
SOFTWARE DESIGN DOCUMENT

for

HEALTHKARD

Prepared by :

Umang Thadani (1914061)

Anurag Singh (1914058)

Dhruv Solanki (1914059)

Aayush Kapoor (1914066)

Guide: Prof. Era Johri

Contents

1	Introduction	4
1.1	Design Overview	4
1.2	Requirements Traceability Matrix	4
2	System Architectural Design	5
2.1	Client-Server Architecture	5
2.2	Model-View-Controller	5
2.3	System Interface Description	5
2.3.1	Ethtreum Blockchain	5
2.3.2	File System	5
2.3.3	Hardware Interfaces	5
2.3.4	Software Interfaces	6
2.3.5	Communication Protocols	6
3	Detailed Description of Components	7
3.1	Authentication	7
3.2	Health Card creation	7
3.3	Epidemic Analysis	7
4	User Interface Design	8
4.1	Landing Page	8
4.1.1	Screen Images	8
4.1.2	Objects and Actions	9
4.2	Sign Up	9
4.2.1	Objects and Actions	9
4.2.2	Screen Images	9
4.3	Sign In	10
4.3.1	Screen Images	10
4.3.2	Objects and Actions	10
4.4	Options	11
4.4.1	Screen Images	11
4.5	User Profile	11
4.5.1	Objects and Actions	11
4.5.2	Screen Images	11
4.6	Health expert Profile	11
4.6.1	Objects and Actions	11
4.6.2	Screen Images	12

5	System Architecture	13
6	Data Flow Specifications	16
6.1	Level 0 DFD with description	16
6.2	Level 1 DFD with description	16

1 Introduction

1.1 Design Overview

HealthKard aims to develop the foundations necessary for supporting digital health infrastructure for maintaining health data in a decentralized and secure way. A few major advantages to this project will be ease of access, user consent for every sophisticated transaction, and portability across national borders.

HealthKard aims to implement the following modules:

- Creation of a unique Health ID using Aadhaar Number.
- Storage of Electronic Health Records (EHRs) mapped to Health Identity in the blockchain.
- Integration of different sectors in the medical industry.
- Encourage better administration of the health sector by utilizing health data analytics.

1.2 Requirements Traceability Matrix



2 System Architectural Design

2.1 Client-Server Architecture

We choose client-server architecture for our system. There are two types of clients: Normal Users and Health Experts, each having a different interface. Both of these types of users interact with the same server with a common database and blockchain. Therefore, we separate concerns for application program and data management in our system.

2.2 Model-View-Controller

This is our second choice of architectural design. We chose Client-Server over MVC due to the fact that there is minimal dynamic content and need for dependency mechanism since the entire data needs to be changed, if any changes are applicable, only when the page reloads.

2.3 System Interface Description

2.3.1 Ehtreum Blockchain

We use the Ethereum Blockchain to save users' health records linked to the respective users' MetaMask account. The users' MetaMask Account is further linked to their Aadhar Card number.

2.3.2 File System

We use a modular file system where the Front End and Back End of the application are separated into different modules to improve maintainability.

2.3.3 Hardware Interfaces

1. Processor: x86 or x64
2. RAM: 512 MB (minimum), 1 GB (recommended)
3. Hard disk: up to 512 MB of available space may be required.

2.3.4 Software Interfaces

1. Operating System: Any OS that can support a gunicorn server.
2. Front End Stack: ReactJS, Recoil, MUI, TailwindCSS.
3. Back End Stack: Django REST Framework, PostgreSQL.
4. Browser Requirements: Any modern web browser having MetaMask extension.

2.3.5 Communication Protocols

1. The client side and the server endpoint will communicate using standard HTTP (Hyper Text Transfer Protocol) which is a generic stateless protocol.
2. The email system will use the SMTP protocol.

3 Detailed Description of Components

3.1 Authentication

1. Allow new users to login/register.
2. Verify their identity using the Aadhar APIs.
3. Change Password and Logout
4. Create profile for Health Expert (Only by admin).

3.2 Health Card creation

1. Link users' MetaMask to their Aadhar Card.
2. Save users' Health Records over the Blockchain.
3. Allow health experts to access users' data only after their consent.

3.3 Epidemic Analysis

1. Collecting anonymous health symptoms.
2. Analyzing anonymous health data and predicting the spread of a disease in a region.

4 User Interface Design

4.1 Landing Page

A landing page is any web page that a customer can land on, but in marketing, it's usually a standalone page that serves a single and focused purpose, separate from your homepage or any other page.

4.1.1 Screen Images



screenshots/Landing Page.jpeg

4.1.2 Objects and Actions

Header

It comprises of the navigation bar with hyperlink to the following pages:

1. Login
2. About
3. Logo
4. Dashbaord/Login
5. Profile/Login

Hero Image

It is just a simple image used to beautify the Design

Features

It showcases the features of our web application

About

It is a short description of our web application

Footer

It contains copyright information and social media links


4.2 Sign Up

A user will be required to register into our application before they are able to use any of our features. If a user already has an account, they can instead go for Sign In.

4.2.1 Objects and Actions

The page consists of a Sign Up form which asks for details such as Name, Email, Username and Password.

4.2.2 Screen Images

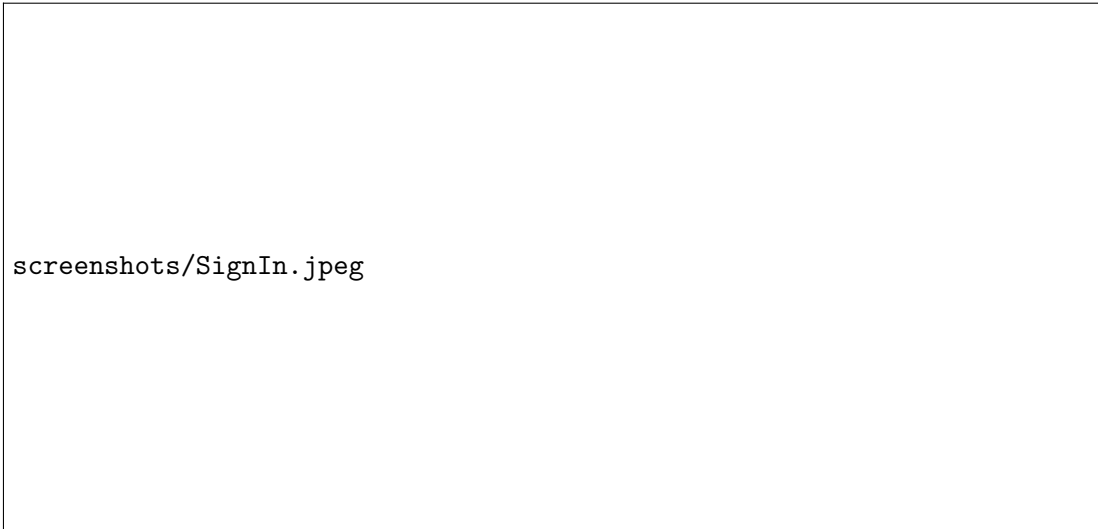


screenshots/SignUp.jpeg

4.3 Sign In

When a user has created an account, they go to the Sign In page to log in to our application and enjoy its features.

4.3.1 Screen Images



screenshots/SignIn.jpeg

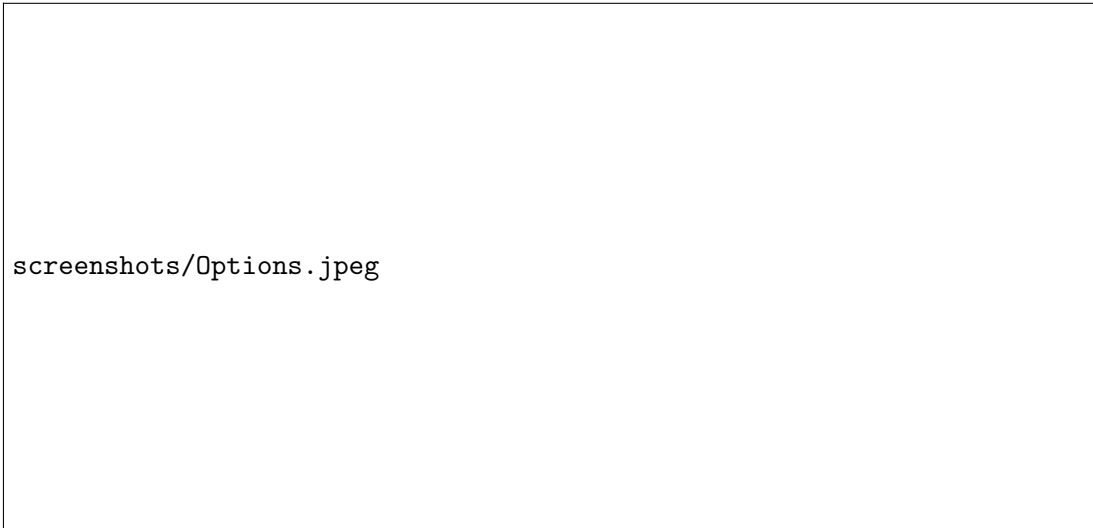
4.3.2 Objects and Actions

The page consists of a Sign In form which asks for details such as Aadhar Number and Password.

4.4 Options

When a user logs in for the very first time, we ask the user if they want to login as a normal user or a health expert.

4.4.1 Screen Images



4.5 User Profile

4.5.1 Objects and Actions

The page consists of a form which asks for details such as Phone Number, Company, Company URL (for verification). This data will then be verified by the Aadhar data and then the user can upload his health records to the system which are saved to the blockchain.

4.5.2 Screen Images

4.6 Health expert Profile

4.6.1 Objects and Actions

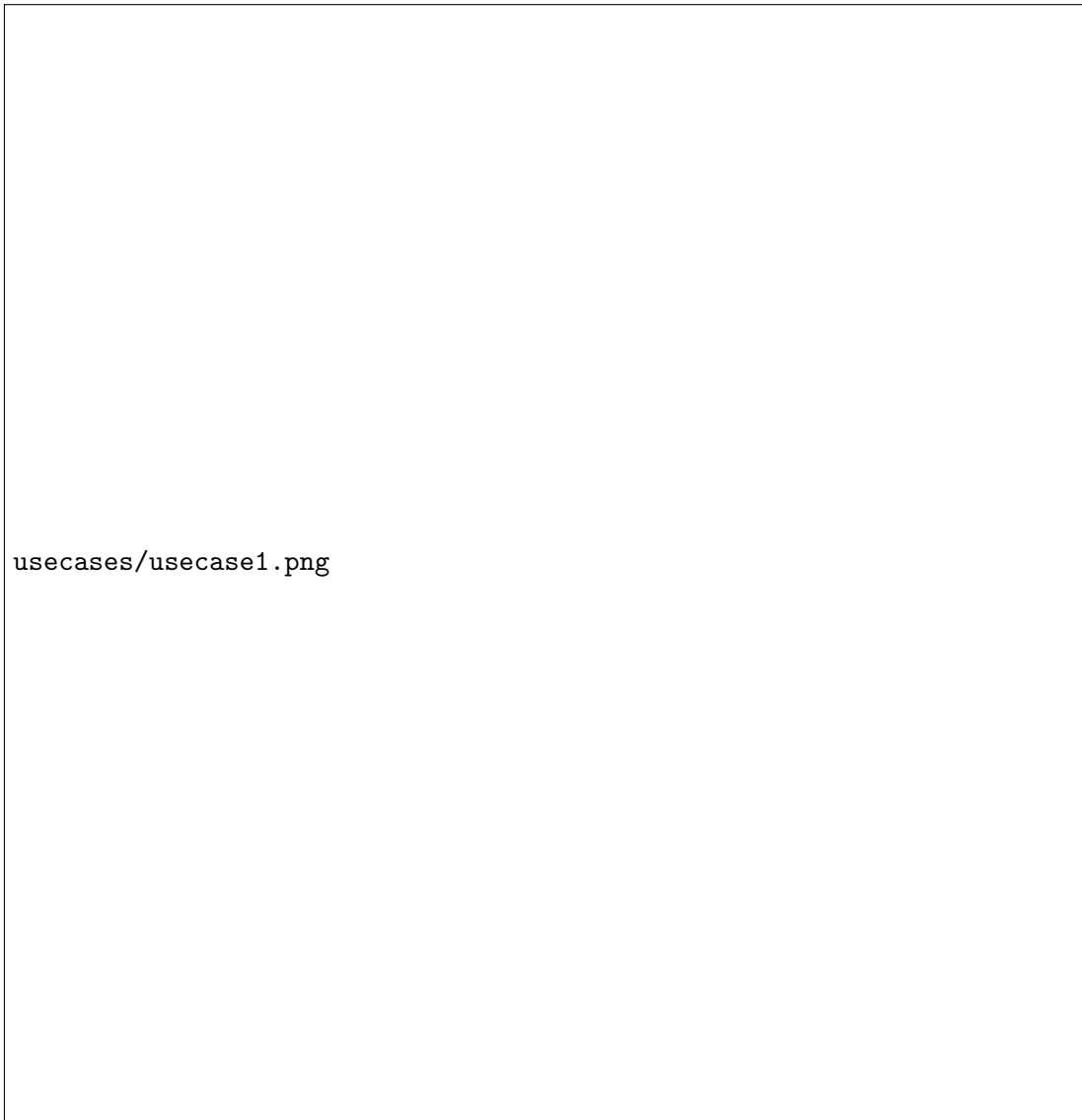
The page consists of a form where they can ask for user's consent to view their health records. If a user allows it, then only they can view their health data.

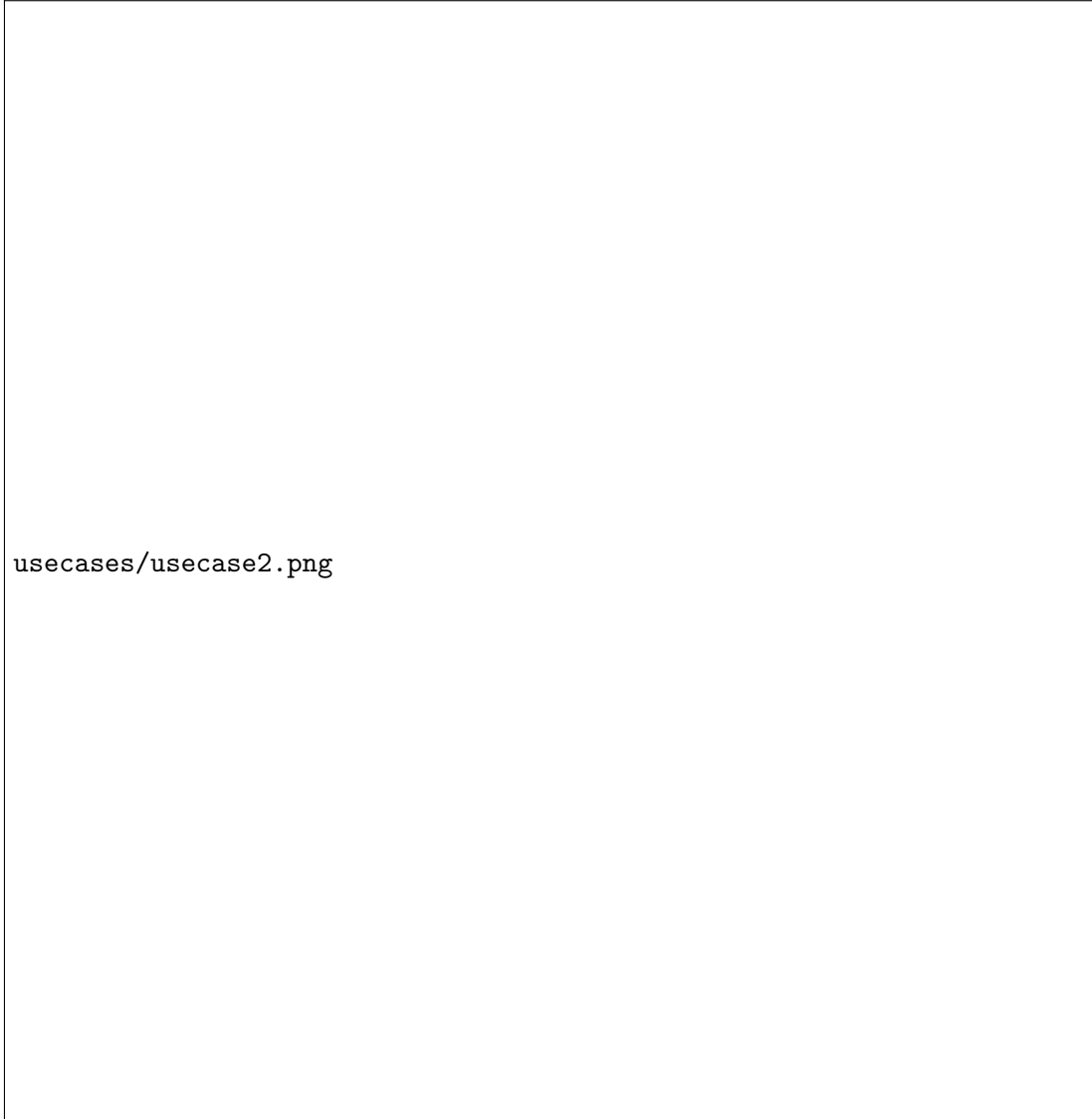
screenshots/Recruiter.jpeg

4.6.2 Screen Images

screenshots/Student.jpeg

5 System Architecture





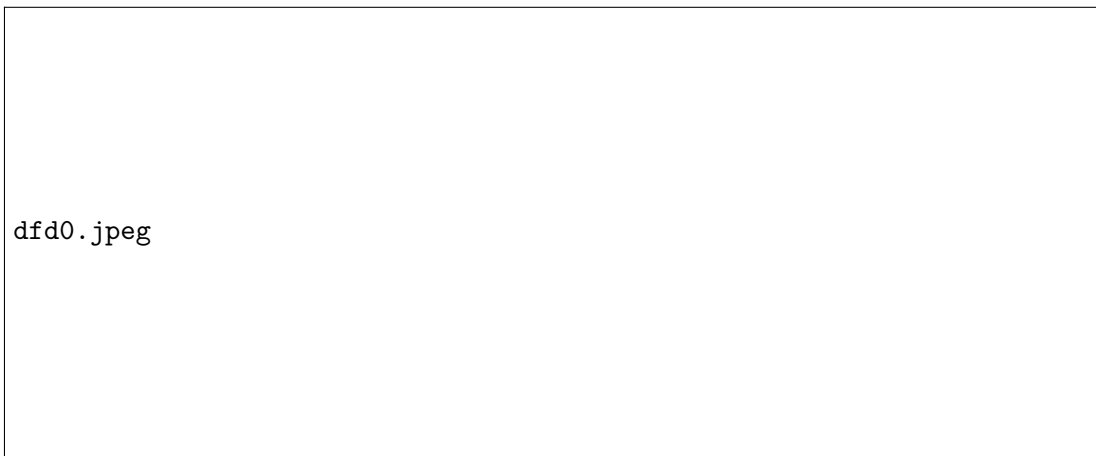
usecases/usecase2.png

usecases/usecase3.png

6 Data Flow Specifications

DFD is created from the SRS document provided.

6.1 Level 0 DFD with description



6.2 Level 1 DFD with description

dfd1.jpeg