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
10/06/2021

Updated Date
12/11/2021

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

Scope

The project will be a multiple page web application. We will support the latest version of Chrome at the date that we start development (we will maintain our Chrome versions at this version or downgrade the presenters version to it when necessary). Our initial target audience will be American English speaking residents of the United States above the age of 15 to comply with Children's Online Privacy Protection Act (COPPA). The IP range will be restricted to the United States to maintain our target customers. The web application will utilize cookies in order to enhance the user experience and for any other functionalities that may be enhanced through their use that we choose to apply the cookies towards. Cookies will adhere to the strongest regional regulations within the United States.

Use Case 1: Creating a node

Purpose:

The purpose of this Use Case is for the User to be able to create a new tree or to grow an existing tree

Scope:

This Use Case only deals with creating one node at a time for a given tree

Actors:

- User

Pre-Conditions:

- User is an active and authenticated user
- User has a stable internet connection
- User has navigated to the user's tree overview page
- The number of nodes created by the user has not reached the hard limit

Functional Requirements:

- User must enter a valid name, description, tag and privacy mode
- Valid node name consists of
    - Minimum of 1 character and are limited to 50 characters
    - Unique in the user's account. The name must not already exist within any of the user's nodes
- Valid node descriptions consist of:
    - Minimum of 1 character and are limited to 300 characters
- Valid tags are found from a bank of tags provided
    - A created node does not require a tag
- Privacy toggles between private and public
    - By default a root node will be set public
    - Any of the root node's children will inherit it's privacy feature
    - A child cannot be public if it's parent is privacy
- A maximum of 150 nodes can be created or pasted within 24 hours with the same account before a hard limit is reached and node creation is disabled
- A maximum of 10 nodes can be created or pasted before a user reaches a soft limit.

Functions:

1. User fills input fields
    a. Title field
    b. Description field
2. User selects tags
    a. Beginner. Intermediate, Advanced, Research
3. Toggle private/public button

Variations:

1. Creating a child node
    a. Branch is automatically made to connect parent node to child node
    b. If parent is private, child is also made private

Success Conditions:

- User creates and saves a node with valid inputs. The user is automatically navigated to a page containing an overview of the created tree containing the created node. Log entry is created.

- User creates and saves a node with valid inputs. User meets the soft limit and successfully solves captcha. The user is automatically navigated to a page containing an overview of the created tree containing the created node. Log entry is created.

Error Conditions:

- User creates and saves a node with valid inputs. The automatic navigation does not take place.
- User creates and saves a node with valid inputs. User is navigated to a page other than the new tree's overview.
- User creates and saves a node with valid inputs. Database connection is broken. System message displays "This feature is currently undergoing maintenance. Please try again at a later time."
- User creates and saves a node with empty inputs. System message displays "Input field cannot be left empty."
- User attempts to create a node after the user's hard limit has been reached. System message displays "You have reached your limit for nodes today. The node was not created. Please delete some nodes or try again tomorrow."
- User creates and saves a node with a name that already exists in the user's portal. System message displays "There is already a node with that title."
- User creates a node and the log entry is not created.

Non-Functional Requirements:

- User is loaded with view containing input fields within 5 seconds
- Node is saved before moving to another page otherwise node contents are deleted.
- All create node attempts will be logged with the
    - UTC timestamp of attempt to create node
    - Log level that accurately describes event
    - Valid category that accurately describes event
    - Accurate description
- User tree data is updated within 5 seconds
- User tree view is updated with new node within 5 seconds
- Feature will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average
- All instructions and error messages will be written at the 8th grade reading level
- This feature is only authorized on a user's own tree
- Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 2: Copying a node

Purpose:

The purpose of this Use Case is for the User to be able to copy and store the data of nodes from another User's tree to later be pasted onto the User's tree

Scope:

This Use Case deals with copying and storing the data of one or more nodes owned by another User simultaneously at a single time

Pre-Conditions:

- User is an active and authenticated user
- User has a stable internet connection
- User has navigated to another user's tree page that contains at least one public node

Functions:

1. Make a copy of nodes from another User's tree

Variations:

1. Copy data already has data of other nodes
    a. Copy data is replaced with new data

Success Conditions:

- User copies a node. System message displays "Node copied."Node information is copied onto the user's clipboard. Log entry is created.

Error Conditions:

- User copies node(s). The System message does not display.
- User copies node(s). The page is navigated to any page other than the current page.
- User copies node(s). Database connection is broken. System message displays "This feature is currently undergoing maintenance. Please try again at a later time."
- User copies node(s). System message displays but log entry is not created.

Non-Functional Requirements:

- Copy data will be updated within 5 seconds
- All copy node attempts will be logged with the
    - UTC timestamp of attempt to copy node
    - Log level that accurately describes event
    - Valid category that accurately describes event
    - Accurate description
- Feature will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average
- All instructions and error messages will be written at the 8th grade reading level
- This feature is only authorized on a another users tree
- Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 3: Pasting a Node

Purpose:

The purpose of this Use Case is for the User to be able to paste any stored node data in the copy data to the User's own tree

Scope:

This use case deals with pasting any and all nodes that can be found in the copy data storage at that time

Functional Requirements:

- A maximum of 150 nodes can be created or pasted within 24 hours with the same account before a hard limit is reached and node creation is disabled
- A maximum of 10 nodes can be created or pasted before a user reaches a soft limit.

Pre-Conditions:

- User is an active and authenticated user
- User has a stable internet connection
- User is navigated to their tree page
- The number of nodes created by the user has not reached the hard limit

Functions:

1. Paste nodes in copy data to the User's tree

Success Conditions:

- User pastes nodes onto the tree page. The user is automatically navigated to a page that contains an overview of the new tree. Log entry is created.
- User pastes nodes onto the tree page. The user hits the soft limit and successfully solves the captcha. The user is automatically navigated to a page that contains an overview of the new tree. Log entry is created.

Error Conditions:

- User pastes node(s). User doesn't see the new tree with the pasted nodes
- User attempts to paste a node(s) after the user's hard limit has been reached. Message displays "You have reached your limit for nodes today. The node(s) were not created. Please delete some nodes or try again tomorrow."
- User attempts to paste multiple nodes and the user's hard limit has been reached while pasting. Message displays "You have reached your limit for nodes today. The nodes were not created. Please delete some nodes or try again tomorrow."
- Copy data storage is empty. System message displays "You have not copied any nodes. Please copy some nodes"
- Log entry is not created
- User pastes node(s) hard limit is reached but nodes are still pasted

Non-Functional Requirements:

- Tree database will be updated within 5 seconds.
- Function will be accessible 90% of the time.
- Time to repair the function will be at least 1 hour and within 24 hours on average.
- All instructions and error messages will be written at the 8th grade reading level
- This feature is only authorized on a user's own tree
- Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 4: Setting node to private/public state

Purpose:

The purpose of this Use Case is for the User to be change the visibility state of any of their nodes

Scope:

This use case deals with changing the visibility state of one or more nodes that are owned by the User simultaneously at a single time

Actors:

- User

Pre-Conditions:

- User is an active and authenticated user
- User has a stable internet connection
- User has navigated to their own tree page with at least one node

Functions:

1. User changes private/public state for any owned nodes

Success Conditions:

- User changes the privacy setting of a node. Log entry is made

Failure Conditions:

- User changes the privacy setting of a node. Privacy mode does not appear to change to updated mode
- User changes the privacy setting of a node. The user is automatically navigated to a page other than the overview of the tree containing the updated modes.
- Log entry is not made

Non-Functional Requirements:

- Overview of the tree containing the updated nodes is loaded within five seconds
- User tree database is updated within 5 seconds
- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average
- All instructions and error messages will be written at the 8th grade reading level
- This feature is only authorized on a user's own tree
- Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 5: Changing contents of a node

Purpose: The purpose of this Use Case is for the User to be able to change the content of any of their nodes

Scope: This Use Case only deals with changing the contents of a single node that is owned by the User at a time

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 modifies input field contents
    a. Title - 50 character limit
    b. Description - 300 character limit

Success Conditions:

1. User tree data updated with changed content of selected node

Error Conditions:

1. Loss of internet connection
    1.1. Error message (“Internet connection was lost. Your changes were not saved. Please try again once you reconnect.”)
    1.2. Prevent User tree data update
2. No connection to database
    2.1. Error message (“This feature is currently undergoing maintenance. Please try again at a later time.”)
    2.2. Feature is disabled until connection to the database is reestablished.

Non-Functional Requirements:

User tree data updated within 5 seconds
User tree page visually updated within 5 seconds
Feature will be accessible 90% of the time
Time to repair feature will be at least 1 hour and within 24 hours on average
All errors will be logged by the Logger, which will include the error type, date/time of the error, and the User who triggered the error
All successful node content changes will be logged by the Logger, which will include the date/time of the change was made, the data of the affected node from before the change, the change that was made, and the User who made the change
This feature is only visible and usable by valid Users who are logged in and currently on their own tree page
Ability for system admins to change character limit of input fields
Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 6: Changing tag of a node

Purpose: The purpose of this Use Case is for the User to be able to change the tags of any of their nodes

Scope: This Use Case deals with adding and removing tags from one or more nodes that are owned by the User simultaneously at a single time

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 adds or removes tags from their nodes

Success Conditions:

1. User tree data updated with changed tags of selected node(s)

Error Conditions:

1. Loss of internet connection
   - 1.1. Error message (“Internet connection was lost. Your changes were not saved. Please try again once you reconnect.”)
   - 1.2. Prevent User tree data update
2. No connection to database
   - 2.1. Error message (“This feature is currently undergoing maintenance. Please try again at a later time.”)
   - 2.2. Feature is disabled until connection to the database is reestablished.

Non-Functional Requirements:

User tree data updated within 5 seconds
User tree page visually updated within 5 seconds
Feature will be accessible 90% of the time
All instructions and error messages will be written at the 8th grade reading level
All text will be at a minimum of 16 pixels (12 point font)
Time to repair feature will be at least 1 hour and within 24 hours on average
All errors will be logged by the Logger, which will include the error type, date/time of the error, and the User who triggered the error
All successful tag changes will be logged by the Logger, which will include the date/time of the change, the data of the affected node from before the change, the change that was made, and the User who made the change
This feature is only visible and usable by valid Users who are logged in and currently on their own tree page
Ability for system admin to add, change, or remove valid tags as needed
Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 7: Changing parent of a node

Purpose: The purpose of this Use Case is for the User to be able change the parent of any of their nodes as a way to move nodes around a tree or even detach from a tree

Scope: This Use Case only deals with changing the parent of a single node owned by the User at a time

Actors: User

Pre-Conditions:

Stable internet connection
User is logged in
User is on own tree page

Functions:

1. User changes parent of a node

Variations:

1. User selects “none” for parent
    a. Selected node becomes the root node of a new tree

Success Conditions:

1. User tree data updated with change to selected nodes parent
2. If selected node has descendants, they move visually with the selected node

Error Conditions:

1. Loss of internet connection
    1.1. Error message (“Internet connection was lost. Your changes were not saved. Please try again once you reconnect.”)
    1.2. Prevent parent change
2. No connection to database
    2.1. Error message (“This feature is currently undergoing maintenance. Please try again at a later time.”)
    2.2. Feature is disabled until connection to the database is reestablished.

Non-Functional Requirements:

User tree data updated within 5 seconds
User tree page visually updated within 5 seconds
Feature will be accessible 90% of the time
All instructions and error messages will be written at the 8th grade reading level
All text will be at a minimum of 16 pixels (12 point font)
Time to repair feature will be at least 1 hour and within 24 hours on average
All errors will be logged by the Logger, which will include the error type, date/time of the error, and the User who triggered the error
All successful parent changes will be logged by the Logger, which will include the date/time of the change, the data of the affected node from before the change, the change that was made, and the User who made the change
This feature is only visible and usable by valid Users who are logged in and currently on their own tree page
Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 8: Deleting a node

Purpose: The purpose of this Use Case is for the User to be able to delete any of their own nodes. This will also allow the User to create more nodes if they have made any nodes during the time frame pertaining to the hard limit of node creation.
Scope: This Use Case deals with deleting one or more nodes that are owned by the User simultaneously at a single time
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 deletes a node(s)

Success Conditions:

1. Selected nodes are removed from user tree data
    1.1. If selected nodes had descendants that were not selected for deletion, parent(s) of highest level descendant(s) changes to parent of selected highest level selected node
        1.1.1. If it had a parent, User tree data updated with changes to highest level descendant(s) parent(s) (has a new parent)
        1.1.2. If it did not have a parent, User tree data updated with changes to highest level descendant(s) parent(s) (no parent)
2. If the User's hard limit count is greater than zero
    2.1. Subtract one from the hard limit count for each deleted node
        2.1.1. Stop of hard limit count reaches zero
    2.2. Reduce soft limit count according to amount that the hard limit count is reduced
        2.2.1. If soft limit count would reach zero, decrement soft limit number and repeat process with remaining hard limit count reduction

Error Conditions:

1. Loss of internet connection
    1.1. Error message ("Internet connection was lost. The node(s) was(were) not deleted. Please try again once you reconnect.")
    1.2. Prevent User tree data update
2. No connection to database
    2.1. Error message ("This feature is currently undergoing maintenance. Please try again at a later time.")
    2.2. Feature is disabled until connection to the database is reestablished.

Non-Functional Requirements:

User tree data updated within 5 seconds
User tree page visually updated within 5 seconds
Feature will be accessible 90% of the time
All instructions and error messages will be written at the 8th grade reading level
All text will be at a minimum of 16 pixels (12 point font)
Time to repair feature will be at least 1 hour and within 24 hours on average
All errors will be logged by the Logger, which will include the error type, date/time of the error, and the User who triggered the error

All successful node deletions will be logged by the Logger, which will include the date/time of the deletion, the number of nodes deleted, the data of the nodes that were deleted, and the User who deleted the nodes
Any changes to the Users hard or soft limits will be logged by the Logger, which will include the date/time of the change, and the change that was made (decrement of limit counts and limit number)
This feature is only visible and usable by valid Users who are logged in and currently on their own tree page
Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 9: Searching for a topic

Purpose: The purpose of this Use Case is for the User to be able search for nodes that may be found on other User's trees, with the option to apply a filter on the initial search and/or on the results of the search

Scope: This Use Case deals with the User searching for a single topic/keyword/phrase at a time, applying a filter to that initial search if they so choose, and applying a filter to the results if they so choose

Actors: User

Pre-Conditions:

Stable internet connection

Functions:

1. User searches for a node that may be found on another User's tree
2. User can apply any filters before searching
3. User can apply any filters to the results of the search

Success Conditions:

1. System returns a list of Users whose tree page contains a public node titled exactly with the searched phrase/topic (ignoring casing), filtered by selected filters
2. System search data updated
    2.1. Number of times searched phrase/topic has been searched for is incremented by 1

Error Conditions:

1. No results for searched phrase/topic
    1.1. No results found message ("No results were found for that search.")
2. Loss of internet connection
    2.1. Error message ("Internet connection was lost. Please try again once you reconnect.")
    2.2. Prevent search
3. No connection to database
    3.1. Error message ("This feature is currently undergoing maintenance. Please try again at a later time.")
    3.2. Feature is disabled until connection to the database is reestablished.

Non-Functional Requirements:

Return search results within 5 seconds
Feature will be accessible 90% of the time
All instructions and error messages will be written at the 8th grade reading level
All text will be at a minimum of 16 pixels (12 point font)
Time to repair feature will be at least 1 hour and within 24 hours on average
All errors will be logged by the Logger, which will include the error type, date/time of the error, and the User who triggered the error
All successful searches will be logged by the Logger, which will include the date/time of the search, the topic that was searched for, the number of results, and the User who performed the search (this is logged as Guest if the User is not logged in)
This feature is visible and usable by all Users who are on the search page, regardless of whether or not they are logged in
Ability for system admins to add, change, or remove ways to filter as needed
Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 10: Rating a node

Purpose:

The purpose of this Use Case is for the User to be able to rate any nodes owned by another User

Scope:

This Use Case deals with rating one or more nodes owned by another User simultaneously at a single time

Actors:

- User

Pre-Conditions:

- User is an active and authenticated user
- User has a stable internet connection
- User has navigated to another user's tree page with at least one public node

Functions:

1. User rates another User's nodes on a scale of 1 to 5

Success Conditions:

- Other User tree data updated with changes to selected nodes ratings

Error Conditions:

1. Loss of internet connection
    1.1. Error message ("Internet connection was lost. Your changes were not saved. Please try again once your reconnect")
    1.2. Prevent rating change
2. No connection to database
    2.1. Error message ("This feature is currently undergoing maintenance. Please try again at a later time.")
    2.2. Feature is disabled until connection to the database is reestablished.

Non-Functional Requirements:

Other User tree data updated within 5 seconds

Other User tree page visually updated within 5 seconds

Feature will be accessible 90% of the time

Time to repair feature will be at least 1 hour and within 24 hours on average

All errors will be logged by the Logger, which will include the error type, date/time of the error, and the User who triggered the error

All successful ratings will be logged by the Logger, which will include the date/time of the rating, the number of nodes being rated, the data of the nodes from before being rated, the change that was made, the original owner of the nodes, and the User who performed the rating

This feature is only visible and usable by valid Users who are logged in and currently on a different Users tree page

Current amount of stored data for the system must not exceed the storage amount of the databases (10GB each), more databases may be added as deemed necessary

Use Case 11: User Authentication

Purpose:

The purpose of this use case is for the user to authenticate themself as a valid user of the system.

Scope:

This Use Case deals with Users attempting to use the system

Actors:

- User
- System Administrator

Requirements:

- A valid user is defined as a user that has been authenticated by the system with valid security credentials
- Valid security credentials consist of a username and a password
    - Valid usernames will consist of the following
        - a-z
        - 0-9
        - .,@!
    - Valid security credentials will be a valid time-based one-time password (OTP) as defined in NIST SP 800-63b section 5.1.4.1. Valid passwords will be defined as follows
        - Password must be at minimum 8 characters and with a maximum of 100 characters
        - OTP passwords expire after 2 minutes
        - OTP passwords are changed after reach use
        - Valid characters will include
            - a-z
            - A-Z
            - 0-9
            - ,.@!
- Five failed authentication attempts within 24 hours will result in the account being disabled
    - The first failed authentication attempt will be the start of this 24 hour period. Once the 24 hour period is up, the number of failed attempts will be reset to zero.
    - Every failed authentication attempt will log the account attempted to be signed in and the IP address used

Pre-Conditions:

- User must **not** be an active and authenticated user otherwise the user can not attempt to authorize.
- User must navigate to the login page.
- User must have access to valid security credentials

Functions:

1. User fills in
    a. Title field
    b. Description field
2. User selects tags
    a. Beginner. Intermediate, Advanced, Research
3. Toggle private/public button

Success Conditions:

- User submits valid security credentials and is automatically navigated to the user's tree page

Failure Conditions:

- User submits valid security credentials and is not automatically navigated to any page.
- User submits valid security credentials and is automatically navigated to a page other than the user's tree page.
- User submits invalid username. A system message displays "Invalid username or password. Try again." The failed attempt count increases by one.
- User submits invalid password. A system message displays "Invalid username or password. Try again." The failed attempt count increases by one.
- User submits invalid username or invalid password. A system message displays "Invalid username or password. Try again.". The failed attempt count does not increase by one.
- User submits valid security for a disabled account. A system message displays "Account disabled. Contact system admin." Failed attempt count is increased by one.
- User submits valid security for a disabled account. A system message displays "Account disabled. Contact system admin." Failed attempt count is not increased by one.
- User submits valid security for a disabled account. A system message displays "Account disabled. Contact system admin." IP Address and account is not logged.

Non-Functional Requirements:

- Users redirection to tree overview is loaded within five seconds
- Database containing user's failed attempt count is updated within five seconds
- Database containing log is updated within five seconds
- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use Case 12: Registration Account Creation

Purpose:

Mechanism for creating new user accounts within the system.

Scope:

Any user attempting to use the system.

Actors:

- User

Requirements:

- System administrators can not be created by this feature
- All user account information will be stored in a persistent data store
- Valid security credentials consist of a username and a password
    - The users username will be a valid email and will consist of the following
        - a-z
        - 0-9
        - .,@!
    - Valid passwords will be defined as follows
        - Password must be at minimum 8 characters and with a maximum of 100 characters
        - Valid characters will include
            - a-z
            - A-Z
            - 0-9
            - ,.@!
- The user must provide their first and last name
    - Valid first and last names will consist of the following
        - a-z
        - A-Z
        - -,
    - User must certify that the user is at least 13 years of age
        - The user must be at least 13 years of age to user our system in order to comply with the Children's Online Privacy Protection Act (COPPA)

Preconditions

- User must not be an active and authorized user
- User has access to a valid email
- User has navigated to the sign up page

Function

1. User enters valid first name
2. User enters valid last name
3. User enters valid email
4. User enters valid password
5. User agrees to the policy agreement
6. User logins

Success Condition

- User registers with a valid email and valid password. User is automatically directed to a new tree overview view.

Failure Condition

- User registers with a valid email and password. User is not automatically directed to any page and displays no system message
- User registers with a valid email and password. User is automatically directed to a page other than the new tree overview view.
- User registers with an invalid email. System displays “Unable to create account. Invalid username.”
- User registers with an invalid password. System displays “Unable to create account. Invalid password. Please use a-z, A-Z, 0-9 and ,.@!”

Non-Functional Requirements:

- Users redirection to tree overview is loaded within five seconds
- Database containing accounts is updated within five seconds
- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use Case 13: Logout

Purpose:

The purpose of this mechanism is to end a session for an active and authorized user

Scope:

Any active and authorized user attempting to use the system

Requirements:

- User's sessions ends within five seconds on logout request
    - User's end of session will redirect user to homepage

Preconditions:

- User is an active and authenticated user
- User has navigated to to the users tree page overview

Function

1. User clicks *Log Out*

Success Conditions

- User clicks logout and ends the active session. The user is automatically navigated to the homepage with the options to log in /sign up at the top right.

Failure Conditions

- User clicks logout and ends the active session. The automatic navigation does not take place
- User clicks logout and ends the active session. The user is automatically navigated to a page other than the homepage.
- User clicks logout and ends the active session. User is automatically navigated to the homepage but there is no login/sign up option.
- The logout request takes longer than 5 seconds

Non-Functional Requirements:

- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use case 14: Authorization

Purpose:

The purpose of authentication is to identify if a user is a valid user of the system.

Scope:

The scope of this component covers all users that are attempting to use the system.

Requirements:

- Unauthenticated users will only be given access to resources that do not require knowledge of the user's identity. This includes
    - The home page
    - About page
    - Search feature
- Unauthorized users will not be able to view, modify or delete any protected data
- Unauthorized can not view or interact with protected views or functions
- The operation and timestamp of each unauthorized access will be recorded
- Users access modifications will be restored with unauthorized user is authenticated

Function:

1. User attempts to navigate to feature/view/data that is authorized

Preconditions

- User must is an active and authorized user

Success Condition

- User attempts to use a protected feature within authorization scope and access is granted
- User attempts to view protected data within authorization scope and access is granted
- User attempts to modify protected data within authorization scope and access is granted
- User attempts to to view protected view within authorization scope and access is granted

Failure Condition:

- User attempts to use a protected feature outside authorization scope and access is denied
- User attempts to view protected data outside authorization scope and access is denied
- User attempts to modify protected data outside authorization scope and access is denied
- User attempts to to view protected view outside authorization scope and access is denied

Non functional requirements

- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use Case 15: Deletion

Purpose:

The purpose of this mechanism is to delete a user account

Scope:

Any registered user of the system

Requirements:

- System administrators are only able to be deleted by another system administrator
- All personal identifiable information (PII) along with user's account data is deleted from the system permanently
    - This deletion is irreversible

Preconditions:

- User must be an active and authenticated user
- User be navigated to the delete account view
- User has permission to delete the account. See system administrator permission in requirements.

Function:

1. User deletes account
2. User confirms account deletion

Success Conditions:

- User deletes account and confirms deletion. All personal identifiable information is deleted and a System message displays "Account deleted." User is automatically navigated to the homepage

Failure Conditions:

- User deletes account and confirms deletion. Personal identifiable information is not deleted.
- User deletes account and confirms deletion. Automatic navigation does not take place
- User deletes account and confirms deletion. User is automatically navigated to a page other than the homepage.
- User deletes account and confirms deletion. System message does not display.

Non functional requirements

- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use Case 16: User Account Management

Purpose:

The purpose of having a User Account Management feature is to provide a mechanism for system administrators to observe and make changes to accounts if necessary

Scope:

The scope of this component covers all users that have a registered account within our system.

Requirements:

- Operations will be applied to a persistent data store.
- Users must be a system administrator to access this feature
- The system administrator has access to view and modify accounts and its corresponding data within the entire system
- System administrators can perform single operations as follows
    - Create Account
    - Update Account
    - Delete Account
    - Disable Account
    - Enable Account
    - Single Operations will be completed within 5 seconds
- System administrators can perform bulk operations
    - Bulk operations can consist of multiple operations of the same type or mixed type
    - A maximum of 10,000 operations per request
    - Requests can be made from an uploaded file extract
        - The uploaded file must be less than 2 GB in size
    - Bulk operations must be completed within 60 seconds
- Single and bulk operations will affect all users and their attributes within the system
- System administrators can create new system administrator accounts
- At any given type, the system must have at least one system administrator

Preconditions:

- User must be an active and authenticated user
- User must have system administration privileges
- User must be navigated to the user management view

Success Conditions:

- User performs a single user management operation within 5 seconds upon request. System message displays “Single operation successful.”
- User performs bulk operations containing less than 10,000 operations within 60 seconds. System message displays “Bulk operation successful.”
- User performs bulk operations containing 10,000 operations within 60 seconds. System message displays. “Bulk operation successful.”

Failure Conditions:

- User's single operation takes more than 5 seconds
- User's bulk operation takes more than 60 seconds
- 10k bulk operation takes more than 60 seconds
- Single operation takes five seconds but no system message is displayed
- Single operation takes five seconds but wrong system message is displayed
- Bulk operation takes 60 seconds but no system message is displayed
- Bulk operation takes 60 seconds but wrong system message is displayed
- 10k bulk operation takes 60 seconds but no system message is displayed
- 10k bulk operation takes 60 seconds but wrong system message is displayed
- Single operation takes five seconds and displays system message but data isn't stored in persistent database
- Bulk operation takes 60 seconds and displays system message but data isn't stored in persistent database
- 10k bulk operation takes 60 seconds and displays system message but data isn't stored in persistent database

Non-functional Requirements

- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use Case 17: Usage Analysis Dashboard

Purpose:

The purpose of this mechanism is to provide a visualization on user behavior in the system

Scope:

The scope of this component covers system administrators account of the system

Requirements:

- All data is fetched from a persistent data store
- Usage analysis dashboard contains the following key performance indicators (KPI)
    - A bar chart containing the top five most most visited views of all time
    - A bar chart containing the top five average duration per view of all time
    - A trend chart of the number of logins per day within a 3 month span
    - A trend chart of the number of registrations per day within a 3 month span
    - A bar chart containing the top five most searched topics within a week
    - A bar chart containing the top five most tagged topics within a week
- All key performance indicators are refreshed automatically every sixty seconds
- Usage analysis dashboard view must be loaded within 15 seconds

Pre-conditions:

- User is an active and authenticated user
- User is a system administrator
- User has a stable internet connection
- User has navigated to to the usage analysis dashboard

Function:

1. User can see KPIs

Success Conditions:

- User is able to navigate to the usage analysis dashboard view. The view is loaded within 15 seconds and refreshes its data within 60 seconds

Failure Conditions:

- User is unable to navigate to the usage analysis dashboard view but is a system administrator
- User is able to navigate to the usage analysis dashboard view but doesn't load within 15 seconds
- User is able to navigate to the usage analysis dashboard view within 15 seconds but the key performance indicators data is not refreshed every 60 seconds
- User is able to navigate to the usage analysis dashboard view within 15 seconds but not all of the key performance indicator data is refreshed within 60 seconds
- User is able to navigate to the usage analysis dashboard view within 15 seconds

Non-Functional Requirements:

- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use Case 18: Logging

Purpose:

The purpose of this mechanism is to provide event tracking for system audits

Scope:

The scope of this component covers system events and user events.

Requirements:

- Log entries are immutable (unable to be changed)
- Log entries will be stored in a persistent database
- All log entries will consist of
    - UTC timestamp of occurrence
    - Log Level
    - User who performed operation
    - Category of log
    - Description of log
- Valid log levels include
    - Info - tracks the flow of the system
    - Debug - information to help maintainers of the system
    - Warning - events that can show system failures
    - Error - shows system errors
- Valid categories
    - View
    - Business
    - Server
    - Data
    - Data store
- Logging process must not interfere or block interactions within the system
- Logging process must not take more than five seconds.

Function:

1. Deployed with every feature and component

Pre-conditions:

- Data is stored within an active persistent data store
- Persistent data store must be accessible by the system
- Persistent data store must have storage capacity for a new log entry

Success Conditions:

- System logs successful user events within five seconds
- System logs failed user events within five seconds
- System logs successful user events within five seconds
- System logs failed user events within five seconds

Failure Conditions:

- System log process takes longer than five seconds
- System log process interferes or blocks interactions within the system
- System log doesn't store to a persistent database
- System log data store doesn't have the capacity to store a new long entry
- System log event is able to be modified
- System log event is stored with inaccurate entry

Non-Functional Requirements:

- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use Case 19: Archiving

Purpose:

The purpose of this mechanism is to offload log entries to preserve system resources

Scope:

The scope of this component covers all log entries made by the system

Requirements

- On every 1st of the month at 00:00:00AM (local time) the archival process will execute
- Only log entries older than 30 days will be offloaded
- Archived entries will be consolidated and compressed
- Offloaded log entries will be moved to another location
- Offloaded log entries will be removed after successful archival
- Log entries archival process must be completed within 60 seconds of request

Pre-conditions:

- Log entries must be stored in a persistent data store
- Persistent data stored must be accessible from the system
- Location to offload entries must have storage capacity

Function:

1. User archives logs

Success Conditions:

- Archival process begins at 00:00:00AM (local time) on the 1st of each month. Logs older than 30 days are offloaded. Each offloaded log is consolidated, compressed, moved to another location and removed within 60 seconds.

Error Conditions:

- Archival process did not start at 00:00:00AM
- Archival process did not begin at 00:00:00AM local time
- Archival process began at 00:00:00AM local time but not on the first of the month
- Archival process began at 00:00:00AM local time on the first of the month but logs older than 30 days are not offloaded
- Archival process began at 00:00:00AM local time on the first of the month. Logs older than 30 days are offloaded but did not offload all log entries older than 30 days.
- Archival process began at 00:00:00AM local time on the first of the month. All log entries older than 30 days are offloaded but not consolidated.
- Archival process began at 00:00:00AM local time on the first of the month. All log entries older than 30 days are offloaded and consolidated. Log entries are not compressed
- Archival process began at 00:00:00AM local time on the first of the month. All log entries older than 30 days are offloaded, consolidated and compressed but not moved to another location.
- Archival process began at 00:00:00AM local time on the first of the month. All log entries older than 30 days are offloaded, consolidated and compressed. Log entries are moved to another location but not removed
- Archival process began at 00:00:00AM local time on the first of the month. All log entries older than 30 days are offloaded, consolidated and compressed. Log entries are moved to another location and not removed.

- Archival process began at 00:00:00AM local time on the first of the month. All log entries older than 30 days are offloaded, consolidated and compressed. Log entries are moved to another location and removed. The process takes longer than 60 seconds.

Non-Functional Requirements:

- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average

Use Case 20: User Experience

Purpose:

The purpose of this mechanism is to inform and protect user data from being used without consent

Scope:

The scope of this component covers all user related data within the system

Requirements:

- User will be notified of EULA per GDPR or California Consumer Privacy Act (CCPA) / California Privacy Rights Act (CPRA)
- Provide an opt-out of user data collection and data sale
- Provide an explanation of data collected
- Deletion of user and and its associated data

Use Case 21: Error Handling

Purpose:

The purpose of this mechanism is to prevent system failures from bringing the system offline

Scope:

The scope of this component covers the entire system

Requirements:

- System failures allowed to occur
    - Loss of web server internet access
    - Outage of cloud/host provider

Use Case 22: UI/UX

Purpose:

The purpose of this mechanism is to provide a interface for users to interact with the system

Scope:

The scope of this component covers all features that require user interaction within the system

Requirements

- Text will be in english and follow American culture settings
- All formats align with imperial units of measure
- All views will be intuitive so that they must not require assistance from another human to understand how to use the view
- All system messages will be displayed in english and follow american culture settings
- All system messages must appear within 5 seconds of operation resolution

Use Case 23: Documentation

Purpose:

The purpose of this mechanism is to provide artifacts gives a description of a system in detail

Scope:

The scope of this component covers all features in the system

Requirements

- Low level design documents
- User manuals
- FAQs

Use Case 24: Tree History

Purpose:

The purpose of this mechanism is to keep a history of all changes a user makes to a tree, so they and other users can see past changes

Scope:

The purpose of this use case is to keep track of all the operations a user does on a tree they create, as a form of version control.

Pre-conditions:

- User is an active and authenticated user
- User is a system administrator
- User has a stable internet connection
- User has navigated to to the usage analysis dashboard

Functional Requirements:

- All data is fetched from a persistent data store
- User must have created and made changes to an existing tree

Function:

1. User can see previous tree history
2. User can view specific previous trees

Success Conditions:

- Users are able to view previous tree history

Failure Conditions:

- User is unable to view tree history
- Tree history is unable to load within 5 seconds
- Incorrect view is loaded

Non-Functional Requirements:

- Function will be accessible 90% of the time
- Time to repair function will be at least 1 hour and within 24 hours on average
- Previous tree history must load within 5 seconds
- Program must be connected to database

Use Case 25: Progress Tracker

Purpose:

The purpose of this Use Case is to provide Users with a way to track their progress in learning and in comparison to how long others have taken to learn the information in a given node

Scope:

This Use Case deals with aggregate data from Users who have implemented Progress Tracking on a given node(s)

Preconditions:

- User is logged in
- Must have stable internet connection
- User is on own tree page

Functions

1. User selects nodes to be tracked by the Progress Tracker

Success Condition

1. User progress comparison is displayed once a node is completed

Failure Condition

- User is unable to see the tracker
- Tracker gathers the wrong data
- Tracker does not appear after five seconds after user selection of node(s).
- Tracker does not return

Non-functional Requirements

- The tracker must appear one to two seconds after the user selects one or more nodes.