COLLEGE OF APPLIED SCIENCE ADOOR

(Government of Kerala, Managed by IHRD)

(Affiliated to University of Kerala)

Adoor, Kerala

College of Applied Science, Adoor

PROJECT REPORT

on

FITVEN

Submitted by

AKASH R NAIR (33221802011)

ARAVIND R KRISHNA (33221802016)

DEVANARAYANAN S B (33221802022)

SIDHARTH N (33221802048)

in partial fulfillment of the requirements for the award of
Bachelor of Computer Application degree of

University of Kerala

Department of Computer Science
COLLEGE OF APPLIED SCIENCE ADOOR

2021-2024

COLLEGE OF APPLIED SCIENCE ADOOR
(Government of Kerala, Managed by IHRD)

(Affiliated to University of Kerala)

Adoor, Kerala

College of Applied Science, Adoor

CERTIFICATE

Certified that this report titled *"FITVEN"* is a bonafide record
of the project work done by

AKASH R NAIR (33221802011)

ARAVIND R KRISHNA (33221802016)

DEVANARAYANAN S B (33221802022)   SIDHARTH
N (33221802048)

Under our supervision and guidance, towards partial fulfillment of the requirements
for the award of the Degree of BCA (Bachelor of Computer Applications) of the
University of Kerala

Internal Guide
Asst. Prof. Femeena Jinoob

Head of Department
Prof. Viji Balakrishnan

Head of Institution   **External Guide**
Dr. Santhosh Babu K

ACKNOWLEDGEMENT

Akash R Nair

Aravind R Krishna

Devanarayanan S B

Sidhardh N

ABSTRACT

▶ In today's health-conscious society, maintaining a balanced diet and fitness routine is crucial for overall well-being. Our web development project aims to revolutionize the way people approach their fitness goals by offering a comprehensive platform that provides personalized calorie tracking, workout planning, and one-on-one support from certified fitness coaches. The website will act as an all-in-one fitness companion, catering to users of all fitness levels and empowering them to lead healthier lives.

▶ The core feature of the website is the advanced calorie calculator, which will enable users to effortlessly track their daily caloric intake and expenditure. The system will consider individual factors such as age, gender, weight, height, and activity level to generate accurate caloric estimates. Additionally, the platform will offer customized workout plans based on users' fitness goals and preferences. These plans will be curated by fitness experts and encompass various exercise routines, from strength training to cardiovascular activities, ensuring a well-rounded approach to fitness.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

S. No.	Abbrevation	Full Form
1	Asst. Prof.	Assistant Professor
2	Prof.	Professor
3	HTML	HyperText Markup Langauge
4	CSS	Cascading Style Sheet
5	JS	JavaScript
6	PHP	PHP Hypertext Preprocessor
7	SQL	Structured Query Language
8	DFD	Data Flow Diagram

INTRODUCTION

1.1 PROJECT OVERVIEW

FITVEN is a comprehensive web platform designed to assist users in achieving their health and fitness goals. The platform focuses on three primary pillars: calorie calculation, interactive coaching, and personalized diet planning. FITVEN aims to empower users to lead a healthier lifestyle by providing them with the necessary tools and guidance.

1.2 OBJECTIVES

1. Calorie Calculation

   FITVEN will offer a user-friendly tool that allows users to calculate their daily caloric needs based on their specific goals (e.g., weight loss, maintenance, muscle gain). This tool will consider factors such as age, weight, height, activity level, and desired goal to provide an accurate calorie target. Users will also have access to a food database to help track their daily caloric intake.

2. Interactive Coaching

   FITVEN will feature a platform for users to interact with certified fitness coaches and nutritionists. Users can ask questions, seek advice, and receive personalized guidance to optimize their fitness journey. The interactive coaching will be facilitated through chat, video calls, and forums, fostering a sense of community and support.

3. Diet Planning

   FITVEN will provide users with a tool to create personalized diet plans based on their dietary preferences, restrictions, and goals. The platform will offer pre-designed meal plans and recipes, allowing users to customize and tailor their meal options. Nutritional information, including calorie count, macronutrient breakdown, and micronutrient details, will be provided for each meal.

4. Progress Tracking

   FITVEN will incorporate a tracking feature to allow users to monitor their progress over time. Users can track various metrics, including weight, body measurements, workout performance, and dietary adherence. Progress charts and graphs will visualize the user's journey, providing motivation and insights into their fitness and nutrition progress.

2.SYSTEM ANALYSIS

2.1 INTRODUCTION

A system is simply a set of components to accomplish an objective. Developing a new system, investigating into the operation and making possible changes in the existing system are called System Analysis. Analysis comprises a detailed study of the various operations performed by a system and their relationships within and outside the system. It is the process of gathering and interpreting facts, diagnosing problems and improving the system using the information obtained.

The objectives of System Analysis include the following ●

Identifying the user's needs.

- Performing economic and technical analysis.

- Establishing cost and schedule constraints.

Here the system analyst should study a system with an eye on solving the problem using computers. It is an essential part for the development of a project by a system analyst. System analysis is for finding out what happens in the existing systems, deciding on what changes and new features are required and defining exactly what the proposed system must be. This process of system analysis is largely concerned with determining, developing and agreeing to the user's requirements. It provides prime opportunities to communicate well with the user and conceive a joint understanding of what a system should be doing, together with a view of the relative importance of the system facilities using interactive techniques.

To analyze a system, one has to study the system's work in detail. The system analyst has to understand the functioning and concept of the system in detail, before designing the appropriate computer based system that will meet all the requirements.

2.2 EXISTING SYSTEM

The web application is able to allow end users or requestors to sign up for an account. This requires a lot of hard work and time consumption to complete the task. This may include human errors. In the existing system, it is difficult to retrieve some particular information. Also, all the records are stored manually and it is a tedious task. As a result the security of these records is always a challenging task. In spite of all the efforts undertaken, the destruction of data may happen often.

Hence, the computerization of the system of record maintenance is the only solution to reduce the shortcomings of the existing system

Advantages

- Pre Defined workout plan.
- PreDefined recipe.
- One on One coach support.
- User friendly interface.
- Region based client accessing.

2.3 PROPOSED SYSTEM

This application , to book an dietician t need to visit users for their diet plan and food chart. As a result, most of the process consumes lots of time and man power. The software 'BEFIT' will allow speeding up the works of users. The main objective of this application is to make it interactive and easy to use. This application is helpful in finding diet plans and calorie counting. The main motive of this project is to find the finding diet plans and calorie counting , dietician , neutrian very easily and time saving. **Advantages**

- Reduces time consumption.
- Avoids unauthorized access on data.
- Redundancy can be avoided.
- Generating food chart
- Generating diet plan chart
- Generate chat for neutrian .

2.4 FEASIBILITY STUDY

A preliminary investigation examines project feasibility. Feasibility study is a small scale system analysis. It is necessary as it evaluates the feasibility of a project at the earliest possible time.

Types of feasibility study:

- Technical feasibility
- Operational feasibility
- Functional feasibility
- Economical feasibility

- Social feasibility

Technical feasibility

It is the study of resource availability that may affect the ability to achieve an acceptable system. The system must be evaluated from the technical viewpoint first. The assessment of this feasibility must be based on an outline design of the system requirements in terms of input, output, program procedure, etc. Having identified the outline of the system, the investigation must go on to suggest the type of equipment, required method of developing the system, and the method of running the system. The outcome of the study was found to be positive.

Operational feasibility

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. The outcome of the study was satisfactory.

Functional feasibility

Here we examine the functions of the system which may work properly when implemented. The proposed system has functions that can be implemented successfully.

Hence the project was found to be functionally feasible.

Economical feasibility

It is considered as the final stage of most systems, it includes a broad range of concerns that include cost benefit analysis. The proposed system was found to be economically feasible as its requirements did not require huge expenditure, the group also had the knowledge to undertake this task without any difficulty.

Social feasibility

Social feasibility is a detailed study on how one interacts with others within a system or an organization. Social impact analysis is an exercise aimed at identifying and analyzing such impacts in order to understand the scale and reach of the project's social impacts. This

project has a great impact on the adoption of Linux on PCs by making software more accessible and easy to access and maintain.

3. SYSTEM ENVIRONMENT

3.1 INTRODUCTION

A system environment refers to the collection of hardware, software, and data that make up a computer system. This includes the physical components of a computer such as the CPU, memory, storage devices, and input/output devices, as well as the software applications and operating system that run on it. The system environment also includes the

network connections and protocols used to communicate with other devices and systems. Understanding the system environment is crucial for troubleshooting, optimizing performance, and developing software applications that work effectively in a particular system environment.

3.2 SYSTEM REQUIREMENTS

3.2.1 Website

Server requirements:

Hardware:

Processor: Any modern dual core processor or greater

RAM: 512MB or greater

Storage: 512MB of free space or greater Software:

Operating system: Windows

Database: MySQL

Web server: Apache

Programming languages: HTML,CSS,JAVASCRIPT

Client requirements:

Any device with an active internet connection and a browser with HTML5 support

3.3 TECHNOLOGIES USED

3.3.1 Programming languages

1. HTML

HTML, or Hypertext Markup Language, is a markup language used for creating web pages and applications. HTML provides the structure and content of a web page, defining headings, paragraphs, links, images, and other elements that make up the page. It is often assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript. HTML is a cornerstone technology for web development and is essential for

creating any kind of web page or application. HTML is easy to learn and understand, making it accessible to developers of all skill levels.

2. CSS

CSS, or Cascading Style Sheets, is a style sheet language used to describe the presentation and styling of HTML documents. CSS provides the means to control the layout, typography, colors, and other visual aspects of a web page, allowing developers to create visually appealing and engaging websites. CSS is an essential component of web development, working in conjunction with HTML to create a seamless and visually appealing user experience.

3. JavaScript

JavaScript is a popular programming language that is used to create interactive web applications and dynamic user interfaces. Developed by Netscape in 1995, JavaScript has become one of the most widely used programming languages in the world. JavaScript is a client-side scripting language, which means that it runs in the user's web browser and can be used to modify the content and behavior of a web page. JavaScript is an essential tool for web developers, enabling them to create engaging and interactive web applications that respond to user input and events. JavaScript is also used extensively in web development frameworks and libraries, such as React and Angular, which provide a range of pre-built components and tools for building complex web applications. The popularity of JavaScript has led to a large community of developers who contribute to its development, making it a powerful and versatile language that is constantly evolving to meet the needs of modern web development.

The project has made use of the following external JavaScript libraries: a. Ajax

Ajax (Asynchronous JavaScript and XML) is a web development technique that allows data to be retrieved from a server without reloading the entire page. Ajax enables websites to be more responsive and interactive, as it allows for data to be loaded in the background while the user continues to interact with the page. This technique involves

using a combination of JavaScript and XML (or JSON) to send and receive data between the client and server, without disrupting the user experience.

b. jQuery

jQuery is a fast, small, and feature-rich JavaScript library that simplifies HTML document traversal and manipulation, event handling, and AJAX. It is designed to make client-side scripting of HTML easier, and it is widely used for creating interactive web applications. jQuery is an open-source library that supports a variety of browsers, and its easy-to-use syntax makes it a popular choice for web developers.

3.3.2 Services/Tools

1. Apache Web Server

Apache Web Server is a popular open-source web server software that is used to host and serve web pages and other content over the internet. It is highly configurable, scalable and secure, and can run on various operating systems, including Linux, Windows, and macOS. Apache Web Server is widely used and popular because of its simplicity, flexibility, and extensive documentation. Apache Web Server is also popular for its support of multiple programming languages such as PHP, Perl, and Python, making it a great choice for developers who want to build dynamic web applications.

MySQL

Relational database systems are the most important database systems used in the software industry today.

One of the most outstanding systems is MySQL.

The important aspects of SQL Server are:

- MySQL is easy to use.

- Embedded database library.

- Commit grouping, gathering multiple transactions from multiple connections together to increase the number of commits per second.

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

On all platforms except Windows, MySQL ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or install MySQL workbench via a separate download. Many third-party GUI tools are also available.

4.SYSTEM DESIGN

4.1 INTRODUCTION

System design is a crucial phase in the software development life cycle where a highlevel conceptual design is created for the proposed system. It involves defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. The objective of system design is to translate the requirements gathered during the analysis phase into a detailed design that can be implemented by the developers. A

welldesigned system is essential for ensuring that the final product meets the user's expectations, is maintainable, scalable, and can be easily modified or updated in the future. A thorough system design is also important for estimating the development effort, cost, and resources required to build the system.

4.2 PROCESS DESIGN Modules

1. Website

    a. User

        i.   Dietician information page

            1.   View all dieticians

            2.   Send application for dietician

            3.   View application status

            4.   View diet chart

            5.   View food chart ii. Available nutrition page

                1.   View all nutrition

                2.   Send message for nutrition

                3.   View reply from nutrition iii.

                            Payment

    b. Admin

        ii.   Manage dietician

        iii. Manage nutrition

        iv.   View users

        v.   View diet report for users

        vi.   View payment

    c. Dietician

        i.   View applications

        ii.   ii.   Manage diet plan

        iii. iii. Send report to admin

iv.   iv.   View food calories v.   Add food chart

vi.   Add exercise

d. Nutrition

i. View message   ii. Send

reply for message.

4.3 DATA FLOW DIAGRAM (DFD)

A data-flow diagram is a way of representing a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops.

For each data flow, at least one of the endpoints (source and / or destination) must exist in a process. The refined representation of a process can be done in another data-flow diagram, which subdivides this process into sub-processes. The data-flow diagram is a tool that is part of structured analysis and data modeling. The basic elements of Data Flow Diagram are :
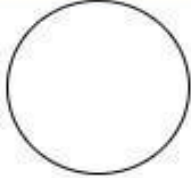
SYMBOL	DESCRIPTION
○	**Process** A process denotes some amount of work being done on data
▭	**External Entity** This represents any outside agency, interacting with the system. It represents the source or destination of data
→	**Data Flow** It represents flow of data between process or external entity and data store
▭	**Data Store** A data store is place for holding information within the system

Fig 4.4.1 Elements of Data Flow Diagram (DFD)
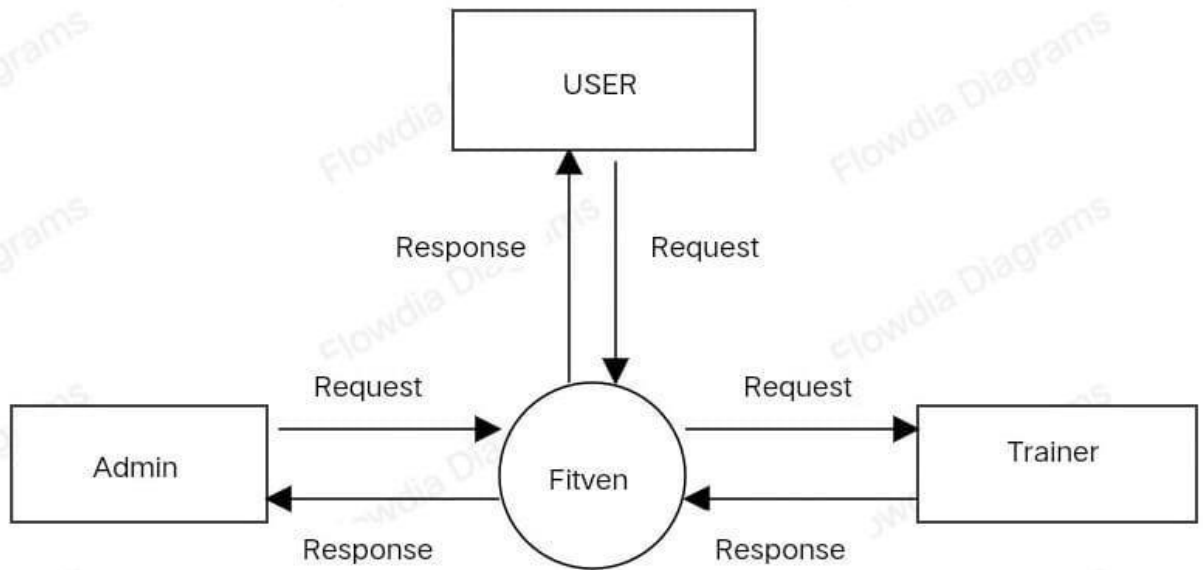
FITVEN

Context level

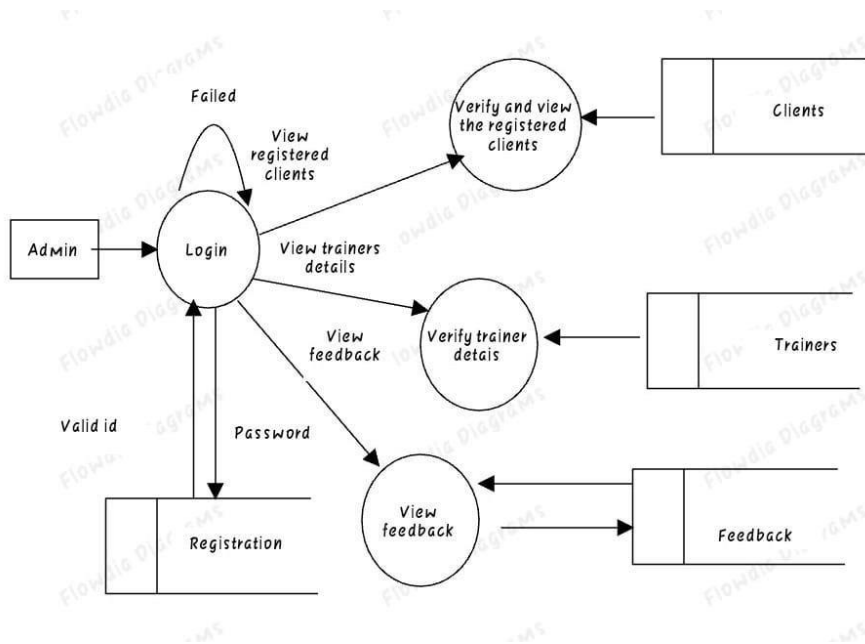Fig 4.4.2 Context level Data Flow Diagram of FITVEN

Level 1

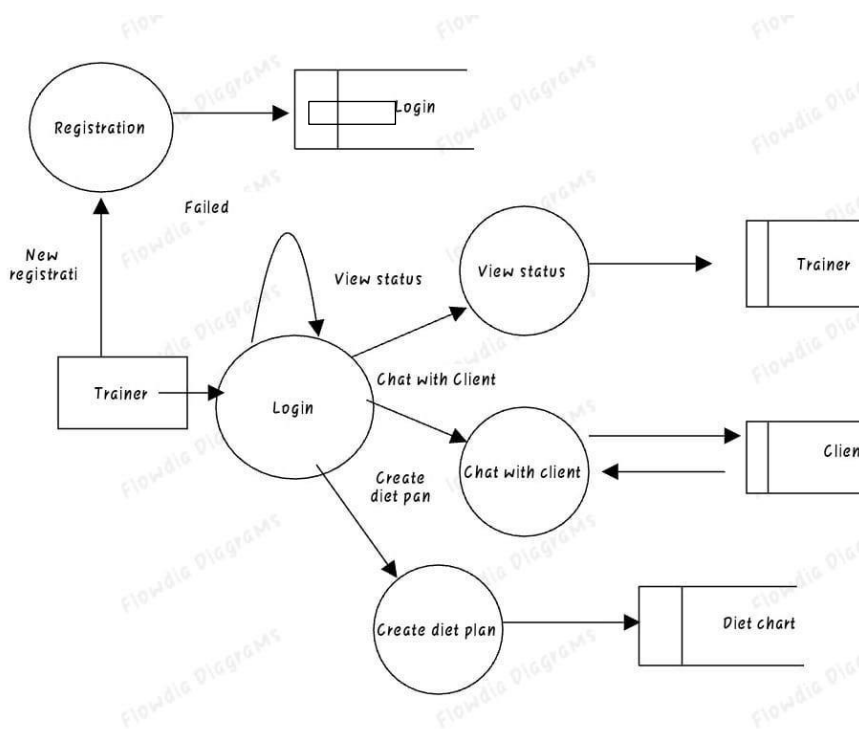Fig 4.4.3 Level 1 Data Flow Diagram of FITVEN

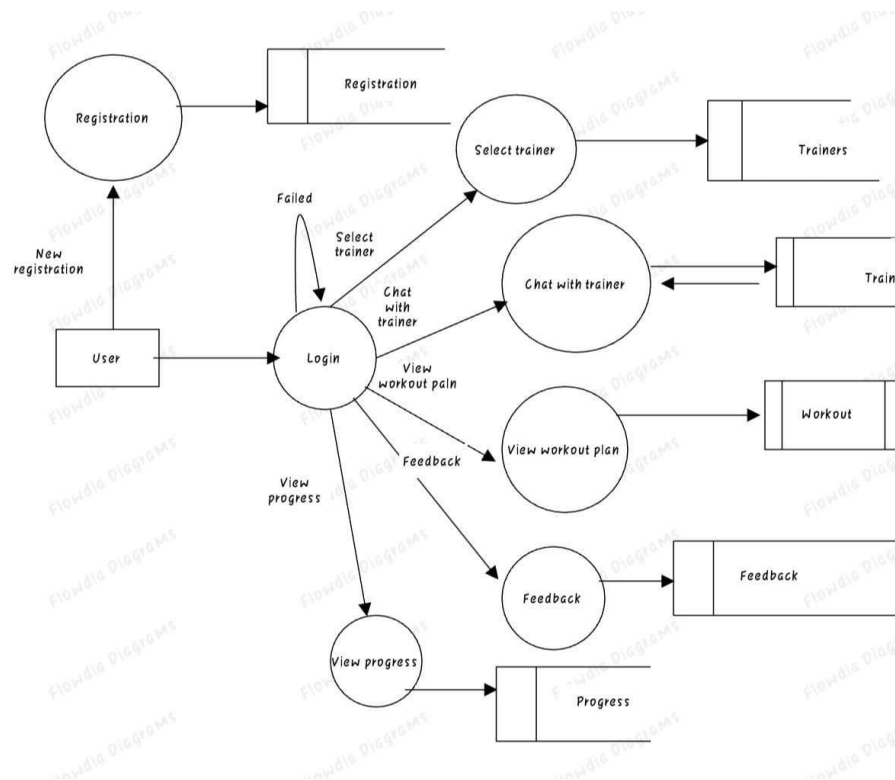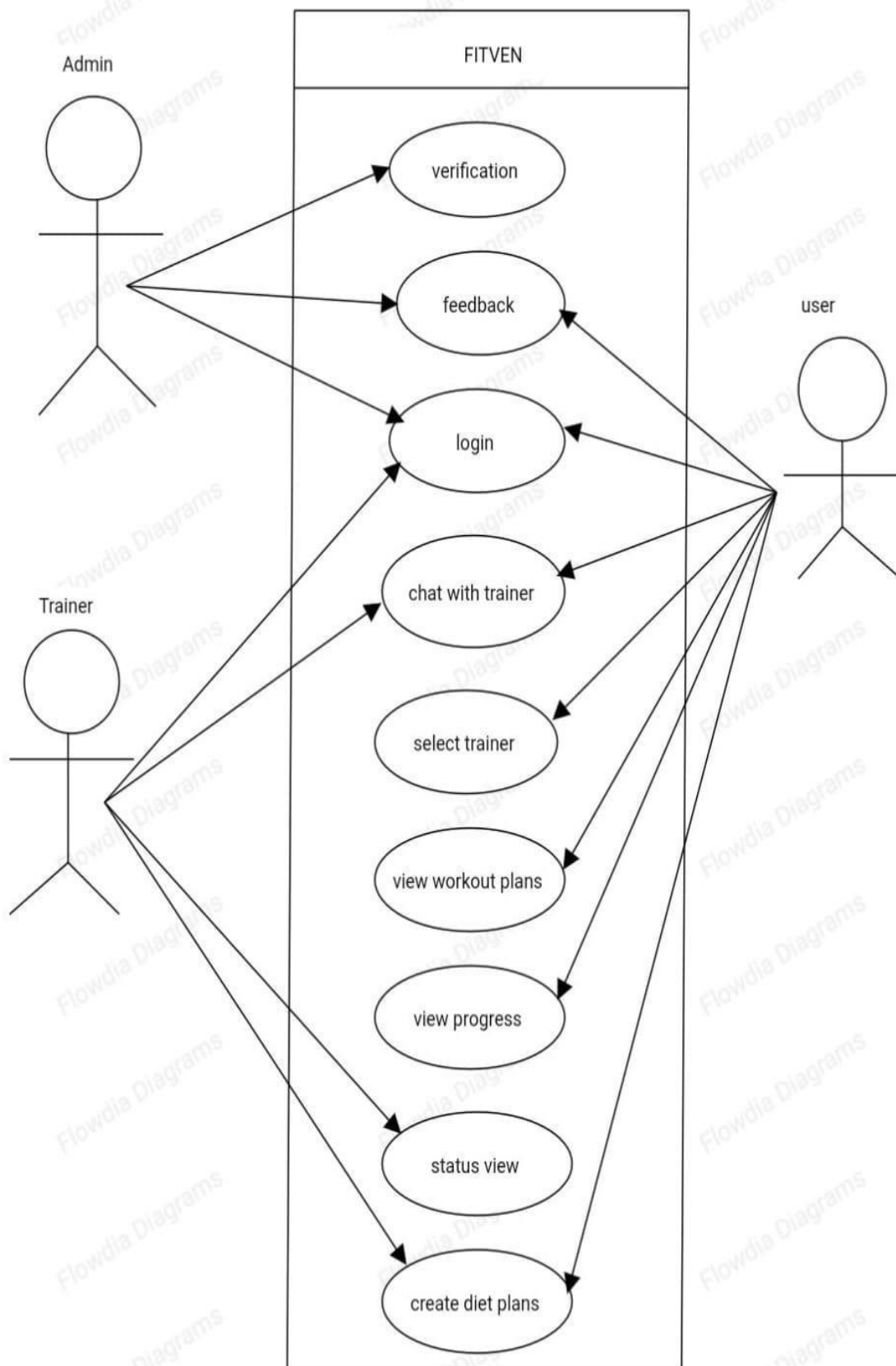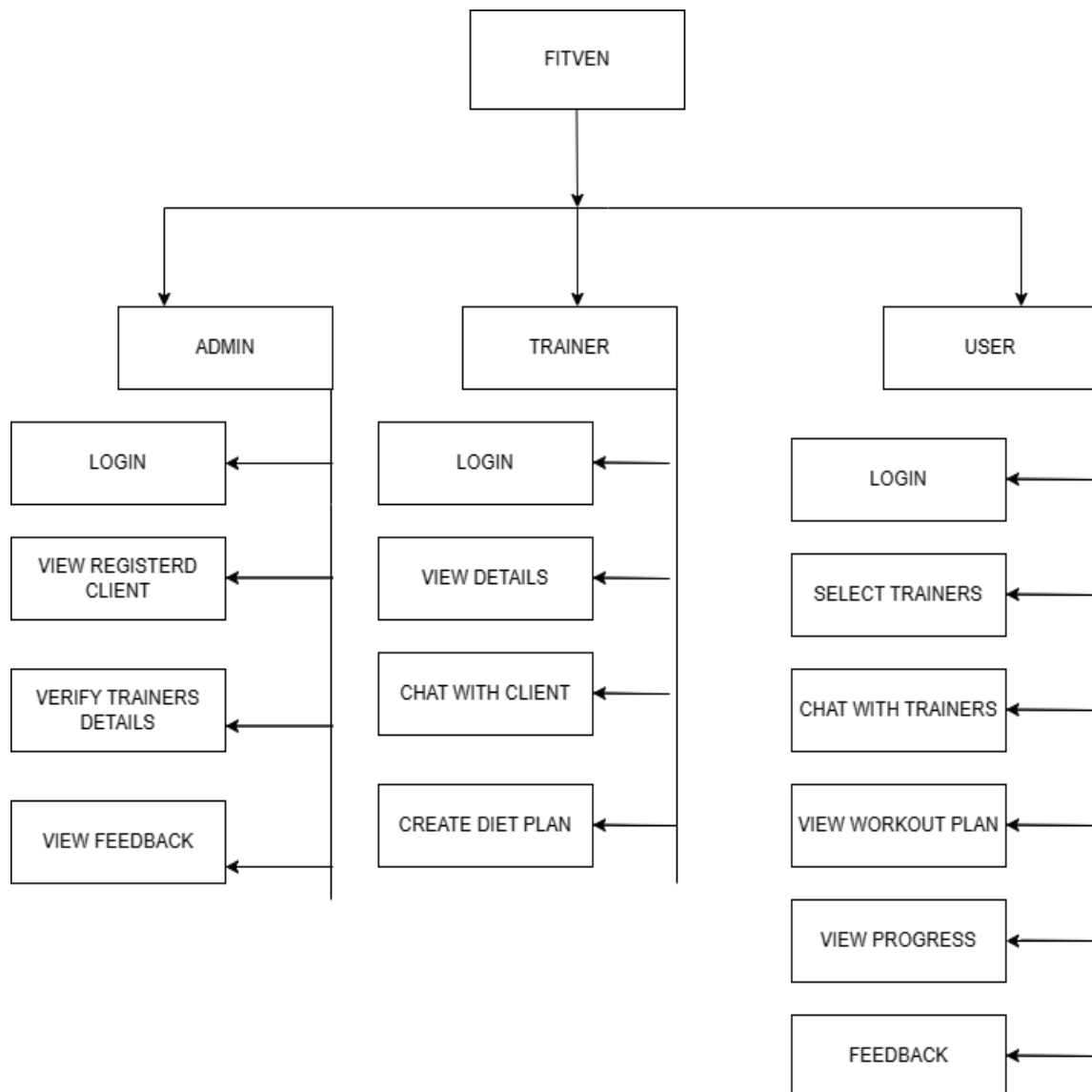Fig 4.4.5 Level 1 Data Flow Diagram of Website -Trainer

Fig 4.4.6 Level 2 Data Flow Diagram of Website - User

USECASE DIAGRAM

STRUCTURE CHART

4.4 SYSTEM DATA DESIGN

System data design is an essential aspect of system development that involves the creation of a structured and organized data storage and retrieval system. The database and configuration design are critical components of the data design process, as they provide the foundation for storing and managing the system's data. The database design includes the creation of tables, relationships, and constraints to ensure that data is stored accurately and efficiently. Configuration design, on the other hand, involves defining the various settings and parameters that govern the behavior of the system. Together, these design elements enable the system to operate effectively and efficiently, ensuring that data is stored, retrieved, and utilized in a manner that supports the system's overall functionality.

Website

1.Table Name: Admin   Description: This table is used for Admin details.

Field name	Datatype	Constrain	Description
Adminname	Varchar	Primary key	Video id
Password	Varchar	Not null	video name

2. Table Name: Trainersdetails

Description: This table is used for Trainer details

Name	Datatype	Constrain	Description
UserName	Varchar	Primarykey	Name of user
Email	Varchar	None	Email
Country	Varchar	none	Region
Password	Varchar	None	Password

Certification	Varchar	none	Certification details

3.Table Name: User

Description: This table is used for user details

Name	Datatype	Constrain	Description
UserName	Varchar	Primarykey	Name of user
Email	Varchar	None	Email
Country	Varchar	none	Region
Password	Varchar	None	Password
Date of birth	Date	none	DOB

4.Table Name: Login

Description: This table is used for User login details

Name	Datatype	Constrain	Description
UserName	Varchar	Primarykey	Name of user
Password	Varchar	None	Password

5.Table Name: Trainers Login

Description: This table is used for Trainer login details

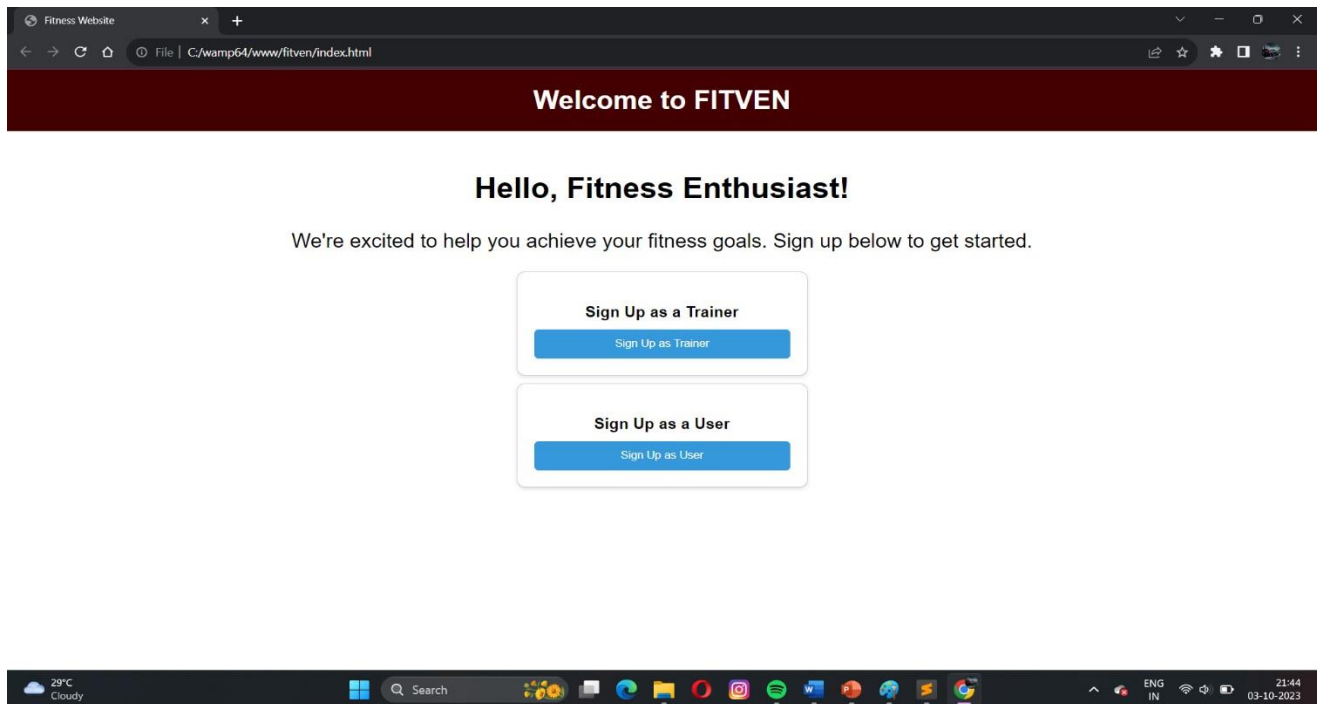| Name	Datatype	Constrain	Description
TrainerName	Varchar	Primarykey	Trainername
Password	Varchar	None	Password

Fig 4.7.6 Screenshot of Website's home page
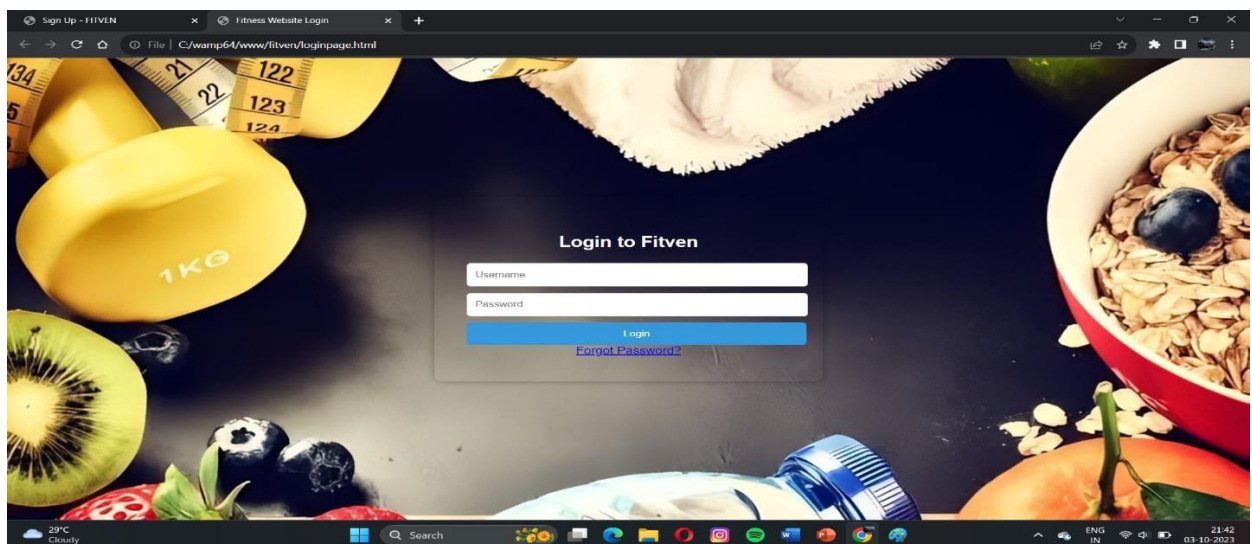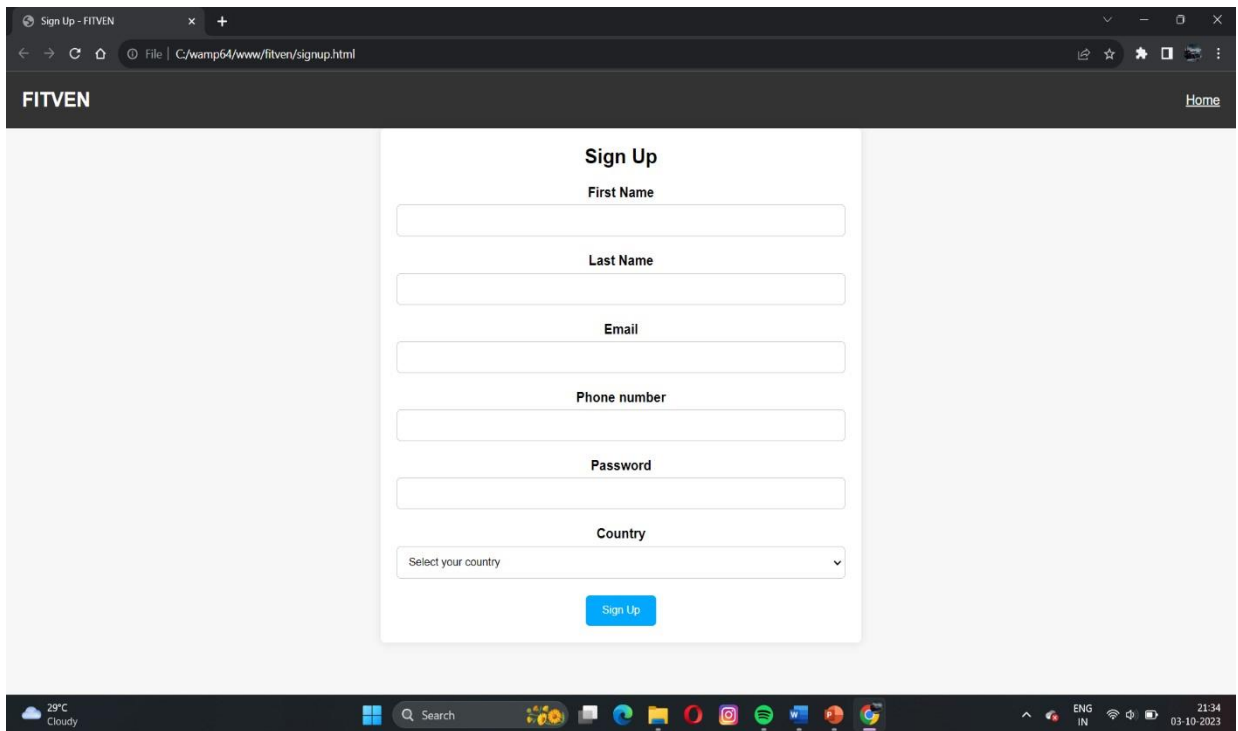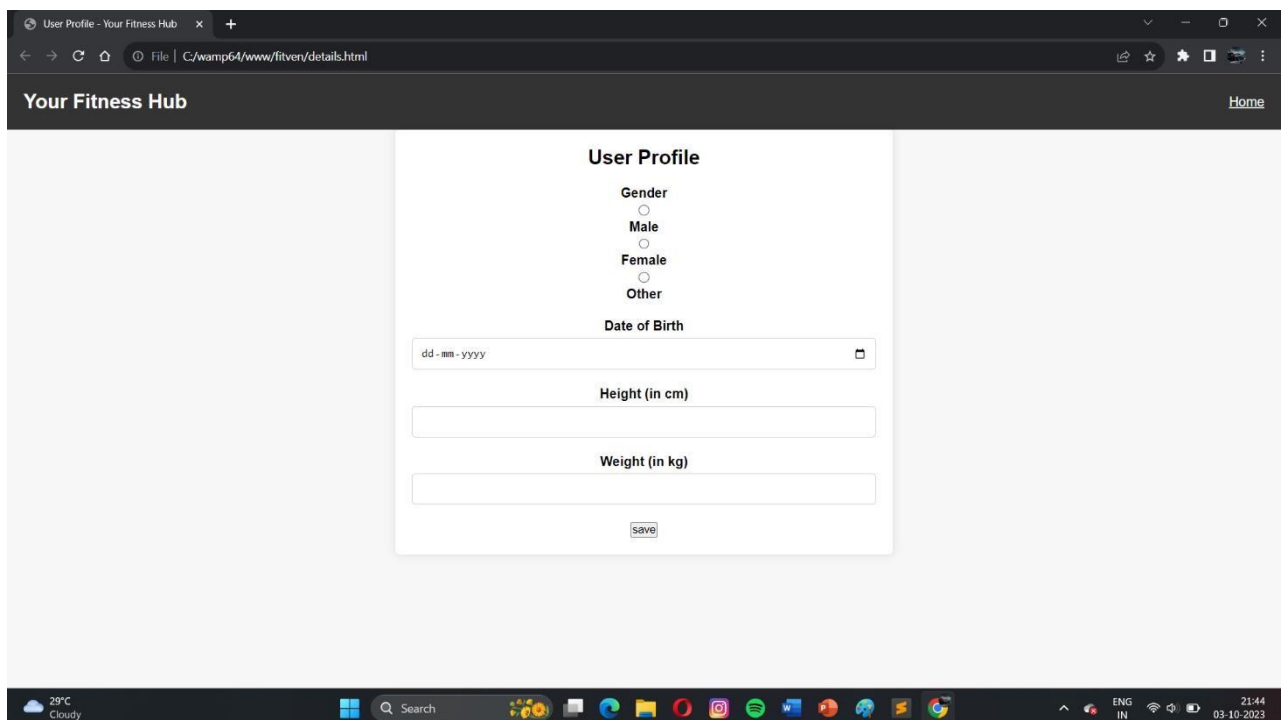
Fig 4.7.7 Screenshot of login page in website

Fig 4.7.8 Screenshot of registration in website

Fig 4.7.9 Screenshot of Basic details in website
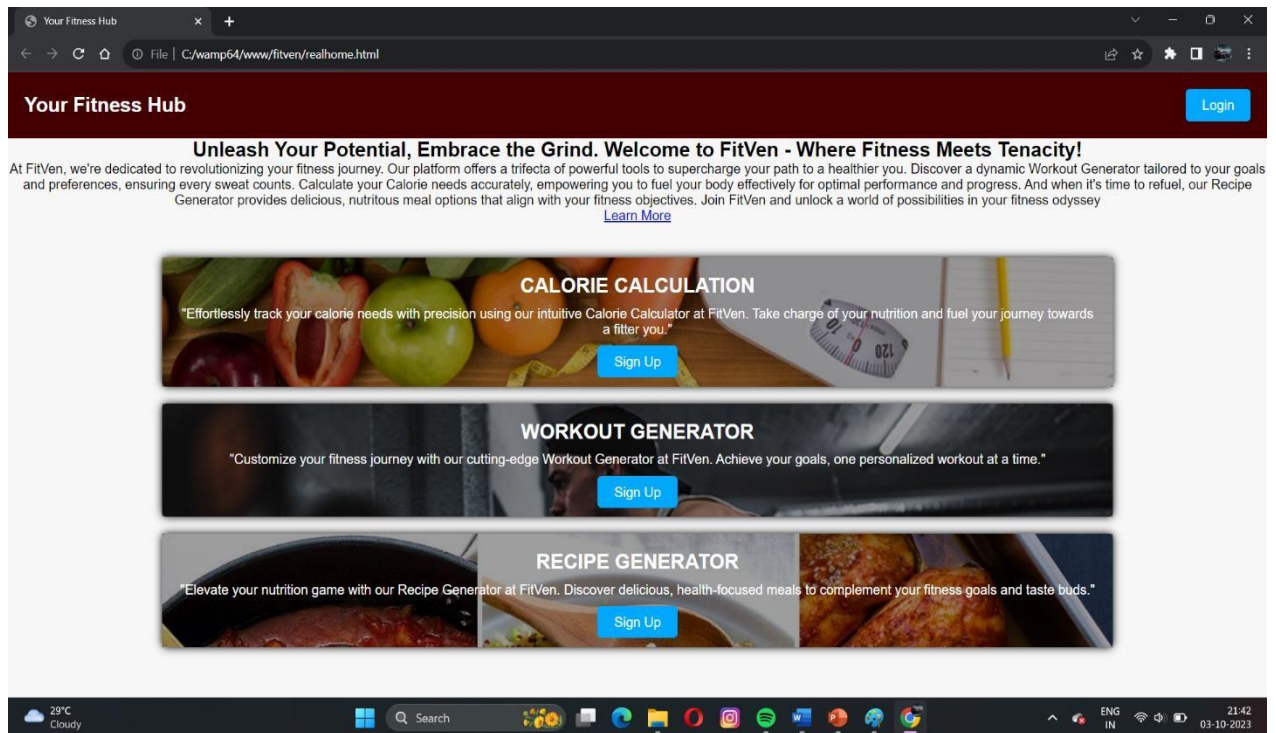
Fig 4.7.10 Screenshot of Features in website
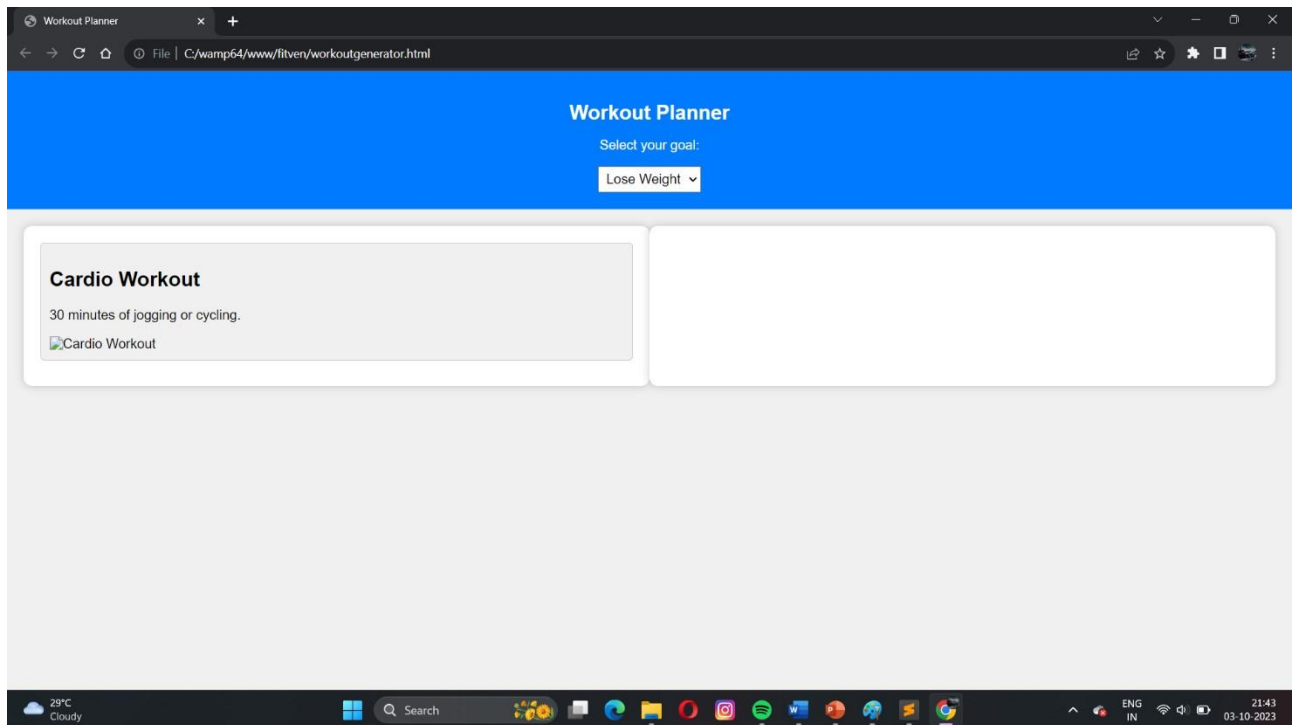
Fig 4.7.11 Screenshot of workout in website
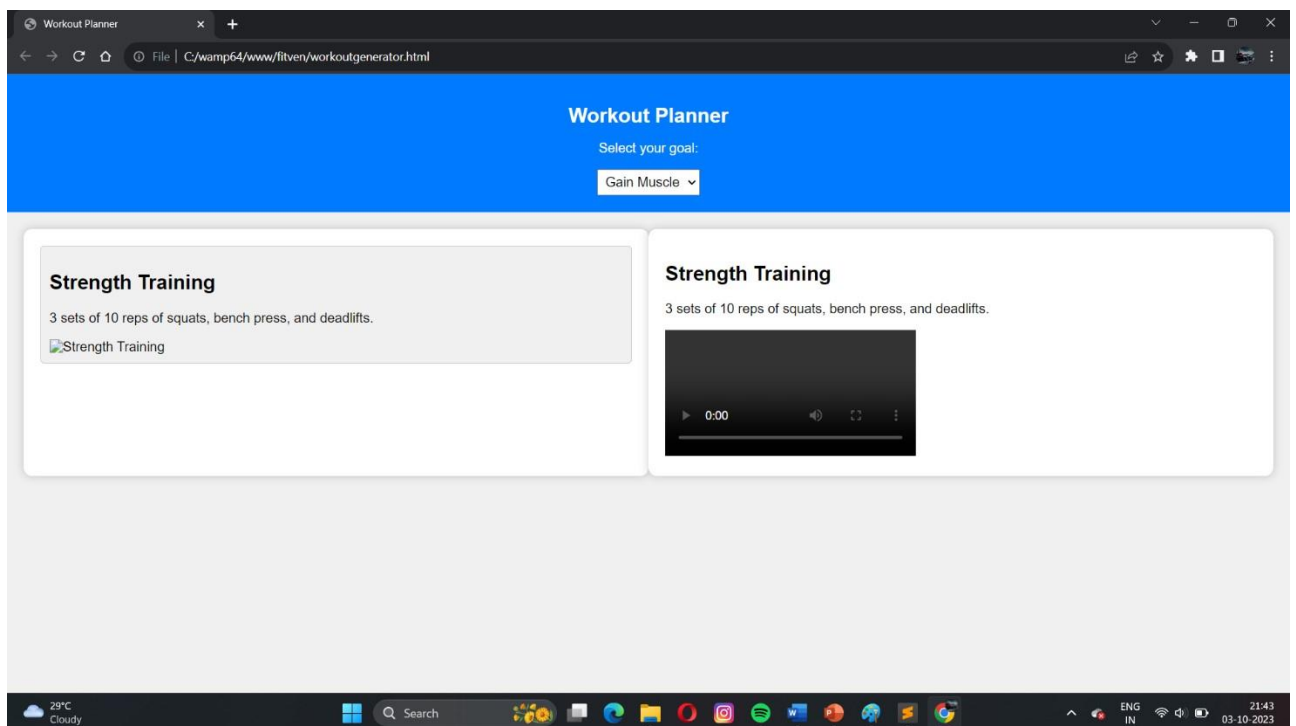
Fig 4.7.12 Screenshot of Workout tutorial in website
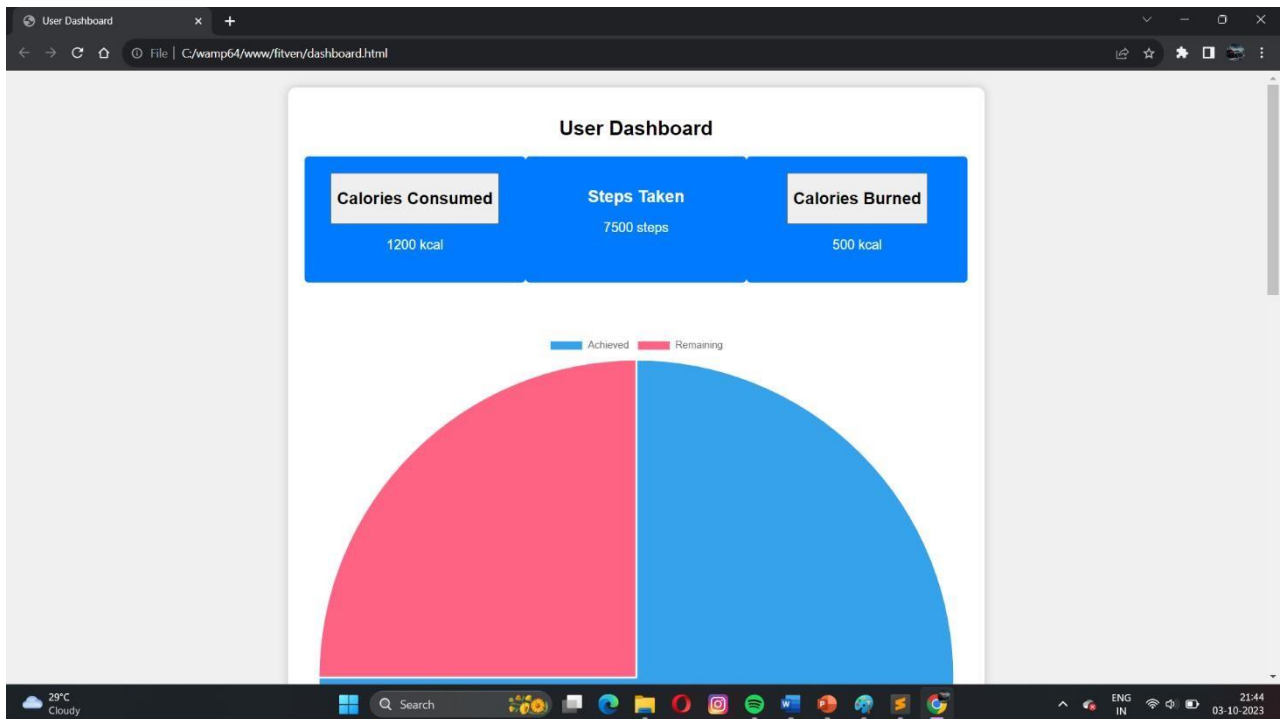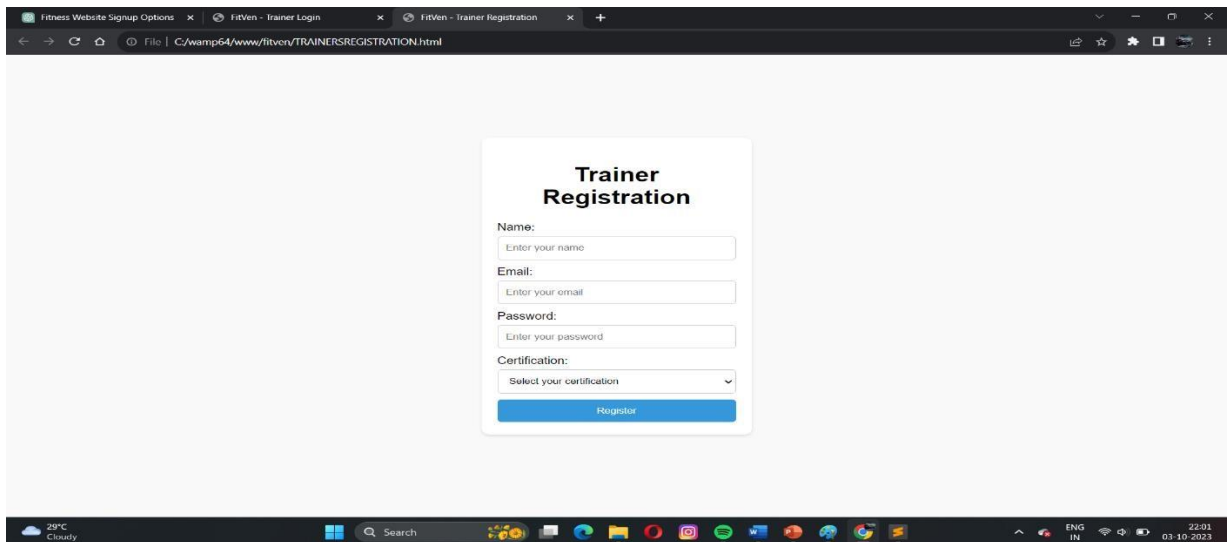
Fig 4.7.13 Screenshot of user home page

Fig 4.7.15 Screenshot of Trainer login page

Fig 4.7.16 Screenshot of Trainer registration page

6.CODING

INTRODUCTION

The coding phase in the software engineering paradigm is usually defined after the designing phase. In this phase, the developers or the coders have to implement the software design practically using any computer language(s) so that the software can be created and the user can use it. Now, coding is not just a matter of implementing the code in any suitable language as per the developer's choice. Some norms and standards are set for this purpose which is known as the coding standards.

In this project we have used GoLang as the programming language for the CLI application while HTML, PHP and JS have been used for the website. We Have adhered to the coding standards and ensured that good programming practices have been followed and there is a uniform appearance of the code.

5.1 CODING SNIPPETS

Code for trainerregistration.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>FitVen - Trainer Registration</title>
  <style>    body {    font-
family: Arial, sans-serif;
margin: 0;    padding: 0;
box-sizing: border-box;
```

```css
background-color: #f9f9f9;
display: flex;      justify-content:
center;      align-items: center;
height: 100vh;
    }

    .registration-container {      background-
color: #fff;      padding: 20px;      border-
radius: 10px;      box-shadow: 0 2px 4px
rgba(0, 0, 0, 0.1);      max-width: 300px;
width: 100%;
    }

    h1 {      text-align:
center;
    }

    form            {
display: flex;
     flex-direction: column;
    }

    label  {           margin-
bottom: 5px;
    }

    input, select {
padding: 10px;
margin-bottom: 10px;
border: 1px solid #ccc;
border-radius: 5px;
    }

    button {      padding: 10px;
background-color: #3498db;
color: #fff;      border: none;
border-radius: 5px;      cursor:
pointer;
    }
```

```
   button:hover {        background-
color: #2980b9;
   }
  </style>
</head>
<body>
 <div class="registration-container">
   <h1>Trainer Registration</h1>
   <form>
    <label for="name">Name:</label>
    <input type="text" id="name" name="name" placeholder="Enter your name" required>

    <label for="email">Email:</label>
    <input type="email" id="email" name="email" placeholder="Enter your email" required>        <label
for="password">Password:</label>
    <input type="password" id="password" name="password" placeholder="Enter your password"
required>

    <label for="certification">Certification:</label>
    <select id="certification" name="certification">
     <option value="" disabled selected>Select your certification</option>
     <option value="ACE">ACE - American Council on Exercise</option>
     <option value="NASM">NASM - National Academy of Sports Medicine</option>
     <option value="ISSA">ISSA - International Sports Sciences Association</option>
<option value="ACSM">ACSM - American College of Sports Medicine</option>
     <!-- Add more certification options as needed -->
    </select>

    <button type="submit">Register</button>
   </form>
  </div>
</body> </html>
```

Code for workoutplanner.html

```
<!DOCTYPE html>
<html lang="en">
<head>
```

```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Workout Planner</title>
    <style>
       /* Your CSS styles here */
body {          font-family: Arial,
sans-serif;          background-
color: #f0f0f0;          margin: 0;
padding: 0;
       }

       header {          background-
color: #007BFF;
          color: #fff;
text-align: center;
padding: 20px;
       }
h1 {
font-
size:
24px;
font-
weight
: bold;
       }          p {
font-size: 16px;
       }
        select {
padding: 5px;
font-size: 16px;
       }

       main {          display: flex;
justify-content: space-between;
padding: 20px;
       }

       #workout-list {
flex: 1;          background-
color: #fff;          padding:
```

```
20px;          border-radius:
10px;
       box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
     }
     #workout-list .workout-card {
border: 1px solid #ccc;
padding: 10px;          margin-
bottom: 10px;          border-
radius: 5px;          cursor:
pointer;
     }
     #workout-list .workout-card:hover {          background-
color: #f0f0f0;
     }

     #tutorial {          flex: 1;
background-color: #fff;
padding: 20px;          border-
radius: 10px;          box-
shadow: 0 0 10px rgba(0, 0, 0,
0.2);
     }
   </style>
</head>
<body>
   <header>
     <h1>Workout Planner</h1>
     <p>Select your goal:</p>
     <select id="goalSelect">
       <option value="lose-weight">Lose Weight</option>
       <option value="gain-muscle">Gain Muscle</option>
       <option value="stay-fit">Stay Fit</option>
     </select>
</header>

   <main>
     <section id="workout-list">
       <!-- Workout cards will be added dynamically here -->
     </section>
```

```html
    <section id="tutorial">
       <!-- Tutorial content will be displayed here -->
    </section>
</main>
```

```html
  <script>        const goalSelect =
document.getElementById('goalSelect');        const workoutList =
document.getElementById('workout-list');        const
tutorialSection = document.getElementById('tutorial');

    const workouts = {
       'lose-weight': [
          {
             name: 'Cardio Workout',              description: '30
minutes of jogging or cycling.',              image:
'image_processing20210907-11148-m3b472.gif'
          },

       ],
       'gain-muscle': [
          {
             name: 'Strength Training',              description: '3 sets of 10
reps of squats, bench press, and deadlifts.',              image:
'Weightlifting_V6-SD'
          },

       ],
       'stay-fit': [
          {
             name: 'Full-Body Workout',              description:
'Combination of cardio and strength exercises.',              image:
'gym-guy-8x6'
          },

       ],
};
```

```javascript
    goalSelect.addEventListener('change', () => {
const selectedGoal = goalSelect.value;
displayWorkouts(selectedGoal);
    });

    function displayWorkouts(goal) {
        workoutList.innerHTML = '';

        const goalWorkouts = workouts[goal];

        if (goalWorkouts) {
goalWorkouts.forEach(workout => {                const
workoutCard = document.createElement('div');
workoutCard.classList.add('workout-card');
workoutCard.innerHTML = `
<h2>${workout.name}</h2>
                <p>${workout.description}</p>
                <img src="${workout.image}" alt="${workout.name}">

`;

        workoutCard.addEventListener('click',        ()        =>        {
displayTutorial(workout);
        });

        workoutList.appendChild(workoutCard);
      });
    }
}

    function              displayTutorial(workout)              {
tutorialSection.innerHTML = '';

        const        tutorialTitle        =        document.createElement('h2');
tutorialTitle.textContent = workout.name;

        const            tutorialDescription            =            document.createElement('p');
tutorialDescription.textContent = workout.description;
```

```
      const tutorialVideo = document.createElement('video');
tutorialVideo.controls = true;          const source =
document.createElement('source');          source.src =
workout.video;          source.type = 'video/mp4';
      tutorialVideo.appendChild(source);

      tutorialSection.appendChild(tutorialTitle);
tutorialSection.appendChild(tutorialDescription);
tutorialSection.appendChild(tutorialVideo);
    }

    // Initial display based on the default selected goal
displayWorkouts(goalSelect.value);
  </script>
</body> </html>
```

6.SYSTEM TESTING

6.1 INTRODUCTION

System testing is an important part of the software development life cycle that evaluates the system's compliance with its specified requirements. This testing process is conducted on a complete and integrated system to verify its functionality and performance. The objective of system testing is to detect any defects in the system, including any mismatches in the software and hardware components, and ensure that it meets the user's requirements. It also verifies that the system is ready for deployment and is capable of handling real-world scenarios. System testing can be performed manually or using automated testing tools and techniques to ensure the system is reliable, scalable, and secure.

6.1.1 Unit Testing

Unit testing is a software testing method in which individual units or components of a software application are tested in isolation. The purpose of unit testing is to ensure that each individual unit is functioning as expected and that there are no defects or errors. This method involves writing test cases for each unit, running those test cases, and analyzing the results. Unit testing is typically automated, and it is performed by developers during the development process. By detecting and correcting defects early in the development cycle, unit testing helps to reduce the overall cost of software development and improve the quality of the final product.

6.1.2 Integration Testing

Integration testing is a software testing method in which individual units or components of a software application are combined and tested as a group. The purpose of integration testing is to ensure that the different units or components of the software work together as expected and that there are no defects or errors introduced by the integration. This method involves testing the interactions between different units, verifying the data flow between them, and validating the overall behavior of the software application.

Integration testing can be performed manually or automated, and it is typically performed by developers or testers during the development cycle.

6.1.3 Functional Testing

Functional testing is a software testing method in which the functionality of a software application is tested against the functional requirements or specifications. The purpose of functional

testing is to ensure that the software application is performing as expected and that it meets the functional requirements of the end-users. This method involves testing the software application through various scenarios, verifying the input and output data, and analyzing the results. Functional testing can be performed manually or automated, and it is typically performed by testers during the testing phase of the software development life cycle.

6.1.4 Regression Testing

Regression testing is a software testing method in which the software application is retested after a change or modification has been made to the software. The purpose of regression testing is to ensure that the change or modification has not introduced any new defects or errors into the software application. This method involves running a suite of previously executed test cases on the modified software application, verifying the results, and comparing them with the results from the previous version of the software. Regression testing can be performed manually or automated, and it is typically performed by testers during the testing phase of the software development life cycle.

6.1.5 Performance Testing

Performance testing is a software testing method in which the performance of a software application is tested under different load conditions. The purpose of performance testing is to ensure that the software application can handle the expected load and that it meets the performance requirements of the end-users. This method involves simulating realworld scenarios, measuring the response time of the software application, and analyzing the results. Performance testing can be performed manually or automated, and it is typically performed by testers during the testing phase of the software development life cycle.

6.1.6 Acceptance Testing

Acceptance testing is a software testing method in which the software application is tested against the acceptance criteria or requirements specified by the end-users. The purpose of acceptance testing is to ensure that the software application meets the expectations of the end-users and that it is ready for release. This method involves testing the software application through various scenarios, verifying the input and output data, and analyzing the results. Acceptance testing can be performed manually or automated, and it is typically

performed by the end-users or a representative during the acceptance phase of the software development life cycle.

6.2 WEBSITE

The following methods of testing were carried out to ensure reliability of the website

6.2.1 Functional testing

Functional testing is a type of software testing that verifies the behavior of a software application against the functional requirements or specifications. The purpose of functional testing is to ensure that the software application performs all the functions as per the requirement and there are no defects or errors in the functionalities. It involves identifying the functional requirements, creating test cases, executing them, and comparing the actual results with the expected results. Functional testing can be done manually or automated, and it is usually performed by testers during the testing phase of the software development life cycle.

6.2.2 Performance testing

Performance testing is a type of software testing that checks how well the software application performs under different load conditions. The purpose of performance testing is to identify the performance issues and bottlenecks in the software application, and to ensure that the application meets the performance requirements of the end-users. It involves simulating the real-world scenarios, measuring the response time of the software application, and analyzing the results. Performance testing can be done manually or automated, and it is usually performed by testers during the testing phase of the software development life cycle.

6.2.3 Security testing

Security testing is a type of software testing that checks the security of the software application. The purpose of security testing is to identify the vulnerabilities and loopholes in the software application, and to ensure that the application is secure and free from any potential threats. It involves identifying the security requirements, creating test cases, executing them, and analyzing the results. Security testing can be done manually or automated, and it is usually performed by testers during the testing phase of the software development life cycle.

6.2.4 Usability testing

Usability testing is a type of software testing that evaluates the ease of use of the software application. The purpose of usability testing is to identify the issues and difficulties faced by the end-users while using the software application, and to ensure that the application is user-friendly and easy to use. It involves identifying the usability requirements, creating test cases, executing them, and analyzing the results. Usability testing can be done manually or automated, and it is usually performed by testers during the testing phase of the software development life cycle.

6.2.5 Compatibility testing

Compatibility testing is a type of software testing that checks the compatibility of the software application with different hardware, software, and operating systems. The purpose of compatibility testing is to ensure that the software application works smoothly on different platforms, and to identify any compatibility issues. It involves identifying the compatibility requirements, creating test cases, executing them on different platforms, and analyzing the results. Compatibility testing can be done manually or automated, and it is usually performed by testers during the testing phase of the software development life cycle.

6.2.6 Regression testing

Regression testing is a type of software testing that verifies that the changes made in the software application have not affected the existing functionalities. The purpose of regression testing is to ensure that the changes made in the software application have not introduced any new defects or errors, and to ensure that the software application is still working as expected. It involves identifying the changes made, creating test cases, executing them, and analyzing the results. Regression testing can be done manually or automated, and it is usually performed by testers during the testing phase of the software development life cycle.

6.2.7 User acceptance testing

User acceptance testing is a type of software testing that checks whether the software application meets the expectations of the end-users. The purpose of user acceptance testing is to ensure that the software application is ready for release and that it meets the

requirements of the end-users. It involves creating test cases based on the user requirements, executing them, and analyzing the results. User acceptance testing is usually performed by the end-users or the representatives of the end-users during the acceptance phase of the software development life cycle.

7.SYSTEM IMPLEMENTATION

6.3 INTRODUCTION

System implementation is the process of putting a software system into action or making it operational. It involves a series of activities that are necessary for deploying the system and making it ready for use by the end-users. This includes the installation and configuration of hardware and software components, database setup, and integration with other systems. The implementation process also involves testing the system to ensure it meets the requirements and specifications outlined during the design and development phases. The goal of system implementation is to ensure a smooth and seamless transition from the old system to the new one, minimizing disruption to business operations and maximizing the benefits of the new system.

6.4 WEBSITE

Implementation is an activity that is contained throughout the development phase. It is the process of bringing a developed system into operational use and turning it over to the user. The new system and its components are to be tested in a structured and planned manner. A successful system should be delivered and users should have the confidence that the system would work efficiently and effectively. The more complex the system being implemented the more involved will be the system analysis and design effort required for implementation. Implementation is the stage of the system when the theoretical design is turned into working system. The implementation involves careful planning investigation of the current system and its constraints on implementing, design of methods to achieve the changeover, training of user over procedure and evaluation change over method. There are three types of implementation:

Maintenance corresponds to restoring something to original conditions, covering a wide range of activities including correcting codes and design errors and updating user support. Maintenance is performed most often to improve the existing software rather than to a crisis or risk failure. The system would fail if not properly maintained. The software maintenance is an important one in the software development because we have to spend more efforts for maintenance. Software maintenance is to improve the software quality according to the requirements. After a system is successfully implemented, it should be maintained in a proper manner. The need for system maintenance is to make the system adaptable to the changes in the system environment. There may be social, economical or technical changes, which affect system being implemented.

8.CONCLUSION

All the requirements and specifications was followed as for as possible and few additional features were added that can make the application more user friendly and less complicated. The project was successfully completed within the time span allotted. All the modules are tested separately and put together to form the main system. Finally the system is tested with real data and it worked successfully. Thus the system has fulfilled the entire objective defined.

9.FUTURE ENHANCEMENT

Any system which has been in use for a number of years gradually decays and become less effective because of change in environment to which it has to be adapted. For the time being it is possible to overcome problems by amendments and minor modifications to acknowledge the need of fundamental changes.

Nothing is perfect; in future we would like our prototype to be more perfect by including future improvements. The proposed system is very easy to handle so everyone can access.

*   The entire process of the firm can be computerized.
*   More number of item details can be inserted into the database.
*   It can be integrated with the web for universal access.
*   Upgrading the performance .This system is now implemented at the client machine only but as a future enhancement we can modify the system in such a way to make it work on a client- server network.

The system can be even more enhanced by making it an internet based system.

The efficiency of the developed system can be improved by modification. Today most of the services are based on web services. This system can also be connected online.

10.BIBLOGRAPHY

1. Max Kanat-Alexander, K. (2012). "Code Simplicity: The Fundamentals of Software" [PDF]. O'Reilly. Available at: www.codesimplicity.com/wp-content/uploads/2022/05/CodeSimplicity.pdf

2. Bootstrap documentation [online]. Available at: getbootstrap.com/docs

3. W3Schools [online]. Available at: www.w3schools.com

4. www.slideshare.com

5. www.project management basics.com

6. www.SDLC tutorial.com

7. MySql:https://www.w3schools.com/java/java_mysql_insert.asp