

BRD

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
9/30/2021

California State University, Long Beach
College of Engineering
CECS491 Section 6 11950, Fall 2021

Functional Requirements

1. The user should be able to login
2. The user should be able to create nodes/trees
3. The user should be able to search
4. The user should be able to copy trees
5. The user should be able to hide their trees
6. The user should be able to rate other people's trees

Non-Functional Requirements

1. The website should load within 5 seconds
2. The user should be able to login within 3 seconds
3. The user should be able to create a new node within 5 seconds
4. Copying a tree/subtree/node to the user's tree should be within 5 seconds
5. The search should return results within 3 seconds
6. The user should be able create a tree up to 200 nodes in length within 24 hours
7. Nodes cannot be rapidly created or spammed in order to prevent maliciousness.
8. Application is built to comply with California law.
9. Certain material will not be allowed to be posted, i.e. harmful and deceitful material.
10. User(s) may have access/permission revoked and account suspended if found in violation of terms of service.
11. If too much traffic occurs at once, the site will go down until traffic eases up or until the problem is resolved.
12. The website is built to xzzzzzx szxc

User Requirements

1. The user is a human being, and not a bot.
2. The user must be registered/logged in to website to use the application
3. The user must have a stable internet connect for optimal performance
4. The user must be using an IP that is registered in the USA in order to use our website
5. The user has the required peripherals in order to use the site, such as monitor, mouse, and keyboard.

Use Cases

Use Case 1	Creating a node
------------	-----------------

Actors	User
Pre-Conditions	<p>Stable internet connection</p> <p>User is logged in</p> <p>User is on own tree page</p> <p>Number of nodes created by the User has not reached the hard limit</p>
Functions	<ol style="list-style-type: none"> 1. User left clicks on create node button 2. User fills input fields <ol style="list-style-type: none"> a. Title field <ol style="list-style-type: none"> i. Limited to ### characters b. Description field <ol style="list-style-type: none"> i. Limited to ### characters 3. User selects tags 4. Toggle inherit tags button 5. Toggle private/public button 6. User clicks confirm button
Variations	<ol style="list-style-type: none"> 1. User left clicks existing node <ol style="list-style-type: none"> a. User left clicks on create child button b. Same process as default c. Branch is automatically made to connect parent node to child node d. Child inherits parent private/public state e. If parent is public, child can be made private during creation
Post Conditions	<p>User tree data updated</p> <p>User soft limit count incremented by one</p> <p>If within 3 of soft limit, bring up soft limit message</p> <ul style="list-style-type: none"> • Nodes left before soft limit reached <p>If soft limit reached</p> <ul style="list-style-type: none"> • Message for soft limit reached • Trigger recaptcha • Message for how many soft limits left, how many nodes left before hard limit <p>User hard limit count incremented by one</p> <p>If within (soft limit amount) of hard limit, bring up hard limit message</p> <ul style="list-style-type: none"> • Nodes left before hard limit reached <p>If hard limit reached</p> <ul style="list-style-type: none"> • Message for hard limit reached • Trigger recaptcha
Error-Condition	<ol style="list-style-type: none"> 1. Already existing node with inputted title <ol style="list-style-type: none"> a. Error message b. Node not created c. User can change title <ol style="list-style-type: none"> i. If title changed, check again, if good then node is created d. If user exits function, data is deleted

	<ol style="list-style-type: none"> 2. Loss of internet connection <ol style="list-style-type: none"> a. Error message b. Maintain content data until User leaves function or reconnects c. If User leaves function before reconnect, node is deleted 3. User hard limit reached <ol style="list-style-type: none"> a. Error message when User clicks create button b. User can't create nodes
Non-Functional Requirements	User can begin filling out input fields within 3 seconds User tree data updated within 3 seconds User tree page visually updated within 3 seconds

Use Case 2	Copying a node
Actors	User
Pre-Conditions	Stable internet connection User is logged in User is on another user's tree page
Functions	<ol style="list-style-type: none"> 1. User left clicks node 2. User left clicks copy button
Variations	<ol style="list-style-type: none"> 1. User holds shift and left clicks multiple nodes <ol style="list-style-type: none"> a. User left clicks copy button 2. User left clicks select descendants button <ol style="list-style-type: none"> a. User left clicks copy button 3. User has nodes selected to be copied <ol style="list-style-type: none"> a. User left clicks a selected node to remove it from selection b. If removed node has descendants, those descendants are also deselected c. User left clicks copy button
Post Conditions	User copy data updated
Error-Condition	<ol style="list-style-type: none"> 1. Loss of internet connection <ol style="list-style-type: none"> a. Error message b. User copy data does not store the selected nodes c. User needs to relick copy when reconnected to internet in order to store data
Non-Functional Requirements	Copy data will be updated within 3 seconds

Use Case 3	Pasting a node
------------	----------------

Actors	User
Pre-Conditions	<p>Stable internet connection</p> <p>User is logged in</p> <p>User is on own tree page</p> <p>Number of nodes created by the User has not reached the hard limit</p>
Functions	<ol style="list-style-type: none"> 1. User right clicks open space 2. User left clicks paste button
Variations	<ol style="list-style-type: none"> 1. User left clicks a node <ol style="list-style-type: none"> a. User left clicks paste button b. Selected node is made parent of top level node in copy data
Post Conditions	<p>User tree data updated</p> <ul style="list-style-type: none"> • All copied nodes titles have original creators name appended • Nodes in copy data placed onto user tree page <p>Copy data cleared</p> <p>User tree data updated</p> <p>User soft limit count incremented by number of pasted nodes</p> <p>If within 3 of soft limit, bring up soft limit message</p> <ul style="list-style-type: none"> • Nodes left before soft limit reached <p>If soft limit reached or exceeded</p> <ul style="list-style-type: none"> • Message for soft limit reached • Trigger recaptcha • Message for how many soft limits left, how many nodes left before hard limit • If limit exceeded, number of exceeded carries over to next soft limit <p>User hard limit count incremented by number of pasted nodes</p> <p>If within (soft limit amount) of hard limit, bring up hard limit message</p> <ul style="list-style-type: none"> • Nodes left before hard limit reached <p>If hard limit reached</p> <ul style="list-style-type: none"> • Message for hard limit reached • Trigger recaptcha
Error-Condition	<ol style="list-style-type: none"> 1. Loss of internet connection <ol style="list-style-type: none"> a. Error message b. Paste prevented c. Copy data is not deleted d. User needs to reclick paste once reconnected in order to paste nodes 2. Hard limit would be exceeded <ol style="list-style-type: none"> a. Error message, notify user of how many nodes left before hard limit would be reached b. Prevent paste c. Clear copy data
Non-Functional	User tree data updated within 3 seconds

Requirements	User tree page visually updated within 3 seconds
--------------	--

Use Case 4	Setting node public/private
Actors	User
Pre-Conditions	Stable internet connection User is logged in User is on their own tree page User tree page has nodes
Functions	<ol style="list-style-type: none"> 1. User left clicks a node 2. User toggles private button 3. User left clicks save button
Variations	<ol style="list-style-type: none"> 1. Selected node has descendants <ol style="list-style-type: none"> a. All descendents private/public state changed to match selected node's state 2. User holds shift and left clicks multiple nodes <ol style="list-style-type: none"> a. Same as default
Post Conditions	User tree data updated
Error-Condition	<ol style="list-style-type: none"> 1. Loss of internet connection <ol style="list-style-type: none"> a. Error message b. State not changed until reconnect c. If User exits function before reconnect, change is not made (all nodes reverted to previous state)
Non-Functional Requirements	User tree data updated within 3 seconds User tree page visually updated within 3 seconds

Use Case 5	Changing content of a node
Actors	User
Pre-Conditions	Stable internet connection User is logged in User is on their own tree page User tree page has nodes
Functions	<ol style="list-style-type: none"> 1. User left clicks on node 2. User left clicks edit button 3. User modifies input field contents <ol style="list-style-type: none"> a. Title b. description

	4. User left clicks save button
Variations	
Post Conditions	1. User tree data updated a. Modified node content
Error-Condition	1. Loss of internet connection a. Error message b. Maintain data until User exits edit mode or reconnects c. If User exits edit mode before reconnect, revert to state before edit
Non-Functional Requirements	User tree data updated within 3 seconds User tree page visually updated within 3 seconds

Use Case 6	Changing tag of a node
Actors	User
Pre-Conditions	Stable internet connection User is logged in User is on their own tree page User tree page has nodes
Functions	1. User left clicks on node 2. User adds tags 3. User removes tags 4. User left clicks save button
Variations	1. User holds shift and left clicks multiple nodes a. Tags shown are based on last selected node b. User adds tags i. If a node already has the given tag, no changes to that nodes tags are made c. User removes tags i. If a node does not have the given tag, no changes to that nodes tags are made d. User left clicks save button
Post Conditions	2. User tree data updated a. Modified node tags
Error-Condition	2. Loss of internet connection a. Error message b. Maintain data until User exits function or reconnects c. If User exits function before reconnect, revert node(s) to previous state
Non-Functional	User tree data updated within 3 seconds

Requirements	User tree page visually updated within 3 seconds
--------------	--

Use Case 7	Deleting a node
Actors	User
Pre-Conditions	Stable internet connection User is logged in User is on their own tree page User tree page has nodes
Functions	<ol style="list-style-type: none"> 1. User left clicks on a node 2. User left clicks delete button 3. User left clicks confirm button
Variations	<ol style="list-style-type: none"> 1. User holds shift and left clicks multiple nodes <ol style="list-style-type: none"> a. User clicks delete button b. User clicks confirm button <ol style="list-style-type: none"> i. All selected nodes will be deleted 2. User left clicks select descendants button <ol style="list-style-type: none"> a. User left clicks a node b. User clicks delete button c. User clicks confirm button <ol style="list-style-type: none"> i. All selected nodes will be deleted
Post Conditions	<ol style="list-style-type: none"> 1. User tree data updated <ol style="list-style-type: none"> a. Remove node(s) b. If possible, remaining children nodes attach to parent node of highest level deleted node c. Otherwise, remaining children nodes are left hanging in the tree space
Error-Condition	<ol style="list-style-type: none"> 1. Loss of internet connection <ol style="list-style-type: none"> a. Error message b. Deletion is prevented c. User must relick delete button after reconnection in order to delete
Non-Functional Requirements	User tree data updated within 3 seconds User tree page visually updated within 3 seconds

Use Case 8	Searching for a topic
Actors	User
Pre-Conditions	Stable internet connection

	User is logged in
Functions	<ol style="list-style-type: none"> 1. User clicks on search input field 2. User types in phrase/topic to search for 3. User clicks filter button <ol style="list-style-type: none"> a. User selects filters 4. User clicks enter key or search button
Variations	<ol style="list-style-type: none"> 1. User clicks on phrase/topic provided in recent search list <ol style="list-style-type: none"> a. Input field auto fills with phrase/topic 2. User clicks on phrase/topic provided in other lists <ol style="list-style-type: none"> a. Automatically searches
Post Conditions	<ol style="list-style-type: none"> 1. System returns list of users whose tree page contains a node titled with the searched phrase/topic, filtered by selected filters 2. System search data updated <ol style="list-style-type: none"> a. Number of times searched phrase/topic has been searched for is incremented by 1
Error-Condition	<ol style="list-style-type: none"> 1. No results for searched phrase/topic <ol style="list-style-type: none"> a. No results found message 2. No internet connection <ol style="list-style-type: none"> a. Error message b. Prevent search until reconnect
Non-Functional Requirements	Return search results within 5 seconds

Use Case 9	Rating a node
Actors	User
Pre-Conditions	Stable internet connection User is logged in User is on another user's tree page Other user's tree page has public nodes
Functions	<ol style="list-style-type: none"> 1. User clicks on node 2. User clicks on rate button 3. User selects rating 4. User clicks save button
Variations	<ol style="list-style-type: none"> 1. User selects multiple nodes (either by manually clicking or toggling select descendants) <ol style="list-style-type: none"> a. User click
Post Conditions	
Error-Condition	

Non-Functional Requirements	Return search results within 5 seconds
-----------------------------	--