**A MINOR PROJECT REPORT ON**

**EVENT REGISTRATION &**

**VERIFICATION USING QR CODE**

*Submitted in partial fulfilment of the requirements*

*for the award of the degree of*

**BACHELOR OF COMPUTER APPLICATIONS**

*To*

**Guru Gobind Singh Indraprastha University, Delhi**

****

***Under the Guidance of: Submitted by:***

Dr. Brahampal Singh N. Krishna Khanth

BCA-V Sem

02220602018

Sharad Jain

BCA-V Sem

04020602018



**Session 2018 – 2021**

**TRINITY INSTITUTE OF PROFESSIONAL STUDIES**

(Affiliated to Guru Gobind Singh Indraprastha University, Delhi)

Ranked “A+” Institution by SFRC, Govt. of NCT of India

Recognized under section 2(f) of the UGC Act, 1956

NAAC Accredited “B++” Grade Institution



**TRINITY INSTITUTE OF PROFESSIONAL STUDIES**

(Affiliated to Guru Gobind Singh Indraprastha University, Delhi)

Ranked “A+” Institution by SFRC, Govt. of NCT of India

Recognized under section 2(f) of the UGC Act, 1956

NAAC Accredited “B++” Grade Institution

**To Whom It May Concern**

I **N. Krishna Khanth**, Enrolment No. **02220602018** from BCA-V Sem of the Trinity Institute of Professional Studies, Delhi hereby declare that the Minor Project Report entitled **“Event Registration & Verification Using QR Code”** at **Trinity Institute Of Professional Studies** is an original work and the same has not been submitted to any other Institute for the award of any other degree.

Date: Signature of the Student

Certified that the Project Report submitted in partial fulfilment of Bachelor of Computer Applications (BCA) to be awarded by G.G.S.I.P. University, Delhi by **N. Krishna Khanth**, Enrolment No. **02220602018** has been completed under my guidance and is Satisfactory.

Date: Signature of the Guide

Name of the Guide: Dr. Brahampal Singh

Designation: Associate Professor



**TRINITY INSTITUTE OF PROFESSIONAL STUDIES**

(Affiliated to Guru Gobind Singh Indraprastha University, Delhi)

Ranked “A+” Institution by SFRC, Govt. of NCT of India

Recognized under section 2(f) of the UGC Act, 1956

NAAC Accredited “B++” Grade Institution

**To Whom It May Concern**

I **Sharad Jain**, Enrolment No. **04020602018** from BCA-V Sem of the Trinity Institute of Professional Studies, Delhi hereby declare that the Minor Project Report entitled **“Event Registration & Verification Using QR Code”** at **Trinity Institute Of Professional Studies** is an original work and the same has not been submitted to any other Institute for the award of any other degree.

Date: Signature of the Student

Certified that the Project Report submitted in partial fulfilment of Bachelor of Computer Applications (BCA) to be awarded by G.G.S.I.P. University, Delhi by **Sharad Jain**, Enrolment No. **04020602018** has been completed under my guidance and is Satisfactory.

Date: Signature of the Guide

Name of the Guide: Dr. Brahampal Singh

Designation: Associate Professor

# Acknowledgement

We would like to take this opportunity to thank The Guru Gobind Singh Indraprastha University (GGSIPU) for having projects as a part of the BCA curriculum. Many people at the university have influenced the shape and content of this project and supported us through it.

We express our sincere gratitude to Dr. Brahampal Singh for mentoring us in this project. He has been an inspiration and role model for this topic. His guidance and active support have made it possible for us to work on the assignment.

We would like to extend our thanks to the members involved in making this project successful and without whom this project would not have been possible. Also, we would like to express sincere gratitude to our parents and a load of thanks to our friends for helping us throughout the course of the project.

**N. Krishna Khanth (02220602018)**

**Sharad Jain (04020602018)**

# Abstract

Every institution organizes events and meetings. It is a major task to organize participants, manage reports, registrations, messages, make groups, collect fees etc. There are multiple societies and clubs in every institution and all of them organize a variety of events throughout the year, sometimes at the same time. Their publicity messages and posts are often ignored or lost in the sea. Further, the hassle of making groups or participants, arranging meetings to get payment, circulating messages for any changes, reminding participants etc., often leads to chaos and confusion.

We offer a solution which would prevent the struggle of managing participants and reducing the work force required, making it possible to focus their attention on other matters.

The purpose of this application is to automate existing manual system with the help of computerized equipment and full-fledged software.

This report discusses the project developed and implemented by us along with the challenges we faced during the development period of the project.

**Table of Contents**

[Acknowledgement iv](#_Toc55941371)

[Abstract v](#_Toc55941372)

[Table of Figures vii](#_Toc55941373)

[List of Tables viii](#_Toc55941374)

[List of Abbreviations viii](#_Toc55941375)

[1. Introduction 1](#_Toc55941376)

[1.1 Idea and Purpose 2](#_Toc55941377)

[1.2 Problem Definition 2](#_Toc55941378)

[1.3 Existing Methods 2](#_Toc55941379)

[1.4 Motivation 2](#_Toc55941380)

[1.5 Suggested Solutions 2](#_Toc55941381)

[2. System Requirement Analysis 1](#_Toc55941382)

[2.1 Project Plan 2](#_Toc55941383)

[2.2 Software Requirement Specifications 2](#_Toc55941384)

[2.3 Tools and Technologies 2](#_Toc55941385)

[2.4 Hardware Requirements 2](#_Toc55941386)

[2.5 Software Requirements 2](#_Toc55941387)

[3. System Feasibility Study 1](#_Toc55941388)

[3.1 Feasibility Study 2](#_Toc55941389)

[4. System Design 1](#_Toc55941390)

[4.1 Use Case 2](#_Toc55941391)

[4.2 Flow Chart 2](#_Toc55941392)

[4.3 Entity Relationship Diagram: 2](#_Toc55941393)

[4.4 Database Tables Diagram: 2](#_Toc55941394)

[4.5 Data Flow Diagram (DFD): 2](#_Toc55941395)

[5. System Development 1](#_Toc55941396)

[4.1 Frontend 1](#_Toc55941397)

[4.2 Backend 1](#_Toc55941398)

[4.3 Database 1](#_Toc55941399)

[6. System Implementation 2](#_Toc55941400)

[7. System Testing 3](#_Toc55941401)

[8. Future Scope 4](#_Toc55941402)

[References 4](#_Toc55941403)

# Table of Figures

[Figure 1: Use Case Diagram 15](#_Toc55860568)

[Figure 2: Flow Chart Diagram 24](#_Toc55860569)

[Figure 3: Entity Relationship Diagram 25](#_Toc55860570)

[Figure 4: Entity Relationship Table Diagram 26](#_Toc55860571)

[Figure 5: DFD Level 0 Diagram 27](#_Toc55860572)

[Figure 6: DFD Level 1 Diagram 28](#_Toc55860573)

[Figure 7: DFD Level 2 User Account Information Management 29](#_Toc55860574)

[Figure 8: DFD Level 2 Login 29](#_Toc55860575)

[Figure 9: DFD Level 2 Participant Details Management 30](#_Toc55860576)

[Figure 10: DFD Level 2 Event Details Management 30](#_Toc55860577)

[Figure 11: DFD Level 2 Event Participation Information Management 31](#_Toc55860578)

# List of Tables

[Table 1: List of Libraries 10](#_Toc55847951)

# List of Abbreviations

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Abbreviated Name** | **Full Name** |
| 1. | ERVQR | Event Registration and Verification using QR (Project Title) |
| 2. | QR | Quick Response |
| 3. | TIPS | Trinity Institute of Professional Studies |
| 4. | SRS | Software Requirement Specification |
| 5. | ID | Identification |
| 6. | DFD | Data Flow Diagram |
| 7. | ER | Entity Relationship |

# 1. Introduction

Every organization, big or small, organizes events and has challenges to overcome, people, and operations to manage smoothly. From attendees to the location to the ambiance, there are many different areas in need of careful management. This application is being developed to override the problems prevailing the present manual systems. This system is being designed, keeping in mind the event requirements. This will reduce the struggle of maintaining groups, records of participants and payments, etc. Moreover, it will provide all the features with a user-friendly interface. The collection will be understandable and straightforward.

We offer a solution which would prevent the struggle of managing participants and reducing the work force required, making it possible to focus their attention on other matters. The purpose of this application is to automate existing manual system with the help of computerized equipment and full-fledged software.

This report discusses the project developed and implemented by us along with the challenges we faced during the development period of the project.

## 1.1 Idea and Purpose

We suggest an event particular app which will act as a common platform for serving different societies and clubs of an organization for easy and effective event management. This application will bridge the gap between the organizers and the attendees. The application will reduce the manual work significantly and allow easy management of participation.

## 1.2 Problem Definition

During our fest, workshop, seminars, society events it was observed that there were lot of coordination problems and everything was being handled manually. There was no proper system to handle the incoming participation, event information management and report management.

Also, the manual handling of events led to a lot of disturbance in the classrooms as the students would request to leave the class for organising events, making announcements, designing posters, attending meetings etc. This would also disturb the faculty and thus they would not support the missing of the classes. Even the society coordinators would find it difficult to manage multiple events, participation.

## 1.3 Existing Methods

Presently, the following methods are used to manage publicity and participations:

1. Flooding messages on WhatsApp and making groups of participants on WhatsApp to circulate messages require recording contact numbers of everyone and making sure that they read the previous messages.
2. Repetitive announcements in classrooms – often not possible to make in all classes.
3. Posters on notice boards – cluttering noticeboards with multiple posters, important notices go unnoticed.
4. Arranging meetings with the organizers to give payments – unreliable and causes confusion as people are often in different places.

## 1.4 Motivation

This system application’s motivation arose due to the current manual system problems, some of which are:

1. Very narrow span of attention towards long messages on WhatsApp.
2. Infrequent announcements disturb the environment of classes.
3. Wastage of resources in the form of paper and personnel.
4. Very time-consuming.
5. Difficulty in steady coordination and scheduling of different activities.

As participants and organizers, we feel that these problems cause confusion and a bad reputation.

## 1.5 Suggested Solutions

We suggest the following solutions to the above-discussed problems:

1. A common platform for organizers to handle records of the event.
2. Auto-ticket generation (QR).
3. A central database for storage of records.

# 2. System Requirement Analysis

## 2.1 Project Plan

The project management plan has been broken down in the following parts:

1. Stakeholders and Expectations:

Technical Team: Have ready access to individuals with the authority to make decisions regarding events.

Project Manager: Have an application in the form of a desktop application.

Client: Gain an application which event organizers can use easily and enhance their customer’s experience.

1. Project Priorities and Degrees of Freedom:

The main focus of the project is to develop an event management system on time and within budget. Further we aim to conduct a comprehensive testing of the system to detect bugs and defects. Another priority area is proper allocation of resources and team members to the various roles and responsibilities that arise during the software development.

1. Approach

We will be following an Incremental approach for this project. The first iteration will focus on basic functionality of the application and subsequent iterations will depend upon that and incorporate more features as time allows based on their priority and importance to the whole application.

1. Assumptions

* Server terminals are available and functional when needed.
* Feasibility of interfacing the ERVQR application with the other applications.
* The professional societies and clubs will adopt this desktop application meant for smooth running of their events.

1. Success Criteria

The project will be considered a success if the team delivers an operational prototype at the end of the semester with the earlier mentioned features.

1. Risks and Obstacles to Success

The risks and hurdles that might occur during the development of the portal and operations of the application include-net speed, network connectivity issues (because our system depends on a reliable and fast connected network to operate), a secure and trusted medium of communication between users and societies and other risks including improper use of the server, power failures etc.

1. Scope

This project will consist of creating an application of event management based upon the cultural, technical societies and student chapters of different college/universities. The project will:

* Allow students to register for events and societies.
* Allow societies to create and manage events and their registrations.
* Allow students to enter an event with just the use of QR code.
* Allow administrator to manage societies.

## 2.2 Software Requirement Specifications

Software Requirement Specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements and may include a set of use cases that describe user interactions that the software must provide. It establishes the basis for an agreement between customers and the software providers on what the software product is to do and what it is not expected to do so that there is no room for confusion in the future. If used appropriately, SRS can help prevent software project failure.

1. **Functional Requirements**

Our proposed system has the following requirements:

1. The system requires storing the information about a new participant being registered and, most notably, the event and its organizers.
2. The system needs to help the internal staff manage the entries of the database and keep information on activities.
3. The system needs to update, delete, and modify the records.
4. The system needs to maintain quality and quantity records.
5. The system requires the verification and authentication process of users.
6. The system needs to provide a QR code as identification to each participant at the time of registration.
7. The system should be able to provide as easy to read record of all participants and events.
8. Every participant should be able to enter an event just by showing the QR code that was provided at the time of registration.
9. System needs to maintain security to prevent unauthorized modification of data.
10. **Non-Functional Requirements**

Performance Requirements

* The system needs to be reliable.
* If unable to process the request then appropriate error message.
* Web pages are loaded within few seconds.

Security Requirements

* After entering the password and user-id the user can access his profile.
* The details of user must be safe and secure.
* Sharing of details.
* Unauthorized modification of records must be prevented.

Safety Requirements

* The details need to be maintained properly.
* Users must be authenticated.

## 2.3 Tools and Technologies

The following tools and technologies are expected to be used in development. Further may be added as the operations are implemented.

1. **Development Language**
2. *Python 3.8*

Python is a general-purpose programming language that can be used on any modern computer operating system. It can be used for processing text, numbers, images, scientific data and just about anything else you might save on a computer. It is used daily in the operations of the Google search engine, the video-sharing website YouTube, NASA and the New York Stock Exchange. These are but a few of the places where Python plays important roles in the success of the business, government, and non-profit organizations; there are many others.

1. *Pip 20.2.4*

pip is a de facto standard package-management system used to install and manage software packages written in Python. Many packages can be found in the default source for packages and their dependencies — Python Package Index. Most distributions of Python come with pip preinstalled.

1. **Development Platforms**
2. *PyCharm 2020.20*

PyCharm is an Integrated Development Environment (IDE) used for programming in Python. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django.

1. *Atom 1.52.0 x64*

Atom is a free and open-source text and source code editor for macOS, Linux, and Microsoft Windows with support for plug-ins written in Node.js, and embedded Git Control, developed by GitHub. Atom is a desktop application built using web technologies. Most of the extending packages have free software licenses and are community-built and maintained. Atom is based on Electron (formerly known as Atom Shell), a framework that enables cross-platform desktop applications using Chromium and Node.js. It is written in CoffeeScript and Less.

1. *MySQL Workbench 8 CE*

MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modelling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more. MySQL Workbench is available on Windows, Linux and Mac OS X.

1. **Database**
2. *MySQL 8.0.19*

MySQL is a freely available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing content in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use. MySQL is an essential part of almost every open source PHP application. Good examples for PHP & MySQL-based scripts are WordPress, Joomla! and Drupal.

1. *Excel*

Microsoft Excel is a spreadsheet developed by Microsoft for Windows, macOS, Android and iOS. It features calculation, graphing tools, pivot tables, and a macro programming language called Visual Basic for Applications. It has been a very widely applied spreadsheet for these platforms, especially since version 5 in 1993, and it has replaced Lotus 1-2-3 as the industry standard for spreadsheets. Excel forms part of the Microsoft Office suite of software.

1. **Hosting Service**
2. *PythonAnywhere*

PythonAnywhere is an online integrated development environment (IDE) and web hosting service based on the Python programming language. It provides in-browser access to server-based Python and Bash command-line interfaces, along with a code editor with syntax highlighting. Program files can be transferred to and from the service using the user's browser. Web applications hosted by the service can be written using any WSGI-based application framework.

1. **Libraries**

|  |  |
| --- | --- |
| **Library** | **Version** |
| certifi | 2020.11.8 |
| chardet | 3.0.4 |
| click | 7.1.2 |
| et-xmlfile | 1.0.1 |
| Flask | 1.1.2 |
| Flask-Cors | 3.0.9 |
| idna | 2.10 |
| itsdangerous | 1.1.0 |
| jdcal | 1.4.1 |
| Jinja2 | 2.11.2 |
| MarkupSafe | 1.1.1 |
| mysql-connector-python | 8.0.22 |
| numpy | 1.19.3 |
| opencv-contrib-python | 4.4.0.46 |
| opencv-python | 4.4.0.46 |
| openpyxl | 3.0.5 |
| Pillow | 8.0.1 |
| protobuf | 3.13.0 |
| PyQRCode | 1.2.1 |
| pyzbar | 0.1.8 |
| requests | 2.24.0 |
| six | 1.15.0 |
| urllib3 | 1.25.11 |
| Werkzeug | 1.0.1 |
| Tkinter | 8.6 |
| re | 2.2.1 |
| hashlib | 20081119 |
| datetime | 4.3 |

Table 1: List of Libraries

## 2.4 Hardware Requirements

The following hardware requirements are recommended to be fulfilled in order to run this software.

1. CPU: Intel core i3 3rd Generation / AMD FX-6100
2. RAM: 2 GB
3. GPU: Integrated Graphics
4. Storage: 1 GB
5. Camera: Any 3MP camera

## 2.5 Software Requirements

The following software requirements are recommended to be fulfilled in order to run this software.

1. OS: Any Operating System
2. Database: MySQL
3. Programming Language: Python

# 3. System Feasibility Study

## 3.1 Feasibility Study

An Assessment of the feasibility of the project.

1. **Economic Feasibility**

The project is economically feasible as it works with functions with low-cost services such as laptops and desktops.

1. **Technical Feasibility**

The current project is technically feasible as the application requires:

1. Any python supported IDE
2. Server-Side Services
3. GUI development tools

All these are readily available and can be successfully deployed on any available computer.

1. **Behavioural Feasibility**

The application is behaviourally feasible since it requires no technical guidance; all the modules are user friendly.

1. **Operational Feasibility**

The application is operationally feasible as:

1. Complete GUI-Base, which is user friendly.
2. Inputs to be taken are self-explanatory.
3. The system cuts down the load and cost of clients by high margins.

# 4. System Design

## 4.1 Use Case

Use Case Model

The Use Case Model describes the proposed functionality of our system. The diagram represents a discrete unit of interaction between the user (society or a new member) and the website. This interaction is a single unit of meaningful work, such as Create Event or View Event Details.

The project’s use case model consists of the following type of actors:

* Main Actors – Participant
* Supporting Actors – Admin and User

The primary modules (different functions) are:

* Maintain User Details
* Maintain Event Details
* Maintain Participant Details
* Quick Response (QR) Code

The model is given below:

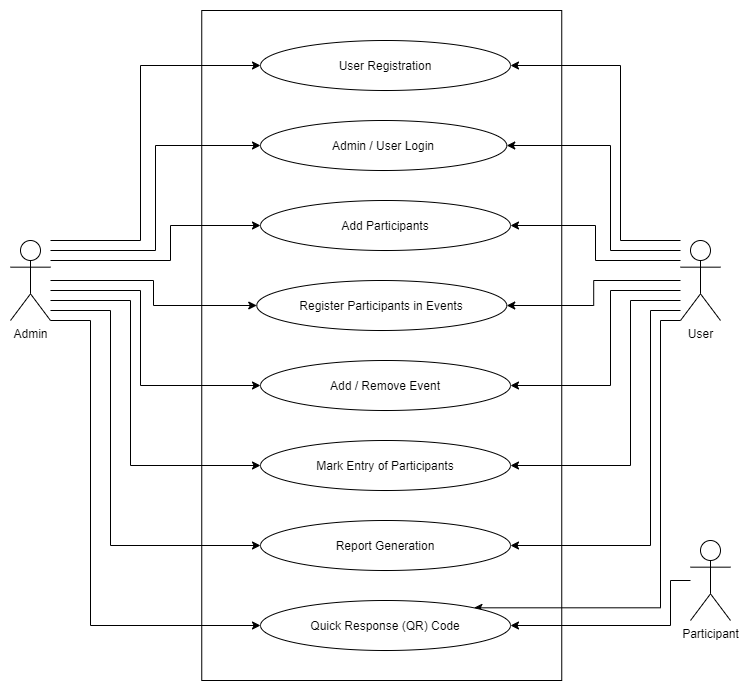


Figure 1: Use Case Diagram

The project has the following use case specifications:

1. **Use Case: User Registration**

**Description:** This use case takes care of registration of users (Organizers) in the MySQL database and gives them access to the application by signing in using their user id (email) and password.

Note: Only Administrator is allowed to add new users.

**Primary Flow:**

1. User information is entered by an administrator.
2. The information is sent to the database to register the user.
3. The user provides his email and password.
4. The user is authenticated.
5. The new user can access the application.

**Alternate Flow at:** The user is already registered.

1. The administrator is warned that the email is already registered.
2. The administrator may register another user or retry with another email id.

**Error Flow E1:** The user credentials are wrong.

1. The administrator is told to recheck the credentials.
2. The administrator may try again.

**Error Flow E2:** The server or internet is not working.

1. The administrator is told to recheck internet connection.
2. The administrator may try again.
3. **Use Case: Admin / User Login**

**Description:** This use case takes care of login of users (Organizers), check their credentials in MySQL database and gives them access to the application by signing in using their user id (email) and password.

**Primary Flow:**

1. User enters their email id and password.
2. The information is sent to the database.
3. The database returns the permission level.
4. The user is authenticated.
5. The user has access to the application.

**Error Flow E1:** The user credentials are wrong.

1. The user is told to recheck the credentials.
2. The user may try again.

**Error Flow E2:** The server or internet is not working.

1. The user is told to recheck internet connection.
2. The user may try again.
3. **Use Case: Add Participants**

**Description:** This use case takes care of registration of participants (customers/clients of user) in the MySQL database and gives them a unique QR code as participant ID.

**Primary Flow:**

1. Participant provides his/her information and events.
2. Participant information is entered by user.
3. The information is sent to the database to register the participant.
4. The participant is given a QR code.
5. The participant is registered.

**Alternate Flow 1:** The Participant is already registered in all given events.

1. The user is warned that the participant is already registered.
2. The user informs the participant.

**Alternate Flow 2:** The Participant is already registered in any 1 of 2 given events.

1. The user is warned that the participant is already registered in one of the events and registration is complete for the other event.
2. The user informs the participant.

**Error Flow E1:** The participant information is wrong.

1. The user is told to recheck the information.
2. The user may try again.

**Error Flow E2:** The server or internet is not working.

1. The user is told to recheck internet connection.
2. The user may try again.
3. **Use Case: Register Participant in Events**

**Description:** This use case takes care of registration of already registered participants (customers/clients of user) in events in the MySQL database.

**Primary Flow:**

1. Participant provides QR code and events.
2. Participant information is entered by user.
3. The information is sent to the database.
4. The participant is registered in events provided.

**Alternate Flow 1:** The Participant is already registered in all given events.

1. The user is warned that the participant is already registered.
2. The user informs the participant.

**Alternate Flow 2:** The Participant is already registered in any 1 of 2 given events.

1. The user is warned that the participant is already registered in one of the events and registration is complete for the other event.
2. The user informs the participant.

**Error Flow E1:** The participant QR code is wrong.

1. The user is told to recheck the QR code.
2. The user may try again.

**Error Flow E2:** The server or internet is not working.

1. The user is told to recheck internet connection.
2. The user may try again.
3. **Use Case: Add / Remove Events**

**Description:** This use case takes care of adding and removing events from the MySQL database.

**Primary Flow:**

1. User enters event information.
2. The information is sent to the database.
3. Event is added in database.

**Alternate Flow 1:** The event is already registered.

1. The user is warned that the event is already registered.
2. The user may try again.

**Error Flow E1:** The event information is wrong.

1. The user is told to recheck the event information.
2. The user may try again.

**Error Flow E2:** The server or internet is not working.

1. The user is told to recheck internet connection.
2. The user may try again.
3. **Use Case: Mark Entry of Participants**

**Description:** This use case takes care of marking entry participants (customers/clients of user) in events in the MySQL database. Participant are only provided entry once in an event.

**Primary Flow:**

1. Participant provides QR cod.
2. Participant QR code is entered by user.
3. The information is sent to the database.
4. The participant is provided entry in the event.

**Alternate Flow 1:** The Participant is already registered provided entry on that QR code.

1. The user is warned that the participant has already Entered.
2. Entry is denied to the participant.

**Error Flow E1:** The participant QR code is wrong.

1. The user is told to recheck the QR code.
2. The user may try again.

**Error Flow E2:** The server or internet is not working.

1. The user is told to recheck internet connection.
2. The user may try again.
3. **Use Case: Report Generation**

**Description:** This use case takes care of generation of report from the data in MySQL database. Report is generated for all participants along with events they are participating in.

**Primary Flow:**

1. User requests to make a report.
2. The request is sent to the database.
3. A report is generated and user is informed.

**Error Flow E1:** The server or internet is not working.

1. The user is told to recheck internet connection.
2. The user may try again.
3. **Use Case: Quick Response (QR) Code**

**Description:** This use case takes care of generation of QR code for participants (customers/clients of user) identification.

**Primary Flow:**

1. Participant is registered.
2. A QR code is generated for the user.
3. QR code is provided to participant.

**Alternate Flow 1:** Unable to register participant.

1. The user is warned that the participant is not registered.
2. No QR code is generated
3. The user informs the participant.

**Error Flow E1:** The server or internet is not working.

1. The user is told to recheck internet connection.
2. The user may try again.

## 4.2 Flow Chart

A flowchart is a picture of the separate steps of a process in sequential order. It is a generic tool that can be adapted for a wide variety of purposes, and can be used to describe various processes, such as a manufacturing process, an administrative or service process, or a project plan.

**Flow Chart Components:**

1. Process

rectangle - flowchart process step

Rectangle - One step in the process. The step is written inside the box. Usually, only one arrow goes out of the box.

1. Flowline (Arrowhead)

arrow - flowchart flow direction

Arrow - Direction of flow from one step or decision to another.

1. Decision

diamond - flowchart decision step

Diamond - Decision based on a question. The question is written in the diamond. More than one arrow goes out of the diamond, each one showing the direction the process takes for a given answer to the question. (Often the answers are "yes" and "no.")

1. Terminal



Circle or oval - Indicates the beginning and ending of a program or sub-process. Represented as a stadium, oval or rounded (fillet) rectangle. They usually contain the word "Start" or "End", or another phrase signalling the start or end of a process, such as "submit inquiry" or "receive product".

1. Input / Output



Shows a conditional operation that determines which one of the two paths the program will take. The operation is commonly a yes/no question or true/false test. Represented as a diamond.

1. Data File or Database

Flowchart database

Data represented by a cylinder (disk drive).

1. Internal Storage



File represents storage of a file in internal storage.

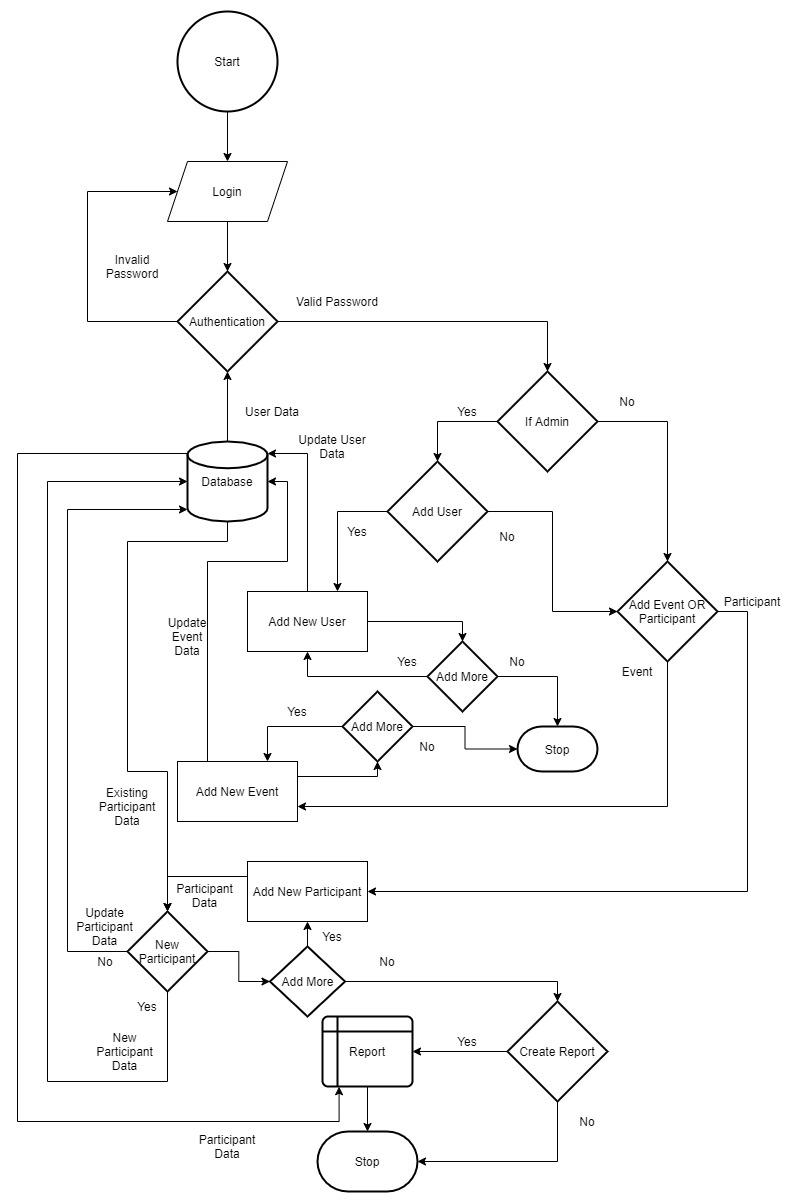


Figure 2: Flow Chart Diagram

## 4.3 Entity Relationship Diagram:

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.

**ER Diagram Components:**

1. Entity

rectangle - flowchart process step

Rectangle: An entity is an object or component of data. An entity is represented as rectangle in an ER diagram.

2. Key Attribute

Oval: An attribute describes the property of an entity. An attribute is represented as Oval in an ER diagram.

3. Relationship

diamond - flowchart decision step

Diamond: A relationship is represented by diamond shape in ER diagram, it shows the relationship among entities.

4. Flowline (Arrowhead)

arrow - flowchart flow direction

Arrow - Direction of relational flow between objects.

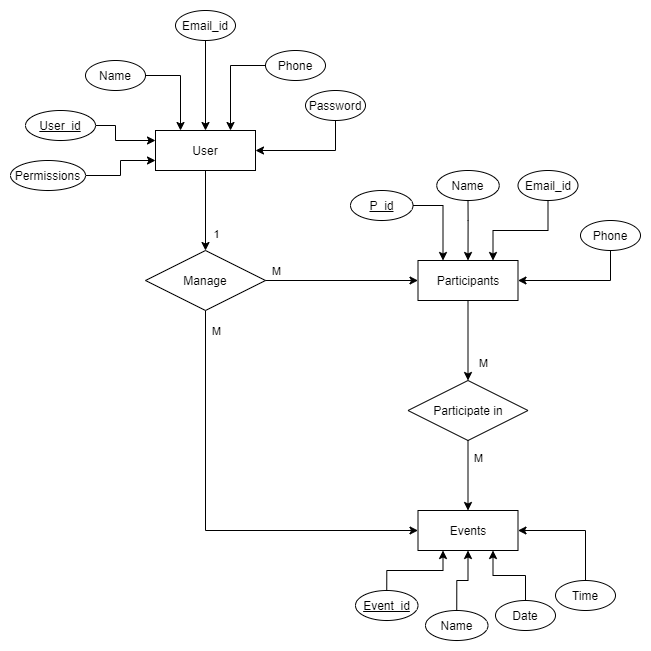


Figure 3: Entity Relationship Diagram

## Database Tables Diagram:

1. User table:

This table stores data of user (Admins & Organizers) login credentials.

It contains attributes like:

* The user ID attribute

It is the primary key of type int which cannot be left empty and hence it auto increments which contains ID of users.

* The username attribute

It is of type varchar which cannot be left empty and can contain complex usernames of users up to a length of 50 alphanumeric values.

* The email ID attribute

It is of type varchar which cannot be left empty and contains email ID of user to be used for verification/authentication during login.

* The phone attribute

It is of type varchar which cannot be left empty and can contain phone number of users from any region up to 15 digits.

* The password attribute
* It is of type varchar which cannot be left empty and contains password of users in an encrypted format.
* The permission attribute

It is of type int which cannot be left empty and contains the rights of the user to check if they are admin or normal user.

2. Participants table:

This table stores data of participants who have registered for various events.

It contains attributes like:

* The participant ID attribute

It is the primary key of type int which cannot be left empty and hence it auto increments which contains ID of participants.

* The name attribute

It is of type varchar which cannot be left empty and can contain names of participants up to a length of 50 alphanumeric values.

* The email ID attribute

It is of type varchar which cannot be left empty and contains email ID of participant for later use.

* The phone number attribute

It is of type varchar which cannot be left empty and can contain phone number of participants from any region up to 15 digits.

3. Events table:

This table stores information regarding events that will be organized in which participants can register.

It contains attributes like:

* The event ID attribute

It is the primary key of type int which cannot be left empty and hence it auto increments which contains ID of events being organized.

* The name attribute

It is of type varchar which cannot be left empty contains names of events being organized.

* The date attribute

It is of type date which cannot be left empty contains the date of the event to be organized.

* The time attribute

It is of type time which cannot be left empty contains the time of the event to be organized.

4. Registration table:

This table stores information of participant’s registration in one or more events and weather they have entered in that event or not.

It contains attributes like:

* The registration ID attribute

It is the primary key of type int which cannot be left empty and hence it auto increments which contains registration ID of relation between events and participants.

* The participant ID attribute

It is the first foreign key obtained from participant table to link participants to their registered events.

* The event ID attribute

It is the second foreign key obtained from events table to link participants to their registered events.

* The present attribute

It is of type int which cannot be left empty and contains the status of participant’s entry into registered events.

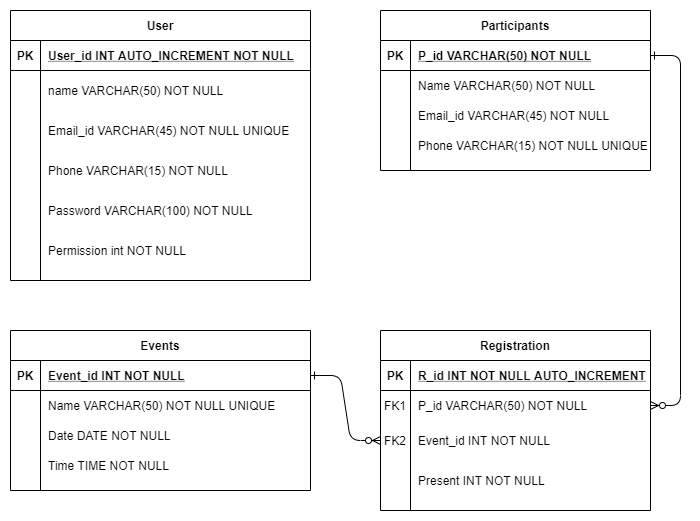


Figure 4: Entity Relationship Table Diagram

## 4.5 Data Flow Diagram (DFD):

**Level 0:**

Inputs:

1. User login

Credentials entered by user to login & access the software.

2. User data

Data provided by individuals to Admin to get login credentials.

3. Event data

Data added by user about events being organized.

4. Participant data

Data provided by participants to register themselves in various events.

Outputs:

1. Events

Information about the event being organized and participants registered in it.

2. Participants

Information about participant and the event they are registered in.

3. Entry

Information to & after validating a participant to enter an event.

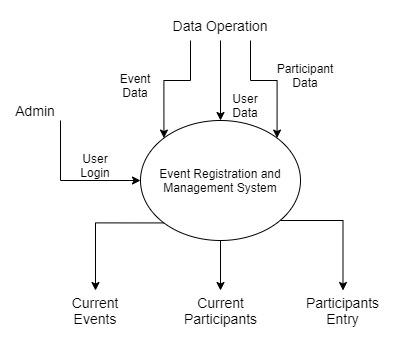


Figure 5: DFD Level 0 Diagram

**Level 1:**

1. Login

Takes user input and verifies, validates & authenticates and logs them in as Admin or regular user.

2. Event registration / QR Generation

The user enters the data provided by a participant to register them in any event of their choice and provides them QR code.

3. Event verification / QR Verification

The user validates and admits a participant by the QR code provided by them during the event day.

4. Event management

The user adds details of events to be organized during the first use so that participants can register in them.

5. Registry management

The data of participant is linked to the event in which they have been registered to create their entrance records.

6. Report

The user can generate a report of participant registry at any point of use which is in excel format.

7. Login as admin

If the credentials provided by user is that of an admin they have access to all the above abilities and to add more users of the software.

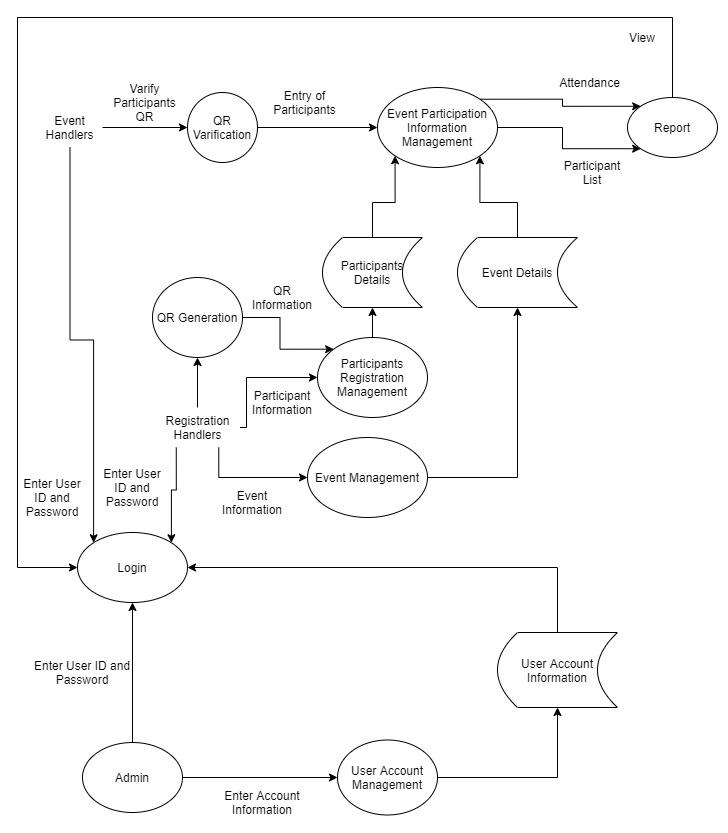


Figure 6: DFD Level 1 Diagram

**Level 2:**

User Account Information Management:

The admin manages:

* User account information,
* They can add more users, it can be more admins or handlers,
* They can update details of existing users,
* They can also remove existing users.

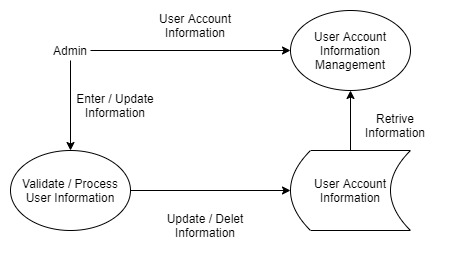


Figure 7: DFD Level 2 User Account Information Management

Login:

It validates user data against the data store to log in a user as:

* Admin
* Event handler
* Registration handler

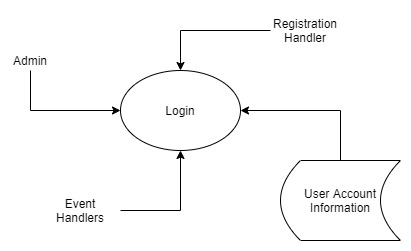


Figure 8: DFD Level 2 Login

Participant Details Management:

The registration handler manages:

* Participant data store
* Registering participants to events
* Providing QR code for participants
* Re-register existing participants to more events

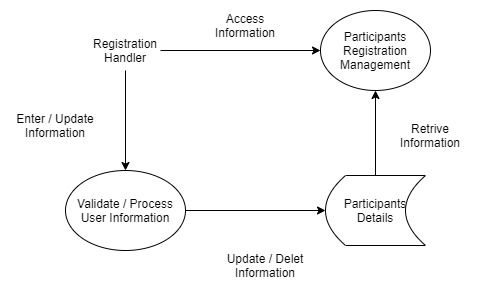


Figure 9: DFD Level 2 Participant Details Management

Event Details Management:

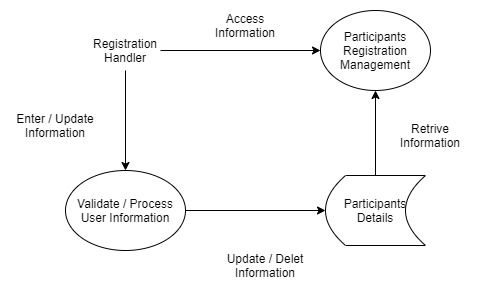


Figure 10: DFD Level 2 Event Details Management

Event Participation Information Management:

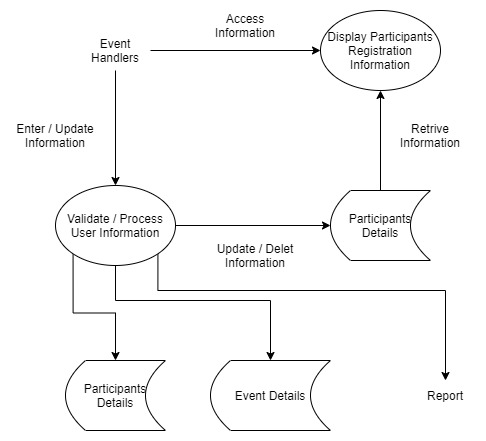


Figure 11: DFD Level 2 Event Participation Information Management

# 5. System Development

## 4.1 Frontend

4.1.1 Script “config.py”

4.1.2 Script “ERVQR.py”

4.1.3 Script “frontend\_api.py”

## 4.2 Backend

4.2.1 Script “backend\_api.py”

4.2.2 Script “manage\_op.py”

4.2.3 Script “db\_operations.py”

4.2.4 Script “database.py”

## 4.3 Database

* + 1. Database Script

# 6. System Implementation

# 7. System Testing

# 8. Future Scope

As of now, this project is a desktop application. The participants have to reach out to the registration desk to register themselves. In the near future, this application could be redesigned as a web-based application to make online registrations possible. This will reduce the waiting time of participants standing in line to register. This will make the registration process easier for the participant and encourage more people to participate.

# References

The application would provide a simple and better way to organize events and manage participation. We hope it would ease the hassle of arranging payments, making groups, and all the effort it takes to manage public operations.