Hazard Analysis
REVITALIZE

Team 13, REVITALIZE
Bill Nguyen
Syed Bokhari
Hasan Kibria
Youssef Dahab
Logan Brown
Mahmoud Anklis

Table 1: Revision History

Date	**Developer(s)**	**Change**
October 15th, 2022	Bill Nguyen	FMEA
October 19th, 2022	Youssef Dahab	Intro, Purpose, Scope, Roadmap
October 19th, 2022	Logan Brown	System Boundary, Components, and Critical Assumptions
October 19th, 2022	Syed Bokhari	FMEA
October 19th, 2022	Hasan Kibriai	FMEA
October 19th, 2022	Mahmoud Anklis	Safety and Security Requirements

Contents

1 Introduction

This document is a hazard analysis of REVITALIZE.

1.1 Hazard Definition

As per Nancy Leveson's work, a hazard is a property or condition in a system together with a condition in the environment that has the potential to cause harm or damage. In REVITALIZE, there are safety (keeping records) and security (restricting access to data) hazards.

1.2 Other Used Terms in this Document

1.2.1 Failure

A failure in the REVITALIZE mobile application could occur when there is a deviation between the actual and expected output. Furthermore, a failure could also occur when a certain state causes REVITALIZE, or a component of REVITALIZE (login, database, etc.), to fail and therefore not achieve its necessary function.

1.2.2 Safety

REVITALIZE's safety is its freedom from harm. However, it's not an absolute. Safety is a global property of REVITALIZE. Therefore, when two or more REVITALIZE components interact together, the resulting emergent behaviour may or may not be safe.

2 Purpose of Hazard Analysis and Scope

Hence, the purpose of this document is to identify hazards pertaining to the REVITALIZE mobile application, and their causes, and then specify ways to eliminate them or mitigate their effect. The team cannot "bolt on" safety after implementing the application. Therefore, by identifying hazards and developing safety requirements before implementing, REVITALIZE team members will be able to react when their application fails to be safe. This becomes beneficial when something "bad" happens. The team will have a recovery plan.

The scope of this project is to create an all in one health and wellness mobile application that allows users to manage their diet, exercise, and sleep by providing them with meal recipe's based on their nutritional preferences, a personalized workouts planner and a sleep tracker.

3 System Boundaries and Components

3.1 System Boundaries

The system boundary can be outlined using the following elements of the system:

1. Main application (with the following subsystems)
    - Login/Authentication
    - Database
    - Backend server
    - Main calendar
    - Diet menu
    - Workout menu
    - Sleep menu
2. APIs
    - Recipe API
    - Workout tracker API
    - Google sleep tracker API
3. Android device (device app is installed on)

The system boundary includes the application as a whole, the APIs used for its function, and the Android device the app is installed onto. Some of these elements are outside the direct control of REVITALIZE, namely the APIs, server uptime, database uptime, and the Android device itself. However, they are important elements of the system that must be taken into account for proper hazard analysis of the system.

3.2 Components

The components of the system are outlined as followed:

3.2.1 Login/Authentication System

System that allows users to create an account, login, and have their credentials verified. Users can then access their data and modify health goals and plans.

3.2.2 Database

System that stores and organizes user health data. Data will be stored to the user's account and is accessible to any device that logs in with said account.

3.2.3 Server Backend

System that controls the flow of data between the APIs and the application.

3.2.4 Main Calendar Interface

System that displays current diet, workout, and sleep goals/plans for selected day. Users can select a desired day on the calendar and navigate to the diet, workout, and sleep interfaces which are used to modify user goals and plans.

3.2.5 Diet Section Interface

System that allows users to search and add recipes aligned with their dietary goals. The recipes chosen by the user are then added to the plan of the selected day. Previously planned recipes are stored for easy access in the future.

3.2.6 Workout Section Interface

System that allows users to add existing or custom workouts to the selected calendar day. Previous workouts are stored for easy access in the future.

3.2.7 Sleep Section Interface

System that tracks and displays user sleep data. Also displays sleep/wake-up times based on user sleep goals and habits.

4 Critical Assumptions

1. Assume user is not intentionally acting in a malicious way
2. Assume user is not sharing username and password info with anyone else
3. Assume user's Android device is functioning properly
4. Assume user's Android device does not contain any vulnerabilities
5. Assume APIs are up and function properly
6. Assume APIs do not contain any vulnerabilities

5 Failure Mode and Effect Analysis

The next pages will show the full failure mode and effect analysis (FMEA) for REVITALIZE:

Component	Failure Modes	Effects of Failure	Causes of Failure	Recommended Actions	SR	Ref
Login	Login failed with invalid credentials	User cannot login and use/view app	Invalid login credentials	Display error message that is detailed and message that all users can understand After multiple failed attempts, send message to reset user's credentials	SR-1 SR-2 SR-5	H1-1
	Login failed with valid credentials	User cannot login and use/view app	Account locked temporarily Credentials are expired Mismatch of credentials in user input and database	Check if account is locked and if appropriate unlock account Wait for account to be unlocked Check if the credentials inputted by user matches credentials in database	SR-3	H1-2
	Unauthorized login by invalid user	Invalid user can login and use/view/edit app where they do not have permission to do so	Illegal access to credentials that are gathered from potential tactics such as phishing, leaks, hackages etc.	Rollback changes made by invalid user to previous state Restrict access to invalid user and fix account permissions Flag issues that can be related to phishing, leaks, hackages etc.	SR-4	H1-3
	Login database/server down	Percentage of users ranging from 0-100% cannot access app	All technologies/services related to login (ex. Backend services, database) is down	Have constant checks of overall health of related services Have alarms to inform when these issues have arised	SR-5	H1-4
User Account	Backend services are compromised	Potential loss of user data	Database services are compromised/manipulated Failures such as memory errors, service crashes etc.	Have alarm and monitor for potential compromises and make the appropriate changes/rollbacks when needed	SR-5	H2-1
	Spam accounts	Can cause an unnecessary usage and an excess of users that will take available resources from actual users	DDOS/Cyber attacks on app usually from automated programs	Adding potential CAPTCHAs and validation checks to prove that this is real user and not a "bot"	SR-6	H2-2
Database	Data is deleted	Percentage of data ranging from 0-100% will be lost	Failures related to the database Cyber attacks Invalid access to database services	Have backup databases Rollback changes to previous state where data was not deleted Have preventive measures to make sure data is not deleted unintentionally	SR-4	H3-1
	Unable to use database	Users and all stakeholders are unable to use app data	Database/server is down/locked Database connection is lost Other causes refer to H3-1	Display error message that is detailed and message that all users can understand Give potential timeline to user of when issue can be fixed Have preventive measures to avoid this issue	SR-5	H3-2
	Unable to add new user data	Inaccurate user data and functionality of app being compromised	Refer to H3-1 and H3-2	Refer to H3-2	SR-5	H3-3

Figure 1: Part 1 of FMEA

Component	Failure Modes	Effects of Failure	Causes of Failure	Recommended Actions	SR	Ref
General	Unplanned app closure	Unsaved data will be lost, app page being viewed before closure will be unknown and not necessarily retrieved upon app reboot	Low storage space on android machine, too many apps running simultaneously on processor	Automatically and frequently store user's current location and app-progress in their machine's local cache	SR-9	H4-1
Privacy	Leaked user data to a 3rd party	Data of user(s) is leaked from database storage into the hands of a 3rd party	Poor security in database storage and amidst server-initiated data trafficking	Use cloud-based databasing supported by large companies that handle their database security well Deploy backend code to a secured server location ie. AWS EC2 or something like that	SR-10	H5-1
	Private user data shown to developing team	User's private data regarding their everyday habits will become exposed to developers in the team	Improper data abstraction and database access-related rules	Store the data in a manner which hinders all developers from seeing certain personal information i.e. by using text encryption and narrowing access to certain parts of the database down to a few individuals	SR-11	H5-2
User Calendar	Information for correct date does not load	The corresponding information for user-selected date doesn't display, and instead nothing or the wrong information is displayed	Faulty business logic in data retrieval code, faulty display functionality in the front end	Use targeted end-to-end testing to ensure that the correct corresponding data for a selected date is displayed by testing various use cases and size-differing pieces of data.	SR-12	H6-1
	Clicking on a sub-page of the application doesn't perform as expected	Clicking on a sub-page of the application takes user no where or takes them to the wrong page	Faulty front-end routing functionality	Rigorous testing ensuring correct page routing through a multitude of use cases.	SR-12	H6-2

Figure 2: Part 2 of FMEA

Component	Failure Modes	Effects of Failure	Causes of Failure	Recommended Actions	SR	Ref
Recipe	Recipe API failure	User cannot access list of recipes	API connection issues prevent loading of list data	Display error message that is detailed that all users can understand Load static recipe data from prior session that is saved in database	SR-5	H7-1
	Add recipe actions not tracked in Database	User cannot add recipe to calendar	Unable to add new user data in Database Refer to H3-3	Display error message that is detailed that all users can understand Give potential timeline to user of when issue can be fixed	SR-5	H7-2
	Cannot load user recipe data	User cannot view prior recipe list and tracked calendar entries	Unable to use database data particular to user account Refer to H3-2	Display error message that is detailed that all users can understand Give potential timeline to user of when issue can be fixed	SR-5	H7-3
Workout	Exercise list unable to load	User is not able to choose from an exercise list when adding an exercise	Unable to use database Refer to H3-2	Display error message that is detailed that all users can understand Give potential timeline to user of when issue can be fixed	SR-5	H8-1
	Cannot load user exercise data	User cannot view prior exercise list and tracked calendar entries	Unable to use database particular to user account Refer to H3-2	Display error message that is detailed that all users can understand Give potential timeline to user of when issue can be fixed	SR-5	H8-2
Rest	Sleep API failure	User sleep was not correctly tracked	Sleep API incorrectly tracked user sleep or missed sleep entry	Display error message that is detailed that all users can understand Allow option for user to manually change sleep entry for accurate tracking	SR-5 SR-8	H9-1
	Cannot load user sleep data	User cannot view prior sleep calendar entries	Unable to use database particular to user account Refer to H3-2	Display error message that is detailed that all users can understand Give potential timeline to user of when issue can be fixed	SR-5	H9-2

Figure 3: Part 3 of FMEA

6 Safety and Security Requirements

1. The system should check if the credentials inputted by the user match the credentials in the database.
    (a) Rationale: The user should be able to access their account if the credentials are inputted correctly (i.e matches the database).
    (b) Associated Hazards
        i. H1-1
2. The system should allow users to reset credentials if a user requests it.
    (a) Rationale: In the case that a user forgets their password or username, they should be able to retrieve their account be resetting the credentials.
    (b) Associated Hazards
        i. H1-1
3. The system should lock a user's account when the user has inputted the incorrect password 5 times in a row.
    (a) Rationale: To ensure that the account is secured, the account of a user will get locked if multiple unsuccessful login attempts occur.
    (b) Associated Hazards
        i. H1-2
4. The system should be able to roll back changes.
    (a) Rationale: To ensure the user does not lose their data if an illegal access or a system failure occur and can continue using the application.
    (b) Associated Hazards
        i. H1-3, H3-1
5. The system notifies users when the system is down (planned outings or unexpected outings) or when an error occurs and when the system will be back online.
    (a) Rationale: Users can avoid using the application or parts of the application during these times.
    (b) Associated Hazards
        i. H1-1, H1-4, H2-1, H3-2, H3-3, H7-1, H7-2, H7-3, H8-1, H8-2, H9-1, H9-2
6. The system should limit the number of bots using the application.
    (a) Rationale: The application is intended for human users.

(b) Associated Hazards

i. H2-2

7. The system will have daily backups to multiple databases.

(a) Rationale: Redundancy in data will ensure the safety of the data and protect it from being lost.

(b) Associated Hazards

i. H3-1

8. The system allows the user to manually input sleep information.

(a) Rationale: The sleep functionality can still be used in the case of a sleep API failure.

(b) Associated Hazards

i. H9-1

9. The system should store app-progress information on the user's local machine.

(a) Rationale: Easier and quicker to access information if stored in machine's local cache.

(b) Associated Hazards

i. H4-1

10. The system should use cloud-based databases.

(a) Rationale: Provides greater security to users' data.

(b) Associated Hazards

i. H5-1

11. The system will use encryption when storing users' information.

(a) Rationale: Provides security for users' data and ensures developers cannot see users' data.

(b) Associated Hazards

i. H5-2

12. The system will incorporate rigorous weekly testing.

(a) Rationale: Ensures proper functionality.

(b) Associated Hazards

i. H6-1, H6-2

7 Roadmap

The REVITALIZE team's hazard mitigation strategy began first with identifying potential hazards and then creating new safety and security requirements to mitigate those hazards. This has been accomplished in the previous two sections. After that, each hazard was mapped to its corresponding safety or security requirement. Many of the listed hazards have a medium or high chance of occurring. Hence, SR-1 to SR-5, SR-8, and SR-10 to SR-12 will be implemented as part of the capstone timeline. H2-2, H3-1 and H4-1 have a lower probability of occurring. Therefore, SR-6, SR-7 and SR-9 will be implemented in the future. The last steps in the REVITALIZE team's hazard mitigation strategy entail developing the application and testing to verify that safety and security requirements have been met. If they have, then the hazards corresponding to that particular safety or security requirement have been successfully mitigated. The team is aware that future potential vulnerabilities could arise where they will have to be assessed based on severity and probability of occurrence.