

# **AUTOMATION OF FACULTY ACTIVITIES**

**BY**

**K. L. Sai Kumar (18331A0568)**

**P. Ganesh(18331A05B4)**

**M. Varsha(18331A0592)**

**N. Usha Avanthi(19335A0512)**

*Under the esteemed guidance of*

***K. SANTOSH JHANSI Madam***



**Department of Computer Science and Engineering**

**MAHARAJ VIJAYARAM GAJAPATHI RAJ COLLEGE OF ENGINEERING (A)**

**VIZIANAGARAM**

## **ABSTRACT:**

Faculty are investing their efficient time in various activities such as publishing many of their works such as papers, patents, and receiving awards, achievements. Faculty are attending conferences, seminars, completing Ph.D.'s during their time at the college, publishing books, and many more.

Departments conduct workshops, seminars, conferences. The above and many more activities and their chronological records are not available in a central database, no automation for the Queries to retrieve departmental activity data exists as of now. There is no means for each individual faculty to update their activities in a central database.

## **PROBLEM STATEMENT:**

We are proposing a platform which automates the departmental activities of the faculty. The platform features a Central Database which holds all the refined collection of activities. It would also provide Interface for faculties to interact with the data base to update their activities in real time. Users will apply prebuilt queries to retrieve summaries and can generate valuable status reports.

## **Solution**

A activity automation software application provides user with services which help him to update activities to a central Database, and allows user to generate reports based on user requirements.

- A Central Database which reflects all the departmental activities is created
- The mobile application will reduce the effort for department to maintain every faculty details since each faculty will be given opportunity and access to ADD/EDIT their content.
- The Database access API leverages secure access to the data and gives better way to do the CRUD operation on the Central DB.
- The Web application for admin allow to run custom queries and generate reports which are useful to submit to NAAC,NBA etc. and many other committees.

# **SRS DOCUMENTATION**

## **INTRODUCTION**

A restaurant or any business firm near an isolate location draw less customers. Where restaurant near a shopping mall or a movie theatre draws more customer's . Our goal of this report is to examine the present data on the list of neighbor-hoods and venues in the city of Vizag. We are concerned with the presence of an Indian restaurant in our neighbor-hood. We have to find correlation of the Indian restaurant with the other Venue. We should train a model that predicts the possibility and presence of an Indian restaurant in that neighbor-hood.

## **PURPOSE**

**Purpose:** The purpose to design this model is to give a clear view to the entrepreneur, where to start his new firm. It suggests the best surrounding which make his firm more flexible to grow up.

**Intended Audience:** The users mostly we expect to use this model were Faculty who needs a platform to store their activities and admin users to generate reports for review committees.

**Scope:** Currently we are offering platform services for CSE department, so further we can extend our services to other departments and to the entire college.

## **Definition, Acronyms, Abbreviation:**

JAVA platform independence

SQL> Structured query Language

DFD > Data Flow Diagram

CFD Context Flow Diagram

ER -> Entity Relationship

IDE >Integrated Development Environment SRS Software Requirement Specification.

## **HARDWARE & SOFTWARE REQUIREMENT**

### **Hardware:**

Central Database:

- Unix Server to run Database with storage

Mobile Application:

- Android or IOS Device to Access the application

API or Admin web Interface:

- Unix Server to run API (python or php)

Web application:

- Chrome, Mozilla Firefox, Microsoft Edge Installed system with basic Intel Pentium Processor

### **Run time Environment Software:**

Windows or Linux system to access a web browser(chrome or edge or firefox) Installed.

Android KitKat and above for Android Users

IOS 10 and above for IOS Users

MySQL, phpMyAdmin, php, python, flutter, android java, iOS swift.

Python

### **Application Development Software:**

- Android Studio Bumblebee | 2021.1.1 for Windows 64-bit (872 MiB)
- phpMyAdmin 5.1.0 or less
- flutter plugin, dart
- Java 8 or above
- Swift
- Python Django
- Visual studio code

## **FUNCTIONAL REQUIREMENTS**

- **USER 1: FACULTY**

**FR 1: FACULTY REGISTRATION**

INPUT: Details of the faculty like ID, Name, Phone Number, E-mail, New Password.

PROCESSING:

- i. The system validates the existing data with the entered details of the faculty.

OUTPUT: Account creation status is shown.

## **FR 2: LOGIN**

INPUT: Username, Password.

PROCESSING:

- i. The faculty can login only if valid credentials are provided.
- ii. An error message is displayed if incorrect details are given.

OUTPUT: The faculty gains access over the database after a successful login.

## **FR 3: SHOW ACTIVITY**

INPUT: Faculty ID

PROCESSING: Queries Data Base for all activities.

OUTPUT: Displays faculty data in a chronological order.

## **FR 4: ADD EVENTS**

INPUT: Event type, Event Details.

PROCESSING: Sends the list of data to server and data is reflected in Data Base.

OUTPUT: Displays the status updation.

## **FR 5: VIEW**

INPUT: Department

PROCESSING: Retrieves all activity data corresponding to the department.

OUTPUT: Displays department data to users.

## **FR 6: EDIT**

INPUT: New User Data.

PROCESSING: Update user data in Data Base.

OUTPUT: Displays updated user profile.

## **FR 7: SEARCH EVENTS**

INPUT: Filter by year and event category

PROCESSING: Queries data for all events related to selected year and event type/

OUTPUT: Lists out the events.

- **USER 2: ADMIN**

- I. FR 1: ACCOUNT LOGIN**

INPUT: Username and Password

PROCESSING:

- ii. The admin can login only if valid username and password are provided.
    - iii. An error message is displayed if incorrect details are entered.

OUTPUT: Gains access over the Data Base after a successful login.

- FR 2: UPDATE ACTIVITIES**

INPUT: Username, Password.

PROCESSING:

OUTPUT:

- FR 3: WRITE QUERIES**

INPUT:

PROCESSING:

OUTPUT:

- FR 4: PROCESS QUERIES**

INPUT:

PROCESSING:

OUTPUT:

- FR 5: REPORTS**

INPUT:

PROCESSING:

OUTPUT:

- FR 6: DEPARTMENT DASHBOARD**

INPUT:

PROCESSING:

OUTPUT:

## **FR 7: FACULTY PROFILE**

INPUT:

PROCESSING:

OUTPUT:

## **FR 8: PERFORM ANALYSIS**

INPUT:

PROCESSING:

OUTPUT:

## **FR 9: WEB SCRAPING**

INPUT:

PROCESSING:

OUTPUT:

(TO BE CONTINUED)

## **Non-Functional Requirements**

### **Availability Requirement**

The system is available 100% for the user and is used 24 hrs a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.

### **Efficiency Requirement**

Mean Time to Repair (MTTR) - Even if the system fails, the system will be recovered back up within an hour or less.

### **Accuracy**

The system should accurately provide real time information taking into consideration various concurrency issues. The system shall provide 100% access reliability.

### **Performance Requirement**

The information is refreshed depending upon whether some updates have occurred or not in the application. The system shall respond to the member in not less than two seconds from the time of the request submittal. The system shall be allowed to take more time when

doing large processing jobs. Responses to view information shall take no longer than 5 seconds to appear on the screen.

### **Reliability Requirement**

The system has to be 100% reliable due to the importance of data and the damages that can be caused by incorrect or incomplete data. The system will run 7 days a week, 24 hours a day.

### **CONCLUSION:**

The platform saves time to maintain a central repository of faculty activity data. And provides efficient ways to extract valuable insights.



## PROJECT PLAN:

S.NO	STAGES	DURATION
1.	Requirements Gathering and Literature Survey	2 WEEKS
2.	High-level and detailed design	2 WEEKS
3.	Coding	14 WEEKS
4.	Result Analysis	2 WEEKS
5.	Documentation	2 WEEKS



**SUPERVISOR:** K. Santosh Jhansi

## PROJECT TEAM:

K. Lavanya Sai Kumar(18331A0568)

*K. L. Sai Kumar*

M. Varsha(18331A0592)

*M. Varsha*

P. Ganesh(18331A05B4)

*@Ganesh..*

N. Usha Avanthi(19335A0512)

*N. Usha Avanthi*