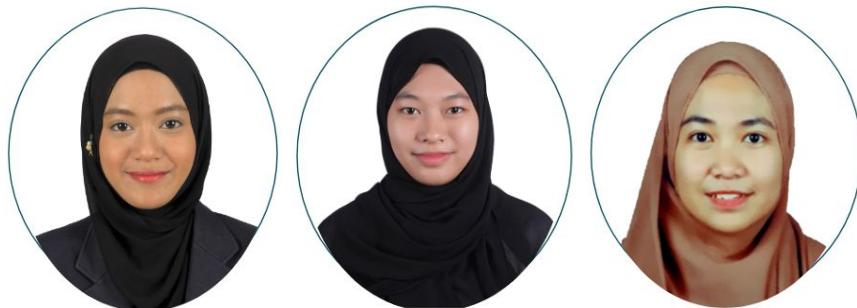




SEMESTER I SESSION 2020/2021

SCSV3104 - 01 APPLICATION DEVELOPMENT



PROJECT TITLE

TASK MANAGEMENT MOTIVATOR APPLICATION (POND)

LECTURER

PROF. DR. MOHD SHAHRIZAL BIN SUNAR

PREPARED BY GROUP IRIS

AIMI BINTI RUSDI (B19EC0001) <Project Manager>

MIRHANIEZA BINTI MATHARUZAMAN (A18CS0106) <Full Stack Developer>

NURAMYRA NATASHA BINTI ISMALLUDIN (B19EC0035) <UI/UX Developer>

Bachelor of Computer Science (Computer Graphic and Multimedia Software)

© 2021 All Right Reserved

Table of Contents

No	Title	Page
Executive Summary.....		5
1. Introduction.....		7
1.1 Problem Statement.....		7
1.2 Requirement Specification.....		9
1.3 Needs, Approach, Benefit & Competitors (NABC).....		10
1.4 Functional Requirements.....		11
1.5 Non-Functional Requirements.....		12
1.6 Team Members and Roles.....		13
2. Gamification Concept.....		15
3. Specific Requirements Diagram.....		23
3.1 External Interface Requirements.....		23
3.1.1 User Interface Implementation.....		23
3.1.2 Hardware Interface Implementation.....		24
3.1.3 Software Interface Implementation.....		24
3.1.4 Runcloud Deployment.....		25
3.1.5 Tenenet Gamification Implementation.....		27
3.2 System Features.....		30
3.2.1 Use Case Diagram.....		30
3.2.2 Domain Model Diagram.....		31
3.2.3 Architectural Design Diagram.....		32
3.2.4 Component Diagram.....		33

4. Subsystem 1 : Task Management.....	34
4.1 Specific Requirement.....	34
4.2 UC1 : Use Case <Manage Task>.....	34
4.2.1 Use Case Description.....	34
4.2.2 Sequence Diagram.....	36
4.3 Data Design.....	37
4.3.1 Data Description.....	37
4.3.2 Data Dictionary.....	37
4.4 User Interface Design.....	39
4.4.1 Overview of User Interface.....	39
4.4.2 Screen Images.....	39
4.5 Test Cases.....	42
4.5.1 TC1_01 : Test <Scenario of Manage Task (UC1)>.....	42
4.6 Test Approach Analysis.....	42
5. Subsystem 2 : Pond Management.....	43
5.1 Specific Requirement.....	43
5.2 UC2 : Use Case <Claim Item>.....	43
5.2.1 Use Case Description.....	43
5.2.2 Sequence Diagram.....	45
5.3 UC3 : Use Case <Decorate Pond>.....	46
5.3.1 Use Case Description.....	46
5.3.2 Sequence Diagram.....	48
5.4 UC4 : Use Case <View Hall of Fame>.....	49
5.4.1 Use Case Description.....	49
5.4.2 Sequence Diagram.....	51

5.5	Data Design.....	52
5.5.1	Data Description.....	52
5.5.2	Data Dictionary.....	52
5.6	User Interface Design.....	54
5.6.1	Overview of User Interface.....	54
5.6.2	Screen Images.....	54
5.7	Test Cases.....	57
5.7.1	TC2_01 : Test <Scenario of Claim Item (UC2)>.....	57
5.7.2	TC2_02 : Test <Scenario of Decorate Pond (UC2)>.....	57
5.7.3	TC2_03 : Test <Scenario of View HOF (UC2)>.....	58
5.8	Test Approach Analysis.....	58
6.	Code Versioning	60
7.	Software Release	62
7.1	Pre-alpha Version.....	62
7.2	Alpha Version.....	62
7.3	Laravel and CSS.....	62
8.	Bugs Record Document	64
9.	Appendices	66

Executive Summary

As individuals, especially when we have a lot of things to do, it is not easy to accomplish specific tasks without organization, we do not know how to start and end up letting something be done beyond our control. Missed deadlines, improper workflow and poor quality of work can lead to low self-esteem, so an application for productivity is what these individuals need! We come out with the idea to build a Personal Motivator App called '**POND**' to assist them in managing their time effectively. The proposed solution to this problem is to create an application where these individuals can manage their tasks well by customizing their own activities at their own pace.

Does your lifestyle seem to prevent you from enjoying your time due to too much work and things that need to be complete? Many individuals fail because they do not try to plan, set priorities and obey their tasks. So, doing a routine list check and stuff to do is the best way to enhance it. Moreover, application for productivity will not move on their own, it requires human encouragement to stand up and do all the work. Notifying and reminding the user to perform the task are all applications that can be made to help the users. Nevertheless, we come up with all the interesting features that make it easy for users to use them efficiently. In addition, procrastination is not because you are unresponsive, it's become everyone's habit to put off doing something. Therefore, interesting alerts notifications are provided in this application to alert users of incomplete tasks in order to warn them of their future submitted activities.

The productivity application would target the people at all ages and intended as personal management. By allowing them to schedule upcoming activities such as setting appointments, assignment deadlines or bill payments and more, the calendar features in this app would enable them to set information about future dates. It enables users to arrange all these important tasks in this way, providing them with ease and the ability to improve in handling tasks. If settings are available, a reminder will be needed. Yes, in the application, we have alarms to warn users of what needs to be completed before a deadline. The reminder feature will notify you of important things to do and help you remember them. In addition, there is a list of activities that users want

to complete on a specific day, they can write into a list of things to do. Task lists are where users can follow tasks in their pace, making it easier to prioritize tasks if you have a to-do list. This will ensure that you focus on the right track at all times. Reflection is the most incredible function supported by this app! After completing their assignment, users should write personal reflections to help them evaluate their success instead of only doing what they do.

They will gain a deeper understanding of the challenges, feelings, strengths and shortcomings you have been through to complete the evaluation by writing reflections. Last but not least, in the real world, you deserve to be rewarded after working hard as well as with our apps. These apps will provide a very beautiful ecosystem well-prepared pool with fish, iris flower and more interesting and cute decoration for you to decorate the pool. Once you have completed your assignment, the Pond application will reward you for decorating your pool nicely. The outcome of this application allows users to monitor their progress in the iteration of their plans.

1. Introduction

1.1 Problem Statement

It is understood that motivation is a crucial push every student, worker or even an ordinary person needs to accomplish a certain task. Not everyone can easily become well oriented and diligent. Sometimes it is known as a very big step of change for certain people in creating new roles for themselves, new responsibilities and new challenges that need to be faced. One of the most common problems is that many people have struggled into reaching their new goals sometimes not knowing which step to take first or what task should be accomplished. In order to embrace these new changes, it is best to always start organizing their plans in any form that they are comfortable with such as a to-do-list, or even a calendar reminder. This is usually a difficult task because most people lack the discipline onto themselves which will lead to zero motivation into completing or even trying to achieve the required goals. Sometimes, getting rid of a certain bad habit such as nail biting, lack of water consumption or even a bad habit of smartphone usage causing late night sleeping can also be hard to quit. Of course, most people would have even tried starting in order to replace the bad habits with good ones, sometimes simply trying to remember in taking their prescription medications.

Unfortunately, even with a new plain planned schedule or timetable may not be strong enough to provoke self-motivation or it might be too boring to even get a head start. Moreover, after already making progress, keeping in check with constant push and updates is also not easy as sometimes they require an award after a completion. Not only that, some applications are also quite hard to keep track of one's progress especially if the user gets too lazy into updating it every day. This is why the created personal motivator application will change all these negative reviews into positive ones by having creative and attractive features that will be available for all kinds of platform purposes such as business plannings, habit checker, calendar checklist, a medication reminder and more! This may reduce the boredom of staying on one application after completion as there will be a digital reward after a certain progress is done which will be a completed digital ecosystem that is created by the user itself. This personal motivator application will be able to change many lives and create new possibilities.

Goals

The goals of this project are as follows:

1. Create a pleasing and attractive user interface design.
2. Successfully construct a good login and registration system.
3. Make the application as user friendly as possible.
4. Develop a fully functional application.
5. Make the application able to support cross-platform using cloud technology.

Objectives

The objectives of this project are as follows:

- Able to aid and support users to successfully reach their plans and aspirations by themselves.
- Cultivate discipline in users by not putting much pressure.
- Teach users to be well-organized in a way that is entertaining for them.
- Create a mindset that nothing is impossible.
- Draw out the true potential of a user.

Outcome

1. Application

- a. Application to perform essential daily tasks in a more organized manner.
- b. Functions such as calendars, to-do lists, note-taking, organizing lists for categories such as business plans, project planning and reminders are available.
- c. User-friendly application with pleasant and cute interaction design to attract users.
- d. Give rewards to decorate their pond nicely when they accomplish their tasks.
- e. Multi-platform applications can be used on Android, iOS and Windows.
- f. Using cloud technology to store the data.

2. User

- a. Users can login with email.
- b. Insert their to-do list for a day.
- c. Set appointments in the calendar as reminders.
- d. Deadlines set to ensure in the calendar as reminders.
- e. Customize their routine activities.
- f. Write a self-reflection at the end of the tasks to reflect their activities.
- g. Decorate their pond nicely as they complete their tasks.
- h. Edit, save and delete the data that they insert.
- i. Provide behaviour impact and become more well organized after using the application.

1.2 Requirement Specification

1. User :

- a. Manage user account
 - i. Update, View – Account
- b. Task Management
 - i. Users are allowed to create, complete and edit tasks.
 - ii. Unlock tiers and rewards after task completion.
 - iii. Points will increase after each completion.
 - iv. Users may create more than one task.
- c. Create Pond
 - i. Users can create their pond with gamification characters.
 - ii. Can increase characters by unlocking tiers and buy at the “Pond Shop”.
- d. Ranking
 - i. Users can view points and ranking at the Hall of Fame section.
 - ii. They will receive a badge after achieving each mission.

1.3 Needs, Approach, Benefits, Competitor (NABC)

Elements	Description
Needs	<p>The need we are covering is to provide a creative and more active approach to routine and schedule management, as well as giving ample encouragement throughout the process so that clients are able to get in track of their regiment better. This can help people to be able to grow and improve themselves for the better in the long run and thus bringing a positive impact in their lives. Our target audience will be people of all ages and is intended as a personal use rather than collaboration.</p>
Approach	<p>For schedule management, we will provide a calendar which enables users to view their routine. Besides that, users are also able to customize features such as reminders from the settings section. The growth and progress of a plan is visualized as a pond that will get prettier as each goal is achieved, rather than just reminders that will come and go. When confirming that a task has been made, users need to submit a short reflection of how the task went. This allows them to summarize their journey better later when they have reached their goal, as well as to make them feel somewhat guilty and unaccomplished for cheating.</p>
Benefit	<p>Visuals play an important role in influencing an individual. By visualizing the progress, users are able to see where they are better. If they are following the plan well, they will see that every progress matters even if it is small. This also brings boredom out of managing tasks and the stress of trying to achieve something. Users will feel rewarded and satisfied upon unlocking new things.</p>
Competitors	<p>Our current competitor is the Forest app. To make our application different from Forest, instead of only being able to monitor phone screen time we make it more flexible and multipurpose by allowing users to customize their own routine to suit their own needs.</p>

1.4 Functional Requirements

- a. The system shall enable the user to view the Pond application.
- b. The system shall enable the user to manage task features at the application.
- c. The system shall enable the user to create more than one of their own tasks on add tasks according to their submission.
- d. The system shall enable the user to edit the tasks and change the title or dates at the task bar according to their suitability.
- e. The system shall enable the user to complete the tasks at the complete button according to their completion time.
- f. The system shall enable the user to view the updated or keyed in tasks at the task management feature.
- g. The system shall enable the user to view their current Pond points according to the completion of each task.
- h. The system shall enable the user to unlock tiers and characters based on each goal level such as Tier 1, Tier 2, Tier 3, Tier 4 and Tier 5.
- i. The system allows the user to buy the characters of each tier according to the points earned after completing the tasks.
- j. The system allows the user to decorate their Pond with draggable animated characters according to their achievements.
- k. The system allows the user to view the ranking points with other users at the Hall of Fame based on the amount of points and completion they have achieved through the completion of each task.
- l. The system allows the user to view and edit their profile information by clicking the “Profile” button at the navigation bar.

1.5 Non-Functional Requirements

- a. Pond shall be able to operate on the web browser, local host and mobile application.
- b. Pond shall display invalid login messages when users enter an incorrect email and password at the login interface within 10 seconds.
- c. Pond shall redirect the users to the respective main interface by clicking the “Continue” button after the users enter the correct email and password at the login interface within 10 seconds.
- d. Pond will update the task management after users have created a new task by displaying it as the latest input after 10 seconds.
- e. Pond will update the tasks after users click the “Complete” button redirecting it to delete the completed task after 10 seconds.
- f. Pond shall be able to rearrange the position of tasks according to the amount of created new tasks from the oldest to the latest.
- g. Pond shall be able to generate the updated points after users have completed each task by clicking the “Complete” button.
- h. Pond shall be able to generate connection of the points to the unlocking of new tier levels to access new draggable characters for the pond decoration after users have completed their tasks.
- i. Pond shall be able to generate the list of user’s points from the highest to the lowest score by connecting with tenenet at the leaderboards section.
- j. Pond shall be able to redirect the users to the respective login main page interface after the “Logout” button is clicked within 10 seconds.

1.6 Team Members and Roles

Name	Roles	Description
Aimi Binti Rusdi (B19EC0001) 	1. Project Manager 2. Application Developer	<ul style="list-style-type: none"> - Manage the production of delivered requirements - Plan and monitor the project development - Prepare any follow-up action recommendations. - Does front-end and back-end developer for the system function. - Ensuring a normalised data structure for the application. - Help other teammates with interactive design. - Ensuring application is accessible web based and mobile based through cloud computing using Run Cloud.
Mirhanieza Binti Matharuzaman (A18CS0106) 	1. UI/UX Designer 2. Main Developer 3. Application Developer	<ul style="list-style-type: none"> - Design the application layouts according to requirements. - Update the application. - Solve code problems. - Act as front-end and back-end developer for the system function. - Integrating data from various back-end services databases. - Ensuring accessible data using Run Cloud.

<p>Nuramyra Natasha Binti Ismailudin (B19EC0035)</p> 	<p>1. UI/UX Designer 2. Application Developer</p>	<ul style="list-style-type: none">- Write well-designed, testable, efficient codes using software development.- Create layout and user interfaces using standard HTML/CSS/PHP codes.- Integrating data from various back-end services databases.- Ensuring accessible data using Run Cloud.- Act as front-end and back-end developer for the system function.
---	---	---

2. Gamification Concepts

Gamification is the idea of incorporating game mechanics and methods of game design to involve and inspire individuals to accomplish their objectives. Gamification taps into the fundamental desires and needs of the impulses of users that revolve around the Status and Achievement concept. Pond productivity application helps users to gamify their daily routine and make their lives more productive as it assists users to manage their time more effectively. The use of conquering the gamification concept in POND is meant to remind users of the task they need to complete in order to accomplish their goal of completing tasks on time and achieving greatness. Moreover, the function of grasping the concept is to engage users to use POND.



1. Goal

The goal of POND gamification is to engage users in using this application by enjoying the various features provided. In the meantime, our application provides a checklist or to-do list that will be inserted by the users themselves. Completion of the tasks on time will show a reflection feature to remind the users of them achieving their goals in a certain timeframe. This allows users to be more productive while achieving the goal to decorate their pond ecosystem attractively.

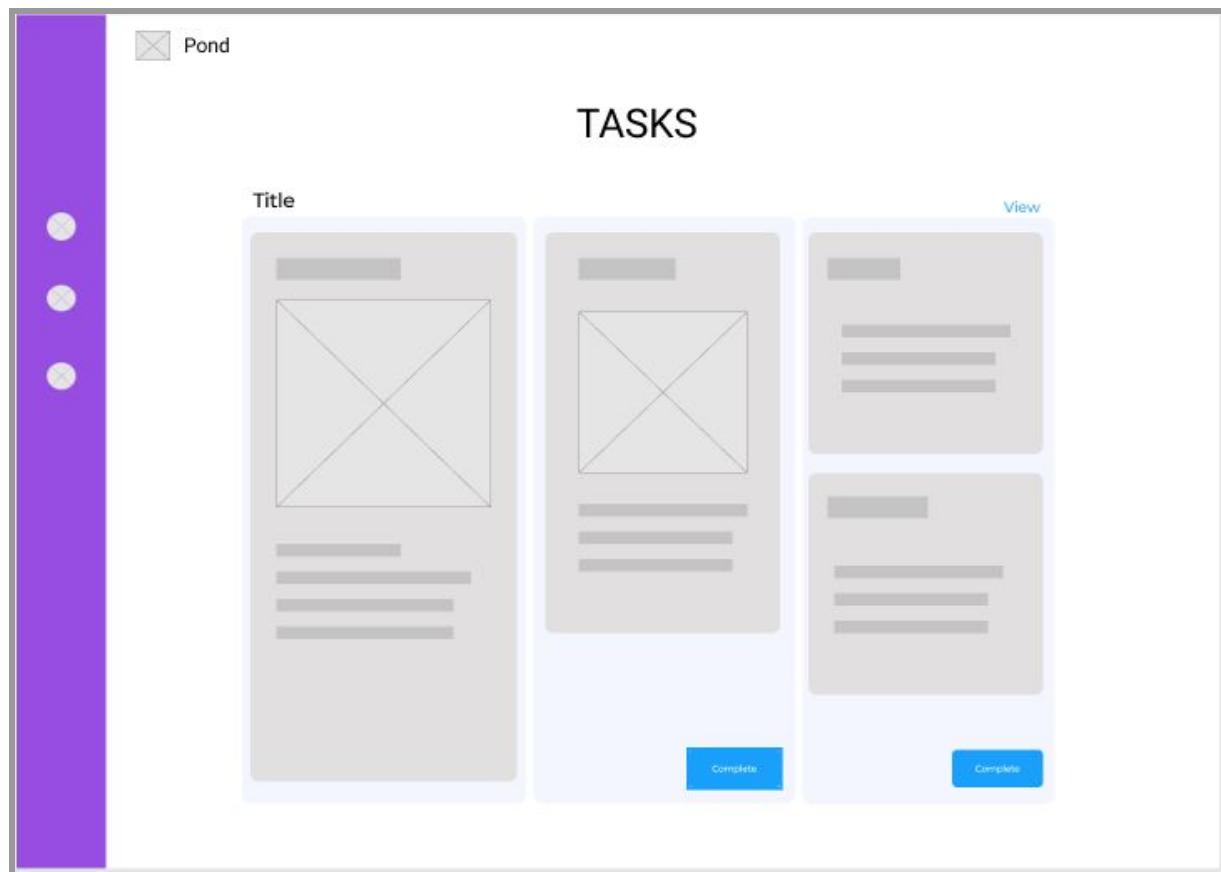


Figure 2.1.1 Task Screen prototype for desktop app

2. Achievement

For achievement, the application would include three types of badges. Each badge achieved will be attached on the user's profile. The first level rank badges will be dependent on how many goals are achieved over time and the badge will have a range of stars depending on their level such as Novice, Intermediate, Advanced and Expert. Second level is if users have a frequency of getting a spot in the Hall of Fame. User's will be rewarded a Ruby badge if they acquired a spot three times, Topaz for six times and if they continue to maintain and track their work perfectly, Diamond will be on the ninth time users have reached the Hall of Fame. Lastly, for the Special rank, an Overachiever badge will be rewarded after completing all previous achievements.



Figure 2.1.2 Rank Badges in application and profile display.

3. Challenge

In POND, we have features like the Hall of Fame where users who keep maintaining their ecosystem attractiveness will have a spot in the feature. To maintain it, the challenge is to remain aware of the deadline of their tasks. If the accumulated point becomes negative for not taking care of their tasks, it is possible that the pond ecosystem will become dirty. For example, the first degree which is 0 points will cause the pond to be dirty. If it becomes -50 points, the living things in the pond will fall ill. However, there is still a chance to heal or reverse the effect. But if it reaches the third degree where it becomes -100 points, the probability of the sick living things will die and they cannot be saved.

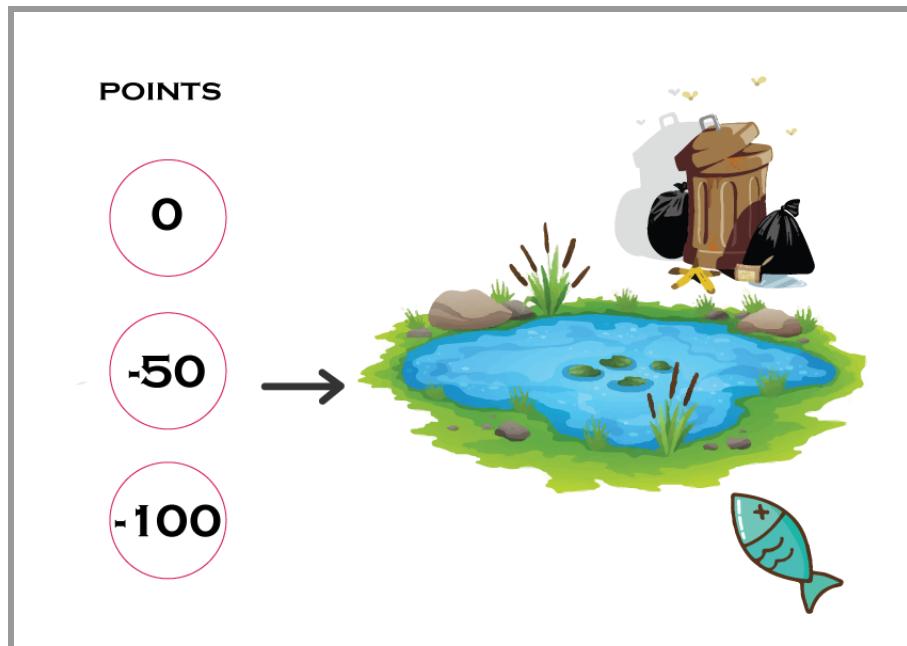


Figure 2.1.3 Point deduction and game effect

4. Reward

The concept of gamification for rewards to be used in POND is points that will be rewarded after the users accomplish their task to unlock five tiers. Each tiers will have their own specific point requirement. If the user completes their task on time 100 points will be awarded, however if the user completes the task earlier before the deadline 200 points will be rewarded. Moreover, what is more interesting is that there will be special tiers where the user can unlock many upgraded items for their pond with over 500 points. Users can collect all points to unlock tiers to get items from the Pond Shop, after that they can use those items to decorate their pond ecosystem using their creativity.

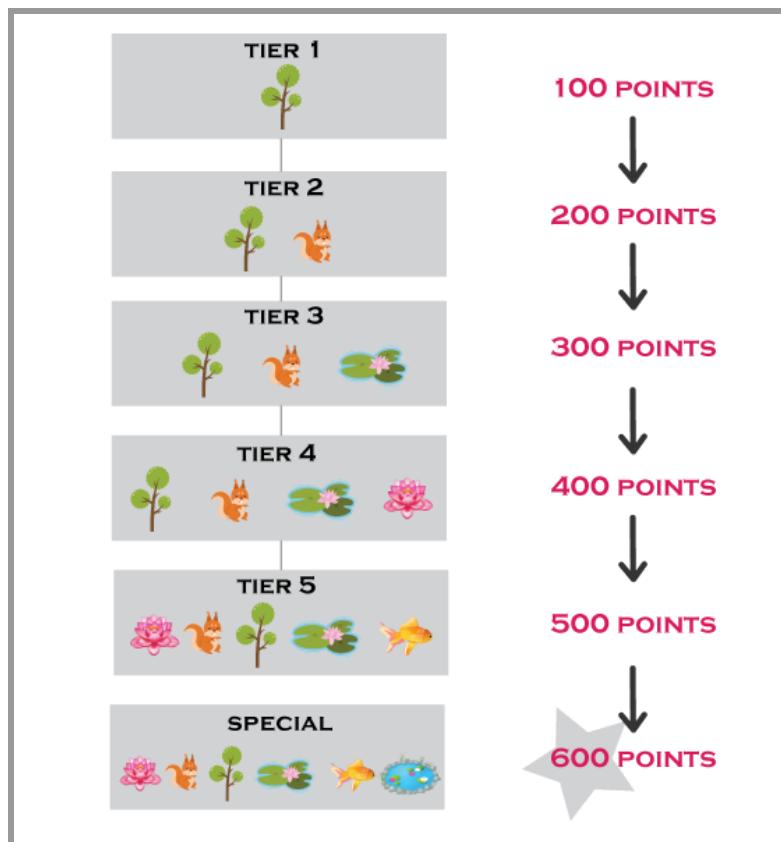


Figure 2.1.4 Tier rewards for item collection.

5. Competition

The concept of competition for gamification in POND will depend on the likes and views of the user's ecosystem publishing in the application. The pond app has the choice to publicize or private their pond creation. If users make their pond public, they have the opportunity to enter the Hall of Fame. Usually, users who earn badges at a certain level tend to have beautiful ponds and their pond will be displayed in the hall of fame. The Hall of Fame will only display top five user pond ecosystems.

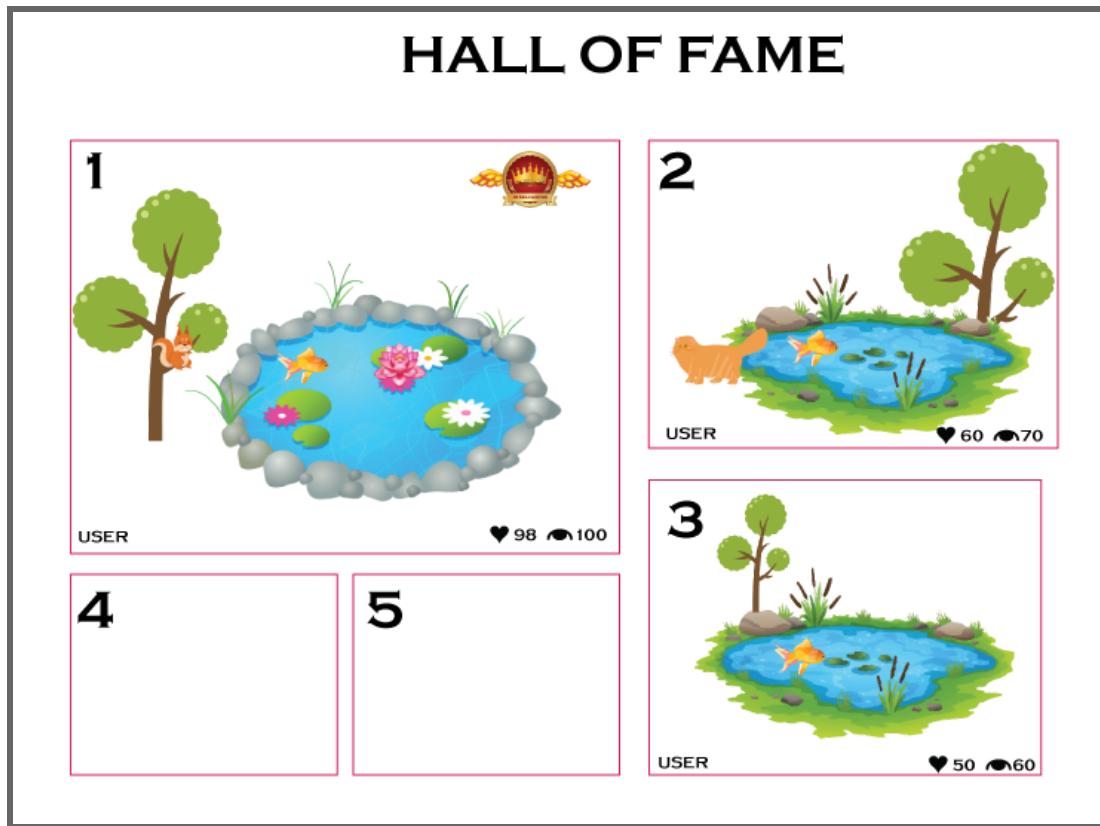


Figure2.1.5 Hall of Fame placing.

6. User engagement

The user engagement for the POND app is to ensure that users will continue to take care of their ponds as it is entertaining and appealing to have a pond ecosystem that has a variety of exciting lives such as flowers, fish, ducks and so on using e-platform. Of course, the layout of the Hall of Fame would engage the user's interest in using the application. A reminder will appear for the user and this will increase the engaging concept for the user.

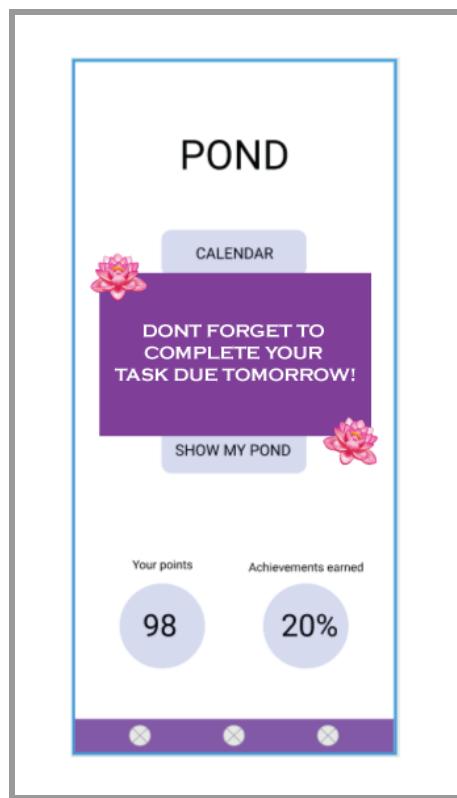


Figure 2.1.6 Notification of task prototype

7. Learning

In this application, user's will be able to learn to manage their time more effectively thanks to our applications notification features. Not only that, they are able to increase their creativity skills by decorating an ecosystem and learning from other user's success and reflections.

8. Improving Skill

Improving skill gamification concept in POND will refer to how users strive to improve their management time in doing their task activities. To increase the time to complete their task, they need to always complete it on time or before the deadline, they will be rewarded by unlocking each tier. Indirectly, they have improved their skills to complete all activities as short of time as possible and not procrastinate.

3. Specific Requirement Diagram

3.1 External Interface Requirements

3.1.1 User Interface Implementation

Logical characteristics: The GUI of the system is built using HTML, PHP, JavaScript and CSS based on the framework of Laravel 8 specifications. The framework is selected as the choice for this project in order to ensure the mobility and flexibility of the system's GUI to achieve an attractive interface. Pond application applied the scroll down feature which allows the application to stay in only one page containing all the features with only scrolling. A side navigation bar will be visible at all times on the left of the screen containing all the menu items. This shows easy access for the users to instantly reach their desired feature within one click such as tasks, pond shop and achievements.

Aspects:

1. When users key in the wrong email and username, the system will show which field contains an error and the status is updated and redirected to the main interface after users click the continue button.
2. The users will have to login or register beforehand to access the main interface of the system. Otherwise, access is not granted to public users.
3. Notification of the completion of task management is visible as a pop-up feature of the page. Users will be able to view the notifications and click the close button to redirect to the main page.
4. Draggable animated gamification characters will be accessible for the users to use to create their pond system when the users click and drag the characters.

3.1.2 Hardware Interface Implementation

It is stated that this system is supported by multiple hardware devices or technology, better known as cross-platforms that are commonly used nowadays to access an application such as a laptop or a personal computer with Windows 10 operating system for the best support. Smartphones are also accessible for mobile application use that can be installed or even with stable browsers such as Google Chrome or Internet Browser for mobile view if users choose to not download the application. The system will write the information directly to the user's database located in their own system account. The system uses a database on a cloud service known as Runcloud which is a network server as it receives data by using basic networking protocols. All the system's information is stored in the server's database such as task management and the link connection for points and gamification features with Tenenet.

3.1.3 Software Interface Implementation

Name	Mnemonic	Version	Description
Operating System	Windows 10	Mobile Enterprise	Windows operating system is the best support.
Browser	Google Chrome	81.0.4044.138	To retrieve the content of the web page, we used Google Chrome and Mozilla Firefox.
Browser	Mozilla Firefox	76	
Mobile	iOS	14.4	The version of mobile is best starting at the selected version.
Mobile	Android	7.1.0 - 7.1.2	
Database	MySQL	15.0	To save all records, we chose MySQL and NaviCat.
Database	NaviCat	15.0	
Programming Language	HTML	5.2	To create the website, we choose HTML.
Style Sheet Language	CSS	3	To describe the presentation of the document, we use CSS.
Framework	Laravel	8	Laravel as the main Framework to contain functionalities.

3.1.4 Runcloud Deployment

In this final application development project, we created the deployment process by using Upcloud in order to create our own server and added a new web application called “Pond” which is owned by user runcloud. This website uses a test domain. To explain further, using Runcloud will ease our access link by providing our own domain server link for the deployed website.

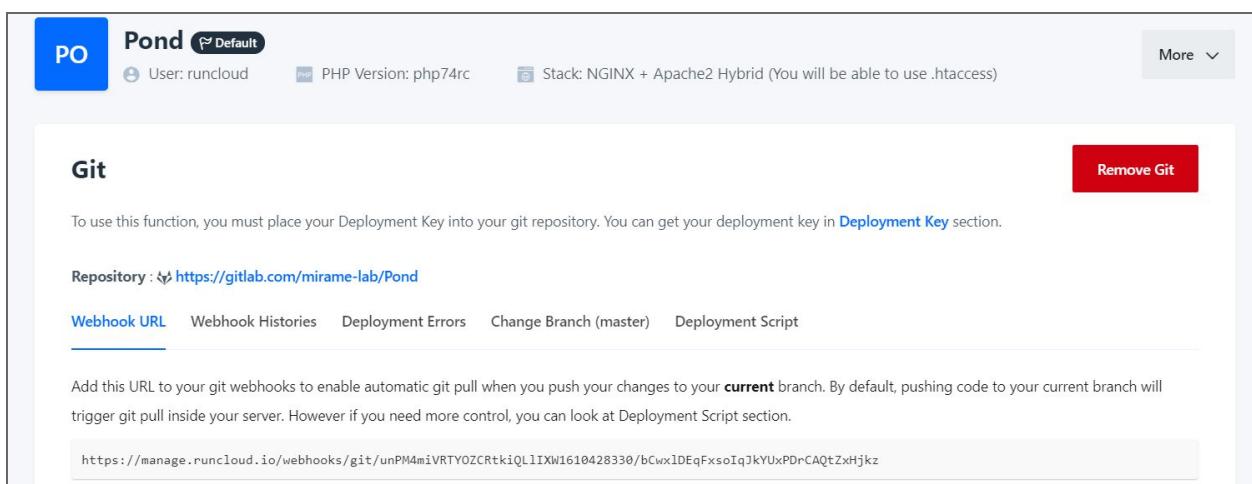


Deployment Key

To use Git Deployment inside a Web Application, you must use and generate a Git Deployment Key. Each user can only have **one** deployment key, and it can be used for multiple Web Applications (except you are using GitHub).

User	View / Generate
runcloud	
mirmssmsms	

Instead of transferring files using an ftp client, the website is connected to the repo link of the development website “GitLab”. This allows us to continue updating and make changes to the application without having any disruptions or manually transferring files every now and then. The steps into creating the repository link for our application is by selecting the “Git” option at the Web Application menu. The deployment key of the website is then added to the git repository, and the webhook is also set. The webhook url plays an important role on keeping track of the commits that have been pushed so that the website is always up to date with repository.



Pond

User: runcloud PHP Version: php74rc Stack: NGINX + Apache2 Hybrid (You will be able to use .htaccess)

Git

Use this function, you must place your Deployment Key into your git repository. You can get your deployment key in [Deployment Key](#) section.

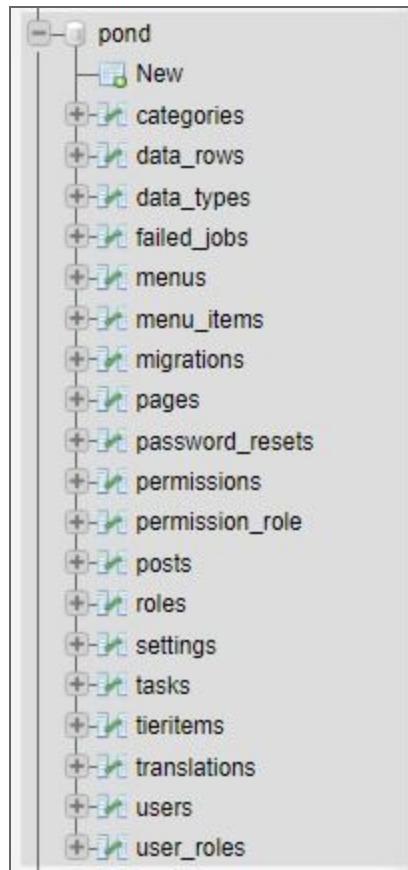
Repository : <https://gitlab.com/mirame-lab/Pond>

Webhook URL [Webhook Histories](#) [Deployment Errors](#) [Change Branch \(master\)](#) [Deployment Script](#)

Add this URL to your git webhooks to enable automatic git pull when you push your changes to your **current** branch. By default, pushing code to your current branch will trigger git pull inside your server. However if you need more control, you can look at Deployment Script section.

<https://manage.runcloud.io/webhooks/git/unPM4m1VRTYOZCRtkiQL1IXW1610428330/bCwx1DEqFxsoIqJkyUxPDrCAQtZxHjkz>

Remove Git



Not to mention, our application of course requires a database to store all the information and data of our application's process and identifications. For the database, we used phpmyadmin. The database name for our application that has been created is known as "pond". In order to synchronize the database and the website changes at the codings after every change, we had to also link the pond database to our Runcloud project. Lasly, Runcloud will provide our own domain link for the database which will then be redirected to our application.

Pond website link : <http://pond.ci85h8b6lh-xmz4qd1vw62o.p.runcloud.link/>

3.1.5 Tenenet Gamification Implementation

In this project, we were assigned to implement gamification concepts into our application to attract more users attention into using the created productivity application. We were introduced with the concept of using Tenenet as our gamification platform of creating players and game concepts. As for our application, the login and registration would require players information and how it needs to be sorted out. Therefore, using Tenenet, we were able to automatically detect the user or player, along with their points and rank in the application. To explain further on the overview of the gamification concept in our application, the points are gained or increased once a user creates a task or tasks and successfully completes the task at the right time to achieve the full assigned points. Therefore, the more tasks the user completes, the more points they are able to gain to reach the top place in the Hall of Fame of the Leaderboard. Through the amount of points gained, users will be able to unlock Tiers which consist of draggable animated characters that can be claimed to decorate onto the Pond canvas board.

In order to create and link the data to our project, we must firstly create and set each metrics and rules that is implemented in our project at the Tenenet platform. The first metric that was created is classified as “Points”. This metric functions on calculating the points gained by the user after every completed task. Next, “Tiers” metric was created as a way to calculate the points onto when a tier should be unlocked for the user to claim the items at the Pond Shop. Once these points are totalled up, the metrics for “Level” will determine the rank position of the user. Moreover, further connection of the metrics, it will then proceed with the implementation of the rules section. Below are the details of the metric and rules created for the gamification concept that is applied into our application by using Tenenet.net.

Metric	Id	Name	Description	Type	Value
Points	mainpoints	Points	The amount of points user receives	Point	
Tiers	tiers	Tiers	To unlock tier items	Set	Tier 1 - Special Tier
Completed Schedule	completed_points	Completed schedules	For each schedule completed	Point	
Level	rank_badge	Level	To achieve badge level for users	State	Level 1 - Level 4

Rule_id	Name	Achieve_metric	Requirements
tiers_rule	Tier 1 Rule	The player will achieve [Tier 1] from the metrics [Tiers].	The player should have the metric [Points] and it's value should be greater and equal to [100].
tier2_rule	Tier 2 Rule	The player will achieve [Tier 2] from the metrics [Tiers].	The player should have the metric [Points] and it's value should be greater and equal to [200].
tier3_rule	Tier 3 Rule	The player will achieve [Tier 3] from the metrics [Tiers].	The player should have the metric [Points] and it's value should be greater and equal to [300].
Tier	Tier 4 Rule	The player will achieve [Tier 4] from the metrics [Tiers].	The player should have the metric [Points] and it's value should be greater and equal to [400].
tier5_rule	Tier 5	The player will achieve	The player should have the metric

	Rule	[Tier 5] from the metrics [Tiers].	[Points] and it's value should be greater and equal to [500].
tierS_rule	Special Tier Rule	The player will achieve [Special Tier] from the metrics [Tiers].	The player should have the metric [Points] and it's value should be greater and equal to [600].
level1	Level 1 rule	The player will achieve [novice] from the metric [Level].	The player should have the metric [Completed schedules] and it's value should be greater and equal to [3].
level2_rule	Level 2 rule	The player will achieve [intermediate] from the metric [Level].	The player should have the metric [Completed schedules] and it's value should be greater and equal to [9].
level 3_rule	Level 3 rule	The player will achieve [advanced] from the metric [Level].	The player should have the metric [Completed schedules] and it's value should be greater and equal to [18].
level4_rule	Level 4 rule	The player will achieve [expert] from the metric [Level].	The player should have the metric [Completed schedules] and it's value should be greater and equal to [32].

3.2 System Features

The system features include the use case, domain model, architectural design and component diagram of the Pond task management system as shown in figure 3.2.1, figure 3.2.2, figure 3.2.3 and figure 3.2.4. Each use case of the system is also structured by elaborating on use case description, sequence diagram and activity diagram. There are four use cases which are managing tasks, claiming items, decorating ponds and viewing the Hall of Fame.

3.2.1 Use Case Diagram

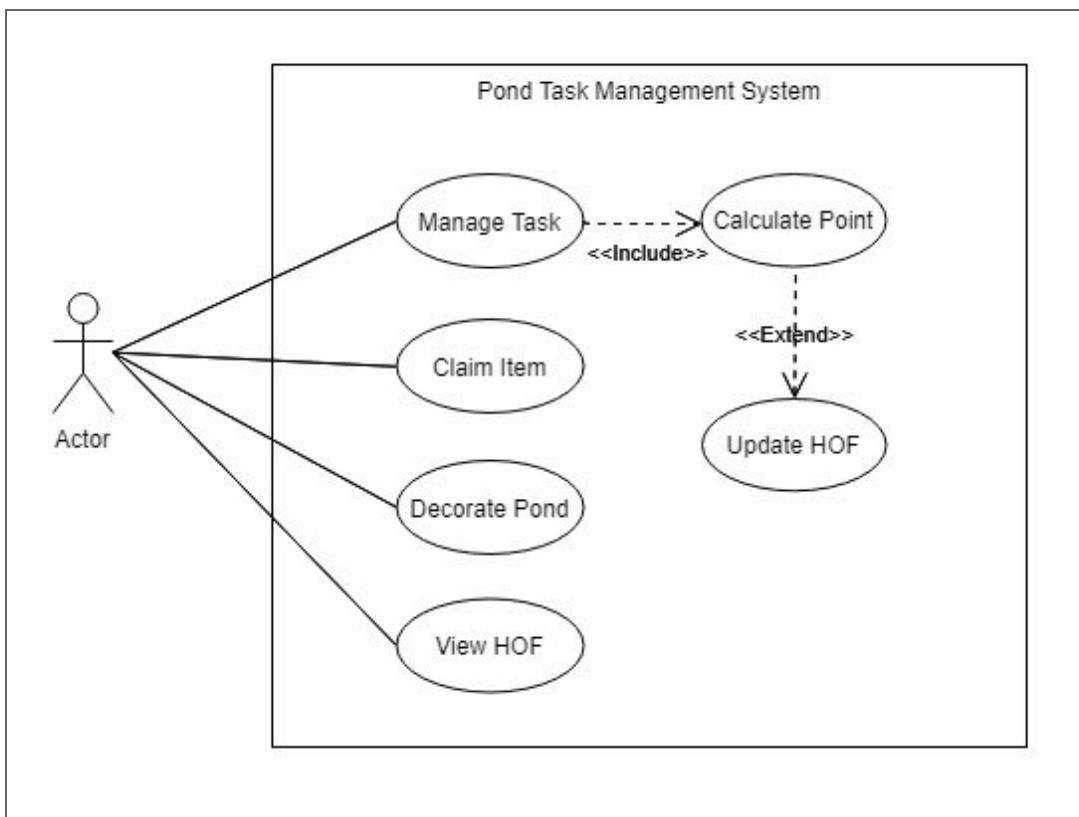


Figure 3.2.1 Use Case Diagram for <Pond Task Management System>

3.2.2 Domain Model Diagram

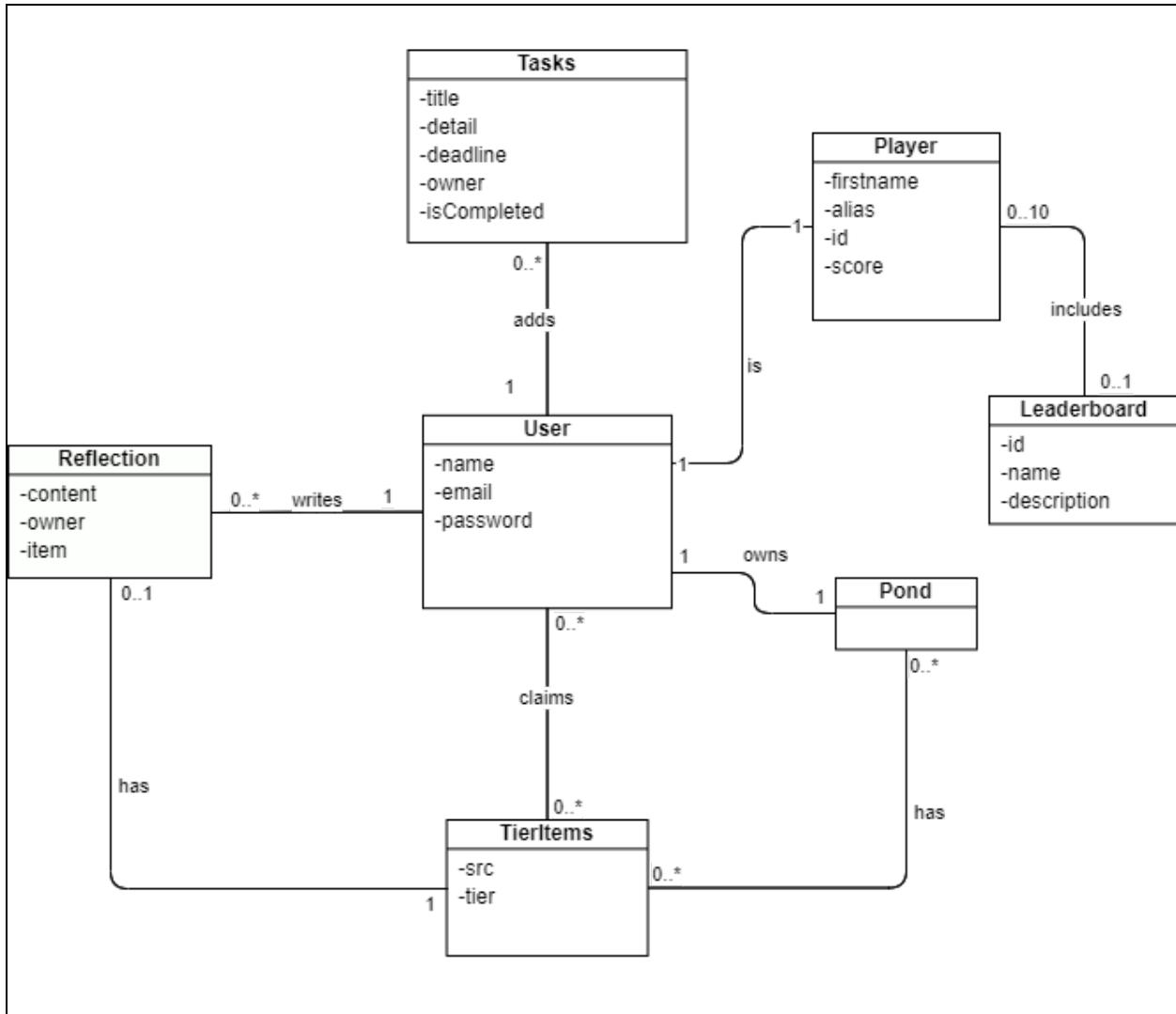


Figure 3.2.2 Domain Diagram for <Pond Task Management System>

3.2.3 Architectural Design Diagram

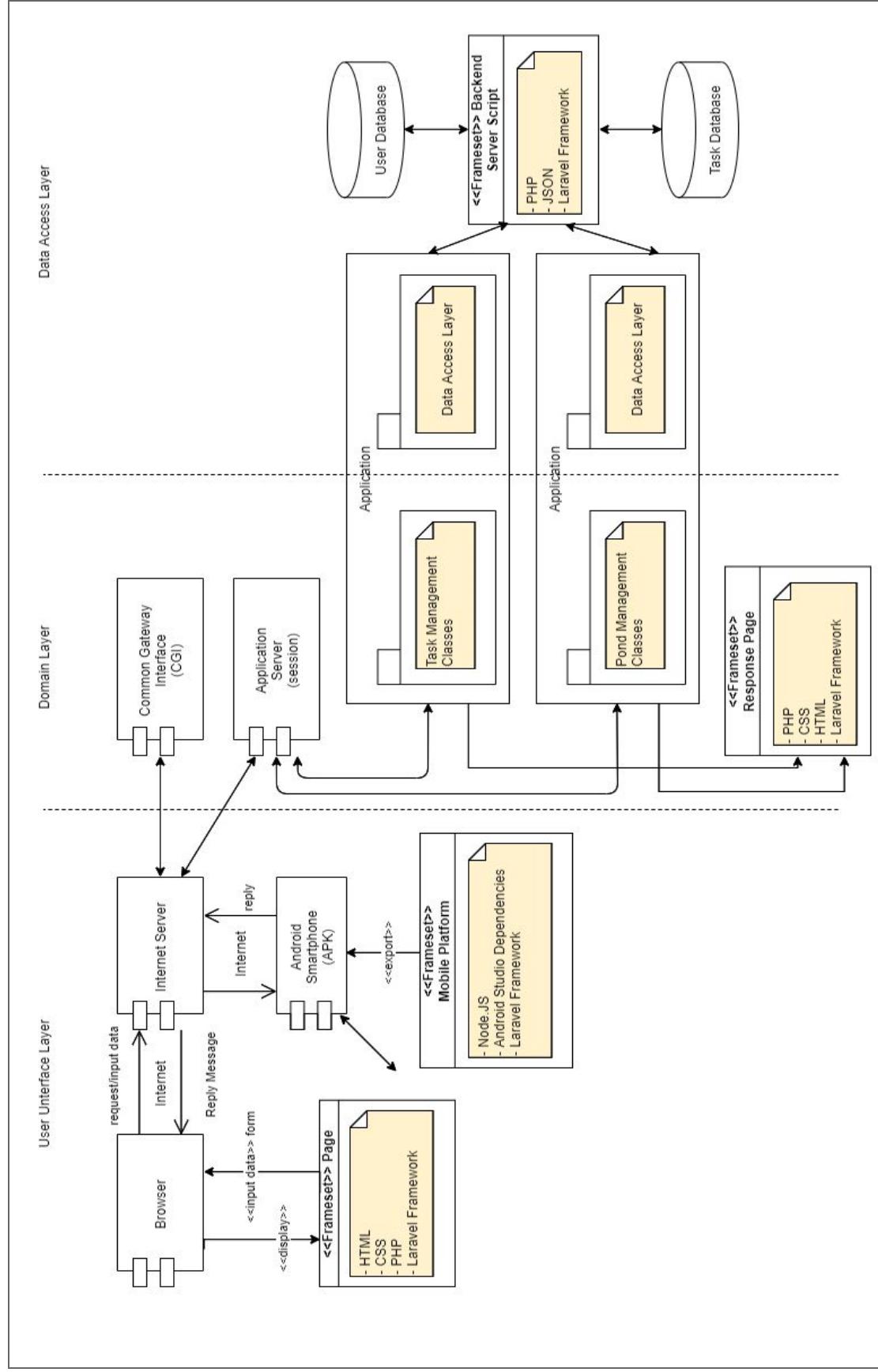


Figure 3.2.3 Architectural Design Diagram for <Pond Task Management System>

3.2.4 Component Diagram

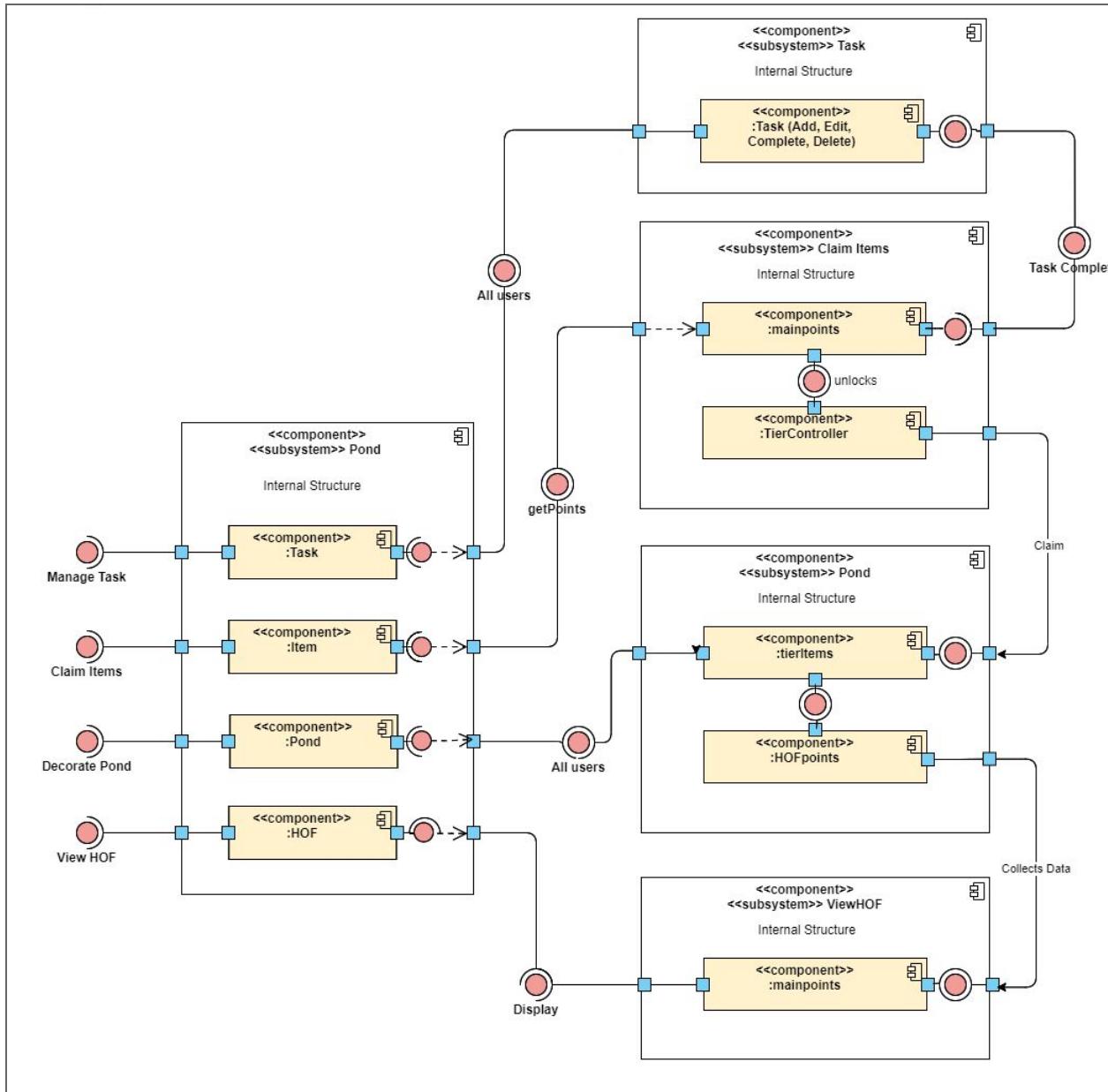


Figure 3.2.4 Component Diagram for <Pond Task Management System>

4. Subsystem 1 : Task Management

4.1 Specific Requirement

The use case for the first subsystem which is the task management subsystem, covers the function of managing the tasks for the users. The use case includes functions of connecting the flow with calculated points for update which will extend the process to the Hall of Fame to update the current points of the users.

4.2 UC1 : Use Case <Manage Task>

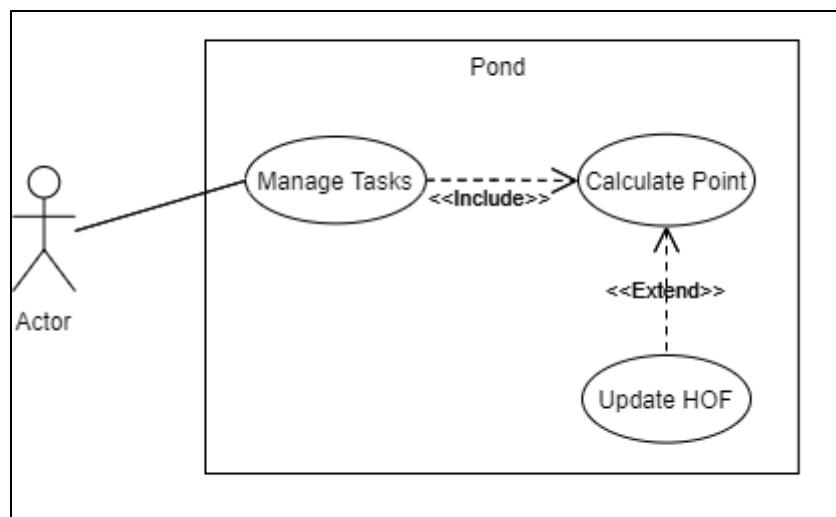


Figure 4.2 Use Case Diagram of <Manage Task>

4.2.1 Use Case Description

Use Case : Manage Task	
ID : UC1	
Actors : User	
Pre-conditions	

1. User has registered
2. User has logged in
3. User has active internet connection

Normal Flow

1. Use case begins when the user navigates to the Tasks section.
2. Pond will display the user's progress and list of Tasks cards.
3. If the user clicks the "Add Task" button, Pond will display a pop up form to create a new task card.
4. If the user clicks the "Edit" button in the task card, Pond will display a pop up form for the users to edit the current task.
5. If the user clicks the "Delete" button in the task card, Pond will remove the task.
6. If the user clicks the "Complete" button in the task card, Pond will change the task's status to "Completed" and <Calculate Points> will be executed.
7. Use case ends.

Alternative Flow

1. User's points exceed the points of any user in the Hall of Fame after <Calculate Points> is executed.
 - 1.1. <Update HOF> will be executed.
 - 1.2. User's name will appear in the Hall of Fame.

Exception

-

Related Requirements

-

Post Conditions

Changes to the tasks are successfully made.

4.2.2 Sequence Diagram

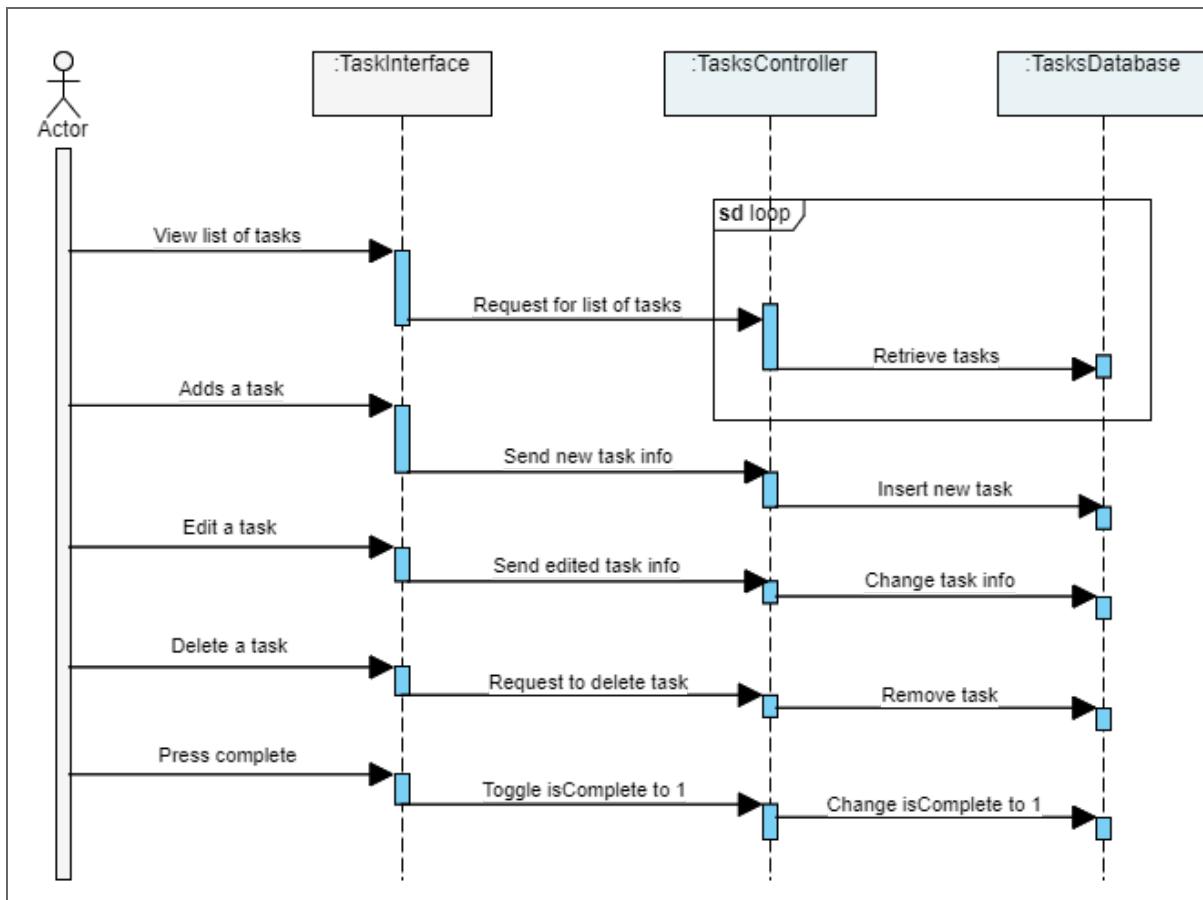


Figure 4.2.2 Sequence Diagram of <Manage Task>

4.3 Data Design

4.3.1 Data Description

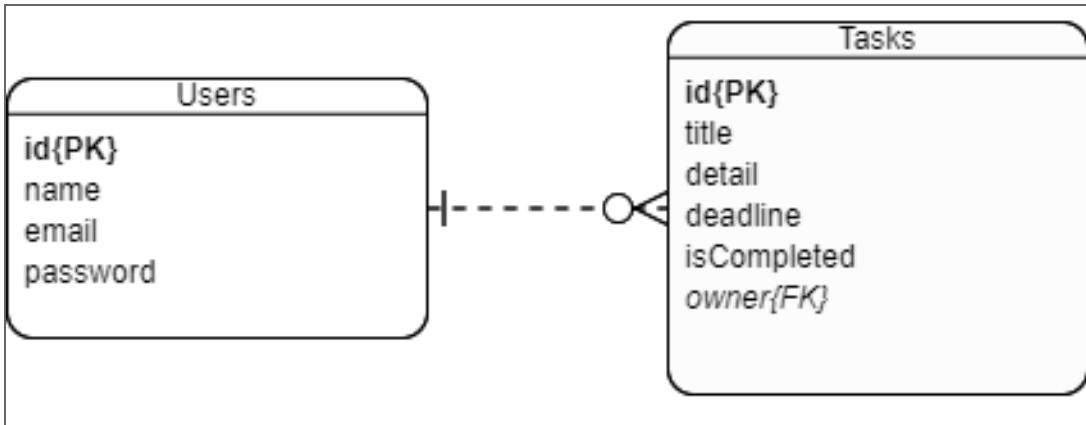


Figure 4.3.1 Entity Relationship Diagram (ERD) of Task Management

4.3.2 Data Dictionary

Name	Description	Occurrence
Users	Information of the users that have created an account in Pond	User managing the tasks
Tasks	Details of each task that the user added	Each tasks that were added are shown

Name	Multiplicity	Relationship	Name	Multiplicity	Description
Users	1	Manages	Task	0..*	Users can manage multiple tasks whereas a task can only be managed by one user.

Entity Name	Attribute	Description	Data Type	Constraint
Users	id	Unique id for the users	BIGINT(20)	PRIMARY KEY
Users	name	Username of the user's account	VARCHAR(191)	NOT NULL
Users	email	Email of the users	VARCHAR(191)	NOT NULL
Users	password	Password of the user's account	VARCHAR(191)	NOT NULL
Tasks	id	Unique id for the tasks	INT(10)	PRIMARY KEY
Tasks	title	Title of the task	VARCHAR(191)	NOT NULL
Tasks	detail	Description of the task	VARCHAR(191)	NOT NULL
Tasks	deadline	When the task should end	VARCHAR(191)	NOT NULL
Tasks	isCompleted	Indicates if task is completed or not	VARCHAR(191)	NOT NULL
Tasks	owner	Email of the user that owns this task	VARCHAR(191)	FOREIGN KEY

4.4 User Interface Design

4.4.1 Overview of User Interface

For the first subsystem, the interface is only accessible once the users achieve the first step which is by logging in or registering an account to enter. Once that is completed, users will be redirected to the main page of the application which will then visualize the navigation bar on the left to click on their desired menus or even scroll down to reach the task section. From there, users will be able to proceed with creating their tasks and completing them while increasing their progress percentage.

4.4.2 Screen Images

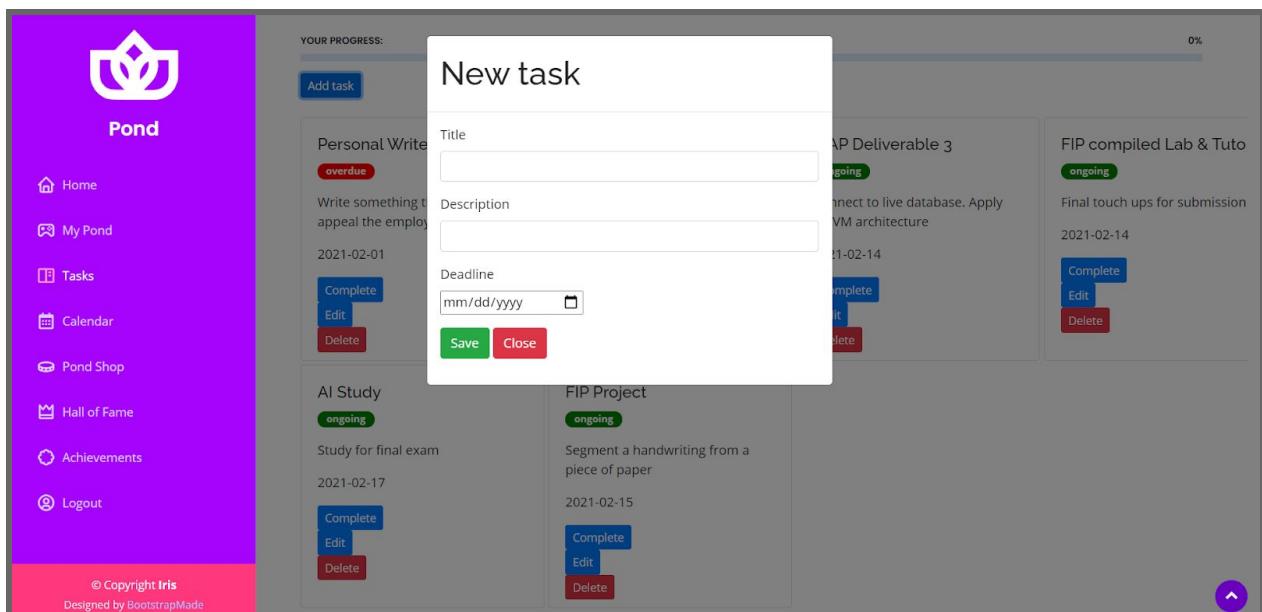


Figure 4.4.2.1 Generate “Add Task” button.

YOUR PROGRESS: 17%

Add task

Personal Write-up <small>completed</small> Write something that would appeal the employers 2021-02-01 <small>Complete</small> <small>Edit</small> <small>Delete</small>	POC (AI) <small>overdue</small> Make a short video. Modify the KR. Do report 2021-02-05 <small>Complete</small> <small>Edit</small> <small>Delete</small>	MAP Deliverable 3 <small>ongoing</small> Connect to live database. Apply MVVM architecture 2021-02-14 <small>Complete</small> <small>Edit</small> <small>Delete</small>	FIP compiled Lab & Tuto <small>ongoing</small> Final touch ups for submission 2021-02-14 <small>Complete</small> <small>Edit</small> <small>Delete</small>
AI Study <small>ongoing</small> Study for final exam 2021-02-17 <small>Complete</small> <small>Edit</small> <small>Delete</small>	FIP Project <small>ongoing</small> Segment a handwriting from a piece of paper 2021-02-15 <small>Complete</small> <small>Edit</small> <small>Delete</small>		

Figure 4.4.2.2 Overall view of Task section.

Edit task

Title: AI Study

Description: Study for final exam

Deadline: 02/17/2021

Save Close

Figure 4.4.2.3 Display Edit Task pop up

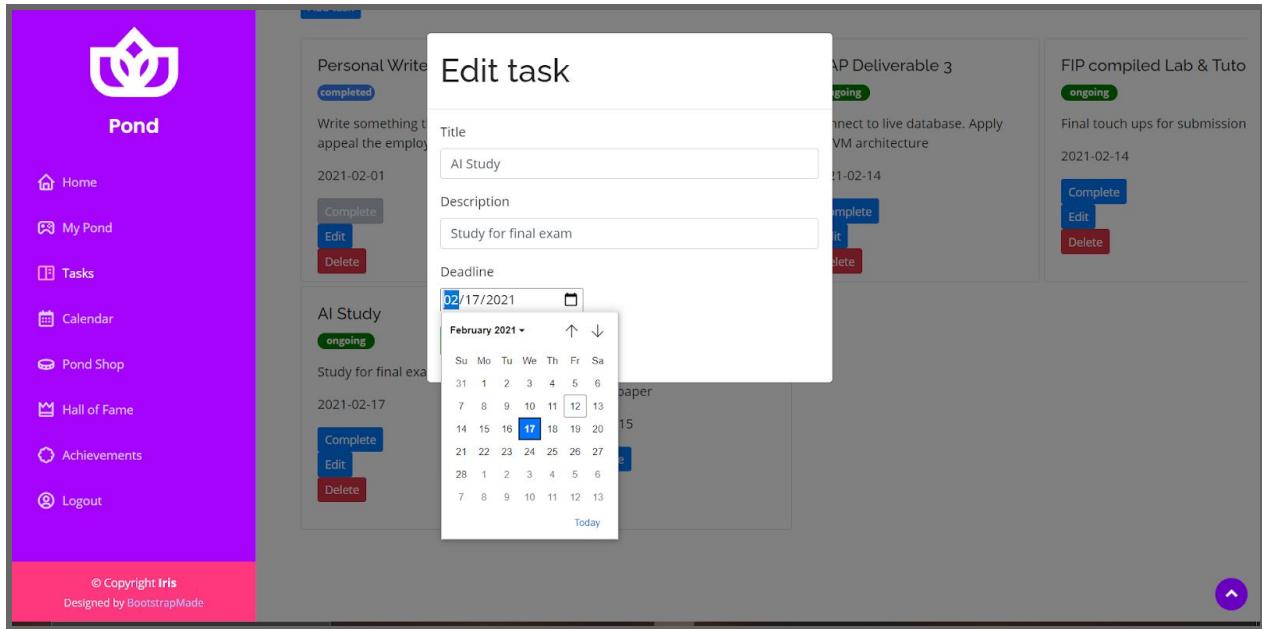


Figure 4.4.2.4 Select date at Mini Calendar at Task card

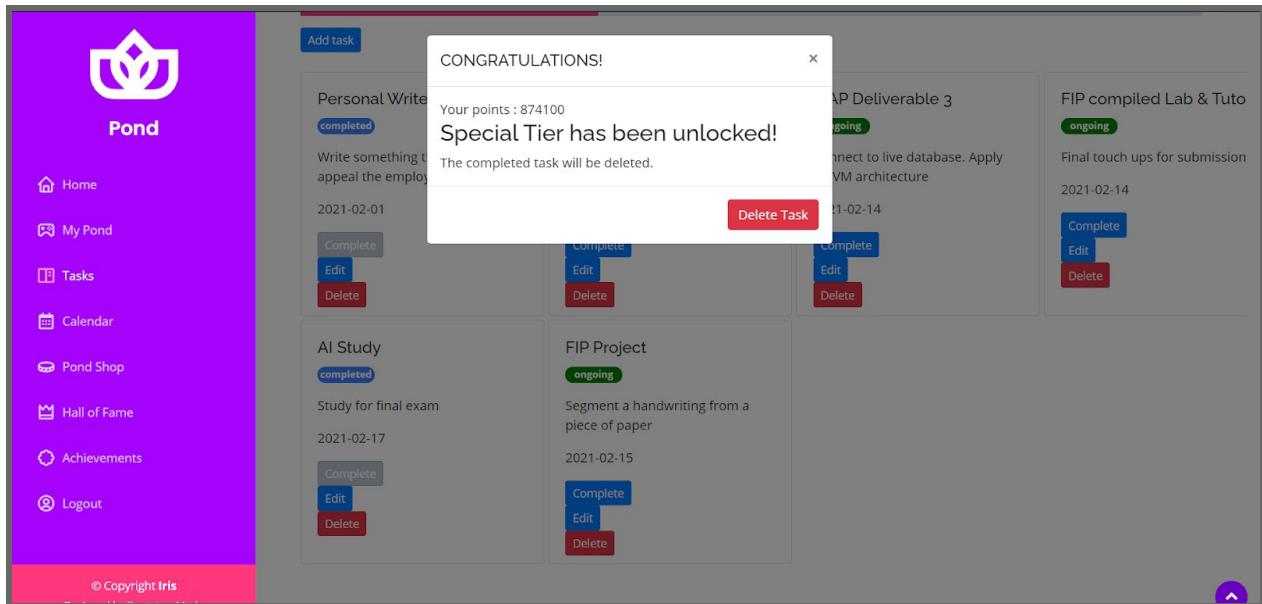


Figure 4.4.2.5 Update of tier after completing task

4.5 Test Cases

4.5.1 TC1_01 : Test <Scenario of Manage Task (UC1)>

Test Case ID	TC1_01	Test Case Description	Test the Manage Task functionality	
Created By	Aimi	Reviewed By	Mirhanieza	Version <input checked="" type="checkbox"/> 1.0
QA Tester's Log	Review comments from Mirhanieza incorporate in version 1.0			
Tester's Name	Aimi	Date Tested	February 9, 2021	Test Case [Pass/Fail/Not]
S #	Prerequisites:			
1	Stable internet connection.			
2	Access to Pond website or application.			
3	User logged in.			
4	User access to main page.			
S #	Test Data			
1	email = aimirusdi@gmail.com			
2	Pass = Abcd1234@			
3	TaskID = AD Submission			
4	Date = 14/2/2021			
Test Scenario	The user add and complete tasks.			
Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Navigate to Pond Application	Site should open		
2	Enter Userid & Password	Credentials can be entered		
3	Click Submit	Customer is logged in		
4	Click Add Task	Details can be entered		
5	Click Save	Task is created		
6	Click Complete	Progress complete and user may delete the task		

Figure 4.5.1 Test Case of <Manage Task (UC1)>

4.6 Test Approach Analysis

Test Case ID	Step#	Input Data	Valid	Invalid
TC1_01	6	Email, Password,	Valid email, Valid password	Invalid email, Invalid password, Null
		Add Task	Valid task details, Valid date	Invalid task details, Invalid date, Null.

5. Subsystem 2 : Pond Management

5.1 Specific Requirement

The use case for the second subsystem which is the pond management subsystem, covers three features of the Pond application for the users. The use case shows the activity that users will be allowed to achieve after completing the first subsystem that categorizes in claim items, decorate ponds and view the Hall of Fame.

5.2 UC2 : Use Case <Claim Item>

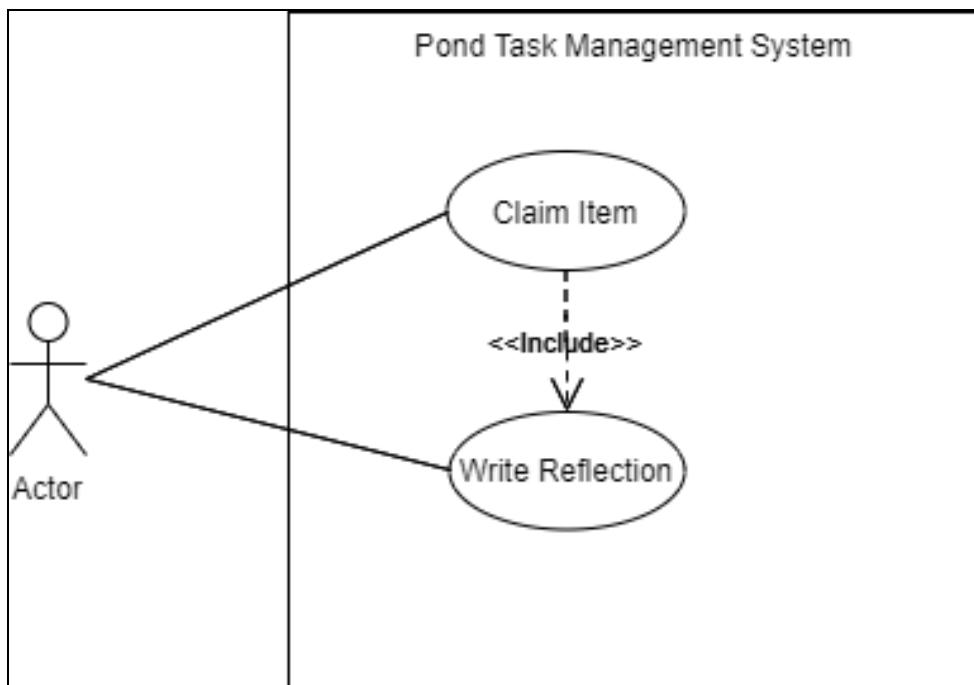


Figure 5.2 Use Case Diagram of <Claim Item>

5.2.1 Use Case Description

Use Case : Claim Item
ID : UC2

Actors : User
Pre-conditions <ol style="list-style-type: none">1. User has registered2. User has logged in3. User has active internet connection4. User has unlocked at least the first tier.
Normal Flow <ol style="list-style-type: none">1. Use case begins when the user navigates to the Pond Shop section.2. User will choose an item that has been unlocked.3. <Write Reflection> is executed.4. User writes reflection.5. User clicks on the “Claim” button.6. Use Case ends.
Alternative Flow -
Exception -
Related Requirements -
Post Conditions Items at each tier are successfully claimed.

5.2.2 Sequence Diagram

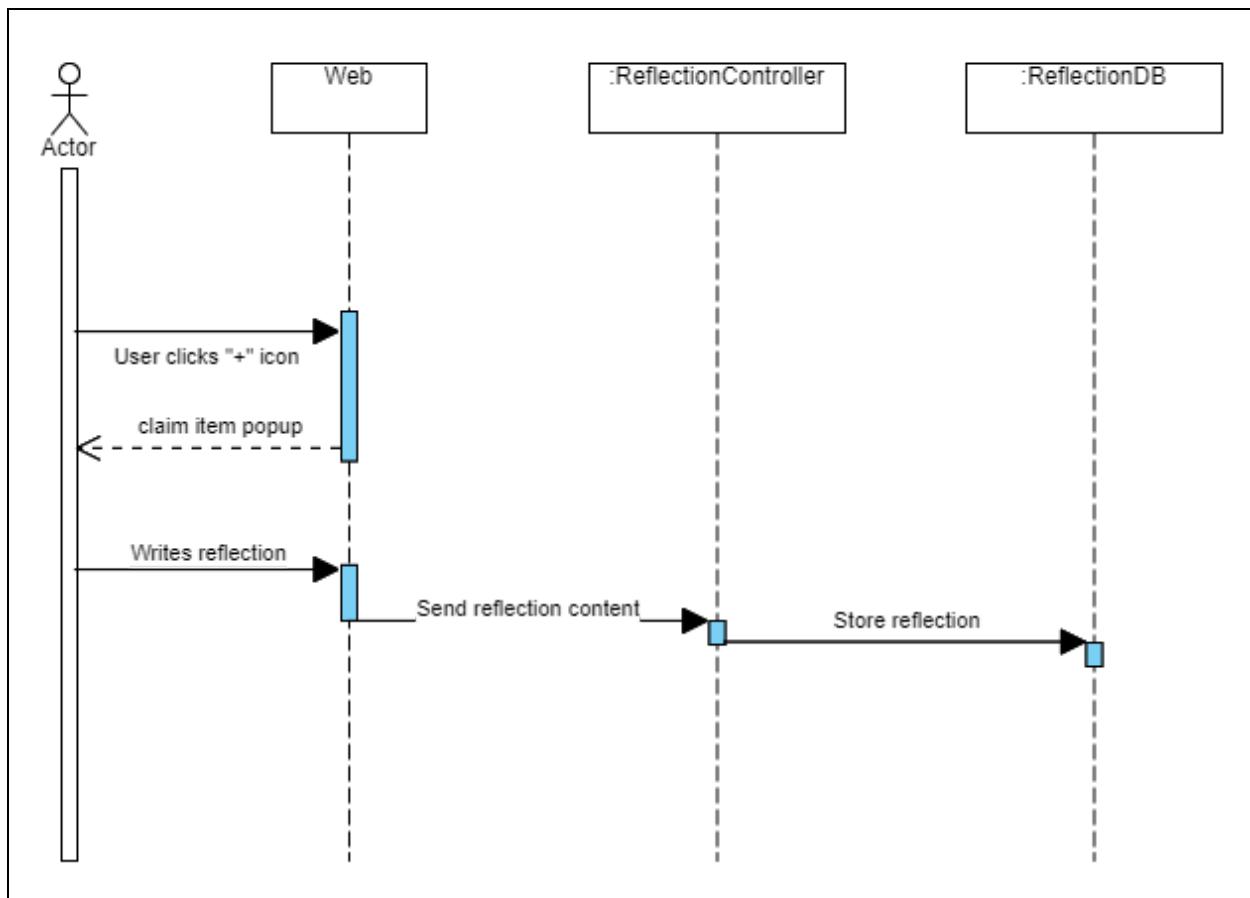


Figure 5.2.2 Sequence Diagram for <Claim Items>

5.3 UC3 : Use Case <Decorate Pond>

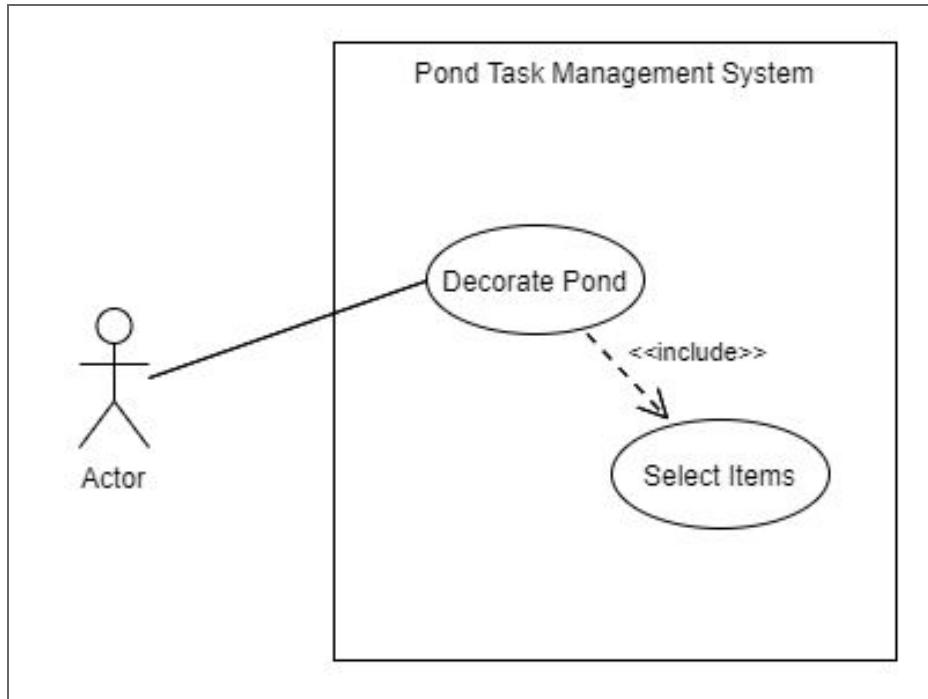


Figure 5.3 Use Case Diagram for <Decorate Pond>

5.3.1 Use Case Description

Use Case : Decorate Pond	
ID : UC3	
Actors : User	
Pre-conditions	<ol style="list-style-type: none">1. User has registered2. User has logged in3. User has active internet connection4. User has unlocked tier items.
Normal Flow	<ol style="list-style-type: none">1. Use case begins when the user navigates to the My Pond section.

<ul style="list-style-type: none">2. The user will be able to decorate the pond by claiming items after unlocking each tier.3. To decorate more, the user will select items by clicking the “Items” bar.4. The user will be able to select the items to decorate the pond.5. Use Case ends.
Alternative Flow -
Exception -
Related Requirements -
Post Conditions Pond can be decorated successfully.

- 2. The user will be able to decorate the pond by claiming items after unlocking each tier.
- 3. To decorate more, the user will select items by clicking the “Items” bar.
- 4. The user will be able to select the items to decorate the pond.
- 5. Use Case ends.

Alternative Flow

-

Exception

-

Related Requirements

-

Post Conditions

Pond can be decorated successfully.

5.3.2 Sequence Diagram

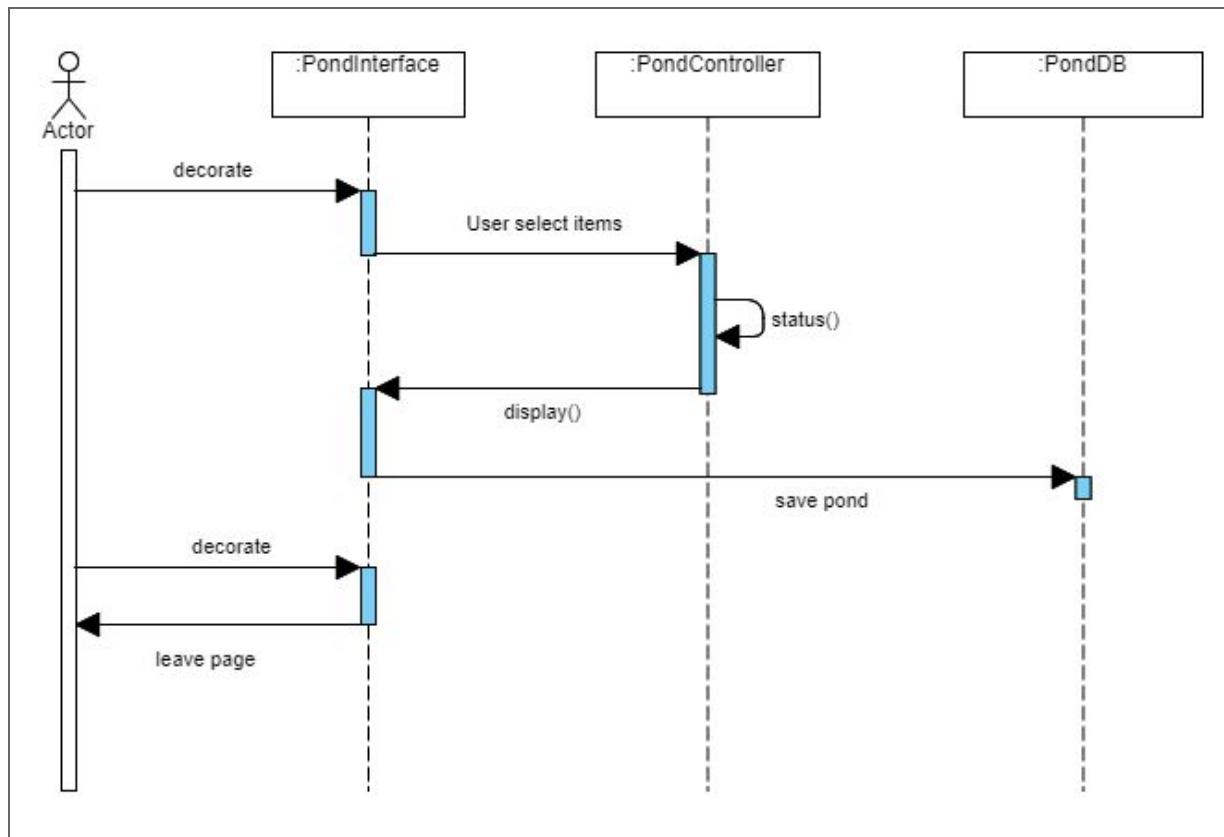


Figure 5.3.2 Sequence Diagram for <Decorate Pond>

5.4 UC4 : Use Case <View Hall of Fame>

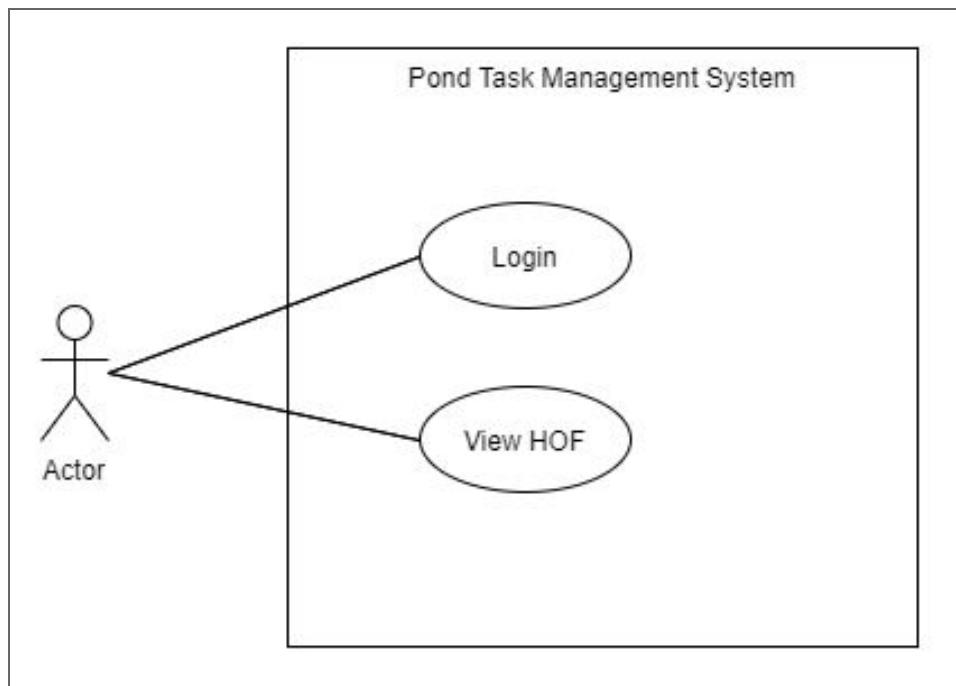


Figure 5.4 Use Case for <View Hall of Fame>

5.4.1 Use Case Description

Use Case : View HOF	
ID : UC4	
Actors : User	
Pre-conditions	
1. User has registered 2. User has logged in 3. User has active internet connection 4. User has completed the task.	
Normal Flow	

<ol style="list-style-type: none">1. Use case begins when the user navigates to the Hall of Fame section.2. The user will be able to view their points and position at the Hall of Fame ranking.3. Use Case ends.

Alternative Flow

-

Exception

-

Related Requirements

-

Post Conditions

Current points in the Hall of Fame are visible.

5.4.2 Sequence Diagram

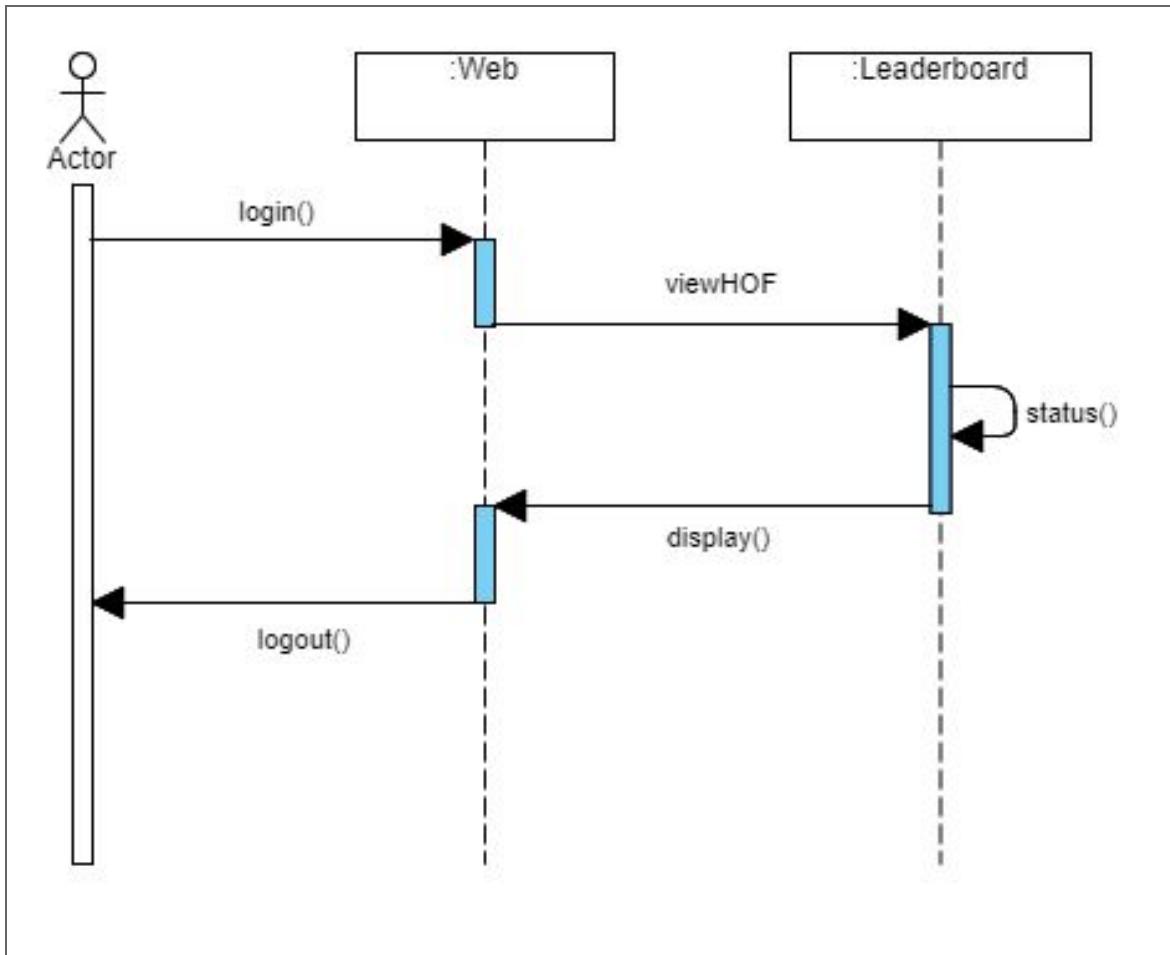


Figure 5.4.2 Sequence Diagram for <View HOF>

5.5. Data Design

5.5.1 Data Description

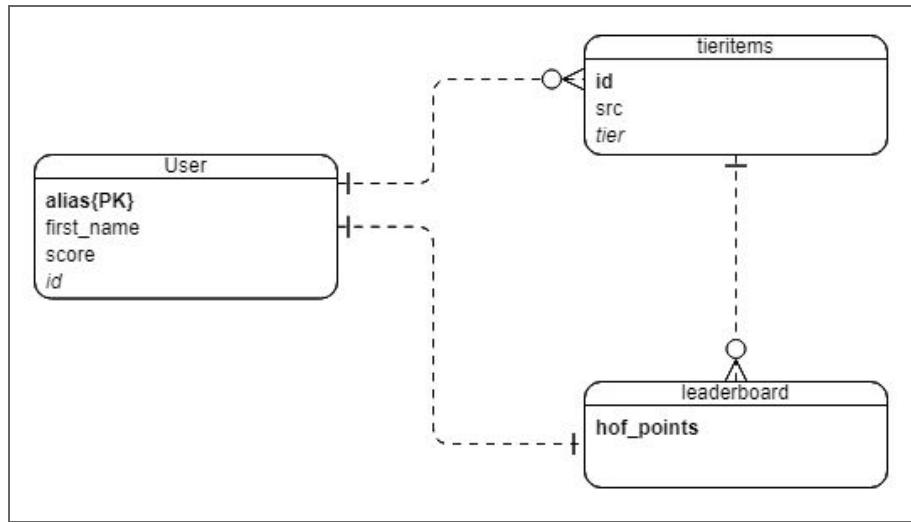


Figure 5.5.1 Entity Relationship Diagram (ERD) of Pond Management

5.5.2 Data Dictionary

Name	Description	Occurrence
Users	Information of the users that have created an account in Pond	User managing the pond
TierItems	Completed points collected to unlock tiers for users to claim items at Pond Shop	Each tier item unlocked can be claimed
Leaderboard	Information of calculated current points of users are updated	Current points are visible at Hall of Fame

Name	Multiplicity	Relationship	Name	Multiplicity	Description
Users	1	Unlock	TierItems	0..*	Users can unlock one task at a time and can claim more than one item.
Users	1	View	Leaderboard	1	Users can view one leaderboard which displays all users.

Entity Name	Attribute	Description	Data Type	Constraint
Users	id	Unique id for the users	BIGINT(20)	PRIMARY KEY
Users	name	Username of the user's account	VARCHAR(191)	NOT NULL
Users	email	Email of the users	VARCHAR(191)	NOT NULL
Users	password	Password of the user's account	VARCHAR(191)	NOT NULL
tierItems	id	Unique id for the tasks	BIGINT(20)	PRIMARY KEY
tierItems	tier	Detail of tiers	VARCHAR(191)	NOT NULL

5.6 User Interface Design

5.6.1 Overview of User Interface

The second subsystem's interface is accessible once the users have achieved the first step which is by logging in or registering an account. Once the first step is accomplished, users will be redirected to the main page of the application. After completing the tasks, points calculation will generate automatically. Once a certain mark is achieved, users will be able to unlock each tier and claim the tier items to decorate their pond. Users will also have access into viewing their current position and points at the Hall of Fame section of the application.

5.6.2 Screen Images

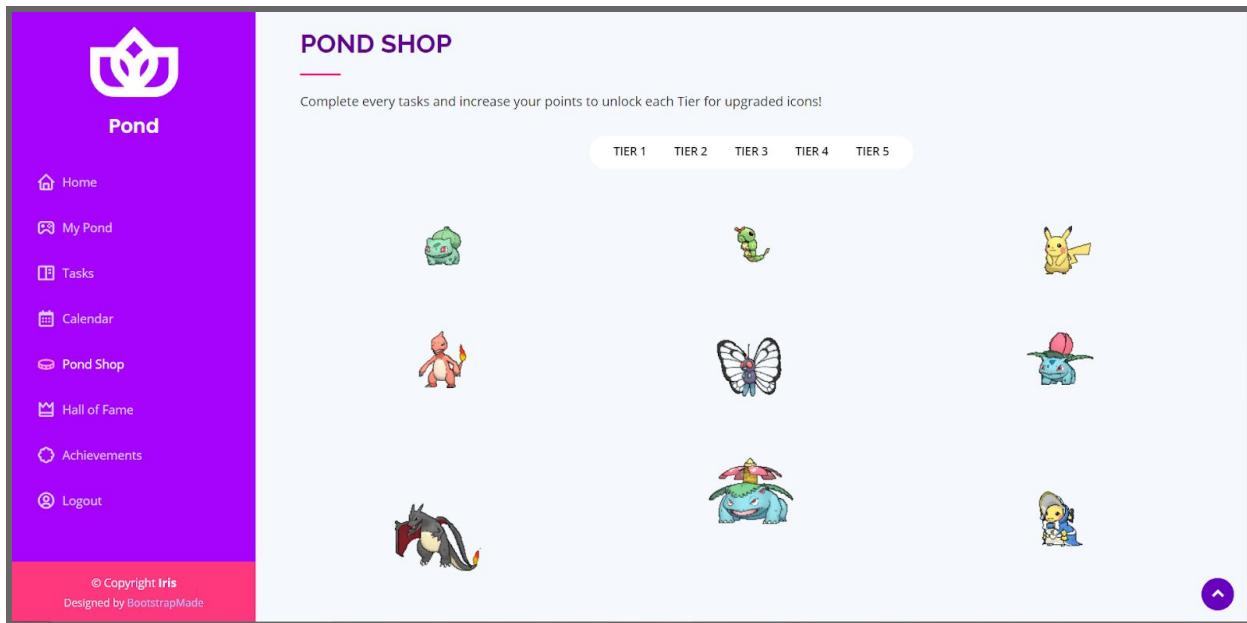


Figure 5.6.2.1 View of Pond Shop Tier Items

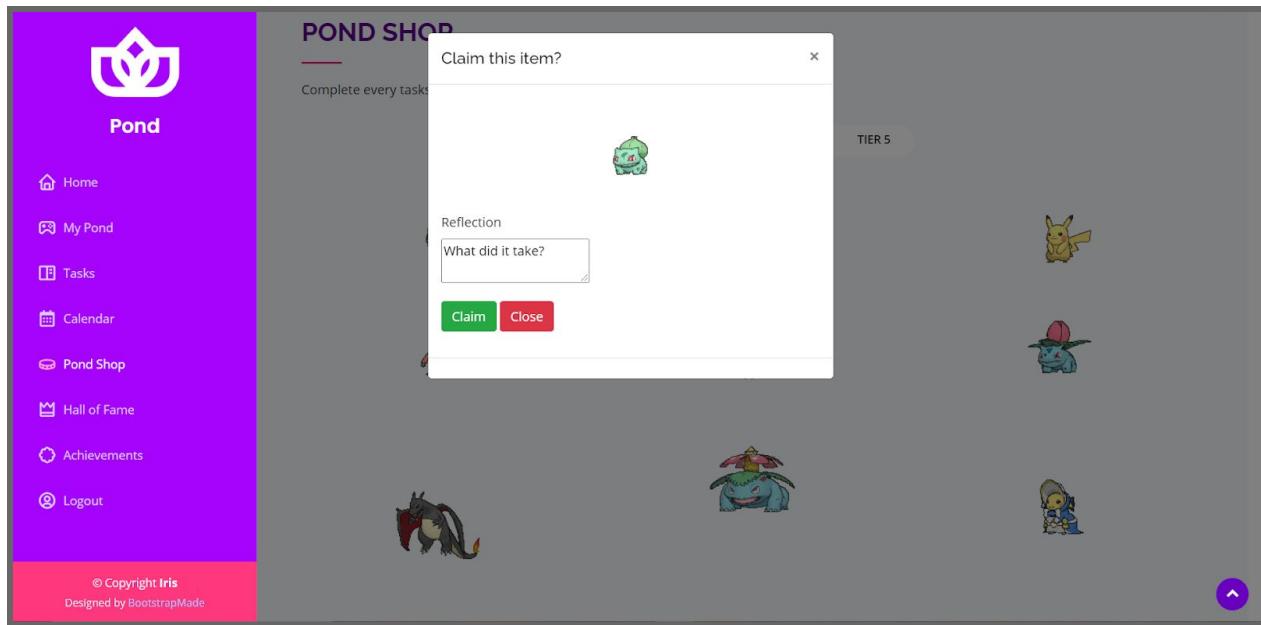


Figure 5.6.2.2 Page of claim item and write reflection

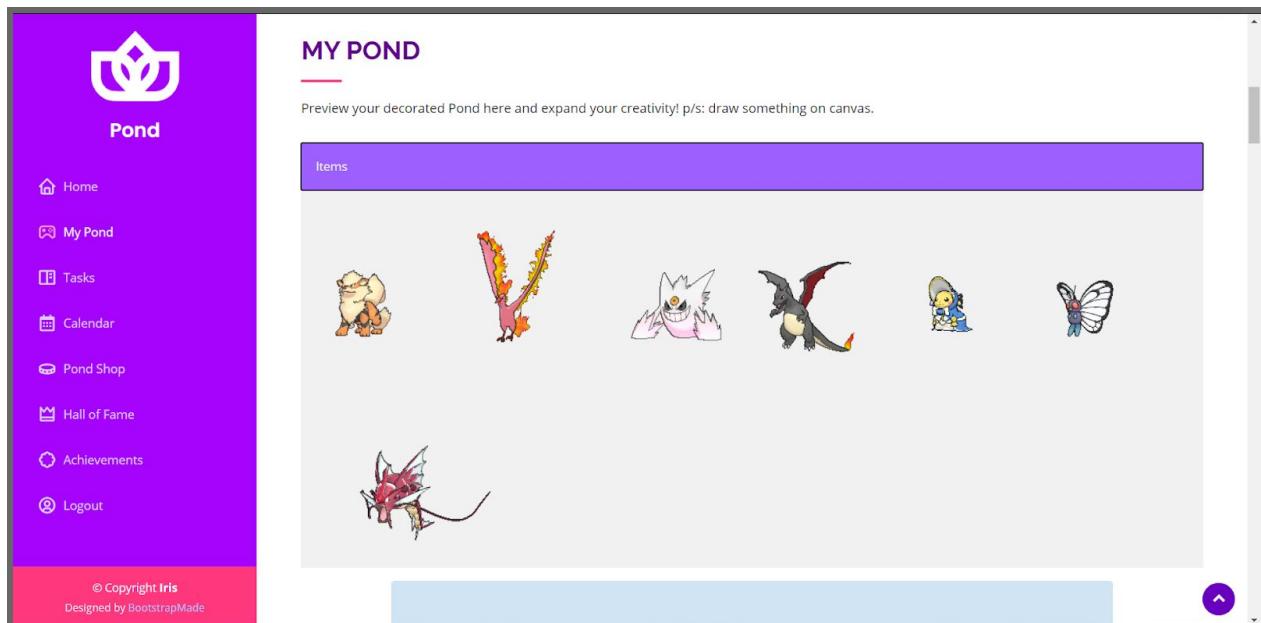


Figure 5.6.2.3 Claimed items retrieved at My Pond section

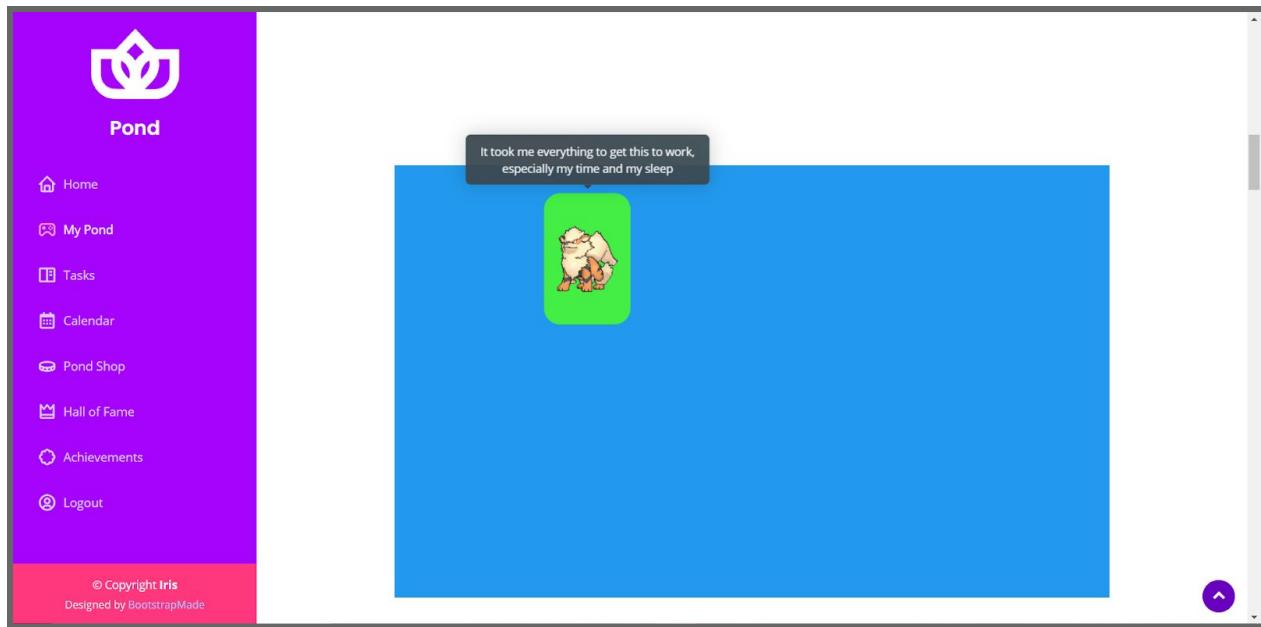


Figure 5.6.2.4 Reflection can be displayed at selected items

 A screenshot of the Hall of Fame page. The left sidebar is identical to Figure 5.6.2.4. The main content area has a purple header with the text 'HALL OF FAME'. Below the header is a sub-header 'Hall Of Fame'. The page contains a table with two columns: 'Players' and 'Scores'. The data is as follows:

Players	Scores
stardust	872500
amyr	59800
Aimi	54900
balqis	14900
Mirhanieza	8500
Ponder	4200
syhr	3200
Miraculous	1600
Textiedata	1100
Amyra Natasha	1000

Figure 5.6.2.5 List of Hall of Fame of users

5.7 Test Cases

5.7.1 TC2_01 : Test <Scenario of Claim Item (UC2)>

Test Case ID	TC2_01	Test Case Description	Test the Claim Item functionality										
Created By	Aimi	Reviewed By	Mirhanieza										
QA Tester's Log		Review comments from Mirhanieza incorporate in version 1.0											
Tester's Name	Aimi	Date Tested	February 10, 2021	Test Case (Pass/Fail/Not)									
<table border="1"> <thead> <tr> <th>S #</th> <th>Prerequisites:</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Stable internet connection.</td> </tr> <tr> <td>2</td> <td>Access to Pond website or application.</td> </tr> <tr> <td>3</td> <td>User logged in.</td> </tr> <tr> <td>4</td> <td>User access to main page.</td> </tr> </tbody> </table>				S #	Prerequisites:	1	Stable internet connection.	2	Access to Pond website or application.	3	User logged in.	4	User access to main page.
S #	Prerequisites:												
1	Stable internet connection.												
2	Access to Pond website or application.												
3	User logged in.												
4	User access to main page.												
<table border="1"> <thead> <tr> <th>S #</th> <th>Test Data</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>email = aimirusdi@gmail.com</td> </tr> <tr> <td>2</td> <td>Pass = Abcd1234@</td> </tr> <tr> <td>3</td> <td></td> </tr> <tr> <td>4</td> <td></td> </tr> </tbody> </table>				S #	Test Data	1	email = aimirusdi@gmail.com	2	Pass = Abcd1234@	3		4	
S #	Test Data												
1	email = aimirusdi@gmail.com												
2	Pass = Abcd1234@												
3													
4													
Test Scenario The user unlocks tiers and claim items.													
Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended									
1	Navigate to Pond Application	Site should open											
2	Enter Userid & Password	Credentials can be entered											
3	Click Submit	Customer is logged in											
4	Click Complete	Points calculated and tiers unlocked											
5	Click Items	Items can be claimed											

Figure 5.7.1 Test Case for <Claim Item (UC2)>

5.7.2 TC2_02 : Test <Scenario of Decorate Pond (UC2)>

Test Case ID	TC2_02	Test Case Description	Test the Decorate Pond functionality										
Created By	Aimi	Reviewed By	Mirhanieza										
QA Tester's Log		Review comments from Mirhanieza incorporate in version 1.0											
Tester's Name	Aimi	Date Tested	February 10, 2021	Test Case (Pass/Fail/Not)									
<table border="1"> <thead> <tr> <th>S #</th> <th>Prerequisites:</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Stable internet connection.</td> </tr> <tr> <td>2</td> <td>Access to Pond website or application.</td> </tr> <tr> <td>3</td> <td>User logged in.</td> </tr> <tr> <td>4</td> <td>User access to main page.</td> </tr> </tbody> </table>				S #	Prerequisites:	1	Stable internet connection.	2	Access to Pond website or application.	3	User logged in.	4	User access to main page.
S #	Prerequisites:												
1	Stable internet connection.												
2	Access to Pond website or application.												
3	User logged in.												
4	User access to main page.												
<table border="1"> <thead> <tr> <th>S #</th> <th>Test Data</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>email = aimirusdi@gmail.com</td> </tr> <tr> <td>2</td> <td>Pass = Abcd1234@</td> </tr> <tr> <td>3</td> <td></td> </tr> <tr> <td>4</td> <td></td> </tr> </tbody> </table>				S #	Test Data	1	email = aimirusdi@gmail.com	2	Pass = Abcd1234@	3		4	
S #	Test Data												
1	email = aimirusdi@gmail.com												
2	Pass = Abcd1234@												
3													
4													
Test Scenario The user claim and drag items to decorate pond													
Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended									
1	Navigate to Pond Application	Site should open											
2	Enter Userid & Password	Credentials can be entered											
3	Click Submit	Customer is logged in											
4	Go to My Pond	Items and canvas is visible											
5	Click Items	Items appear and can be dragged											

Figure 5.7.2 Test Case for <Decorate Pond (UC2)>

5.7.3 TC2_03 : Test <Scenario of View HOF (UC2)>

Test Case ID	TC2_03	Test Case Description	Test the View Hall of Fame functionality	
Created By	Aimi	Reviewed By	Mirhaneza	
QA Tester's Log		Review comments from Mirhaneza incorporate in version 1.0		
Tester's Name	Aimi	Date Tested	February 1, 2021	Test Case {Pass/Fail/Not}
S # Prerequisites:		S # Test Data		
1	Stable internet connection.	1	email = aimirusdi@gmail.com	
2	Access to Pond website or application.	2	Pass = Abcd1234@	
3	User logged in.	3	Id = Aimi	
4	User access to main page.	4		
Test Scenario The user add and complete tasks.				
Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Navigate to Pond Application	Site should open		
2	Enter Userid & Password	Credentials can be entered		
3	Click Submit	Customer is logged in		
4	Navigate to Hall of Fame	Points and id can be viewed		

Figure 5.7.3 Test Case for <View HOF (UC2)>

5.8 Test Approach Analysis

Test Case ID	Step#	Input Data	Valid	Invalid
TC2_01	5	Email, Password,	Valid email, Valid password	Invalid email, Invalid password, Null
		Claim Items	Valid tier points, Valid items	Invalid tier points, Invalid items, Null.

Test Case ID	Step#	Input Data	Valid	Invalid
TC2_02	5	Email, Password,	Valid email, Valid password	Invalid email, Invalid password, Null
		Drag Items	Valid items, Valid pond	Invalid items, Invalid pond, Null.

Test Case ID	Step#	Input Data	Valid	Invalid
TC2_03	4	Email, Password,	Valid email, Valid password	Invalid email, Invalid password, Null
		View HOF	Valid current points, Valid id	Invalid current points, Invalid id, Null.

6. Code Versioning

Recently, we were assigned to create the first or Alpha version of the application as a form of progress update. At the time, the application was simple and most of the features were still incomplete. In the GitHub commit, it reached merely a 0.8 version as most of the proposed idea was still unavailable. The first version, we have completed the task feature where the users will be able to add, edit, complete and delete their task. Some of the gamifications such as animated objects were also implemented into the application although the claiming item feature was still unavailable. Moreover, at the time, we were still unable to convert the website into a mobile application.

However, once we have reached version 1.0.0, our application was upgraded in terms of landing page interface, interaction, more features were also available and most of the proposed idea was available. In our latest version, the features are mostly similar to the proposal with a touch of different arrangement and change of features compared to the prototype. Not only that, we have also updated most of the features for example the landing page used to be a still image, which is now a more attractive front page for the users. The landing page can be publicly viewed by all users who have registered or not as there will be a button choice for “Login” representing the existing users and “Register” for the non-users. This concludes that users must have an account before accessing the main page.

It is understood that our application does not have any specific type of users. This is because all users are allowed to use this application for any purposes either work, education or even chores as a reminder, not to mention even for fun to accomplish the gamification features of the application. All information of the user’s login or registration will automatically be stored in the database and updated at the Tenenet platform under the “Players” section. However, the updates on the features of each menu in the application had slight changes compared to the last version.

Firstly, in the first version, only task features such as add, edit, delete and complete were achieved. Nonetheless, in the latest version, the feature was updated in terms of functionality, interface and attractiveness. Once the user clicks the “Add Task” button, similar features will emerge. The difference is that now, instead of manually typing in the date, a mini calendar will appear for the users to select. Once the task is saved, a progress bar will fade up with the percentage amount out of 100% to show the progress of the user.

Secondly, in the previous version, the tier items only had the feature of displaying the draggable animations. We were not able to create the coding onto where the user will be able to select and claim the item to be displayed on their pond canvas. In the latest version we were able to implement the following upgrades. Pond Shop consists of five tier levels and each tier consists of three items. Users are only able to select and claim their desired item based on their rank level. If they have unlocked Tier 1, the user will be able to claim the items in Tier 1 by simply clicking on the “+” button. In addition, we created an extra feature of the user being able to write their reflection based on the completed task as to why they would like to claim the item and recall back the journey of them completing the task.

Lastly, in the first version, users were only provided with two draggable characters as items could not be claimed. In the latest version of Pond, after users claim their items, it is retrievable by clicking the “Items” bar button at the “My Pond” section. A list of items will be visible and the users have the advantage of reading back their reflection. All these features will also be available at the mobile application version as a scroll down feature.

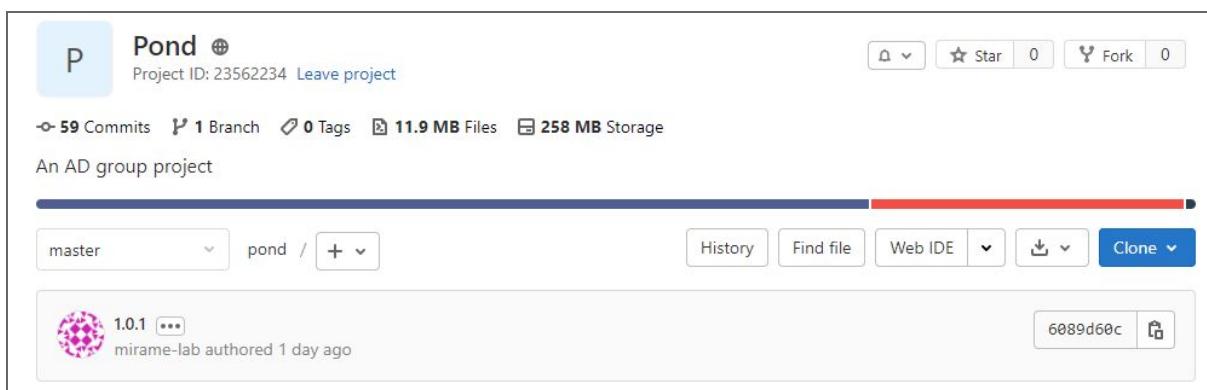


Figure 6.1 GitLab final version of Pond.

7. Software Release

7.1 Pre-alpha

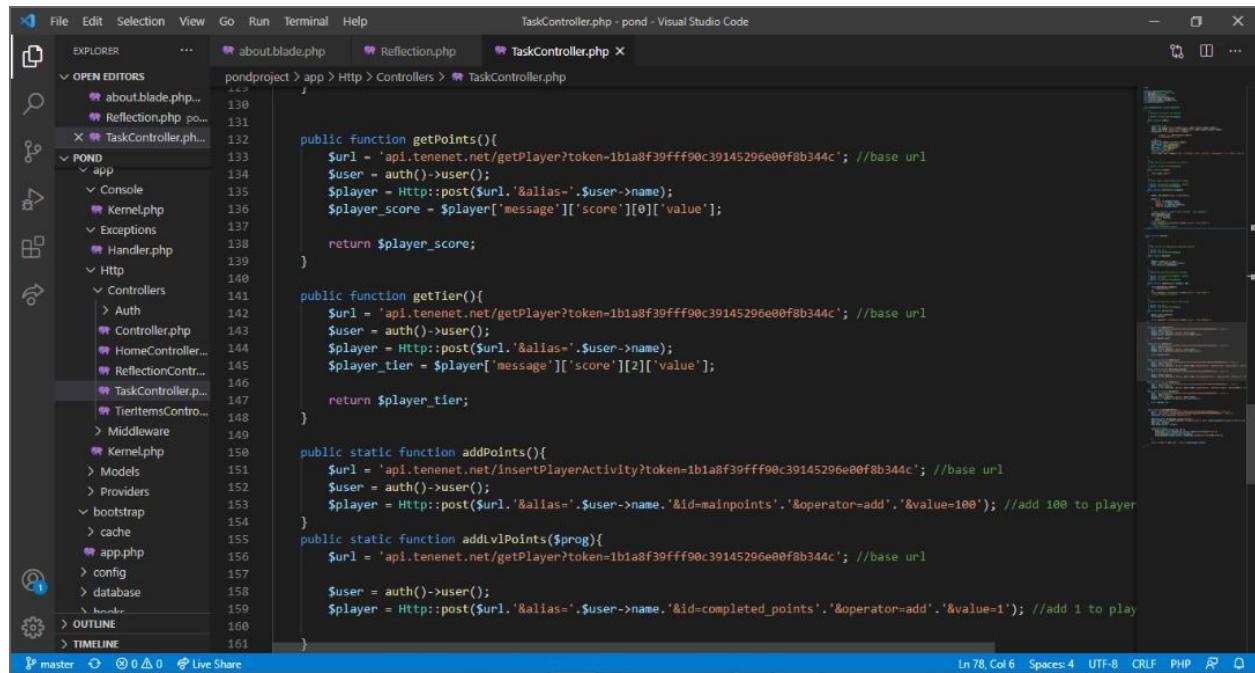
The pre-alpha version is referred to as when all the activities or features are performed during the software project implementation which is before the formal testing. From the start of the project, we accomplished the requirements analysis, the software design using figma and the software development using Visual Studio Code. Once these progress was complete, the application proceeded with the specific sets of functions to achieve the alpha version.

7.2 Alpha Version

The alpha phase or version is known as the first phase of the software development testing. The alpha software was not thoroughly tested by the developers which is ourselves, however, we have tried to reach out to a few users to test the software application. Along the way, there were some errors and incomplete features of bugs reports that needed to be fixed. During the alpha software, not all features that were planned were available until the final version. After several testing and updates, a complete alpha version of the application was successful.

7.3 Laravel and CSS

In creating the Pond application, we applied using the Laravel 8 framework into achieving maximum features and functionality on creating a web application. For the design, we implemented using Bootstrap design templates which is also responsive to any devices in computers or even mobile phones. We combined the Laravel implementation feature into the code using *php artisan migrate* to insert the functionalities into the codes. The frontend and backend development was created by using PHP, HTML and CSS. After coding, we access the localhost server to view the progress using *php artisan serve*. The CSS is used for the navigation of design on the web application. For Pond, blade.php was created as to where all the main codings are inserted to functionise the application thoroughly.



```

File Edit Selection View Go Run Terminal Help TaskController.php - pond - Visual Studio Code
EXPLORER ... about.blade.php Reflection.php TaskController.php
OPEN EDITORS pondproject > app > Http > Controllers > TaskController.php
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161

public function getPoints(){
    $url = 'api.tenenet.net/getPlayer?token=1b1a8f39fff90c39145296e00f8b344c'; //base url
    $user = auth()->user();
    $player = Http::post($url.'&alias='.$user->name);
    $player_score = $player['message'][0]['score'][0]['value'];

    return $player_score;
}

public function getTier(){
    $url = 'api.tenenet.net/getPlayer?token=1b1a8f39fff90c39145296e00f8b344c'; //base url
    $user = auth()->user();
    $player = Http::post($url.'&alias='.$user->name);
    $player_tier = $player['message'][2]['value'];

    return $player_tier;
}

public static function addPoints(){
    $url = 'api.tenenet.net/insertPlayerActivity?token=1b1a8f39fff90c39145296e00f8b344c'; //base url
    $user = auth()->user();
    $player = Http::post($url.'&alias='.$user->name.'&id=mainpoints'.'&operator=add'.'&value=100'); //add 100 to player
}

public static function addLvlPoints($prog){
    $url = 'api.tenenet.net/getPlayer?token=1b1a8f39fff90c39145296e00f8b344c'; //base url
    $user = auth()->user();
    $player = Http::post($url.'&alias='.$user->name.'&id=completed_points'.'&operator=add'.'&value=1'); //add 1 to play
}

```

Figure 7.3.1 Backend Coding at Visual Studio Code

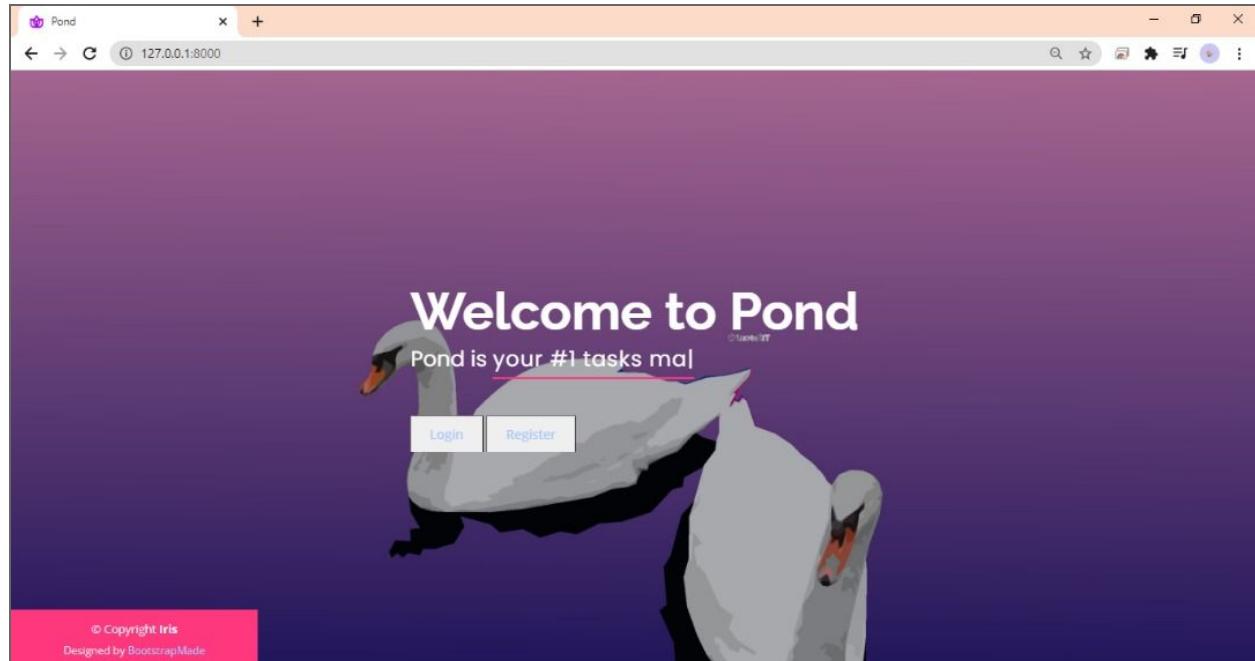


Figure 7.3.2 Testing at the localhost server

8. Bugs Record Document

During the development of Pond, there were many bug issues that needed fixing and updates in order to achieve the final outcome of a well developed website. Below is the documented record for the bugs report for the Pond application.

Title : Task Cards

Summary : The task cards are not aligned properly after created

Steps to Reproduce : Go to the Task section

Expected Results : Task cards are formed and aligned neatly beside each other

Actual Results : The task cards are some small some are elongated

Title : Claim Items

Summary : The items at each tier could not be claimed

Steps to Reproduce : Go to the Pond Shop section

Expected Results : The items are able to be claimed when clicking “+”

Actual Results : The button redirects to a not found page

Title : Hall of Fame

Summary : The Hall of Fame images could not be displayed

Steps to Reproduce : Go to the Hall of Fame section

Expected Results : Decorated Pond canvas can be displayed as Top 5

Actual Results : Only boxes can be displayed

Title : Progress Bar

Summary : The progress bar could not increase percentage

Steps to Reproduce : Go to Task section

Expected Results : Progress bar can increase once user completes a task

Actual Results : Progress bar stays static

Title : Mini Calendar

Summary : The mini calendar at the task create would not pop up

Steps to Reproduce : Go to the Task section and add task

Expected Results : Mini calendar will pop up for users to select date

Actual Results : Calendar would not pop up and has to insert manually

Title : Ranked Badges

Summary : Rank Badges could not all be displayed

Steps to Reproduce : Go toAchievements

Expected Results : All eight badges can be displayed

Actual Results : Some of the badges could not be displayed

9. Appendices

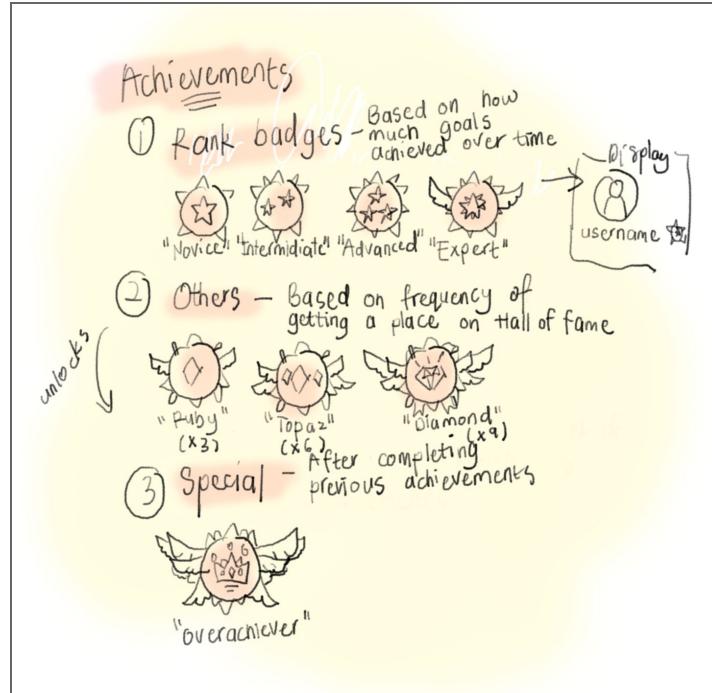


Figure 9.1 Sketch of Achievements.

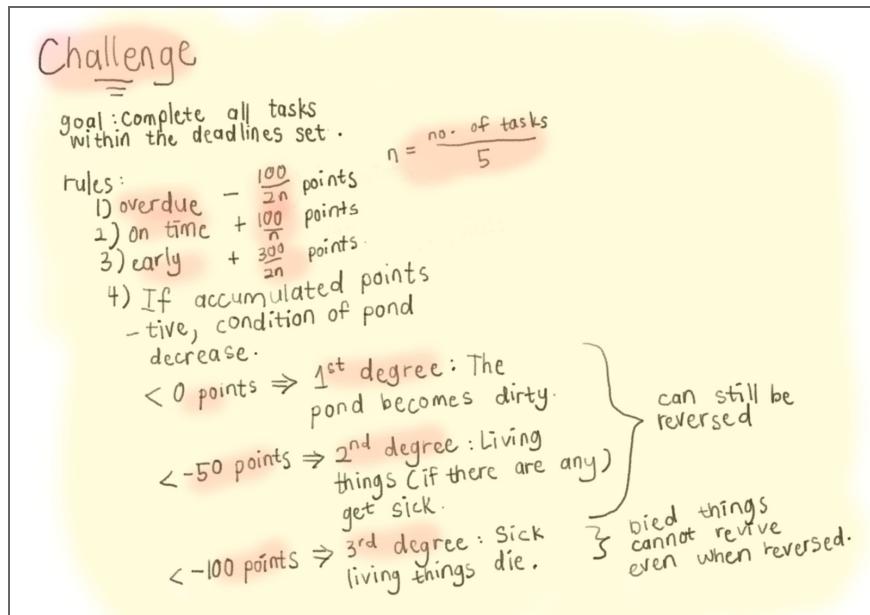


Figure 9.2 Sketch of Challenge Concept.

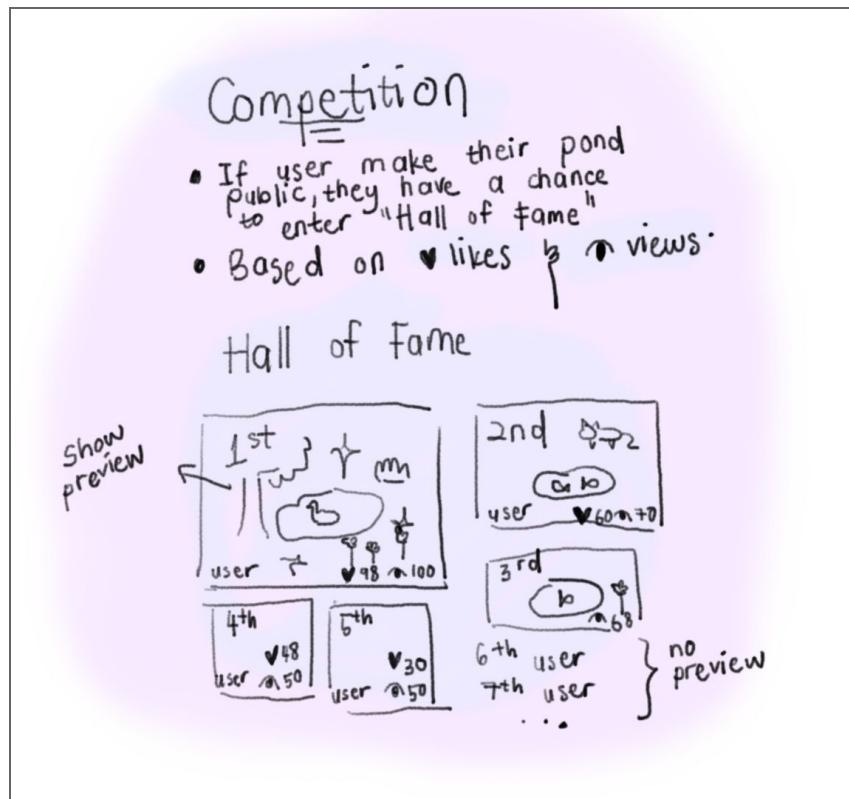


Figure 9.3 Sketch of Competition Concept.

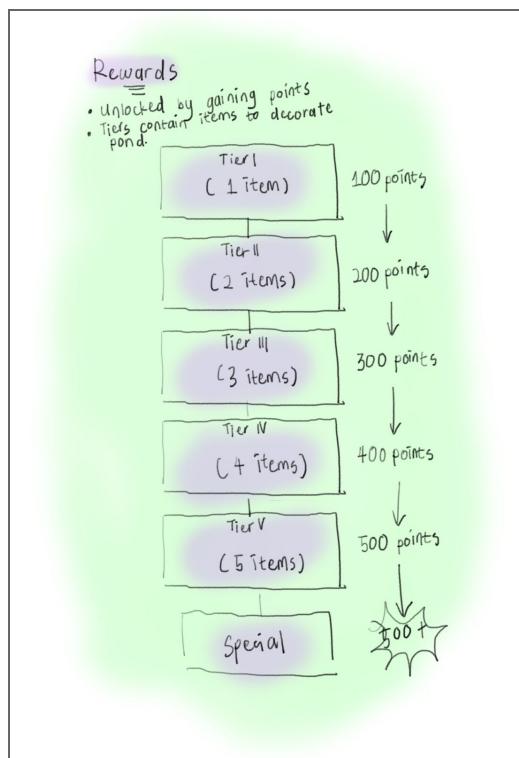


Figure 9.4 Sketch of Rewards.

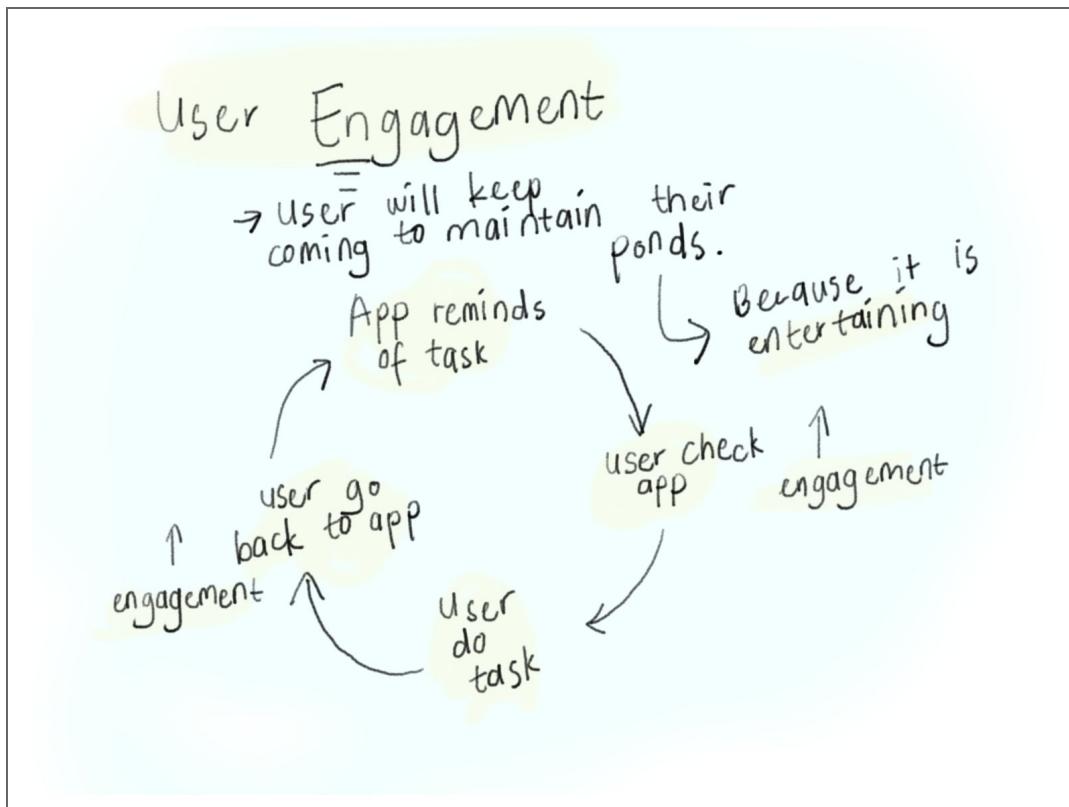


Figure 9.5 Sketch of user engagement

Website link : <http://pond.ci85h8b6lh-xmz4qd1vw62o.p.runcloud.link/>