A Project Report

On

"Live Project on MySQL Database Management System for Faculty Interaction Management"

by

Yash Joshi (TEIT134)
Sanket Kankarej (TEIT38)
Snehdeep Tambe (TEIT173)
Aniruddha Tupe (TEIT174)

Under the guidance of

Mrs. Sandhya S. Waghere

Department of Information Technology

SAVITRIBAI PHULE PUNE UNIVERSITY

2019-2020

PIMPRI CHINCHWAD COLLEGE OF ENGINEERING

Date:

CERTIFICATE

This is to certify that,

Yash Joshi (TEIT134)
Sanket Kankarej (TEIT138)
Snehdeep Tambe (TEIT173)
Aniruddha Tupe (TEIT174)

of class T.E IT; have successfully completed their project work on "MySQL Database Management System for Faculty Interaction Management" at PIMPRI CHINCHWAD COLLEGE OF ENGINEERING in the partial fulfillment of the Graduate Degree course in T.E at the Department of **Information Technology**, in the academic Year 2019-2020 Semester – I as prescribed by the Savitribai Phule Pune University.

Mrs. Sandhya S.Waghere
Guide

Head of the Department
(Department of Information Technology)

Acknowledgements

Right from the very beginning of this project, we have been getting help from many people in numerous ways. Through this acknowledgement, we are trying to express our gratitude to all those who have helped us in one way or the other in the successful completion of this project.

Although a single sentence hardly suffices, we would like to thank almighty god for blessing us with his grace and taking my endeavour to a successful culmination. We express my sincere gratitude to our project guides Prof. Sandhya S.Waghere and Mrs. Swati Jadhav for constant support , valuable suggestions and providing me with adequate facilities , ways and means without which the successful completion of this project would not have been possible . We also thank our HOD Prof. Dr. Sonali Patil for their timely support and their suggestions.

Yash Joshi (TEIT134)
Sanket Kankarej (TEIT138)
Snehdeep Tambe (TEIT173)
Aniruddha Tupe (TEIT174)

Contents

List of Tables

List of Forms

Acronyms

DBMS	Database Management System
GUI	Graphical User Interface
SQL	Structured Query Language
API	Application Program Interface
FDP	Faculty Development Program
IDE	Integrated Development Environment
RDBMS	Relational Database Management System

Abstract

This live, mini-project focuses on making a database maintained for management of faculty interaction with the outside world. The database management system uses MySQL at the back-end to store data. The front-end is developed in HTML, CSS Bootstrap and Vue to make an interactive website.

Workshops attended as well as organized by the faculty, conferences attended etc. are managed by the system. The system allows referencing of data from different tables in a database. The final project developed is in the form of a website.

Chapter 1

Introduction

1. Motivation

The faculty of our department attends many workshops, courses and conferences held in different colleges. Also, our college hosts numerous workshops/courses. All this data needs to be recorded. But as this data increases, difficulty arises in managing it.

If this data is stored using a Database Management System, and a GUI is made to maintain this data, it will reduce paper-work and improve efficient handling of this data. This motivated us to create a system for faculty of IT department of Pimpri Chinchwad College of Engineering to help them manage their data.

2. Problem Statement

A database has to be maintained for management of Faculty Interaction with the Outside World. The faculty of this institute attend as well as conduct workshops, seminars and conferences. The seminars are based on various topics and are conducted in different colleges. Some courses are held online too. The duration of these seminars' ranges from one to five days. Some of the workshops are held in the institute itself.

3. Framework of proposed work in project

The general flow of data is as follows:

Faculty interaction data will be added through Xampp to an interaction database created in SQLite. The front-end website retrieves and uses this data as per the user's requirements.

- The requirement analysis was performed.

- Taking into consideration the flaws of existing methodologies we discussed and listed down the requirements.
- Entity Relationship model for the mini project was developed
- A demo model was made and implemented.
- Later forms were made and faculty details were introduced.
- Testing was conducted using certain test data taken from the client.

3.1.Admin Module

The Admin has a separate login page. Only one account can be created as an admin. The admin signup stores admin's username and password.

The admin can login using this username and password for 2 main functions as follows:

1.3.1.1 Add a new faculty

Here, admin can add a faculty into the database. The following details must be mentioned for each faculty to be created:

1. Faculty name
2. Faculty username and corresponding password

After a faculty is added, his/her name automatically gets appended in the database and a unique id is associated with that faculty is generated. So when that data is to be retrieved, it is done through that id.

1.3.1.2 Remove a faculty

Here, admin can remove a faculty if he/she resigns. Admin can select a faculty from the list and delete all of it's interaction details.

1.3.1.2 Restore a deleted faculty

Here, admin can restore the details of any faculty deleted from the database.

3.2. General Module

This includes all faculties other than admin who wish to access the data and can perform the following actions:

1. View data of all the faculties.
2. Add a new workshop activity to their own record in the database.
3. Edit or delete data of their own record.

1.3.2.1 Student Module

This includes a general student login that can only access the interaction data. The student cannot modify or delete any data from the database.

Chapter 2

Literature Review

4. Introduction

This section includes the technology stack that has been used for the development of this project.

4.1.HTML5

HTML5 is a software solution stack that defines the properties and behaviors of web page content by implementing a markup based pattern to it.

HTML5 is the fifth and current major version of HTML, and subsumes XHTML. The current standard, the HTML Living Standard is developed by WHATWG, which is made up of the major browser vendors (Apple, Google, Mozilla, and Microsoft), with the Living Standard also existing in an abridged version.

HTML5 was first released in public-facing form on 22 January 2008, with a major update and "W3C Recommendation" status in October 2014. Its goals were to improve the language with support for the latest multimedia and other new features; to keep the language both easily readable by humans and consistently understood by computers and devices such as web browsers, parsers, etc., without XHTML's rigidity; and to remain backward-compatible with older software. HTML5 is intended to subsume not only HTML 4, but also XHTML 1 and DOM Level 2 HTML.[5]

HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and application programming interfaces (APIs) for complex web applications. For the same reasons, HTML5 is also a candidate for cross-platform mobile applications, because it includes features designed with low-powered devices in mind.

4.2.CSS3

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

The name *cascading* comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents.

In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

4.3.VueJS

Vue.js is an open-source JavaScript framework for building user interfaces and single-page applications. It is first created by Evan You, and now is maintained by him and the rest of the active core team members coming from various companies such as Netlify and Netguru. Vue.js features an incrementally adoptable architecture that focuses on declarative rendering and component composition. Advanced features required for complex applications such as routing, state management and build tooling are offered via officially maintained supporting libraries and packages, with Nuxt.js as one of the most popular solutions.

Vue uses an HTML-based template syntax that allows binding the rendered DOM to the underlying Vue instance's data. All Vue templates are valid HTML that can be parsed by specification-compliant browsers and HTML parsers. Vue compiles the templates into virtual DOM render functions. A virtual Document Object Model (or "DOM") allows Vue to render components in its memory before updating the browser. Combined with the reactivity system, Vue is able to calculate the minimal number of components to re-render and apply the minimal amount of DOM manipulations when the app state changes.

Vue users can use template syntax or choose to directly write render functions using JSX. Render functions allow application to be built from software components.

4.4.ExpressJS

Express.js, or simply Express, is a web application framework for Node.js, released as free and open-source software under the MIT License. It is designed for building web applications and APIs. It has been called the de facto standard server framework for Node.js.

The original author, TJ Holowaychuk, described it as a Sinatra-inspired server, meaning that it is relatively minimal with many features available as plugins. Express is the back-end component of the MEAN stack, together with the MongoDB database software and AngularJS front-end framework.

Express.js was founded by TJ Holowaychuk. The first release, according to Express.js's GitHub repository, was on the 22nd of May, 2010. Version 0.12

In June 2014, rights to manage the project were acquired by StrongLoop. StrongLoop was acquired by IBM in September 2015; in January 2016, IBM announced that it would place Express.js under the stewardship of the Node.js Foundation incubator.

4.5.MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language.

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB.

MySQL is a component of the LAMP web application software stack (and others), which is an acronym for *Linux, Apache, MySQL, Perl/PHP/Python*. MySQL is used by many database-driven web applications, including Drupal, Joomla, phpBB, and WordPress. MySQL is written in C and C++. The MySQL server software itself and the client libraries use dual-licensing distribution. They are offered under GPL version 2, or a proprietary license.

Support can be obtained from the official manual. Free support additionally is available in different IRC channels and forums. Oracle offers paid support via its MySQL Enterprise products. They differ in the scope of services and in price. Additionally, a number of third party organisations exist to provide support and services, including MariaDB and Percona.

MySQL has received positive reviews, and reviewers noticed it "performs extremely well in the average case" and that the "developer interfaces are there, and the documentation (not to mention feedback in the real world via Web sites and the like) is very, very good". It has also been tested to be a "fast, stable and true multi-user, multi-threaded sql database server".

5. Existing Methodologies

The system which is used nowadays has some drawbacks which need to be improved for better performance. As the technology is developed day by day we need to use this technology so we can get an efficient result in adequate time. Today, the institute maintains faculty record in spreadsheets. They have to employ people who personally handle this data and associated tasks. They insert faculty data into spreadsheets. Similarly, report generation has limited functionality since it takes lot of efforts to manually scan the data and get the required report. Mostly, these spreadsheet records are maintained by lab assistants and due to miscommunication or human errors the data may get erased.

Also students don't have access to this data. As a result they are unaware of their research interest. If a student has the same research interest they can approach the respective faculty.

6. Proposed Methodologies

The system architecture has a web app, a database server and various types of users as its components. We propose a system that will advance the organization, management and use of information and information technology, and enhance our understanding of the impact of information on the institution. The proposed system will use the approaches of professional and technical disciplines to address a core set of information. The proposed system is based on VueJS architecture and MySQL database.

In this project, we intend towards providing service based product with a vision to reduce tedious tasks to manage the data and also prevent manual errors. The proposed system uses MySQL as its database because of their simplicity and flexibility.

The application can be further extended to manage workshops or activities in which students have participated. Also this project can be very useful in managing

various placement records. Thus our proposed model is extendable for any institute which requires a database to be maintained.

Chapter 3

Software Requirement Specification

7. Hardware Requirements

- This project can be executed on any computer.
- Processor (CPU) with 2 gigahertz (GHz) frequency or above.
- Internet Connection Broadband (high-speed) Internet connection with a speed of 1 Mbps or higher.
- Keyboard and a Microsoft Mouse or some other compatible pointing device.

8. Software Requirements

- This project can run on the latest version (77.0.3865.90) of Chrome.
- The browser must be able to support HTML5 and CSS3.

Chapter 4

Assumptions

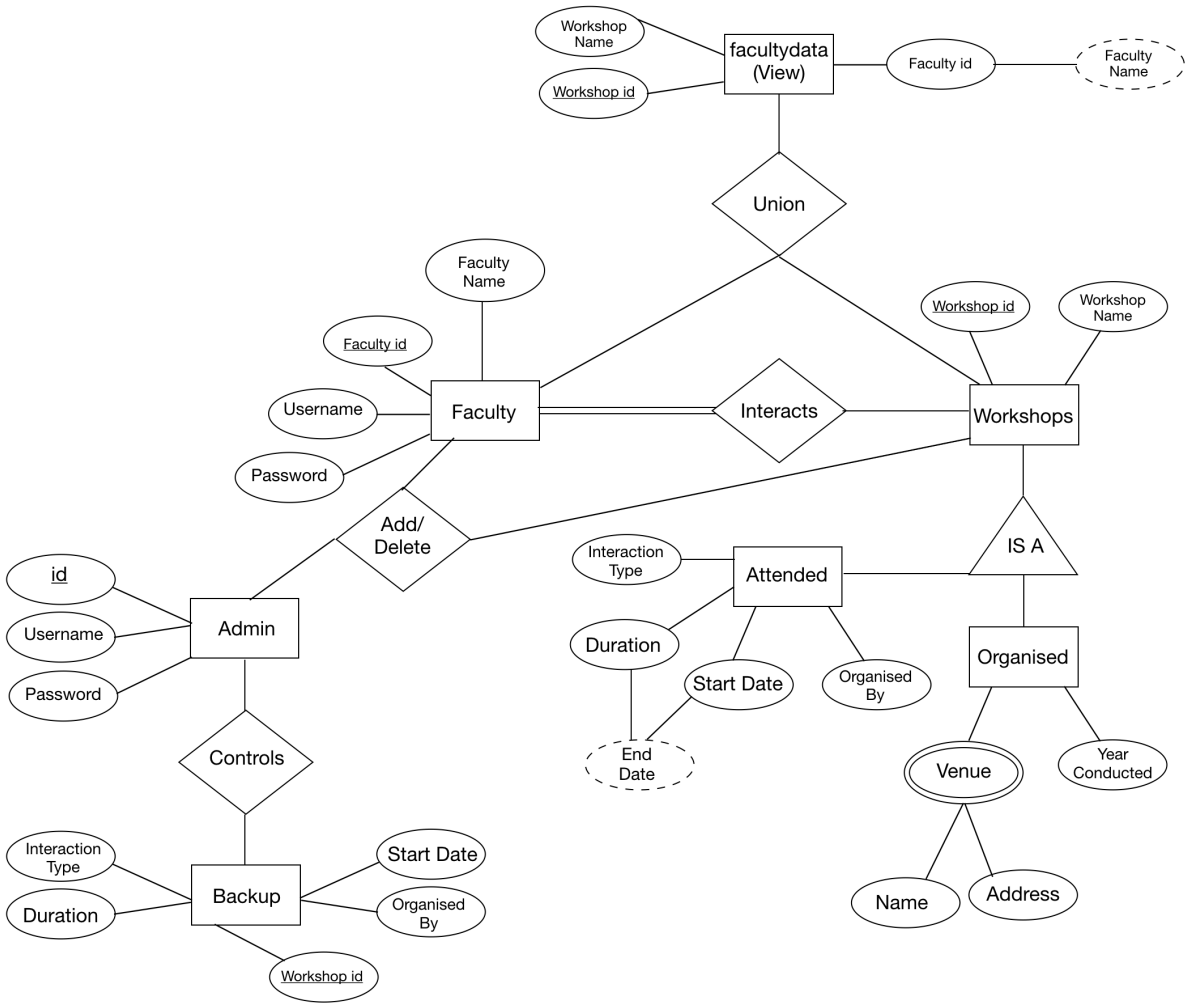
While creating this Web Application, we assumed that only three primary users would be accessing the database, them being:

1. Administrator
2. Faculties
3. Students

The application designed is a web application. It can be run on any personal computer and the data can be accessed by anyone. But only the admin can add or remove faculties. Also, it is assumed that there is only a single administrator.

Chapter 5

Entity Relationship Diagram



Chapter 6

Tables

Faculty Login:

fi_login(username, password, unique_id, faculty_name);

Table 1.1. Faculty login.

Attributes	Data Type	Specifications
username	Varchar	Not null
password	Varchar	Not null
unique_id	Integer	Primary Key
faculty_name	Varchar	Not null

```
CREATE TABLE fi_login (username varchar(20),
                       password varchar(100),
                       unique_id integer(3) PRIMARY AUTO_INCREMENT,
                       faculty_name varchar(20));
```

Faculty Interaction:

faculty_interaction_info(workshop_id, faculty_number, workshop_name, interaction_type, duration, start_date, organized_by);

Table 1.2 faculty_interaction_info

Attributes	Data Type	Specifications
workshop_id	Integer	Primary Key
faculty_number	Integer	Not null, Foreign key with reference to faculty_login
workshop_name	Varchar	Required
interaction_type	Varchar	Required
duration	Integer	Required
start_date	Date	Required
organized_by	Varchar	Required

```
CREATE TABLE faculty_interaction_info (
                        workshop_id integer PRIMARY AUTO_INCREMENT,
                        faculty_number integer NOT NULL,
                        workshop_name varchar(100),
                        interaction_type varchar(50),
                        duration integer(3),
                        start_date date,
                        organized_by varchar(50)
);
```

Organized Workshops:

workshop_organized(workshop_id, faculty_number, workshop_name, venue, year_conducted);

Table 1.3 workshop_organized

Attributes	Data Type	Specifications
workshop_id	Integer	Primary Key
faculty_number	Integer	Not null, Foreign key with reference to faculty_login
workshop_name	Varchar	Required
venue	Varchar	Required
year_conducted	Date	Required

```
CREATE TABLE workshop_organized (
                        workshop_id integer PRIMARY AUTO_INCREMENT,
                        faculty_number integer(3),
                        workshop_name varchar(100),
                        venue varchar(50),
                        year_conducted date
);
```

Chapter 7

Forms and Reports

GUI forms:

Figure 7.1. Login Form

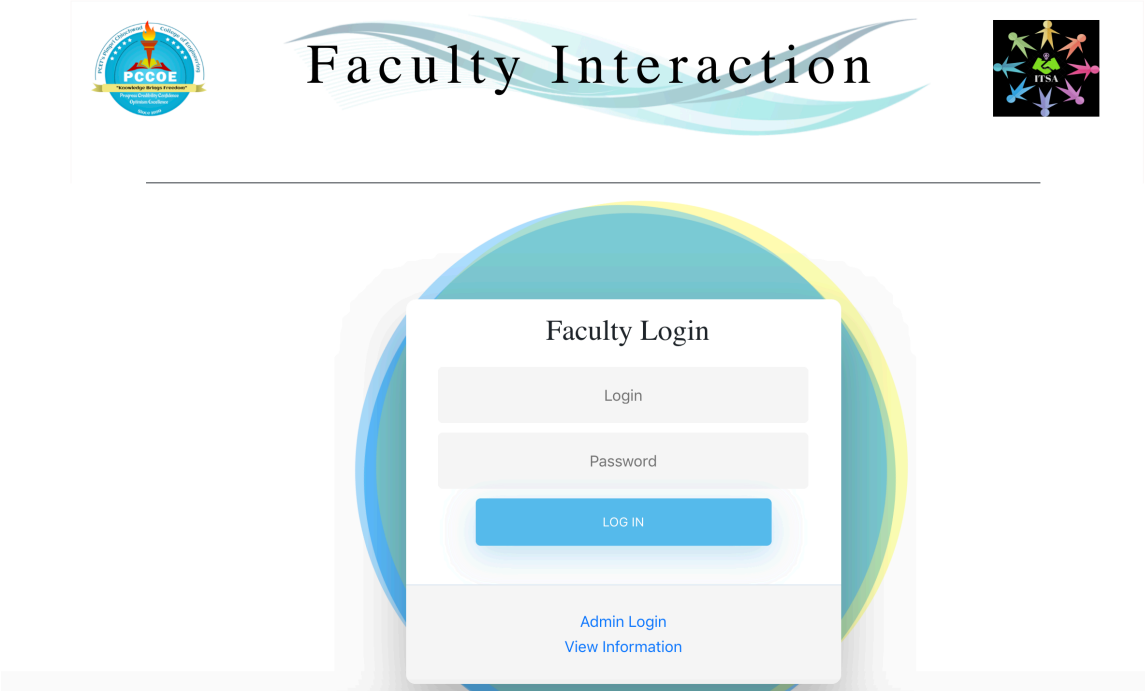


Figure 7.2 Search

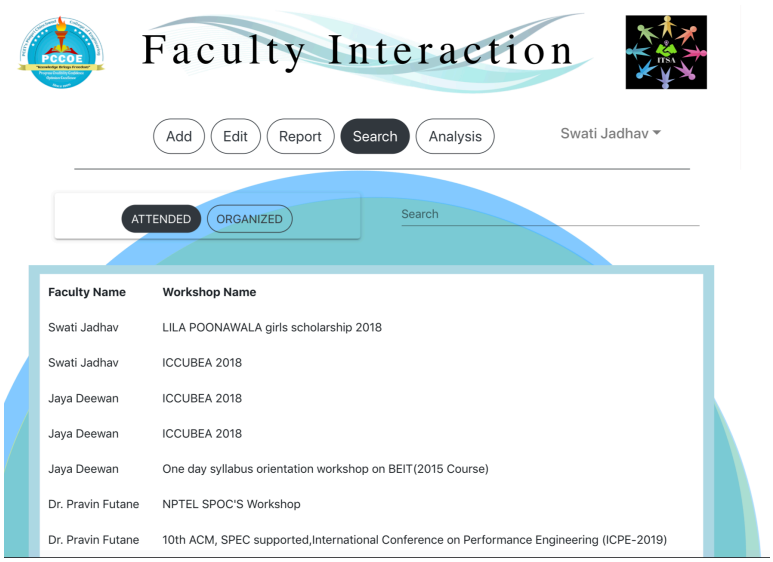


Figure 7.3. Report Select

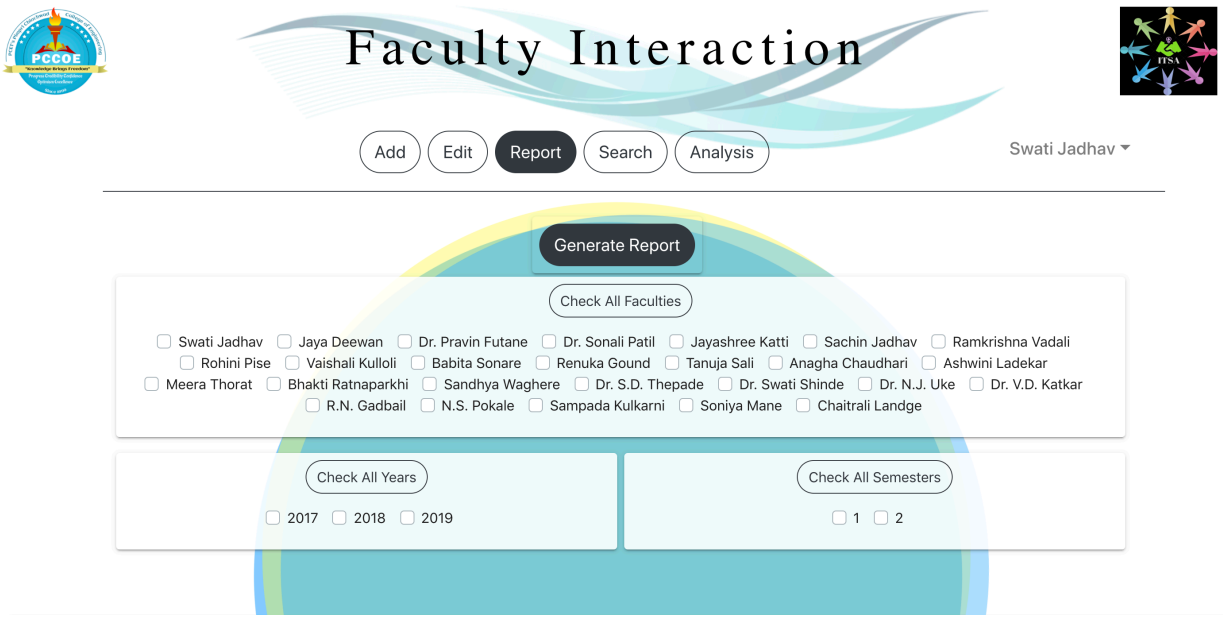
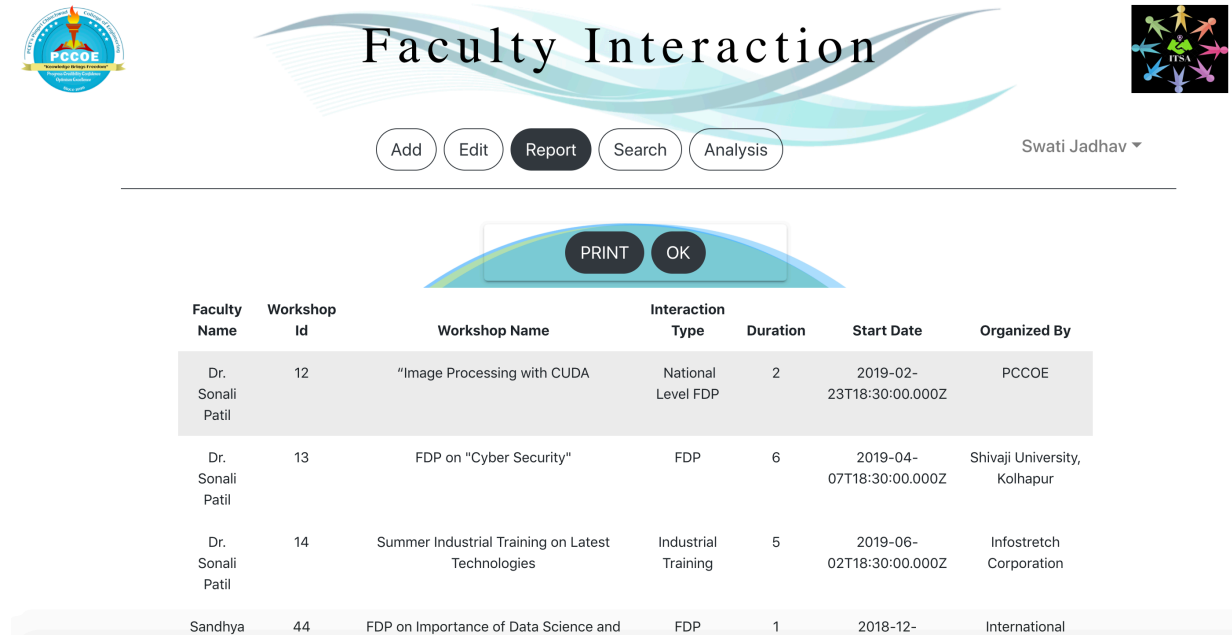


Figure 7.4 Report result



Faculty Name	Workshop Id	Workshop Name	Interaction Type	Duration	Start Date	Organized By
Dr. Sonali Patil	12	“Image Processing with CUDA	National Level FDP	2	2019-02-23T18:30:00.000Z	PCCOE
Dr. Sonali Patil	13	FDP on "Cyber Security"	FDP	6	2019-04-07T18:30:00.000Z	Shivaji University, Kolhapur
Dr. Sonali Patil	14	Summer Industrial Training on Latest Technologies	Industrial Training	5	2019-06-02T18:30:00.000Z	Infostretch Corporation
Sandhya	44	FDP on Importance of Data Science and	FDP	1	2018-12-	International

Figure 7.4. Add workshop

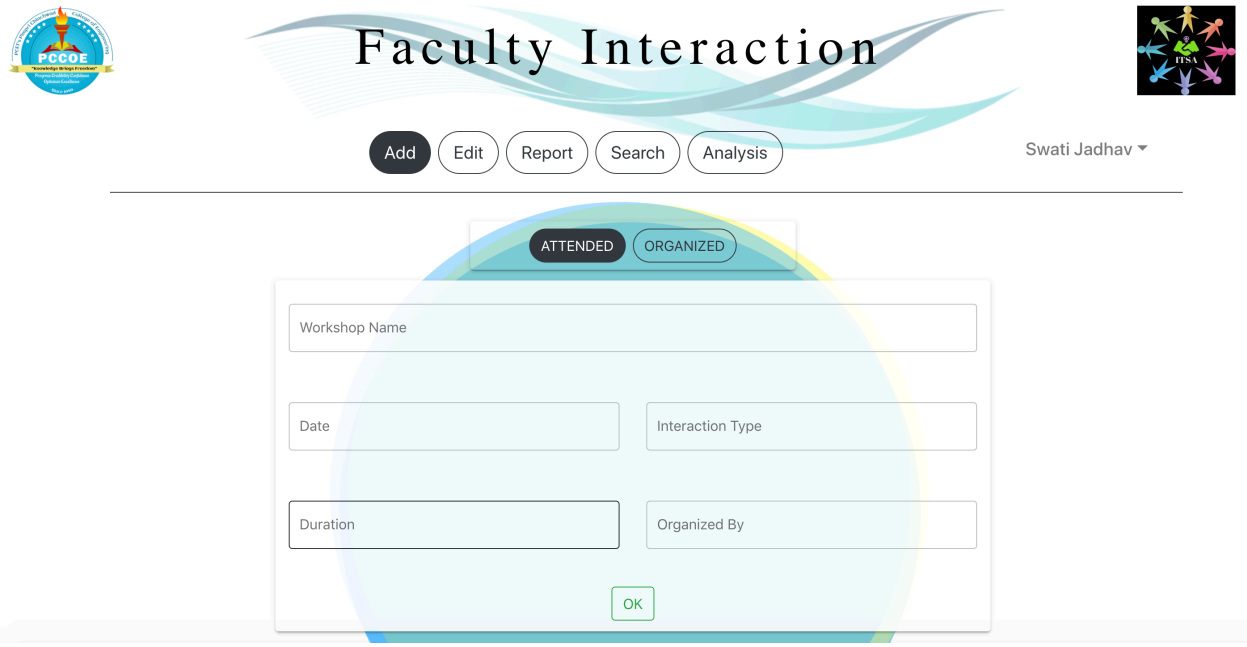


Figure 7.6 Edit Workshop

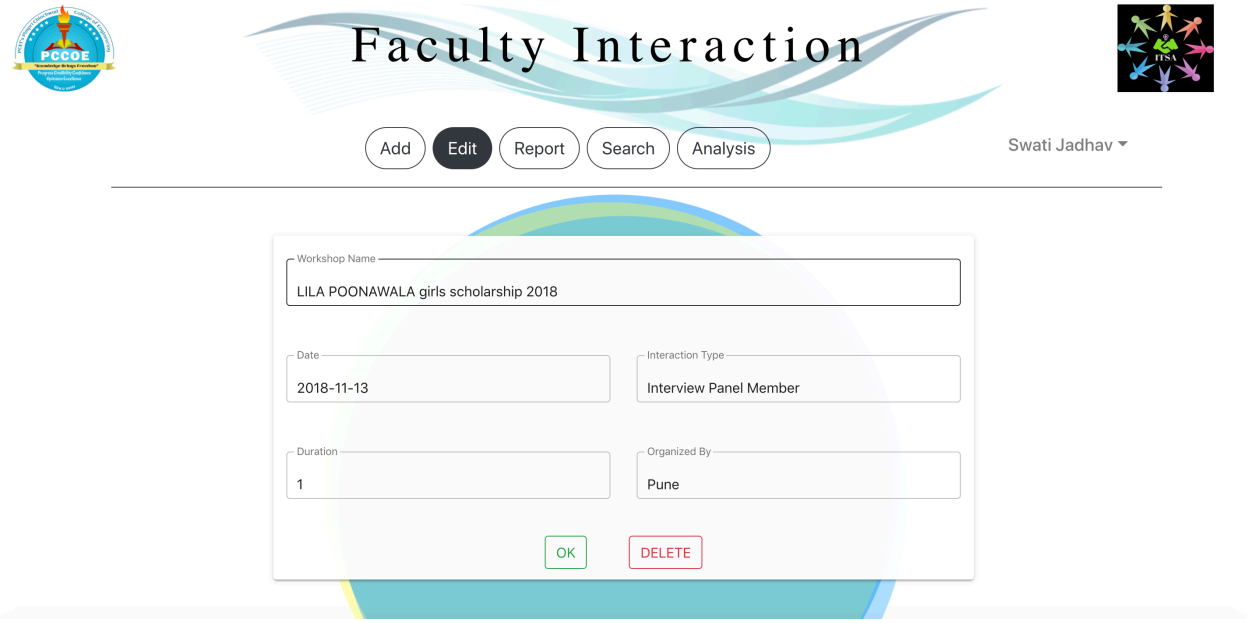
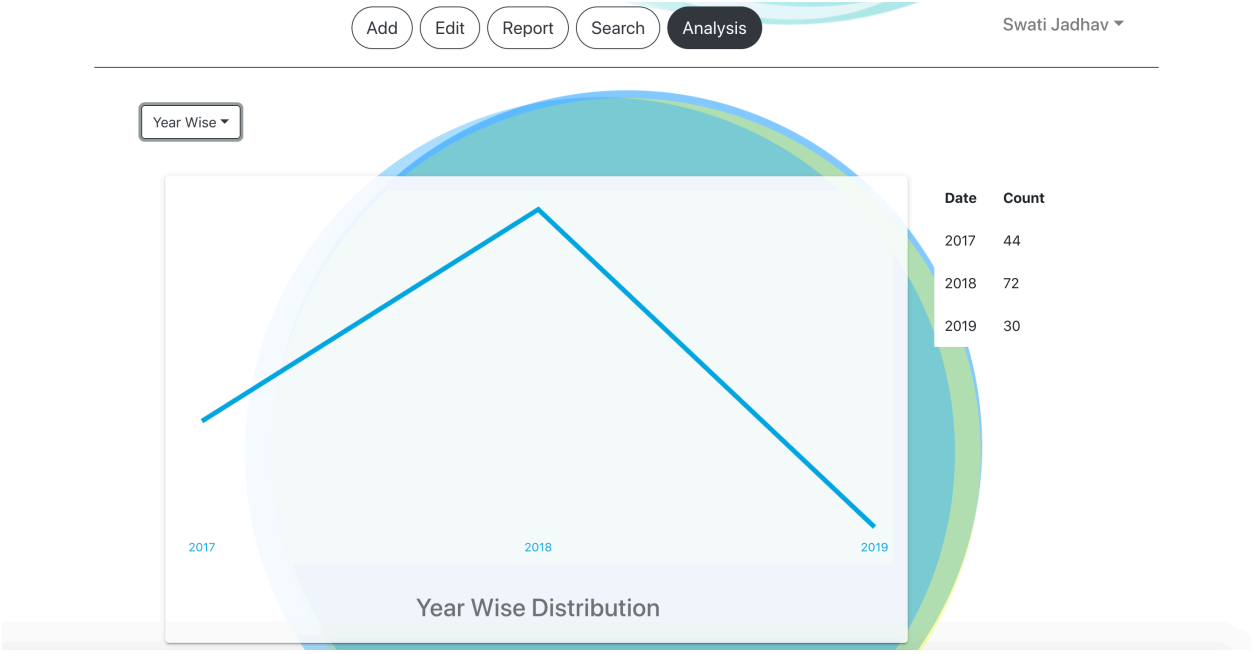


Figure 7.3. Analysis



Chapter 8

Conclusion

In this project, we have created a system for the department of Information Technology of Pimpri Chinchwad College of Engineering to manage the data of interaction of faculty with the outside world. The database management system uses MySQL at the backend to store data. The GUI is developed in HTML, CSS Bootstrap and Vue to make a web app. Faculty's data like name, type of interaction, duration and the location where it was conducted is stored in the database. The system allows referencing of data from different tables in the database.

The implementation of this project will make the department's faculty data management easier. It will reduce paperwork as well as manual errors during data handling.

References

Links-

- https://www.youtube.com/watch?v=SK98ayjhk1E //database items in listview
- https://www.youtube.com/watch?v=28jA5-mO8K8 //spinners
- https://www.youtube.com/watch?v=8vPtePCFzo8 //loading spinner values from database
- https://www.youtube.com/watch?v=a-dvLs1g3Ec //using checkbox and listview
- https://stackoverflow.com
- https://dzone.com/articles/list-editable-textboxes //editable text boxes in lists
- https://www.vuejs.org/