### 

### 

Milestone 1: Team Project Proposal and Description

### CEN 4010 Principles of Software Engineering

### Summer 2023

*Project Name* ***-* Task Buddy**

*Team Name - Task Buddies*

*Team Number - 18*

*Github repo link:* [*https://github.com/leoalfonso14/TaskBuddy.git*](https://github.com/leoalfonso14/TaskBuddy.git)

*Jira Board: https://taskbuddy.atlassian.net/jira/software/projects/TAS/boards/1*

Names of students

(team lead first) and emails:

[Sofia Torres](mailto:sofiatorres2022@fau.edu)

[Mia Marte](mailto:mmart2020@fau.edu)

[Hunter Padilla](mailto:hpadilla2020@fau.edu)

[Leandro Alfonso](mailto:lalfonso2022@fau.edu)

[Dominique Nelson](mailto:nelsond2015@fau.edu)

To be submitted on 7/18/23

\*History table can be provided upon request

# **Executive Summary**

Our project will be a Web Application called TaskBuddy. Our app will be designed to help you stay organized, track your habits, and accomplish your goals, all while nurturing a virtual pet. Your pet will serve as your digital companion, encouraging you to complete your daily habits and to-do lists efficiently by keeping you responsible for the pet’s health.

In general, many college students struggle with procrastination; college in itself can already be difficult when balancing family, work, and a social life. We developed this project in hopes to allow students and other individuals to consistently be more productive and try to build more positive habits. We know there are already websites to keep your life organized but this project is more focused on the users growth, for this reason we added the virtual pet companion to incentivize the user to be more proactive about their own task, habits, and goals. If the user does not acknowledge their to do list tasks or habits, the well being of their virtual pet diminishes. Even though it's just a virtual pet, the idea is to have the user form this bond, make them feel responsible for their pet in order to encourage them to take initiative in their own growth.

The app provides a comprehensive dashboard where you can easily set reminders, prioritize tasks, and track your progress. Each time you complete a task you earn points or coins that are used to sustain and care for your adorable virtual companion. By nurturing your pet, you develop a sense of responsibility and attachment, further fueling your motivation to stay productive and achieve your goals. The coins you earn will not dictate directly the state of your pet, but rather they are used to buy necessities for them, adding an element of fun and engagement to your productivity journey. With these coins you will be able to buy food and clothes for your pet, among other things. Once you have reached a certain number of coins, you are eligible for level ups.

With each level up, you gain the ability to design your pet's home, creating a personalized sanctuary for your virtual friend. This creative outlet provides a sense of achievement and allows you to express your individuality as you continue to thrive in your daily life. We will also add a community feature that allows you to connect with friends, view their pets, and view their worlds, further enhancing the overall experience. This feature enables you to add friends within the app, so you can see how your friends are nurturing their companions and what level they've reached. This feature also creates a sense of camaraderie and encourages healthy competition, inspiring everyone to stay productive and motivated.You will also be able to participate in challenges with your friends, such as who can achieve the longest streak of completed tasks or the highest number of coins earned in a week. This app harnesses the power of technology to enhance our productivity and motivate ourselves through the caring of a digital pet and healthy competition with other users.

# **2 Competitive analysis**

| Name | Task Buddy | Microsoft Outlook Calendar |
| --- | --- | --- |
| Key Features: | * Ability to manage and maintain happiness of pets * Pets will encourage users to continue to complete events * User landing page will sync with google Calendar * Data storage in a private database, directly linked to the site. | * Calendar Sharing * Targets a Workplace environment * Can Sync calendars with co-workers. |

As per the chart above, the main goal is to assist users actively participating in everyday tasks and use the concept of Pet Life to reinforce consistency. Microsoft Outlook has features that allow users to work with others, and even synchronize their calendars with other employees. Our goal is to take the features that outlook excels in, and make them our own in a slightly different way, and expand upon them as well as add our unique ideas. The idea of having a Task Buddy will help to reinforce your attendance in the workplace and desire to complete tasks, and when you aren’t meeting requirements of the workplace it will be shown through the happiness level of your task buddy. Updates, if time permits, will include a challenge:

* Challenges: Competitions within TaskBuddy where users can participate in tasks or achievements against their friends or other users. Challenges such as who can achieve the longest streak of completed tasks or the highest number of coins earned in a week.
* Level Up: The process of progressing to higher levels within TaskBuddy by earning experience points or achieving specific milestones. Each level unlocks new features and customization options.
* Community Feature: A feature within TaskBuddy that allows users to connect with friends, view their pets, and engage in challenges and healthy competition. It promotes camaraderie and inspires productivity.

# **3 Data definition**

**\*List items subject to change as we progress throughout the project\***

Terms and Entities:

TaskBuddy: The web application being developed, which combines task management, habit tracking, and gamification elements with a virtual pet.

Virtual Pet: A digital companion within TaskBuddy that users nurture by completing tasks and earning points or coins. The virtual pet serves as a source of motivation and responsibility.

Dashboard: The central hub within TaskBuddy where users can set reminders, prioritize tasks, and track their progress. It provides an overview of the user's productivity and serves as a management tool.

Coins: In-app currency earned by completing tasks in TaskBuddy. Coins are used to purchase necessities for the virtual pet and add an element of fun and engagement to the productivity journey.

Shop: A shop is going to be implemented, so the user can buy their virtual pet food, water, toys, and other accessories. This will be an interactive way to involve the user, and make them feel tangible progress through the inventory from the shop.

Calendar: A simple monthly view format that is not overwhelming and incorporates color organization to differentiate between tasks more easily. Users can add new events to their calendar by clicking on a specific date and filling in event details. They can set the event title, start and end times, location, description, and optionally, assign it to a specific category or project.

To-do List: Users can add new tasks to their todo list using a simple input field. They can enter a task description, as well as edit and delete. Additionally, users can set a due date that later on will help them track their progress for the week. Some tasks may be repetitive, such as weekly tasks or habits. However, our web app will include an option to create recurring tasks, allowing users to set tasks to repeat at specific intervals.

These terms and entities will be used consistently throughout the application, providing users with a clear understanding of the functionality and features of TaskBuddy.

# **4 Overview, scenarios and use cases**

*Task Buddy Project Overview:*

Introduction: TaskBuddy is a website with the purpose to help individuals stay organized, track their habits, and achieve their goals while caring for a virtual pet. The webapp aims to help people overcome their procrastination and get them to build positive habits through a gamified approach for more incentive.

Target Audience: TaskBuddy’s audience was originally thought of for college students, however it is truly for any individual looking to improve their productivity and build positive habits.

Objectives:

* Provide a dashboard that will help users organize their tasks, habits, and goals.
* Motivate users to keep completing their tasks with the responsibility of a virtual pet.
* Have the user build a bond with their virtual pet companion to keep them active in their habit building.
* Allow users the option to invite friends and see each other's virtual pet world progress for a sense of friendly competition that further adds motive to keep being productive.

Key Features and Functionality:

Dashboard

* Have a dashboard that displays user’s to-do list, habits, and goals.
* Dashboard can be viewed in monthly or listed(columns) view.
* Reminders feature for to-do list and habits.

Virtual Pet

* The user is responsible for their virtual pets well-being
* By checking off tasks on their to-do list and being consistent with habits, the user will get coins. The coins will be used to provide care for their pet.
* The virtual pet aims to help the user be consistent in their tasks.

Community

* User’s can invite and connect with their friends to see how each others pet are doing
* Being able to view each other's virtual pet world progress will put more pressure on users to keep on track with their dashboard.

Technology Stack: TaskBuddy will be built using HTML, CSS, JavaScript, and Parse for backend framework. Software to help facilitate the development process of TaskBuddy will be Jira, GitHub, and VSCode.

Risks: Potential risks include competition from similar productivity apps. To limit this risk we want to implement these features, such as the virtual pet, and even more so with the community feature.

Likelihood Usage Scenarios:

Sarah - The Achiever:

Sarah is a young professional who wants to excel in her career and personal life. She uses TaskBuddy to create a to-do list and prioritize her tasks. With the virtual pet companion by her side, she feels a sense of responsibility to complete her tasks to ensure the well-being of her pet. She sets reminders, schedules her activities, and monitors her progress. As she completes her tasks, her virtual pet thrives and grows, reinforcing her motivation to stay productive.

David - The Habit Builder:

David is determined to establish new habits and break old ones. He utilizes TaskBuddy to set up habit trackers and reminders for activities like exercise, reading, and meditation. The virtual pet companion becomes a visual representation of his progress, rewarding him with achievements and unlocking new features as he consistently practices his habits. David feels a sense of accomplishment and is motivated to maintain his streaks.

Emily - The Organizer:

Emily is a busy mom managing her household, work, and social commitments. TaskBuddy becomes her go-to tool for keeping everything organized. She uses it to create shopping lists, plan meals, schedule appointments, and manage family events. The virtual pet companion adds an element of fun for her children, who help take care of the pet by completing their own tasks. Emily feels more in control and less overwhelmed, knowing that she has a reliable system to manage her responsibilities.

Alex - The Procrastinator :

Alex often struggles with procrastination and finds it challenging to stay focused. TaskBuddy becomes his accountability partner. He breaks down his tasks into smaller, manageable steps and sets timers to work in focused intervals. As he completes each task, his virtual pet rewards him with virtual treats and unlocks mini-games for short breaks. Alex feels a sense of achievement, and the gamification aspect keeps him engaged and motivated to conquer his procrastination tendencies.

In all these scenarios, the end users - Sarah, David, Emily, and Alex - have varying levels of experience and skill when it comes to productivity and habit-building. TaskBuddy simplifies the process by providing a visually appealing interface, intuitive task management features, and the added motivation of a virtual pet companion. The users focus on defining their goals, organizing their tasks, and tracking their progress, while TaskBuddy handles the backend complexities to ensure a smooth and enjoyable user experience.

Use case -Sign up

The user comes to the home page and wants to enhance productivity and establish habits with the help of a virtual pet. User selects the sign up button to begin using the platform.

1. Description:
   1. Use case describes the process of how a user will initiate the registration process.
2. Actors:
   1. User
   2. System
3. Preconditions:
   1. User has an active internet connection
   2. System is available
4. Primary Flow of Events:
   1. User arrives to homepage
   2. User is prompted to provide his personal information such as name, email address, and a chosen password.
   3. User receives a verification email containing a unique link.
   4. User clicks on the link to confirm his email address and activate his account.
   5. User logs in and is greeted with an interactive process that show key features of the platform
   6. Terminate Use Case Sign Up
5. Alternative Flow of Events:
   1. User did not enter valid email address
      1. System displays an error message, prompting the user to re-enter their email address
      2. System will keep displaying an error message until the user enters a valid email address.

Use Case - Login

The user comes to the home page and wants to continue being productive and building habits. User selects the login button to open their dashboard.

1. Description
   1. Use case describes how a user will login to the system
2. Actors
   1. User
   2. System
3. Preconditions:
   1. User has an active internet connection
   2. System is available
   3. User has in active account
4. Primary Flow of Events:
   1. User locates the "Log In" or "Sign In" button and clicks on it.
   2. Users are redirected to the login page, where they are prompted to enter a registered email address and password.
   3. User enters email address and password accurately
   4. System verifies user email address and password combination for authentication
   5. Upon valid credentials, login request is approved, user is granted access to their account
   6. Useris redirected to their personalized dashboard
   7. Terminate Use Case Login
5. Alternative Flows:
   1. User enters incorrect credentials
      1. System displays an error message, prompting the user to re-enter their credentials.
      2. If user initiate a password reset, the user receives an email with instructions to reset their password
   2. User does not have an active account
      1. User is directed to registration page to create a new account

Use Case - Dashboard

The user comes to the home page and interacts with the "Dashboard" button.

1. Description
   1. Use case describes how a user will use the dashboard feature
2. Actors
   1. User
   2. System
3. Preconditions:
   1. User has an active internet connection
   2. System is available
   3. User has an active account
   4. User is logged into their account
4. Primary Flow of Events:
   1. User has logged in (see Use case Login).
   2. Users clicks the dashboard button
   3. System checks if user is logged in
   4. If user is not logged in go to Use Case Login
   5. System redirects the user to their personalized dashboard.
   6. User can see an overview of their ongoing tasks, habits, and progress towards goals on the dashboard.
   7. The dashboard may display a calendar or list view, allowing the user to schedule and track tasks, set deadlines, and receive reminders.
   8. Terminate Use Case Dashboard
5. Alternative Flows:
   1. User is not logged in
      1. System prompts the user to enter their credential ( see: Use case -login).
   2. User does not have an active account
      1. User is directed to the registration page to create a new account (see use case - sign up).

Use case - Virtual Pet

The user comes to the homepage and wants to interact with the virtual pet companion feature.

1. Description
   1. Use case describes how a user will interact with their virtual pet through the system
2. Actors
   1. User
   2. System
   3. Virtual Pet
3. Preconditions:
   1. User has an active internet connection
   2. System is available
   3. User has an active account
   4. User is logged into their account
   5. User account must be associated with a virtual pet
4. Primary Flow of Events:
   1. User has logged in (see Use case Login).
   2. Users clicks the Virtual Pet button
   3. System checks if user is logged in
   4. If user is not logged in go to Use Case-Login
   5. System redirects the user to the Virtual Pet page, this may look like a virtual pet habitat
   6. The user will see their virtual companion on the screen
   7. User can customize their virtual pet settings such as appearance, hunger status, and happiness status
   8. The User can use their coins to buy food to feed their pet
   9. The user can earn coins by completing tasks on their dashboard and maintaining consistent habits
   10. The user can return to their pet by clicking the Virtual Pet button at any time
   11. Terminate Use Case-Virtual Pet
5. Alternative Flows:
   1. User is not logged in
      1. System prompts the user to enter their credential ( see: Use case -login)
   2. User does not have an active account
      1. User is directed to the registration page to create a new account (see use case - sign up)

# **5 High-level functional requirements**

* **1. User Landing page**:
  + Users will be able to enter the website and be greeted by a landing page. The landing page will prompt the user to enter credentials for the website and allow them into their calendar space. If users are brand new to the application, they will also be able to enter credentials for storage in our User database as well as be prompted to link their google calendar account.
* **2. Authentication:**
  + Users will be able to sign in with their Google account, Apple ID, or their email. If the user is already logged in they will see our Internal Home Page and will have access to our platform.
* **3. User Records:**
  + User Records will be stored in a third party database. Currently the information is planned to be stored in a database known as Firebase. This is where we will keep track of things such as missed/completed tasks or events as well as basic login credentials.
* **4. Animated Interactivity:**
  + Users will be able to view/ interact with their pet depending on the amount of events that have been marked as cleared on their calendars. This will promote users to enter more events onto their calendar and then mark them as completed. Everytime they are able to complete more events they are able to earn more coins which they can use to maintain the happiness of their pets. Pets will be designed to have particular animations or gifs which will display their happiness levels.

# **6 List of non-functional requirements** Some of our non-functional requirements are:

# **Usability:**

# The web-app should have an user-friendly interface.

# It should also provide instructions that are clear, and error messages to help users.

# Should be accessible with different browsers, and devices.

1. **Performance:**
   1. By adding a fast load speed, we can minimize user wait time.
   2. The web-app should handle concurrent user interactions efficiently.
   3. Actions like adding or updating tasks should have minimal response time.
2. **Security:**
   1. We should use secure authentication mechanisms, such as password hashing and encryption.
   2. We will implement a Login/Password system, where users will be able to access their virtual pet.
   3. The web-app should protect user data and prevent unauthorized access or data breaches.
   4. The app should implement measures to prevent cross-site scripting and SQL injection attacks.
3. **Reliability:**
   1. The web-app should be available and accessible to users with a high uptime percentage.
   2. The web-app should provide informative error messages when errors occur.
4. **Scalability:**
   1. The web-app should be able to handle a growing number of users and tasks without significant performance degradation.
   2. The web-app should efficiently manage database connections and handle increasing amounts of data.
5. **Compatibility:**
   1. The web-app should be compatible with different operating systems, including Windows and macOS.
   2. The web-app should support major web browsers such as Chrome, Firefox, Safari, and Edge.
   3. The web-app should adapt to various screen sizes, including desktop, tablet, and mobile devices.
6. **Maintainability:**
   1. The codebase should be well-structured, modular, and stick to industry-standard coding practices.
   2. The web-app should have proper documentation, including a user manual and developer documentation.
   3. The web-app should be easily upgradable, allowing for the addition of new features or enhancements.
7. **Privacy:**
   1. The web-app should provide options for users to control their privacy settings, including data sharing and visibility of tasks.
   2. The web-app should have a clear and comprehensive privacy policy explaining the collection and usage of user data.

# **7 High-level system architecture**

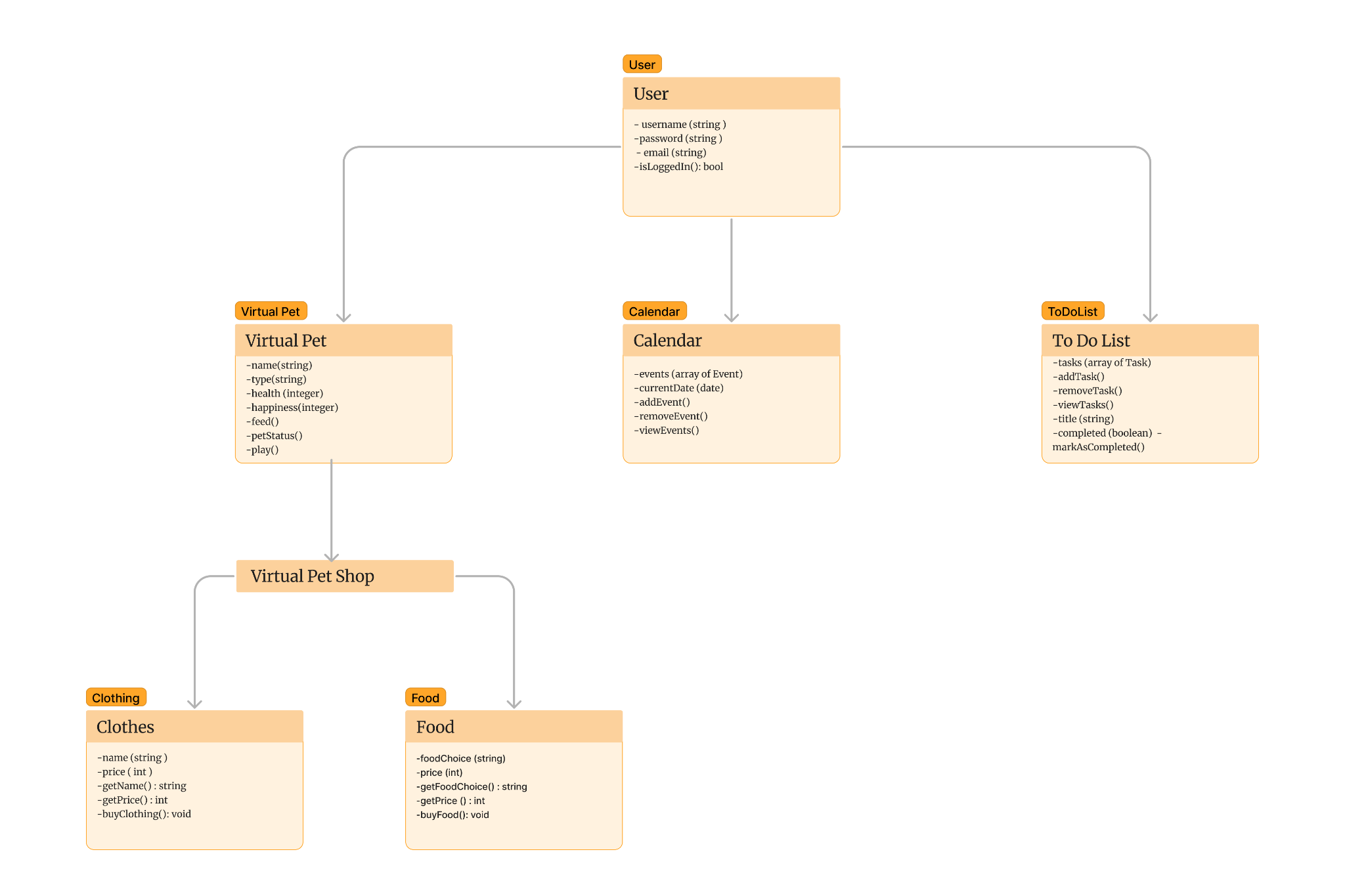
These are our high-level system architecture for our web-app.

* Our main software products are:
* Jira: We will use Jira to plan, track and manage our work efficiently. Where each team will be in charge of a task or a problem that needs to be addressed.
* GitHub: Will be used to manage code changes, review code, track issues, automate process and document projects. Where each team is responsible for updating and keeping track of their task in GitHub.
* VSCode: Will be used as our code editor since our main language will be JavaScript, and everyone in the team is familiar with VSCode.
* Figma: Will be used to design the interface of the web-app. Where everyone in our team will get together, and implement ideas on how we want our web-app to look like.
* Software tools:
* SheCodes Coding Tools: We will be using these tools to search for color palettes that go according to our web-app.
* Favicon Generator: Will be used to generate our favicon.
* Languages:
* HTML 5- Will be the language that is going to be used to display the website.
* CSS- Will be the language used to style the page.
* Javascript - Will be used for back- end as well as front- end to make the coding process easier.
* Bootstrap - Will be the framework of the web-app that will be used to construct the front-end of our web page. Link for LICENSE :

<https://github.com/leoalfonso14/TaskBuddy/blob/main/LICENSE>

* Databases:
* Firebase- Will be used to cover everything from hosting and authentication to real-time database management and cloud functions. By using firebase, we will get to monitor, track and analyze the performance of our web app. Where we can monitor our app startup times, screen rendering and network latency, which can later help us optimize the user experience.
* APIs
* Google Calendar API- Since our web-app is not pulling data from anywhere else, the use of API’s is not needed, but if we end up using it, we will be using Google Calendar. We chose it because our web-app would be linked to a user's google calendar, and depending on successful completion of events marked through the google calendar, the virtual pet would be rewarded.
* Supported browsers:
* Our web-app will operate in two major browsers: Google Chrome and Microsoft Edge. We will also try to operate in Apple’s Safari, so that the users that prefer their iPhones, iPads or Macbook are able to access our web-app.

# **8 High-Level UML Diagrams**



# **9 Risk Assessment**

1) Technical risks: There may be challenges in integrating the virtual pet functionality and gamification elements into the web application. Ensuring smooth animations, responsiveness, and user interaction with the virtual pet could pose technical challenges.

2) Schedule risks: Meeting the project timeline and delivering the web application within the specified timeframe may be a risk. Adequate planning and allocation of resources will be crucial to ensure timely completion.

3) Teamwork risks: Collaboration and effective communication among team members will be essential for the success of the project. Ensuring that team members are aligned with the project goals, coordinating tasks, and resolving conflicts or challenges that may arise will be critical.

4) Skills risks: Everyone involved in the project should possess the necessary skills to develop the web application and handle the technologies involved (HTML, CSS, JavaScript, Parse). Identifying any skill gaps and addressing them through training or recruitment may be necessary.

5) User adoption risks: The success of the web application relies on user adoption and engagement. Ensuring that the features, usability, and overall experience of TaskBuddy meet user expectations and needs will be important to drive user adoption.

6) Legal/content risks: Ensuring compliance with relevant legal and privacy regulations when handling user data and integrating with external services such as Google Calendar API is crucial. Adhering to copyright laws when using animations or graphics for the virtual pet and maintaining appropriate content standards will be important.

To mitigate these risks, as a team we should conduct thorough planning, define clear objectives and deliverables, allocate resources effectively, regularly communicate and collaborate, conduct testing and quality assurance, and stay updated with relevant legal and privacy requirements. Regular risk assessments throughout the project lifecycle will help identify and address potential risks promptly.

# **10 Submission**

TBD

# 