Project Plan Document: BitOHealth

The 6 Bits:

Angel Cueva

Dante Secundino

Emily Sahyoun

Jacob Munoz

Rami Iskender

Rifat Hasan

Sebastian Vasquez (Team Leader)

2/4/2022

Revision History

Date	Version	Description
10/24/21	1.0	First Draft
10/28/21	1.2	Implemented feedback from professor
11/30/21	1.4	Added Core Components
12/02/21	1.4.1	Removed Project Roadmap
2/2/22	1.4.2	Added Hot Topic Subfeatures

Table of Contents

Cost Analysis 4

Risk Analysis 5

Timeline Overview 7

Timeline Breakdown 8

Timeline 13

Cost Analysis

The project has been to minimize the cost of the entire system by using borth free software as well as free trials wherever possible. The cost of the system is close to $0 even though we will rent a single web server and a single database from AWS for the development of the system using their free tier. The end goal for this product is not to profit but to provide a helpful system for our users.

Higher costs may arise if our website receives a lot of traffic. We will need to pivot to a more expensive EC2 instance and larger databases.

Below is an in-depth breakdown of our cost analysis.

Category	Description	Resource	Cost
Deployment	The cost to purchase and maintain the website	Domain	$10.00
		SSL Certificate	$0
		Web Host	$0
		AWS EC2	$0
		AWS	$0
Technology	Map APIs	Maps Static API	$0
		Geocoding API	$0
		Good RX API	$0
	Other APIs	Google Maps	$0
		Bing News	$0
		Spoonacular	$0
	Browser	Chrome	$0
	Operating System	Windows 10 Home	$0
	Text Editor	Visual Studio Code 1.59+	$0
	IDE	Visual Studio Code 1.59+	$0
	Front End	ECMAScript 11+	$0
	Back End	C# 9.0	$0

	Front End Frameworks	React 17+	$0
	Back End Frameworks	.NET 5+	$0
	Database	SQL Server 2019 Developer/ExpressEdition	$0
	SQL Management	SQL Server Management Studio	$0
	Server	IIS 10+	$0
	Project Management	Trello	$0
	Communication	Discord	$0
	Diagram Software	Draw.io	$0
	Domain Registration	Namecheap	$0
	Project Code Hosting	Github using Git 2.33.0	$0
	Document Storage/Management	Google Drive	$0
	Email	Outlook	$0
	Documentation	Google Docs & Google Sheets	$0
		Total Cost	$10.00

Risk Analysis

The largest in our project is that each functionality may take longer to implement than intended. This may occur due to the lack of experience of the technology or a lack of understanding of how to implement the functionality. Both of these may lead to research and design to take longer. To reduce this risk, developers have been researching these items before the need to use the technology or the need to implement the functionality.

I. Scope Creep

To mitigate scope creep we have discussed with the team to gauge the amount of work we are able to handle per sprint. We have accounted for constraining factors in the coming months such as midterms and finals. If scope creep still occurs we will need to speak with the client to potentially postpone the items we have not finished. We may also have some of our team members work extra hours and assist on these tasks.

II. Team Member Drops

To mitigate this we have assigned heavy and low workloads on some sprints. This way, if a team member drops we will be able to move features around in the timeline to try to accommodate this. If we find that it is impossible to finish due to the lack of free time then we will need to speak to our client to remove whatever feature we can not finish.

III. Team Member Lack of work

To mitigate this we have chosen to do two week long sprints. This way we can go in depth with what has been accomplished and what has not. If this happens we will need to assign another team member to help out. We will also discuss as a team to figure out what went wrong and how to stop this from happening in the future. If the team member completely disregards our concerns and does not care about working as a team we will speak to the client about removing them.

IV. Inaccurate Time Estimates/Insufficient Time

To mitigate this from happening our team has pooled our industry knowledge from previous projects to attempt to have accurate estimates. If our time estimates are still off and we find ourselves running out of time we will need to speak to the client to address what work items we can leave out.

V. Communication Issues

To mitigate this we have bi-weekly stand-up meetings and talk over Discord to discuss whatever is needed. We also have constant text communication where we can reach out to each other if anything is needed in an instant.

VI. Tech Approval

To mitigate this we have to research thoroughly our technology needs. Then discuss the technology in a Discord meeting to arrange for someone responsible for making the DAR report and confirming it with our team leader. The team leader will send the DAR report and await our approval. If the DAR report is rejected due to having insufficient information and/or incorrect format. The person responsible for writing the DAR report will have to correct the mistakes and/or research more on the technology. However, if the DAR report is rejected, then our team will discuss a different technology or use the technology stack provided by the professor.

VII. The Unforeseen Circumstances

For any circumstances if a natural disaster or a medical related event has happened to a team member. We will attempt to contact the team member 24-48 hours. If no response is given, we will contact the professor to discuss the situation. If an individual is sick for more than 3 days, our team will have a team meeting on discord to discuss and divide the work from the team member who is sick. Whenever the team member recovers from a medical related/health related event he or she has to take over the work and complete their work.

Timeline Overview

Work Item	**Total Estimated Hours**
Project Proposal	40
Business Requirement Document & Tech Specs	100.5
Project Plan	44.5
Testing Plan	64
Network Diagram	22.5
High Level Design	73
Site Map	38

Medication/Health Recorder	62
Remainders	71
Health Locator	32
Medication Lookup	76
Weight Management/Calorie Counter	149
Track Records	42
Diet Recommendations	89
Hot Topics	34
Login/Logout	93
Registration	18
Mockup Design	18
TOTAL	1066.5

Time Constraint: The timeline shown depicts an estimated schedule. All work items are subject to change at any time.

Timeline Breakdown

The in-depth breakdown for each feature is the estimated time of individual tasks in our project, which were determined by researching and developing a realistic time of the functionality it would take to design, implement, and test.

Medication/Health Recorder

Functionality	Backend/Frontend Design	Implementation	Testing	Total (Hours)
Create Record	10	7	4	21
Input Data/Edit Record	5	7	4	16
Save Record	4	3	3	10
View Record	3	4	2	9
Delete Record	4	3	2	9
Totals	26	24	15	81

Why: This will be a fairly difficult task to achieve as we have to deal with communicating with the database, and also having it extremely secure. To be scalable we will need to implement HIPAA procedures when recording certain files, and that will be fairly difficult to achieve. We have three members with backgrounds in cybersecurity which will help speed this up.

Reminders

Functionality	Backend/Frontend Design	Implementation	Testing	Total (Hours)
Create Reminder	11	7	5	32
Edit Reminder	7	5	4	16
View Reminder	7	5	3	15
Delete Reminder	7	5	4	16
Export Reminder	8	5	6	19
Enable Notifications	7	6	5	18
Totals	47	33	27	107

Why: This feature will also communicate with our backend. We will have some experience with this though after completing the health recorder. A team member has experience with creating a basic reminder app with React and Django, and we based these time estimates off of that. It may vary based on each team member's skill level.

Health Locator

Functionality	Backend/Frontend Design	Implementation	Testing	Total (Hours)
Search	23	15	12	50
View Search Results	15	8	6	29
Totals	38	23	18	79

Why: This will be a simple feature as it only uses a 3rd party API.

Medication Lookup

Functionality	Backend/Frontend Design	Implementation	Testing	Total (Hours)
Start Search	15	9	5	29
View Medication	8	5	3	16
Favorite Medication	12	8	4	24
View Favorite List	8	6	3	17
Refill Medication	10	6	4	20
Medication reminder	10	6	4	20

Totals	63	40	23	126

Why: A big portion of this is the 3rd part API, which will take time to learn how to use efficiently. The feature will also require reminders, and a fair amount of front end to fully develop, which is important to give the website the best design and implementation. Back end and database will be heavy which will require security.

Weight Management/Calorie Counter

Functionality	Backend/Frontend Design	Implementation	Testing	Total (Hours)
Input Weight Goal Data	11	5	5	21
Edit Weight Goal Data	11	5	5	21
Search Food Item	8	4	3	8
Add Food Item	10	6	5	21
Custom Food Item	14	5	8	27
Delete Food Item	9	6	5	20
Counting Calories	11	5	5	21
Export Food Log	8	5	4	16
Totals	82	41	40	163

Why: This is the largest feature we have and will take some time to complete, for this reason we have dedicated one whole sprint for this.We are basing the timing off of members' previous projects. Every member will have some experience on how our system works by the time we get to this feature, after doing reminders and the health recorder.

Track Records

Functionality	Backend/Frontend Design	Implementation	Testing	Total (Hours)
Search Records	7	4	3	14
Categorize Records	16	7	6	29
View Folder	11	6	5	22
Totals	34	17	14	65

Why: This is simpler than the diet recommendations since it has less features, but it will be difficult as it is dealing with backend and database as well.

Diet Recommendations

Functionality	Backend/Frontend Design	Implementation	Testing	Total (Hours)

Create Diet Recommendations	14	7	5	26
View Details of Recommendations	7	4	3	14
Add Meal to Meal List	11	6	4	21
VIew Meal List	6	5	4	15
Delete Meal from Meal List	10	6	4	20
Totals	48	28	24	100

Why: This should be a difficult task to work on since we deal with pinging our own backend and database when dealing with the meal list. This will be difficult as most of our team members have only experienced this in a classroom setting.

Hot Topics

Functionality	**Backend/Frontend Design**	**Implementation**	**Testing**	**Total (Hours)**
View Hot Topic	7	5	4	16
Opt-in Notifications	13	4	4	21
Related News Page	7	4	3	14
Favorite News Articles	5	3	2	10
Remove Favorited Article	4	2	1	7
Shareable Articles	5	4	4	13
Filter/Search Bar	8	3	4	15
Totals	18	8	8	96

Why: Multiple of our team members have experience with 3rd party API’s and displaying it so it should not be a difficult task to implement.

Login

Functionality	**Backend/Frontend Design**	**Implementation**	**Testing**	**Total (Hours)**
Login/Logout	15	11	9	32
Authorization	10	12	10	32
2 Factor Authentication	11	7	5	16
Totals	41	37	29	107

Mockup Design

Functionality	Design	Implementation	Testing	Total (Hours)
Main Page(UI/UX)	17	12	8	37

Why: Some members have previous experience doing this and it should not take too long to create after designing a good one.

Core Components

Functionality	Design	Implementation	Testing	Total (Hours)
Authentication	14	8	6	28
Logging	16	10	5	31
Archiving	14	8	6	28
User Management	40	21	10	71
User Analysis Dashboard	18	17	7	42
Account Deletion	15	11	5	31
Registration	15	10	7	32
Account Recovery	15	12	6	33
Totals	132	97	52	281

Why: Some work is already finished by team members, the total hours is an estimation. For example, login/logout, account creation, authentication, and authorization are already mentioned. This portion is taken from the Core Requirements Document, so we added the features that were not added previously and implemented it into our project plan.

Timeline

This is a simple timeline which visualizes the main due dates in simpler terms.