**Software Requirements Specification**

**for**

Life Habitat

**Version 2.3 approved**

**Prepared by**

**PM:** Brianna Gannett

**Designer:** RoisinRumsey

**Developers:** Sarah Turmel

Dean Hauser

Emily Scott

**Team Ethernet**

**10/7/2024**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

**2. Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

**3. External Interface Requirements 3**

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

**4. System Features 4**

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

**5. Other Nonfunctional Requirements 4**

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

**6. Other Requirements 5**

**Appendix A: Glossary 5**

**Appendix B: Analysis Models 5**

**Appendix C: To Be Determined List 6**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
| Sarah Turmel | 10/8/24 | Filling in features and descriptions of features | 1.1 |
| Emily Scott | 10/9/24 | Filled in Main Page | 1.2 |
| Roisin Rumsey | 10/9/24 | Brought ½ Functional Reqs. from reqs. Doc. | 1.3 |
| Dean Hauser | 10/9/24 | Brought ½ Functional Reqs. from reqs. Doc. | 1.3 |
| Sarah Turmel | 10/9/24 | Brought ½ NonFunc. Reqs. from reqs. Doc | 1.4 |
| Brianna Gannett | 10/9/24 | Brought ½ NonFunc. Reqs. from reqs. Doc | 1.4 |
| Brianna Gannett | 10/9/24 | Added UI Mockups To Section 3.1 | 1.5 |
| Emily Scott | 10/16/24 | Completed Section 2 | 2.0 |
| Dean Hauser | 10/16/24 | Completed Section 1 | 2.1 |
| Dean Hauser | 10/20/24 | Addressed feedback from Dr. Greg | 2.2 |
| Roisin Rumsey | 10/20/24 | Added new UI Mockups to Section 3.1 | 2.3 |
| Dean Hauser | 10/30/24 | Filled in sections 4.1.2, 4.2.2, 4.3.2, and 4.4.2 | 3.0 |
| Roisin Rumsey | 11/10/24 | Updated UI Mockups per UT feedback | 3.1 |
| Roisin Rumsey | 11/10/24 | Filled in sections 3.2, 3.3, and 3.4 | 3.2 |

# **Introduction**

## **Purpose**

*This document is a comprehensive requirements specification for the first release of Life Habitat. It contains detailed information for various systems, including task creation, avatar creation, and goal tracking.*

## **Document Conventions**

*This Document was created based on the IEEE template for System Requirements Specification Documents.*

*The following conventions in the document were used as follows:*

| **Convention** | **Description** |
| --- | --- |
| HTTP | Hypertext Transfer Protocol |
| FREQ | Functional Requirement |
| NFREQ | Non Functional Requirement |

## **Intended Audience and Reading Suggestions**

*This document, organized by system requirements, follows a logical order from User creations to Performance requirements. It is intended for the product manager, developers, designers, and users, providing a structured overview of the Life Habitat software.*

## **Product Scope**

*Life Habitat is goal-tracking Tamagotchi-like software that allows users to visualize their goal process by seeing their avatars' health. The more goals you do, the better your avatar's health will be. This software aims to motivate users to do their boring and tedious tasks with a game and reward system integrated into it.*

## **References**

*Life Habitat Team Github Link:* [*https://github.com/COS420-Fall24/Team-E*](https://github.com/COS420-Fall24/Team-E)

# **Overall Description**

## **Product Perspective**

*Life Habitat is an application that is designed to help people focus on completing daily tasks that might seem tedious and sometimes get left behind. Life Habitat is meant to take over for normal calendar apps and other similar applications by focusing on a more emotional connection to its features and always providing positive feedback, instead of relying on punishment to keep the users on track.*

## **Product Function**s

**Homepage/Login Page**

*· Welcomes User*

*· Has link to registration page*

*· Allows users to login*

**Registration Page**

*· Allows user registration*

**Main Page View**

*· Shows user avatar*

*· Shows avatar statistics*

*· Shows user notifications*

*· Shows short list of upcoming tasks*

*· Shows task history*

*· Links to all other pages*

**Task Page**

*· Lists tasks set for the day*

*· Shows upcoming tasks for the week/month*

*· Shows user progress*

*· Allows users to create tasks*

*· Allows users to edit tasks*

*· Allows users to delete tasks*

*· Allows users to mark tasks as completed*

**Avatar Page**

*· Allows users to edit avatar*

*· Allows users to buy accessories for the avatar with points*

## **User Classes and Characteristics**

*Life Habitat is created using the React app, CSS, HTML, and Typescript.*

## **Operating Environment**

*Life Habitat will operate on all major operating systems including Windows 10/11, Mac OS, Android, Linux, etc.*

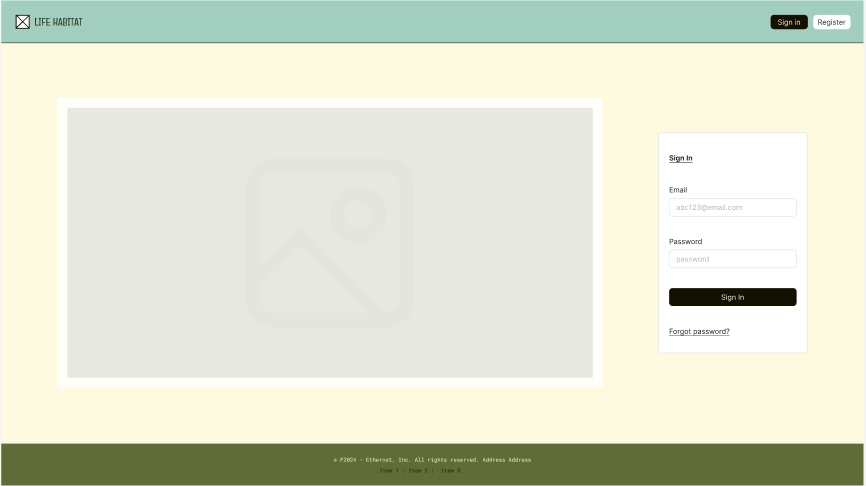
## **Design and Implementation Constraints**

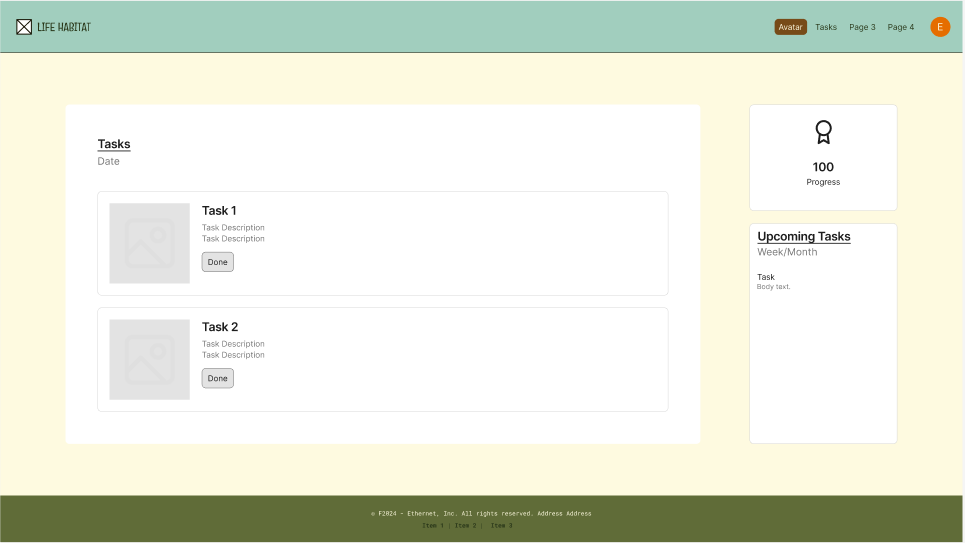
*Life Habitat will contain security systems to protect user data. This is because users will be giving their personal information for the full functionality of the app. For example, a task might list the date and time of a doctor's appointment, or when the user will be out of the house getting groceries. These could put the user at risk if it was released to third parties.*

# **External Interface Requirements**

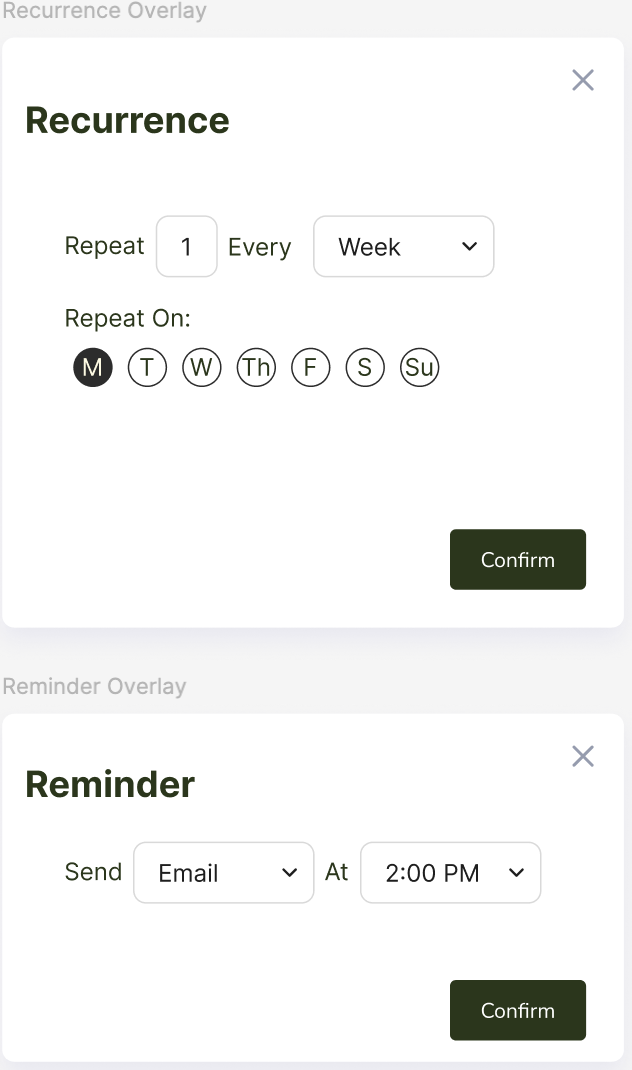
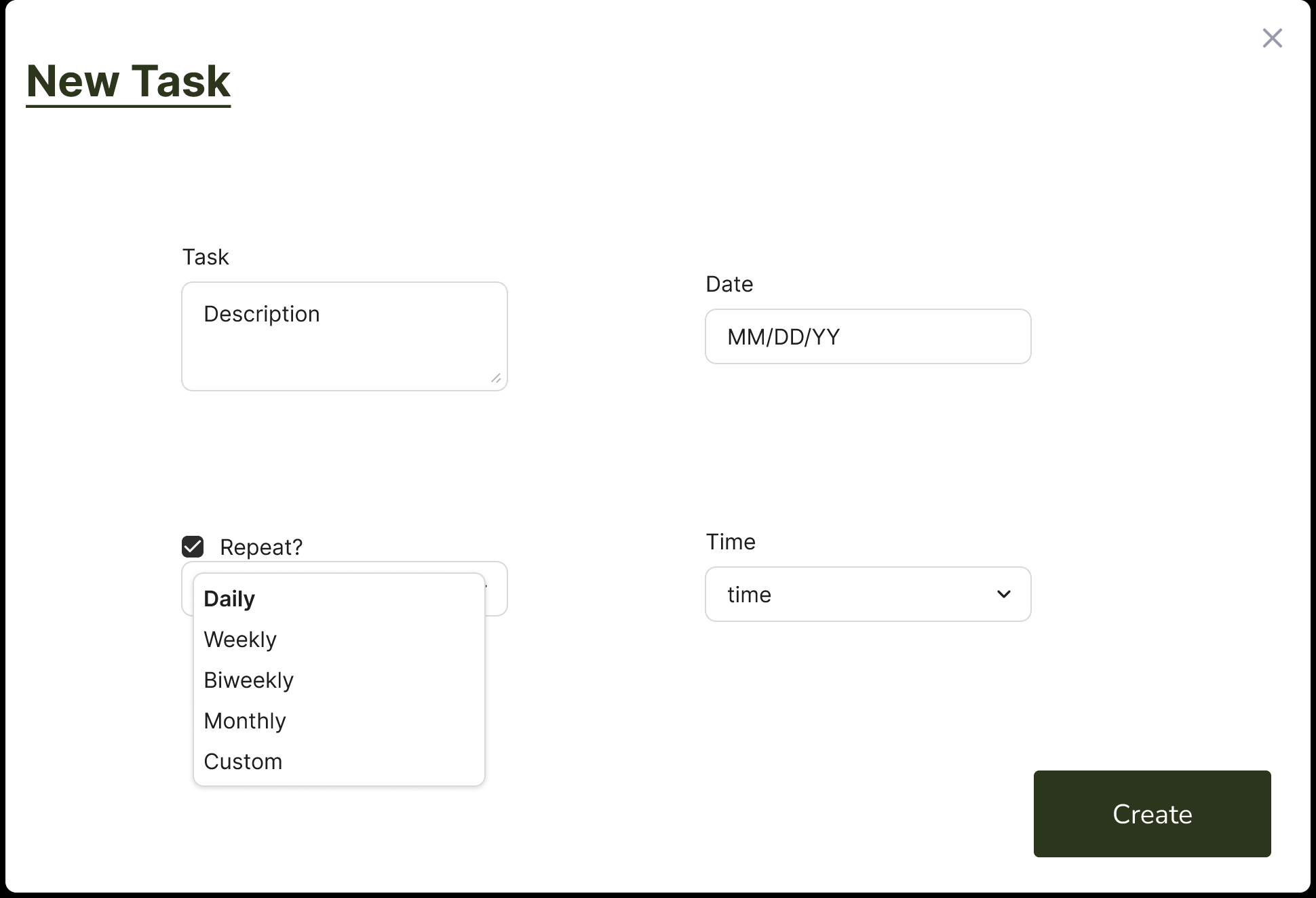
## **User Interfaces** [UI mockup link](https://www.figma.com/design/L6Otl6x2FM9YiG9tJF4XmX/Life-Habitat?node-id=6-1353&node-type=canvas)

This is an example of the Life Habitat Sign-In page. Clicking the Register button at the top of the page will allow the user to create an account if they do not already have one. Clicking Forgot Password will allow the user to send an email to the email address associated with their account to create a new password.

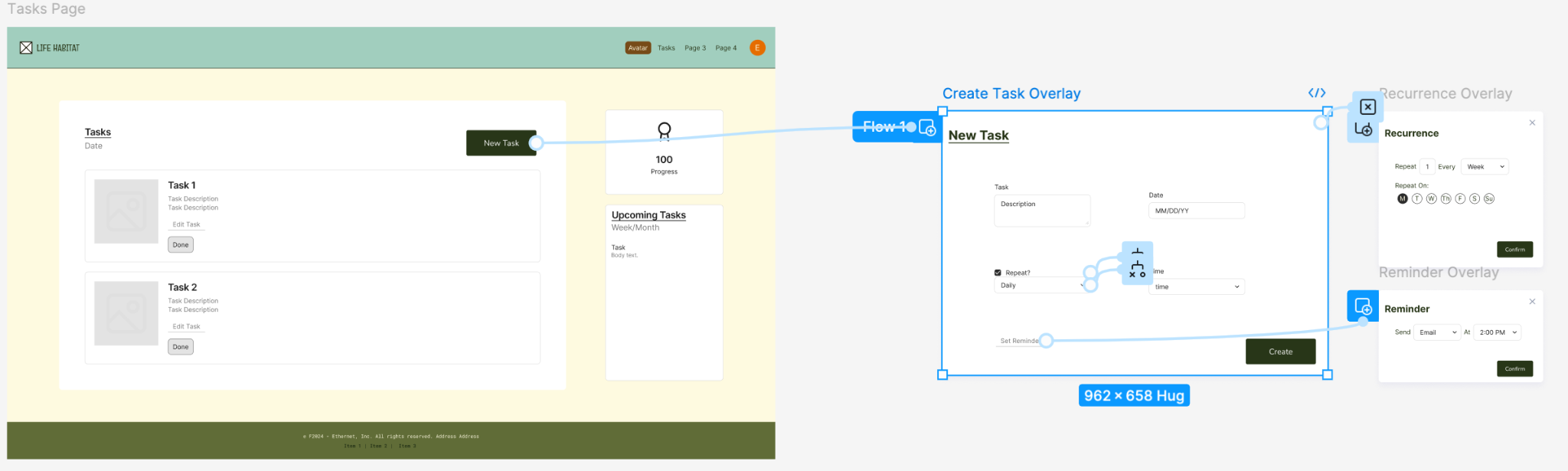


This is an example of Life Habitat’s Task Page. From this page, you can view your current open tasks and see upcoming tasks on the right. You can mark tasks as completed by pressing the ‘Done’ button under their title and description.

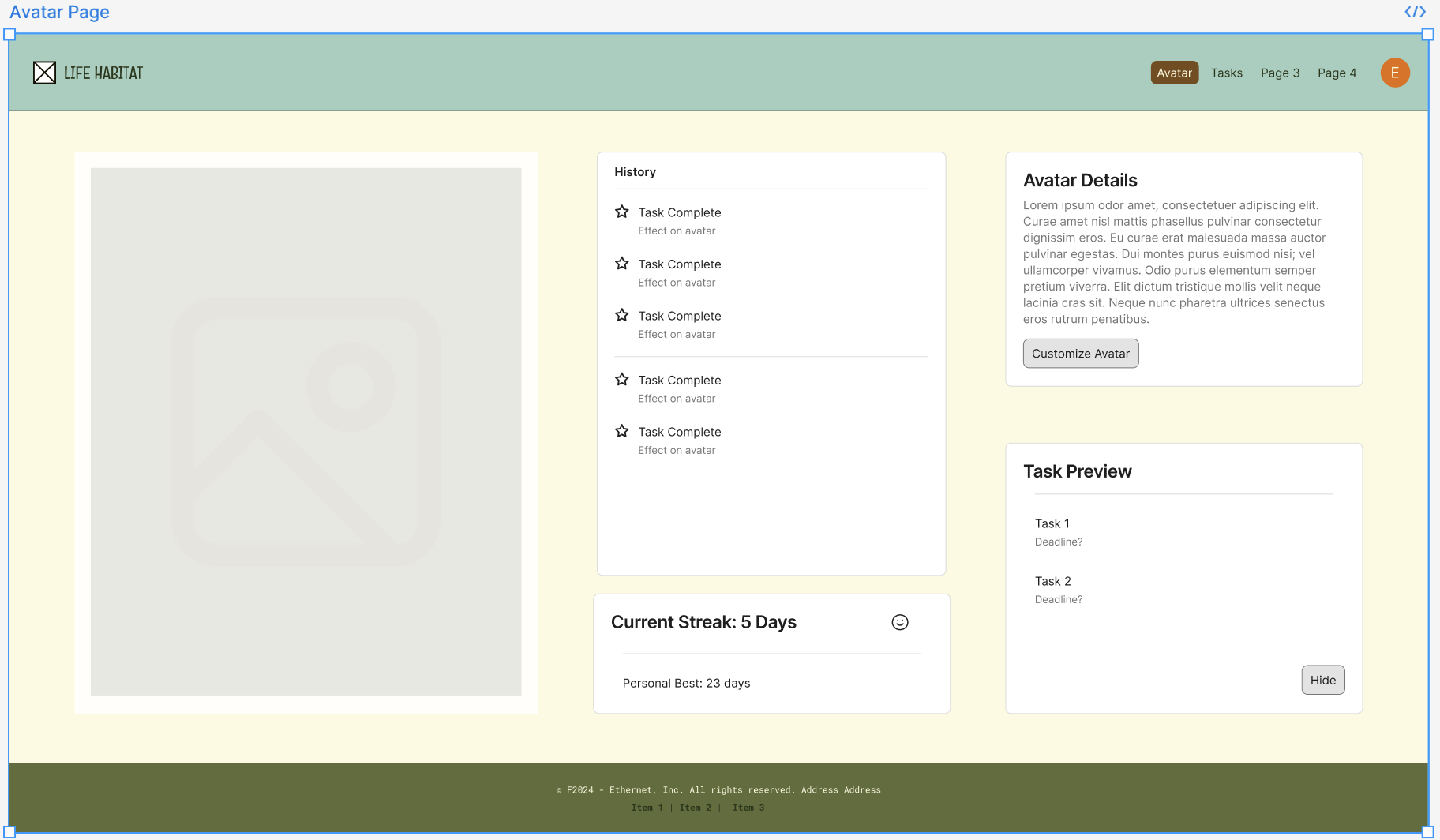
This is an example of the overlay that would appear for a user to create a new task, with the additional recurrence and reminder settings overlays. The repeat checkbox field now has a dropdown menu when clicked, and selecting the custom option opens the more detailed overlay. This was implemented after usability testing for ease of navigation.



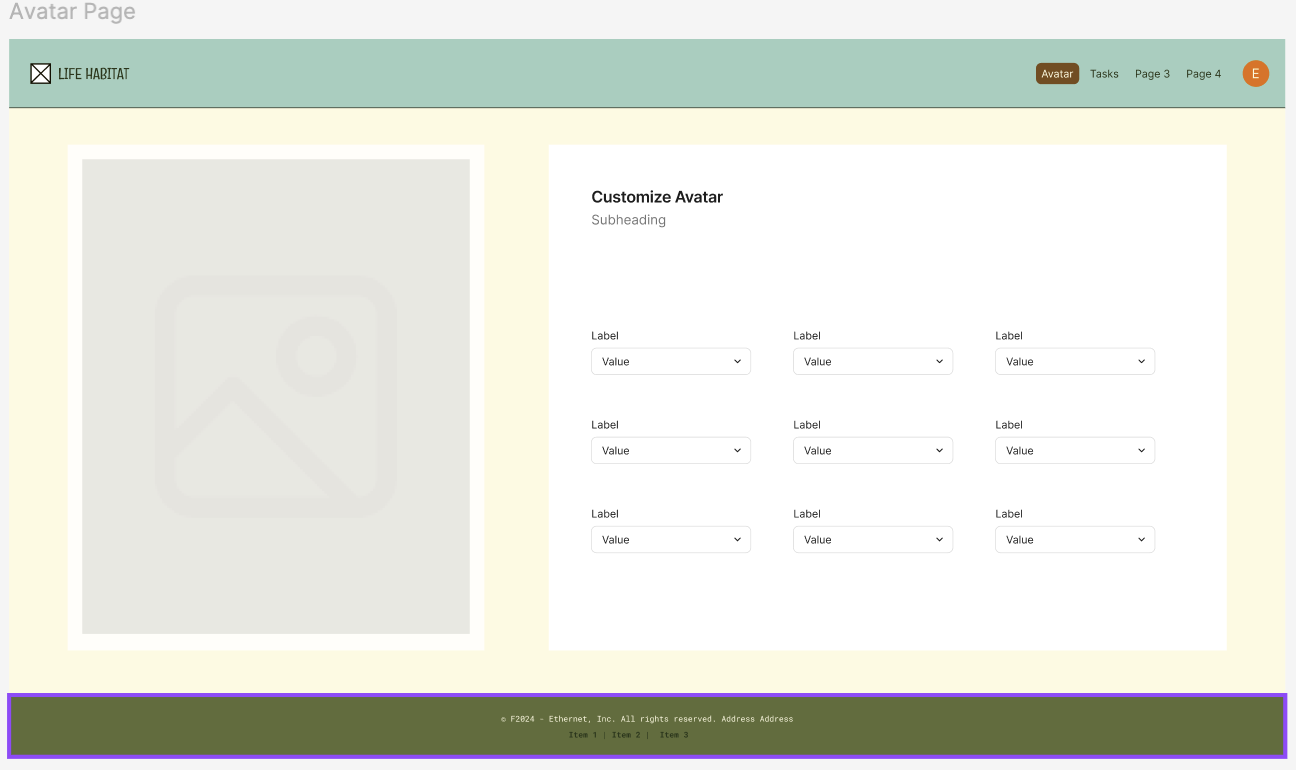
This is an example of the flow for creating tasks. The user hits the New Task button to open the Create Task Overlay, then the user fills in appropriate data and hits the Create button to have the new task appear under the Tasks view.



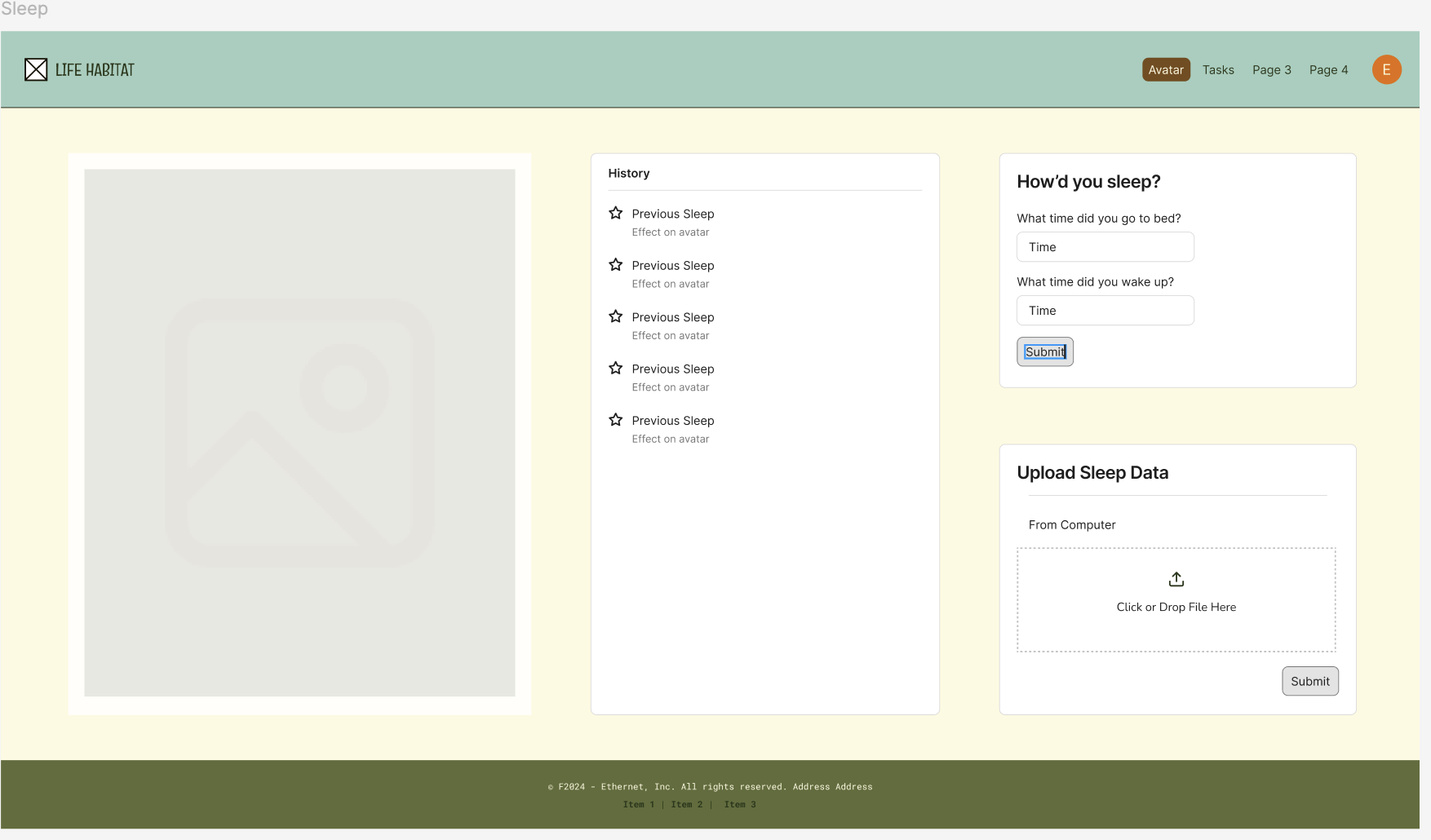
This is an example of the avatar Page. On this page, you can see the avatar (where the gray box currently appears on the left) and its details. You can also preview upcoming tasks and see the history of tasks you have completed. Input from usability testing also recommended adding a “Streak” feature as a means of encouraging users to consistently take care of their tasks.



This is an example of the Customize Avatar Page which is reached through the Customize Avatar Button on the Avatar Page. This page allows users to change certain aspects of their Avatar so it better represents the user’s Self. Drop-down menus will likely include things like Eye Color, Hair Color, etc.



This is an example of the Sleep Tracking Page, where users can log sleep data by manually inputting what times they went to bed and woke up, or they can choose to upload more detailed data collected from other devices.



## **Hardware Interfaces**

Life Habitat will have basic hardware requirements, running on most PCs and other computers, including laptops, as long as they can open a common and current web browser such as Chrome, Safari, Firefox, or Edge. A decent internet connection will also be required for users to log in and save habit data.

Information will be displayed digitally, so a monitor is required, as well as a mouse, trackpad, or touchscreen for interacting with the visual interface.

The website will be hosted by Github, with user data being securely stored on Firebase servers.

*As backend functionality is implemented, this section will be updated.*

## **Software Interfaces**

Life Habitat is built with HTML, CSS, javascript, and React hooks for a simple user interface. The backend is also built with JavaScript and is supported by third-party libraries such as Firebase.

Life Habitat will utilize the Firebase Authentication service in order to collect, verify, and store user login information. This will be done using Firebase’s javascript-based API.

*As functionality is implemented, this section will be updated.*

## **Communications Interfaces**

Life Habitat will take in user email information for the purpose of creating and managing accounts. This data will be stored securely by Firebase.

*Other communication interfaces, such as a designated email address for the application, TBD. As backend functionality is implemented, this section will be updated.*

# **System Features**

## Tasks

4.1.1 Description and Priority

This feature will allow the user to create and modify tasks with different characteristics and to satisfy different needs. Users will be able to specify what a task is for, whether or not it repeats, and more. This feature is High Priority.

4.1.2 Stimulus/Response Sequences

User clicks create task- System opens create task window

User enters task information into the window

User toggles reoccurrence count and reminder count - System applies the appropriate changes

User clicks create button- System gives alerts about the task status

4.1.3 Functional Requirements

FREQ-1.1: The system shall track user-created tasks.

FREQ-1.2: The system shall provide a space for the user to enter a description of each task.

FREQ-1.3: The system shall repeat daily or weekly tasks accordingly.

FREQ-1.4: The system shall allow users to get task recommendations through AI generation.

FREQ-1.5: The system shall request to send the user push notifications for task reminders

FREQ-1.6: The system shall allow users to update prior created tasks and goals with new info.

FREQ-1.7: The system shall recommend new healthy habits to the task list once a user has successfully completed previous repeated tasks three times in a row.

## User Encouragement

4.2.1 Description and Priority

This feature will provide the user with encouragement while or after completion of their tasks and will uphold a friendly atmosphere. Users will receive notifications as often as specified to complete a task and positive feedback on their progress. This feature is High Priority.

4.2.2 Stimulus/Response Sequences

User has a task in progress - System provides the user with reminders of their task

User still has a task in progress - System provides encouragement to finish

User completes their task - System provides user with positive reinforcement

4.2.3 Functional Requirements

FREQ-2.1: The system shall have the avatar provide additional encouragement when the user fails to complete a task.

FREQ-2.2: The system shall provide encouraging feedback to the user by sending motivational messages when a task is failed.

FREQ-2.3: The system shall reward users for completing tasks without using reminders and notifications if set by a user by providing achievements or points when a task is marked completed by the user.

FREQ-2.4: The system shall motivate the user to build positive habits with a customizable avatar that reflects the user’s progress and habits.

## Avatar

4.3.1 Description and Priority

The application will include a customizable avatar to reflect the user’s habits in caring for themself. Failing to complete tasks will lead to a more disgruntled appearance, whereas being timely and making progress will lead to an improvement in the avatar’s health. This feature is High Priority.

4.3.2 Stimulus/Response Sequences

User completes tasks on time - System reflects joy in the user’s avatar

User continuously completes tasks - System rewards the user with points and badges

4.3.3 Functional Requirements

FREQ-3.1: The system shall allow users to customize their avatars with any unlocked rewards.

FREQ-3.2: The system shall allow the user to set or not set reminders and notifications for any created tasks.

FREQ-3.3: The system shall allow the user to buy avatar skins with their achievement points.

## Account

4.4.1 Description and Priority

The application will require that users create accounts. This will allow the system to track user progress and save user data. This feature is High Priority.

4.4.2 Stimulus/Response Sequences

User clicks login - System prompts user to enter login info

User enters login info and submits - Sytem returns valid or not valid login, redirects accordingly

User clicks register - System prompts user to enter user login info

User enters login info and submits - System returns account creation status; if email already in use; prompt to enter new email.

4.4.3 Functional Requirements

FREQ-4.1: The system shall allow users to sign up with an email and password.

FREQ-4.2: The system shall allow users to edit their profile to add personal information.

FREQ-4.3: The system shall allow users to delete their accounts.

# **Other Nonfunctional Requirements**

## **Performance Requirements**

NFREQ 1.1 The system shall allow for up to 100,000 simultaneous users while maintaining optimal performance.

NFREQ 1.2 The system shall allow users to find any app page within five clicks on their mouse.

NFREQ 1.3 The system shall have a page load time of under 3 seconds in 95% of all instances within any given month.

NFREQ 1.4 The system shall run on major web browsers: Chrome, Safari, and Firefox.

NFREQ 1.5 The system shall have a responsive and dynamic UI across different screen sizes to provide a seamless user experience.

## **Safety Requirements**

*There are currently no safety requirements associated with Life Habitat*

## **Security Requirements**

NFREQ 3.1 The system shall encrypt user data to protect it from other users and third parties.

NFREQ 3.2 The system shall follow the General Data Protection Regulation (GDPR).

NFREQ 3.3 The system shall check for user idle time and automatically log out after an hour of inactivity

## **Software Quality Attributes**

NFREQ 4.1 The system shall be able to send or receive data from servers within 10 seconds 99% of the time.

NFREQ 4.2 The system shall be online 95% of the time with the exception of scheduled maintenance blocks.

NFREQ 4.3 The system shall delete accounts that have been dormant for more than two years.

NFREQ 4.4 The system shall backup daily to the database to minimize lost data.

NFREQ 4.5 The system shall recover from any extraneous errors using the most recent backup in the database.

## **Business Rules**

*<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>*

# **Other Requirements**

NFREQ 6.1 The backend database shall be a PostgreSQL object-relational system.

NFREQ 6.2 The frontend client shall be written in React typescript.

NFREQ 6.3 The generative AI client used for task creation will be ChatGPT 4-o

NFREQ 6.4 The system shall follow the FTC guidelines for privacy.

NFREQ 6.5 The system shall not share user data with third parties.

**Appendix A: Glossary**

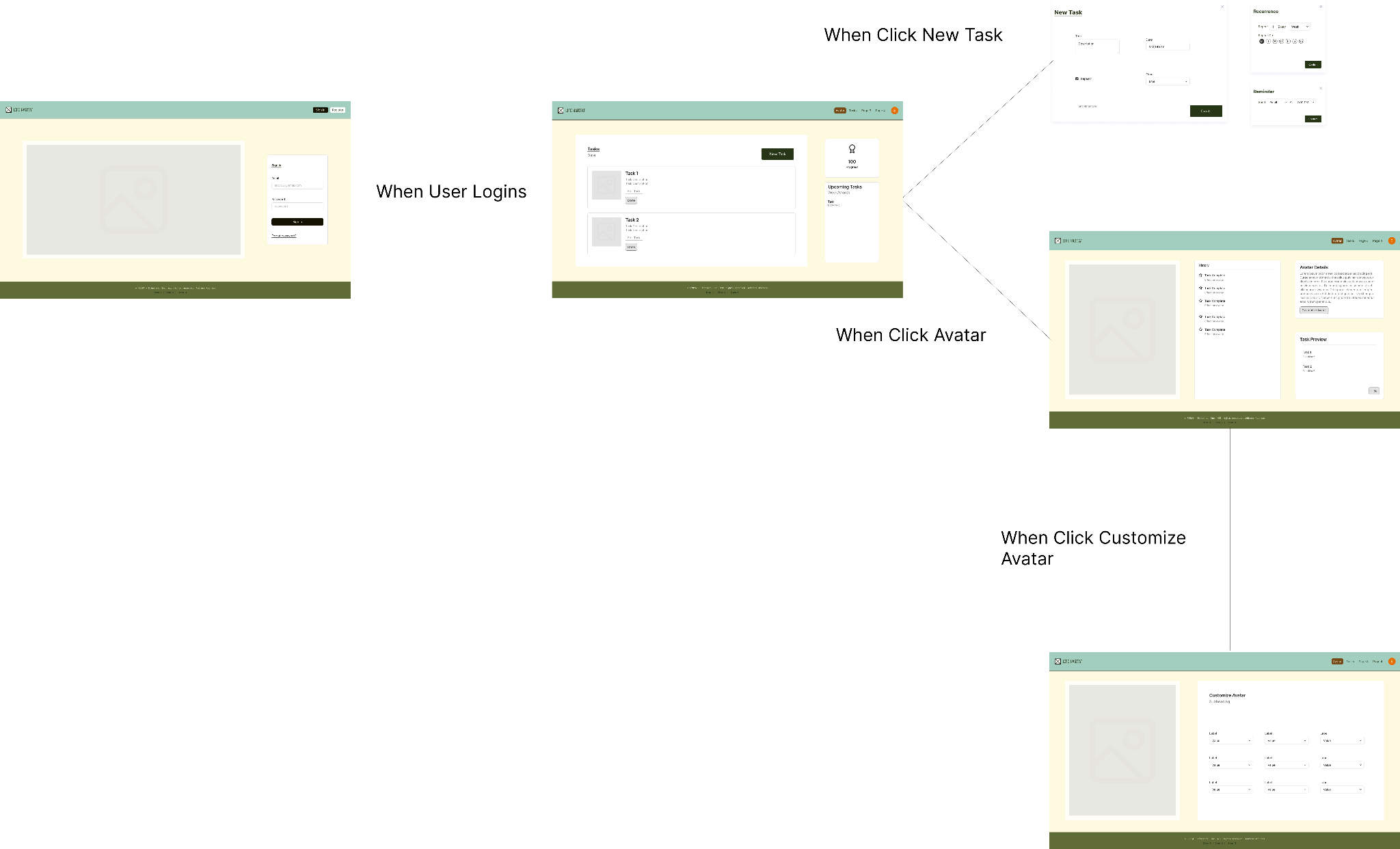
Life Habitat: The Company Name

Tamagotchi: A virtual pet that was first introduced in the 1990s

React: Frontend API client software

Postgresql: Backend SQL database software

**Appendix B: Analysis Models**

**

[*https://www.figma.com/design/L6Otl6x2FM9YiG9tJF4XmX/Life-Habitat?node-id=230-843&t=L34YHQHnWbCViyWI-1*](https://www.figma.com/design/L6Otl6x2FM9YiG9tJF4XmX/Life-Habitat?node-id=230-843&t=L34YHQHnWbCViyWI-1)

**Appendix C: To Be Determined List**

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*