|  |
| --- |
| Technological University Dublin |
| Journ.ie |
| *“Memory is the diary we all carry about with us.” – Oscar Wilde* |

|  |
| --- |
| C20305696 Lovely Fernandez  7-7-2024 |

Table of Contents

[Abstract 2](#_Toc175610324)

[About Journ.ie 2](#_Toc175610325)

[Product Vision 2](#_Toc175610326)

[Focus Group 2](#_Toc175610327)

[Product Management 3](#_Toc175610328)

[Identity 3](#_Toc175610329)

[Palette 3](#_Toc175610330)

[Requirements 3](#_Toc175610331)

[Project Management 4](#_Toc175610332)

[Milestones 4](#_Toc175610333)

[Task Distribution 5](#_Toc175610334)

[Web Application Design 5](#_Toc175610335)

[Use Case Diagrams 5](#_Toc175610336)

[Class Analysis 7](#_Toc175610337)

[Class and Sequence Diagram 8](#_Toc175610338)

[Database Diagram (ERD) 9](#_Toc175610339)

[Summary of Relationships 9](#_Toc175610340)

[Sitemap 9](#_Toc175610341)

[GUI Prototype 10](#_Toc175610342)

[Software Quality Testing 10](#_Toc175610343)

[User Testing Instructions 10](#_Toc175610344)

[Results from User Testing 11](#_Toc175610345)

[Testing Conclusion 12](#_Toc175610346)

[Key Findings 12](#_Toc175610347)

[Suggested Enhancements 12](#_Toc175610348)

[Programming Technologies and Languages 13](#_Toc175610349)

[Set Up Instructions 13](#_Toc175610350)

[Installation 13](#_Toc175610351)

[Open Project Locally 14](#_Toc175610352)

[Deployment 14](#_Toc175610353)

[Demonstration 15](#_Toc175610354)

[References 15](#_Toc175610355)

# Abstract

This report serves to document the process of ‘*Journ.ie*’ from concept to the final product, highlighting critical aspects of each development phase. It includes a summary of product management, web application design principles, and software quality testing methodologies applied, as well as seamless deployment strategies and the utilisation of various technologies and programming languages. Additionally, a demonstration video showcasing the functionality and features of *Journ.ie* is provided via a YouTube link.

*Please find further information about the web application's project plan, code and documentation on GitHub at the following links:*

1. *Code and Development:* <https://github.com/Journ-ie/Application>
2. *Documentation:* <https://github.com/Journ-ie/Documentation>
3. *Project Planning:* <https://github.com/orgs/Journ-ie/projects/1/views/1>

# About Journ.ie

## Product Vision

Journ.ie is a unique web application designed as a journaling platform where individuals can document moments they wish to preserve and treasure—a digital archive for capturing significant milestones and memories. This platform supports the art of personal posting and offers tools for organising life's moments meaningfully.

Unlike traditional social media platforms, Journ.ie prioritises providing a safe environment for individuals to focus on their achievements, growth, and reflections without the pressure of public exposure.

With Journ.ie, users can track their personal growth over time, revisit memories, and reflect on their experiences in a meaningful way. Journ.ie empowers individuals to connect with themselves at a deeper level—supporting users through personal healing, fostering personal growth, and actively pursuing their life goals.

## Focus Group

The primary users are adults between the ages of 20 and 35, who are navigating the early stages of their lives and seeking self-growth and development. They have a strong desire for purpose, self-improvement, and a significant emphasis on traveling.

*“Nearly 3 in 5 young adults (58%) reported that they lacked “meaning or purpose” in their lives in the previous month. Half of young adults reported that their mental health was negatively influenced by “not knowing what to do with my life… Forty-five percent (45%) of young adults reported that a general "sense that things are falling apart” was impairing their mental health.” -* ( On Edge: Understanding and Preventing Young Adults’ Mental Health Challenges, 2023)

# Product Management

## Identity

|  |  |
| --- | --- |
|  | "*Journ.ie*" originates from the concept of journaling one's thoughts throughout their journey. The "ie" is a play on words, referencing Ireland, where the website was developed from.  Additionally, Celtic designs are incorporated to honour the country's rich history and cultural heritage. |

## Palette

To create a calming and serene colour palette for Journ.ie, supporting its mission of inner reflection and journaling, soft pastel shades and earthy tones have been chosen. This palette evokes a sense of calm, serenity, and positivity, enhancing the therapeutic experience of journaling and emotional exploration.

A pink and white background with black text

Description automatically generated

*Figure 01. Colour Palette (Suggested by ChatGPT)*

## Requirements

|  |  |
| --- | --- |
| **Functional**  **\*Additional Features to consider if time permits** | |
| User Authentication and Authorisation | Enable users to create personalised accounts to securely store and access their journal across devices. |
| Digital Journaling Tools | Create and manage journal entries, incorporating text, photos, videos, and various multimedia elements. |
| Mood Tracking\* | Add mood or emotional state on journal entries. |
| Timeline and Progress Tracker\* | Track personal growth and achievements over time using timeline and goal-setting tools. |
| Search and Organisation Tools | Navigate and manage entries efficiently with tags, categories, and filters. |
| **Non-Functional** | |
| Usability | * Aim to create a user-friendly interface – UI is a high importance to improve UX. * Accessible to users with different technology skills   + Technical Skills Levels   + Accessibility Types |
| Compatibility | * The website should be dynamic and work on major screen types, with a primary focus on mobile and web screens. |
| Security | * User data should be secured and sensitive data to be encrypted where necessary. |

# Project Management

In my project planning and management approach, I utilise Agile methodologies, conducting bi-weekly Friday reviews and employing SCRUM practices. This strategy ensures continuous improvement, effective task prioritisation, and a steady workflow.

## Milestones

A close-up of a screen

Description automatically generated

*Figure 02. Milestones (Miro)*

|  |  |
| --- | --- |
| Project and Product Planning | 08 July – 15 July |
| Web Application Design | 16 July – 20 July |
| Product Development | 21 July – 03 August |
| Testing | 04 August – 05 August |
| Product Development Review | 06 August – 09 August |
| Testing 2 | 10 August |
| Product Development Final Review | 11 August |
| Deployment | 12 August – 13 August |
| Documentation, Code & Video Submission | 14 August – 18 August |
| Deadline | 31 August |

## Task Distribution

GitHub Issue Boards offer a flexible platform for managing development workflows and task distribution through a visual interface that aligns with agile project management methodologies.

See *Figure 03* for an overview of how I effectively distributed and tracked tasks:

|  |  |
| --- | --- |
| **To Do (Backlog)**  All tasks and stories for the upcoming iteration are added to this column. It acts as the planning area where tasks are sorted by priority. | **In Progress**  Tasks are moved to this column when they are being worked on. This helps keep track of active tasks and ensures focus on the goal of completing the iteration. |
| **Review & Testing** Testing whilst developing is crucial as it catches early errors, reducing the burden during the testing phase. This stage ensures the work meets the desired standards and functions as intended. | **Done** Completed tasks are moved to this column. ‘Done’ provides a clear view of the project’s progress and achievements, helping maintain motivation and momentum. |



*Figure 03. Issue Board (GitHub Project)*

# Web Application Design

## Use Case Diagrams

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A diagram of a system  Description automatically generated  *Figure 04. Use Case 1 Journ.ie System*  *(StarUML)* | | | |  |  | | --- | --- | | **Use Case 1: Using Journ.ie System** | | | **Goal** | The writer can login and journal | | **Preconditions** | Writer needs a verified account | | **Postconditions (Success)** | Writer can journal and log their day | | **Postconditions**  **(Failed)** | Writer is unable to access their account, therefore, unable to access Journ.ie | | **Actors** | Writer – the main user of the website | | **Trigger** | Writer wants to access their journal | | **Description** | Writer logins or register/create an account and journals using Journ.ie features | | | |
| **Main Flow (MF)** | | | | | |
| **Step** | **Action** | | | | **Alternate** |
| 1.1 | System prompts for user access details (username/email, password) | | | |  |
| 1.2 | Writer inputs their user details | | | | EF.1.2 |
| 1.3 | System verifies user details are correct | | | | EF.1.3 |
| 1.4 | System gives access to the user and directs the page to their journal | | | |  |
| 1.5 | Writer accesses their journal and logs their day | | | | End of MF |
| **Exceptions or Error Flows** | | | | | |
| **Exception/Error Flow 1.2:** MF.1.2 Writer does not have an account | | | | | |
| 1.1 | Writer chooses register option and creates an account | | | | MF.1.1 |
| **Exception/Error Flow 1.3:** MF.1.3 Writer input incorrect user details | | | | | |
| 1.1 | System prompts user of the error – *‘Incorrect email/username or password’* | | | |  |
| 1.2 | Writer re-enters their user details | | | | MF.1.3 |
| **Non-Functional Requirement** | | | | **Priority:** High | |
| **Product Security** | | Adhere to GDPR standards on processing data | | | |
| **Regulation and Law** | | Users should meet the standard age to access the website. | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Figure 05. Use Case 2 Location Tagging*  *(StarUML)* | | | | |  |  | | --- | --- | | **Use Case 2: Adding a Location Tag to a Log** | | | **Goal** | Location Tag is added to the Journal Log | | **Preconditions** | Location Permissions is enabled | | **Postconditions (Success)** | Writer can add a Location Tag | | **Postconditions**  **(Failed)** | Writer is unable to add a Location Tag | | **Actors** | Writer, Browser | | **Trigger** | Writer adds location to the journal log | | **Description** | Writer adds their location on the journal log | | | | |
| **Main Flow (MF)** | | | | | | | |
| **Step** | **Action** | | | | | | **Alternate** |
| 2.1 | Writer adds a log on the journal and adds their location tag | | | | | |  |
| 2.2 | System verifies if Location Permissions are allowed | | | | | | EF.2.2 |
| 2.3 | System verifies the address is valid | | | | | | EF.2.3 |
| 2.4 | System adds the location tag on the log | | | | | | End of MF |
| **Exceptions or Error Flows** | | | | | | | |
| **Exception/Error Flow 2.2:** MF.2.2 Location Permissions are disabled | | | | | | | |
| 2.1 | System prompts user to enable Location Permissions | | | | | | AL.2.1 |
| 2.2 | Writer enables Location Permissions | | | | | | MF.2.2 |
| **Exception/Error Flow 2.3:** MF.2.3 Location is invalid | | | | | | | |
| 2.1 | System prompts user of the error – *‘Location is invalid’* | | | | | |  |
| 2.2 | Writer updates the address or adds a different address | | | | | | MF.2.3 |
| **Alternative Flows** | | | | | | | |
| **Alternative Flow 2.1:** EF.2.2.1 User does not enable Location Permission | | | | | | | |
| 2.1 | | System prompts user of the error – ‘*Unable to add location’* and returns to log draft | | | | End of MF | |
| **Non-Functional Requirement** | | | | | **Priority:** High | | |
| **Product Reliability** | | | Location Tag is accurate | | | | |
| **Product Safety** | | | Location Permissions enabled where only necessary | | | | |

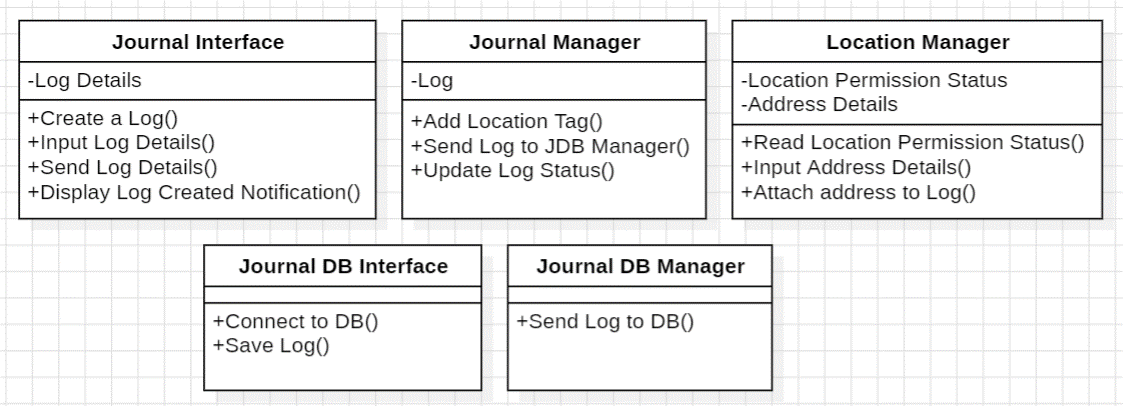
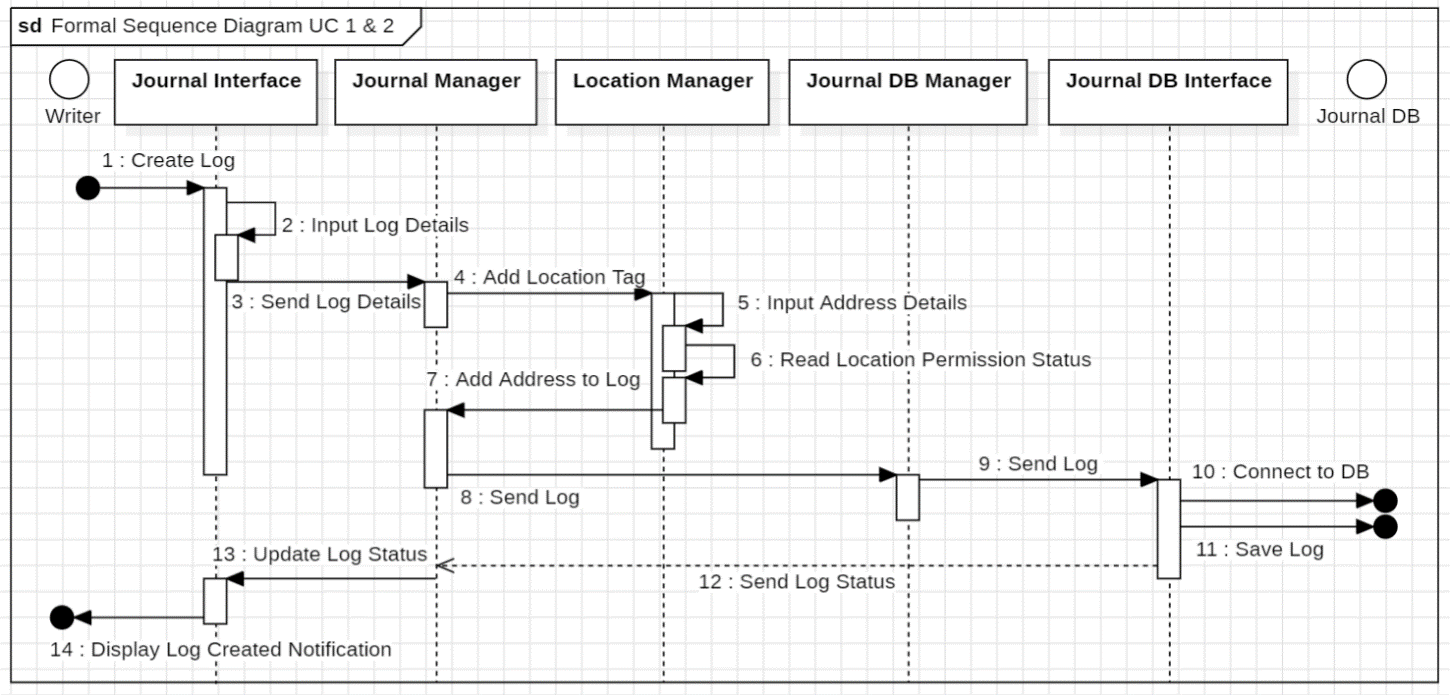
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Figure 06. Use Case 3 Journal System*  *(StarUML)* | | | |  |  | | --- | --- | | **Use Case 3:** Journal System | | | **Goal** | Writer adds a log to their journal | | **Preconditions** | Writer has a verified account | | **Postconditions (Success)** | Log is added into the Journal/Journal database (DB) | | **Postconditions**  **(Failed)** | Log is not added into the Journal/Journal DB | | **Actors** | Writer, Journal DB | | **Trigger** | Writer saves a Log | | **Description** | Writer drafts a log into the journal and saves it into the DB by submitting it through the system | | | |
| **Main Flow (MF)** | | | | | |
| **Step** | **Action** | | | | **Alternate** |
| 3.1 | *Use Case 1 runs here…* | | | |  |
| 3.2 | Writer adds a log into the journal and adds location tag (*Use Case 2 runs here…)* | | | |  |
| 3.3 | Writer saves log | | | | EF.3.3 |
| 3.4 | System checks all mandatory fields and requirements have been completed | | | | EF.3.4 |
| 3.5 | System adds the log into the Journal database | | | |  |
| 3.6 | System closes log and from the database display into journal page | | | | End of MF |
| **Exceptions or Error Flows** | | | | | |
| **Exception/Error Flow 3.3:** MF.3.3 Writer does not save the log | | | | | |
| 3.1 | System disregards user log details and returns to journal page | | | | End of MF |
| **Exception/Error Flow 3.4:** MF.3.4 Mandatory fields are not filled | | | | | |
| 3.1 | System prompts user the fields that are not filled – ‘*Input X’ or underline red.* | | | |  |
| 3.2 | Writer completes the empty fields | | | | MF.3.3 |
| **Non-Functional Requirement** | | | | **Priority:** High | |
| **Product Performance** | | Notification to update user when the log is saved and uploaded | | | |
| **Product Security** | | Logs can only be added but should not be able to access other functions of the database | | | |

## Class Analysis

|  |  |
| --- | --- |
| A diagram of a diagram  Description automatically generated | *Figure 07. Class Analysis Location Tag.*  *(StarUML)* |
|  | *Figure 08. Class Analysis Journal*  *(StarUML)* |

## A screen shot of a diagram Description automatically generatedClass and Sequence Diagram

*Figure 09. Informal Sequence Diagram UC1-2 (StarUML)*

 *Figure 10. Class Diagram (StarUML)*

*Figure 11. Formal Sequence Diagram UC1-2 (StarUML)*

## Database Diagram (ERD)

|  |  |
| --- | --- |
| *Figure 12. Logical ERD (Oracle DB)* | Summary of Relationships  1. **User Profile and User Details**: Each user profile has corresponding user details (one-to-one) 2. **Log and Log Tag**: Each log can have multiple tags (many-to-many) 3. **Log and Log Media**: Each log can be associated with multiple media files (many-to-many) 4. **Log and Log Location**: Each log can only be associated with one location (one-to-one) |

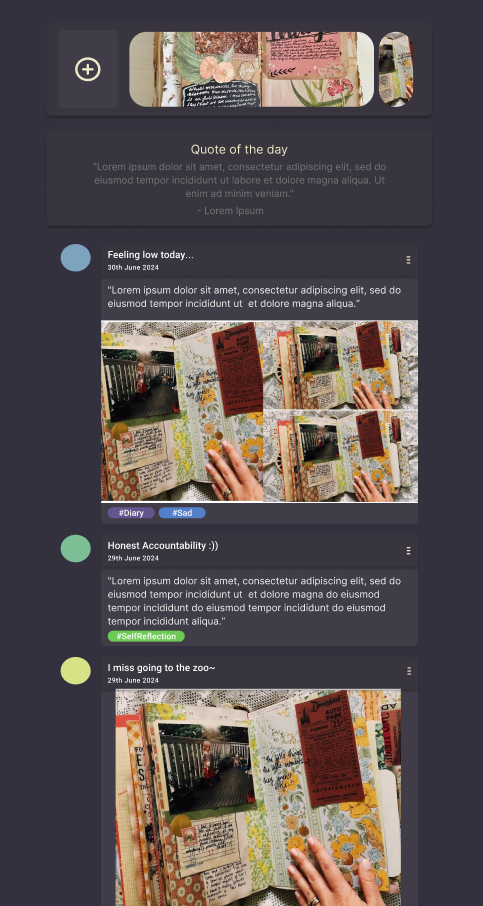
## Sitemap

A screenshot of a computer

Description automatically generatedIn *Figure 13*, the sitemap is provided, highlighting the main pages, which include the *Home Landing Page, Journal Page, Settings Page, Help Page, and Log In/Sign Up Page.*

*Figure 13. Sitemap (Flowmapp)*

## GUI Prototype

In *Figure 15*, a visual prototype is created for the *Home Page, Journal Page, and Entry Log Page,* respectively. More at *Documentation (2), GUI Prototype Folder.*

*Figure 14. GUI Prototype (Figma)*

# Software Quality Testing

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| User Testing Instructions | | | | | | | |
|  | | | | | | | |
| **Scenario 01** | | Creating an Account, Update Information, Log Out | | | | | |
| Register an account, go to settings to update your account information and log out of the account. | | | | | | | |
| 1 | Create an account | | | | | |  |
| 2 | Enter your details for your profile | | | | | |  |
| 3 | Navigate to the settings page | | | | | |  |
| 4 | Edit your profile information and save the changes | | | | | |  |
| 5 | Log out of the account | | | | | |  |
|  | | | | | | | |
| **Scenario 02** | | Sign In, Create 2 Posts, Edit 1 Post, Delete 1 Post | | | | | |
| Sign back into the account and create two journal posts. With one of the posts, edit the details of the log and save the changes. Lastly, delete of the two posts available. | | | | | | | |
| 1 | Sign in with the account you just registered | | | | | |  |
| 2 | Create a Post | | | | | |  |
| 3 | Add details to your post, attach videos or images and choose an appropriate label(s) | | | | | |  |
| 4 | Submit and create another post | | | | | |  |
| 5 | Edit one of the post, update the title and save the changes | | | | | |  |
| 6 | Delete one of the posts | | | | | |  |
|  | | | | | | | |
| **Scenario 03** | | Deactivate Account | | | | | |
| Go to settings and deactivate your account | | | | | | | |
| 1 | Navigate to the settings page | | | | | |  |
| 2 | Click Delete Account | | | | | |  |
| 3 | Re-confirm your action | | | | | |  |
| 4 | Attempt to sign back in with the same account information | | | | | |  |
|  | | | | | | | |
| **Interview & Questionnaire** | | | | | | | |
| How do you rate your experience with Journ.ie? | | | | | | | |
| (1) Bad | | | (2) Not too bad | (3) Okay | (4) Good | (5) Great | |
| How easy was it to register an account? | | | | | | | |
| (1) Very Easy | | | (2) Easy | (3) Okay | (4) Hard | (5) Difficult | |
| How easy was it to add a journal log? | | | | | | | |
| (1) Very Easy | | | (2) Easy | (3) Okay | (4) Hard | (5) Difficult | |
| Do you have any further feedback? | | | | | | | |
|  | | | | | | | |
|  | | | | | | | |

## Results from User Testing

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| User Tester Information | | | 1 | A 27-year-old male college student with a computer science background, actively engages in coding projects. | | | |
| Technical Level | | | High | | | | |
| Rate Experience | 4 | Difficulty of Register Process | | | 1 | Difficulty of Journal Log | 2 |
| Feedback | | | Light mode requested. | | | | |
| Observation Notes | | | * Uses the FR translation of the website (Success) * Password restriction (1 cap/non-cap, special letter with x length.) * Naturally gravitated to the submit when wanting to add a media. Suggestion for UI to swap the buttons? | | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| User Tester Information | | | 2 | A 38-year-old female user with no technical background, but she is familiar with common media platforms and regularly engages with digital content for communication, entertainment, and staying informed. | | | |
| Technical Level | | | Mid | | | | |
| Rate Experience | 3 | Difficulty of Register Process | | | 2 | Difficulty of Journal Log | 2 |
| Feedback | | | Would like to see timestamps on posts. | | | | |
| Observation Notes | | | * Looking for register button as Sign is only visible * Put the date of today - age restriction to be applied. (At least 13<90(?) years of age) | | | | |

## Testing Conclusion

### Key Findings

**Registration Process**

* The registration process was straightforward, with the difficulty rated between "1" (Very easy) and "2" (Easy).
* The visibility of the registration button was a minor issue, with some users having trouble to find it as Sign In is only visible.

**Journal Logging Experience**

* The difficulty of logging journal entries was rated as "2" (Easy), indicating a generally user-friendly process but room for improvement.
* One of the users subconsciously gravitated to the submit button to add medias which suggests UI adjustments to improve UX.

**Feedback on User Interface and Features**

* Light Mode Requested
* Timestamps on Posts

### Suggested Enhancements

* Consider swapping buttons between media submission and log submission to better align with user expectations.
* Enhance the visibility or clarity of the registration button to avoid confusion with the "Sign In" option. Instead of “Sign In”, update this to “Sign In/Register”.
* Implement a light mode option to accommodate varying user preferences for visual themes.
* Incorporate timestamps on journal posts to provide clearer chronological information.
* Apply age restrictions in adherence to the law or a common basic application of 13<80+ years of age rule.
* Clarify age restrictions during the registration process to ensure that users understand the eligibility requirements.

# Programming Technologies and Languages

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project/Product Management | GitHub Logo and symbol, meaning, history, PNG, brand |  |  |  |
| Front-End | Html Logo Vector Art, Icons, and Graphics for Free Download | Free High-Quality CSS Logo for Creative Design | JavaScript PNG, Transparent JS Logo Free Download - Free Transparent PNG  Logos |  |
| Back-End | A logo of a coffee cup  Description automatically generated | Firebase Brand Guidelines |  |  |
| Web Development | A black text on a white background  Description automatically generated | A black square with white letter m in it  Description automatically generated | UML Tools - Javatpoint |  |
| Deployment | Firebase Hosting - Javatpoint |  |  |  |

# Set Up Instructions

This is also available in the *README.md* file of Application Repository: <https://github.com/Journ-ie/Application>

## Installation

1. Requires having nodeJS(npm) and git to set up the project locally.
2. Install Firebase SDK (Run Terminal as Admin):

*npm install firebase*

*npm install -g firebase-tools*

*Set-ExecutionPolicy RemoteSigned*

1. Go to the Application Repo of Journie and Clone the repository:

*git clone https://github.com/Journ-ie/Application.git*

*cd Application*

1. Log into Firebase Account:

*firebase login*

**Request to be added as a collaborator needed**

## Open Project Locally

1. Open terminal to run the website locally:

*firebase serve*

## Deployment

1. Check the following link to try the website online: <https://journie-d4114.web.app/>
2. For deployment:

firebase deploy

**Admin purposes only.**

# Demonstration

In this demonstration, I will guide through the various components of the application as depicted in the flowchart below in *Figure 15*. The process starts at the *Home* page, then proceeds to *Register an Account* where a new user can sign up. After registration, the *Journal* section is explored, where entries can be logged. The *Settings* section allows for managing the user’s profile. Finally, the *User Guide & Help* section offers assistance and guidance within the app for first-time users. This walkthrough will clarify how each part of the application connects and functions together.

A diagram of a register account

Description automatically generated

*Figure 15. Demonstration Flow (Miro)*

*Please find the demonstration video on YouTube via the following link:*

[*https://youtu.be/JF\_TQtBxIns*](https://youtu.be/JF_TQtBxIns)

# References

* Education, H. G. (2023). *On Edge: Understanding and Preventing Young Adults’ Mental Health Challenges*. Retrieved from Making Caring Common Project: https://mcc.gse.harvard.edu/reports/on-edge