Gratitude++ Requirements Specification

By: Joel Scott, Benjamin Seifried

TABLE OF CONTENTS

PROJECT NAME

The team has decided to create a web application journal that helps focus positive events and feelings towards a sense of gratitude for the user. Taking this into consideration as well as this is a computer science project, we have named the project Gratitude++.

TEAM MEMBER NAMES

See Figure 1 in the appendix for detailed description.

ABSTRACT

This project proposal outlines the development of a dynamic web application designed to empower users into cultivating a sense of gratitude through personalized journaling. The application will be built using a technology stack including React, TypeScript, Bootstrap and Firebase.

The application will allow a user to express themselves through journal entries that they can create. The titles and content of the journal entries will be utilized in creating an interactive “word-cloud” that will display keywords more prominently depending on their number of inclusions. By blending technology and personal expression, the application aims to promote mental well-being and foster an “attitude of gratitude" and provide users with an innovative and meaningful experience.

TOOLS AND TECHNOLOGIES

See Figure 1 in the appendix for detailed description.

REQUIREMENTS LIST

1. User Login Page:

1.1: (Onboarding) Users will be given the option to sign in through google services

1.1.1: On click, users will be redirected to another page to complete sign in procedure.

1.1.2: Users will be prompted to enter an email address and password to complete sign in

1.1.3: Authentication for sign in security will be handled through Firebase authentication services

1.1.4: Sign in flow for users will be handled manually and passed to Firebase for authentication

1.1.5: If sign in fails, users will be prompted to try again.

1.2: Upon successful sign-in a user will either be directed to

1.2.1. User’s home page (if account is present)

1.2.2. Registration Modal (No account Present)
(see Registration Modal)

2. Dialog boxes (General)

2.1: Every dialog box will have a submit button and a cancel button

2.1.1: The submit button will confirm the user's selection to complete the action and close the dialog box.

2.1.2: The cancel button will close the dialog box with no change.

3. User account registration page (Onboarding)

3.1.: Users will be required to create an account on their first visit to the site through the create new user page.

3.1.1.: Accounts must have a username.

3.1.2: Account names will be required to meet certain criteria.

3.1.2.1: Names cannot contain any spaces.

3.1.2.2: Names must contain at least five characters.

3.1.2.3: Names can be a mixture of numbers, letters, and special characters.

3.1.3: All account usernames must be unique.

3.1.4: Names will be confirmed by clicking the “confirm” button.

3.1.4.1: On click, checks for existing name in the database.

3.1.4.1.1: Sends a request to the database to check for a matching username, returns true or false.

3.1.4.1.2: If false, confirm the user's selection and send the username to the database under their specific account node.

3.1.4.1.3: If true, prompt the user to try again and repeat (3.1.2 - 3.1.4)

3.1.5.: Accounts must have an account avatar.

3.1.5.1: Users will be give a choice of four avatars to choose from for their profile picture.

3.1.5.2.: Users who do not change their avatar will be given the first avatar in the listing.

3.1.5.3: All avatar selections will be stored in the database along with the user’s username.

3.2.: Upon successful completion of the sign in the user will be directed to their personal home page.

4. Page Navigation

4.1: There will be a navigation menu to navigate the site

4.2: The user’s username and profile picture will be displayed on the navigation bar at the top left

4.3: There will be a tab called “My Homepage.”

4.3.1: This tab will take the user to their homepage.

4.3.1.1: This will display user information, including:

4.3.1.1.1: Word Cloud Component

4.3.1.1.1.1: See word cloud component for details

4.3.1.1.2: Create journal entry button

4.3.1.1.2.1: See journal entry creation page for details

4.3.1.1.3: User article search component

4.3.1.1.3.1: Will display the user’s five most recent post in order from most recent to least

4.3.1.1.3.2: Each post will contain the headline of the entry and the date it was created

4.4: There will be a tab called “Community Page.”

4.4.1: The community page will consist of two components

4.4.1.1: Community page search bar:

4.4.1.1.1: This component will allow users to search for entries in the community page by headline

4.4.1.1.2: Users will be given a list of all entries that match their search results.

4.4.1.1.3: If no entries match their search a message will appear on screen telling them that no matching entries could be found

4.4.2.1: Community article search:

4.4.2.1.1: This component will display posts by all users marked public.

4.4.2.1.1.1: Post will be able to be flagged by any user, removing them from the community page (no other administration needed)

4.4.2.1.1.1.1: Post removed from the community page will stay on the user’s account.

4.4.2.1.1.1.2: A post may be flagged up to twenty times before it is removed from the community page.

4.4.2.1.1.1.3: Each user may only flag a post one time, and users will not see how many times a particular post has been flagged.

4.4.2.1.1: Users will be able to click on any displayed articles to view the content.

4.4.2.1.1.1: Clicking on an article will navigate the user to a new page to view the entry

4.4.2.1.1.2: The entry will display the headline, the content, and the username of the original creator

4.4.2.1.1.3: Information on this page will be read only

4.4.2: Posts on this page will not be editable.

4.5: There will be a tab called “Resources.”

4.5.1: This tab will take the user to the resource page.

4.5.1.1: There will be a label Resources.

4.5.1.2 There will be a list of hyperlinks to resources related to mental health.

4.6: There will be a tab called “Journal History.”

4.6.1: This tab will take the user to their Journal History page.

4.6.1.1: see Journal History page

4.7: There will be a tab called “User Settings.”

4.7.1: This button will take users to their personal settings page.

4.7.2: see User Setting Component

4.8: There will be a button called “Logout.”

4.8.1: This button will log the user out of their account.

4.8.1.1: Users will be prompted if they wish to log out or not.

4.8.1.1.1: If the user chooses to confirm they will be logged out and returned to the sites homepage

4.8.1.1.2: If the user chooses to cancel the dialog box will close and the user will remain logged in

5. Journal Entry Creation page

5.1: There will be a prompt at the top giving the user an idea of what to write about

5.2: There will be a label that prompts a user to enter a headline for the journal entry.

5.3: There will be a text box that will take a user’s headline entry.

5.3.1: Each headline will be limited to 100 characters as a max length

5.4: There will be a text box asking for the user’s content for the journal entry.

5.4.1: Users will only be able to enter text into to create the journal entry.

5.4.2: Each journal entry will be limited to 1000 characters as a max length

5.5: There will be a submit button that allows a user to submit the current entry and headline.

5.5.1: When submit is pressed a dialogue box will pop up to confirm or cancel the post.

5.5.2: If confirmed the entry will be sent to the database.

5.5.2.1: The journal entry will be stored in the user’s section of the database under their specific user’s database location with the headline, date and time, content, and visibility.

5.5.3: If canceled the dialogue box will close.

5.6: If the user navigates away from the page the entry will not be saved and the boxes will be blank when the user returns to the journal entry creation

5.7: There will be a public or private toggle button for entries.

5.7.1: Private will be the default position.

5.7.2: If private is selected posted submissions will not be visible on the community page.

5.7.3: If public is selected then posted submissions will be visible on the community page.

5.7.3.1: Posts set as public will be sent to the “community page” section of the database with the headline, date and time, and the username of the creator.

6. Word Cloud Component

6.1 There will be a Word Cloud component that visually represents the key words taken from all the user's headlines and content that have been posted.

6.1.1 The word cloud will be of varying sizes and colors in relation to the number. of occurrences of key words.

6.1.2: The headlines and content will be acquired by making a call to the database and retrieving all data from the user's journal entries and then compiling them as a string.

6.1.3: The word cloud will utilize dynamic visual tools such as on-hover distinctions

6.1.4: The keywords should individually be clickable.

6.1.4.1: Clicking a keyword will navigate to a new page
that will display all posts made by that user that contain that keyword.

6.1.4.2: User's will be shown a listing of posts in order of most recent to least that contain that keyword in the headline or content.

6.1.4.3: All posts shown will contain the headline of the entry and the date it was created.

6.1.4.4: If the user clicks on any of the posts they will be taken to a new page to view the contents of the entry

6.1.4.5: There will be a button on the screen to go back to view all information on the previous page

7. Journal History Page

7.1 Will display a list of all a current user's posted articles listed from most recent.

7.1.1 Will have label with page name "Journal History."

7.1.2 Will have search bar for keywords in headlines.

7.1.2.1: The search bar will only search for items that start with specific words or letters

7.1.3 Will be listed by headlines with Date Stamp included.

7.1.4 Will be a web page accessible from navigation.

7.1.5: Using the search bar will look through all available entries in the database.

7.1.5.1: Will show all articles that are found matching the headline.

7.1.5.2: Will tell the user that no entries could be found if no headlines match the search

7.2 Clicking a headline will navigate to the journal entry viewing page

7.2.1: See journal entry viewing page for details

8. Admin Component

8.1 Will display a list of all User Accounts currently active.

8.1.1 Will be a component at the bottom of the user settings page

8.1.2 all active users will be listed by username.

8.1.3 All admin will have the ability to delete user accounts by clicking on the username

8.1.3.1 Clicking on the username will create pop-up modal asking to confirm the action

8.1.3.2 Confirming delete action will delete that user account.

8.1.3.2.1: Sends a request to the database to delete the user account from the database and all associated data

8.1.3.1.2: Deletes all associated entries and information from the community page of the site and the "community page" database section.

8.1.4: There will be search bar to search for a specific username in the listing

8.1.4.1: Shows all matching entries to the search

8.1.4.2: Shows a message if no users match the search

9. User Settings Component

9.1 Users will be able to do the following on the page:

9.1.1 Change avatar

9.1.1.1: Users will be able to change avatar when they navigate to the user settings page

9.1.1.2: User avatar update will be similar to user account creation

9.1.1.3: Users will not be able to change their username once they have set it.

9.1.2: Delete account.

9.1.3.1 Users will be given a prompt to confirm if they wish to delete their account before doing so.

9.1.3.2 If the user clicks confirm, their account will be deleted, along with their posts and username from the database.

9.1.3.3 If cancel is selected, the prompt will close.

10. Homepage Article Display page

10.1: Will display personal user entries

10.1.1: If the user is on the journal history page or their homepage it displays their personal entries

10.1.1.1: Entries will include the headline of the article, the content, and the username associated with the account.

10.1.1.2: On the journal entry viewing page users will be able to delete journal entries

10.1.1.3: Clicking on the delete button will create a dialog box asking to confirm the deletion of the entry.

10.1.1.3.1: If the user selects confirm, that specific entry will be deleted from the database and the user will be returned to their homepage

10.1.1.3.2: If the entry was one of the five most recent from the user it will disappear from their homepage listing

10.1.1.3.3: The entry information will be deleted from the user's word cloud

10.1.1.3.4: If the entry was marked as public it will be removed from the community page section as well

10.1.1.3.5: If the user clicks cancel the dialog box will close and no changes will occur.

11. Mobile Site Component

11.1 Site will be friendly for mobile users.

11.1.1 Website will configure containers and information to fit a mobile device.

11.1.2 All pages on mobile components will be easily navigated.

11.1.3 Journal entries will allow for images to be uploaded from mobile devices.

APPENDIX

Figure 1: Developers/Architects

Name	Role
Joel Scott	Developer, Architect

Benjamin Seifried	Developer, Architect

Figure 2: Tech-Stack

Name	Purpose
Node.js	Framework/library to implement server
React	Framework/library to assist with front-end development
Firebase	Database to store information
Replit	Server Host
GitHub	Version Control

Figure 3: Tentative Schedule

Week 1: August 28th – September 6th	**All Members:** Project proposal, assessment of strengths and weaknesses, determination of technology and software, information gathering
Week 2: September 7th – September 13th	**All Members:** Project requirements, refinement of project ideas, setup of technology for project development
Week 3: September 14th – September 20th	**Benjamin:** Focus on front end development, Familiarizing with ReactJS, TypeScript, and JavaScript **Joel:** Focus on familiarizing with back-end technology NodeJS and Firebase
Week 4: September 21st – September 27th	**Benjamin:** Begins to setup overall base of website, sets up front-end environment, and continues to work with ReactJS, TypeScript, and JavaScript **Joel:** Begins to setup the database component, setting up back-end environment, and continues working with NodeJS and Firebase
Week 5: September 28th – October 4th	**Benjamin:** Continue working on front-end environment, uses HTML (Hyper Text Markup Language) and CSS (Cascading Style Sheets) to create basic design of website. **Joel:**

	Connects database to front-end environment, assists with front-end development, sets up connection for server component of the website
Week 6: October 5th – October 11th	**Benjamin:** Continues working on front-end environment, works on creating UI and creating user journal entry page. **Joel:** Assists with front-end, creates data structures to sort and manage user profiles within the server, tests, and debugs current back-end code to ensure quality
Week 7: October 12th – October 18th	**Benjamin:** Begins with word cloud implementation, develops subject categorization. **Joel:** Works on back-end development, works on user-authentication through Firebase
Week 8: October 19th – October 25th	**Benjamin:** Continues front-end development, works with team to integrate front-end and back-end together. **Joel:** Continues back-end development, works with team to integrate front-end and back-end together
Week 9: October 26th – November 1st	**All Members:** Continue integration between front-end and back-end, begin working on project poster
Week 10: November 2nd – November 8th	**All Members:** Continue working on project poster. **Benjamin:** Works on developing navigation bar, creating menu items to navigate around site. **Joel:** Continues working on server component, tests and verifies that site has valid connection to server
Week 11: November 9th – November 15th	**All Members:** Continue working on project poster. **Benjamin:** Integrates front end and back end, works on UI and ensuring that navigation bar is functioning properly. **Joel:**

	Works with database, ensures that information is being stored and brought to the website
Week 12: November 16th – November 22nd	**All Members:** Finish project poster **Benjamin:** Continue testing features of the website ensure that all features function as intended. **Joel:** Test back-end development, ensure that server and database components are working properly
Week 13: November 23rd – November 29th	**All Members:** Continue testing of project, ensure that integration between the components is working
Week 14: November 30th – December 6th	**All Members:** Continue testing and debugging, begin preparing for final interview and project presentation.
Week 15: December 6th – December 12th	**All Members:** Project finalization, continue working with group to ensure that they are prepared for the final interview and presentation