

## Group 8

Mark Martin

Saul Leal-Garcia  
Damandeep Singh

Justin Phuong  
Jared Mallari

Dean Quach  
Can Chai

\*\*\*\*\* Requirements Table Column\*\*\*\*\*

Requirements	Functional/Non-Functional
A user shall be able to download the software	Functional
A user shall be able to connect their test device to Bluetooth	Functional
A user shall be able to upload their results to a database	Functional
A user shall be able to retrieve their results to the database	Functional
A user shall be able to update their results on the database	Functional
A user shall be able to share their results with need to know parties (Doctors)	Functional
A user shall be able to make a generic account that connects to their ID for identity purposes (Such as drivers license, social security, etc.)	Functional
A user shall be able to see their results live on the app	Functional
A user shall be able to see progress of the results retrieval	Functional
A user shall be able to see progress of the results retrieval	Functional
A user shall be able to confirm their symptoms in the app	Functional
A user shall be able to show their cleared status via on the phone or watch app	Functional
A user shall be able to know their health stats such as heart rate or temperature via the app	Functional
A user shall be able to know what variant they have	Functional
A user shall be able to see a heatmap of other users who may have covid	Functional
A website available on every web browser.	Functional
An app available for both IOS and Android users free of charge.	Functional

Share who has access to specific covid info	Functional
Specific Users can access user info	Functional
Both website and app will send users current news on covid(positive tests, new variants, etc)	Functional
The website shall be hosted on an AWS server.	Functional
System should update the existing software automatically	Functional
System should have an easy to use User Interface for both app and website.	Functional
The system should be compatible with devices within the last 20 years	Functional
The UI shall be user friendly, easy to navigate and intuitive	Functional
The system should be fast and responsive to the user. Accessing, loading and sending data shall take less than a second.	Functional
The system should be fast and responsive to the user. Accessing, loading and sending data shall take less than a second.	Functional
The app and website will display a vibrant and colorful scheme of colors to captivate the user	Functional
The app will have the functionality to send user feedback to the developers (for improvement purposes)	Functional
The app or website shall ask for users permission to allow use of sensors.	Functional
The app shall have the functionality for people who require disability accommodations.	Functional
Password and two factor authentication to secure patient data	Functional
User shall have their password encrypted via https	Functional
Guarantee anonymity for users	Non-Functional
Protect user's private info (birthdate, social security number, etc.) by encrypting it	Functional
System must have a team of cybersecurity experts monitoring that system to prevent hacks	Non-Functional
App must have biometric authentication for increased security and ease of use	Functional

App should have the ability to remember user's usernames and passwords if they choose	Functional
Budget shall account for developers	Non-Functional
Budget shall account for Systems engineers	Non-Functional
Budget shall account for Amazon Database to hold user data	Non-Functional
Budget shall account for engineers for the physical device	Non-Functional
Budget shall account for manufacturing	Non-Functional
Budget shall account for lab technicians	Non-Functional
Budget shall account for virologists	Non-Functional
Budget shall account for training	Non-Functional
Budget shall account for management	Non-Functional
Budget shall account for advertisement	Non-Functional
Budget shall account for off the shelf merchandising	Non-Functional
Budget shall account for test engineers	Non-Functional
Budget shall account for maintenance fees	Non-Functional
Hardware sensor detects the presence of all known COVID strains	Functional
Hardware sensor detects the presence of future COVID strains using a common antigen to bind to	Functional
Hardware sensor differentiates between strains of COVID using unique identifiers between strains	Functional
Bluetooth hardware connects through bluetooth signal to phone/smart watch app	Functional
Hardware sensor transfers positive or negative information through bluetooth signal to phone	Functional
Phone/smart watch app details positive/negative results to the user	Functional
Phone/smart watch app details COVID strain to the user	Functional
Phone/smart watch app details COVID related news to the user	Functional
Phone/smart watch app keeps track of user records and sends data to AWS relational database through encrypted protocol	Functional

AWS database is queried for positive test counts by location for analytics by Spark	Functional
Spark analytics program calculates positive test rates per location	Functional
Spark analytics metrics are sent to government and healthcare agencies with consent from user	Functional
Sending covid test data from the device to web UI	Functional
Test search functionality in the database	Non-Functional
Check covid test info with other symptoms using smart watch	Functional
Test duo authentication	Non-Functional
Test the app/website functionality	Non-Functional
Check what happens when the website/app goes down	Non-Functional
bluetooth connectivity	Functional
Test bluetooth location between people in the area	Non-Functional
Check the correctness on covid testing	Non-Functional
Test object movement from a quarantined location	Non-Functional
Test alert/notification function	Non-Functional
Test functionality from a smart watch	Non-Functional
Have pentesters regularly test security system to identify any vulnerabilities	Non-Functional
Stress test the app	Non-Functional
Test for possible data leaks	Non-Functional
Test cross browser compatibility	Non-Functional
Test for data recovery	Non-Functional
Beta testing, using real user devices to test functionality, vulnerability, and other measures	Non-Functional
Release in Q1 2023	Non-Functional

3 Month Jan-March Interview and consult for requirements	Non-Functional
March-May Build architecture	Non-Functional
Begin manufacturing prototypes	Non-Functional
Begin marketing	Non-Functional

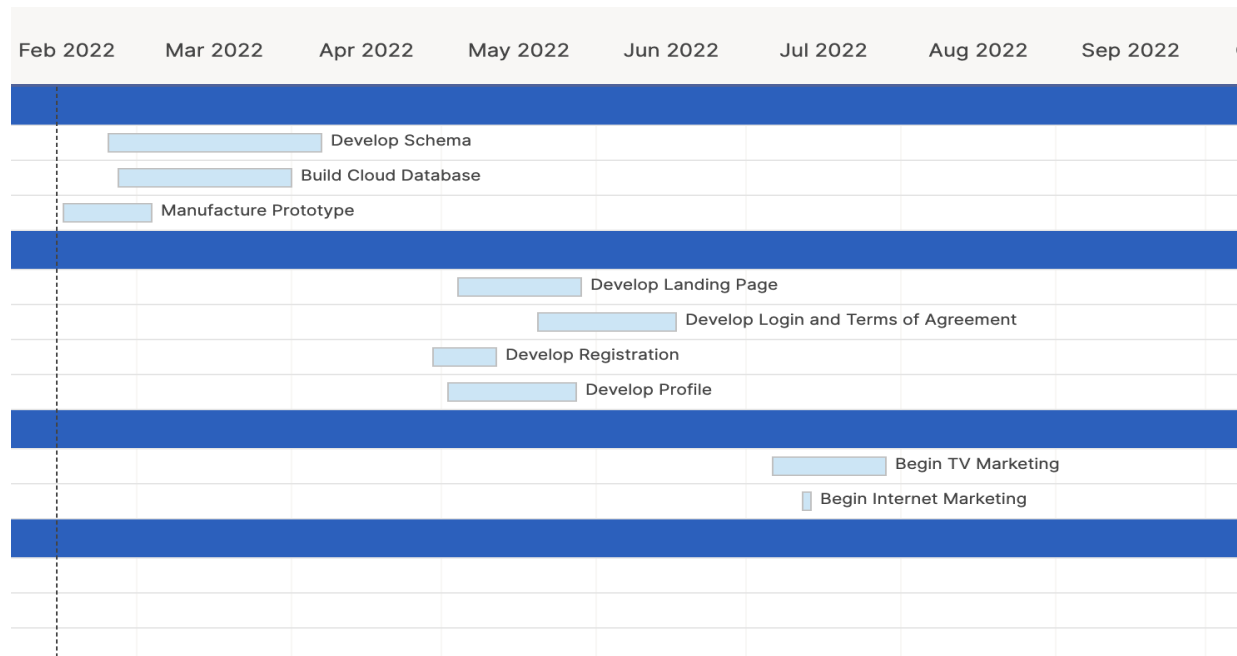
## Requirement Prioritization

- Our team met up with some of the investors from a medical company and they wanted to finance on the app development which is key
  - We need to make sure that users can send information back and forth from their device to the website/app
  - The services that we provide to all users should at least have them create accounts and store their information into our database
  - All users who use the app should be able to access information via app/website
  - Certain companies like hospitals or doctors should also be able access the user's info via their permission
- Our team made a plan to make sure that we also prioritize the right to privacy on all accounts to make sure that nothing gets leaked out
  - Our passwords shall be encrypted to make sure that accounts aren't easy to login to, plus password changes shall be required every 3 months
  - We would implement a two-factor authentication system as a way to make sure that no one would be able to access personal information without permission or identification
  - Our database and user profiles shall be protected as a way to prevent any data leaks/vulnerabilities
- Our team met up with some of the hardware developers in San Diego to show the various kinds of designs of what the covid device would look like and helped us on what functionalities should be prioritized
  - The testing device should be able to detect whether the covid test is positive or negative
  - We should also create a prototype to match the functionalities of the manual covid take-home tests
  - The hardware should also be able to adapt to all current and newer variants of COVID
  - The new variants detected shall be done via hardware device update

## BUDGET PLAN

Project Stages	Task	Assignment	Rate/Hr	No. of people	% of budget	Budget
<b>Database Development</b>		<b>Database Developers</b>	<b>\$60</b>	<b>1</b>	<b>4</b>	<b>\$1.6 Million</b>
	Choose a database type					
	Identify relationships between entity and attributes					
	Develop a schema with listed requirements					
	Setup AWS Relational Database Environment					
	List dependencies between primary and foreign keys					
	Setup tables for covid and non-covid requirements					
<b>Project Management</b>		<b>Project Manager</b>	<b>\$180</b>	<b>1</b>	<b>4</b>	<b>\$1.6 Million</b>
	Prioritize tasks based on requirement order					
	Develop Gantt Chart					
	Develop budget plan					
	Allocate resources					
	Assign different project roles to their assigned tasks					
	Purchase biosensors, Bluetooth, wifi and indicator lights					
<b>Hardware Development</b>		<b>Computer Engineer</b>	<b>\$100</b>	<b>2</b>	<b>4</b>	<b>\$1.6 Million</b>
	Setup architectural design of the covid device					
	Design and implement a basic covid test functionality					
	Prepare and execute a plan that aligns with FDA regulations					
	Assemble covid device with necessary hardware					
	Establish firmware and a simple operating system					
	Test and run a covid test on the device itself					
<b>Software Development</b>		<b>Software Engineers</b>	<b>\$120</b>	<b>3</b>	<b>4</b>	<b>\$1.6 Million</b>
	Setup a design plan for an app and a website					
	Achieve common functionalities for both artifacts					
	Develop a landing page and easy to use UI					
	Develop a registration page					
	Develop a heatmap tab					
	Develop an analytics tab					
	Develop a local news tab					
	Develop a tab for sending data to interesting parties					
	Develop special features for people with disability					
	Test the app and website functionality					
	Integrate the hardware with the app and website					
<b>Testing</b>		<b>Test Engineers</b>	<b>\$70</b>	<b>2</b>	<b>5</b>	<b>\$2.0 Million</b>
	Stress test app					
	Cross functionality test					
	Test each hardware component					
	Test each software component					
	Check sensor connectivity					
	Check front-end and back-end functionality of software					
	Test dual authentication					
	Test CRUD operations for relational database					
<b>Marketing</b>		<b>Coordinator</b>	<b>\$100</b>	<b>3</b>	<b>50</b>	<b>\$20 Million</b>
	Deploy product ads on various source of social media platforms					
	Hold marketing events in person and virtual					
	Emails					
	Television advertisements					
	On road posters					
<b>Final Rollout</b>		<b>Manufacturer</b>	<b>\$45</b>	<b>100</b>	<b>29.6</b>	<b>\$11.6 Million</b>
	Manufacture initial prototype for user feedback					
	Setup up machinery to build the product					
	Packaging costs					
	Shipping costs					
<b>TOTAL</b>				<b>121 (Manpower)</b>		<b>\$40 Million</b>

## Gantt Chart


















## Mix of Agile and Waterfall Model

To fulfill development of our product, we will apply a mix of the agile and waterfall methods.

- Our physical sensor is primarily hardware, and as a result, will benefit from one projected development cycle. For this reason, we will be employing the waterfall method in development of the physical COVID test, including the sensor and bluetooth elements.
- Because our database will likely be designed once with minor updates, we will be applying the waterfall method for development
- The mobile application, smartwatch capabilities, and analytics portion of our project will have to be updated periodically, both due to the possibility of novel COVID strains and updates for user feedback. To accommodate these regular changes, we will be using the agile method.

Estimate resource requirements and dependencies/Plan to meet model

Tasks	People Assigned	Start Date	End Date	Status Summary
<b>Q1: 2022 Planning</b>				
Develop Schema	 mark martin	02/23/22	04/06/22	
Build Cloud Database	Daman	02/25/22	03/31/22	
Manufacture Prototype	Saul	02/14/22	03/03/22	
<b>Q2: 2022 Software Development</b>				
Develop Landing Page	Chai	05/04/22	05/28/22	
Develop Login and Terms of Agreement	Chai	05/20/22	06/16/22	
Develop Registration	Dean	04/29/22	05/11/22	
Develop Profile	Saul	05/02/22	05/27/22	
<b>Q3: 2022 Marketing Campaign</b>				
Begin TV Marketing	Daman	07/06/22	07/28/22	
Begin Internet Marketing	Chai	07/13/22	07/12/22	
<b>Q4: 2022 Final Product Rollout</b>				
Begin finalizing packaging	 mark martin	11/02/22	11/04/22	
Finalize manufacturing to global market	Saul	11/09/22	12/02/22	
Create business and systems plan for next year	 mark martin	12/01/22	12/30/22	