A Field Project Report on

Health Dashboard

Submitted

In partial fulfillment of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE and ENGINEERING

by

B Kavya Reddy	(231FA04E86)
K Chetan Reddy	(231FA04F01)
C Saketh	(231FA04F59)
N Yaswanth	(231FA04F61)

Under the Guidance of

Ch.Swarna Lalitha

Assistant Professor, CSE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SCHOOL OF COMPUTING AND INFORMATICS

VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University)
Vadlamudi, Guntur -522213, INDIA.

April, 2025

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

CERTIFICATE

This is to certify that the field project entitled *"Health Dashboard"* is being submitted by **[B Kavya Reddy], [231FA04E86], [K Chetan Reddy], [231FA04F01], [C Saketh], [231FA04F59]**,and **[N Yaswanth], [231FA04F61],** in partial fulfilment of the requirements for the degree of **Bachelor of Technology (B.Tech.) in Computer Science and Engineering** at the Department of Computer Science and Engineering, Vignan's Foundation for Science, Technology and Research (Deemed to be University), Vadlamudi, Guntur District, Andhra Pradesh, India.

This is a bonafide work carried out by the aforementioned students under my guidance and supervision.

Guide

Project Review Committee

HoD, CSE

HoD
Dept. of Computer Science & Engineering
VFSTR Deemed to be University
VADLAMUDI - 522 213
Guntur Dist., A.P., India.

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

DECLARATION

Date:

We hereby declare that the work presented in the field project titled "**Health DashBoard**" is the result of our own efforts and investigations.

This project is being submitted under the supervision of **Ch.Swarna Lalitha , Assistant Professor , CSE** in partial fulfillment of the requirements for the Bachelor of Technology (B.Tech.) degree in Computer Science and Engineering at the Department of Computer Science and Engineering, Vignan's Foundation for Science, Technology and Research (Deemed to be University), Vadlamudi, Guntur, Andhra Pradesh, India.

B Kavya Reddy	231FA04E86	B Kavya Reddy
K Chetan Reddy	231FA04F01	K Chetan
C Saketh	231FA04F59	C Saketh
N Yaswanth	231FA04F61	N Yaswanth

Chapter No.	Description	Page No.
1	**Introduction** 1.1 Problem Definition 1.2 Existing System 1.3 Proposed System 1.4 Literature Review	1
2	**System Requirement** 2.1 Hardware & Software Requirement 2.2 Software Requirement Specification (SRS)	3
3	**System Design** 3.1 Modules of System 3.2 UML Diagram	4
4	**Implementation** 4.1 Sample Code 4.2 Test Cases	7
5	**Results** 5.1 Output Screens	11
6	Conclusion References	13

1.INTRODUCTION

A **Health Dashboard** is a visual and interactive tool designed to display real-time health-related data in a clear and organized manner. It consolidates key health metrics—such as vital signs, activity levels, or system performance indicators—into a single interface, enabling users, healthcare professionals, or administrators to monitor and assess health status quickly and effectively. By providing instant insights and trends, health dashboards support informed decision-making, early detection of issues, and improved overall management of individual or population health.

1.1 Problem Definition

In today's fast-paced world, managing and monitoring health data effectively has become increasingly challenging due to the large volume and fragmentation of information. Individuals, healthcare providers, and organizations often struggle to access real-time, accurate, and consolidated health metrics needed for timely decision-making. Without a centralized system, critical health indicators can be overlooked, leading to delayed responses, reduced efficiency, and compromised health outcomes. Therefore, there is a need for a comprehensive Health Dashboard that integrates diverse health data sources into a unified, user-friendly platform for continuous monitoring, analysis, and reporting.

1.2 Existing System

Currently, various health monitoring systems and applications are available, such as fitness trackers (e.g., Fitbit, Apple Health), electronic health records (EHR) systems, and hospital management software. These systems focus on specific aspects like personal fitness, clinical data management, or hospital operations. However, they often work in isolation without integration, making it difficult to obtain a comprehensive view of a person's or system's overall health status. Many existing solutions lack real-time data synchronization, user-friendly interfaces, and customizable dashboards tailored to individual or organizational needs.

1.2.1 Limitations of Existing Systems:

1. **Lack of integration** between personal, clinical, and system-level data.
2. **No centralized platform** to view all relevant health metrics in one place.
3. **Limited real-time monitoring**, leading to delayed decision-making.
4. **Complex interfaces** that are not user-friendly, especially for non-technical users.
5. **Poor data visualization**, making it hard to interpret key trends and insights.
6. **Inflexibility** in customizing metrics and dashboards to fit specific user requirements.

1.3 Proposed System

- A centralized and integrated Health Dashboard for real-time health data monitoring.
- Combines personal health metrics, clinical records, and system performance in one platform.
- Provides interactive visualizations such as graphs, charts, and status indicators.
- Offers role-based access for patients, doctors, and administrators.
- Customizable interface based on user needs and preferences.

Advantages:

- **Improved decision-making** through real-time and consolidated data access.
- **Enhanced user experience** with intuitive and interactive interface.
- **Time-saving** by reducing the need to access multiple systems for information.
- **Early detection** of health issues with real-time alerts.
- **Customizability** allows users to focus on relevant metrics.

1.4 Literature Review

Existing studies highlight the growing importance of health dashboards in improving healthcare monitoring and decision-making. Research shows that tools like wearable devices and EHR systems effectively collect health data but often lack integration and real-time visualization. Literature emphasizes the need for unified platforms that combine various data sources into a single, user-friendly interface. Studies also suggest that interactive dashboards can enhance early detection of health issues, reduce emergency response time, and support better patient care. These findings underline the need for an advanced, integrated health dashboard system.

2. SYSTEM REQUIREMENTS

2.1 Hardware and Software Requirements

Hardware Requirements:

- **Processor:** Intel Core i3 or higher
- **RAM:** 4 GB or more
- **Storage:** Minimum 50 MB of free space
- **Display:** Minimum 1024×768 resolution
- **Operating System:** Windows, macOS, or Linux

Software Requirements:

- **Web Browser:** Chrome, Firefox, Edge, or Safari
- **Development Tools:** VS Code, Sublime Text, or Notepad++
- **Front-end Technologies:** HTML, CSS
- **Optional:** JavaScript for enhanced functionality

2.2 Software Requirement Specification (SRS)

Functional Requirements:

- **User Authentication:** Users can register, log in, and manage their profiles securely.
- **Dashboard Display:** The system displays real-time health metrics using charts and graphs.
- **Data Input:** Users or connected devices can input/update health data.
- **Alerts & Notifications:** The system sends alerts for abnormal health readings.
- **Role-Based Access:** Different access levels for patients, doctors, and admins.
- **Report Generation:** Users can generate and download health reports.
- **Data Integration:** Ability to fetch and display data from external health devices or APIs.

Non-Functional Requirements:

- **Performance:** The system should load dashboard data within 2 seconds.
- **Scalability:** The application should handle increasing numbers of users and data.
- **Usability:** Interface must be simple, responsive, and user-friendly.
- **Security:** Data must be encrypted and protected using secure protocols (e.g., HTTPS).
- **Reliability:** The system should have 99.9% uptime with automatic backups.
- **Compatibility:** Should work across all major browsers and devices.

3. SYSTEM DESIGN

3.1 Modules of System:

3.1.1. User Authentication Module

- Handles user registration, login, logout, and password management.
- Supports role-based access for patients, doctors, and administrators.

3.1.2. Dashboard Module

- Displays real-time health data using charts, graphs, and summary cards.
- Provides an overview of key health metrics like heart rate, blood pressure, etc.

3.1.3 Health Data Management Module

- Allows manual input or automatic syncing of health data from devices/APIs.
- Stores and organizes health records for each user.

3.1.4. Alert & Notification Module

- Generates alerts for abnormal health readings or missing data updates.
- Sends notifications via email or within the app.

3.1.5. Report Generation Module

- Generates downloadable reports (PDF/CSV) based on selected health parameters and time periods.
- Useful for medical reviews or personal tracking.

3.1.6 Admin Management Module

- Allows admin to manage user roles, monitor system activity, and perform maintenance.
- Includes tools for system settings and data backup.

3.1.7. Integration Module

- Connects with wearable devices or third-party health apps via APIs.
- Ensures smooth data flow between external sources and the dashboard.

Each module works in coordination to ensure efficient health monitoring, secure access, and user-friendly interaction.

3.2 ER Diagram:

Entities and Attributes:

1. User

- *User_ID* (Primary Key)
- Name
- Email
- Password
- Role (Patient / Doctor / Admin)

2. Health_Data

- *Data_ID* (Primary Key)
- User_ID (Foreign Key)
- Heart_Rate
- Blood_Pressure
- Temperature
- Timestamp

3. Notification

- *Notification_ID* (Primary Key)
- User_ID (Foreign Key)
- Message
- Status (Read/Unread)
- Date_Time

4. Report

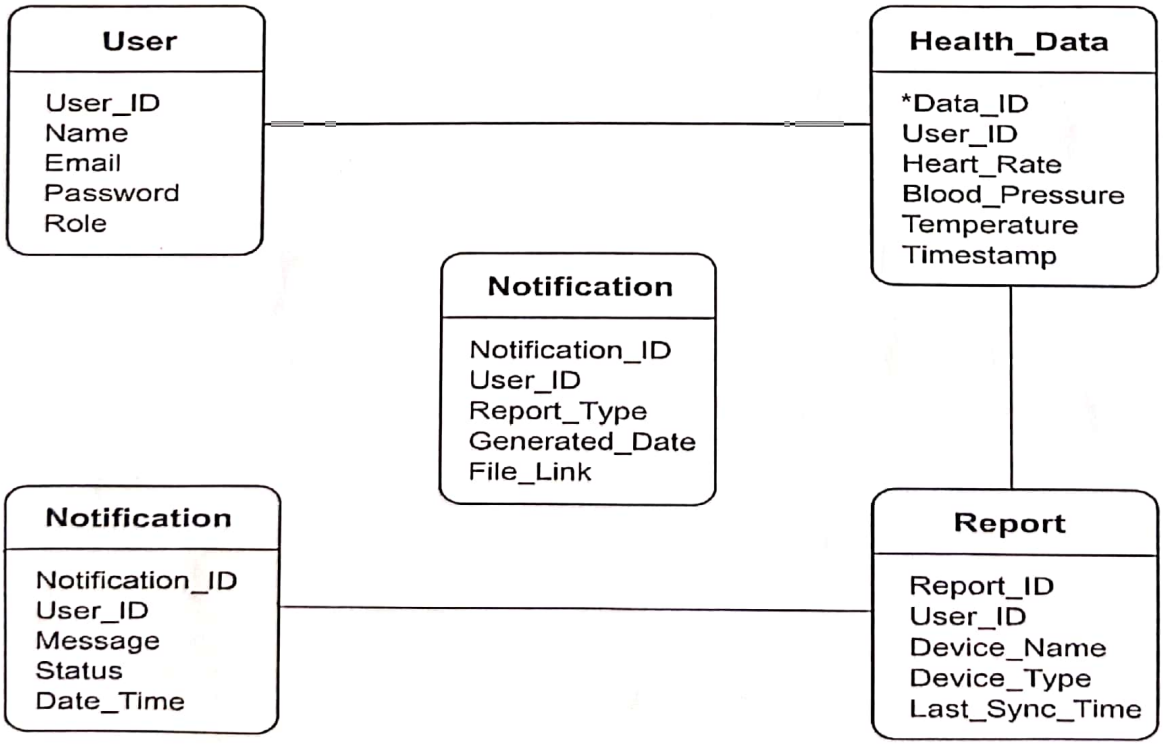
- *Report_ID* (Primary Key)
- User_ID (Foreign Key)
- Report_Type
- Generated_Date
- File_Link

5. Device

- *Device_ID* (Primary Key)
- User_ID (Foreign Key)
- Device_Name
- Device_Type
- Last_Sync_Time

Relationships:

- One User can have many Health_Data entries. *(One-to-Many)*
- One User can receive multiple Notifications. *(One-to-Many)*
- One User can generate multiple Reports. *(One-to-Many)*
- One User can be connected to multiple Devices. *(One-to-Many)*



4.IMPLEMENTATION

4.1 Sample Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport"
content="width=device-width, initial-
scale=1.0">
<title>Health Dashboard</title>
<style>
body {
font-family: Arial, sans-serif;
margin: 0;
padding: 0;
background-color: #f4f7fa;
}

header {
background-color: #4CAF50;
color: white;
padding: 15px;
text-align: center;
}

.dashboard-container {
display: flex;
justify-content: space-around;
padding: 20px;
}
```

```
.card {
background-color: #fff;
border-radius: 8px;
box-shadow: 0 4px 8px rgba(0, 0, 0,
0.1);
padding: 20px;
width: 200px;
text-align: center;
margin: 10px;
}

.card h3 {
margin: 10px 0;
}

.card p {
font-size: 24px;
font-weight: bold;
}

.card .icon {
font-size: 50px;
color: #4CAF50;
}

footer {
background-color: #333;
color: white;
text-align: center;
padding: 10px;
```

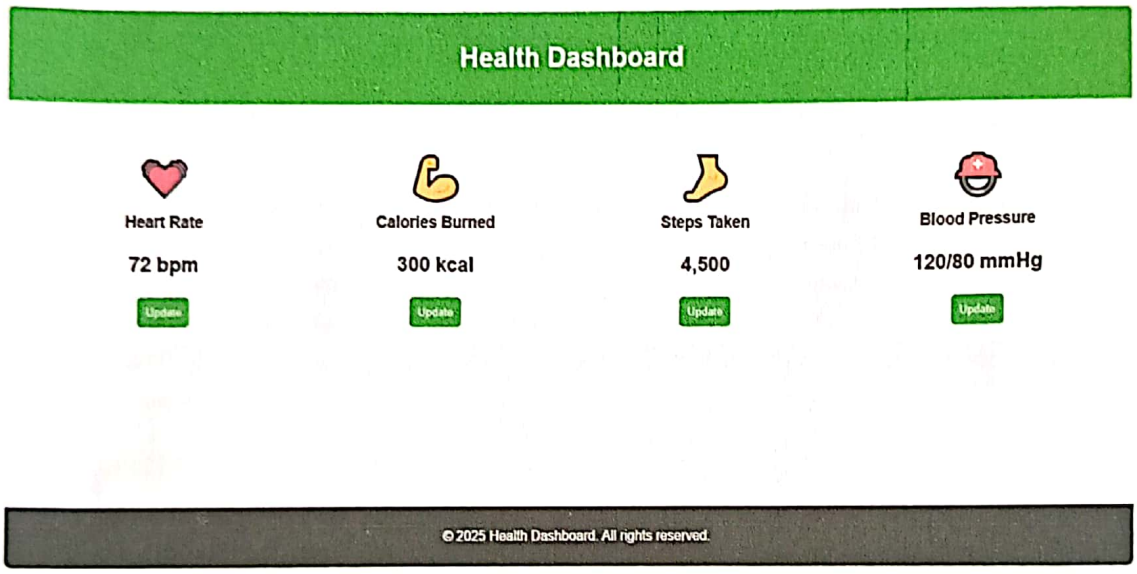
```
position: fixed;
width: 100%;
bottom: 0;
}

.btn {
background-color: #4CAF50;
color: white;
padding: 10px;
border: none;
border-radius: 5px;
cursor: pointer;
}

.btn:hover {
background-color: #45a049;
}
</style>
</head>
<body>

<header>
<h1>Health Dashboard</h1>
</header>

<div class="dashboard-container">
<div class="card">
<div class="icon">💓</div>
<h3>Heart Rate</h3>
<p id="heartRate">72 bpm</p>
```

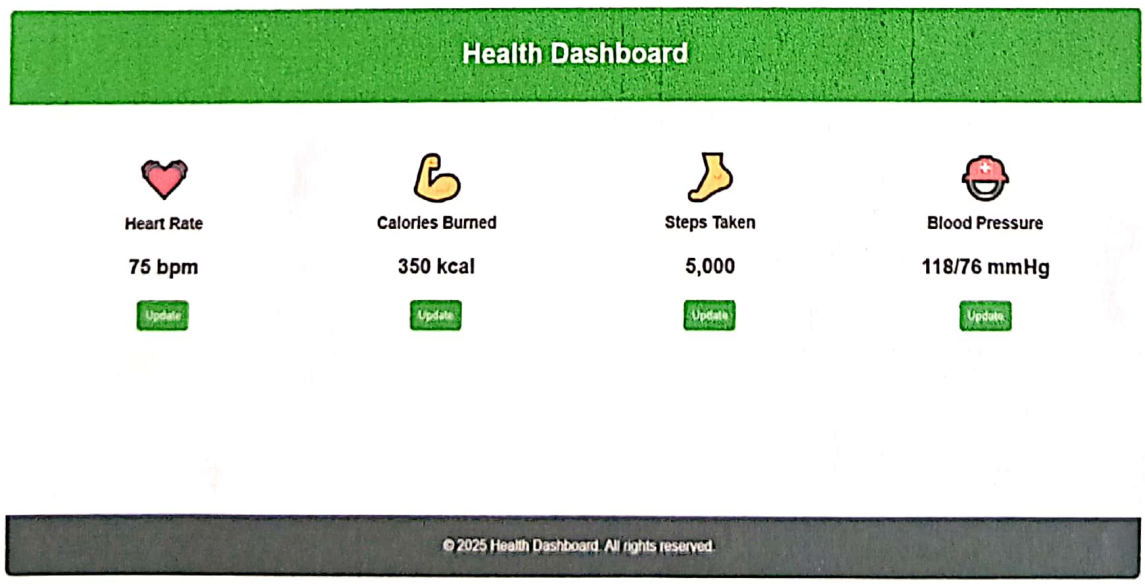
```
<button class="btn"
onclick="updateHeartRate()">Update</butto
n>
</div>
<div class="card">
<div class="icon">🔥</div>
<h3>Calories Burned</h3>
<p id="caloriesBurned">300 kcal</p>
<button class="btn"
onclick="updateCalories()">Update</button
>
</div>
<div class="card">
<div class="icon">□</div>
<h3>Steps Taken</h3>
<p id="stepsTaken">4,500</p>
<button class="btn"
onclick="updateSteps()">Update</button>
</div>
<div class="card">
<div class="icon">□</div>
<h3>Blood Pressure</h3>
<p id="bloodPressure">120/80
mmHg</p>
<button class="btn"
onclick="updateBloodPressure()">Update</b
utton>
</div>
</div>

<footer>
```

```
<p>&copy; 2025 Health Dashboard. All
rights reserved.</p>
</footer>

<script>
function updateHeartRate() {

document.getElementById('heartRate').inner
Text = '75 bpm';
}

function updateCalories() {

document.getElementById('caloriesBurned').i
nnerText = '350 kcal';
}

function updateSteps() {

document.getElementById('stepsTaken').inne
rText = '5,000';
}

function updateBloodPressure() {

document.getElementById('bloodPressure').i
nnerText = '118/76 mmHg';
}
</script>
</body>
</html>
```

5.RESULTS

5.1 Output screens





6.CONCLUSION

The development of a health dashboard provides an effective solution for monitoring, analyzing, and visualizing critical health metrics in real time. By integrating key health indicators into a unified interface, the dashboard enables users—whether patients, healthcare providers, or administrators—to make informed decisions based on accurate and up-to-date information.

Through intuitive visual elements such as charts, graphs, and alerts, the dashboard enhances data accessibility and promotes proactive health management. The implementation demonstrates the potential of data-driven tools in improving healthcare outcomes, supporting early detection of anomalies, and encouraging timely interventions. Future enhancements may include integration with wearable devices, predictive analytics, and personalized health recommendations to further improve functionality and user experience.

6.1 Reference

The code I provided is a custom design, but it uses foundational web development principles. For HTML structure, you can refer to MDN Web Docs: HTML. CSS styling principles can be further explored through MDN Web Docs: CSS and CSS-Tricks. For JavaScript and DOM manipulation, MDN Web Docs: JavaScript is a great resource. The icons used come from popular libraries like FontAwesome.