VISVESVARAYA TECHNOLOGICAL UNIVERSITY
“JNANA SANGAMA”, BELAGAVI-590018

A MINI PROJECT REPORT

ON

GYM MEMBERSHIP MANAGEMENT

Submitted in partial fulfillment of the
requirements for the award of degree of
Bachelor of Engineering
In
Computer Science and Engineering
By DARSHAN SK
[1KS15CS024]
Under the guidance of

Department of Computer Science & Engineering

K.S.INSTITUTE OF TECHNOLOGY

#14, Raghuvanahalli, Kanakpura Main Road, Bengaluru-560109
2017-2018

K.S.INSTITUTE OF TECHNOLOGY

#14, Raghuvanahalli, Kanakpura Main Road, Bengaluru-560109

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that mini project work entitled "GYM MEMBERSHIP MANAGEMNET" carried out by Mr. Darshan Sk bearing USN 1KS15CS024 bonafide student of K.S.Institute of Technology in the partial fulfillment for the award of the Bachelor of Engineering in Computer Science & Engineering of the Visvesvaraya Technological University, Belagavi, during the year 2017. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of mini Project work prescribed for the said degree.

Dr. Rekha. B. Venkatapur	Dr. T.V. Govindaraju
Prof & HOD ,CS & E Department	Principal/Director,KSIT

Mr. K Venkata Rao	Ms. Sougandhika Narayan	Mr. Kumar K
Assoc. Prof, Dept. Of CSE	Asst. Prof, Dept. Of CSE	Asst. Prof, Dept. Of CSE

Name of the Examiners with date	Signature
1.	
2.	

ACKOWLEDGEMENT

I take this opportunity to thank one and all involved in building this project. Firstly I would like to thank the college for providing us an opportunity to work on the project.

First and foremost, I thank the management of K. S. Institute of Technology for providing all the resources required for the project.

I wish to acknowledge my sincere gratitude to our beloved Principal, Dr. T.V. Govindaraju for his encouragement and providing all facilities for the accomplishment of this project.

This project would not have been possible without the support of our beloved Prof & HOD, Dr. Rekha.B.Venkatapur, Dept. of CSE.

I am also highly grateful to my project guide Mr. K Venkata Rao, Ms. Sougandhika Narayan and Mr. Kumar K, Dept. of CSE who have been very generous in assisting and supporting ,to do this Project "Real Estate Listing ", which formally started as just a rough idea and now resulted in this.

I also would like to thank all other teaching and non teaching staff members who had extended their support and co-operation while bringing up this project.

Darshan Sk

(1KS15CS024)

Abstract

The task of managing all the members of a fitness center is a tedious task. This Gym Data Base Management System makes it convenient for customers to find their batches and give feedback. It simplifies the owner-customer relationship. Users can login or register to view batches uploaded by the admin or upload feedback of their respective batches. All available batches can be viewed by the user upon logging in, users can choose from a list of available batches.

Batches are classified based on instructor, location and timing and corresponding details can be suitably added and displayed. The objective is to maintain details of every user and batches on the database for easy access. It allows simplified linkage of user and all the details of the fitness center.

A admin or the owner has to maintain a lot of data and review all the feedback. He is involved with the employees of batches at various locations and all the details of the users and their feedbacks. Hence it involves a lot of information exchange. Using a database, we can organize this information and the process is done faster.

INDEX

Page

1. Introduction

2. Requirement Specification

Introduction

Gym Database Management System is a platform that allows members to upload feedbacks and view all the batches available. There is a requirement to maintain all the records of members, batches and the feedback. To achieve this there is a need to use a system will can store and process all this data. Hence the main objective is to produce a web based system that provides members to register and keep track of batches.

1.1 Project

The Gym Database Management System is designed to provide a simple way for prospective members to view all available batches and gym owners to upload details of fitness center that are available. The objective is to eliminate the need for a maintaining all records or data or any other intermediate step. Instead, owner-customer relationships can be made transparent by direct interaction. It also reduces manual work, which is prone to errors.

This system aims to change the conventional way of looking for all available facilities of the fitness center and makes the task easier for the member. They can view details of batches, location or timings offered by different instrutors . Thus, the effort in looking for suitable facilities of the fitness center is reduced.

Gym owners can draw attention to any of their batches without the hassle of going through informing each and every member og the fitness center. They do not have to find a means to advertise their facilities to potential members. It is a simple and elegant way to access all relevant details. This system is aimed at computerizing and automating the working of gym database management. With the help of a database we can store and organize a large amount of data.

1.2 DBMS

A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data.

A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

A DBMS provides concurrency, security, data integrity, consistency, controls redundancy and data independence.

In this project the Relational DBMS (RDBMS) used is MySQL. It is an open source software which uses SQL (Structured Query Language) which is a standard language for storing, manipulating and retrieving data in databases.

1.3 Java connections

To connect the database with the front end we use a java connector JDBC (Java Database Connectivity). JDBC is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It is Java based data access technology and used for Java database connectivity. It is part of the Java Standard Edition platform, from Oracle Corporation.

To achieve connectivity we use JSPs (JavaServer Pages) and Servlets in this project. JavaServer Pages (JSP) is a technology that helps software developers create dynamically generated web pages based on HTML, XML, or other document types. JSP is similar to PHP and ASP, but it uses the Java programming language.

Requirements Specification

A computerized way of handling information about members and batch details is efficient, organized and time saving, compared to a manual way of doing so. This is done through a database driven web application whose requirements are mentioned in this section.

2.1 Overall Description

A reliable and scalable database driven web application with security features, that is easy to use and maintain is the requisite.

2.2 Specific Requirements

The specific requirements of the Gym Database Management System are stated as follows:

2.2.1 Software Requirements

Technology used:

- Front end – JSP
- Controller – JSP/Servlets
- Backend – mySQL

Software:

- IDE - Netbeans 8.2
- Database support - MySQL 5.7
- Operating system – Windows 8 and above
- Server deployment - Glassfish server

Technology:

- HTML is integrated in JSP. It provides a means to structure text based information in a document. It allows users to produce web pages that include text, graphics and hyperlinks.

- Javascript is a scripting language which supports the development of both client and server applications. It is preferred at client side to write programs that can be executed by a web browser within the context of a web page.
- CSS(Cascading Style Sheets) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document,
- SQL is the language used to manipulate relational databases. It is tied closely with the relational model. It is issued for the purpose of data definition and data manipulation.
- Java Server pages is a simple yet powerful technology for creating and maintaining dynamic-content web pages. It is based on the Java programming language. It can be thought of as an extension to servlet because it provides more functionality than servlet A JSP page consists of HTML tags and JSP tags. The jsp pages are easier to maintain than servlet because we can separate designing and development.

We require a JDBC connection between the front end and back end components to write to the database and fetch required data.

2.2.2 Hardware Requirements

- Processor – Pentium IV or above
- RAM – 2 GB or more
- Hard disk – 3 GB or mor

Detailed Design

3.1 System design

The web server needs a JSP engine, i.e, a container to process JSP pages. The JSP container is responsible for intercepting requests for JSP pages. A server(generally referred to as application or web server) supports the Java Server Pages. This server will act as a mediator between the client browser and a database. The following diagram shows the JSP architecture.

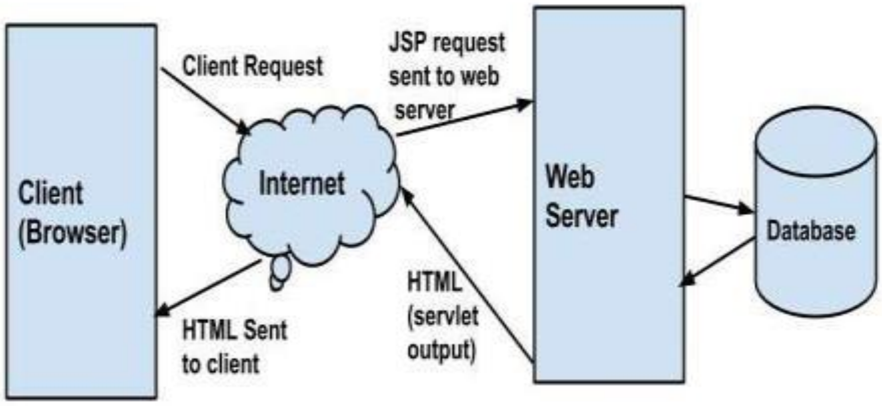


Fig 3.1 JSP architecture

Three-tier Client / Server database architecture is commonly used architecture for web applications. Intermediate layer called Application server or Web Server stores the web connectivity software and the business logic(constraints) part of application used to access the right amount of data from the database server. This layer acts like medium for sending partially processed data between the database server and the client.

Database architecture focuses on the design, development, implementation and maintenance of computer programs that store and organize information for businesses, agencies and institutions.

A database architect develops and implements software to meet the needs of users. Several types of databases, including relational or multimedia, may be created. Additionally, database architects may use one of several languages to create databases, such as structured query language (SQL). SQL is a database computer language designed for the retrieval and management of data in a relational database.

3.2 ER Diagram

An entity–relationship model is usually the result of systematic analysis to define and describe what is important to processes in an area of a business.

An E-R model does not define the business processes; it only presents a business data schema in graphical form. It is usually drawn in a graphical form as boxes (entities) that are connected by lines (relationships) which express the associations and dependencies between entities.

An ER model can also be expressed in a verbal form, for example: one building may be divided into zero or more apartments, but one apartment can only be located in one building.

Entities may be characterized not only by relationships, but also by additional properties (attributes), which include identifiers called "primary keys". Diagrams created to represent attributes as well as entities and relationships may be called entity-attribute-relationship diagrams, rather than entity-relationship models.

An ER model is typically implemented as a database. In a simple relational database implementation, each row of a table represents one instance of an entity type, and each field in a table represents an attribute type. In a relational database a relationship between entities is implemented by storing the primary key of one entity as a pointer or "foreign key" in the table of another entity

There is a tradition for ER/data models to be built at two or three levels of abstraction. Note that the conceptual-logical-physical hierarchy below is used in other kinds of specification, and is different from the three schema approach to software engineering.

While useful for organizing data that can be represented by a relational structure, an entity-relationship diagram can't sufficiently represent semi-structured or unstructured data, and an ER Diagram is unlikely to be helpful on its own in integrating data into a pre-existing information system.

Three main components of an ERD are the entities the relationship between those entities, and the cardinality, which defines that relationship in terms of numbers.

Cardinality notations define the attributes of the relationship between the entities. Cardinalities can denote that an entity is optional (for example, an employee rep could have no customers or could have many) or mandatory (for example, there must be at least one product listed in an order.)

The three main cardinal relationships are:

- One-to-one (1:1) - For example, if each member in a database is associated with one mailing address.
- One-to-many (1:M) - For example, a single member might attend multiple batches. The member is associated with multiple entities, but all those entities have a single connection back to the same member.
- Many-to-many (M:N). For example, at a gym when one instructor takes care of multiple members, one employee will be mapped to many members.

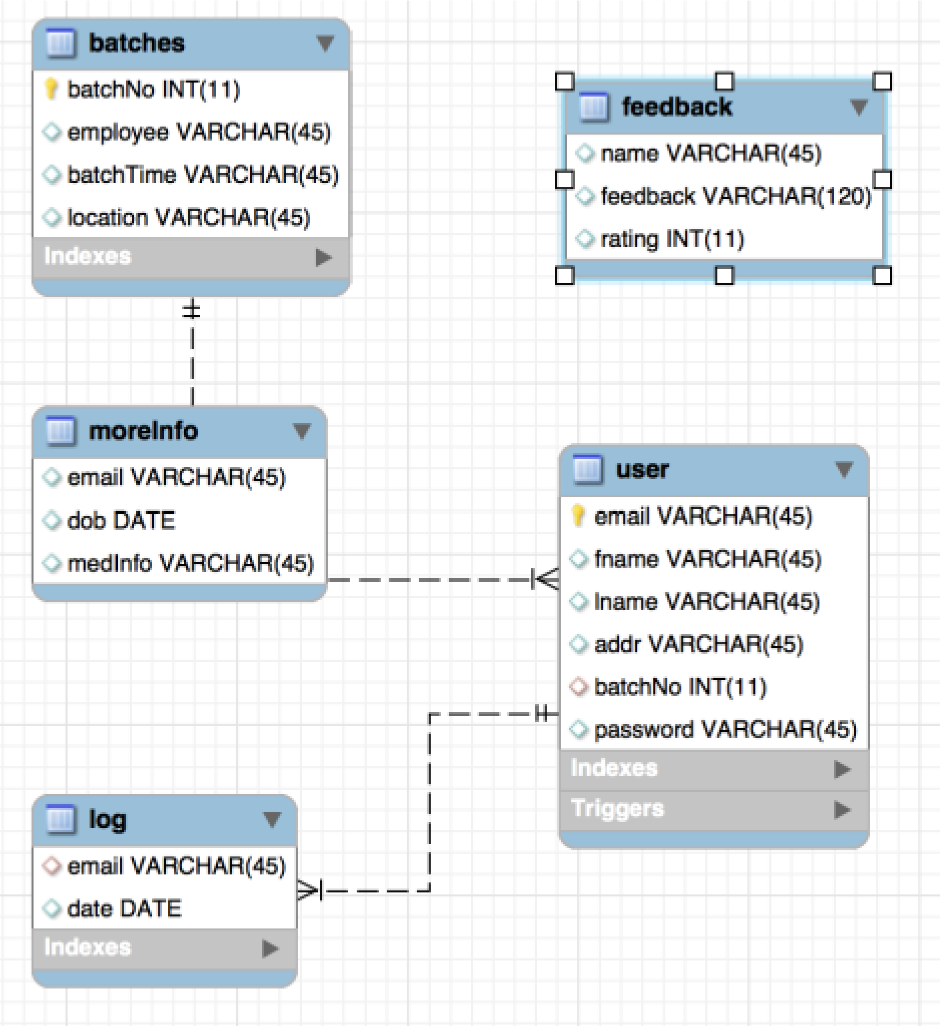
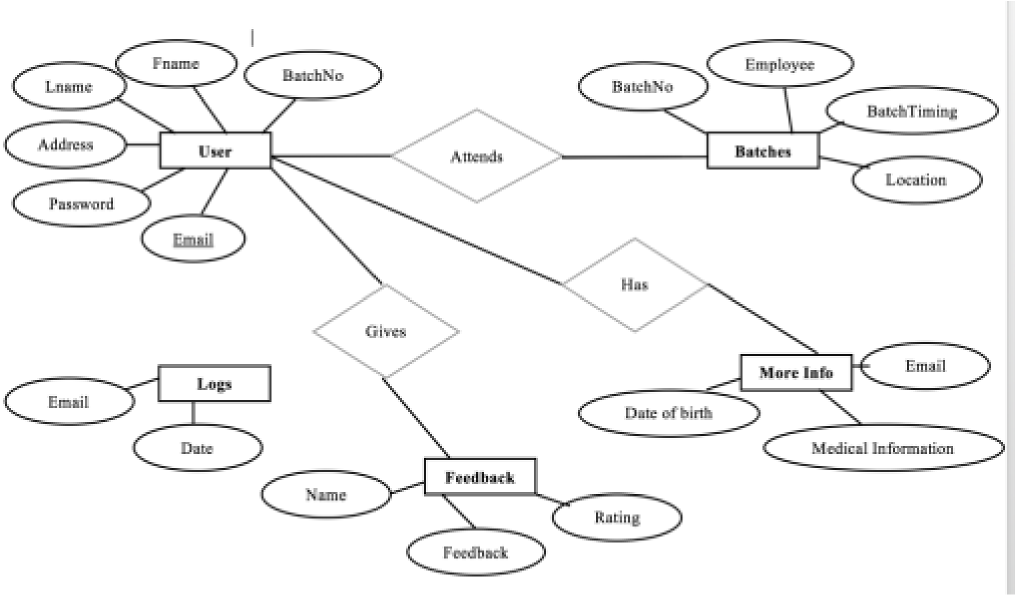


Fig: The ER Diagram for Gym Database Management

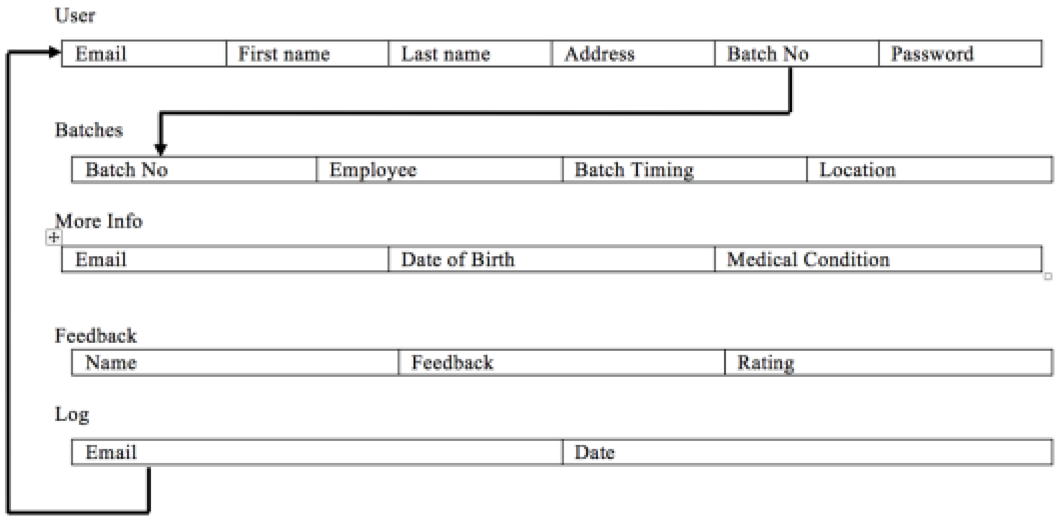

Fname
BatchNo
Lname
Address
User
Password
Email
Attends
Employee
BatchNo
BatchTiming
Batches
Location
Has
Gives
Logs
Email
Date
More Info
Email
Date of birth
Medical Information
Feedback
Name
Rating
Feedback

3.3 Relational Schema

The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database.

A relational schema shows references among fields in the database. When a primary key is referenced in another table in the database, it is called a foreign key. This is denoted by an arrow with the head pointing at the referenced key attribute.

A schema diagram helps organize values in the database. It also gives an idea of what order the tables should be created in. The following diagram shows the schema diagram for the database.



Schema Diagram

3.4 Description of Tables

The database realestate consists of five tables:

1. Users- All user details such as email, name etc
2. Batches – All batch details including the instructors and timings
3. Feedback – contains all feedback given by the members
4. MoreInfo - contains medical conditions and age of all users
5. Log – contains logs of all sign ups

USERS

- Email – Primary key. Identifies user uniquely
- Password – Used during login
- Fname – First name of the user

- Lname – Last name of the user
- Addr –address of the user
- BatchNo – batch no of the user

Batches

- BatchNo – Primary key. Uniquely identifies each bacth
- Employee – Mentions employee assigned to the batch
- BatchTimings– Mentions the batch timings of each batch
- Location- The address of the respective batch

Feedback

- Name – Optional name of the user for feedback
- Feedback – The actual feedback of the member
- Rating- Rating provided by the member

More Information

- Email – Email Id of the member to identify him/her
- Date of Birth – To know the age of the person.
- Medical Info – All medical conditons of the member

Log

- Email – The email used to sign up.
- Date – The date on which the sign up took pla

Implementation

4.1 Modules and their requirements

1. Member registration and login:

A new member can register by providing their full name and contact details with a unique email Id. The user can login with their email and password.

2. Upload Medical Information:

When a user uploads medical details, their emial ID is adding their medical conditions to the moreInfo table so that the admin can tailor make routines and workouts for the particular individual.

3.Upload Feedback:

Any user can upload their feedback of the institution without stating their name or also by providing their names. This information can be viewed by the owner to improve the institution.

4. View Batches

A user can view a list of batches the are available for them to attend along with the instructor details.

Scope:

This system is flexible and efficient and allows easy access to member information. Security, speed and accuracy should be focused on. It is a user friendly system and can overcome some user validation checks. A user can upload any feedback whick will be viewed by the owner to make improvements.

All the batch information is suitably maintained on the server and can be accessed when required. It identifies various sources of information and accordingly provides access to the requested details.

The system maintains details such as type of batch, instructor, locations available, etc.

Triggers and stored procedures:

The project makes use of a trigger to compile details of the member signing up on the portal to keep track of user activity on the site. The trigger automatically logs the information of the member along with the signing up date.

A stored procedure is used to check whether the user is valid and returns their email ID whenthey log in. The procedure takes email as input parameter. It checks the presence of such user in the database. If present, it fetches the corresponding email ID and returns it as an output parameter..

4.2 Result

The resulting system is able to:

- Authenticate user credentials during login
- Allows user to quickly and easily look to all the batches they can attend
- The user can upload their feedback
- Stores medical information as uploaded by the member

Testing

5.1 Software testing

Testing is the process used to help identify correctness, completeness, security and quality of developed software. This includes executing a program with the intent of finding errors. It is important to distinguish between faults and failures. Software testing can provide objective, independent information about the quality of software and risk of its failure to users or sponsors. It can be conducted as soon as executable software (even if partially complete) exists. Most testing occurs after system requirements have been defined and then implemented in testable programs.

5.2 Module testing and integration

Module testing is a process of testing the individual subprograms, subroutines, classes, or procedures in a program. Instead of testing whole software program at once, module testing recommend testing the smaller building blocks of the program. It is largely white box oriented. The objective of doing Module testing is not to demonstrate proper functioning of the module but to demonstrate the presence of an error in the module. Module testing allows to implement parallelism into the testing process by giving the opportunity to test multiple modules simultaneously.

In this Gym DataBase Management System, when a user logs in or registers, their email Id for the session is maintained internally. When a user wants to upload medical, their user ID is automatically saved into the database for the corresponding member. The system displays all batces uploaded by the owner when a user logs in. All the feedback provided by the members are stored and viewed by the owner to make the necessary improvements. Hence all the modules are linked by identifying entities that are maintained in the database.

Testing has been conducted using various test cases. The system sustains some errors such as invalid batch No. A user cannot upload data unless they are logged into the system.

Limitations:

The Current project can only maintain details of the batches. It does not allow interaction between users. This could be turned into an fitness training platform with suitable resources. Edit access isn't allowed to any user as it requires some kind of user authentication process. This will be implemented in future versions of the project.

Snapshots

This chapter consists of working screenshots of the project with code snippets.

6.1 Registration and login

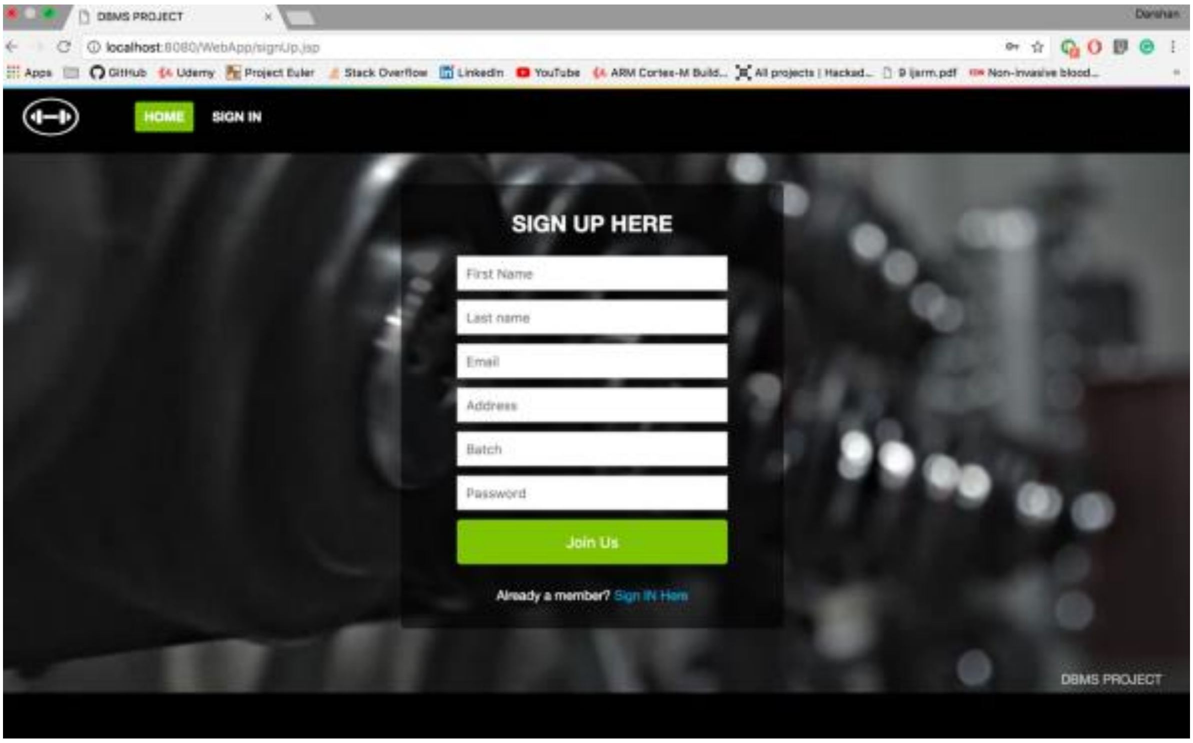


Fig 6.1 The registeration page

A new member can register with a unique email, password, first name, last name, address and batch number. A unique email id identifies each user in the back end. A new entity is created in the database on successful registration. The user is then redirected to the sign in page.

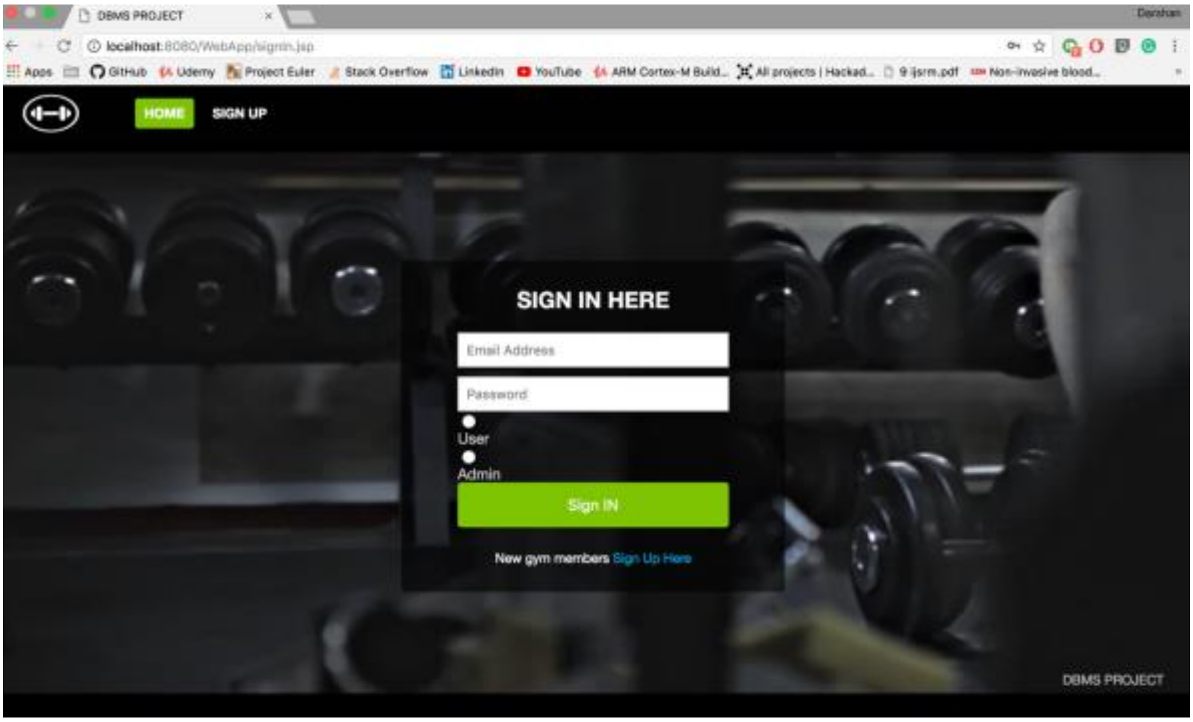


Fig 6.2 The login page

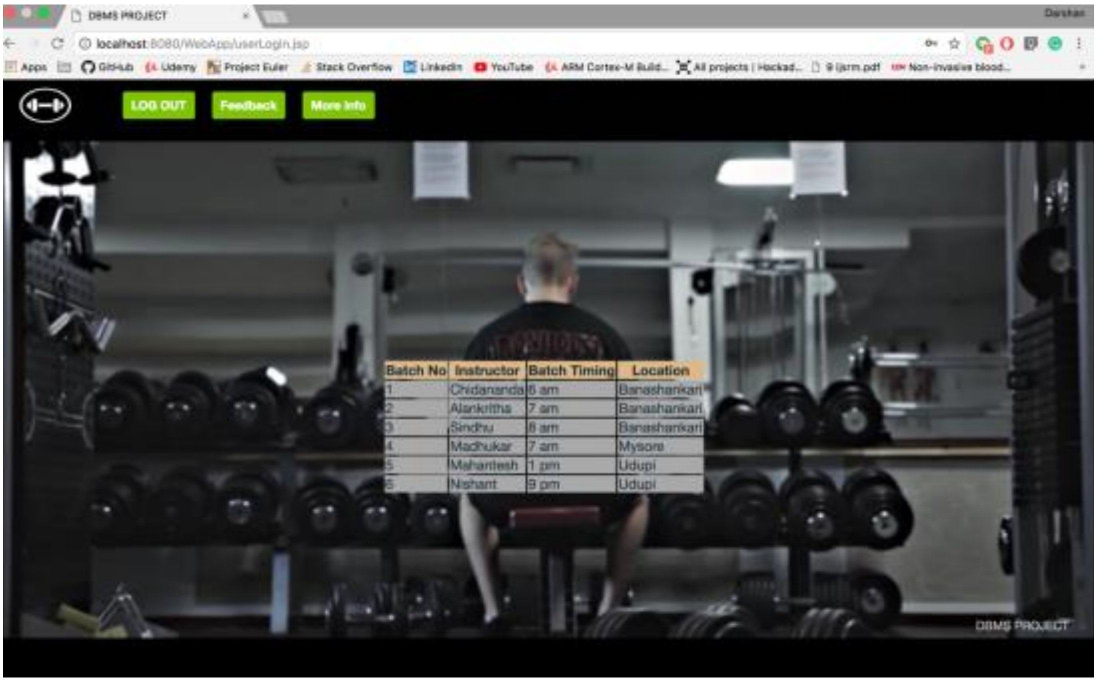
A previously registered member can login through this page with valid email and password.

```
String query="call sp()";
Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,name,password);
        PreparedStatement st=con.prepareStatement(query);
        String user=request.getParameter("usr");
        String pass=request.getParameter("pwd");
        String cust=request.getParameter("cust");


              ResultSet rs=st.executeQuery();
              while(rs.next()){
                 user1=rs.getString(1);
                 pass1=rs.getString(2);
                 if(cust.equals("1")){
```

```
if(user.equals(user1)&&pass.equals(pass1))
{ response.sendRedirect("userLogin.jsp");
state=1;
User.name=user1;
User.email=user;
}
}
if(cust.equals("2")){
if(user.equals(user1)&&pass.equals(pass1))
{ response.sendRedirect("adminLogin.jsp");
state=1;
User.name=user1;
User.email=user;
}
}
```

The above snippet makes use of stored procedure sp to retrive email and password

6.2 The Homepage



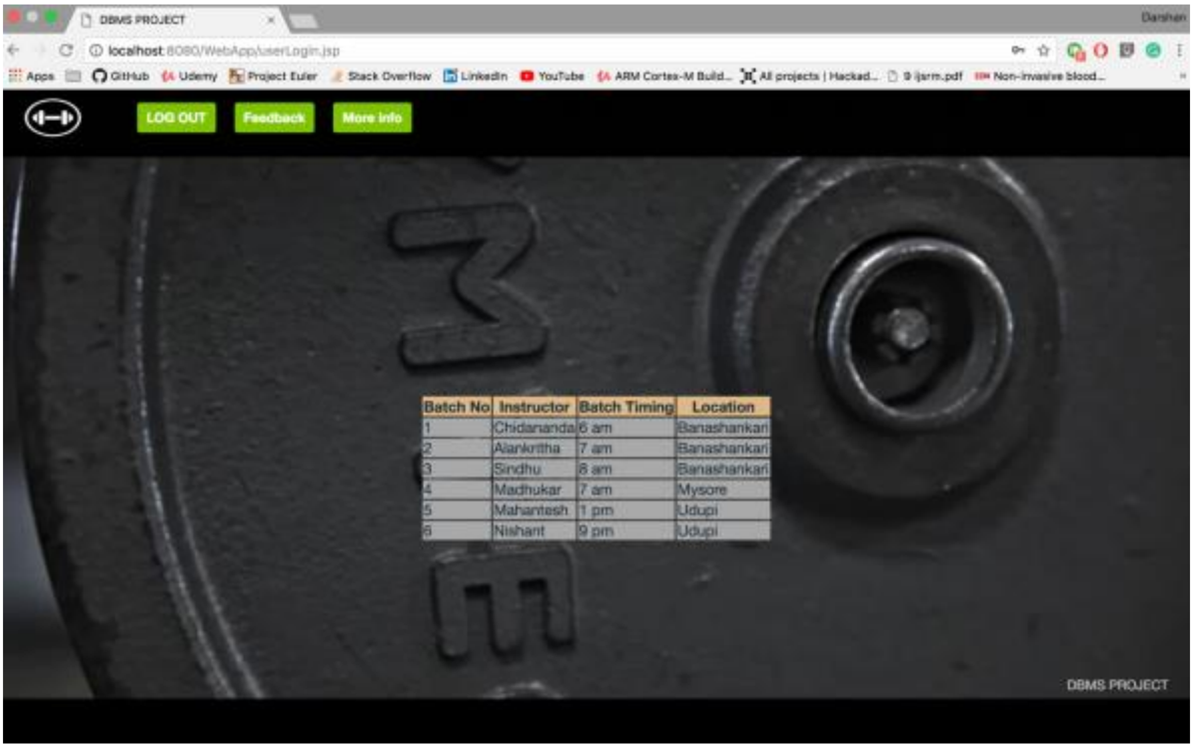
Batch No	Instructor	Batch Timing	Location
1	Chidananda	6 am	Banashankari
2	Alankritha	7 am	Banashankari
3	Sindhu	8 am	Banashankari
4	Madhukar	7 am	Mysore
5	Mahantesh	1 pm	Udupi
6	Nishant	9 pm	Udupi

Fig 6.4 Homepage navigation

The user is redirected to the homepage on successful registration or login. They will have three options to navigate from here

1. Look for batches uploaded by the admin.
2. Upload medical information.
3. Upload Feedback.

1. The batches are displayed on signing in.



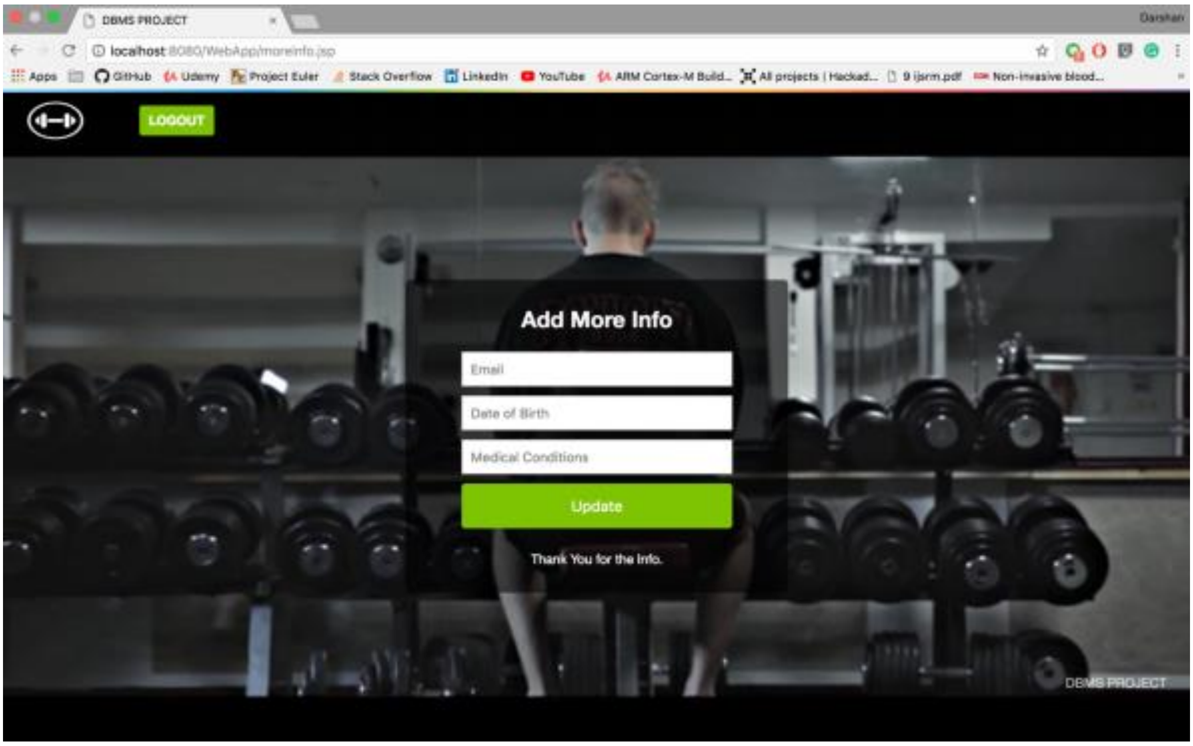
Batch No	Instructor	Batch Timing	Location
1	Chidananda	6 am	Banashankari
2	Alankritha	7 am	Banashankari
3	Sindhu	8 am	Banashankari
4	Madhukar	7 am	Mysore
5	Mahantesh	1 pm	Udupi
6	Nishant	9 pm	Udupi

CODE:

```
Class.forName("com.mysql.jdbc.Driver").newInstance();
                    Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/gymManagement",
          "root", "root");
```

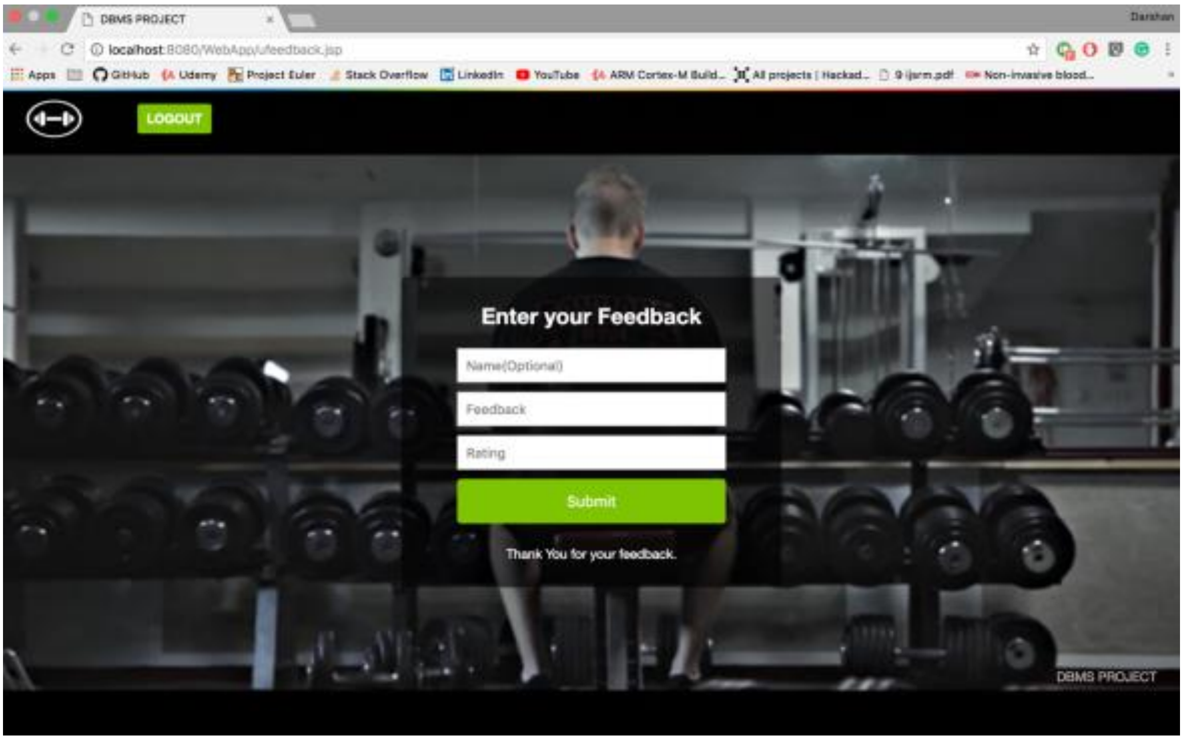
```
     String Query="select* from feedback";
     Statement st=con.createStatement();
     ResultSet rs=st.executeQuery(Query);
     while(rs.next())
     {
        %>
         <tr style="background-color: darkgray">
         <td><%=rs.getString("name")%></td>
         <td><%=rs.getString("feedback")%></td>
         <td><%=rs.getString("rating")%></td>

         </tr>
        <%
     }
  }
```

2.Upload medical information



Thank You for the Info.

3.Upload feedback



Thank You for your feedback.

Fig 6.6 Houses
available

The user can upload the feedback without stating their name.

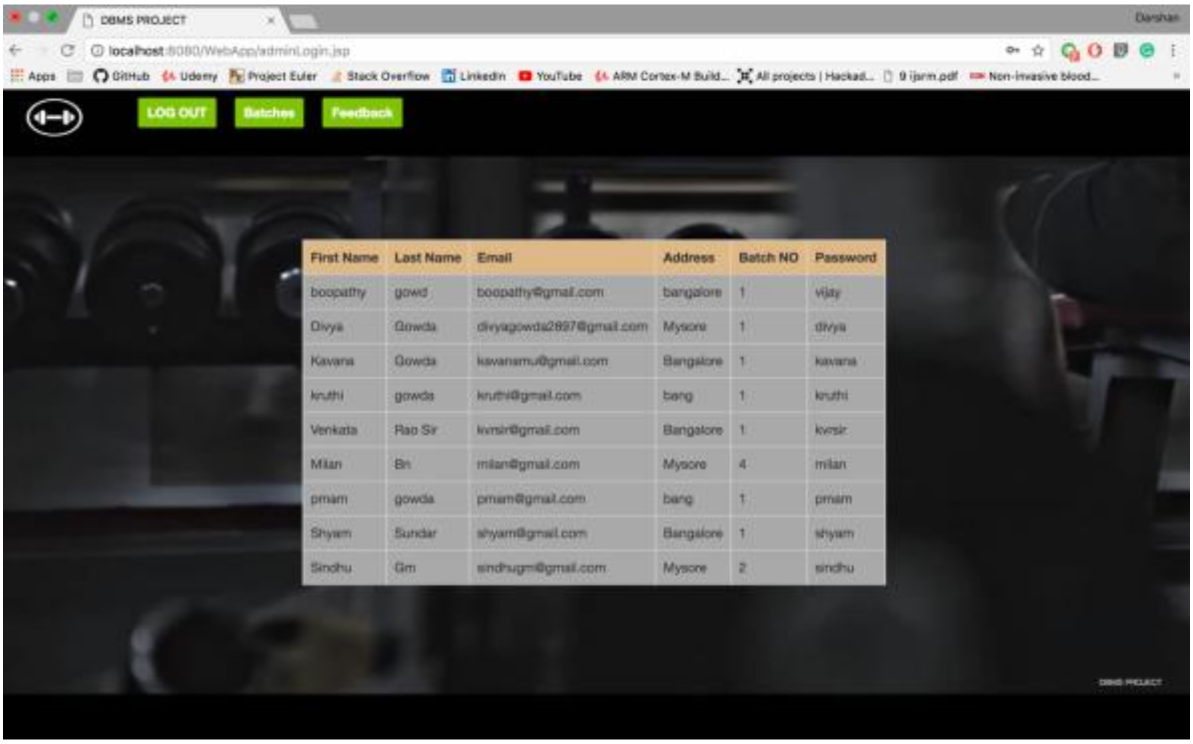
The Admin Side Login.

The admin has many more views than the normal member. On logging in the admin is showed all the existing members of the fitness center

The owner has five options upon logging in they are:

1. View all existing members
2. Add or update all available batches to the database
3. Look at all the feedback given by the users.
4. View all medical conditions of the gym members
5. Verify all the logs of the portal

1. View all existing members

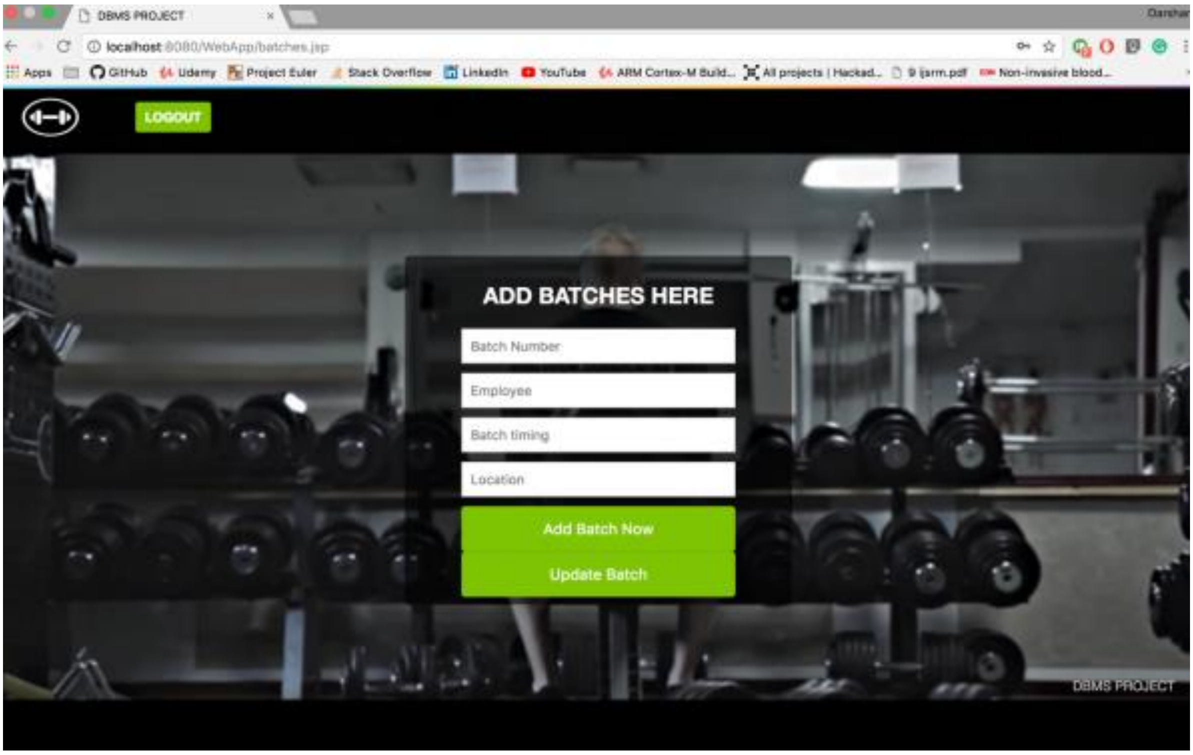


First Name	Last Name	Email	Address	Batch NO	Password
boopathy	gowd	boopathy@gmail.com	bangalore	1	vijay
Divya	Gowda	divyagowda2897@gmail.com	Mysore	1	divya
Kavana	Gowda	kavanamu@gmail.com	Bangalore	1	kavana
kruthi	gowda	kruthi@gmail.com	bang	1	kruthi
Venkata	Rao Sir	kvrsir@gmail.com	Bangalore	1	kvrsir
Milan	Bn	milan@gmail.com	Mysore	4	milan
pmam	gowda	pmam@gmail.com	bang	1	pmam
Shyam	Sundar	shyam@gmail.com	Bangalore	1	shyam
Sindhu	Gm	sindhugm@gmail.com	Mysore	2	sindhu

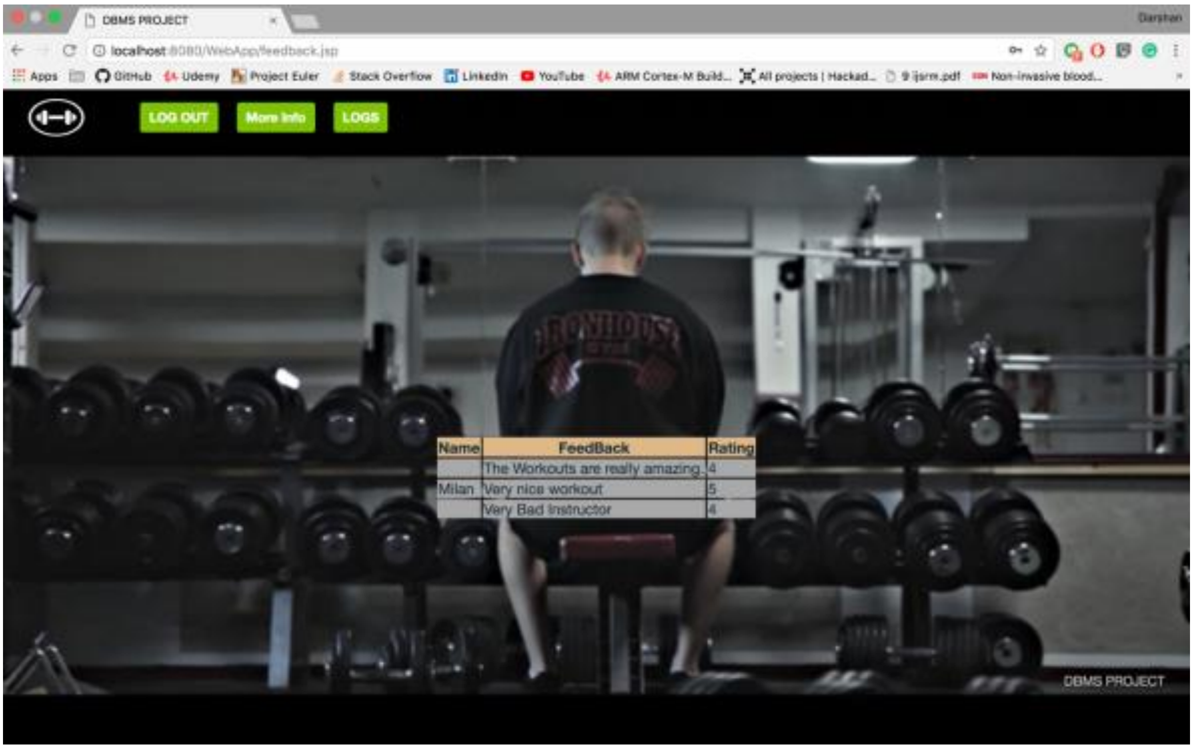
Code:

```
Class.forName("com.mysql.jdbc.Driver").newInstance();
                    Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/gymManagement",
          "root", "root");
                String Query="select* from user";
                Statement st=con.createStatement();
                ResultSet rs=st.executeQuery(Query);
                while(rs.next())
                {
                   %>
                    <tr style="background-color: darkgray">
                    <td><%=rs.getString("fname")%></td>
                    <td><%=rs.getString("lname")%></td>
                    <td><%=rs.getString("email")%></td>
                    <td><%=rs.getString("addr")%></td>
                    <td><%=rs.getString("batchNo")%></td>
                    <td><%=rs.getString("password")%></td>

                    </tr>
                   <%
                }

             }
```

2. Add or update all available batches to the database

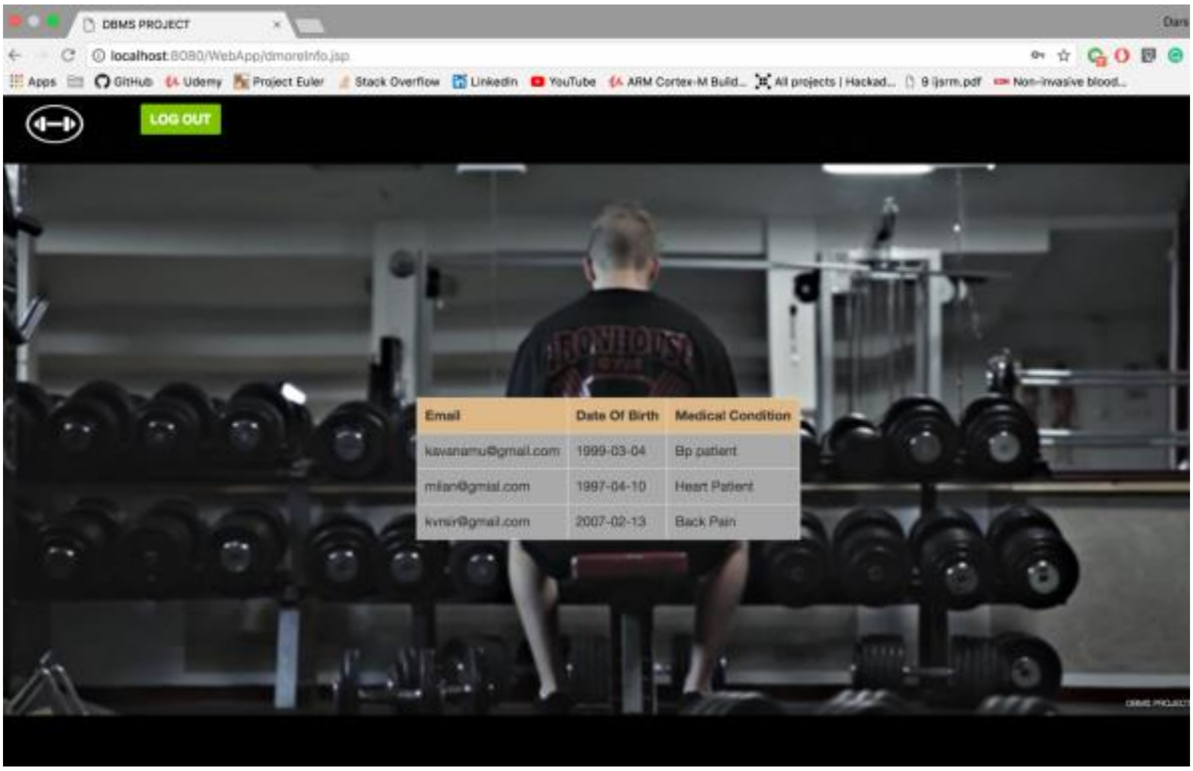


3. Look at all the feedback given by the users.



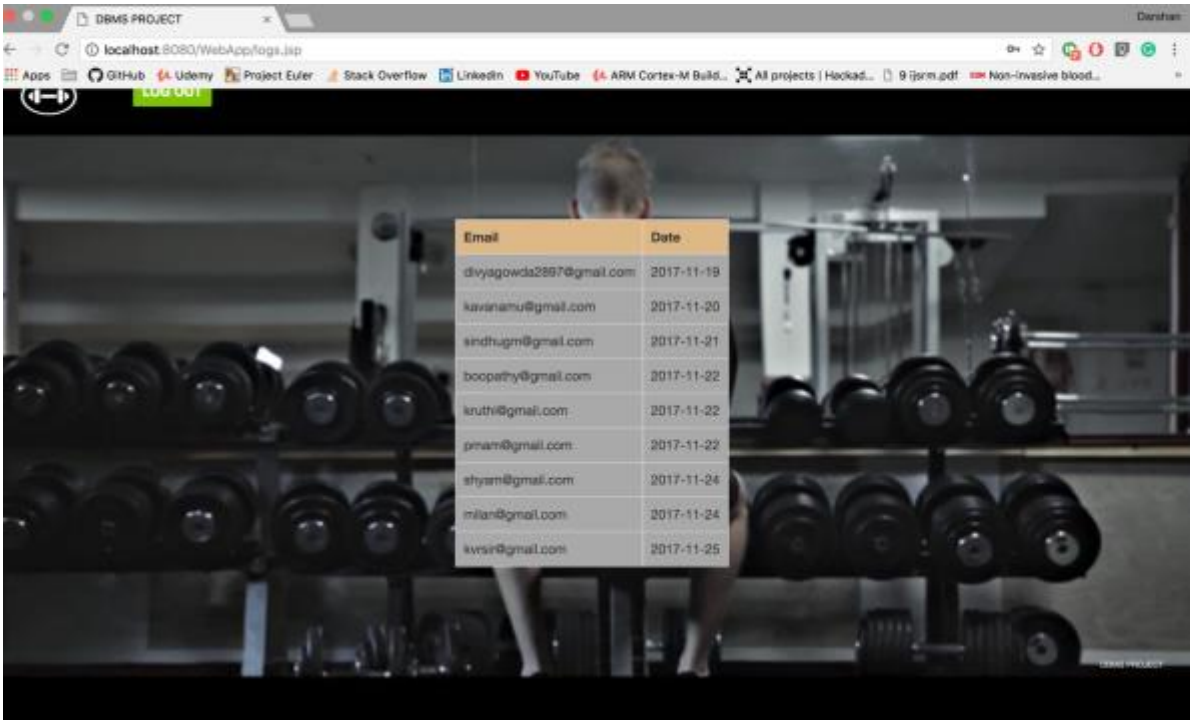
Name	FeedBack	Rating
	The Workouts are really amazing.	4
Milan	Very nice workout	5
	Very Bad Instructor	4

4. View all medical conditions of the gym members



Email	Date Of Birth	Medical Condition
kavanamu@gmail.com	1999-03-04	Bp patient
milan@gmial.com	1997-04-10	Heart Patient
kvmsir@gmail.com	2007-02-13	Back Pain

5. Verify all the logs of the portal



Email	Date
divyagowda2897@gmail.com	2017-11-19
kavanamu@gmail.com	2017-11-20
sindhugm@gmail.com	2017-11-21
boopathy@gmail.com	2017-11-22
kruthi@gmail.com	2017-11-22
pmam@gmail.com	2017-11-22
shyam@gmail.com	2017-11-24
milan@gmail.com	2017-11-24
kvrsir@gmail.com	2017-11-25

Code:

```
{
                Class.forName("com.mysql.jdbc.Driver").newInstance();
                   Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/gymManagement",
          "root", "root");
                String Query="select* from log";
                Statement st=con.createStatement();
                ResultSet rs=st.executeQuery(Query);
                while(rs.next())
                {
                   %>
                    <tr style="background-color: darkgray">
                    <td><%=rs.getString("email")%></td>
                    <td><%=rs.getString("date")%></td>



                    </tr>
                   <%
                }

             }
```

6.6 Logout

When the user logs out, their session id is cleared and they are redirected to the home page. A new user can log in after this.

6.7 Navigation

The user can navigate through pages from the homepage.

An additional navigation functionality is provided by a HTML 5 component called navbar. This is present at the top of every webpage to allow easy navigation

Sample HTML code:

```
<nav class="topnav_desktop_menu" role="navigation">
<ul class="topnav_menu_desktop_main" role="menubar">

<li class="topnav_desktop_menu_items" data-menu-id="join">
<a class="topnav_menu_join js-topnav_menu_auth" href="index.jsp" rel="toggle"
role="button">
LOG OUT </a>

</li>

</ul>
</nav>
```

Chapter 7

Conclusion

The Gym DataBase Management System allows a member to easily upload about themselves and access details about fitness center. It allows simplified operation and is a time saving platform. It offers a seamless interface between gym owners and members.

The application has been completed successfully and tested with suitable test cases. It is user friendly and contains suitable options for users.

This is developed using Java as front end and MySQL as back end in Windows environment. The goals achieved by this project are:

- Instant access
- User friendly environment
- Efficient management of records
- Simplification of operations

Chapter 8

Future enhancements

Future upgrades to this project will implement:

- Filtering options – Batch of a particular type can be filtered based on exact requirements
- Modify uploaded data – This will allow members to modify details of previously uploaded information.
- Training Platform – The system currently allows only listing. Training sessions through video or audio can be incoporated.

Chapter 9

References

1) Ramakrishnan, R., &amp; Gehrke, J. (2011). Database management systems. Boston: McGraw-Hill.
2) Monson-Haefel, R. (2007). J2EE Web services. Boston, Mass.: Addison-Wesley. Silberschatz, A., Korth, H. F., &amp; Sudarshan, S. (2011).
3) Database systems concepts. Estados Unidos: McGraw-Hill Companies, Inc.
4) Hanna P. (2002): JSP 2.0 The Complete Reference, Second Edition McGraw Hill Education;
5) David F. (2011). JavaScript: The Definitive Guide Sixth edition. Shroff