VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI, KARNATAKA, INDIA

DBMS MINI-PROJECT REPORT

on

FITNESS MANAGEMENT SYSTEM

Submitted in partial fulfillment of the requirement

for the V semester B.E in Information Science and Engineering

DBMS MINI-PROJECT-18CSL58

Bachelor of Engineering

in

Information Science & Engineering

Submitted by

HITESH S	**1EW20IS036**
GAURAV V BHAT	**1EW20IS031**

Under the guidance of

Dr. SHRUTHI T V

Associate Professor

Dept. of I.S.E, E.W.I.T

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

EAST WEST INSTITUTE OF TECHNOLOGY

Bengaluru-57

2022-23

SRI SRINIVASA EDUCATIONAL AND CHARITABLE TRUST®

EAST WEST INSTITUTE OF TECHNOLOGY

(Affiliated to Visvesvaraya Technological University, Belagavi and Approved by AICTE, New Delhi)
(Accredited by NAAC with "A" grade)(NBA Accredited-CSE, ECE,EEE,ISE, ME) (An ISO 9001:2015 & ISO 14001:2015 Certified)

DEPARTMENT OF ISE

Department of Information Science & Engineering

CERTIFICATE

This is to certify that the DBMS Mini-Project work entitled **"Fitness Management System"** is a bonafide work carried out by **Mr. Gaurav V Bhat(1W20IS031), Mr. Hitesh S(IEW20IS036)**, in partial fulfillment for the award of the degree of **Bachelor of Engineering** in **Information Science and Engineering** of the **Visvesvaraya Technological University**, Belagavi during the year 2022-23. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The Project has been approved as it satisfies the academic requirements in respect of Project work prescribed for the Bachelor of Engineering Degree.

Signature of the Guide	**Signature of the HOD**
Dr. SHRUTI T V	**Dr. SURESH M B**
Associate Professor	Head of Dept, ISE
Name of the Examiners	**Signature with date**

ACKNOWLEDGEMENT

Any achievement depends on the individual efforts but on the guidance, encouragement and co-operation of intellectuals, elders and friends. A number of personalities have helped us. We would like to take this opportunity to thank them all.

We would take this opportunity to express our heart-felt gratitude to **Sri. G. Dayanand**, Chairman, and **Dr. H Ramakrishna,** Principal, Sapthagiri College of Engineering, Bengaluru for providing the necessary infrastructure to complete this mini-project.

Our deep sense of gratitude to **Dr. H.R. Ranganatha**, Head of the Department, Information Science and Engineering, Sapthagiri College of Engineering, Bengaluru, for his constant support and encouragement.

We wish to express sincere thanks to our guide **Dr. Prerana Chaithra,** Associate Professor, Department of Information Science and Engineering, Sapthagiri College of Engineering, Bengaluru for helping us throughout and guiding us from time to time.

We also extend our sense of gratitude and sincere thanks to all the **faculty members and non-teaching staff members** of Department of Information Science and Engineering, Sapthagiri College of Engineering, Bengaluru who have helped me with their views and encouraging ideas.

We are in debt to our **Parents, siblings and friends** for being an epitome of support system throughout our life and being with us through our thick and thin situations.

Shyam G (1SG20IS090)

Thanush T D(1SG20IS102)

ABSTRACT

Fitness Management System developed using PHP is an excellent solution for fitness clubs with a large/growing number of members, or ones serving elite client. This solution helps to identify the user and manage their timely memberships.

In its working, each member is issued a membership card which is valid for a fixed number of fitness sessions, or for a particular period of time, or a combination of the two, totally based on the payment policy. Once the time-frame or number of sessions expire, the machine notifies the member about the payment of renewal.

Hence, the system reduces hassle and any chances of quarrels between the members and the fitness management. It can also generate multiple reports like monthly, weekly, daily, session wise.

Unlike modern Gyms, the Fitness clubs provide membership for people of any age group, from college students to old age people with a suitable fitness training session.

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 Data, Database and DBMS

Data is a known fact that can be recorded. It can also be defined as an information that has been translated into a form that is efficient for movement or processing. Eg: name, USN, address, crop name, etc. A datum is a unit of data. Meaningful data combines to form information. Hence, information is Interpreted data i.e. data provided with semantics Database is a collection of related data without an implicit meaning. It can also be defined as a structured set of data held in a computer, especially one that is accessible in various ways. Eg. student database, farmer database, employee database, etc.

Database management system (DBMS) is a collection of programs that enable users to create and maintain the database. It is a general-purpose software which contains defining, constructing, manipulating, sharing, protecting the database. Basically, a DBMS is a software tool to organize (create, retrieve, update and manage) data in a database. The main aim of a DBMS is to supply a way to store and retrieve database information that is both convenient and efficient. By data, we mean known facts that can be recorded and that have embedded meaning. Normally people use software such as DBASE IV or V, MySQL, Microsoft ACCESS, or EXCEL to store data in the form of database.

Database systems are meant to handle large collection of information. Management of data involves both defining structures for storage of information and providing mechanisms that can do the manipulation of those stored information. Moreover, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access.

1.2 Components of a DBMS

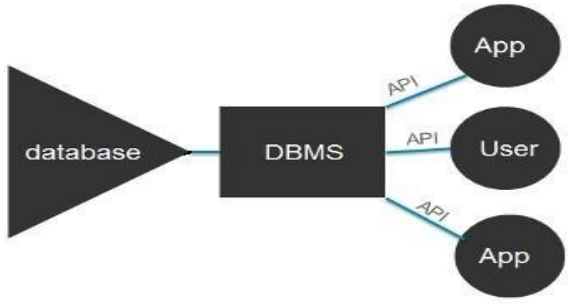


Fig.1.1 Components of a DBMS

- Users: Users may be of any kind, such as database administrators, system developers or database users.
- Database application: Database application may be Departmental, Personal, Organizational and /or Internal.
- DBMS: Software that allows users to create and manipulate database access.
- Database: Collection of logical data as a single unit.

1.3 Relational Data Model

The relational model (RM) for database management is an approach to managing data using a structure and language consistent with first-order predicate logic, first described in 1969 by English computer scientist Edgar F. Codd, where all data is represented in terms of tuples, grouped Into relations. A database organized in terms of the relational model is a relational database.

The purpose of the relational model is to provide a declarative method for specifying data and queries: users directly state what information the database contains and what information they want from it, and let the database management system software take care of describing data structures for storing the data and retrieval procedures for answering queries.

Most relational databases use the SQL data definition and query language; these systems implement what can be regarded as an engineering approximation to the relational model. A table in an SQL database schema corresponds to a predicate variable; the contents of a table to a relation; key constraints, other constraints, and SQL queries correspond to predicates. However, SQL databases deviate from the relational model in many details, and Codd fiercely argued against deviations that compromise the original principles.

1.4 ER To Relational Model Mapping Algorithm

This algorithm consists of 7 steps:

1. Mapping strong entity to relations:

   For each strong entity type ‘E’ create a table or relation R that includes all the simple attributes of E. Since the strong entities will have their own primary keys, those attributes will become the keys for respective table.

2. Mapping weak entity to relations:

For each weak entity 'W' in the ER diagram, with owner entity type 'E', create a new table or relation and include all simple attributes of W as attributes of R. In addition, include primary key attribute of the relation that corresponds to owner entity as foreign key of R. The primary key of R is the primary key of owner entity and partial key of weak entity.

3. Mapping binary 1:1 relationship type:

   For each binary 1:1 relationship type 'R' in the ER diagram, identify the relations S and T that corresponds to entity type participates in R. There are three approaches:

   - Foreign key approach: Choose of the relations S (total participation side) and include primary key of T as foreign key of S. Include all the simple descriptive attributes of 1:1 relationship types as the attributes of S.
   - Merge relationship approach: In this approach, merge the two entity types and relationship types Into a single relation. This maybe appropriate when both participations are total.
   - Cross reference / relationship approach: In this approach, setup a new relation R for the purpose of cross-functioning the primary keys of S and T.

4. Mapping binary 1:N relationship types:

   For each 1:N relationship type R, identify the relation S that represents the participating entity type at the N side of relationship type. Include primary key of S as he foreign key of T and also include all descriptive attributes.

5. Mapping binary M:N relationship types

   For each M:N relationship R, create a new relation S to represent R. Include primary keys of the relations that represent the participating entity types as foreign keys of S and their combination will form the primary key for S and also include all the simple descriptive attributes of R.

6. Mapping multivalued attributes:

   For each multivalued attribute 'A', create a new relation and include an attribute corresponding to A and primary key 'k' of the participating entity or relationship. The primary key of new relation is combination of A and k.

7. Mapping of n-ery relationship type:

   For n-ery relationship type R, n>2, create a new relation to represent R. Include primary keys of all participating entity types as foreign keys and also include all

descriptive attributes of R, The primary key of newly created relation is the combination of all the foreign keys.

1.5 Normalization

It is a design for producing a set of good tables with all constraints incorporated as per the business constraints. Normalization process takes a relation schema through a series of tests to verify whether it satisfies a certain normal form. Normalization of relations is based on the functional dependencies and primary keys to achieve desirable properties of minimizing redundancy and minimizing update anomaly. Types of normal form are:

1. First Normal Form (1NF)

   A relation schema R is in 1NF if every attribute of R takes only a single value. When a table contains multivalued attribute, we say that its not in first normal form. We identify multivalued attributes and remove using following techniques:

   (i) Use multiple tuples one per value

   (ii) Use multiple columns one per value

   (iii) Use a separate table

2. Second Normal Form (2NF)

   A relation schema R is in 2NF if it satisfies 1NF and also if all non-prime attributes A in R should be fully functionally dependent on primary key in R. The test for 2NFinvolves testing for functional dependencies whose LHS attribute are part of primary key. If primary key contains a single attribute, the test need not be applied. If a relation schema is not in 2NF, it can be second normalized Into a number of 2NF relations in which non-prime attributes are associated only with the part of the primary key on which they are fully functionally dependent.

3. Third Normal Form (3NF)

   A relation schema R is in 3NF if it is in 2NF and no non-prime attributes of R is transitively dependent on primary key of R. A functional dependency X->Y is a transitive dependency if there's a set of attributes Z that is not a subset of any key of R and both X->Z and Z->Y holds.

1.6 Objective of Project

Our proposed "Fitness Management System" is for those who run a gym business. Before doing anything we did a decent research on major difficulties for gym owners. We

examined carefully about how to make a huge registering system without failure as well as different functions for different kind of user depending on their privilege.

The Gym Management requires a system that will handle all the necessary and minute details easily and proper database security accordingly to the user. They require software, which will store data about members, Trainers, classes, payments & all transactions that occur in Gym.

1.7 Objectives

The objectives of this study are summarized below:

- The main objective of the project is to design and develop a user friendly efficient computerized Gym Management System.
- An accurate system without any data redundancy.
- Secured data storage for Authority end.
- Secure the user ends data by providing each user's own personal credentials.
- A flexible system which can maneuver the customer-staff relationship in an effective manner.
- To provide better graphical user interface.
- Computerization can be helpful as means of saving time & money.

Chapter 2

LITERATURE SURVEY

Usually, the client uses MS Excel or paper, and maintains their records, however it is not possible for them to share the data from multiple system in multi user environment, there is lot of duplicate work, and chance of mishap. Every Excel file need to be updated once records are been changed The Fitness World System rejects most of the disadvantages of the existing software. Increasing efficiency and effectiveness, automation, accuracy, user-friendly interface, information availability, communication capacity, maintenance, cost deduction makes our system smarter than the existing system. We intermingle some new and blatant features along with all the necessary features. Some of them are user login, platform independent.

Fitness World enables the effortless scheduling of training resources, massage facility, appointment with Nutritionist, and generation of fitness reports. The system helps manage meetings with various resources, giving details of the involved members and employees in charge. The members of the gym can be classified on the basis of their membership programmers and they can be assisted in planning out their diet if needed. Based on the calorie need of individual diet is been categorized and allotted accordingly. The system enables generation of exercise cards, fitness reports for each member to track individual member progress.

This system not only evaluates all the information of each and every member but it also gives us to future updations and also this system showing a messaging system where user can easily understand about it. Fitness

Freak allows the gym to gain significant time benefit, enhance productivity and cost savings, allowing the owner to better monetize your gym's resources and staff, thus helping to improve the overall profitability. The system of Gym Management also included complaints and feedback from our employee base and user base to greatly develop our system features. Fitness World provides a wide range of reporting on members and programmers.

Everyone knows that health is a wealth and without a health We do not need a anything which give us happiness so that's why healthy personality is a most important thing we need to keep in our mind.in a normal and secure life view our energy to do anything is got Literature Introduction Survey

through healthy and fitness of our body. For a healthy and tension free life a Physical fitness is very necessary and important. Physical fitness includes exercise and sleep, diet. there are three basic things have own importance in each individual and personal life and everyone should be sensible with regard to these for a healthy, fit as well as happy life. Fitness World is a gym and health club membership management system. It is a system designed to manage membership data files, employee records and their related functions. Fitness World enables the effortless scheduling of appointment with Nutritionist, massage facility, training resources and creation reports of fitness.

The system helps manage appointments with various resources, giving details of the involved members and employees in charge. The members of the gym can be classified on the basis of their membership programs and they can be assisted in planning out their diet if needed. Diets are classified based on the calorie needs of an individual. The system enables generation of exercise cards, fitness reports for each member to track individual member progress. Fitness World system not only encapsulate all the important information of each and every member but it also provides for future updations and also provide a user login as well as manager login for security purpose. Fitness World allows the enhance productivity and cost-savings, gym to gain significant time benefit, allowing the owner to better monitoring your gym's resources and staff provide owner login for handling system, thus helping to develop and improve the overall profitability and ability of system. Our Fitness World System also included messaging, mail to customer feed-back, complaints from our employee base and user base to develop our system features. Fitness World provides a wide range of reporting on members and programs. Fitness World helps the gym to retain members, save time, give better service and track its performance, keep information.

Chapter 3

SYSTEM REQUIREMENT SPECIFICATION

3.1 Hardware Requirements

- CPU: Intel(R) Core(TM) i3-7700HQ
- RAM: 8GB
- GPU: Nvidia GeForce 1050 Ti or Built in GPU
- Peripherals: Standard Laptop Keyboard, and Laptop Scroll pad.

3.2 Software Requirements

- MySQL server 5.7.14
- phpMyAdmin 8.0.25
- xampp server 127.0.0.1
- Operating System: Windows 10

3.3 Software Tools Used

The whole project is divided in two parts the front end and the back end.

3.3.1 Front End -The front end is designed using of HTML, PHP, CSS, BOOTSTRAP.

HTML - HTML or Hyper Text Markup Language is the main markup language for creating web pages and other information that can be displayed in a web browser.HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like), within the web page content. HTML tags most commonly come in pairs like <h1> </h1>, although some tags represent empty elements and so are unpaired, for example **Error! Filename not specified.**. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, further tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to

Interpret the content of the page.HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create Interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behaviour of HTML web pages.

CSS - Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and Interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification Library Management System Division Of Computer Science And Engineering Page 17 of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design).CSS can also allow the same mark-up page to be presented in different styles for different rendering methods, such as on-screen, in prInt, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. However if the author or the reader did not link the document to a specific style sheet the default style of the browser will be applied.CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities or weights are calculated and assigned to rules, so that the results are predictable.

PHP - PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244

million websites and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1995, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Pre-processor, a recursive backronym. PHP code is Interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly Into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line Interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

Bootstrap – Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript based design templates for typography, forms, buttons, navigation and other Interface components. Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of colour, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-coloured tables, page headings, more prominent pull quotes, and text with a highlight.

3.3.2 Back End

The back end is designed using MySql which is used to design the databases.

MySQL - MySQL ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, Me. The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation .MySQL is a popular choice of database for use in web applications, and is central components of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, and Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases.

XAMPP - XAMPP is a small and light Apache/Mysql distribution containing the most common web development technologies in a single package. Its contents, small size, and portability make it the ideal tool for students developing and testing applications in PHP and MySQL. XAMPP is available as a free download in two specific packages: full and lite. While the full package download provides a wide array of development tools, XAMPP Lite contains the necessary technologies that meet the Ontario Skills Competition standards. The light version is a small package containing Apache HTTP Server, PHP, MySQL, phpMyAdmin, and SQLite.

Obtaining and Installing XAMPP - For various web development tasks. As previously mentioned, XAMPP is a free XAMPP packages and add-ons are package available for download and use distributed through the Apache Friends website at the address: http://www.apachefriends.org/. Once on the website, navigate and find the Windows version of XAMPP and download the self-extracting ZIP archive. After downloading the archive, run and extract its contents into the root path of a hard disk or USB drive. For example, the extract path for a local Windows installation would simply be C:\. If

extracted properly we will notice a new xampp directory in the root of your installation disk. In order to test that everything has been installed correctly, first start the Apache HTTP Server by navigating to the xampp directory and clicking on the apache_start.bat batch file.

Chapter 4

SYSTEM DESIGN

4.1 Creating Database

Now that we have run and tested PhpMyAdmin, the next step is running MySQL and creating a database and table which will hold information to be used by our database. In order to start MySQL, navigate to the xampp directory and run the mysql_start.bat batch file. The XAMPP package contains an application called phpMyAdmin which allows developers to administer and maintain MySQL databases. We will be using phpMyAdmin to create a database and table, and enter test data. Before testing phpMyAdmin, makesure that both Apache and MySQL are running by opening their respective batch files: apache_start.bat and mysql_start.bat. Along with Apache and MySQL running in thebackground, we type http://localhost/phpmyadmin/index.php into our web browser.

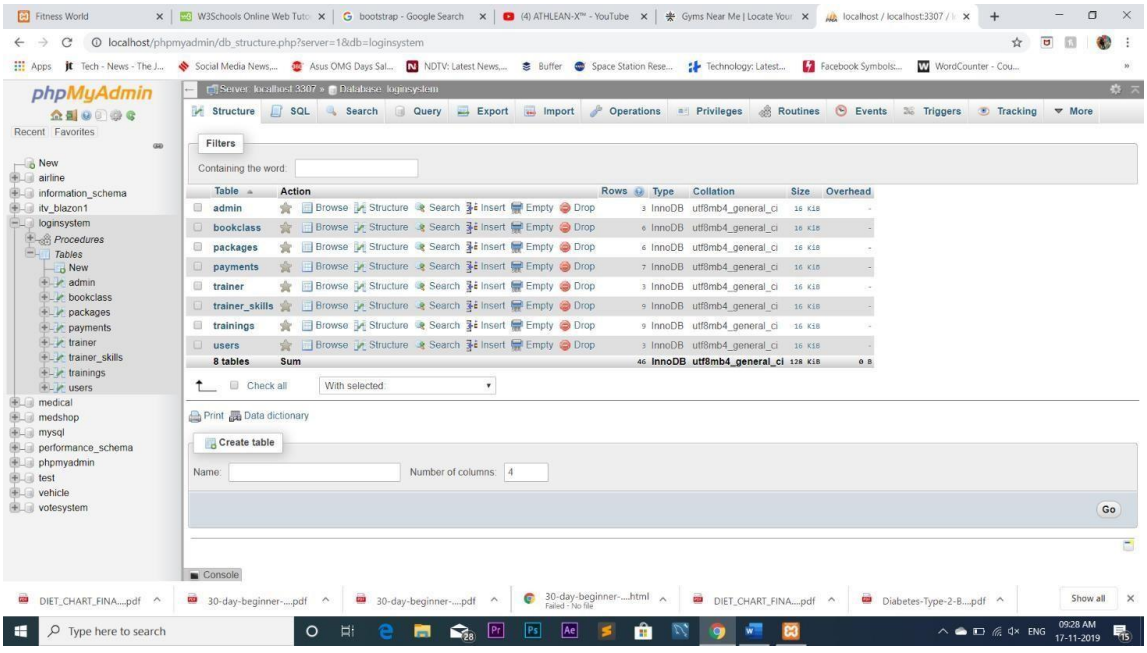


Fig.3.1 Project Database Description

The above picture shows how exactly the xampp phpMyAdmin page looks like. All the sql commands can be executed.

4.2: Entities, Attributes and their Domains

The following are the entities, attributes and their respective domains:

4.2.1 Admin

Attribute	Domain
Username	Varchar(30)
Password	Varchar(30)
Aid	Varchar(30)

Table: 4.2.1 shows the structure of table Admin which has 3 attributes.

4.2.2 Book Class

Attribute	Domain
Training_name	Varchar(30)
Trainer_name	Varchar(30)
Start_date	Date
Trainingid	Int(30)
Useruid	Varchar(30)

Table: 4.2.2 shows the structure of table bookclass which has 5 attributes.

4.2.3 Trainer

Attribute	Domain
Trainerid	Int(30)
Name	Varchar(40)
Email	Varchar(30)
Phone	Varchar(30)
Traineruid	Varchar(30)
Trainerpwd	Varchar(30)
Gender	Varchar(10)
Aid	Varchar(30)

Table: 4.2.3 shows the structure of table trainer which has 8 attributes.

4.2.4 Packages

Attribute	Domain
Amount	Int(30)
Packagename	Varchar(100)

Table: 4.2.4 shows the structure of table packages which has 2 attributes.

4.2.5 Payments

Attribute	Domain
Amount	Int(30)
Userid	Int(30)

Table: 4.2.5 shows the structure of table Payments which has 2 attributes.

4.2.6 Users

Attribute	Domain
Userid	Int(10)
User_first	Varchar(30)
User_last	Varchar(30)
User_email	Varchar(30)
User_uid	Varchar(30)
User_pwd	Varchar(30)
Phone	Varchar(30)
Gender	Varchar(10)

Table: 4.2.6 shows the structure of table Users which has 8 attributes.

4.2.7 Trainer Skills

Attribute	Domain
Trainerid	Int(30)
Training_name	Varchar(30)
Trainingid	Int(10)

Table: 4.2.7 shows the structure of table trainer skills which has 3 attributes.

4.2.8 Trainings

Attribute	Domain
Trainingid	Int(30)
Training_name	Varchar(30)
Duriation	Varchar(30)

Table: 4.2.8 shows the structure of table trainings which has 3 attributes.

4.3 Relationship Types

Relationship	Cardinality	Participation
SALES: AGENT, POLICY	N:M	Total participation on 'AGENT' side. Partial participation on 'POLICY' side.
INSURED BY: POLICY, POLICY HOLDER	N:M	Total participation on 'POLICY' side. Partial participation on 'POLICY HOLDER' side
CLAIMED BY: POLICY, CLAIMANT	1:1	Total participation on both the sides.
CHOOSES: POLICY HOLDER, CLAIMANT	1:1	Total participation on both the sides.

Table 4.6 Relationship Types

These are the various relationships, their cardinality and the participation. The relationships, cardinality ratios and participation are important factors while drawing an ER diagram.

4.4 ER Diagram

The ER or (Entity Relational Model) is a high-level conceptual data model diagram. Entity-Relation model is based on the notion of real-world entities and the relationship between them. ER modeling helps you to analyze data requirements systematically to produce a well-designed database.

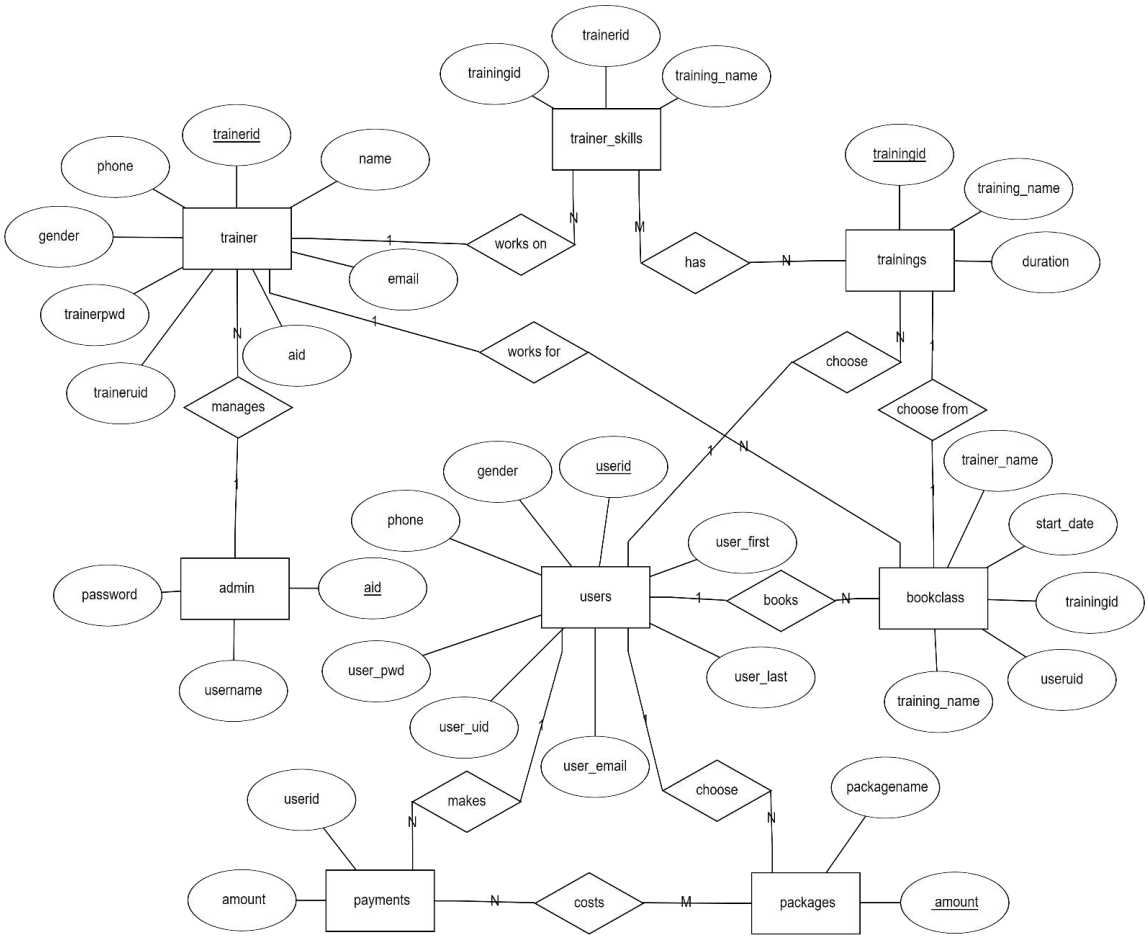


Fig.4.1 ER diagram

Now that we have the ER diagram ready, a relational schema can be easily obtained.

4.5 Relational Schema Diagram

Admin table

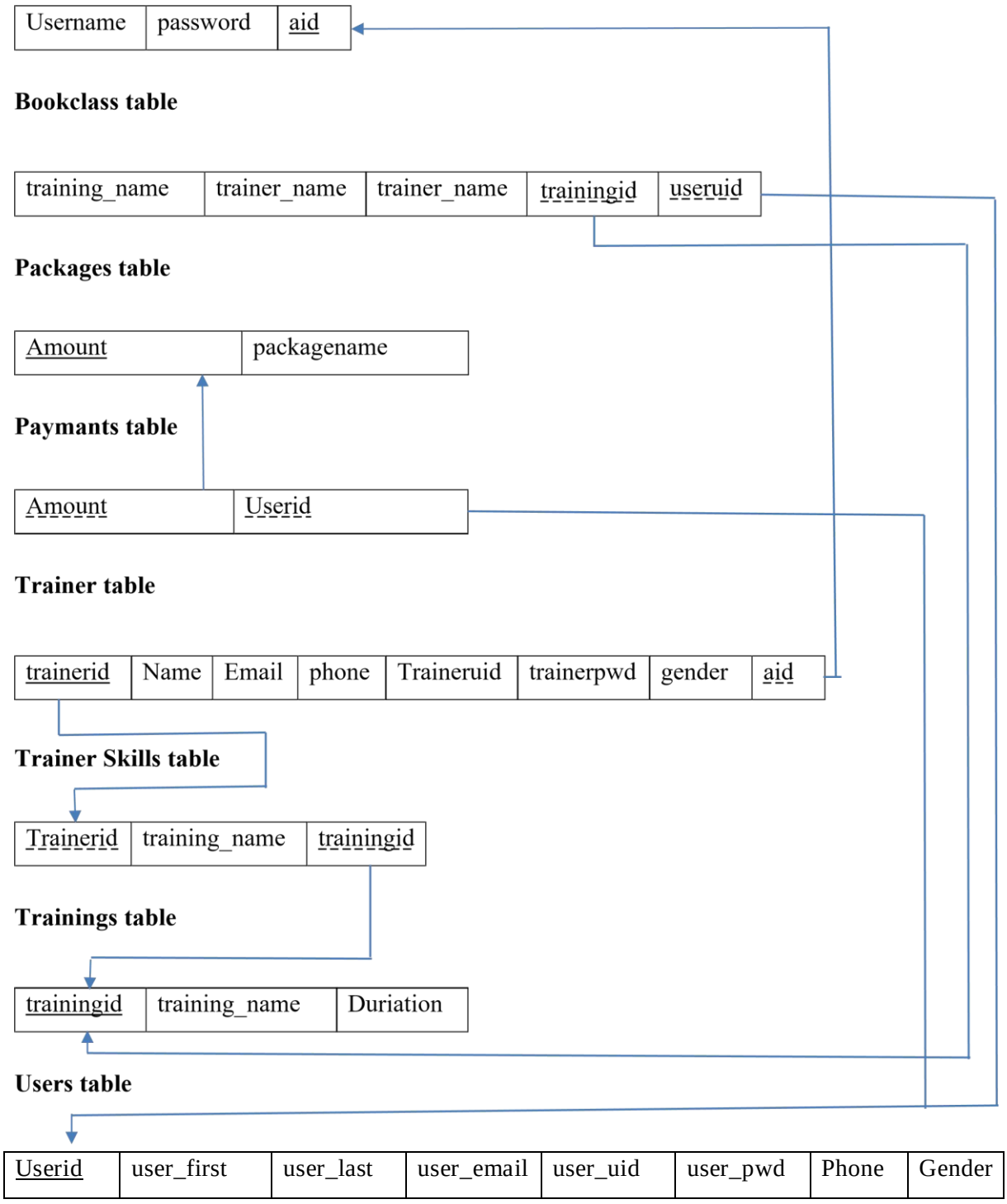


Fig 4.2: Schema Diagram

4.6 Normalization

DEF: A relation schema is in 1NF if all of its attributes are:

- Single valued
- Restricted to assuming atomic values

1st Norm Form:

Functionally dependent on the primary key Domain is atomic if its elements are considered to be indivisible units.

Examples of non-atomic domains Set of names, composite attributes

- Identification numbers like ph_key that can be broken up into parts
- A relational schema R is in first normal form if the domains of all attributes of Rare atomic.
- All domains in our database are atomic since they are indivisible.
- No Duplication of data, Insert Anomaly, Delete Anomaly, Update Anomaly found, therefore our database clears the first normal form test.

2nd Norm Form:

DEF: A relational table is said to be in second normal form 2NF if it is in 1NF and every non-key attribute is fully functionally dependent upon primary key.

The criteria for second normal form (2NF) are:

- The table must be in! NF.
- Every non-key attributes of the table must be dependent upon the entire primary key.
- Tables agent, policy holder, claimant, insurance, sales are also in 2NF.
- Our database satisfies all the conditions of 2NF since the tables are in 1NF and Every non-key attributes of the table must be dependent upon the entire primary key.

3rd Norm Form:

- A relation is in 3NF if and only if, it is in 2NF and there are no transitive functional dependencies.
- Transitive functional dependencies arise.
- When one non-key attribute is functionally dependent on another non-key attribute.
- Functional Dependency: non-key attribute>non-key attribute.
- When there is redundancy in the database.
- The tables insurance, claimant, policy holder and sales have been converted into 3NF.

Chapter 5

IMPLEMENTATION

5.1 Table Creation

5.1.1 Table 1: Admin

The Table creation code for admin is shown below:

```
CREATE TABLE `admin` (
  `username` varchar(30) NOT NULL,
  `password` varchar(30) NOT NULL,
  `aid` varchar(30) NOT NULL
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
2	password	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
3	aid	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More

Fig: 5.1.1 shows the structure of table Admin which has 3 attributes.

5.1.2 Table 2: Book Class

The Table creation code for bookclass is shown below:

```
CREATE TABLE `bookclass` (
  `training_name` varchar(30) NOT NULL,
  `trainer_name` varchar(30) NOT NULL,
  `start_date` date NOT NULL,
  `trainingid` int(30) NOT NULL, `useruid` varchar(30) DEFAULT NULL
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	training_name	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
2	trainer_name	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
3	start_date	date			No	None			Change Drop More
4	trainingid	int(30)			No	None			Change Drop More
5	useruid	varchar(30)	utf8mb4_general_ci		Yes	NULL			Change Drop More

Fig: 5.1.2 shows the structure of table bookclass which has 5 attributes.

5.1.3 Table 3: Packages

The Table creation code for packages is shown below:

```
CREATE TABLE `packages` (
  `amount` int(30) NOT NULL,
  `packagename` varchar(100) NOT NULL
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	amount	int(30)			No	None			Change Drop More
2	packagename	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More

Fig: 5.1.3 shows the structure of table packages which has 2 attributes.

5.1.4 Table 4: Payments

The Table creation code for payments is shown below:

```
CREATE TABLE `payments` (
  `amount` int(30) NOT NULL,
  `userid` int(30) NOT NULL
);
```

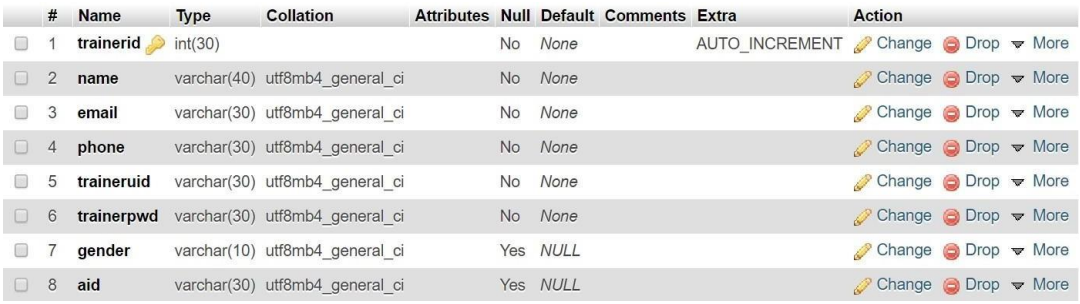
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	amount	int(30)			No	None			Change Drop More
2	userid	int(30)			No	None			Change Drop More

Fig: 5.1.4 shows the structure of table Payments which has 2 attributes.

5.1.5 Table 5: Trainer

The Table creation code for trainer is shown below:

```
CREATE TABLE `trainer` (
  `trainerid` int(30) NOT NULL,
  `name` varchar(40) NOT NULL,
  `email` varchar(30) NOT NULL,
  `phone` varchar(30) NOT NULL,
  `traineruid` varchar(30) NOT NULL,
  `trainerpwd` varchar(30) NOT NULL,
  `gender` varchar(10) DEFAULT NULL, `aid` varchar(30) DEFAULT NULL);
```



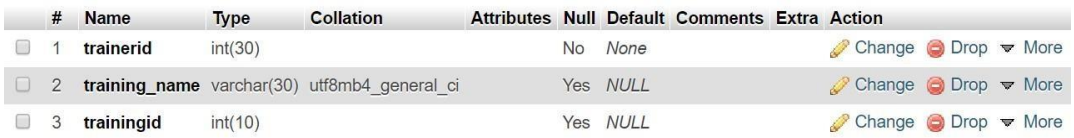
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	trainerid	int(30)			No	None		AUTO_INCREMENT	Change Drop More
2	name	varchar(40)	utf8mb4_general_ci		No	None			Change Drop More
3	email	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
4	phone	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
5	traineruid	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
6	trainerpwd	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More
7	gender	varchar(10)	utf8mb4_general_ci		Yes	NULL			Change Drop More
8	aid	varchar(30)	utf8mb4_general_ci		Yes	NULL			Change Drop More

Fig: 5.1.5 shows the structure of table trainer which has 8 attributes.

5.1.6 Table 6: Trainer Skills

The Table creation code for trainer skills is shown below:

```
CREATE TABLE `trainer_skills` (
  `trainerid` int(30) NOT NULL,
  `training_name` varchar(30) DEFAULT NULL,
  `trainingid` int(10) DEFAULT NULL
);
```



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	trainerid	int(30)			No	None			Change Drop More
2	training_name	varchar(30)	utf8mb4_general_ci		Yes	NULL			Change Drop More
3	trainingid	int(10)			Yes	NULL			Change Drop More

Fig: 5.1.6 shows the structure of table trainer skills which has 3 attributes.

5.1.7 Table 7: Trainings

The Table creation code for trainings skills is shown below:

```
CREATE TABLE `trainings` (
  `trainingid` int(30) NOT NULL,
  `training_name` varchar(40) NOT NULL,
  `duration` varchar(30) NOT NULL
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	trainingid	int(30)			No	None		AUTO_INCREMENT	Change Drop More
2	training_name	varchar(40)	utf8mb4_general_ci		No	None			Change Drop More
3	duration	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More

Fig: 5.1.7 shows the structure of table trainings which has 3 attributes.

5.1.8 Table 8: Users

The Table creation code for users is shown below:

```
CREATE TABLE `users` (
  `userid` int(10) NOT NULL,
  `user_first` varchar(30) NOT NULL,
  `user_last` varchar(30) NOT NULL,
  `user_email` varchar(30) NOT NULL,
  `user_uid` varchar(30) NOT NULL,
  `user_pwd` varchar(30) NOT NULL,
  `phone` varchar(30) DEFAULT NULL,
  `gender` varchar(10) DEFAULT NULL
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**userid**	int(10)			No	*None*		AUTO_INCREMENT	Change Drop More
2	**user_first**	varchar(30)	utf8mb4_general_ci		No	*None*			Change Drop More
3	**user_last**	varchar(30)	utf8mb4_general_ci		No	*None*			Change Drop More
4	**user_email**	varchar(30)	utf8mb4_general_ci		No	*None*			Change Drop More
5	**user_uid**	varchar(30)	utf8mb4_general_ci		No	*None*			Change Drop More
6	**user_pwd**	varchar(30)	utf8mb4_general_ci		No	*None*			Change Drop More
7	**phone**	varchar(30)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
8	**gender**	varchar(10)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More

Fig: 5.1.8 shows the structure of table Users which has 8 attributes.

5.2 Stored procedures

A stored procedure is a prepared SQL code that you can save, so the code can be reused over and over again.

So if you have an SQL query that you write over and over again, save it as a stored procedure, and then just call it to execute it. You can also pass parameters to a stored procedure, so that the stored procedure can act based on the parameter value(s) that is passed.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost:3306` PROCEDURE `usertable`()
   NO SQL
SELECT * from users$$
DELIMITER ;
```

The above-mentioned stored procedure code is used in this project to display the list of users.

5.3 Triggers

A trigger is a set of actions that are run automatically when a specified change operation (SQL INSERT, UPDATE, or DELETE statement) is performed on a specified table. Triggers are useful for tasks such as enforcing business rules, validating input data, and keeping an audit trail.

```
CREATE TRIGGER `usertaken` BEFORE INSERT ON `users`
FOR EACH ROW IF 1= (SELECT COUNT(*) from users where user_uid =
new.user_uid) THEN SIGNAL SQLSTATE '45000' ;
END IF
```

The above-mentioned trigger is used in this project to stop the insertion of data if the user name is taken by some other users.

Chapter 6

TESTING

The aim of the system testing process was to determine all defects in our project. The program was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not.

Functionality	**Action**	**Expected Result**	**Actual result**	**Test Result**
Accepting user input for login.	Login is clicked	Should enter to User home page	Entered into User Home page	PASS
If invalid input for password and username.	Display invalid username/password	Should refer to the user login page again	Refreshed user login page	PASS
Creating a new User, Signup button is clicked. And fill up the form and click the signup button.	Signup button is clicked	Should enter to the user home page	Entered into the user home page	PASS
To view the logged in user profile, click on the user image icon.	Profile icon is clicked	To display the current users' details.	Displayed the user details.	PASS

To view the fitness packs, click the image on the user home page.	Click the image	Enters into the pack details page	Entered into the pack details page	PASS
To Buy the pay click on the buy pack button.	Click on the buy pack button	Enters into the payment page	Entered into the payment page	PASS
To Buy the pack click on Proceed to pay button.	Click on proceed to pay button	Displays payment successful	Displayed payment successful	PASS
To see the trainings, click on the training session in the user home page.	Go to the training session	Entered into the trainings page	Entered into the trainings page	PASS
To join into the training click on the training which you need.	Click on the training session	Enters into the Training details page	Entered into the training details page	PASS

To book the class just click on the Book class button	Click on the book class button	Enters to the Book class page where you need to choose the trainer and the start date	Entered into the Book class page	PASS

Choose the trainer and choose the date which you want to take the class for.	Click on the Book class button	Displays class booked. If already booked than displays already taken	Booked the class/ Already taken	PASS
To see the timetable, click on the timetable session in the user home page.	Go to the time table session	Enter into the timetable page	Entered into the timetable page	PASS
To see the Nutrition, click on the Nutrition session in the user home page.	Go to the Nutrition session	Enter into the Nutrition page	Entered into the Nutrition page	PASS

Click on the links on the nutrition page, The file gets downloaded with pdf format.	Click on the link	Download the nutrition/died pdf	Downloaded the pdf	PASS
To see the Fitness Videos, click on the	Go to the Videos session	Enter into the Videos page	Entered into the Videos page	PASS
Videos session in the user home page.				
To watch the Video, click on the video.	Click on the video	Video should play playing	Video Started playing	PASS
To see the Contact Us, click on the Contact Us session in the user home page.	Go to the Contact Us session	Enter into the Contact Us page	Entered into the Contact Us page	PASS

To see the Packs Taken and classed booked, Enter into the user profile page And click on the Skills session.	Click on the skills session	Shows the details of the packs and classes booked	Pack and class details view	PASS
Logout	Logout is clicked	Should successfully logout and should go to the home page.	Successfully redirect to the Home page	PASS

Accepting Trainer input for Trainer login.	Login is clicked	Should enter to Trainer home page	Entered into Trainer Home page	PASS
Creating a new Trainer, Signup button is clicked. And fill up the form and click the signup button.	Signup button is clicked	Should redirect to the Trainer login page.	Successfully redirected to the trainer login page	PASS

Accepting Trainer input for Trainer login.	Login is clicked	Should enter to Trainer home page	Entered into Trainer Home page	PASS
To view the logged in Trainer profile, click on the Trainer image icon.	Profile icon is clicked	To display the current Trainer details.	Displayed the trainer details.	PASS
The trainer should choose the skills.	Click on the skills he is trained in	To add the skill	Skill added successfully	PASS
To see the Skills of the trainer, Enter into the Trainer profile	Click on the skills session	Shows the skills he has chosen.	Skills details view.	PASS
page, and click on the Skills session.				
Logout	Logout is clicked	Should successfully logout and should go to the home page.	Successfully redirect to the Home page	PASS

Accepting Admin input for Admin login.	Login is clicked	Should enter to Admin home page	Entered into Admin Home page	PASS
To see all Users, click on the user Session in the admin home page.	Click on the users	Should display all users list	Displayed the users list	PASS
To see the individual user information, type the userid in the search bar, and click on the search button.	Click on the search button	Should display the user details	Displayed the user Information	PASS
To see all Trainer who choose the	Click on the Trainers	Should display all Trainers list	Displayed the Trainers list	PASS

current admin as his Master, click on the Trainer Session in the admin home page.				
To see the Single Trainer information, type the Trainerid in the search bar, and click on the search button.	Click on the search button	Should display the Trainer details	Displayed the Trainer Information	PASS
Logout	Logout is clicked	Should successfully logout and should go to the home page.	Successfully redirect to the Home page	PASS

Table 6.1: Functionality and Actions Table

Chapter 7

SNAP SHOTS

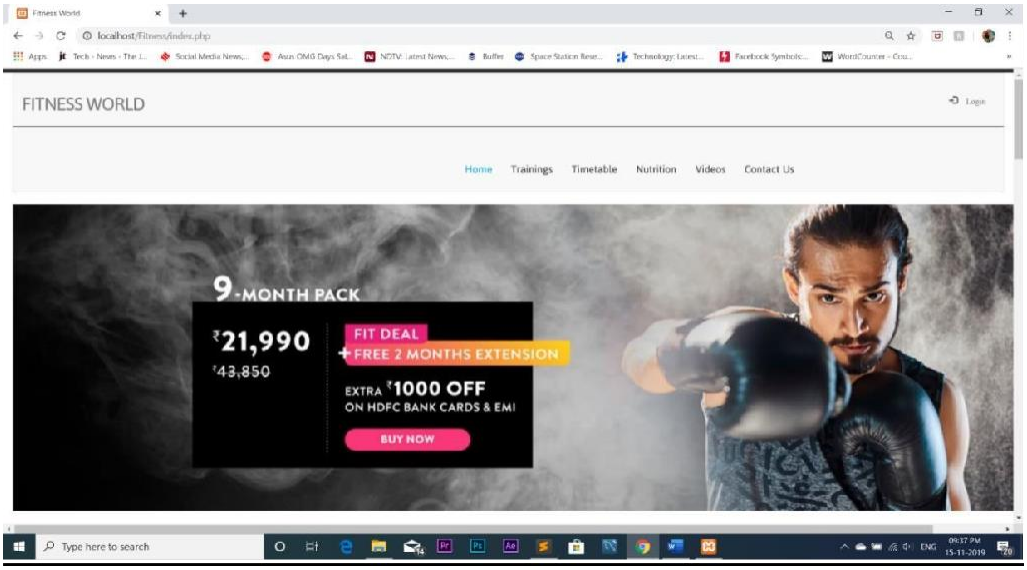


Fig 7.1 Home Page

Fig 7.1 Shows the Home Page which is the starting page of the website.

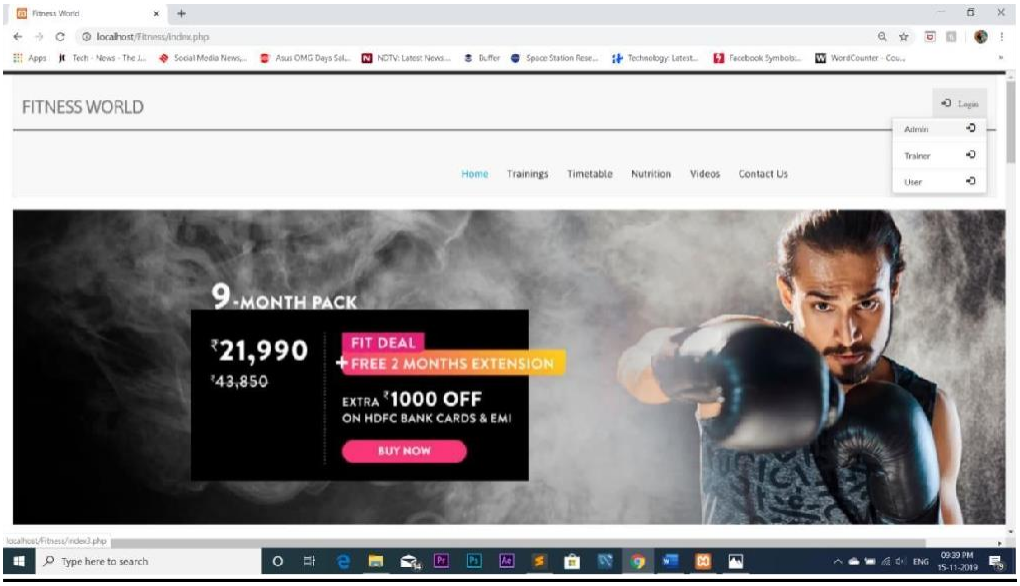


Fig 7.2 Admin Login

Fig 7.2 Shows the Admin Login option to choose.

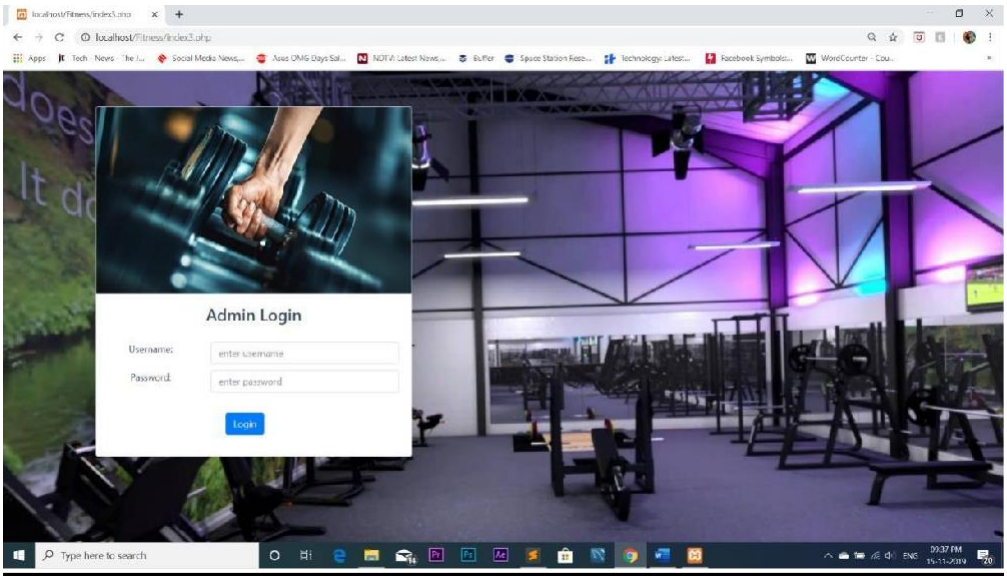


Fig 7.3 Admin Login Page

Fig 7.3 It is Admin Login Page is used to login to the admin account.

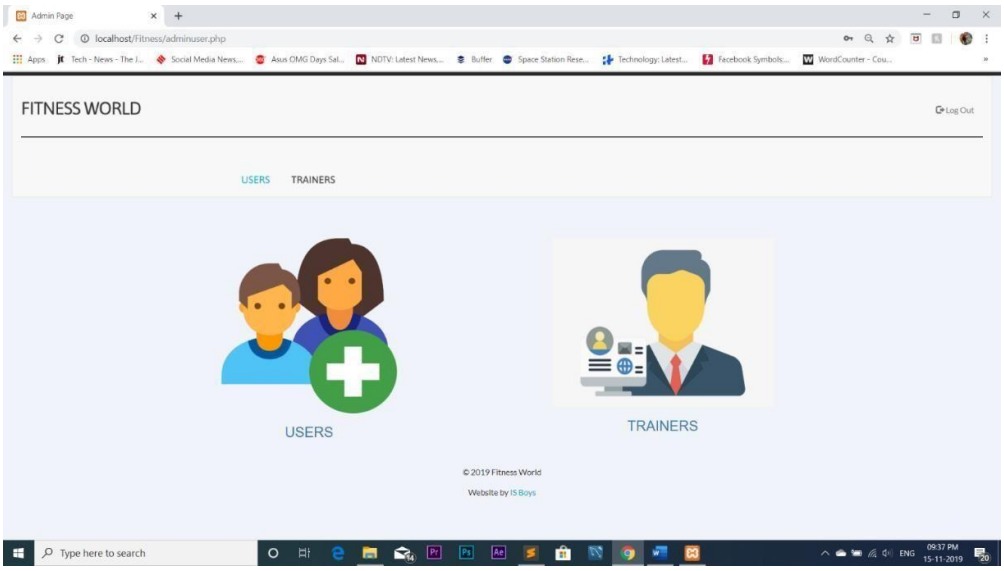


Fig 7.4 Admin Home Page

Fig 7.4 It is Admin Home page is used by the admin to know the details of user and trainer.

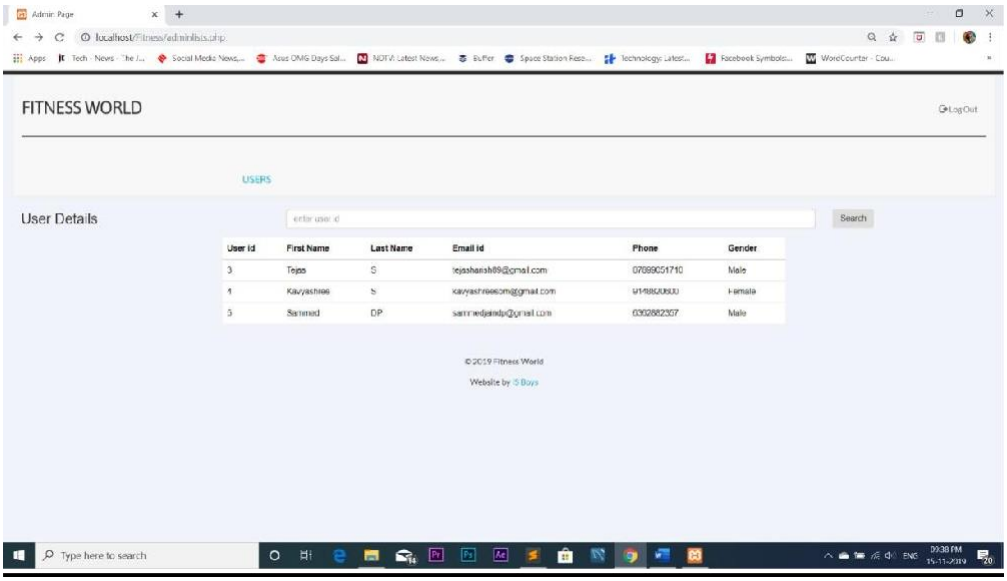


Fig 7.5 User Search Page

Fig 7.5 It is User Search page is used see all the users and search individual users details.

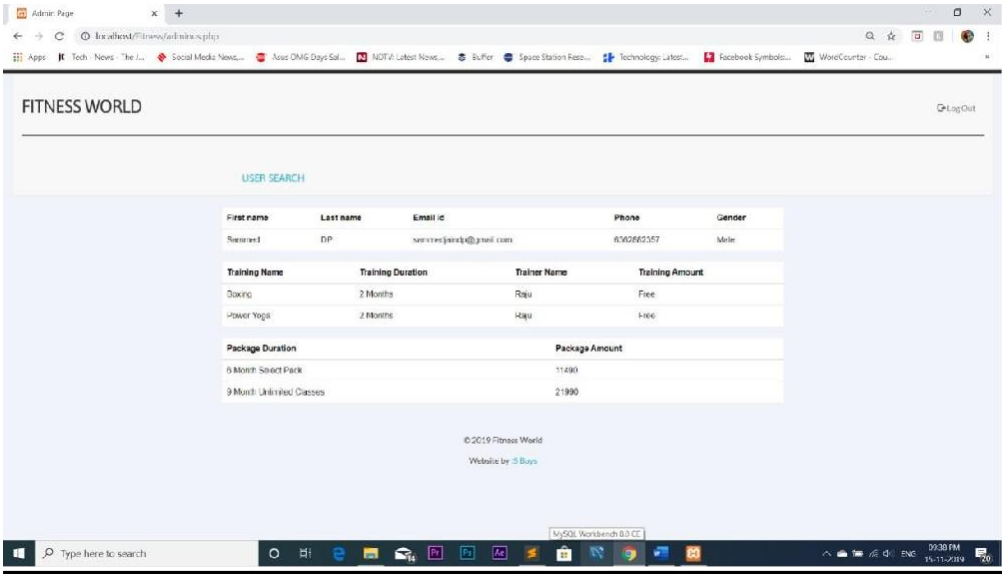


Fig 7.6 User Details Page

Fig 7.6 It is User Details Page where admin can see the user full details.

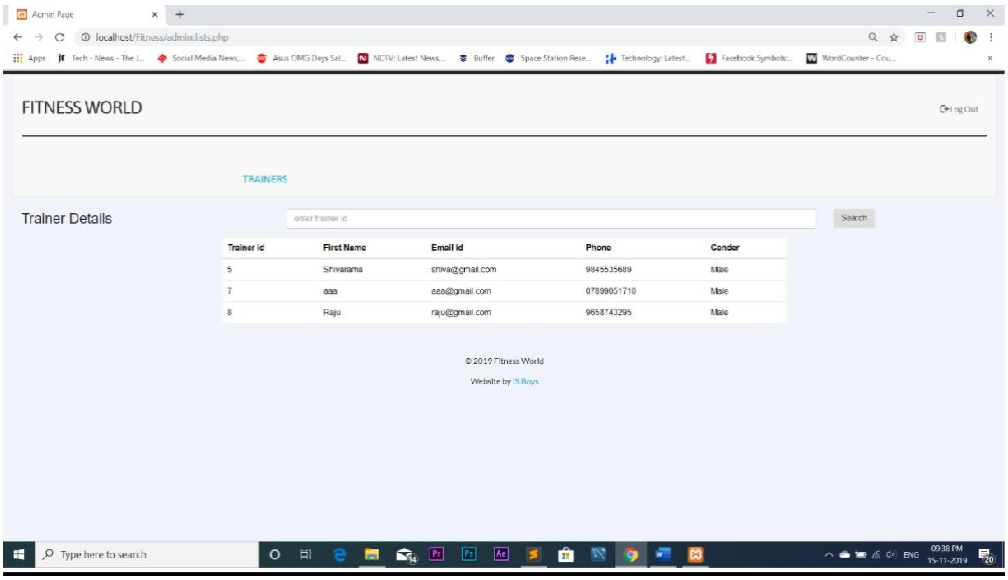


Fig 7.7 Trainer Search Page

Fig 7.7 It is Trainer Search Page where admin can see the trainer who has chosen particular admin.

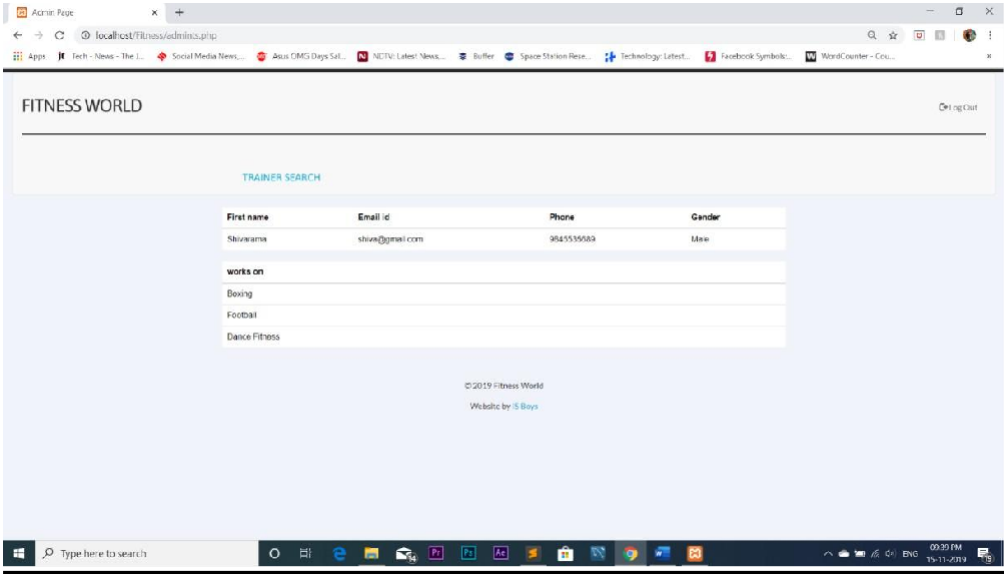


Fig 7.8 Trainer Details Page

Fig 7.8 It is Trainer Details Page where admin can see the Trainer full details.

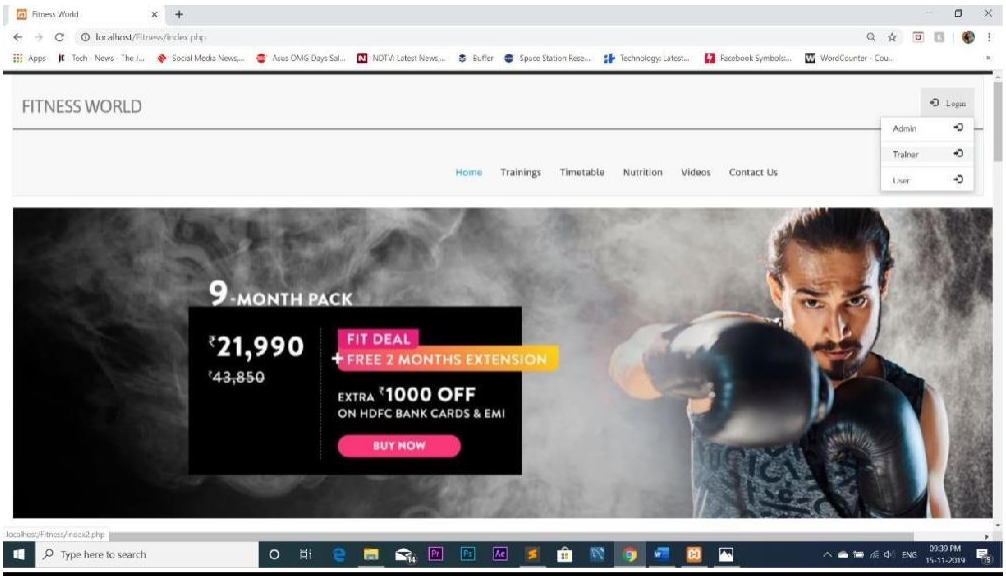


Fig 7.9 Trainer Login

Fig 7.9 Shows the Trainer Login option to choose.

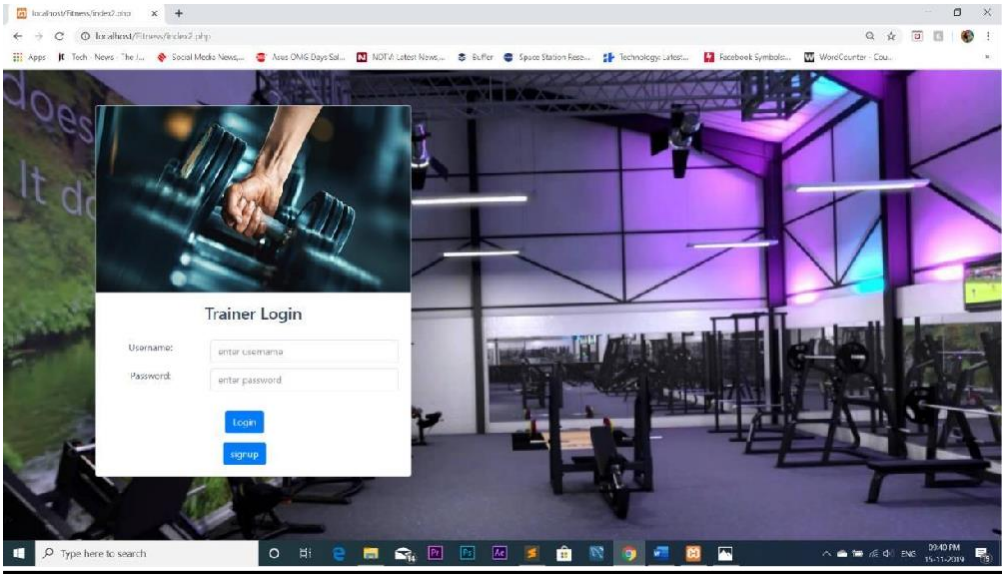


Fig 7.10 Trainer Login Page

Fig 7.10 It is Trainer Login Page is used to login to the Trainer account.

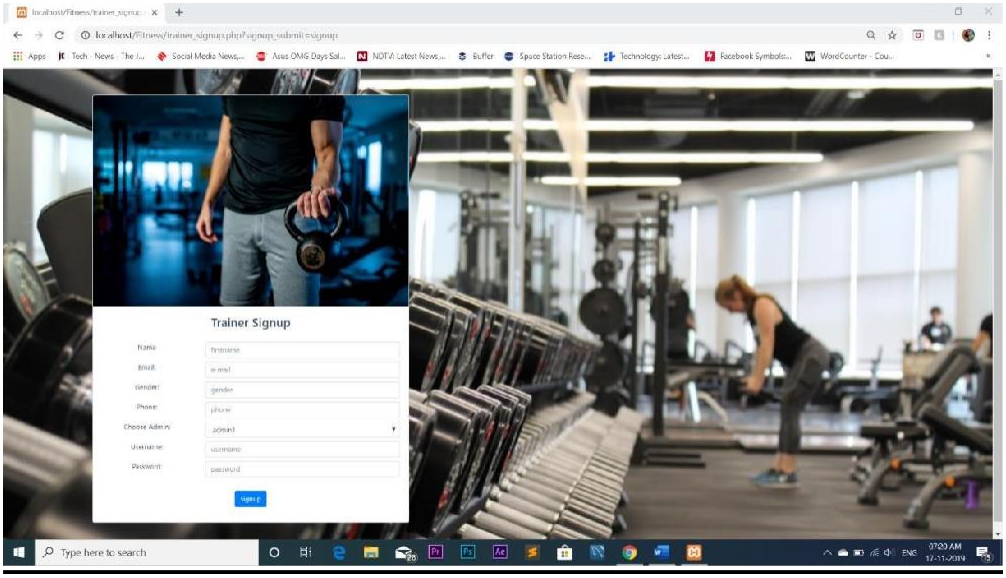


Fig 7.11 Trainer Signup Page

Fig 7.11 It is Trainer Signup Page is used to add new trainer.

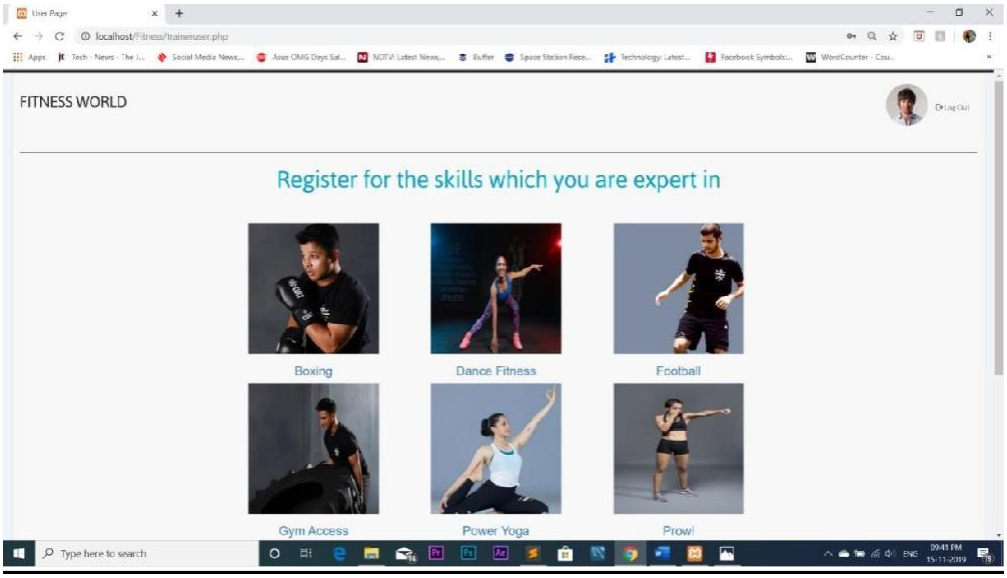


Fig 7.12 Trainer Home Page

Fig 7.12 It is Trainer Home Page is used by the Trainer to add the skills which he expertin.

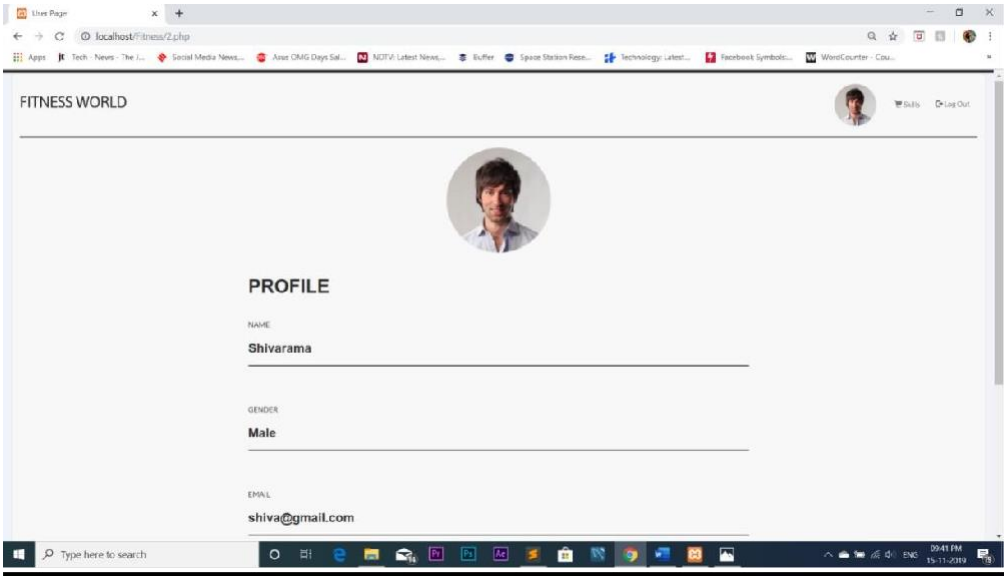


Fig 7.13 Trainer Profile Page

Fig 7.13 It is Trainer Profile Page is used to view the profile details of the trainer.

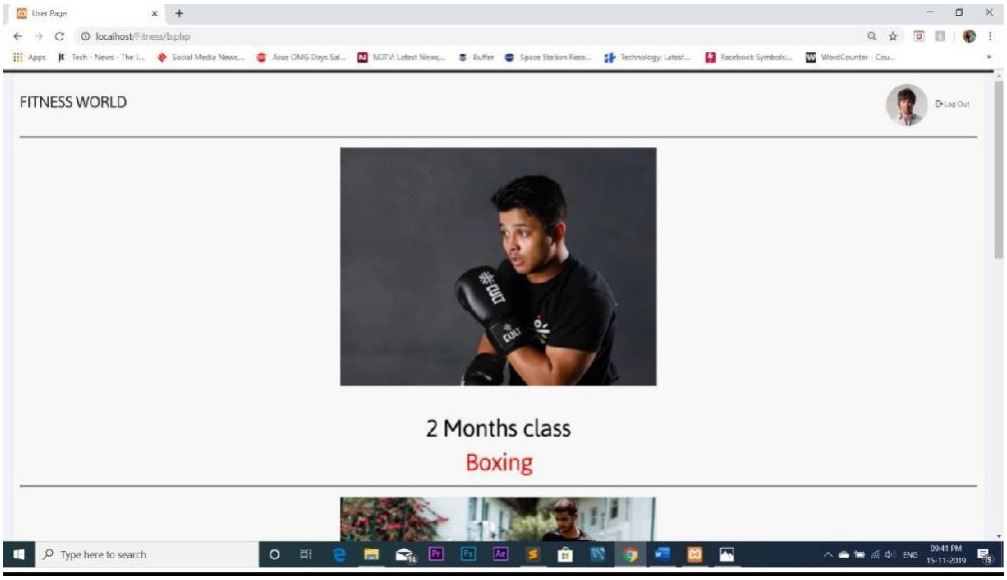


Fig 7.14 Trainer Skills Page

Fig 7.14 It is Trainer Skills Page is used to view the skills added by the trainer.

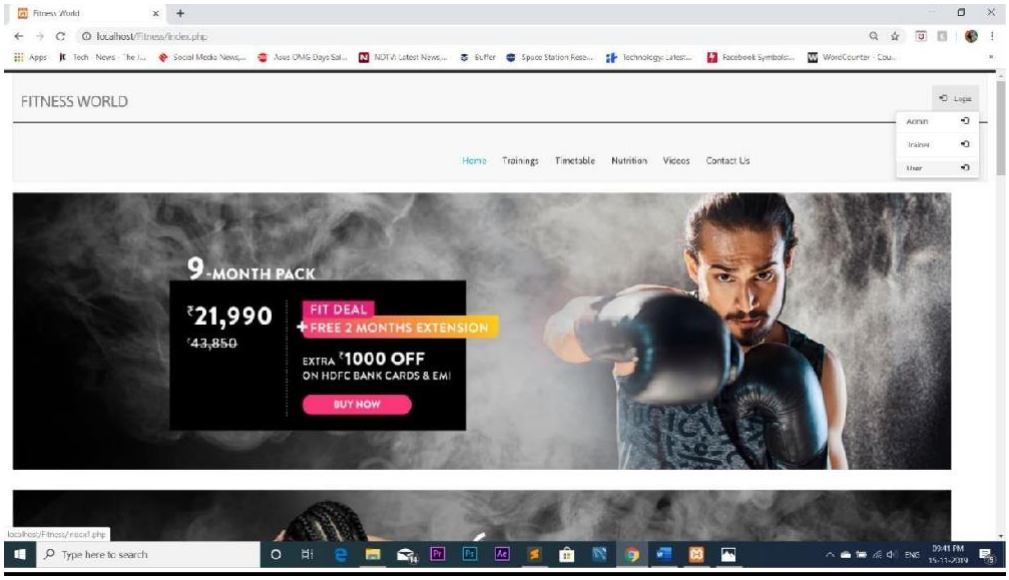


Fig 7.15 User Login

Fig 7.15 Shows the User Login option to choose.

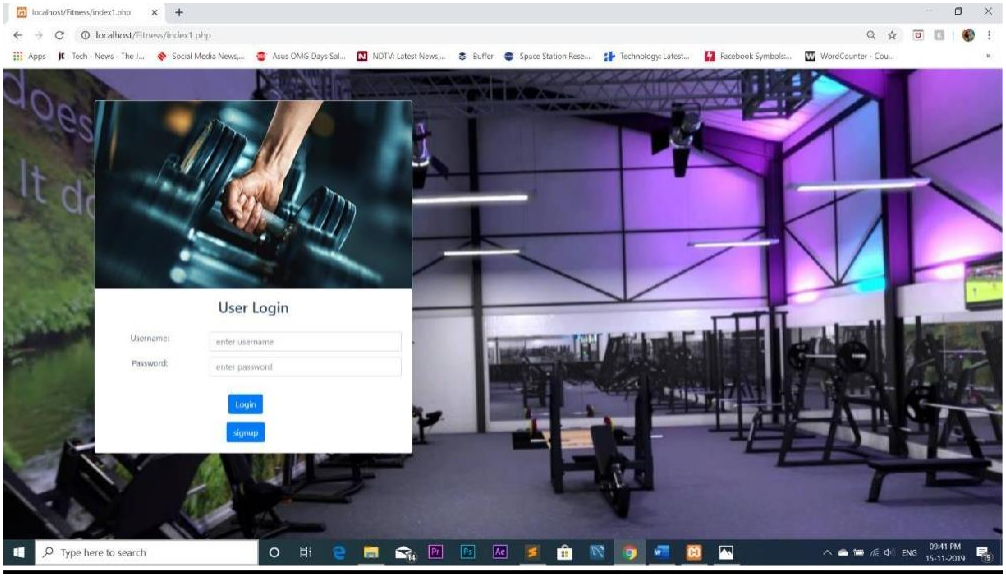


Fig 7.16 User Login Page

Fig 7.16 It is User Login Page is used to login to the User account.

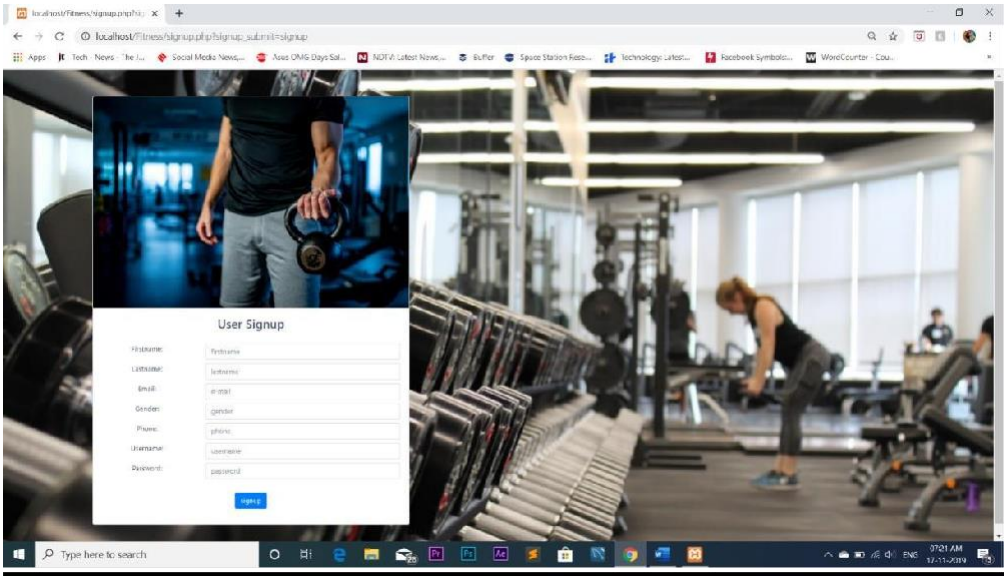


Fig 7.17 User Signup Page

Fig 7.17 It is User Signup Page is used to add new user.

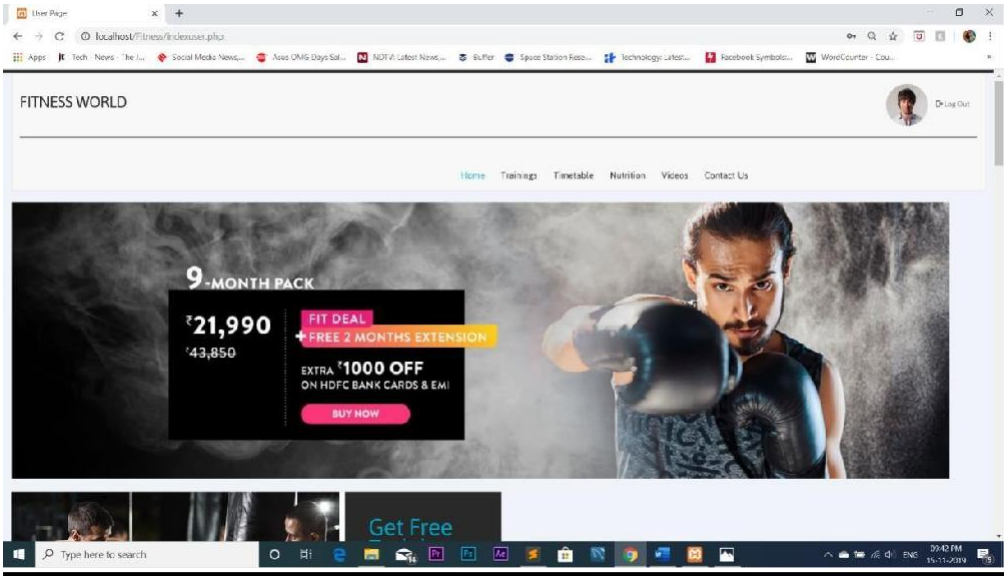


Fig 7.18 User Home Page

Fig 7.18 It is User Home Page is used by the User to choose the trainings and explore new packs.

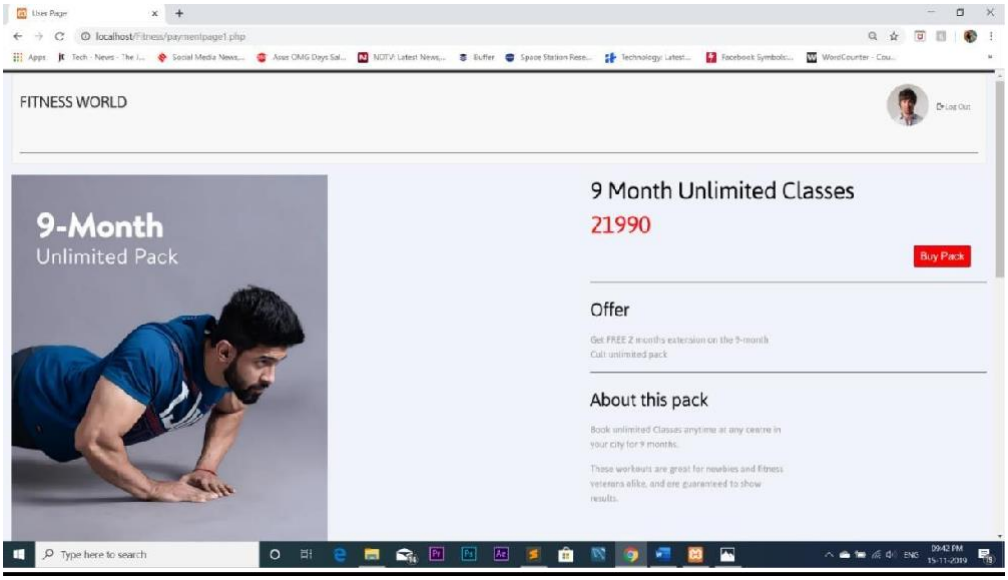


Fig 7.19 Booking Packs Page

Fig 7.19 It is Booking Packs Page is used by the User to book the pack.

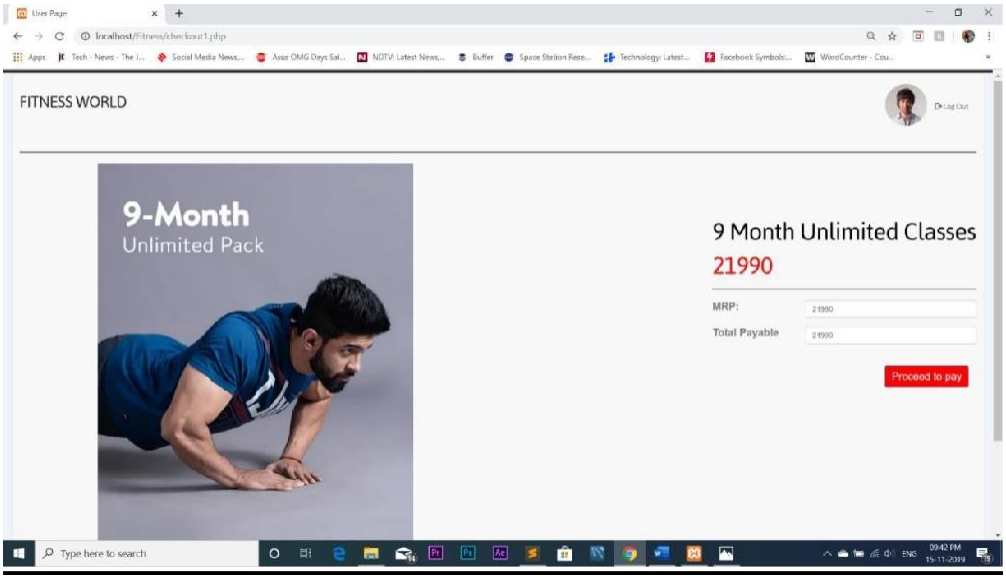


Fig 7.20 Pack Payment Page

Fig 7.20 It is Packs Payment Page is used by the User to make the payment.

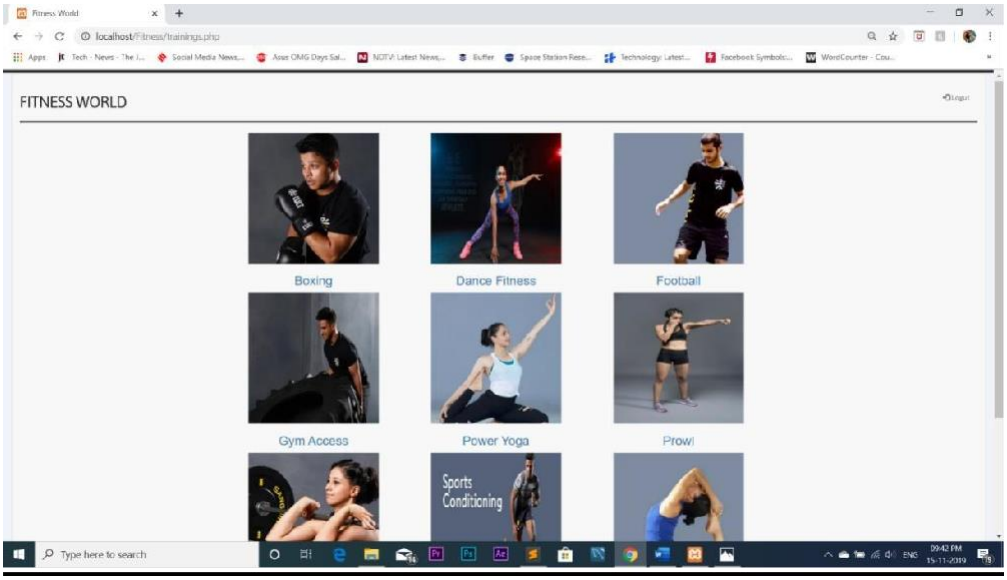


Fig 7.21 Trainings Classes

Fig 7.21 It is Trainings Classes is used to book the training classes.

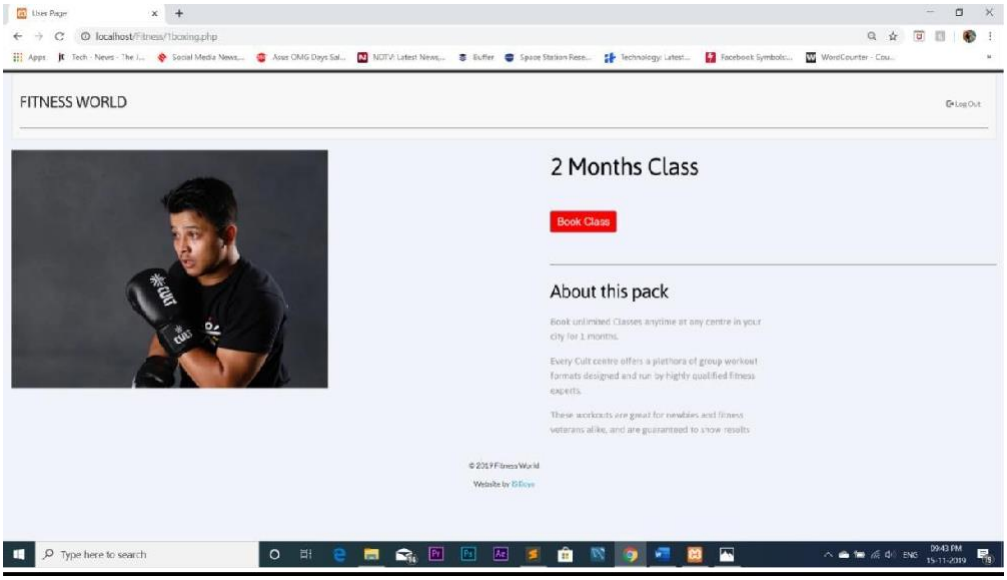


Fig 7.22 Book Training Classes

Fig 7.22 It is Book Training Classes is used to book the trainings classes.

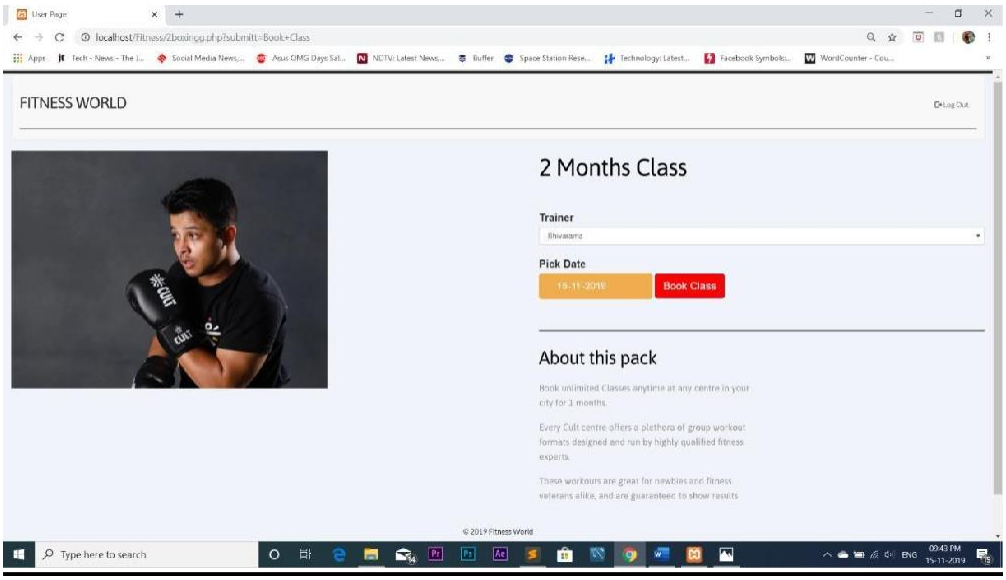


Fig 7.23 Schedule Training Classes and Book Page

Fig 7.23 It is Schedule Training Classes and Book Page where users books the classes by choosing the trainer and the start date.

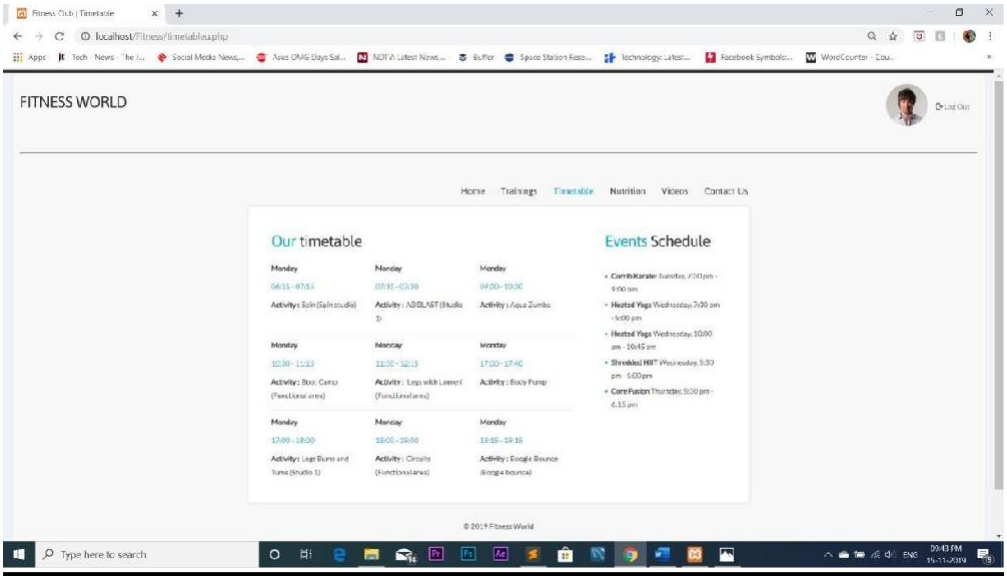


Fig 7.24 Time Table Page

Fig 7.24 It is Time Table Page is used to view the time table of the day.

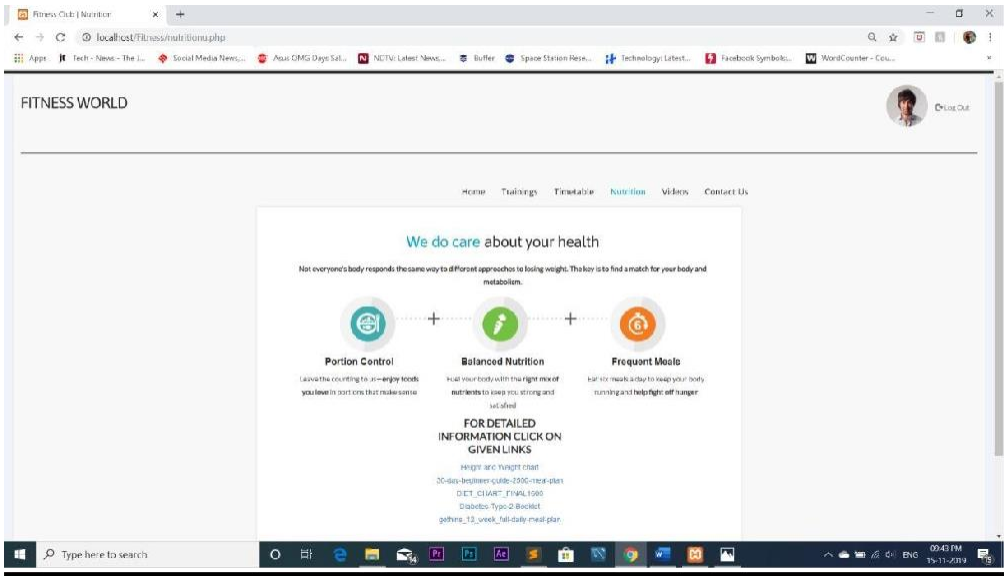


Fig 7.25 Nutrition Content Page

Fig 7.25 It is Nutrition Content Page used to show the diet plan and more about diet.

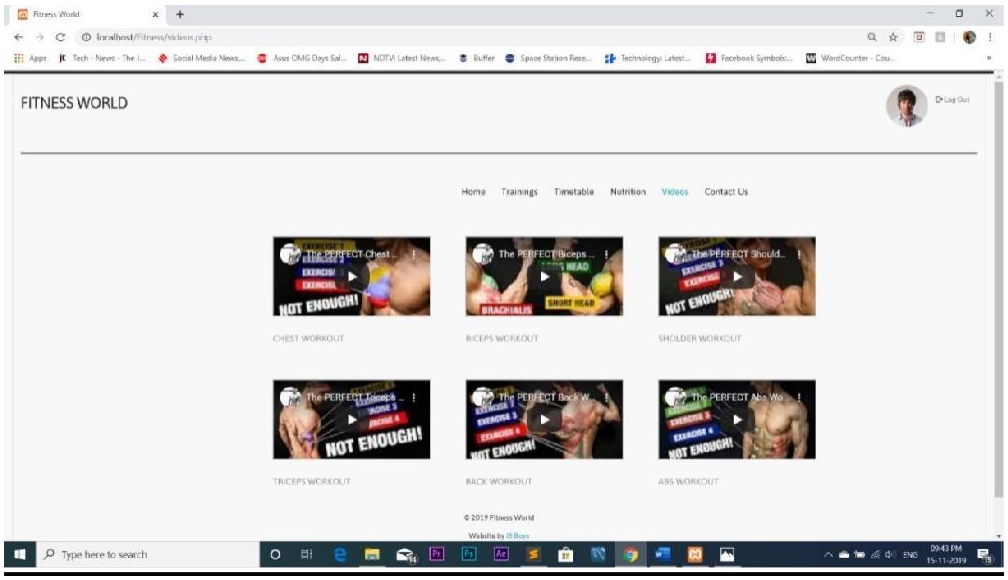


Fig 7.26 Workout Videos Page

Fig 7.26 It is Workout Video Page used to see the videos and learn fitness trainings.

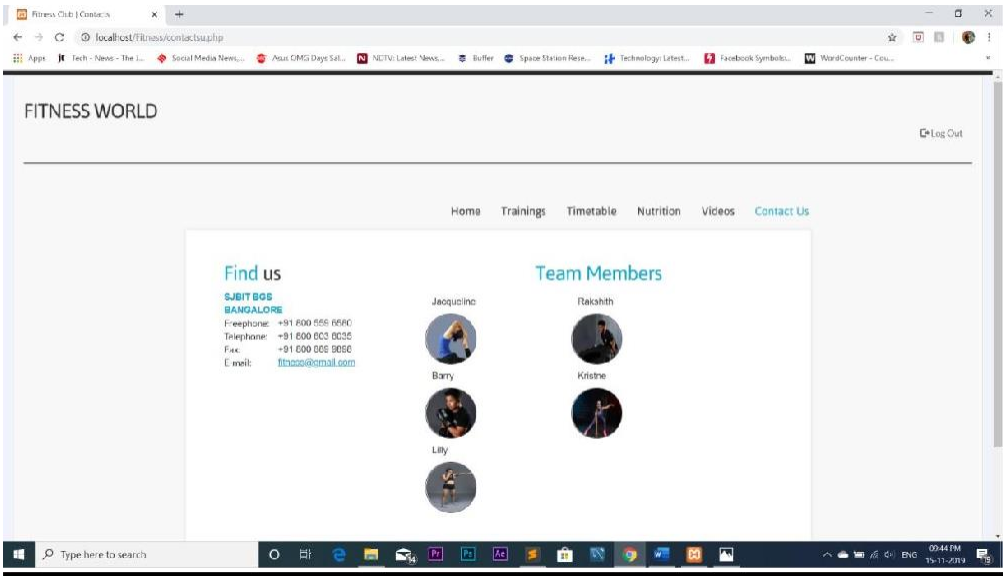


Fig 7.27 Contact Us Page

Fig 7.27 It is Contact Us Page used to contact the fitness trainers for any query.

CONCLUSION AND FUTURE ENHANCEMENT

Conclusion

The project titled as "Fitness Management System" is a web-based application. This software provides facility of viewing the fitness packs details, Booking the training classes, exploring new classes of trainings, Choosing the appropriate diet plans, for the Users. This software is developed with scalability in mind. Additional modules can be easily added when necessary. The software is developed with modular approach. All modules in the system have been tested with valid data and invalid data and everything work successfully.
Thus, the system has fulfilled all the objectives identified.

The project has been completed successfully with the maximum satisfaction of the organization. The constraints are met and overcome successfully. The system is designed as, like it was decided in the design phase. The project gives good idea on developing a full-fledged application satisfying the requirements. Hence the software has proved efficiently.

Future Enhancement

- **Space Constraints:** It may not be possible to record all the desired information at the same place e.g. large number of registers will be needed. As the information grows number of registers having data also increases.

- **Memory Lapse:** To easily access the record within the registers some alternate method is also needed, which will keep track of each register containing information.

- **Maintaining copies:** It is not easy to maintain the multiple copies of the database, because all are maintained manually.

- **Indexing:** It is difficult to index the information stored in the manual helpline database.
- **Searching:** To find any album from manual helpline database is not so easy and is time consuming. As the number of records increase the average time required for the search also increases.
- **Report generation:** People need reports, like list of students, list of sub-categories etc. It will take a lot effort as well as time to perform the job manually.

BIBLIOGRAPHY

[1] RamezElmasri and Shamakant B. Navathe, "Database system Models, Languages, Design and Application Programming", 6th Edition, Pearson, 2017.

[2] Ramakrishna and Gehrke, "Database management system", 3rd Edition McGraw Hill 2014.

[3] Silberschatzzkorth and Sudhershan, "Database system concepts", 6th Edition McGraw Hill, 2013.

[4] Coronel, Morris,and Rob "Database Principles Fundamentals of Design, Implementation and Management" Cengage Learning 2012.

[5] Abraham Silberchatz, Henry korth and S.Sudarshan, "Database System Concepts", McGraw Hill Education, 2010.

[6] PHP tutorials, retrieved from https://www.youtube.com.

[7] Front-end retrieved from https://www.w3schools.com.

[8] PHP connection, retrieved from https://www.phptpoint.com.

[9] E-R Diagram construction, retrieved from https://www.erdplus.com.