© MMH

American International University-Bangladesh (AIUB)

Department of Computer Science
Faculty of Science & Technology (FST)

Self-Care Sanctuary

A Software Quality and Testing Project Submitted
By

Semester: Spring 22-23		**Section: I**	**Group Number: 01**	
SN	Student Name	Student ID	Contribution (CO1+CO2)	Individual Marks
1	DAS, ANIK	18-38080-2	35%	
3	RABBI, A.S.M. FAZLE	19-39714-1	35%	
23	FARABY, MD. FARHAN ISRAK	20-42861-1	15%	
26	URMI, NURSHAD JAHAN	20-43153-1	15%	

The project will be Evaluated for the following Course Outcomes

Evaluation Criteria	Total Marks (50)	
Revision History, Test Plan Identifier, Reference Materials, Problem Background, Solutions	[10 Marks]	
Requirements Specification (System feature, Quality Attributes, System Interface, Project Requirements)	[10 Marks]	
Item Not to be tested, Testing approach (Testing levels, tools, meetings), Test cases	[10 Marks]	
Item pass/fail criteria, Test deliverables, Staffing and Training, Responsibilities, Scheduling, Risk	[10 Marks]	
Approval, Format, Submission, and Defense	[10 Marks]	

Software Test Plan

for

Self-Care Sanctuary

Version 1.0 approved

Prepared by Group 01

Group 01

28/04/2023

Table of Contents

Revision History

Revision	Date	Updated by	Update Comments
0.1	2023.03.27	Anik, Rabbi	First Draft
0.2	2023.04.05	Anik, Rabbi, Faraby, Urmi	Second Draft
0.3	2023.04.12	Anik, Rabbi,	Third Draft
0.4	2023.04.18	Anik, Rabbi, Faraby	Fourth Draft
0.5	2023.04.20	Anik, Rabbi, Urmi	Fifth Draft
0.6	2023.04.21	Anik, Rabbi	Sixth Draft
0.7	2023.04.23	Anik, Rabbi	Seventh Draft
0.8	2023.04.27	Anik, Rabbi, Urmi	Eighth Draft
1.0	2023.04.28	Anik, Rabbi, Faraby, Urmi	Final

1. TEST PLAN IDENTIFIER: SCS-TP01.0

2. REFERENCES

- *wEvolve*. Available at: https://wevolvebd.org/ (Accessed: April 21, 2023).
- *The Real Food Dietitians*. Available at: https://therealfooddietitians.com/ (Accessed: April 21, 2023).
- *MyFitnessPal*. Available at: https://www.myfitnesspal.com/ (Accessed: April 21, 2023).
- *The Art of Healthy Living*. Available at: https://artofhealthyliving.com/ (Accessed: April 21, 2023).
- *Healthy Mind, healthy life*, *Mindful*. Available at: https://www.mindful.org/ (Accessed: April 21, 2023).

3. INTRODUCTION

3.1 Background to the Problem

University students face a wide range of stressors and challenges that can impact their mental and physical health. The demands of academic workloads, extracurricular activities, social pressures, managing personal responsibilities and the transition to adulthood can all contribute to feelings of stress, anxiety, and overwhelm. However, many university students struggle to prioritize their self-care and well-being due to a variety of factors. For example, they may have limited access to mental health resources, be unaware of healthy habits and stress-reduction techniques, or simply feel overwhelmed by the demands of university life.

Academic pressure: Students are expected to perform well academically, which can lead to stress, anxiety, and burnout.

Financial challenges: University tuition, living expenses, and other costs can be overwhelming for many students, leading to financial stress.

Time management: Balancing coursework, extracurricular activities, and personal responsibilities can be difficult, leaving students feeling overwhelmed and stressed.

Social pressure: University is often a time of transition and adjustment, and students may struggle to fit in socially or feel pressure to conform to certain social norms.

Mental health issues: Depression, anxiety, and other mental health issues are common among university students, and many struggle to access the support they need.

Health and wellness: Maintaining a healthy lifestyle, including eating well and getting enough exercise and sleep, can be challenging for busy university students.

Research has shown that university students are at higher risk for mental health problems such as depression, anxiety, and substance abuse than the general population. Moreover, many students do not seek help for these issues due to stigma, lack of access to resources, or a perceived lack of need.

The importance of addressing this problem is significant, as mental health issues among university students can have long-lasting effects. Left unaddressed, these issues can impact students' academic and career trajectories, their relationships, and their overall quality of life. By providing a solution that promotes mental health and well-being, this app can help reduce the negative effects of stress and anxiety among university students, and promote a culture of self-care and wellness within the university community.

3.2 Solution to the Problem

The solution we are proposing to address the problem of high levels of stress and mental health challenges among university students is a wellness and self-care app that provides a variety of resources and tools to help students manage their mental health and well-being. These resources include guided meditations, exercises, yoga, personalized journaling features, and connections with mental health professionals for virtual appointments and consultations.

We believe that this solution is particularly appropriate to solve the problem because it provides a convenient and accessible way for students to manage their mental health and well-being. The app can be used on a smartphone, which is a device that many students already use frequently, making it easy to incorporate into their daily routines. The app also provides a wide range of resources and tools that can be personalized to meet the individual needs of each student, helping them to manage their stress and anxiety in a way that works best for them.

Furthermore, the app's connection with mental health professionals provides a way for students to access professional support from the comfort of their own homes. This can help reduce the stigma associated with seeking mental health services, and make it easier for students to get the help they need.

In terms of feasibility, the solution is achievable, as the technology and resources required to develop the app are readily available. There is also a growing demand for mental health resources among university students, which suggests that there is a viable market for the app. Additionally, the app's connection with mental health professionals can provide a revenue stream through virtual consultations and appointments. The app also could be monetized through subscriptions, in-app purchases, or advertising. By offering valuable wellness resources and support, the app could attract a large user base and generate revenue. Developing a wellness and self-care app for university students could differentiate a company or organization from its competitors, especially in the education or healthcare sectors.

The software being specified is a wellness and self-care app designed for university students to manage their mental health and well-being. The purpose of the app is to provide students with a convenient and accessible way to reduce stress and anxiety, improve their mental health, increasing productivity and academic performance and promote overall well-being while balancing the demands of university life.

The app offers a variety of resources and tools, including guided meditations, exercises, yoga, personalized journaling features, and connections with mental health professionals for virtual appointments and consultations. These features are designed to help students identify patterns and triggers that impact their mental health, provide a sense of accountability and progress tracking, and ultimately promote healthy and balanced lifestyles.

The objectives and goals of the app include providing a comprehensive resource for managing mental health and well-being, increasing accessibility and reducing stigma associated with seeking mental health support, and promoting positive behaviors and habits that contribute to overall wellness.

There are several existing studies that have explored the issue of high levels of stress and mental health challenges among university students. There are a variety of mental health and wellness apps available on the market. Examples include Headspace, Calm, and Sanvello, which offer guided meditations, breathing exercises, and other stress-reducing activities.

However, many of these existing solutions focus primarily on stress reduction and mindfulness, rather than providing a comprehensive resource for managing mental health and well-being. Additionally, they may not provide personalized support or connections with mental health professionals, which can be particularly important for students who are dealing with more complex mental health challenges.

Overall, while there are some existing software solutions available to address the problem of high levels of stress and mental health challenges among university students, there is still a need for a more comprehensive and personalized resource that can effectively meet the diverse needs of this population. The proposed wellness and self-care app aims to fill this gap by offering a wide range of resources and tools, personalized support, and connections with mental health professionals, all in one convenient and accessible platform.

4. REQUEIREMNT SPECIFICATION

4.1 System Features

1. Register/Sign up

Functional Requirements

1.1 If the user doesn't have an account in our system, the system shall ask them to register first for creating a new account.
1.2 Then the system shall ask them to provide the information: name, username, password, phone number or email address.
1.3 Then to verify the user, the system shall send them an OTP number into their phone number or email address.
1.4 By providing the OTP number, the user can successfully register into our system.

Priority Level: High
Precondition**:** User must have a valid email address or phone number
Cross-reference: N/A

2. User Profile

Functional Requirements

2.1 The system shall allow the users to set their profile with their name, email, and a profile picture.
2.2 Users should be able to edit their profile information and update it in the system.

Priority Level: Medium
Precondition: User is logged in.
Cross-reference: 1

3. Guided meditations

Functional Requirements

3.1 The app shall provide users with a variety of guided meditations to choose from.
3.2 The app shall allow users to select the length of the meditation session.
3.3 The app shall provide users with audio and visual guidance during the meditation session.

Priority Level: Medium
Precondition: User has headphones and is in a quiet environment.
Cross-reference: 1,10

4. Personalized journaling

Functional Requirements

4.1 The app shall allow users to create and save journal entries about their mood, emotions, and thoughts.
4.2 The app shall provide users with prompts and suggestions for journal entries.
4.3 The app shall allow users to track their progress and identify patterns in their mental health.

Priority Level: Low
Precondition: User has access to a device with a keyboard or touch screen input.
Cross-reference: 1

5. Virtual appointments with mental health professionals

Functional Requirements

5.1 The app shall allow users to schedule virtual appointments with mental health professionals.
5.2 The app shall provide users with a list of available mental health professionals to choose from.
5.3 The app shall allow users to message mental health professionals before and after appointments.

Priority Level: High
Precondition: User has access to a stable internet connection and a device with a camera and microphone.
Cross-reference: 1

6. Routine Exercise

Functional Requirements

6.1 The app shall provide users with a variety of exercise routines to choose from, such as yoga or strength training.
6.2 The app shall allow users to select the duration and intensity of the routine exercise.
6.3 The app shall provide users with audio and visual guidance during the routine exercise.

Priority Level: Medium
Precondition: User has access to a quiet space and exercise equipment (if needed).
Cross-reference: 1

7. Water intake reminders

Functional Requirements

7.1 The app shall allow users to set reminders to drink water throughout the day.
7.2 The app shall allow users to track their water intake.
7.3 The app shall notify users with notifications when it's time to drink water.

Priority Level: Low
Precondition: User has access to a device with push notifications enabled.
Cross-reference: 1

8. Sleep hygiene tips

Functional Requirements

8.1 The app shall provide users with tips and suggestions for improving their sleep hygiene.
8.2 The app shall allow users to track their sleep patterns.
8.3 The app shall provide users with reminders and suggestions for getting better sleep.

Priority Level: Low
Precondition: User has access to a device with a clock or alarm feature.
Cross-reference: 1

9. Personalized Dashboard

Functional Requirements

9.1 The software shall provide a personalized dashboard for each user.
9.2 The dashboard shall display information such as user's goals, progress, and upcoming activities.
9.3 The dashboard shall allow users to customize their view and prioritize the information displayed.

Priority Level: Medium
Precondition: User is logged in and has access to their personalized dashboard.
Cross-reference: 1,3,6,10,11,13

10. Activity Tracking

Functional Requirements

10.1 The software shall allow users to track their physical activities such as steps taken, distance walked/run, and calories burned.
10.2 The software shall support integration with wearable devices such as fitness trackers and smartwatches.
10.3 The software shall provide users with daily, weekly, and monthly reports of their physical activity.

Priority Level: High
Precondition: User has access to a device capable of tracking physical activity.
Cross-reference: 1,6,7,12,13

11. Nutrition Tracking

Functional Requirements

11.1 The software shall allow users to log their daily food intake.
11.2 The software shall provide nutritional information for a variety of food items and meals.
11.3 The software shall allow users to set daily calorie and macro-nutrient goals and track their progress.

Priority Level: High
Precondition: User has access to nutritional information for the food they are consuming.
Cross-reference: 1,13,14

12. Integration with Wearable Devices

Functional Requirements

12.1 The system shall allow users to connect their wearable devices, such as fitness trackers and smartwatches, to track their physical activities and sleep patterns.
12.2 The system shall synchronize data from the wearable devices to provide users with a comprehensive view of their health and wellness data.
12.3 The system shall allow users to customize their wearable device settings from within the app.

Priority Level: Medium
Precondition: User has a compatible wearable device and has granted the app permission to access its data.
Cross-reference: 1,7,8,10

13. Goal Setting and Tracking

Functional Requirements

13.1 The software shall allow users to set and track their fitness and wellness goals.
13.2 The software shall provide recommendations for goals based on user preferences and fitness levels.
13.3 The software shall allow users to view their progress towards their goals through visual graphs and charts.

Priority Level: High
Precondition: User is registered and logged in.
Cross-reference: 1,10,12

14. Meal Planning and Tracking

Functional Requirements

14.1 The software shall provide a library of healthy meal options for users to choose from, including vegetarian, vegan, and other dietary restrictions.
14.2 The software shall allow users to create customized meal plans based on their goals and preferences.
14.3 The software shall track user progress and provide feedback and recommendations for improvements.

Priority Level: Medium
Precondition: User is registered and logged in.
Cross-reference: 1,11

15. Notifications

Functional Requirements

15.1 The system shall provide users with customizable notifications to remind them of their goals, achievements, and upcoming activities.
15.2 The system shall allow users to turn off or adjust the frequency of notifications as per their preference.
15.3 The system shall send push notifications to users in real-time to remind them of any urgent or time-sensitive information.

Priority Level: High
Precondition: User has granted the app permission to send notifications on their device.
Cross-reference: 7,10

16. Log out

Functional Requirements

16.1 If the user gets the service, then they can simply log out and leave by this feature.

Priority Level: Medium
Precondition: User is registered and logged in.
Cross-reference: 1

4.2 System Quality Attributes

QA 1 - **Usability**: The app shall have a user-friendly interface, allowing users to easily navigate through different features and functionalities. Users shall be able to complete common tasks within an average of **two minutes**.

QA 2 - Availability: The system shall be available almost 24/7. The system shall be at least 96.5 percent available on weekdays between 6:00 a.m. and 11pm local time. At least 98.5 percent available on holidays between 10:00 a.m. and 8:00 p.m. local time.

QA 3 - Integrity: The system can block unauthorized access**.** Our system can ensure to protect the private data. Users' personal information and data shall be stored securely with appropriate security measures.

QA 4 - Performance: Every page of our system takes less than 3 seconds to load. And every page shall load less over a 50 Kbps network connection. Login session of our system will not take more than 5 seconds.

QA 5 - Flexibility: A maintenance programmer who has at least six months of experience supporting the system can make any changes or add any features, including designing, coding and testing within 5-6 hours.

QA 6 - Efficiency: At least 35 percent of the processor capacity and RAM available to the application shall be unused at the planned peak load conditions.

QA 7 - Portability: The system is platform independent. The system can be developed by any programming languages. The system can be run on various platform like Windows, Linux, Mac or any other platform. The system will be also available on IOS and Android platform.

QA 8 - Reusability: The software input functions such as login function, register function shall be designed to be reusable in other applications.

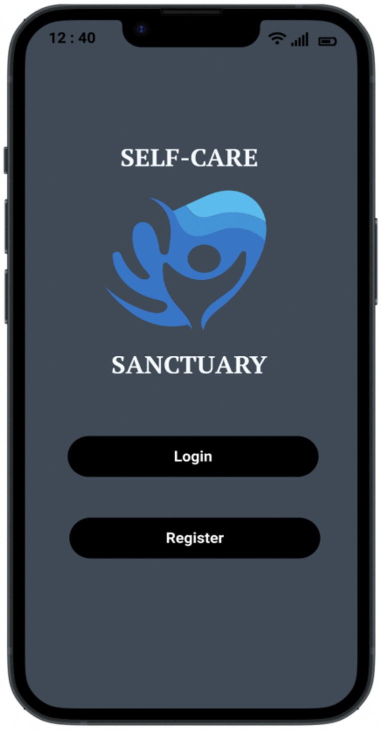
QA 9 - Maintainability: A maintenance programmer shall be able to modify any feature or solve any technical issues within 5 hours or less of development effort.

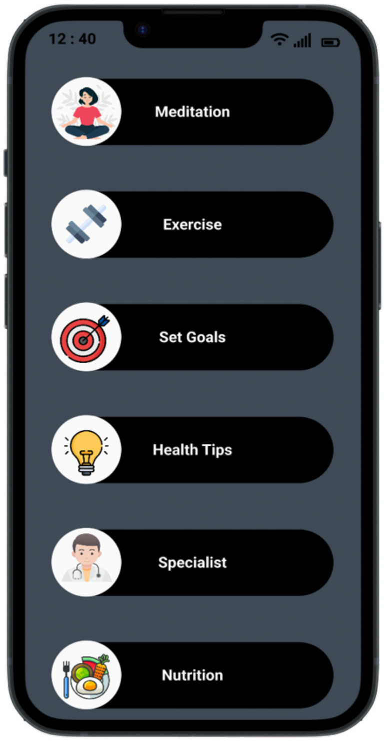
QA 10 - Testability: The maximum cyclomatic complexity* of a module shall not exceed 15.

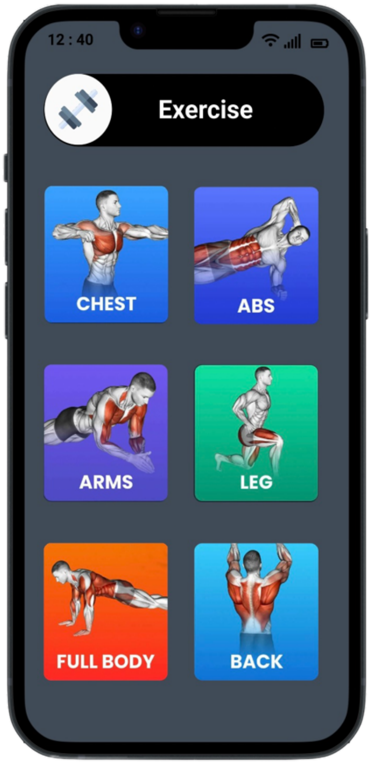
QA 11 - Reliability: In case of system failure the chance of app losing track of goals is not more than 2 in 100.

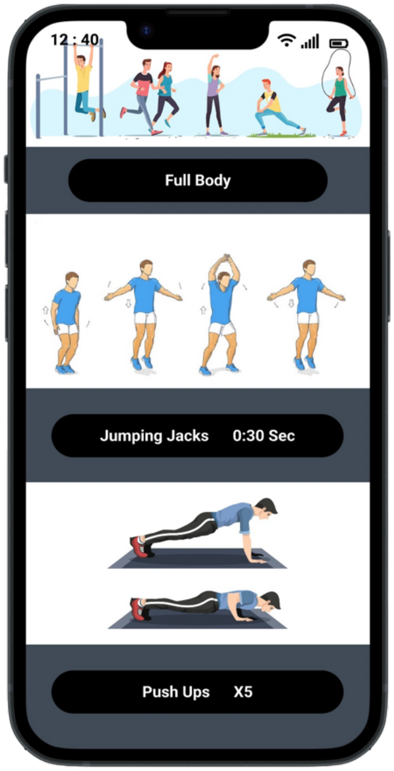
QA 12 - Interoperability: The software shall be interoperable with a wide range of devices, including fitness watches, smartwatches, and other wellness-related applications, to provide users with a seamless and integrated experience.

4.3 System Interface









4.4 Project Requirements

PM: person-months needed for project (labor working hours)

SLOC: source lines of code

P: project complexity (1.04-1.24)

DM: duration time in months for project (week days)

T: SLOC-dependent coefficient (0.32-0.38)

ST: average staffing necessary

We assume that SLOC=16*500 = 8000 [500 lines per function]

Effort=PM= Coefficient<Effort Factor>*(SLOC/1000) ^P

=3.0*(8000/1000) ^1.12

=30.80 ~31

Person-months needed for project= 5

DM = 2.50*(PM)^T

=2.50*(31) ^0.35

= 8.31~8.5

Development time = 8.5 months(approximately)

Required number of people = ST = PM/DM

=31/8.5

= 3.65 ~ 4

Developer Salary in 8.5 months:

Estimated Developers average salary per month= 30,000 BDT

All Developers salary per month =4*30,000 BDT =1,20,000 BDT

Total Developers Salary in 8.5 months = 8.5 * 1,20,000 = 10,20,000 BDT

Requirements Analysis:

Time needed =1 month

Estimated Requirement Analysis Person needed=2

Estimated Requirement Analysis Person's wage per month= 35,000 BDT

Total Requirement Analysis Expense =35,000*2 =70,000 BDT

Transportation Cost:

Estimated Cost for transportation=80,000 BDT

Training and Hardware Expense:

Estimated Cost for Training and Hardware =1,00,000 BDT

Rent Expenses:

Estimated Rent per month= 15,000 BDT

Total rent in 8.5 months =8.5*15000 BDT

=1,27,500 BDT

Utilities Cost:

Utility Expense Estimation: 1,50,000 BDT

Miscellaneous:

Total Miscellaneous cost =50,000 BDT

Total Estimated Expense:

Total Estimated cost = (10,20,000 + 70,000 + 80,000 +1,00,000 + 1,27,500 + 1,50,000 +50,000)

= 15,97,500 BDT

Resources: For developing the app, some resources shall be needed such as:

- Man power: Programmer, Developer, Tester etc.
- Device: PC, Network Connections etc.

Environment: The development team is required to be in one place. Scattering of programmers on multiple floors can cause inconvenience.

5. FEATURES NOT TO BE TESTED

The following is a list of the areas that will not be specifically addressed. All testing in these areas will be indirect because of other testing efforts. Some features that may not require extensive testing based on their nature and potential complexity. Here are some features that may not require extensive testing:

Personalized Dashboard: While the design and layout of the dashboard may require testing for usability and visual appeal, the functionality of displaying personalized data, such as activity tracking and nutrition tracking, may not require extensive testing as long as the data is accurately retrieved and displayed from the backend.

Water intake reminders: Testing requires resources, including time, personnel, and equipment. If the development team lacks the necessary resources, they may prioritize testing other critical features and may not have had the capacity to test the water intake reminders. It may have been prioritized lower in terms of testing due to limited resources or time limitations.

Sleep hygiene tips: Like water intake reminders, this feature may have been prioritized lower in terms of testing due to limited resources or time limitations.

Integration with Wearable Devices: This feature involves integrating the app with wearable devices, such as fitness trackers or smartwatches, to collect data on activities, sleep, or other health-related metrics. While the integration itself may require some testing, the basic functionalities of data collection and syncing can be tested based on the specifications of the wearable devices' APIs.

Meal Planning and Tracking: All testing for this feature will be done indirectly as a result of testing the features such as Goal Setting and Tracking, Nutrition Tracking. So, it may not require extensive testing unless there are unique requirements.

Notifications: This feature involves sending notifications to users based on predefined triggers or user settings, such as water intake reminders and sleep hygiene tips. While the content and timing of notifications may require testing, the basic functionality of sending notifications can be relatively straightforward and may not require extensive testing.

Log out: This feature involves allowing users to log out of their accounts. This is a standard feature in most apps, and its functionality is usually straightforward, requiring minimal testing.

It's important to note that the level of testing required for each feature may vary depending on the app's complexity, integration with other systems, and specific requirements.

6. TESTING APPROACH

6.1 Testing Levels

- The testing for our project will consist of Unit, System/Integration (combined) and Acceptance test levels. It is hoped that there will be at least one full time independent test person for system/integration testing. However, with the budget constraints and timeline established; most testing will be done by the test manager with the development teams' participation.

- UNIT Testing will be done by the developer and will be approved by the development team leader. Proof of unit testing (test case list, sample output, data printouts, defect information) must be provided by the programmer to the team leader before unit testing will be accepted and passed on to the test person. All unit test information will also be provided to the test person.

- SYSTEM/INTEGRATION Testing will be performed by the test manager and development team leader with assistance from the individual developers as required. No specific test tools are available for this project. Programs will enter into System/Integration test after all critical defects have been corrected. A program may have up to two Major defects as long as they do not impede testing of the program (i.e., there is a work around for the error).

- ACCEPTANCE Testing will be performed by the actual end users with the assistance of the test manager and development team leader. The acceptance test will be done in parallel with the existing process for a period of one month after completion of the System/Integration test process.

6.2 Test Tools

The only test tools to be used are the standard AS/400 provided utilities and commands.

- The Program Development Manager (PDM) will be used as the source version configuration management tool in conjunction with the in-house check-in/check-out control utility. The check-in/out utility is part of each developer’s standard AS/400 access menu.
- The initial prototypes for the new screens will be developed using the AS/400 Screen Design Aid (SDA). The initial layout and general content of the screens will be shown to the sales administration staff prior to proceeding with testing and development of the screens.
- Selenium can be used to test our project. As it is a web based automated testing tool our work will be easier and faster. The reasons behind to use Selenium are it is easy to use, it is user friendly, it accepts many programming languages, it accepts multiple OS, it is open-source platform.

6.3 Meetings

The test team will meet once every two weeks to evaluate progress to date and to identify error trends and problems as early as possible. The test team leader will meet with development and the project manager once every two weeks as well. These two meetings will be scheduled on different weeks. Additional meetings can be called as required for emergency situations. Daily meetings are organized to keep track of the progress of the development team continuously and they also serve as planning meetings: what has been done since the last meeting and what is to be done before the next one. Also, problems and other variable matters are discussed and controlled in this short (approximately 15 minutes) meeting held daily.

7. TEST CASES/TEST ITEMS

<table>
<tr><td colspan="3">Project Name: Self-Care Sanctuary</td><td colspan="2">Test Designed by: Rabbi</td></tr>
<tr><td colspan="3">Test Case ID: TC01_Registration</td><td colspan="2">Test Designed Date: 20/04/2023</td></tr>
<tr><td colspan="3">Test Priority (Low, Medium, High): High</td><td colspan="2">Test Executed by: Anik</td></tr>
<tr><td colspan="3">Module Name: Register/Sign up</td><td colspan="2">Test Execution Date: 24/04/2023</td></tr>
<tr><td colspan="3">Test Title: Verify user registration process</td><td colspan="2"></td></tr>
<tr><td colspan="5">Description: Verify that the system shall allow users to register a new account by providing the required information and verifying with an OTP number sent to their phone number or email address.</td></tr>
<tr><td colspan="5">Precondition (If any): None</td></tr>
<tr><td>Test Steps</td><td>Test Data</td><td>Expected Results</td><td>Actual Results</td><td>Status (Pass/Fail)</td></tr>
<tr><td>1. Go to the registration page
2. Enter name, username, password, phone number or email address
3. Click on "Send OTP" button
4. Enter the OTP number received in the provided phone number or email address
5. Click on "Verify OTP" button</td><td>Username: Alex123
Password: Alex@123
Phone Number: 1234567890
Email Address: alex123@email.com
OTP Number: 123456</td><td>User should be able to register and create an account</td><td>As expected,</td><td>Pass</td></tr>
<tr><td colspan="5">Post Condition: User is registered in the system and can log in to their account with the provided username and password. The account details are stored in the database.</td></tr>
</table>

<table>
<tr><td colspan="3">Project Name: Self-Care Sanctuary</td><td colspan="2">Test Designed by: Rabbi</td></tr>
<tr><td colspan="3">Test Case ID: TC02_Login</td><td colspan="2">Test Designed Date: 20/04/2023</td></tr>
<tr><td colspan="3">Test Priority (Low, Medium, High): Medium</td><td colspan="2">Test Executed by: Anik</td></tr>
<tr><td colspan="3">Module Name: Login</td><td colspan="2">Test Execution Date: 24/04/2023</td></tr>
<tr><td colspan="3">Test Title: verify login with valid username and password</td><td colspan="2"></td></tr>
<tr><td colspan="5">Description: Verify that the user can log into the app using valid user name and password</td></tr>
<tr><td colspan="5">Precondition (If any): User must have valid username and account</td></tr>
<tr><td>Test Steps</td><td>Test Data</td><td>Expected Results</td><td>Actual Results</td><td>Status (Pass/Fail)</td></tr>
<tr><td>1. Go to login
2. Enter username
3. Enter password
4. Click Login</td><td>Username: Alex123
Password: Alex@123</td><td>User should login into the application</td><td>As expected,</td><td>Pass</td></tr>
<tr><td colspan="5">Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.</td></tr>
</table>

<table>
<tr><td colspan="3">Project Name: Self-Care Sanctuary</td><td colspan="2">Test Designed by: Anik</td></tr>
<tr><td colspan="3">Test Case ID: TC03_Update_Profile</td><td colspan="2">Test Designed Date: 20/04/2023</td></tr>
<tr><td colspan="3">Test Priority (Low, Medium, High): Low</td><td colspan="2">Test Executed by: Rabbi</td></tr>
<tr><td colspan="3">Module Name: User Profile</td><td colspan="2">Test Execution Date: 24/04/2023</td></tr>
<tr><td colspan="3">Test Title: Verify user profile functionality</td><td colspan="2"></td></tr>
<tr><td colspan="5">Description: Test the functionality of the user profile feature in the app.</td></tr>
<tr><td colspan="5">Precondition (If any): User must be logged in and have access to the user profile feature.</td></tr>
<tr><td>Test Steps</td><td>Test Data</td><td>Expected Results</td><td>Actual Results</td><td>Status (Pass/Fail)</td></tr>
<tr><td>1. Navigate to the user profile section of the app.
2. Verify that the user's name, email address, and profile picture are displayed correctly.
3. Attempt to edit the user's profile information by changing the name, email address, and profile picture.
4. Save the changes and verify that the changes were saved correctly.</td><td>Name:
John Steve
Email:
johnsteve@gmail.com
Profile Picture:
[Upload any image file]</td><td>User will be able to change personal information and update profile picture</td><td>As expected,</td><td>Pass</td></tr>
<tr><td colspan="5">Post Condition: The user's profile information is updated and saved in the system. The user can continue to use the app with their updated profile information.</td></tr>
</table>

<table>
<tr><td colspan="3">Project Name: Self-Care Sanctuary</td><td colspan="2">Test Designed by: Anik</td></tr>
<tr><td colspan="3">Test Case ID: TC04_Guided_Meditation</td><td colspan="2">Test Designed Date: 20/04/2023</td></tr>
<tr><td colspan="3">Test Priority (Low, Medium, High): Medium</td><td colspan="2">Test Executed by: Rabbi</td></tr>
<tr><td colspan="3">Module Name: Guided Meditation</td><td colspan="2">Test Execution Date: 24/04/2023</td></tr>
<tr><td colspan="3">Test Title: Verify guided meditation session functionality</td><td colspan="2"></td></tr>
<tr><td colspan="5">Description: Verify that the app provides users with guided meditation functionality and allows them to select the length of the session.</td></tr>
<tr><td colspan="5">Precondition (If any): User must be logged into the app</td></tr>
<tr><td>Test Steps</td><td>Test Data</td><td>Expected Results</td><td>Actual Results</td><td>Status (Pass/Fail)</td></tr>
<tr><td>1. Open the app and navigate to the guided meditation section
2. Verify that the app provides users with a variety of guided meditations to choose from
3. Select a guided meditation and verify that the app allows users to select the length of the meditation session
4. Start the meditation session and verify that the app provides audio and visual guidance during the session
5. End the session before the selected length and verify that the app provides a prompt to end the session or continue the meditation
6. End the session after the selected length and verify that the app automatically ends the session and provides a prompt to rate the session</td><td>Meditation Type: Relaxation
Duration: 15 minutes</td><td>User will able to select meditation type, set duration and rate the session when it ends</td><td>As expected,</td><td>Pass</td></tr>
<tr><td colspan="5">Post Condition: User completes a guided meditation session and the session details are logged in the database.</td></tr>
</table>

Project Name: Self-Care Sanctuary			Test Designed by: Rabbi	
Test Case ID: TC05_ Personalized_journaling			Test Designed Date: 20/04/2023	
Test Priority (Low, Medium, High): Medium			Test Executed by: Anik	
Module Name: Personalized Journaling			Test Execution Date: 24/04/2023	
Test Title: Verify user can create and save journal entries				
Description: Test app's personalized journaling feature				
Precondition (If any): User must be logged in and have access to the journaling feature				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Log in with valid credentials 2. Click on the "Personalized Journaling" feature 3. Click on the "Create new journal entry" button 4. Write a journal entry about your mood, emotions and thoughts 5. Click on the "Save" button 6. Verify that the entry is saved successfully in the user's journal history	Journal Entry: "I felt anxious and stressed today due to upcoming exams. I tried to calm myself down by taking deep breaths and listening to music."	The app should save the journal entry and display it in the user's journal history.	As expected,	Pass
Post Condition: The app should allow the user to view and edit their saved journal entries.				

© MMH

Project Name: Self-Care Sanctuary	Test Designed by: Anik
Test Case ID: TC06_ Virtual_Appointments	Test Designed Date: 20/04/2023
Test Priority (Low, Medium, High): High	Test Executed by: Urmi
Module Name: Virtual Appointments with Professional	Test Execution Date: 24/04/2023
Test Title: Verify scheduling a virtual appointment with a mental health professional	

Description: Test if the user can successfully schedule a virtual appointment with a mental health professional

Precondition (If any): User must have an account and logged in the app.

Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Launch the app and login with valid credentials 2. Click on the Virtual Appointments menu option 3. Select the mental health professional from the list of available professionals 4. Select the date and time for the appointment 5. Provide any additional information or message to the professional, if required 6. Click on the Schedule Appointment button	Mental health professional: Steven Smith Appointment Date: 02/05/2023 Appointment Time: 04:25 PM	The user should be able to successfully schedule a virtual appointment with the selected mental health professional.	As expected,	Pass

Post Condition: The scheduled appointment details should be saved in the database and the user should receive a confirmation message. The mental health professional should receive the appointment details and any additional information/message provided by the user.

Project Name: Self-Care Sanctuary			Test Designed by: Anik	
Test Case ID: TC07_ Routine_Exercise			Test Designed Date: 20/04/2023	
Test Priority (Low, Medium, High): Medium			Test Executed by: Urmi	
Module Name: Routine Exercise			Test Execution Date: 24/04/2023	
Test Title: Verify exercise routine selection and guidance				
Description: Test if users can select and follow exercise routines				
Precondition (If any): User must have access to the app and be logged in				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Launch the app and login with valid credentials 2. Click on the "Exercise" module 3. Choose an exercise type from the available options 4. Select the desired duration and intensity level for the exercise routine 5. Click on "Start Exercise" 6. Follow the audio and visual guidance during the exercise routine 7. Complete the exercise	Exercise: Full Body Duration: 30 minutes Intensity: Beginner	User should be able to select exercise type and duration and follow the given instruction	As expected,	Pass
Post Condition: User completes the exercise routine and their progress is logged in the app				

<table>
<tr><td colspan="3">Project Name: Self-Care Sanctuary</td><td colspan="2">Test Designed by: Faraby</td></tr>
<tr><td colspan="3">Test Case ID: TC08_ Activity_Tracking</td><td colspan="2">Test Designed Date: 20/04/2023</td></tr>
<tr><td colspan="3">Test Priority (Low, Medium, High): High</td><td colspan="2">Test Executed by: Rabbi</td></tr>
<tr><td colspan="3">Module Name: Activity Tracking</td><td colspan="2">Test Execution Date: 24/04/2023</td></tr>
<tr><td colspan="3">Test Title: Verify Activity Tracking Functionality</td><td colspan="2"></td></tr>
<tr><td colspan="5">Description: Test the activity tracking feature of the app.</td></tr>
<tr><td colspan="5">Precondition (If any): User must have a device that can track physical activity.</td></tr>
<tr><td>Test Steps</td><td>Test Data</td><td>Expected Results</td><td>Actual Results</td><td>Status (Pass/Fail)</td></tr>
<tr><td>1. Navigate to the Activity Tracking module.
2. Ensure that the app is able to integrate with the device that tracks physical activity.
3. Verify that the app is displaying the user's physical activity data accurately.
4. Check if the app is able to calculate the number of steps taken, distance walked/run, and calories burned.
5. Check if the app is providing daily, weekly, and monthly reports of the physical activity data.
6. Verify if the user can set goals related to physical activity and the app is able to track progress towards those goals.</td><td>Walk 20 minutes</td><td>1. The app should display accurate physical activity data
2. The app should provide daily, weekly, and monthly reports of the physical activity data.
3. The app should be able to track progress towards user-defined goals related to physical activity.</td><td>As expected,</td><td>Pass</td></tr>
<tr><td colspan="5">Post Condition: The app should accurately track physical activity data and provide reports and goal tracking for users.</td></tr>
</table>

<table>
<tr><td colspan="3">Project Name: Self-Care Sanctuary</td><td colspan="2">Test Designed by: Faraby</td></tr>
<tr><td colspan="3">Test Case ID: TC09_ Nutrition_Tracking</td><td colspan="2">Test Designed Date: 20/04/2023</td></tr>
<tr><td colspan="3">Test Priority (Low, Medium, High): Medium</td><td colspan="2">Test Executed by: Rabbi</td></tr>
<tr><td colspan="3">Module Name: Nutrition Tracking</td><td colspan="2">Test Execution Date: 24/04/2023</td></tr>
<tr><td colspan="3">Test Title: Verify that user can log their daily food intake</td><td colspan="2"></td></tr>
<tr><td colspan="5">Description: Test the functionality of the app to log daily food intake and track progress towards nutritional goals.</td></tr>
<tr><td colspan="5">Precondition (If any): User must be registered and logged into the app</td></tr>
<tr><td>Test Steps</td><td>Test Data</td><td>Expected Results</td><td>Actual Results</td><td>Status (Pass/Fail)</td></tr>
<tr><td>1. Navigate to the nutrition tracking module
2. Enter the details of the food item such as name, brand (if applicable), serving size, and quantity consumed.
3. Verify that nutritional information is displayed such as calories, macro-nutrient breakdown, and serving size.
4. Save the log entry
5. Verify that the log entry is displayed in the user's daily food diary.</td><td>Food Item: Grilled Chicken
Brand: None
Serving Size: 4 oz
Quantity: 1</td><td>Nutritional information is displayed correctly for the entered food item
The log entry is successfully saved and displayed in the user's daily food diary</td><td>As expected,</td><td>Pass</td></tr>
<tr><td colspan="5">Post Condition: User's daily food diary is updated with the logged entry and progress towards nutritional goals is tracked accordingly.</td></tr>
</table>

© MMH

<table>
<tr><td colspan="2">Project Name: Self-Care Sanctuary</td><td colspan="3">Test Designed by: Rabbi</td></tr>
<tr><td colspan="2">Test Case ID: TC10_ Goals</td><td colspan="3">Test Designed Date: 20/04/2023</td></tr>
<tr><td colspan="2">Test Priority (Low, Medium, High): High</td><td colspan="3">Test Executed by: Anik</td></tr>
<tr><td colspan="2">Module Name: Goal Setting and Tracking</td><td colspan="3">Test Execution Date: 24/04/2023</td></tr>
<tr><td colspan="2">Test Title: Verify user is able to set and track fitness and wellness goals</td><td colspan="3"></td></tr>
<tr><td colspan="5">Description: This test case verifies if the user is able to set and track their fitness and wellness goals successfully in the app.</td></tr>
<tr><td colspan="5">Precondition (If any): User must be registered and logged into the app</td></tr>
<tr><td>Test Steps</td><td>Test Data</td><td>Expected Results</td><td>Actual Results</td><td>Status (Pass/Fail)</td></tr>
<tr><td>1. Go to "Goals" option
2. Click on the "Add Goal" button
3. Enter a goal name
4. Select a category for the goal (fitness, nutrition, sleep, etc.)
5. Set a goal target (e.g. run 5 km in 30 minutes, eat 5 servings of vegetables daily)
6. Set a deadline for the goal
7. Click on the "Save" button
8. Perform activities to work towards achieving the goal
9. Update the progress towards the goal by clicking on the "Update Progress" button
10. Enter the progress made towards the goal (e.g. distance covered, servings consumed)
11. Click on the "Save" button
12. Verify that the goal has been marked as achieved if the user has met the target before the deadline</td><td>Goal Name:
Run 5 km in 30 minutes
Category:
Fitness
Target:
Run 5 km in 30 minutes
Deadline:
30 days from today's date</td><td>The user should be able to add the goal successfully
The progress towards the goal should be updated successfully</td><td>As expected,</td><td>Pass</td></tr>
<tr><td colspan="5">Post Condition: The goal is displayed on the user's dashboard and progress can be viewed through visual graphs and charts.</td></tr>
</table>

8. ITEM PASS/FAIL CRITERIA

Item pass/fail criteria for test cases are typically based on the expected behavior or requirements of the system or application being tested. These criteria are established in advance to determine whether a test case has passed or failed based on the actual outcome observed during testing. Here are some examples of item pass/fail criteria for test cases:

Functional Requirements: Test cases may have pass/fail criteria based on the functional requirements of the system or application being tested. For example, if a requirement states that a "Submit" button should save data to the database, the pass criterion for a test case related to this requirement may be that the data is successfully saved to the database upon clicking the "Submit" button, and the fail criterion may be if the data is not saved or saved incorrectly.

Expected Outputs: Test cases may have pass/fail criteria based on the expected outputs of the system or application being tested. For example, if a test case is designed to verify the calculation of a discount based on certain inputs, the pass criterion may be that the calculated discount matches the expected value, and the fail criterion may be if the calculated discount does not match the expected value.

Error Handling: Test cases may have pass/fail criteria based on error handling requirements. For example, if a system is expected to display an error message when invalid input is entered, the pass criterion for a test case related to this requirement may be that the error message is displayed as expected, and the fail criterion may be if no error message is displayed or an incorrect error message is displayed.

Performance Benchmarks: Test cases may have pass/fail criteria based on performance benchmarks or thresholds. For example, if a performance test is designed to evaluate the response time of a page, the pass criterion may be that the response time is within an acceptable range, and the failure criterion maybe if the response time exceeds the acceptable range.

Compatibility: Test cases may have pass/fail criteria based on compatibility requirements, such as cross-browser or cross-device compatibility. For example, if a test case is designed to verify the functionality of a web application on different browsers, the pass criterion may be that the application functions correctly on all supported browsers, and the fail criterion may be if the application fails to function on any of the supported browsers.

It's important to define clear and objective pass/fail criteria for test cases to ensure consistency and accuracy in evaluating the results of testing efforts. These criteria should be established in alignment with the requirements and expected behavior of the system or application being tested, and should be communicated to the testing team to ensure a common understanding of the expectations for test case outcomes.

9. TEST DELIVERABLES

- Acceptance test plan
- System/Integration test plan
- Unit test plans/turnover documentation
- Screen prototypes
- Report mock-ups
- Defect/Incident reports and summaries
- Test logs and turnover reports

10. STAFFING AND TRAINING NEEDS

It is recommended that the project team consists of the following roles:

- Project Manager
- Software Developers
- UX/UI Designer
- Quality Assurance Tester
- Database Administrator

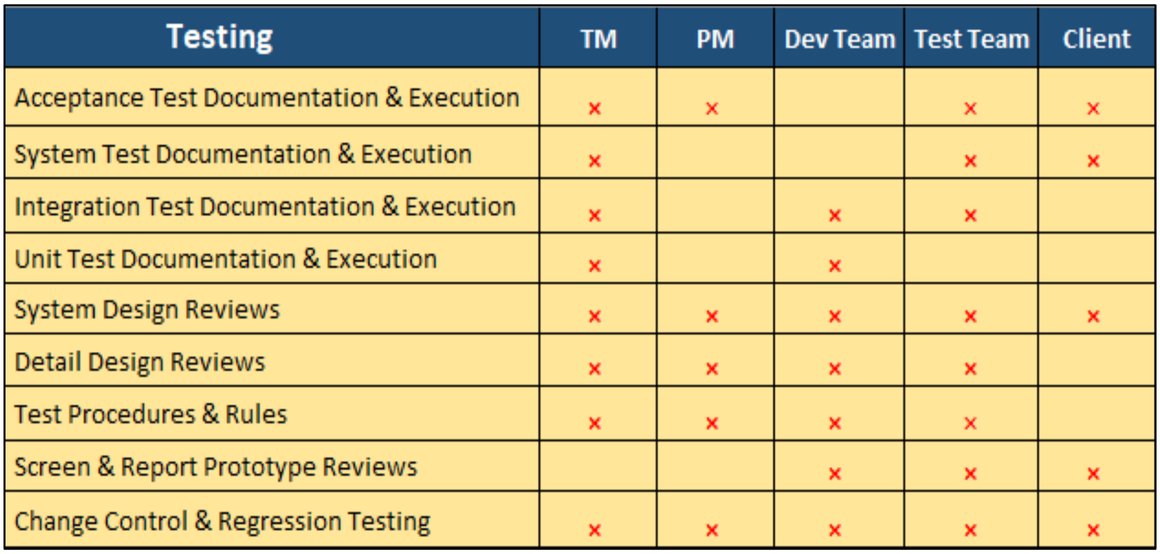
During the development and testing phases, it is preferred that the Quality Assurance Tester works on the project on a full-time basis to ensure that the app is thoroughly tested before release. The tester should be assigned part-time at the beginning of the project to participate in reviews and provide feedback. Approximately half way into the project, the tester should be assigned full-time to carry out system/integration and acceptance testing.

In order to provide complete and proper testing, the following areas need to be addressed in terms of training:

- The developers and testers will need to be trained on the functionality and features of the app, including the login and authentication process, account recovery, personalized dashboard, wellness resources, journaling feature, mental health professional connection, nutritional advice, water intake reminders, self-care tips, goal setting, notifications and personalized recommendations.
- The UX/UI Designer should receive training on designing mobile apps with a focus on user-centered design principles.
- The Database Administrator should be trained on the specific database used for the app and how to manage and maintain it.

Additionally, once the app is released, the customer support staff should be trained on how to troubleshoot any issues that users may encounter and how to provide assistance with the app's features.

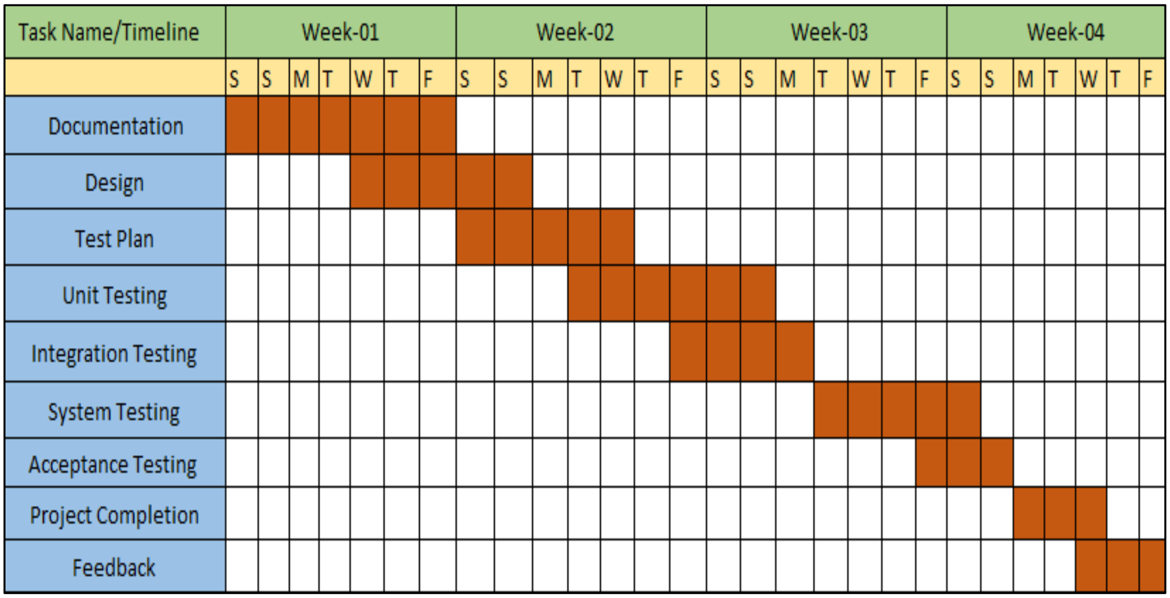
11. RESPONSIBILITIES



Testing	TM	PM	Dev Team	Test Team	Client
Acceptance Test Documentation & Execution	×	×		×	×
System Test Documentation & Execution	×			×	×
Integration Test Documentation & Execution	×		×	×	
Unit Test Documentation & Execution	×		×		
System Design Reviews	×	×	×	×	×
Detail Design Reviews	×	×	×	×	
Test Procedures & Rules	×	×	×	×	
Screen & Report Prototype Reviews			×	×	×
Change Control & Regression Testing	×	×	×	×	×

12. TESTING SCHEDULE

Time has been allocated within the project plan for the following testing activities. The specific dates and times for each activity are defined in the project plan timeline. The persons required for each process are detailed in the project timeline and plan as well. Coordination of the personnel required for each task, test team, development team, management and customer will be handled by the project manager in conjunction with the development and test team leaders.



Task Name/Timeline	Week-01							Week-02							Week-03							Week-04						
	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F
Documentation	■	■	■	■	■	■	■																					
Design					■	■	■	■	■																			
Test Plan								■	■	■	■	■																
Unit Testing											■	■	■	■	■	■												
Integration Testing														■	■	■	■											
System Testing																		■	■	■	■	■						
Acceptance Testing																					■	■	■					
Project Completion																								■	■	■		
Feedback																										■	■	■

13. PLANNING RISKS AND CONTINGENCIES

- Risk: Delays in the development process due to unforeseen technical challenges.

  Contingency: Have a dedicated team of developers who can address technical challenges in a timely manner. Develop a comprehensive plan with contingency time to account for delays.

- Risk: Budget overrun due to unanticipated expenses.

  Contingency: Have a detailed budget plan that accounts for potential expenses, and establish a process to monitor expenses throughout the project. Identify and prioritize the critical components of the app and focus on those first, deferring less critical features until later.

- Risk: Inadequate resources, such as staff or equipment, to meet project deadlines.

  Contingency: Conduct a thorough analysis of staffing and equipment needs, and ensure that they are available when needed. Establish a contingency plan that includes temporary staff augmentation, outsourcing, or other options to address resource shortfalls.

- Risk: Unforeseen security vulnerabilities that put user data at risk.

  Contingency: Have a dedicated security team that can monitor and address vulnerabilities in real time. Conduct regular security assessments and testing to identify potential weaknesses, and ensure that all necessary security measures are in place.

- Risk: Poor user adoption due to lack of marketing and promotion.

  Contingency: Develop a comprehensive marketing and promotion plan that includes targeted outreach to potential users, social media campaigns, and other strategies to increase awareness of the app. Conduct user surveys and gather feedback to refine and improve the app over time.

14. APROVALS

Project Sponsor – DR. Mohammad Mahmudul Hasan	
Development Management – Anik Das	
EDI Project Manager – A.S.M. Fazle Rabbi	
RS Test Manager - Nurshad Jahan Urmi	
RS Development Team Manager – Sakib Ahmed	
Reassigned Sales - MD. Farhan Israk Faraby	
Order Entry EDI Team Manager – Mehedi Hassan	