Test Plan

Project Name: Tresearch
Application Type: Web Application

Trial By Fire
Jessie Lazo
Matthew Chen (Team Lead)
Pammy Poor
Viet Nguyen

Instructor: Vatanak Vong

Submission Date
10/27/2021

California State University, Long Beach
College of Engineering

1. **Introduction**
   - **1.1 Purpose**

     This test plan documents the approach and methodologies used to ensure that each application feature meets the full requirements and satisfies the use case scenarios listed in the business requirement document.
   - **1.2 Project Overview**

     Tresearch is an interactive mind mapping tool for documenting users' learning journey. Users can create their own knowledge tree(s) to document what they are currently learning and have learned throughout their life. Knowledge trees are made up of nodes that consist of a title, description/summary, and optional tag. Branches that come off of nodes point to nodes that utilize or require knowledge from the previous node. Users can view any other user's public knowledge trees to see what they are learning or have learned and how. If a user finds another user's tree or a portion of their tree to be particularly useful or helpful, they can rate a particular node or section of the tree and they can also copy that section over to their own tree. Users can add additional information to their public profiles such as what they are currently learning, and where they are working/what they currently do or have done. Users can utilize a search function in order to find people whose trees contain a certain topic or keyword/phrase, and can also utilize a filter to narrow searches by users who are learning said topic, are using said topic (i.e. in their work or otherwise), by rating, and by tags.
   - **1.3 References**

     Documents that will be referenced include:
     - Business Requirements Document
     - Technical Specifications
2. **Scope**
   - **2.1 Features to be tested**

     The features to be tested include:
     - Creating a root node
     - Creating new node(s)
     - Copying a node(s)
     - Pasting a node(s)
     - Setting a node(s) private/public state
     - Changing the contents of a node
     - Changing the parent of a node
     - Deleting node(s)
     - Searching for a topic
       - Applying filters before searching
       - Applying filters after searching
     - Rating node(s)

3. **Testing Environment**

   Testing will be done using the latest version of the supported browser, Google Chrome v.94.0.4606.71+, as noted in the Technical Specifications documented. The hardware and operating system will go as follows:

   - Processor - Intel(R) Core™ i7-8750H CPU at 2.20 GHz
   - RAM - 16 GB DDR4-2667 MHz Dual Channel
   - Storage - Samsung 970 EVO Plus 1 TB NVMe SSD (Read 3,500 MB/s, Write 3,300 MB/s)
   - Windows 10 Home 64 bit

   Tools to be used include NUnit for automatic unit testing scripts.

4. **Testing Criteria**

   4.1 Exit Criteria

   In order to meet the exit criteria, each test case must meet a 100% run rate as well as a 95% pass rate.

   4.2 Suspension Criteria

   If testing has a failure rate of more than 50%, testing will be suspended until bugs identified are fixed by developers. Testing will be restarted once the bugs identified are fixed.

5. **Testing Type**

   5.1 Unit Testing

   During the development phase of each application feature, the developer will test that each feature performs as expected. Unit testing ensures that bugs are fixed early in development. Unit tests will be done manually automatically using NUnit.

   5.2 Integration Testing

   After code is pieced together, developers will perform testing done on each feature. This is to make sure that each feature cohesively works together. These tests are the same done in the unit tests of each feature but done so when all code is working together.

   5.3 System Testing

   Testing will be done on features not worked by testers that did not have a part in writing the specific feature. These tests will be done both manually and automated. Automated testing will be done using NUnit.

6. **Test Logistics**

Task	Members	Estimated Effort (hours)
Create Test Scripts	Test designer (cannot be individual's code)	30 hours
Perform Tests	Tester	5 hours
Test Report	Tester	10 hours

7. **Roles and Responsibilities**

7.1 Developer

- Develop individual features that align with specific use cases
- Create Scripts used for unit testing
- Conduct Unit Testing
- Produce Report for Unit Testing

7.2 Test Designer

- Must not be developer for specific feature
- Create scripts for system testing

7.3 Tester

- Conduct System Test scripts
- Produce Report for System Tests

8. **Test Schedule**

Feature	Estimated Effort Points	Expected Starting Date	Expected Completion Date
Creating Nodes	5 Hours	2/18/2022	3/2/2022
Copying Nodes	5 Hours	3/18/2022	3/30/2022
Pasting Nodes	5 Hours	4/1/2022	4/13/2022
Setting Privacy Modes	3 Hours	3/4/2022	3/16/2022
Editing a Node	2 Hours	4/1/2022	4/13/2022
Editing Node Tags	2 Hours	4/1/2022	4/13/2022
Changing a Parent Node	6 Hours	4/15/2022	4/27/2022
Deleting Nodes	5 Hours	3/4/2022	3/16/2022
Searching	6 Hours	3/4/2022	3/16/2022
Node Rating	2 Hours	4/1/2022	4/13/2022
Total Time To Execute Test Plan	41 Hours	2/18/2022	4/13/2022

Test Case(s)	Estimated Effort Points	Expected Starting Date	Expected Completion Date
Creating a Root Node	2.5 Hours	2/18/2022	3/2/2022
Creating a Node in an Existing Tree	2.5 Hours	2/18/2022	3/2/2022
Copying a Node	2.5 Hours	3/18/2022	3/30/2022
Copying Multiple Nodes	2.5 Hours	3/18/2022	3/30/2022
Pasting Node(s) into a new tree	2.5 Hours	4/1/2022	4/13/2022
Pasting Node(s) into an existing tree	2.5 Hours	4/1/2022	4/13/2022
Setting a node private/public	3 Hours	3/4/2022	3/16/2022
Changing contents of a node	2 Hours	4/1/2022	4/13/2022
Adding a tag to a node	1 Hours	4/1/2022	4/13/2022
Removing a tag from a node	1 Hours	4/1/2022	4/13/2022
Changing a parent of a node(s)	6 Hours	4/15/2022	4/27/2022
Deleting a Single Node	2.5 Hours	3/4/2022	3/16/2022
Deleting Multiple Nodes	2.5 Hours	3/4/2022	3/16/2022
Searching For a Topic	2 Hours	3/4/2022	3/16/2022
Applying Filters to Search Results	1.5 Hours	3/4/2022	3/16/2022
Applying Filters Before Searching	1.5 Hours	3/4/2022	3/4/2022
Rating a Node	1 Hour	4/1/2022	4/13/2022
Rating Multiple Nodes	1 Hour	4/1/2022	4/13/2022

9. **Test Cases**
    9.1 Creating a Root Node (Estimated 2.5 hours)
    a. Test Case Description
        i. Verify that a user can create a root node. A root node is a brand new tree.
    b. Preconditions
        i. User is logged on with a valid username and password
        ii. User has internet connection
        iii. User is currently on user's portal
    c. Test Steps
        i. Click on create a tree button
        ii. If user has reached soft limit, user will
            1. Solve captcha
            2. Click *Ok*
        iii. Enter tree name
        iv. Enter description
        v. Click the save button
    d. Specific Input Example
        i. Name: Underwater Basket Weaving
        ii. Description: A course considered by many to be the most difficult capstone course offered at CSULB
    e. Expected result
        i. Database will create a new entry for the tree
        ii. Tree name and description will be saved in the new entry
    f. Fail Cases
        i. Already existing node/tree with title in user's portal
            1. Error message explaining duplicate title
            2. Root node will not be created
            3. User will be directed back to user portal
        ii. Loss of internet connection
            1. Error message explaining loss of internet connection
            2. No changes made in database
            3. User will be directed to user portal
        iii. User has reached hard limit
            1. Error message explaining user has reached maximum amount of nodes created
            2. No changes made in database
            3. Message displayed to user stating options to create a new tree
            4. User will be directed to user portal
        iv. No connection made to cloud server
            1. Error message explaining loss of connection to cloud server
            2. User will be directed to user portal
    g. Non-functional Requirements
        i. User can begin filling out input fields within 5 seconds
        ii. User tree data is updated within 5 seconds
        iii. User tree page updated within 5 seconds
        iv. Function will be accessible 90% of the time

v. Time to repair function will be at least 1 hour and within 24 hours on average

9.2 Creating a Node in an Existing Tree (Estimated 2.5 Hours)

a. Test Case Description
   i. Verify that a user can create a node in an existing tree
b. Preconditions
   i. User is logged on with a valid username and password
   ii. User has internet connection
   iii. User is currently on user's portal
   iv. User has an existing tree
   v. User knows which node will be parent node
c. Test Steps
   i. User clicks the parent node
   ii. User clicks *Add Node* on the context menu
   iii. If user has reached soft limit, user will
      1. Solve captcha
      2. Click *Ok*
   iv. Enter name
   v. Enter description
   vi. Click *Save*
d. Specific Input Example
   i. Name: Octopush
   ii. Description: Contact sport in which two teams attempt to move a puck into the opposing team's goal. The sport takes place on the bottom of a swimming pool.
e. Expected Result
   i. Database will create a new entry for the node
   ii. Node name and description will be saved in the new entry
f. Fail Cases
   i. Already existing node with title in user's portal
      1. Error message explaining there is a duplicate title in user's portal
      2. Node will not be added to tree database
      3. User will be directed back to user portal
   ii. Loss of internet connection
      1. Error message explaining user has not internet connection
      2. No changes made in database
      3. User will be directed to user portal
   iii. User has reached hard limit
      1. Error message explaining user has reached maximum amount of nodes created
      2. No changes made in database
      3. Message displayed to user stating options to create a new node
      4. User will be directed to user portal
   iv. No connection made to cloud server
      1. Error message explaining loss of connection to cloud server
      2. User will be directed to user portal
g. Non-functional requirements
   i. User can begin filling out input fields within 5 seconds
   ii. User tree data is updated within 5 seconds
   iii. User tree page updated within 5 seconds
   iv. Function will be accessible 90% of the time

v. Time to repair function will be at least 1 hour and within 24 hours on average

9.3 Copying A Single Node (Estimated 2.5 Hours)

a. Test Case Description
   i. Verify that a user can copy a single node into a user's clipboard
b. Preconditions
   i. User is logged on with a valid username and password
   ii. User has internet connection
   iii. User is on another user's tree portal page
   iv. User does not own node to be copied
c. Test Steps
   i. Click on the node to be copied
   ii. Click on *Copy Node* on the context menu
d. Specific Input Example
   i. None
e. Expected Result
   i. Node that are to be copied are now indicated through a blue hue on the copied node
   ii. Node information is copied onto the user's clipboard. This includes
      1. Node name
      2. Node description
      3. All tags
      4. Privacy settings
f. Fail Cases
   i. Loss of internet connection
      1. Error message explaining user has lost internet connection
      2. Message displayed stating that clipboard has been wiped
      3. Clipboard will clear copied noes
      4. User will be directed to original tree portal page
   ii. No Connection to cloud server
      1. Error message explaining loss of connection to cloud server
      2. Message displayed stating that clipboard has been wiped
      3. Clipboard will clear copied nodes
g. Non-functional requirements
   i. Copy data will be updated within 5 seconds
   ii. Function will be accessible 90% of the time
   iii. Time to repair function will be at least 1 hour and within 24 hours on average

9.4 Copying Multiple Nodes (Estimated 2.5 hours)

a. Test Case Description
    i. Verify that a user can copy multiple nodes into a user's clipboard.
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on another user's tree portal page
    iv. User does not own nodes that are to be copied
c. Test Steps
    i. Shift + click nodes that are to be copied
    ii. Click on *Copy Nodes* on the context menu
d. Specific Input Example
    i. None
e. Expected Result
    i. Nodes that are to be copied are now indicated through a blue hue on the copied nodes
    ii. Node information is copied onto the user's clipboard. This includes
        1. Node name
        2. Node description
        3. Node tags
        4. Node privacy settings
f. Fail Cases
    i. Copying nodes from different subtrees
        1. Error message explaining that copied nodes have to come from one branch
        2. Message displaying how to copy nodes that will appear
        3. User sent back to original tree portal page
        4. Nothing is copied into clipboard
    ii. Loss of internet connection
        1. Error message explaining user has lost internet connection
        2. Message displayed stating that clipboard has been wiped
        3. Clipboard will clear copied nodes
        4. User will be directed to original tree portal page
    iii. No Connection to cloud server
        1. Error message explaining loss of connection to cloud server
        2. Message displayed stating that clipboard has been wiped
        3. Clipboard will clear copied nodes
g. Non-functional requirements
    i. Copy data will be updated within 5 seconds
    ii. Function will be accessible 90% of the time
    iii. Time to repair function will be at least 1 hour and within 24 hours on average

9.5 Pasting node(s) into a New Tree (Estimated 2.5 hours)

a. Test Case Description
    i. Verify that a user can paste node(s) from a user's clipboard into a new tree on the user's tree portal
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on user's tree portal page
c. Test Steps
    i. Click on empty space on user portal page
    ii. Click *Paste Node(s)* on context menu
    iii. If user has reached soft limit, user will
        1. Solve captcha
        2. Click *Ok*
    iv. Click the *Save* button
d. Specific Input Example
    i. None
e. Expected Result
    i. Nodes will be created
    ii. Pasted node will include from copied node
        1. Node name
        2. Node description
        3. Node tags
        4. Node privacy settings
    iii. Database will be updated with new entry of node with above entry
f. Fail Cases
    i. Pasted nodes will have exceeded user's hard limit
        1. Error message that pasted nodes will have exceeded user's hard limit
        2. Nodes will not be pasted
        3. User will be directed back to user tree portal page
    ii. Copy data storage empty
        1. Error message that nothing was copied
        2. User will be directed back to user tree portal page
    iii. Loss of internet connection
        1. Error message explaining user has lost internet connection
        2. Message displayed stating that clipboard could not be pasted
        3. User will be directed to original tree portal page with no changes
    iv. No Connection to cloud server
        1. Error message explaining loss of connection to cloud server
        2. Message displayed stating that clipboard could not be pasted
        3. User will be directed to original tree portal page with no changes
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on average

9.6 Pasting node(s) into an Existing Tree (Estimated 2.5 hours)

a. Test Case Description
    i. Verify that a user can paste node(s) into an existing tree on the user's tree portal
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on user's tree portal page
    iv. User's tree portal page contains an existing tree with at least one node
c. Test Steps
    i. Click on the node whom will be the parent node
    ii. Click on *Paste Node(s)* on the context menu
    iii. If user has reached soft limit, user will
        1. Solve captcha
        2. Click *Ok*
    iv. Click the *save* button
d. Specific Input Example
    i. None
e. Expected Result
    i. Nodes will be created
    ii. Pasted node will include from copied nodes
        1. Node name
        2. Node description
        3. Node tags
        4. Node privacy settings
f. Fail Cases
    i. Pasted nodes will have exceeded user's hard limit
        1. Error message that pasted nodes will have exceeded user's hard limit
        2. Nodes will not be pasted
        3. User will be directed back to user tree portal page
    ii. Copy data storage empty
        1. Error message that nothing was copied
        2. User will be directed back to user tree portal page
    iii. Loss of internet connection
        1. Error message explaining user has lost internet connection
        2. Message displayed stating that clipboard could not be pasted
        3. User will be directed to original tree portal page with no changes
    iv. No Connection to cloud server
        1. Error message explaining loss of connection to cloud server
        2. Message displayed stating that clipboard could not be pasted
        3. User will be directed to original tree portal page with no changes
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on average

9.7Setting node(s) private/public settings (Estimated 3 hours)

a. Test Case Description
    i. Verify that a user can toggle the privacy settings between private and public
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on user's tree portal page
    iv. User's tree portal page contains an existing tree with at least one node
c. Test Steps
    i. User shift + clicks on node(s)
    ii. User clicks on Edit Privacy
    iii. User toggle clicks between privacy states
    iv. User clicks *OK*
    v. User clicks *Save*
d. Specific Input Example
    i. None
e. Expected Result
    i. Private/Public status is updated within the database
    ii. User is returned to the user tree portal
f. Fail Cases
    i. Nodes parent has a private setting and the current node is being set public
        1. Error message explaining that a public node cannot have a private parent
        2. Privacy state is not changed
        3. User will be directed back to user tree portal page
    ii. Loss of internet connection
        1. Error message explaining user has lost internet connection
        2. Database privacy setting is not changed
        3. User will be directed to original tree portal page with no changes
    iii. No Connection to cloud server
        1. Error message explaining loss of connection to cloud server
        2. Message displayed stating that clipboard could not be pasted
        3. User will be directed to original tree portal page with no changes
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on average

9.8 Changing contents of a node (Estimated 2 hours)

a. Test Case Description
    i. Verify that a user can change the contents of a node
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on user's tree portal page
    iv. User's tree portal page contains an existing tree with at least one node
c. Test Steps
    i. Click on node
    ii. Click on *Edit Node* on the context menu
    iii. Enter Name
    iv. Enter Description
    v. Click the save button
d. Specific Input Example
    i. Name: Underwater Basket Weaving
    ii. Description: Testing
e. Expected Result
    i. Database will be updated with changes done on the node
    ii. Tree name and description will be saved in the updated entry
f. Fail Cases
    i. Already existing node/tree with title in user's portal
        1. Error message explaining duplicate title
        2. Root node will not be created
        3. User will be directed back to user portal
    ii. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    iii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on average

9.9 Adding a tag to a node (Estimated 1 hour)

a. Test Case Description
   i. Verify that a user can add a tag to an existing node that the user owns
b. Preconditions
   i. User is logged on with a valid username and password
   ii. User has internet connection
   iii. User is on user's tree portal page
   iv. User's tree portal page contains an existing tree with at least one node
c. Test Steps
   i. Click on node (or shift + click multiple nodes)
   ii. Click on *Edit Tag(s)*
   iii. Search for relevant tags in search bar
   iv. Click on tag to add
   v. Click *Close*
   vi. Click *Save*
d. Specific Input Example
   i. Tag Names: Math, UX, Discord
e. Expected Result
   i. Tag is added to database
   ii. Tag pops up on side of node view
   iii. Tag pops up with other existing tags when editing tags
f. Fail Cases
   i. Tag does not exist
      1. Error message explaining tag does not exist in search bar
      2. User can continue searching in search bar
   ii. Loss of internet connection
      1. Error message explaining loss of internet connection
      2. No changes made in database
      3. User will be directed to user portal
   iii. No connection made to cloud server
      1. Error message explaining loss of connection to cloud server
      2. User will be directed to user portal
g. Non-functional requirements
   i. User tree data updated within 5 seconds
   ii. User tree page visually updated within 5 seconds
   iii. Function will be accessible 90% of the time
   iv. Time to repair function will be at least 1 hour and within 24 hours on average

9.10 Deleting a tag from a node (Estimated 1 hour)

a. Test Case Description
    i. Verify that a user can remove a tag to an existing node that the user owns
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on user's tree portal page
    iv. User's tree portal page contains an existing tree with at least one node
    v. User's node has at least one tag to remove
c. Test Steps
    i. Click on node (or shift + click multiple nodes)
    ii. Click on *Edit Tag(s)*
    iii. Click on the *x* on the tag to remove
    iv. Click *Close*
    v. Click *Save*
d. Specific Input Example
    i. None
e. Expected Result
    i. Tag is removed from database
    ii. Tag no longer appears on nodes view
    iii. Tag no longer appears on existing tags when editing tags
f. Fail Cases
    i. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    ii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on

9.11 Changing a parent of a node(s) (Estimated 6 hours)

a. Test Case Description
    i. Verify that the user can change the parent of a node
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on user's tree portal page
    iv. User's tree portal page contains an existing tree with at least two nodes
c. Test Steps
    i. Click on node
    ii. Click on *Change Parent*
    iii. Nodes that can be used will be highlighted
    iv. Click new parent to be used
    v. Click *Save* button
d. Specific Input Example
    i. None
e. Expected Result
    i. Database will update node's parent
    ii. User will see that the node's parent has been changed
f. Fail Cases
    i. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    ii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on

9.12 Deleting a Single Node (Estimated 2.5 hours)

a. Test Case Description
    i. Verify that a user can delete a single node
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on user's tree portal page
    iv. User's tree portal page contains an existing tree with at least one node
c. Test Steps
    i. Click on node
    ii. Click *Delete Node*
    iii. Click Save Button
d. Specific Input Example
    i. None
e. Expected Result
    i. Node is removed from database
    ii. Node no longer appears on user's tree portal page
f. Fail Cases
    i. Deleting a parent of a dangling node
        1. Error message explaining that node cannot be deleted without deleting child
        2. No changes made in database
        3. User will be redirected to user portal
    ii. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    iii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on

9.13 Deleting Multiple Nodes (Estimated 2.5 Hours)

a. Test Case Description
    i. Verify that a user can delete multiple nodes at once
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on user's tree portal page
    iv. User's tree portal page contains an existing tree with at least two nodes
c. Test Steps
    i. Shift + click on node
    ii. Click *Delete Nodes*
    iii. Click Save Button
d. Specific Input Example
    i. None
e. Expected Result
    i. Nodes are removed from database
    ii. Nodes no longer appears on user's tree portal page
f. Fail Cases
    i. Deleting a parent of a dangling node
        1. Error message explaining that node cannot be deleted without deleting child
        2. No changes made in database
        3. User will be redirected to user portal
    ii. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    iii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on

9.14 Searching for a topic without any filters (Estimated 2 hours)

a. Test Case Description
    i. Verify that a user can search for node topics
b. Preconditions
    i. User has internet connection
    ii. User is using search bar (either on search page or search bar on homepage)
c. Test Steps
    i. User enters topic to search
    ii. User presses *Enter*
d. Specific Input Example
    i. Search Example: UX, Cooking
e. Expected Result
    i. User is directed to page listing of all nodes that contains a match to topic
    ii. If there are no resulting matches, user is notified
f. Fail Cases
    i. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    ii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on

9.15 Applying filters to the results of a search (Estimated 1.5 hours)

a. Test Case Description
    i. Verify the user can add filters to the results of a search
b. Preconditions
    i. User has internet connection
    ii. User has already searched a topic
    iii. User is on the search results page
c. Test Steps
    i. User searches for word to add to filter
    ii. User Clicks on word bank to add to filter
    iii. User can switch between radio buttons to sort by relevance or sort by rating
    iv. User clicks *Filter*
d. Specific Input Example
    i. Word bank search: cooking
e. Expected Result
    i. User is directed to page listing of all nodes that contains a match to topic
    ii. If there are no resulting matches, user is notified
f. Fail Cases
    i. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    ii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on

9.16 Applying filters before searching for a topic (Estimated 1.5 Hours)

a. Test Case Description
    i. Verify the user can add filters before searching for a topic
b. Preconditions
    i. User has internet connection
    ii. User has is on the search page
c. Test Steps
    i. User searches for word to add to filter
    ii. User Clicks on word bank to add to filter
    iii. User can switch between radio buttons to sort by relevance or sort by rating
    iv. User enters topic in search bar
    v. User clicks *Enter*
d. Specific Input Example
    i. None
e. Expected Result
    i. User is directed to page listing of all nodes that contains a match to topic
    ii. If there are no resulting matches, user is notified
f. Fail Cases
    i. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    ii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on

9.17 Rating a Single Node (Estimated 1 Hour)

a. Test Case Definition
    i. Verify that a user can rate another user's node
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on another user's tree portal page
    iv. Another user's tree portal page contains at least one nodes
c. Test Steps
    i. User clicks node
    ii. User clicks *Rate Node*
    iii. User clicks on rating to give
    iv. User clicks *close*
d. Specific Input Example
    i. None
e. Expected Result
    i. Database is updated with nodes rating
    ii. User can see rating has changed
f. Fail Cases
    i. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    ii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on

9.18 Rating Multiple Nodes (Estimated 1 Hour)

a. Test Case Definition
    i. Verify that a user can rate multiple nodes of another users
b. Preconditions
    i. User is logged on with a valid username and password
    ii. User has internet connection
    iii. User is on another user's tree portal page
    iv. Another user's tree portal page contains at least two nodes
c. Test Steps
    i. User shift + clicks node
    ii. User clicks *Rate Nodes*
    iii. User clicks on rating to give
    iv. User clicks *close*
d. Specific Input Example
    i. none
e. Expected Result
    i. Database is updated with nodes rating
    ii. User can see rating has changed
f. Fail Cases
    i. Loss of internet connection
        1. Error message explaining loss of internet connection
        2. No changes made in database
        3. User will be directed to user portal
    ii. No connection made to cloud server
        1. Error message explaining loss of connection to cloud server
        2. User will be directed to user portal
    iii.
g. Non-functional requirements
    i. User tree data updated within 5 seconds
    ii. User tree page visually updated within 5 seconds
    iii. Function will be accessible 90% of the time
    iv. Time to repair function will be at least 1 hour and within 24 hours on