# Scope and Goal

Last edited by **Khoa Nguyen** 1 year ago

# Features

The features of this product are defined in the `Requirement` issue board. They are labeled as `req:confirmed-feature` in the features list. Please refer to them for more detail regarding the requirement and implementation details.

For this project, each feature will be divided into smaller, more manageable chunks of tasks so that it would be easier to track the progress of each feature. We also use a collection of issue boards to monitor our tasks. Every implementation will starts in the `Requirement` board as `req:confirmed-feature`. Then, we go to the `Design` board to design the use flow and UI/UX of them. Next, the `Development` board will track the database, backend, and frontend development process of these tasks. Lastly, the `Quality Assurance` board will track the testing process of the finished implementations.

# Project roadmap

> This is our initial project roadmap, based on the 5 iterations of the project. This roadmap will be refined into more detail in our interaction 3, which will become our project plan.

## Project roadmap for TimeCrafter

### Iteration 1: Pre-study and project planning

> This iteration includes the basic learning for the project.

- **Tasks:**

  1. Set up project repository and version control.
  2. Define and document project working standards.
  3. Learn to use the features of GitLab

- **Timeline:** 8/1/2024 - 21/1/2024

- **Milestone:** Iteration 1: Pre-study and Project Planning

### Iteration 2: Repositories and Features, Risks

> This iteration starts defining the product's features and assessing risks. We will break the features down into more manageable tasks to track. In this iteration, we can start designing the basic use flow of the features.

- **Tasks:**

  1. Define the features of the product through requirement analysis.
  2. Risk assessment for the project and the product.
  3. Design the basic user flow of the features.

- **Timeline:** 22/1/2024 - 11/2/2024

- **Milestone:** Iteration 2: Backlog: repositories and features, risks

### Iteration 3: Project roadmap: effort estimation, prioritization, value estimation

> This iteration refines the project's roadmap with more information. We will start design the UI/UX of the product while implementing its design system. The database + backend services can be implemented in this iteration too.

- **Tasks:**

  1. Refine the project roadmap - include more detailed effort estimation and prioritization of sub-tasks.
  2. Start working on the UI/UX design of the products using the use flow defined previously of the features.
  3. Start implementing some of the prioritized features' database and backend. They can be done asynchronously with the design and implementation of the frontend UI.
  4. Start implementing tests for the backend.

- **Timeline:** 12/2/2024 - 10/3/2024

- **Milestone:** Iteration 3: Project roadmap: effort estimation, prioritization, value estimation

## Iteration 4: Reacting to changes

> Continue the development process by implementing the frontend and connect it to the previously made backend. Some requirements will change, so we will need to revise the use flow & UI/UX design.

- **Tasks:**

  1. Requirement analysis + feature update to reflect new changes. This will also update our effort estimation and will strictly follow our prioritization, i.e., not all changes will be guarateed.
  2. Implement the changes into our design (use flow + UI/UX).
  3. Implement the changes (if any) to our existing backend.
  4. Implement the frontend UI based on the modified designs.
  5. Start implementing tests for the frontend.

- **Timeline:** 11/3/2024 - 7/4/2024

- **Milestone:** Iteration 4: Reacting to changes

---

## Iteration 5: Wrap-up & Quality Assurance

> Wrap-up the development project. We will finalized our implementation and conduct testing as needed.

- **Tasks:**

  1. Conduct comprehensive testing.
  2. Address any identified bugs or issues.
  3. Refine user interface and experience based on feedback.

- **Timeline:** 8/4/2024 - 24/4/2024

- **Milestone:** Iteration 5: Wrap-up

---

## Notes:

- **Priority Levels:**
  - Critical: Tasks essential for core functionality. The product needs to have the features defined in these tasks, no matter what.
  - High: These are important tasks but can be prioritized slightly lower than critical.
  - Medium: Tasks that contribute to the completeness of the application.
  - Low: Additional features or refinements that enhance the user experience. We can leave these features out since they are only nice to have.

All of our features are labeled with `req:confirmed-feature`. They are features confirmed based on project requirements.

# Risk

Project risk

## General level risks

### 1. Scope Creep

- **Likelihood:** Medium
- **Impact:** High
- **Overall Risk:** Medium
- **Description:** There's a risk of the project scope expanding beyond the initial requirements, leading to delays and increased development efforts.

### 2. User Adoption

- **Likelihood:** Medium
- **Impact:** Medium
- **Overall Risk:** Medium
- **Description:** Users may resist or have difficulty adopting the virtual journal, impacting the success of the project.

### 3. Data Security

- **Likelihood:** Medium
- **Impact:** High
- **Overall Risk:** Medium
- **Description:** Storing personal and sensitive information in the virtual journal raises the risk of data breaches, potentially harming user trust.

## 4. Integration Challenges

- **Likelihood:** High
- **Impact:** Medium
- **Overall Risk:** High
- **Description:** Integrating the virtual journal with other tools or platforms may pose technical challenges, leading to delays and increased complexity.

## 5. User Experience (UX) Issues

- **Likelihood:** High
- **Impact:** Medium
- **Overall Risk:** High
- **Description:** Poor user experience, such as unintuitive design or navigation issues, may result in user dissatisfaction and a negative impact on adoption.

## 6. Technical Debt

- **Likelihood:** High
- **Impact:** Medium
- **Overall Risk:** High
- **Description:** Rushed development or cutting corners may lead to accumulating technical debt, impacting the maintainability and scalability of the system.

## 7. Unrealistic Timelines

- **Likelihood:** Medium
- **Impact:** High
- **Overall Risk:** Medium
- **Description:** Setting unrealistic project timelines may lead to increased stress, lower quality, and potential burnout among team members.

## 8. Requirements Changes

- **Likelihood:** Medium
- **Impact:** Medium
- **Overall Risk:** Medium
- **Description:** Changes in project requirements, especially late in the development process, can lead to rework and delays.

## 9. Dependency Risks

- **Likelihood:** Medium
- **Impact:** High
- **Overall Risk:** Medium
- **Description:** Dependencies on external libraries, services, or platforms may introduce risks if those dependencies experience issues or changes.

## 10. Communication Breakdown

- **Likelihood:** Medium
- **Impact:** Medium
- **Overall Risk:** Medium
- **Description:** Poor communication within the development team or with stakeholders may lead to misunderstandings and project delays.

# Feature-Related Risks

## 1. Scheduling Feature [#5](#5)

- **Risk: Incomplete Scheduling Interface**

  - **Likelihood:** Medium
  - **Impact:** High
  - **Overall Risk:** Medium
  - **Description:** There's a risk of not fully implementing the scheduling interface, leading to a feature that doesn't meet user expectations.

- **Risk: Unreliable Reminder Functionality**

  - **Likelihood:** Medium
  - **Impact:** High
  - **Overall Risk:** Medium
  - **Description:** The reminder functionality may be unreliable or inconsistent, impacting the user's ability to rely on the application for timely notifications.

## 2. Timeline/Calendar Feature [#16 (closed)](#)

- **Risk: Task Integration Issues**
  - **Likelihood:** High
  - **Impact:** Medium
  - **Overall Risk:** High
  - **Description:** Integrating tasks into the timeline/calendar may pose technical challenges, leading to display issues or incorrect scheduling.

## 3. Create Personal Notes Feature [#17 (closed)](#)

- **Risk: Data Loss or Corruption**
  - **Likelihood:** Medium
  - **Impact:** High
  - **Overall Risk:** Medium
  - **Description:** There's a risk of data loss or corruption within the personal notes feature, potentially affecting user data integrity.

## 4. Search/Sort Item(s) Feature [#18](#)

- **Risk: Inefficient Search Algorithm**
  - **Likelihood:** Medium
  - **Impact:** Medium
  - **Overall Risk:** Medium
  - **Description:** The search/sort feature may be inefficient, leading to slow performance or inadequate search results.

## 5. Item Categorization Feature [#19](#)

- **Risk: Category Assignment Errors**
  - **Likelihood:** Medium
  - **Impact:** Medium
  - **Overall Risk:** Medium
  - **Description:** Users might face issues with category assignment, leading to misclassification and difficulty in organizing items.

## 6. Edit/Delete Item(s) Feature [#20](#)

- **Risk: Data Integrity Issues**
  - **Likelihood:** Medium
  - **Impact:** High
  - **Overall Risk:** Medium
  - **Description:** Editing or deleting items may introduce data integrity issues, causing discrepancies in the application.

## 7. Create Task/Todo/Scheduled/Reminder Items Feature [#21 (closed)](#)

- **Risk: Task Overload**
  - **Likelihood:** Medium
  - **Impact:** Medium
  - **Overall Risk:** Medium
  - **Description:** Users might face challenges managing a large number of tasks, potentially leading to overwhelm and decreased usability.

## 8. Tracking Tasks Progress Feature [#3 (closed)](#)

- **Risk: Inaccurate Progress Tracking**
  - **Likelihood:** Medium
  - **Impact:** Medium
  - **Overall Risk:** Medium
  - **Description:** There's a risk of inaccurate progress tracking, impacting users' ability to assess and manage their tasks effectively.

# Comments