

SPL2 Project Proposal Form, 2024
Institute of Information Technology (IIT)
University of Dhaka

Student's Name:	1. Md. Mahfuz Ibne Ali Ayon 2. Mohammad Ismail Hossain		
Student's Roll:	1. BSSE 1421 2. BSSE 1433	Phone:	1. 01782387550 2. 01537790899

Project Title: Personal Health Tracker using Automated Report Scanning

Motivation: In today's busy world, being regularly updated about personal health is a challenge for most people. But currently the ability to understand, enable early detection and preventative health care and track personal health information is more important than ever. The motivation behind developing this system stems from the realization that people often ignore to maintain their medical records, and this habit can have serious consequences for their health. Unfortunately, people often lose their reports, which can cause delays in the detection of major health issues. Additionally, individuals miss out on the opportunity to track their own health trends and become aware about their health condition. Besides using this application in personal health monitoring, this application can also be used to efficiently manage health care systems globally and locally.

Project Description: The **Personal Health Tracker** is a comprehensive mobile application designed to help users actively manage and monitor their health with ease and efficiency. The app allows users to upload images of their medical diagnostic reports (text and numeric value based test report e.g., blood test), which are then automatically processed to extract important health data such as test names, values, and results by using Optical Character Recognition (OCR) technology. This automation reduces the need for manual data entry, ensuring more accurate and better user experience for users.

Once the data is extracted, the app provides personalized health insights, summarizing the report in a simple, understandable format. It helps users interpret their diagnostic results, identify health trends over time and provides visual analysis of health records over time. Thus the **Personal Health Tracker** empowers users to stay updated about their health status and make data-driven decisions.

In addition, the app serves as a personal health assistant by rapidly reminding users to update their health data on a regular basis. The app will notify users to seek treatment from a healthcare professional to improve their condition if unexpected values or patterns are detected in a report. Regular interaction on the application encourages users to develop a proactive approach towards maintaining their health, raising awareness about personal health conditions, facilitating early detection and avoidance of potential health issues.

Along with monitoring an individual's health, the app provides a news and recommendation section where users may get updates on healthier lifestyle techniques, healthcare news, and general health advice.

Functional Requirements:

1. User Authentication

The app will allow users to securely sign up, log in, and manage their accounts.

2. User Profile Management

Users will be able to create and update personal profiles with their health history and diagnostic reports.

3. Medical Diagnostic Report Upload

The app must support uploading images of medical reports (text and numeric value type test report) directly from the user device's camera or gallery.

4. Quick Upload and Processing

Uploaded reports should be processed and the data extracted without significant delay.

5. Automated Data Entry

The app must provide the scope to automatically extract diagnosis data from uploaded medical reports.

6. Accurately Data Extraction

Automated data extraction must be accurate as much as possible. Besides manual editing will also be possible by users.

7. Summarize Health Report

Users will get a summary of his medical report in an easily understandable format.

8. Diagnosis History Visualization

Users should be able to view their health data history (e.g., test results) via visual elements such as bar charts.

9. Report Storage and Retrieval

All medical reports and extracted data should be securely stored and easily retrievable by users at any time.

10. Report Organization

Users must be able to organize and categorize medical reports (e.g., by date or type of test).

11. Search Functionality

The app must allow users to search for specific reports by test name or date.

12. Data Security

The app must ensure data encryption, password protection, and overall security of

personal and medical information.

13. Health Notifications

The app must send reminders to users to update their health status and issue alerts if any concerning health metrics are detected.

14. User-Friendly Interface

The app must provide an intuitive and clean UI for easy navigation, report uploading, and health data management.

Strategy & Timeline:

Method of Development:

1. Research and Requirement Gathering:

- Understanding the current healthcare system and user's habits on managing personal health.
- Conduct study on Image processing & OCR technology.
- Conduct study on existing OCR technologies to evaluate their accuracy and suitability for medical document extraction.
- Identify the types of medical data commonly found in reports (e.g., blood test results) and standard health metrics to compare against.
- Gather user requirements by understanding how patients or healthcare providers would benefit from the system's features.
- Define the final list of features.

2. Technology and Tool Selection:

- Image Processing: Use some Image processing libraries (e.g., OpenCV) to make the data extraction process more accurate.
- OCR Module: Integrate a reliable OCR library (e.g., Google ML Kit, Tesseract) for text extraction from medical reports.
- Database: Use Firebase for real-time, scalable data storage, handling both extracted data and historical medical records.
- Back-end deploy: Deploy back-end functionalities on a cloud server (e.g., Render).
- Statistical Analysis: Implement algorithms or integrate libraries (e.g. NumPy) to perform statistical comparisons and generate health insights.
- Mobile Development Framework: Use Flutter for cross-platform app development, ensuring seamless image capturing from camera or gallery using Flutter package (e.g., Image Picker).

3. Development Phases:

- Phase 1: Frontend Development
 - Create a user-friendly interface using Dart language and Flutter framework that allows easy interaction for uploading images, viewing extracted data, and accessing insights.
 - Ensure smooth navigation and proper display of health reports, predictions, notifications.
- Phase 2: Backend Development
 - Build the core OCR functionality with image preprocessing using the image processing library.
 - Develop an efficient way to extract structured data from the extracted unstructured data (diagnosis parameters, value).
 - Deploy the functionality on a backend server and create an API to use it.
- Phase 3: Database Integration using Firebase
 - Ensure extracted data is stored in the database for future reference and analysis.
 - Develop a user authentication system using Firebase authentication tools.
- Phase 4: Develop additional functionalities
 - Develop a prediction system that will give predictions to the user about his health risk.
 - Develop a notification system to give warnings or reminders to users.
 - Integrating a searching functionality so that the user can find out his previous report of a specific date or test.
 - Integrating a Bar chart visualization to visually represent the health data history.
 - Add a health news or suggestion section.

4. Testing:

- Conduct validation with real medical reports to verify that extracted data matches the actual content of the reports.
- Correct the issues that detected on testing phase.

5. Security:

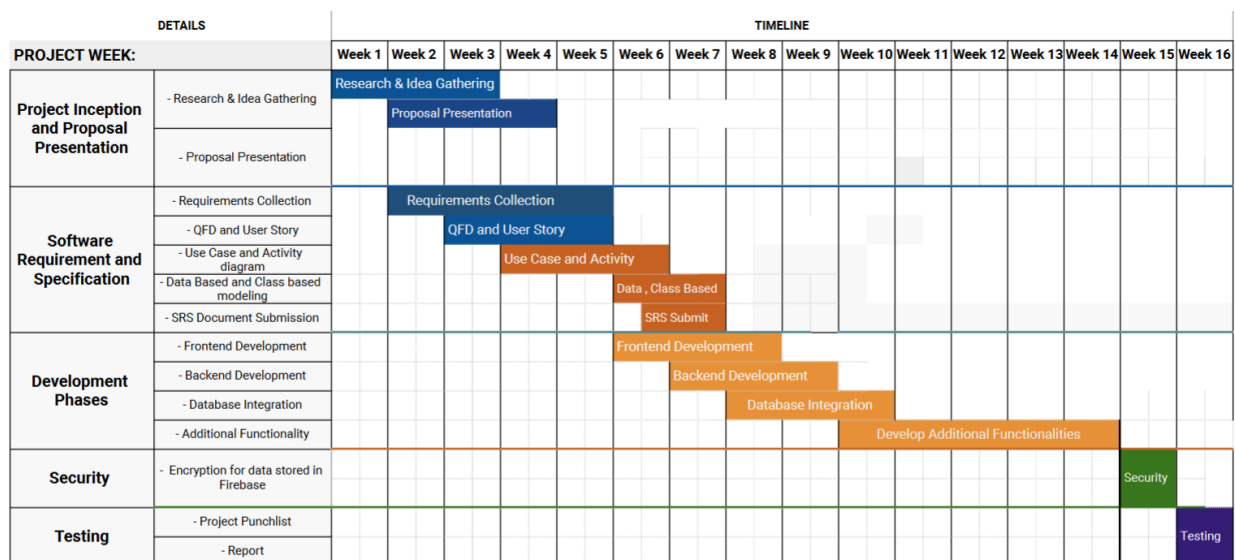
- Implement robust security measures such as encryption for data stored in Firebase and during data transmission.

Timeline:

Task	Timeline (Weeks)
Project Inception and Proposal Presentation	Week 1-4 (Oct 1 - Oct 21)
Software Requirements and Specification	Week 2-7 (Oct 10 - Nov 17)
Frontend Development	Week 6-8 (Nov 7 - Nov 28)
Backend Development	Week 7-9 (Nov 14 - Dec 5)
Database Integration	Week 8-10 (Nov 21 - Dec 10)
Develop Additional Functionalities	Week 10-14 (Dec 5 - Jan 14)
Security	Week 15 (Jan 14 - Jan 21)
Testing	Week 16 (Jan 22 - Jan 30)

Gantt chart:

Personal Health Tracker Project Timeline



Languages or Tools to be used: Dart, Python, Flask, Flutter, Firebase, Git, GitHub

Supervisor's Name: Dr. Ahmedul Kabir

Signature of the supervisor:_____

Date: _____

Before the Midterm Presentation 1:

I confirm that the progress is satisfactory and I am forwarding it for midterm presentation.

Signature of the supervisor:_____

Date: _____

Proposal Presentation Feedback (if any) :

Midterm 1 Presentation Feedback (if any) :

Midterm 2 Presentation Feedback (if any) :