

My Purrsistent Pet

by

Bekah Aylward, Church Myers, Carsten Schumaker

Submitted to
the Faculty of the School of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

© Copyright 2019 Bekah Aylward, Church Myers, Carsten Schumaker

The author grants to the School of Information Technology permission
to reproduce and distribute copies of this document in whole or in part.

<u>Bekah Aylward</u>	<u>4/26/2020</u>
Bekah Aylward	Date

<u>Church Myers</u>	<u>4/26/2020</u>
Church Myers	Date

<u>Carsten Schumaker</u>	<u>4/26/2020</u>
Carsten Schumaker	Date

<u>Tony Iacobelli</u>	<u>4/26/2020</u>
Tony Iacobelli, Faculty Advisor	Date

<u>Bogdan Vykhovanyuk</u>	<u>4/26/2020</u>
Bogdan Vykhovanyuk, Faculty Advisor	Date

University of Cincinnati
College of
Education, Criminal Justice, and Human Services
April 2020

My Persistent Pet



Prepared By
Bekah Aylward, Church Myers &
Carsten Schumaker

Students of
University of Cincinnati
College of Education, Criminal Justice, and Human Services
School of Information Technology

TABLE OF CONTENTS

SECTION	PAGE
LIST OF ILLUSTRATIONS	ii
TABLES	ii
FIGURES	ii
ABSTRACT	1
1. INTRODUCTION	2
1.1 Problem	2
1.2 Solution	2
1.3 Project Description	3
1.4 Overview	3
2. DISCUSSION	4
2.1 Project Concept	4
2.2 Objectives/Deliverables	5
2.2.1 Gantt Chart	6
2.2.2 Work Breakdown Structure	8
2.3 Technical Architecture	11
2.3.1 Network and Database	11
2.3.2 Application	11
2.3.3 Security	12
2.3.4 Technical Architecture Diagram	13
2.4 User Profile	14
2.5 Use Case Diagram	17
2.6 Budget	18
2.7 Testing	19
3. CONCLUSION	24
3.1 Lessons Learned	24
3.2 Skill Development	24
4. BACK MATTER	25
4.1 References	25

List of Illustrations

TABLES

<u>No.</u>	<u>PAGE</u>
TABLE 1: User Profile	5
TABLE 2: Budget	8
TABLE 3: Project Objectives/Deliverables	9

FIGURES

<u>No.</u>	<u>PAGE</u>
FIGURE 1: Gantt Chart	7
FIGURE 2: Technical Architecture Diagram	13
FIGURE 3: Use Case Diagram	17

ABSTRACT

Have you ever actually completed a New Year's Resolution? Neither have we! According to Forbes, 80% of Americans set a New Year's Resolution, but only 8% will actually complete their goal^[1]. The simple fact is that these self-improvement goals require motivation and discipline that most people lack. That's why we've created Purrsistent Pet. While there are many solutions available to keep track of your study and workout habits, there are none that also give you additional inspiration towards completing those goals. Purrsistent Pet provides this by making the user responsible for their own digital pet, and to care for it the user must make progress towards their goals. We've built this application on the fact that it is easier to motivate yourself when your new digital friend is relying on you, too, so please give it a try and improve yourself today!

1.0 INTRODUCTION

1.1 Problem

There are tons of applications, available on both mobile and desktop, that allow people to track their goals and make todo lists. Many of these applications serve their purpose quite well but they often prove to be a chore in themselves for users to use and they lack a certain flare that will have the user using the app for the long term. As stated earlier, people lack self control and become distracted easily. It is much easier to put down the textbook and pick up the controller then it is to do the opposite. Many people will tell themselves that they are still completing what needs to be done while doing other activities and say that they are multitasking. As science has proven, there is no such thing as multitasking. What is actually happening is the brain is switching between tasks at an incredibly fast rate. While this is a great and powerful ability, it gives people the false impression that they are multitasking and it decreases one's productivity compared to if they were focusing on one just one task. To this day, there has been no scientific breakthrough that alters the brain and allows one to work on different tasks in parallel. Until that day comes, the best we can do is try and bandage the problem as much as possible.

1.2 Solution

My Persistent Pet will take a unique approach to tracking goals and keeping people focused on what is important. The app will act as a sort of Tamagotchi like type application, where the user has to care for their pet, or else it will suffer and face the

same fate as the user's goals. By gamifying the process of tracking goals, we hope to give the user an added incentive to continue pursuing what was at one point important to them so they aren't left with regrets and thoughts of what could have been. When looking at one's tasks as a game, work will be more enjoyable and temptations to participate in other tasks will be greatly lowered.

1.3 Project Description

My Persistent Pet will be a mobile application that gamifies the process of goal tracking. Users will be able to set and edit their own goals, as well as view their progress, from within the app. The user will also take care of a virtual pet whose health and well being is dependent on the consistency and success of the owner in their attempts to complete their goals. Users will be able to see how efficient they are being through both their pet and the easy to use user dashboard. The user dashboard will serve as a location to view goals in detail and examine long term progress.

1.4 Overview

The remainder of this final report outlines in detail how the project was completed. The report includes the following sections: Project Concept, Objectives/Deliverables, Technical Architecture, User Profile, Use Case Diagram, Budget, Gantt Chart, and Problems Encountered.

2.0 DISCUSSION

2.1 Project Concept

A question everyone asks themselves from time to time is, “If I had stuck with that, how would my life be different now?” This is an age old question that comes in a variety of forms. When reflecting on life and what could have been, it’s easy to think about what could have been had more effort been put towards accomplishing one’s goals. Regret is a word that best describes this scenario and unfortunately, it isn’t a very good feeling. Too many people realize weeks, months or years later that they have made no progress towards their goals and that they have squandered precious time in the process. There are reasons why this happens and it mainly boils down to one’s level of self control and to some degree, desire for completing the set goal. Providing substantial reason for someone to resist outside temptation and work on what may not be the most enticing matter is not a simple thing to accomplish. It is easy to procrastinate and break habits when an easier alternative presents itself. In an increasingly dependent digital world, we have made it our goal to develop a mobile application that takes the user’s attention away from outside distractions and gives them incentive to work on what really matters.

2.2 Objectives/Deliverables

The project deliverables and objectives are shown in the table below (Table 3: Project Objectives/Deliverables). The major deliverables and objectives listed in the table were

determined by the Project manager and serve as big picture goals for our group to strive for. The dates serve as due dates for each goal.

MAJOR PROJECT OBJECTIVES AND DELIVERABLES			
FALL SEMESTER 2019 OBJECTIVES/DELIVERABLES			
RESEARCH TECH STACK	9/28/2019	DATABASE SCHEMA SETUP	11/3/2019
HOME PAGE SETUP	10/20/2019	FALL PRESENTATION	11/25/2019
SPRING SEMESTER 2020 OBJECTIVES/DELIVERABLES			
PRODUCTION ENVIRONMENT SETUP	2/1/20	COMPLETE VIRTUAL PET DEVELOPMENT	2/24/20
APP BETA TESTING	3/1/20	TECH EXPO FINAL PREPARATION	4/14/20

Table 3: Project Objectives/Deliverables

2.2.1 Gantt Chart

Our Gantt chart on the following page is an illustration of our project deliverables with not only their expected completion date but also the estimated time that we expect each activity to take. As you can see, we spent most of our Fall semester focused on research and developing the front-end framework of our application, leaving us plenty of time in the Spring to fill out the back-end and prepare for IT Expo. The blue line down the middle represents our Winter Break in December and early January, but while we will not be meeting in class during this time and we have no set goals, we do not plan to pause development and will instead use this time to finish up any outstanding tasks as well as get head starts on others.

2.2.2 Work Breakdown Structure

Below is our Work Breakdown Structure for our project, which details the same information presented in the Gantt Chart more specifically.

Task Name	Duration (Days)	Start Date	End Date
1.0 Project Management and Deliverables	TBD	8/20/19	TBD
1.1 Project Selection & Proposal	13	8/20/19	9/2/19
1.2 Fall Semester Assignment 0: Team Members and Project Name	13	8/20/19	9/2/19
1.3 Fall Semester Assignment 1: Team Contract	21	9/2/19	9/23/19
1.3.1 Advisor Project Approval	14	9/2/19	9/16/19
1.3.2 Gantt Chart	21	9/2/19	9/23/19
1.3.3 Work Breakdown Structure	21	9/2/19	9/23/19
1.4 Fall Semester Assignment 2: Project Abstract for Tech Expo	21	9/23/19	10/14/19
1.5 Fall Semester Assignment 3: Team Contract Resubmission	TBD	TBD	10/14/19
1.6 Fall Semester Assignment 4: User Profile	7	10/14/19	10/21/19
1.7 Fall Semester Assignment 5: Use Case Diagram	7	10/14/19	10/21/19
1.8 Fall Semester Assignment 6: Draft Report	14	10/21/19	11/4/19
1.9 Fall Semester Assignment 7: Final Fall Semester Report	28	11/4/19	12/2/19
1.10 Fall Semester Oral Presentation	TBD	11/4/19	TBD
1.10.1 Presentation Practice	TBD	11/4/19	TBD
2.0 Research	42	9/8/19	10/20/19
2.1 Software Requirements	21	9/8/19	9/30/19
2.1.1 Determine Front End Development Languages	21	9/8/19	10/1/19
2.1.2 Determine Back End Development Languages	21	9/8/19	10/1/19
2.2 Network Requirements	42	9/8/19	10/20/19
2.2.1 Determine Hosting Environment	21	9/8/19	9/29/19
2.2.2 Determine Database Environment	21	9/15/19	9/29/19

2.2.3 Research IP Mapping	21	9/29/19	10/20/19
2.2.4 Research Load Balancing Systems	21	9/29/19	10/20/19
2.3 Security Requirements	42	9/8/19	10/20/19
2.3.1 Research HIPAA Compliance	42	9/8/19	10/20/19
2.3.2 User Security and Authentication	21	9/8/19	9/30/19
2.3.3 Research Malicious Injection Methods (OWASP)	34	9/8/19	10/13/19
2.3.4 Research Legal Disclaimers	42	9/8/19	10/20/19
2.4 Miscellaneous Research	21	9/29/19	10/20/19
2.4.1 Determine Analytics Dashboard	21	9/29/19	10/20/19
2.4.2 Budget Analysis	14	9/29/19	10/13/19
3.0 System Design	28	9/22/19	10/20/19
3.1 Create System Diagrams	28	9/22/19	10/20/19
3.1.1 Create Network Diagrams	14	9/22/19	10/6/19
3.1.2 Create Database Diagrams	14	9/22/19	10/6/19
3.1.3 Create Interaction Diagrams	21	9/22/19	10/13/19
3.1.4 Create Wireframe Diagrams	21	9/29/19	10/20/19
3.2 Create Legal Documentation	21	10/6/19	10/27/19
3.2.1 Write Up Legal Disclaimers and Privacy Policies	21	10/6/19	10/27/19
3.2.2 Write Up HIPAA Information and Consent Form	21	10/6/19	10/27/19
4.0 Environment Set-Up	TBD	9/22/19	TBD
4.1 Setup Private GitHub Repo	7	9/22/19	9/29/19
4.2 Setup Database Server	7	9/22/19	9/29/19
4.2.1 Install MongoDB	7	9/22/19	9/29/19
4.2.2 Create Mongo Documents	7	9/22/19	9/29/19
4.3 Setup Backend Server	7	9/22/19	9/29/19
4.3.1 Install Node.js and required npm modules	7	9/22/19	9/29/19
4.4 Setup Load Balancer	14	1/12/20	1/26/20

4.5 Purchase Domain Name	14	1/19/20	2/1/20
4.5.1 Install Site Certificates	7	1/19/20	1/26/20
5.0 Development (Back End and Front End)	175	10/6/19	3/29/20
5.1 Create main React Native App & Home Page	14	10/6/19	10/20/19
5.2 Create Navigation Bar	14	10/6/19	10/20/19
5.3 API setup	21	10/6/19	10/27/19
5.3.1 Configure API's on Node Server	21	10/6/19	10/27/19
5.3.2 Configure Backend API's	21	10/6/19	10/27/19
5.4 Design and Develop Dashboard/User Portal	14	10/13/19	10/27/19
5.5 Create Login and Set Up Authentication	14	10/20/19	11/3/19
5.5.1 Create User Registration and Confirmation	7	10/20/19	10/27/19
5.5.2 Create Forgot Password Feature	7	10/20/19	10/27/19
5.5.3 Set up Redis to handle sessions	14	10/20/19	11/3/19
5.6 Develop Features	113	11/3/19	2/24/20
5.6.1 Design and Develop Virtual Pet	113	11/3/19	2/24/20
5.6.2 Design and Develop Views For Goals	14	11/3/19	11/17/19
5.6.3 Design and Develop Specific Goal Templates	34	11/10/19	12/14/19
5.6.4 Design and Develop Progress Tracker	28	1/19/20	2/17/20
5.6.5 Design and Develop Health System For Pet	21	1/12/20	2/2/20
5.6.6 Design and Develop Task Creation & Editing	14	1/12/20	1/26/20
5.6.7 Design and Develop Notification System (let users know if they're slacking, etc.)	14	2/2/20	2/16/20

Figure 3: Work Breakdown Structure

2.3 Technical Architecture

2.3.1 Network and Database

My Persistent Pet's database and networking needs will be handled by MongoDB Atlas and AWS. This is a database as a service solution, and due to its simplicity, we will be able to spend more time writing front end code and business logic rather than working about the scalability and availability of our application. Instead of spending weeks configuring servers, setting up networks with the proper security and access roles, and making sure the proper dependencies are installed, we can configure and manage our MongoDB Atlas cluster from the web console with just a few clicks. Aws, GCP and Azure are the offered hosting solutions for Atlas clusters, we have chosen Aws, and we have chosen the free hosting cluster tier with minimal hardware specs. Should we need to expand our database and hosting environment, we can simply go to the MongoDB Atlas console and make the appropriate changes for our existing cluster. For the sake of the project, we will assume the app is being pushed to both the Google Play store as well as the Apple App store for download rather than being downloaded from a separate hosting service.

2.3.2 Application

The app will be written in Javascript and we have chosen to use React Native as our frontend framework. This will allow us to use a single Javascript codebase that can be run on both Android and iOS. Built on top of MongoDB Atlas is MongoDB Stitch. Stitch functions allow us to create server side logic for our

application and these functions can be called from our client side code as well as from other functions within Stitch. Stitch functions are written in vanilla Javascript and execute in a servless manner. This lightweight solution, allows us to write the entire application in Javascript, getting rid of the need to learn multiple languages and frameworks for the sake of one project.

2.3.3 Security

Proper security measures for the application will be emphasized with secure coding practices and by making sure that the proper permissions for users are assigned correctly from MongoDB Atlas. To keep our code away from prying eyes, we will be using Uglify to obfuscate the code. This isn't to say that the code won't be able to be deobfuscated since someone who has the time and resources certainly would be able to do so, but it will at least deter some people from looking into accessing our source code. This is the point where secure coding practices come in handy. Making sure sensitive business logic isn't included in the code sent to the user is one major step in the right direction. As for database security concerns, we are in luck. With MongoDB Atlas, we can link our cluster to our Mongo Stitch application where we can define roles that ensure our users can only read and write what they are intended to.

2.3.4 Architecture Diagram

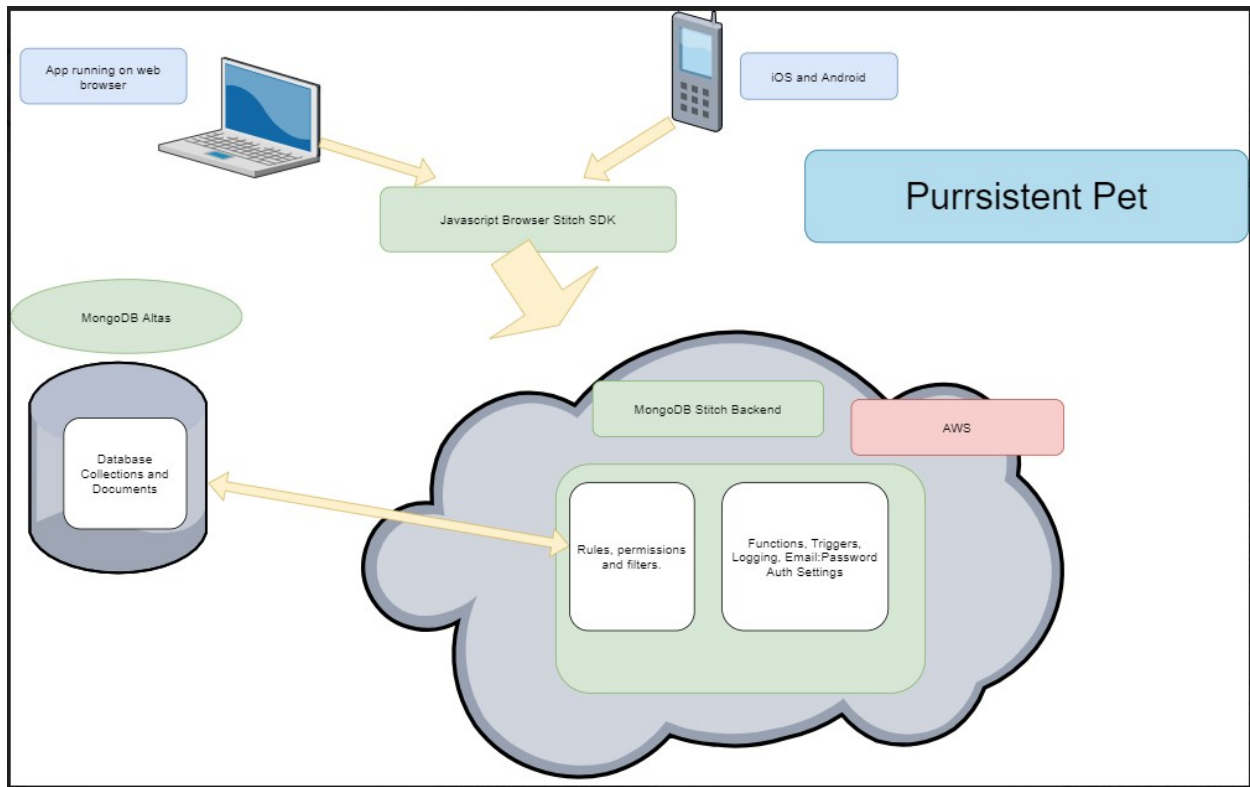


FIGURE 2: Technical Architecture Diagram

2.4 User Profile

Table 1: User Profile. The table below highlights the characteristics of the typical user of My Persistent Pet and provides some information regarding the UX. The goal of the user profile is to create a brief, yet detailed outline for what the end goal of our app should look like and what purpose it should serve. The points declared in the user profile should all be met upon the app's completion.

USER PROFILE		ADMIN PROFILE	
My Persistence Pet			
POTENTIAL USERS			
Individuals looking for a fun way to keep on track with their goals, as well as people who have a difficult time keeping up with their study/exercise regimens.		Individuals who manage the users and track data for the application.	
SOFTWARE AND INTERFACE			
This project is a mobile application targeted towards an individual that needs help setting goals and staying on track to completing those goals. This application gamifies these goals by requiring the user to take care of their virtual pet. As tasks are completed, the pet will become healthier and stronger. If the user veers off course, the pet will suffer.			
EXPERIENCE WITH SIMILAR APPLICATIONS			
My Persistent Pet users may be familiar with the following apps and products:			

§ Tamagotchi

§ Habitica

§ Loop

§ Strides

§ Coach.me Habit Tracking

§ FitBit

TASK EXPERIENCE

No prior goal tracking app experience is required. Upon opening the app, the user will land on their dashboard page where they can view their tasks and select individual tasks for a more detailed view. The app will be intuitive to use and anyone who has used a smartphone before will be able to navigate the app with ease.

Technologically adept users or users with previous application administrator experience will be able to track data and maintain user profiles.

FREQUENCY OF USE

This application will mostly be used daily to keep up with goals that the user has created. The user may also choose to use the application less frequently dependent on how often the user sets goals.

The administrator will be able to assist users in the case of account trouble or if the administrator would like to gather data as needed.

KEY PROJECT DESIGN REQUIREMENTS THAT THE PROFILE SUGGESTS

§ Easy to navigate UI § Visually pleasing virtual pet including its needs § Ability to easily create goals § Views that encapsulate a user's overall progress in a presentable manner	§ Easy to navigate UI § Will be able to track data § Will be able to manage user accounts
---	--

Table 1: User Profile

2.5 Use Case Diagram

The following diagram displays the use case for My Persistent Pet. Users and Admins will be able to access and interact with their associated roles. Should new roles be discovered during the development process, the changes will be reflected in the use case diagram.

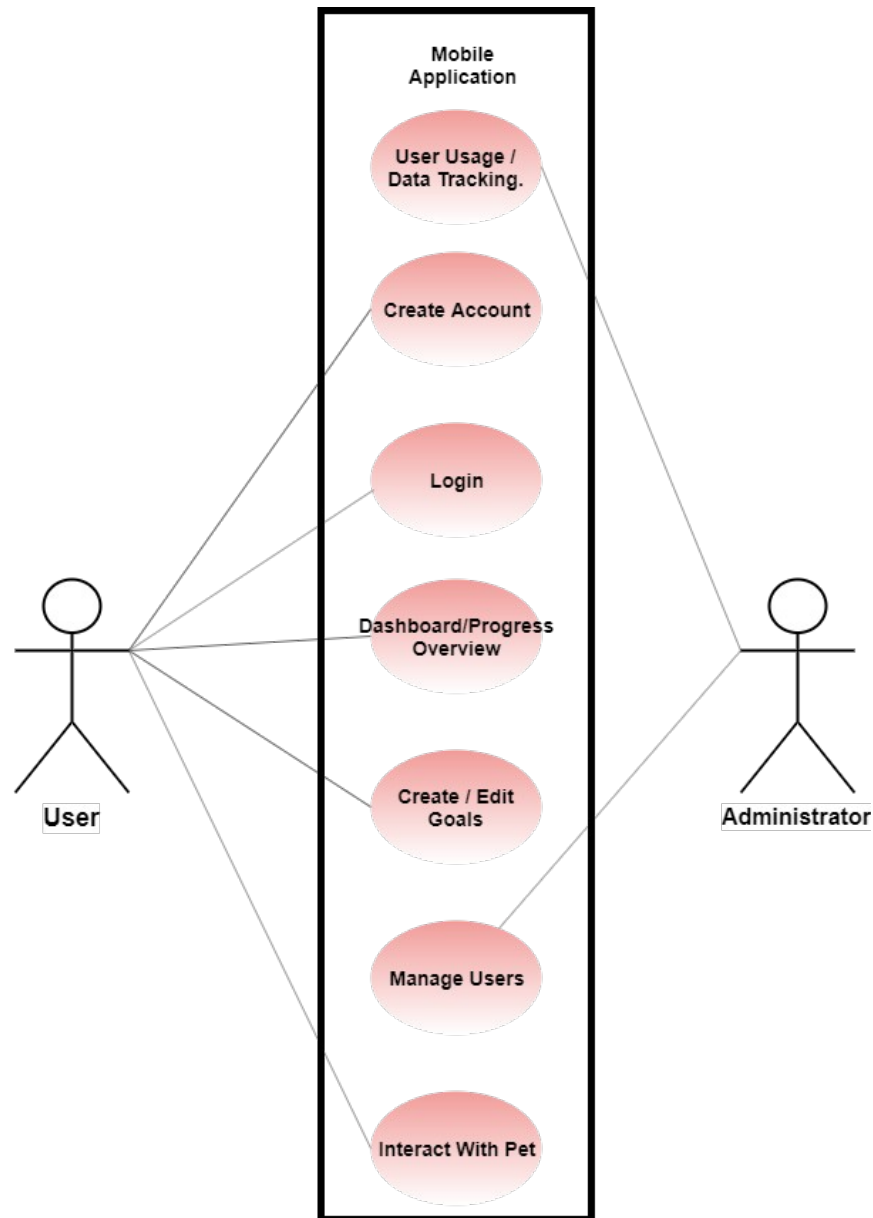


Figure 3: Use Case Diagram

2.6 Budget

The budget for this project is displayed in the table below. There is an actual cost and an expected cost column. The actual cost column details what the costs for the project would be in the real world and the expected cost column details what the costs will be for the purpose of Senior Design. Also, it is worth noting that for the backend expenses, we are using MongoDB Atlas. We will be sticking to the free tier for the course of the project.

Expenses	Description	Expected Cost	Actual Cost
Software	Software (completely open source) used to develop the application. In addition, third party testing software may be needed.	\$0	TBD
Hardware	Physical devices used for networking and storage.	\$0	TBD
Employee Wages	Wages for the 3 employees working on the projects. It is assumed that their wages are \$25/hr. The formula for Actual cost here = 3 people x 25 weeks x 25 \$/hr x 20/hr work per week. (The 25 weeks is based off the WBS showing 25 weeks total for backend and frontend development.)	\$0	\$37,500
Miscellaneous Wages	Possible wages for marketing, outside consulting/contracting, paid learning resources, etc.	\$0	TBD
Total		\$0	\$37,500 + TBD

Table 2: Budget

3.7 TESTING

3.7.1. Overview / Methodology

This document outlines the testing procedures that were used to test the Purrfect Pet mobile application. The application is able to be used on both Android and iOS. In order to determine what qualifies as a working product, specific tests for different components were created and strict pass/fail requirements were set. Different tests were conducted at different stages of the development lifecycle, in order to ensure components work in sync with other components and properly perform their individual functions.

3.7.2 Scope of Testing

At a high level view, the scope of testing is to ensure that the application functions properly without crashes on the user. When looking at individual tests, the scopes can vary. This is simply due to the fact that different components within the application perform different functions. For example, one test may be written for a component that displays a user's goals. Certain tests could be written to ensure that that particular component only display's that user's goal and not some other user's. There could also be a filter function that sorts the user's goals by date. Perhaps the user only wants to see currently ongoing goals. Inside the scope of the test for that specific component, there would need to be a requirement stating that the filter function should always display the proper

information based on the filter requirements. In this case, the ongoing goals. In theory, as long as individual components are functioning properly, the entire app should function properly. Similar to proper development practices for writing clean code and keeping it “DRY”, tests should be the same. The scope should be tailored to the function of the component(s) and test that the component(s) does nothing more and nothing less than what is desired.

3.7.3 Objective

The objective of the testing is to ensure that individual components in the source code are performing their desired functions. If a certain component isn't functioning as desired, then the app cannot be put in production mode. The same goes for if a component is doing more than expected. This can pose a security issue or in the future, cause unnecessary work for developers. The application should also be tested as a whole, by different audiences, in order to gain different perspectives on app functionality and the UI.

3.7.4 Test Procedures

In order to perform our tests, we prepared our application to run on a local Android emulator on one of our team's devices, and solicited responses from our peers on campus. We asked each respondent to perform a series of tasks and to

issue a simple pass or fail response for each test, in order to determine if the application's workflows function acceptably.

3.7.5 Test Scenarios

5.1 Account Creation	
Given	an unregistered user on the login page
When	they tap the Create Account Button
And	enter the required information
Then	they will receive a confirmation email with a link to log in
And	clicking this link will enter the user onto the Dashboard page

5.2 Goal Creation and Completion	
---	--

Given	a user on the goals page
When	the user taps Create Goal button
And	enters the required information
Then	the goal is created in the list
And	the user can complete the goal for points

5.3 Epic Creation	
Given	a user on the goals page
When	the user taps on create epic
And	enters the required information
Then	the epic is created and added to the page
And	the user can add goals to this epic

5.4 Reward Redemption	
Given	A user on the pet page

When	The user clicks on a reward to unlock
And	They have the required amount of points
Then	The reward is unlocked
And	The associated stat on the pet is raised

4.0 CONCLUSION

4.1 Lessons Learned

Throughout this project, we have learned too much about both software development and project management to list here. We have had a lot of trouble working together and effectively communicating this year, mostly since our team was so remote, and this taught us how better to work together and collaborate proactively in a remote working scenario. We learned that without a dingle, motivated leader that is willing to push people and track progress, nothing will get done and things will stall as everyone waits for everyone else to do something. During our final presentation, we learned that we had done a poor job clarifying what each person's role was, leading to people talking over one another. This could have been avoided had our leadership taken a bit more active role in delegating responsibility and giving clear direction.

4.2 Skill Development

Our software developers have become more familiar with MongoDB Atlas and Stitch, while our system admin has learned more about Javascript and React. We all learned a little bit about how to host a web application with AWS, as well as how to use a web view to use a web application on a mobile device. All of us also spent a lot of time learning about different JS frameworks and how the entire stack of a mobile application works together to serve the application. We have tried to ensure all team members are familiar with each part of the project, so everyone is available to help when needed.

5.0 BACK MATTER

5.1 References

^[1]<https://www.forbes.com/sites/ashiraprossack1/2018/12/31/goals-not-resolutions/#139fd7ad3879>