access to admin settings view
    ii. User can read and write to user management functions
    iii. Unauthorized access attempt is not recorded by the system

3. Authorize User to Access Aid Eligibility - Success
    a. Requirements Tested
        i. AU02, AU06, AU07
    b. Dependencies
        i. The user must be logged in already
        ii. User must be active
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input
            1. Username: abrio
            2. Password: Pas$word1
        ii. Output

1. True - access granted

e. Test Execution Steps
   i. User is on home page
   ii. User selects aid eligibility
   iii. User logs in using above abrio as username and Pas$word1 as password
   iv. User is taken to aid eligibility view

f. Success Cases
   i. User granted access to aid eligibility view and can perform read and write operations

g. Failure Cases
   i. User is not taken to aid eligibility view

4. Authorize User to Access Aid Eligibility - Failure
   a. Requirements Tested
      i. AU02, AU03, AU04, AU05, AU07
   b. Dependencies
      i. The user must be logged in already
      ii. User must be active
   c. Assumptions
      i. User is connected to internet
      ii. User does not lose connection to internet
   d. Test Data
      i. Input
         1. Username: abrio
         2. Password: Pas$word2
      ii. Output
         1. False - access not granted
   e. Test Execution Steps
      i. User is on home page
      ii. User selects aid eligibility
      iii. User logs in using above abrio as username and Pas$word2 as password
      iv. User is denied access and not taken to aid eligibility view
   f. Success Cases
      i. User is denied access with message saying, “Unauthorized Access”
      ii. Unauthorized access attempt is recorded by system
      iii. User cannot access aid eligibility view with message saying,

"Unauthorized Access to View"

iv. User cannot access any data privileged to an authorized user and message displays saying, "Unauthorized access to data"

g. Failure Cases
    i. User is granted access to User eligibility view
    ii. Unauthorized access attempt is not recorded by the system

Account Deletion Test Cases

1. Account Deleted - Success - Admin Deleting Admin
    a. Requirements Tested
        iii. AD01, AD02, AD03, AD04
    a. Dependencies
        i. The user must be logged in already
        ii. User must be active
        iii. User must be an admin
    b. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    c. Test Data
        i. Input
            1. User account to delete: bnickle
            2. Role: admin
        ii. Output
            1. True - user deleted
    d. Test Execution Steps
        i. User is on homepage
        ii. User selects settings
        iii. User selects admin
        iv. User selects delete user bnickle
        v. User selects yes
    e. Success Cases
        i. Bnickle is successfully deleted from the database along with all personal information
        ii. System message says “Account deletion successful”
        iii. User navigated to home view
    f. Failure Cases
        i. Bnickle is not successfully deleted nor is all personal information permanently deleted
        ii. System message not displayed or is displayed showing incorrect message
        iii. User is not taken to home page
        iv. System goes offline
2. Account Deleted - Success - User Deleting Themselves
    a. Requirements Tested

iv. AD02, AD03, AD04

b. Dependencies
    i. The user must be logged in already
    ii. User must be active
c. Assumptions
    i. User is connected to internet
    ii. User does not lose connection to internet
d. Test Data
    i. Input
        1. User account to delete: bnickle
        2. Role: student
    ii. Output
        1. True - user deleted
e. Test Execution Steps
    i. User is on homepage
    ii. User selects settings
    iii. User selects delete account
    iv. User selects yes
f. Success Cases
    i. Bnickle is successfully deleted from the database along with all personal information
    ii. System message says “Account deletion successful”
    iii. User navigated to home view
g. Failure Cases
    i. Bnickle is not successfully deleted nor is all personal information permanently deleted
    ii. System message not displayed or is displayed showing incorrect message
    iii. User is not taken to home page
    iv. System goes offline

3. Account Deleted - Failure- User Deleting Themselves
    a. Requirements Tested
        v. AD02, AD03, AD04
    b. Dependencies
        i. The user must be logged in already
        ii. User must be active
    c. Assumptions

i. User is connected to internet
ii. User does not lose connection to internet

d. Test Data
    i. Input
        1. User account to delete: bnickle
        2. Role: student
    ii. Output
        1. False- user not deleted

e. Test Execution Steps
    i. User is on homepage
    ii. User selects settings
    iii. User selects delete account
    iv. User selects yes

f. Success Cases
    i. Bnickle is not successfully deleted nor is all personal information permanently deleted
    ii. System message not displayed or is displayed showing incorrect message
    iii. User navigated to home view

g. Failure Cases
    i. User is not taken to home page
    ii. Bnickle is successfully deleted from the database along with all personal information
    iii. System message says “Account deletion successful”
    iv. System goes offline

4. Account Deleted - Failure- Admin Deleting Admin
    a. Requirements Tested
        vi. AD01, AD02, AD03, AD04
    b. Dependencies
        i. The user must be logged in already
        ii. User must be active
        iii. User must be an admin
    c. Assumptions
        i. User is connected to internet
        ii. User does not lose connection to internet
    d. Test Data
        i. Input

1. User account to delete: bnickle
2. Role: admin

ii. Output

1. False - user not deleted

e. Test Execution Steps
   i. User is on homepage
   ii. User selects settings
   iii. User selects admin
   iv. User selects delete user bnickle
   v. User selects yes
f. Success Cases
   i. Bnickle is not successfully deleted nor is all personal information permanently deleted
   ii. System message not displayed or is displayed showing incorrect message
   iii. User is not taken to home page
g. Failure Cases
   i. Bnickle is successfully deleted from the database along with all personal information
   ii. System message says “Account deletion successful”
   iii. User navigated to home view
   iv. User can acknowledge the system message
   v. System goes offline