ENTRYWAY BOOKING SYSTEM

A PROJECT REPORT

Submitted by

AMRUTHA BIJU

Reg.No: SJC21MCA-2003

To

the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree

of

MASTER OF COMPUTER APPLICATIONS

DEPARTMENT OF COMPUTER APPLICATIONS

ST. JOSEPH'S COLLEGE OF ENGINEERING AND TECHNOLOGY, PALAI
CHOONDACHERRY P.O, KOTTAYAM
KERALA

MAY,2023

ST. JOSEPH'S COLLEGE OF ENGINEERING & TECHNOLOGY, PALAI

CHOONDACHERRY P.O, KOTTAYAM KERALA

(*An ISO 9001-2015 Certified College*)

Department of Computer Applications

CERTIFICATE

*This is to certify that the report entitled "**Entryway Booking System**" submitted by "**AMRUTHA BIJU, Reg.No: SJC21MCA-2003**" to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications is a bonafide record of the project work carried out by her under our guidance and supervision. Thisreport in any form has not been submitted to any other University or Institute for any purpose.*

Mr. Anish Augustine k	**Dr. Rahul Shajan**	**Mr. Anish Augustine K**
(Project Guide)	**(Project Coordinator)**	**(HOD-MCA)**

Viva-voce held on: ……

External Examiner 1

External Examiner 2

DECLARATION

I undersigned hereby declare that the project report "**EntryWay Booking System**", submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of **Mr.Anish Augustine K, Head of the Department-Incharge of Computer Applications** This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violationof the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formedthe basis for the award of any degree, diploma or similar title of any other University.

Place : Choondacherry AMRUTHA BIJU

Date : SJC21MCA-2003

ACKNOWLEDGEMENT

The success of any project depends largely on the encouragement and guidelines of manyothers. I would like to take this opportunity to express our gratitude to those people who havebeen instrumental in the successful completion of this project.

First and foremost, I give all glory, honor and praise to **God Almighty** who gave me wisdom and enabled me to complete the project successfully. I also express sincere thanks, from the bottom of my heart, to my parents for their encouragement and support in all my endeavors and especially in this project.

I express my sincere gratitude to **Dr. V.P Devassia, Principal, SJCET, Palai** for givingme the provision to do the project in the required way. I extend my sincere gratitude to **Mr. Anish Augustine K, Head of the Department-Incharge of Computer Applications, SJCET, Palai** who has been a constant source of inspiration and without his tremendous help and support this project would not have been materialized.

I owe a particular debt of gratitude to our internal project guide **Mr. Anish Augustine K Head of the Department-Incharge of Computer Applications, SJCET, Palai** for all the necessary help and support that he has extend to me. His valuable suggestions, corrections and the sincere efforts to accomplish my project even under a tight time schedule were crucial in the successful completion this project.

I express my sincere thanks to all the faculties of the department of Computer Applications for their help and encouragement. I thank all my friends, for their inspiration and cooperation. Once again, I convey our gratitude to all those persons who had direct or indirect influence on my work.

Amrutha Biju

.

DEPARTMENT OF COMPUTER APPLICATIONS

VISION

To emerge as a centre of excellence in the field of computer education with distinct identity and quality in all areas of its activities and develop a new generation of computer professionals with proper leadership, commitment and moral values.

MISSION

- Provide quality education in Computer Applications and bridge the gap between the academia and industry.
- Promoting innovation research and leadership in areas relevant to the socio economic progress of the country.
- Develop intellectual curiosity and a commitment to lifelong learning in students, with societal and environmental concerns.

PEO'S (Program Educational Objectives)

1. MCA Graduates will be able to progress career productively in software industry, academia, research, entrepreneurship pursuits, government, consulting firms and other IT enabled services.

2. MCA Graduates will be able to achieve peer-recognition as an individual or in a team by adopting ethics and professionalism, and communicate effectively to excel well in crisis and inter-disciplinary teams.

3. MCA Graduates will be able to continue life-long professional development in computing and in management that contributes in self and societal growth.

20MCA246 MAIN PROJECT

Co No	***CO***	***Blooms Category***
CO1	Identify a real-life project which is useful to society / industry	Level 2: Understand
CO2	Interact with people to identify the project requirements	Level 3: Apply
CO3	Apply suitable development methodology for the development of the product / project	Level 3: Apply
CO4	Analyse and design a software product / project	Level 4: Analyse
CO5	Test the modules at various stages of project development	Level 5: Evaluate
CO6	Build and integrate different software modules	Level 6: Create
CO7	Document and deploy the product / project	Level 3: Apply

ABSTRACT

An Entry Way Booking System is an online platform that streamlines the process of reserving and managing the use of the college's various event spaces and facilities. The system provides a centralized and organized method for booking and scheduling events, reducing the administrative burden and increasing efficiency. This project is designed with the goal of making the existing system more informative, reliable, fast, and easier. This system helps the college use common and individual facilities within the facility or a building complex. The venue booking, approval, and views across the college enhance operational efficiency. Meeting rooms, conference halls, seminar halls, college space auditoriums, training facilities, etc. can be booked through the system. The application will be used by HOD, Principle, and faculties. The Entryway booking system is the easiest way to manage the venue booking functionalities of an institute. The system's objective is that users can search the facilities and book them from anywhere. Earlier, faculties used to book these facilities manually, which was time-consuming and included a lot of paperwork. It was in a way difficult to book these facilities. Now, staff in college can book these facilities easily through these platforms. They don’t have to go anywhere to avail of these venues. By checking the availability of the venue, the staff can book this venue online and book these venues for a short period. The booking system helps to simplify the venue booking process and raise management efficiency.

Keywords: booking, venue, facilities, approval, faculty, view.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 GENERAL INTRODUCTION

The project named “Entryway Booking System” is a software provided that deals with college venue automation. This system provides an efficient way of managing the venue booking information. This project is designed with the goal to making the existing system more informative, reliable, fast and easier. This project is based on the college. Entryway Booking System is the easiest way to manage venue booking functionalities of an institute.

The main objective of this project is to mobility and automation to the process of venue booking in an institute. The application will be used by staff and principal As we know there are many venues available in the field of college. This includes facilities like conference halls, seminar hall, college space auditoriums etc. In early stages, staff used to book these venues manually. This process was time consuming and included lot of paper works. It has been very hard to book these venues. Now, staff can book these venues through online platforms. They don’t have to go anywhere to avail these venues. By checking the availability of venue, the staff can book this venue through online and book these venues for a short period of time. It also helps to simplify the venue booking process and raise the efficiency in management.

1.2 PROBLEM STATEMENT

Firstly, the current system of booking seminar halls for faculties is often done manually and is time-consuming, leading to errors and inefficiencies. Secondly, the availability of seminar halls is not easily accessible, causing confusion and delays in the booking process. Thirdly, there is a lack of a centralized platform for managing seminar hall bookings, making it difficult to keep track of bookings and availability. Finally, the current system does not provide an easy way to manage cancellations and rescheduling of bookings, causing further inconvenience for faculties. The project aims to solve these problems by creating an automated system that allows faculties to easily book seminar halls, check availability, and manage their bookings efficiently.

1.3 OBJECTIVE OF THE PROJECT

The objective of Entryway Booking System is to provide simplify the process of reserving seminar halls for academic activities such as conferences, workshops, seminars, and meetings. This system will automate the scheduling and tracking of reservations, provide notifications and reminders, and offer a user-friendly interface with robust functionalities such as data analytics and reporting. The ultimate goal is to improve collaboration, communication, and productivity among faculty members, leading to better academic outcomes.

1.4 SOFTWARE INTRODUCTION

1.4.1 PYTHON

Python is a popular programming language. Working with Python in Visual Studio Code, using the Microsoft Python extension, is simple, fun, and productive. The extension makes VS Code an excellent Python editor, and works on any operating system with a variety of Python interpreters. It leverages all of VS Code's power to provide auto complete and IntelliSense, linting, debugging, and unit testing, along with the ability to easily switch between Python environments, including virtual and conda environments.

Advantages of Python

1 .Easy to read learn and write

Python is a high-level programming language that has English-like syntax. This makes it easier to read and understand the code.

Python is really easy to pick up and learn, that is why a lot of people recommend Python to beginners. You need less lines of code to perform the same task as compared to other major languages like C/C++ and Java.

2. Improved Productivity

Python is a very productive language. Due to the simplicity of Python, developers can focus on solving the problem. They don’t need to spend too much time in understanding the syntax or behavior of the programming language. You write less code and get more things done.

3 .Interpreted Language

Python is an interpreted language which means that Python directly executes the code line by line. In case of any error, it stops further execution and reports back the error which has occurred. Python shows only one error even if the program has multiple errors. This makes debugging easier.

4. Dynamically Typed

Python doesn't know the type of variable until we run the code. It automatically assigns the data type during execution. The programmer doesn't need to worry about declaring variables and their data types.

5. Free and Open-Source

Python comes under the OSI approved open-source license. This makes it free to use and distribute. You can download the source code, modify it and even distribute your version of Python. This is useful for organizations that want to modify some specific behavior and use their version for development.

1.4.2 MYSQL

MYSQL the world"s second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language. The MySQL development project has made its a Source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL is customizable. MySQL works very well combination of various programming language like C, C++, JAVA.

2. ABOUT THE ORGANIZATION

2.1 OVERVIEW

The establishment of St. Joseph's College of Engineering, was the fulfillment of a long- cherished dream of providing facilities for higher education to the people of the diocese and surrounding regions. The main objective is to develop a college with a distinct identity and character, where education and training are imparted in a truly Christian environment conducive to fostering Christian values such as faith in God, love for their fellow men and devotion to the motherland. Every facility is provided in the campus to create an environmentfully conducive to realizing this objective.

Discipline, hard work, positive thinking, commitment to excellence and abiding faith in the Almighty are the guiding principles that propel the college to its vision of emerging as a Centre of Excellence in technical education in the country. Value systems such as eco- friendliness, quality consciousness and work ethics are also being instilled through the specialwork culture and campus life existing in the college.

The college aims to provide an education that WORKS! – An education that helps the students in ensuring a challenging and satisfying career after the course.

2.2 VISION

Developing into a world-class, pace-setting Institute of Engineering and Technology with distinct identity and character, meeting the goals and aspirations of the society.

2.3 MISSION

- To maintain a conducive infrastructure and learning environment for world class education.
- To nurture a team of dedicated, competent and research-oriented faculty.
- To develop students with moral & ethical values, for their successful career byoffering variety of programmers and services.

2.4 OBJECTIVE

St. Joseph's College of Engineering and Technology, Palai was instituted with the objective of developing a center of professional learning with a distinct identity and character, for imparting education and training in a truly Christian environment, fostering Christian values of faith and love to God and fellowmen. The college aims to provide the kind of education that helps to achieve academic excellence and thereby ensures a challenging and satisfying career for the students on the successful completion of the programmer. With this perspective,training is organized on a regular basis for the development of personality, learning and communication skills as well as employability skills.

3. SYSTEM ANALYSIS

System analysis is a structured method for identifying and solving problems. Analysis implies breaking something into its parts so that the whole may be understood. The definition of system analysis not only process analysis but also that of synthesis, which implies the process of putting parts together to form a new whole. All the activities relating to the life cycle phase must be performed managed and document. To design a system, we need requirements of the system and the specification document are prepared in this phase. The purpose of this Entryway Booking is to specify the functional requirement of the software that is to build. The specifications are intended to guide the activities, relationships and all other objectives.

The main thing is to find what is to be done to solve the problems with the current system. In the phase the problems or drawbacks of the current system is identified and the necessary actions to solve these problems are recommended.

3.1 EXISTING SYSTEM

The existing system is completely manual. The staff can book venues manually. There are lot of paper works and time consuming. The current system is dealing on the manual basis. Staff have to go directly to the office to book the venue in a particular date. It needs huge amount of paper works to maintain the booking details. A huge expenditure and lots of time is spending in the existing system. Tracking and retrieving of data from bulk of paper is a difficult process. The existing system is very paper based in institution. Relevant and irrelevant information of venue booking are entered and stored in the same records, which is very clumsy and process. In this process very difficult to maintain the details of venue booking in a proper way. The venue booking, approve and reschedule procedure is very inefficient. Further, there are chances of data misplacement and wrong data entry. The system is still very inflexible. In the existing system, the manual process, receiving data from staffs are done through manual records. These records are entered in manual process.In this process will take long time.

DISADVANTAGES OF EXISTING SYSTEM:

- Almost a completely manual work
- Lack of data security
- Time consuming
- No chance for error detection and timely solving

3.2 PROPOSED SYSTEM

The proposed system is an online web application. This system is proposed to control and avoid the limitations of existing system. The goal of the proposed system is to increase the efficiency by speeding up the process and bringing down the work load. In the proposed system, the staff need not go anywhere, he can just get all information about venues in a single click. The staff needs to login, so that he can select the venues and book by checking the availability. It is possible for the staff to access the details regarding the venues from anywhere around the globe. The main objective of the proposed system is to provide a user-friendly interface. The system, which is proposed, now computerizes all the venue booking details that are maintained manually.

ADVANTAGES OF THE PROPOSED SYSTEM

- Software automation and organizing of data and details make the system easy.
- This system saves time of faculites.
- Provide user friendly interface.
- Users can check availability
- Users can view all other bookings and booking history views

3.3 FEASIBILITY ANALYSIS

System feasibility is a test or evaluation of the complete system plan. Such an evaluation is necessary to define the application area along with the extend and capability to provide the scope of computerization together with suggested output and input format and potential benefits. Feasibility study is a proposal according to the work ability, impact on the organization, ability to meet user's needs and efficient use of resources. The feasibility study isconducted to determine if the proposed system is feasible or not. Feasibility analysis evaluatesthe candidate systems and determines the best system that needs performance requirements. The purpose of feasibility study is to investigate the present system, evaluate the possible application of computer-based methods,

select a tentative system, evaluate the cost andeffectiveness of the proposed system, evaluate impact of the proposed system on existing personnel and ascertain the need for new personnel.

All projects are feasible when given unlimited resources and infinite time. It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. A feasibility study is not warranted for systems in which economic justification is obvious, technical risk is low, few legal problems are expected and no reasonable alternative exists. Anestimate is made of whether the identified user needs may be satisfied using current software and hardware technologies.

The study will decide if the proposed system will be cost effective from the business point ofview and if it can be developed in the given existing budgetary constraints. The feasibility study should be relatively cheap and quick. The result should inform the decision of whether to go ahead with a more detailed analysis. Feasibility study may be documented as a separated report to higher officials of the top-level management and can be included as an appendix to the system specification. Feasibility and risk analysis are related in many ways.If there is more project risk then the feasibility of producing the quality software is reduced.

The key combinations are involved in the feasibility study:

- Economic Feasibility
- Technical Feasibility
- Behavioral Feasibility
- Operational Feasibility

3.3.1 ECONOMIC FEASIBILITY

Economic analysis is the most frequently used method for evaluating the effectiveness of a candidate system which is known as cost benefit analysis. In cost benefit analysis, thebenefits and savings that are expected from candidate systems compared with costs. Ifbenefits outweigh cost then the decision is made to design and implement a system. Otherwise further justifications or alterations in the proposed system will have to be made ifit has to be approved.

This project aims at reducing time, effort and cost for web masters. The system is developed under optimal expenses with the hardware and software. The developed system is available free of cost. Anybody can get the benefit of the system by simply using it as a service. There is no additional cost for using or implementing the system. It can be used in windows-based system, and need not

alter the current system configurations. This makes the system feasible economically. Besides it is good economic to insist in such a kind of software from the project manager's point of view as the benefits over weighs the cost. The resources needed to run the above project should be less in cost, easily available and highly reliable. This is a cost-effective project because of its accuracy, fastness and user-friendly nature. It is only required to host the site in the intranet inside the corresponding software firm or organization.So, there will be no additional expenses to host the site.

3.3.2 TECHNICAL FEASIBILITY

- Does the necessary technology exist to do what is been suggested?
- Does the proposed equipment have the technical capacity for using the new system?
- Are there technical guarantees of accuracy, reliability and data security?

A study of function, performance and constraints may improve the ability to create anacceptable system. Technical feasibility is frequently the most difficult area to achieve at the stage of product engineering process. Considering that are normally associated with the technical feasibility include Development risk, Resource availability, Technology. Technical feasibility study deals with the hardware as well as software requirements. Project requirement system must be functional and multi user one should be based on specific technology the system under study must be practical and platform independent. It should be compactable with all kind of existing system in industry and should not provide any overhead to user. Implementation of existing system does not require changing of the existing configure of the system.

3.3.3 BEHAVIORAL FEASIBILITY

Behavioral feasibility is concerned with the working of the system after its installation. The company has a good record of development, installation and maintenance of systems for its clients. So, this system can be installed in the client environment and the company will helpin the maintenance of the system in future. Proposed projects are beneficial only if they canbe turned into information systems that will meet the organizations operating requirements simply stated, this test of feasibility asks if the system will work when it is developed and installed.

Are there major barriers to Implementation? Here are questions that will help test the operational feasibility of a project.

The proposed project would be beneficial to all Organizations that, it satisfies the objectives when developed and installed. All the behavioral aspects are considered carefully. Thus, the project is behaviorally feasible and it can also be implemented easily.

3.3.4 OPERATIONAL FEASIBILITY

Question that going to be asked are:

- Will the system be used if it developed and implemented?
- If there was sufficient support for the project from the management and from theusers.
- Have the users been involved in planning and development of the project.
- Will the system produce poorer result in any respect or area?

This application can be implemented in an organization because there is adequate support from management and users. And application will be used by them since it doesn't generate poorer results or problems in any area. Therefore, the implementation of this application is operationally feasible. Operational feasibility is concerned with human, organizational and political aspects. The issues considered are the job changes that will be brought about, the organizational structures that will be distributed and the new skills that will be required.

3.4 SOFTWARE REQUIREMENT SPECIFICATION

The main purpose of this project that manage the institution venue booking. The main objective of this project is to mobility and automation to the process of managing venues in an institute. In early stages, staff used to book these facilities manually. This process was time consuming and included lot of paper works. It has been very hard to book these facilities. Now, staff can book these facilities through online platforms. They don't have to go anywhere to avail these facilities. By checking the availability of facility, the staff can book this facility through online. The staff can book these venues for a short period of time and it also helps to simplify the booking process and raise the efficiency in management.

In today's life, manual booking is not the appropriate way of booking things. It is changes according to our workload and situation. The need of this project is to manage the venues and booking venues easily. It is complete time management system.

This project is a web application that aims introducing a wide variety of technology for managing venues and booking the venues. The feature provided by this application including secure login and staff can search the venues. Also they can book the available venue.

3.4.1 SOFTWARE REQUIREMENTS(HARDWARE AND SOFTWARE)

HARDWARE SPECIFICATION

Main Processor	Intel Core i3 10th Gen
RAM	4 GB
Clock Speed	1.66 GHz
Hard Disk Drive	1TB

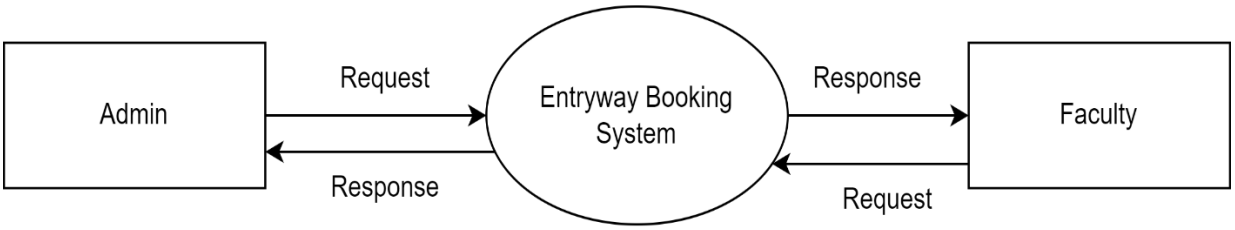
SOFTWARE SPECIFICATIONS

Operating System	Windows 10
Web Browser	Google Chrome
Front-End	HTML, CSS, Javascript
Back-end	MYSQL
Framework	Django
IDE	Visual studio code
Tool	XAMPP

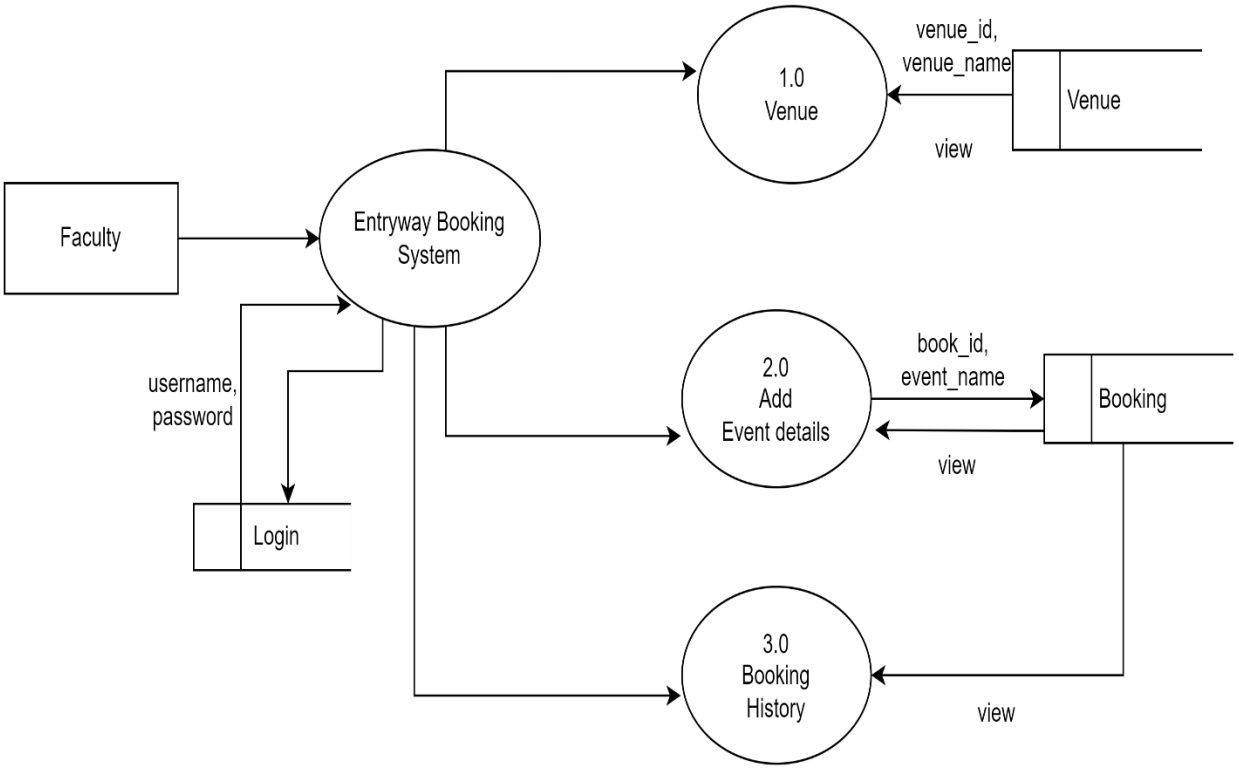
3.5 DATA FLOW DIAGRAM(DFD)

A Data Flow Diagram (DFD) is a graphical technique that depicts information flow and transforms that are applied as data move from input to output. It is otherwise known as bubble chart. It has the purpose of clarifying system requirements and identifying major transformations that will become programs in the system design. It isthe major starting point in the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. A DFD consists of a series of bubbles joined by lines. The bubble represents data flow in the system. The DFD is used to represent increasing information flow and functional details. A Level 0 DFD is called a fundamental system model represents the entire software elements as single bubble with input and output indicated by incoming and outgoing arrows respectively. Additional process and information flow parts are represented in the next level, i.e., Level 1 DFD. Each of the processes represented at Level 1 are sub functions of overall system depicted in the context model. Any processes which are complex in Level 1 will be further represented into subfunctions in the next level, in Level 2. Data Flow diagram is a means of representing a system at any level of detail with a graphic network of symbols showing data flows, data stores, data processes and data sources. The purpose of data flow diagram is to provide a semantic bridge between users and system developers. The diagram are graphical, eliminating thousands of words, logical representation, modelling what system does; hierarchical, showing systems at any level of details; and jargon less, allowing user understanding and reviewing. The goal of data flow diagram is to have a commonly understood model of a system. Data flow diagram area supported by other techniques of structured system analysis such as data structured diagrams, data dictionaries and procedure representing techniques such as decision tables, decision trees.

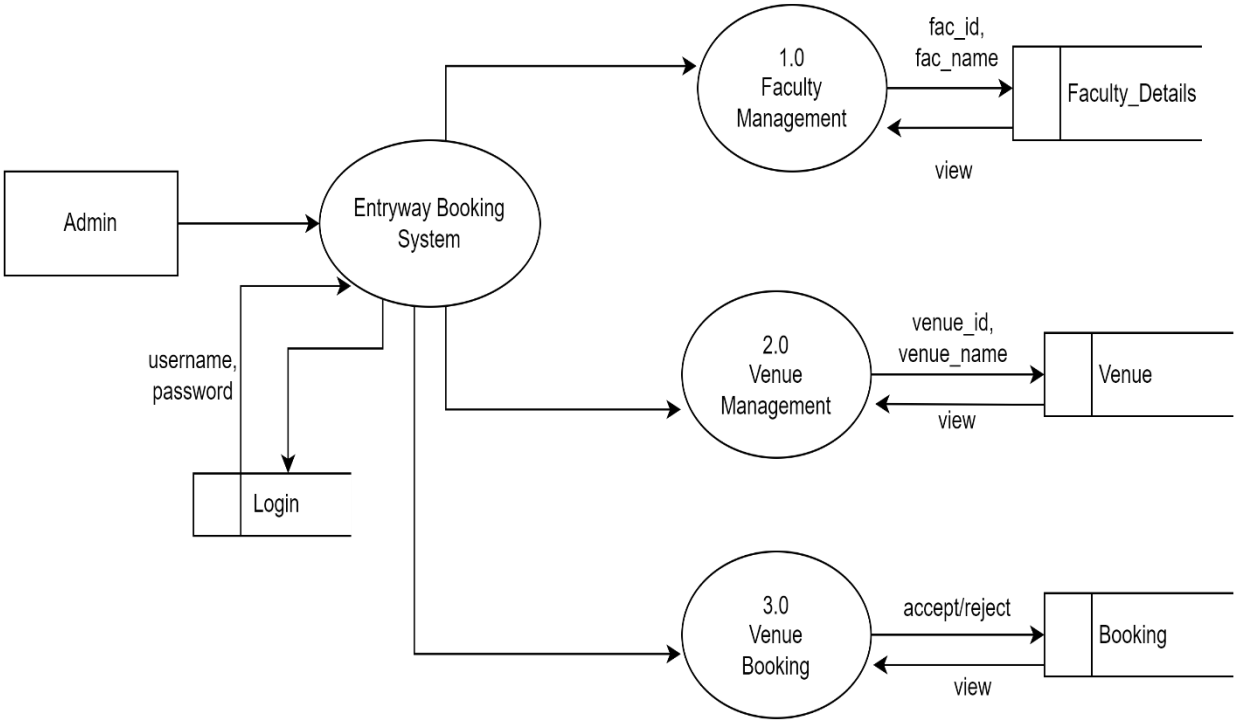
CONTEXT LEVEL-LEVEL 0 DFD



LEVEL 1 DFD – FACULTY/USER



LEVEL 1 DFD – ADMIN



4. TOOLS AND PLATFORMS

4.1 FRONT-END TOOL

HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheet (CSS) and scripting languages such as JavaScript. Web browserreceives HTML documents from a web server or from local storage and renders the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C). former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997. Hypertext Markup Language (HTML).

4.2 BACK-END TOOL

MYSQL

MYSQL the world‟s second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius's daughter, My. The SQL phrase stands for Structured Query Language. The MySQL development project has made its a Source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL is customizable. MySQL works very well combination of various programming language like C, C++, JAVA.

4.3 IDE

VISUAL STUDIO CODE V1.69

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages and runtimes

(such as C++, C#, Java, Python, PHP, Go, .NET). In normal terms, it facilitates users to write the code in an easy manner. Many people say that it is half of an IDEand an editor, but the decision is up to to the coders.

Why Visual studio code?

Visual Studio Code is a very popular coding editor used by millions of developers around the world.If you are a Web Developer, you probably use Visual Studio Code for coding your projects.

Its popularity is due to the growth of the web development field in these years and the need ofthe developers of having a lightweight well-done editor, with few features but less complex than the others available on the market.

It's also free and it's developed and maintained by Microsoft with a modern approachusing Electron.

These are some characteristics that bring VS Code to the top of the coding editors

- Open-source
- Simplicity
- Familiarity
- Minimal Design
- Extensions

5. SYSTEM DESIGN

System designing is the process of defining the architecture, components, modules, interfacesand data for a system to satisfy specified requirements. it is a solution to a “how to” approachcompared to system analysis which is a “what is” orientation. it translates the system requirements into ways of making them operational. The design phase focuses on the detailedimplementation of the system recommended in the feasibility study.

The system which is in making is developed by working on two different modules and combining them to work as a single unit. That single unit is the one which is known as the new software. we go through the different design strategies to design the system we are talking about. In the input design we decide which type of input screens are going to be used for the system in making. In the output design we decide the output screens and the reports that will be used to give the output and in the database design we decide what all tables will be required and what all fields will be there in those tables. Each of them discussed briefly below.

5.1 INPUT DESIGN

Input design converts user-oriented inputs to computer-based formats, which requires careful attention. The collection of input data is the most expensive part of the system in terms of the equipment used and the number of people involved. In input design, data is accepted for computer processing and input to the system is done through mapping via a map support or links. Inaccurate input data is the most common cause of errors in data processing. The input screens need to be designed more carefully and logically. A set of menus is provided which help for better application navigation. While entering data in the input forms, proper validation checks are done and messages will be generated by the system if incorrect data hasbeen entered. The objective of input design is to create an input layout that is easy to follow and prevent operator errors. It covers all phases of input from creation of initial data into actual entry of the data to the system for processing. The input design is the link that ties the system into world of its users. The user interface design is very important for any application.The interface design defines how the software communication within itself, to system that interpreted with it and with human who use it. The input design requirements such as user friendliness, consistent format and interaction dialogue for giving the right message and help for the user at right time are also considered for the development of the project.

5.2 OUTPUT DESIGN

Outputs are the most important and useful information to the user and to the department. Intelligent output designs will improve systems relationships with the user and help much in decision-making. Outputs are also used to provide a permanent hard copy of the results for later use. The forms used in the system are shown in the appendix. The outputs also vary in terms of their contents, frequency, timing and format. The users of the output, its purpose andsequence of details to be printed are all considered. The output forms a system in thejustification for its existence. If the outputs are inadequate in any way, the system itself is inadequate. The basic requirements of output are that it should be accurate, timely and appropriate, in terms of content, medium and layout for its intended purpose. Hence it is necessary to design output so that the objectives of the system are met in the best possible manner.

5.3 TABLE DESIGN

The efficiency of an application using SQLITE-3 Server is mainly dependent upon the database tables, the fields in each table and joined using the fields contained in them to retrieve the necessary information. A table is a set of data elements that is organized using a model of vertical columns and horizontal rows. A table has a specified number of columns, but can have any number of rows. Each row is identified by the values appearing in a particular column subset which has been identified as a unique key index. The primary objective of a database design is fast response time to inquiries, more information at low cost,control of redundancy, clarity and ease of use, accuracy and integrity of the system fast recovery and availability of powerful end-user language.

Table 1: Login

Field	Datatype	Constraints	Description
Username	Varchar(20)	Notnull	Username
Password	Varchar(20)	Notnull	Password
type	Varchar(10)		Type of user

Table 2: Faculty_Details

Field	Datatype	Constraints	Description
fac_id	Varchar(10)	Primary key	Faculty id
fac_name	Varchar(20)	Notnull	Faculty name
gender	Varchar(10)	Notnull	Gender
des	Varchar(20)	Notnull	Designation
dep	Varchar(20)	Notnull	Department
phone	Number(10)	Notnull	Mobile number
Email	Varchar(20)	Notnull	Email id
pass	Varchar(20)	Notnull	password

Table 3: Venue

Field	Datatype	Constraints	Description
venue_id	Varchar(10)	Primary Key	Id of venue
venue_name	Varchar(20)	Notnull	Name of venue
capacity	Int(30)	Notnull	Capacity
loc	Varchar(20)	Notnull	Location of venue

Table 4: Booking

Field	Datatype	Constraints	Description
book_id	Varchar(10)	Primary key	Booking id
fac_id	Varchar(10)	Foreign key	Id of faculty
venue_id	Varchar(10)	Foreign key	Id of venue
event_name	Varchar(20)	Notnull	Name of event
dep	Varchar(20)	Notnull	Department
event_details	Varchar(50)	Notnull	Details of event

n_persons	Int(50)	Notnull	No.of persons
date	date	Notnull	date
start_time	time	Notnull	Start time
end_time	time	Notnull	End time
pletter	image	Notnull	Hod permission letter
status	Boolean		Booking status

5.4 PROCESS DESIGN

A successful process design has to take into account the appropriateness of the process to overall organization objective. Process design requires a broad view of the whole organization and should not have a myopic outlook. And the process should deliver customer value with constant involvement of the management at various stages.

In order to achieve a good process design, effective process strategy is required, which deals with a singular line item required to manufacture the end product. Effective process strategy deals with raw material procurement, customer participation, technology investment, etc.

Over a period of time process design has undergone change and new concepts like flexible manufacturing systems have been developed, which delivers efficient and effective production design and analysis.

5.4.1 MODULE DESCRIPTION

ADMIN

- Login
- Faculty registration/ add faculty
- Update and delete faculty
- Add Venues
- Approve/Reject bookings
- View bookings

USER

- Login
- Booking different venues
- Availability checking before bookings
- View venues
- View all, upcoming, running events
- View booking status

6. SYSTEM TESTING

6.1 TESTING METHODOLOGIES AND STRATEGIES

Software testing is an integral part of to ensure software quality, some software organizations are reluctant to include testing in their software cycle, because they are afraid of the high costassociated with the software testing. There are several factors that attribute the cost ofsoftware testing. Creating and maintaining large number of test cases is a time- consuming process. Furthermore, it requires skilled and experienced testers to develop great quality test cases.

Even with the wide availability of automation tools for testing, the degree of automation mostly remains at the automated test script level and generally significant amount of human intervention is required in testing. In addition, data collected, as testing is conducted providesa good indication of software quality as a while. The debugging process is the most unpredictable part of testing process. Testing begins at the module level and work towards theintegration of entire computer-based system. No testing is completed without verification andvalidation part.

The goal of verification and validation activities are to access and improve the quality of work products generated during the development and modification of the software. Testing plays a vital role in determining the reliability and efficiency of the software and hence is very important stage in software development. Tests are to be conducted on the software to evaluate its performance under a number of conditions. Ideally, it should do so at the level of each module and also when all of them are integrated to form the completed system.

In the project “ENTRYWAY BOOKING SYSTEM” the testing has been successfully handled with the modules.The test data was given to each and every module in all respect and got the desired output. Each module that has been tested is found working properly.

6.1.1 UNIT TESTING

Here we test each module individually and integrated the overall system. Unit testing focuses verification efforts even in the smallest unit of software design in each module. This isknown as” module testing”. The modules of the ” ENTRYWAY BOOKING SYSTEM” are tested separately. This testing is carried out in the programming style itself. In this testing each module is focused to work satisfactorily as regard to expected output from the module. There are some validation checks for the fields. Unit testing gives stress on the modules of “ ENTRYWAY BOOKING SYSTEM”

independently of one another, to find errors. Different modules are testedagainst the specifications produced during the design of the modules. Unit testing is done to test the working of individual modules with test servers. Program unit is usually small enough that the programmer who developed it can test it in a great detail. Unit testing focuses first on that the modules to locate errors. These errors are verified and corrected and so that the unit perfectly fits to the project.

6.1.2 INTEGRATION TESTING

Data can be lost across an interface, one module can have an adverse effect on the other sub-functions, when combined they may not perform the desired functions. Integrated testing is the systematic testing to uncover the errors within the interface. This testing is done with simple data and the developed system has run successfully with this simple data. The need forintegrated system is to find the overall system performance.

After splitting the programs into units, the units were tested together to see the defects between each module and function. It is testing to one or more modules or functions together with the intent of finding interface defects between the modules or functions. Testing completed at as part of unit or functional testing, integration testing can involve putting togetherof groups of modules and functions with the goal of completing and verifying meets the system requirements.

6.1.3 SYSTEM TESTING

System testing focuses on testing the system as a whole. System Testing is a crucial step in Quality Management Process. In the Software Development Life Cycle, System Testing is the first level where the System is tested as a whole. The System is tested to verify whether it meets the functional and technical requirements. The application/System is tested in an environment that closely resembles the production environment where the application will be finally deployed.

The perquisites for System Testing are: -

- All the components should have been successfully Unit Tested.
- All the components should have been successfully integrated.
- Testing should be completed in an environment closely resembling the production environment. When necessary, iterations of System Testing are done in multiple environments.

6.1.4 USER ACCEPTANCE TESTING

The system was tested by a small client community to see if the program met the requirements defined the analysis stage. It was fond to be satisfactory. In this phase, the system is fully tested by the client community against the requirements defined in analysis and design stages, corrections are made as required, and the production system is built. User acceptance of the system is key factor for success of the system.

6.1.5 DATA VALIDATION TESTING

Data validation is the process of testing the accuracy of data. A set of rules we can apply to a control to specify the type and range of data that can enter. It can be used to display error alert when users enter incorrect values in to a form. Now performing validation testing in system Centralized Social Welfare by undergoing validation for each tool and the validation succeeded when the software function in a manner that can be reasonably accepted, by the user.

6.1.6 TESTING STRATEGIES

Top-down testing

Top-Down Testing tests the higher levels of a system before testing its detailed components. The program is represented as a single abstract component with sub components represented by stubs. Stubs have the same interface as the components but very limits functionally. After the top- level component has been tested, its sub- components are implemented and tested in the same way. This process continues recursively until the bottom-level components are implemented. The whole system may then be completely tested.

Bottom-up testing

Bottom-Up Testing is the converse of Top-Down Testing. It involves testing the modules at the lower levels of the hierarchy and then working up the hierarchy of the modules until the final module is tested. The advantage of bottom-up testing is the disadvantage of top-down testing and vice-versa. When using bottom-up testing test drivers must be written to exercise the lower-level components. These test drivers simulate the components environment and arevaluable components; the test drivers and test data should be distributed with the component. Potential re- users can then run these tests to satisfy themselves that the component behavesas expected in their environment.

Black box testing

Knowing the specified function that a product has been designed to perform, test can be conducted that demonstrates each function that is fully operational, at the same time searching for errors in each function. Black Box testing focuses on functional requirement of the software.

White box testing

Knowing the internal working of a product test can be conducted to ensure that "all gears mesh" that is internal operation performs according to specification and all internal components have been adequately exercised.

6.1.7 SAMPLE TEST CASES

SL NO.	TEST CASE	MODULE	EXPECTED RESULT
1	Verify whether an admin can log into the system using his/her username and password	Admin	Admin should be able to login the admin module successfully.
2	Check whether an admin can add and view faculties.	Admin	Admin should be able to add faculties.
3	Check whether an admin can edit faculties.	Admin	Admin should be able to edit faculties.
4	Check whether an admin can to delete faculties.	Admin	Admin should be able to to delete faculties.
5	Verify whether admin can view, accept or reject booking .	Admin	As an admin user he/she should be able to view, accept or reject venue booking successfully.

6	Check whether admin can view all bookings	Admin	Admin should be able to view all bookings.
7	Verify whether user can log into the system using his/her username and password.	User	User should be able to login successfully by provided username and password.
8	Check whether user can view all halls.	User	User should be able to view all halls.
9	Check whether user can select which hall.	User	User should be able to select which hall.
10	Verify whether user can check availability before booking.	User	User should be able to check availability before booking.
11	Check whether user can send request for booking venues.	User	User should be able to send request for booking venues.
12	Check whether user can check booking status .	User	User should be able to check booking status.
13	Verify whether user can view all, upcoming and running events.	User	User should be able to view all, upcoming and running events.
14	User should be able to view all other faculties booking events.	User	User should be able to view all other faculties booking events.

7. SYSTEM IMPLEMENTATIONS

Implementation includes placing the system into operation and providing the users and operation personnel with the necessary documentation to use and maintain the new system. Implementation includes all those activities that take place to convert from the old system to the new. The new system may be totally new, replacing an existing system. Proper implementation is essential to provide a reliable system to meet the organizational requirements. Successful implementation may not guarantee improvement in the organization using the new system, as well as, improper installation will prevent. There are four methods for handling a system conversion.

The Implementation Plan describes how the information system will be deployed, installed and transitioned into an operational system. The plan contains an overview of the system, a brief description of the major tasks involved in the implementation, the overall resources needed to support the implementation effort, and any site-specific implementation requirements. The plan is developed during the Design Phase and is updated during the Development Phase the final version is provided in the Integration and Test Phase and is used for guidance during the Implementation Phase.

The implementation phase ends with an evaluation of the system after placing it into operation of time. The validity and proper functionality of all the modules of the developed application is assured during the process of implementation. Implementation is the process ofassuring that the information system is operational and then allowing user to take over its operation for use and evaluation. Implementation is the stage in the project where thetheoretical design is turned into a working system. The implementation phase constructs, installs and operated the new system. The most crucial stage in achieving a new successful system is that it works effectively and efficiently.

There are three types of implementations:

- Conversion
- User training
- Documenting the system

7.1 IMPLEMENTATION PROCEDURE

Implementation is the process of personnel check out, install the required equipment and application and train user accordingly. Depending on the size of the organization and its requirements, the implementation is divided into three parts.

7.1.1 STAGE IMPLEMENTATION

Here system is implemented in stages. The whole system is not implemented at once. Once theuser starts working with the system and is familiar with it, then a stage is introduced and implemented. Also, the system is usually updated, regularly until a final system is sealed.

7.1.2 DIRECT IMPLEMENTATION

The proposed new system is implemented directly and the user starts working on the new system. The shortcoming, if any, faced are then rectified later.

7.1.3 PARALLEL IMPLEMENTATION

The old and new systems are not used simultaneously. This helps in comparison of the results from two systems. Once the user is satisfied and his intended objectives are achieved by the new system, he stops using the old one.

8. MAINTENANCE

Once the system has been implemented, it cannot be considered as the end of the system life cycle. After the implementation it is necessary that the system be constantly monitored so that it may be decided as how the system is working.If any problem is encountered it is necessary that the in-charge person rectifies the problem so that the clients may not be affected by the problem. This phase of the system development life cycle is known as the maintenance period.

There are three types of maintenances:

- Correctives (fixing bugs/errors)
- Adaptive (updates due to environment changes)
- Perfective (enhancements, requirements change)

9. CONCLUSION

In today‟s fast growing world, manual booking of venues is something that requires a change. Booking things online are the way of new world. The procedures involved in manual booking can now be completed in just a few clicks. My project Entryway booking management system in a great way by avoiding time wastage and many other unwanted expense. I hope that it will create high level of advancements in the field of College and other fields.

9.1 SCOPE FOR FUTURE ENHANCEMENT

The project will be very helpful in future. It saves time, effort and makes a better user- friendly environment to all the beneficiaries who use this project. Making enhancement is all about perfectible maintenance. It means adding, modifying or redeveloping the code to support changes in the specification. It is necessary to keep up with changing user needs and the operational environment.

The following are the future scope for the project:

The system can be developed into a well advanced system for facility booking management. The current web based utility can be incorporated to a mobile based application. The system can be upgraded by including more features like SMS facilities, Email notification etc.

10. BIBILOGRAPHY

10.1 REFERENCES

1.Raghu Ramakrishnan and Johannes Gehrke, “Database Management System”, thirdedition, McGraw-Hill Professional Publication

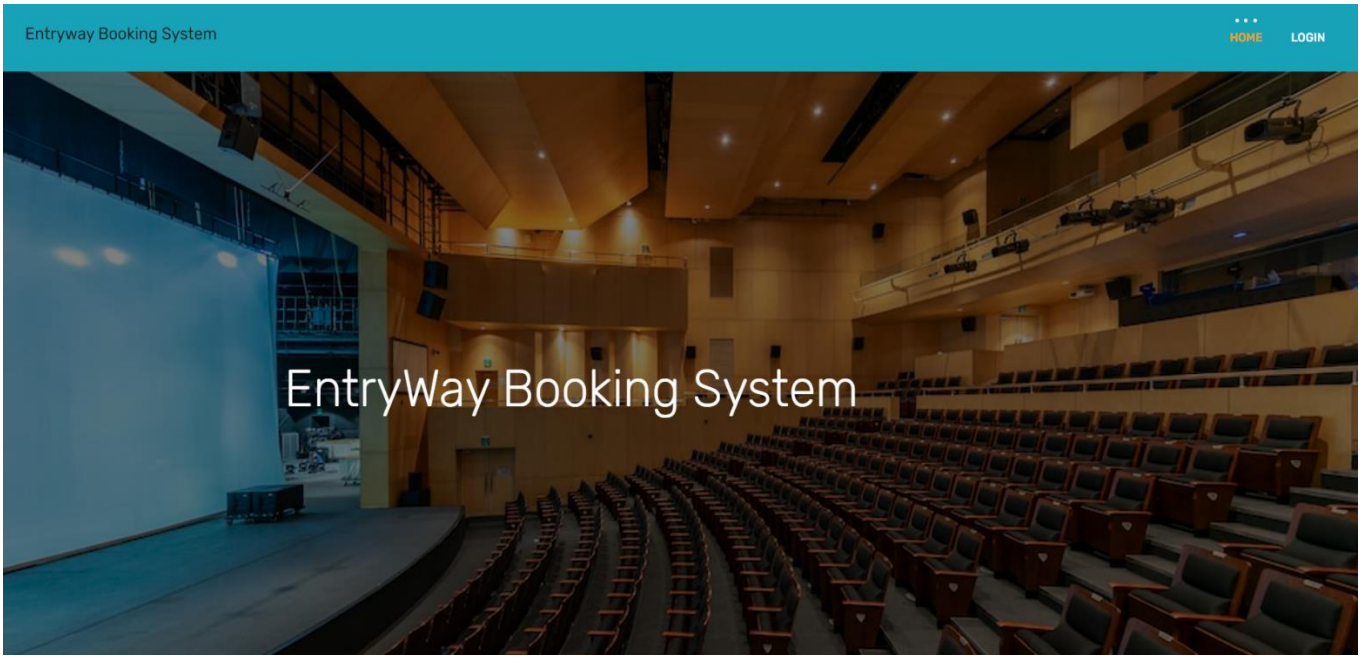
10.2 WEBSITES

- www.wikipedia.com
- www.mysql.com
- http://www.w3schools.com
- https://www.tutorialspoint.com
- https://www.bootstraptemplate.com

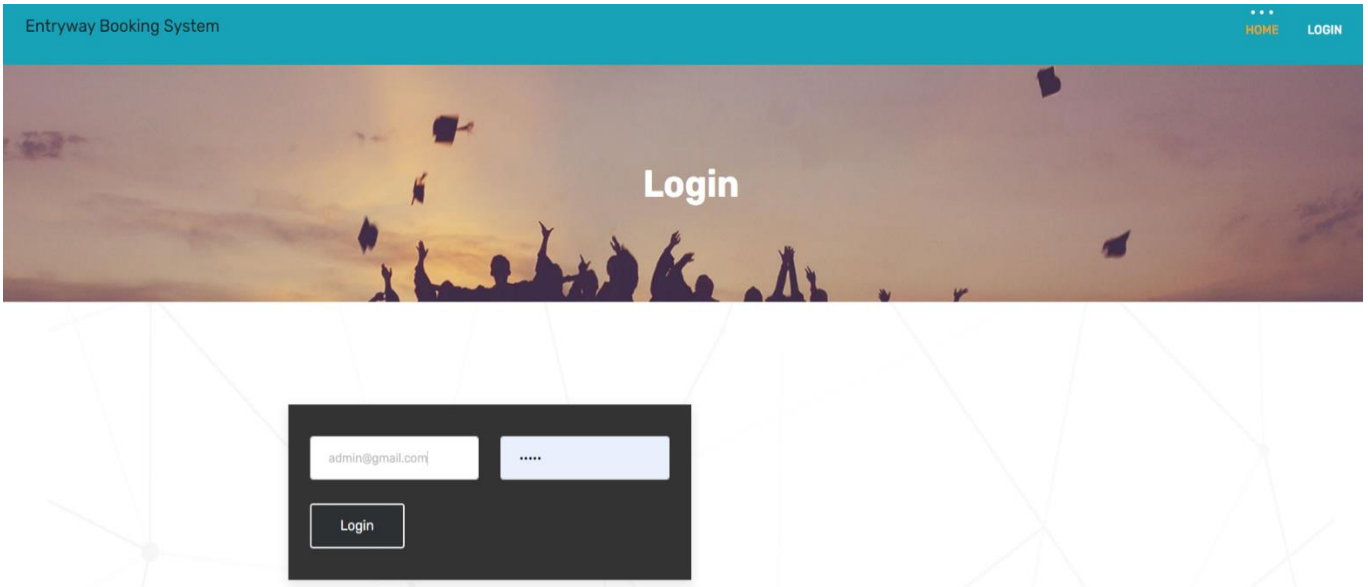
11.APPENDIX

11.1 APPENDIX-A: SCREEN SHOTS

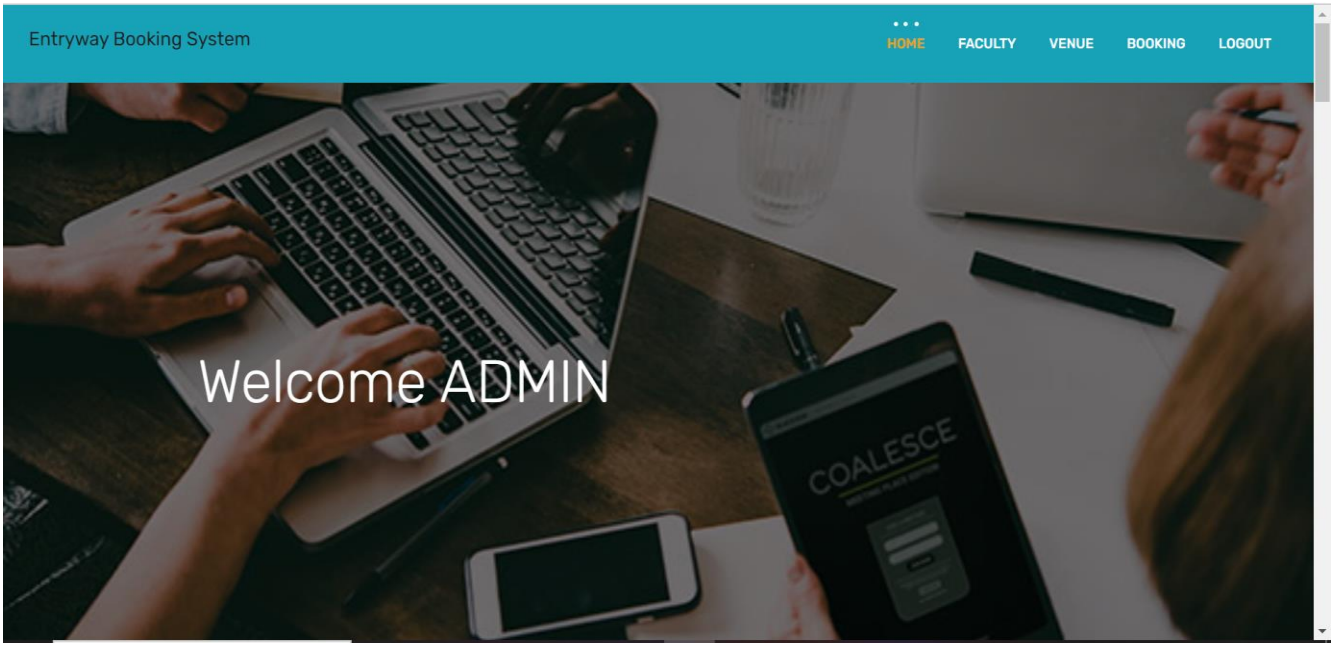
HOME PAGE



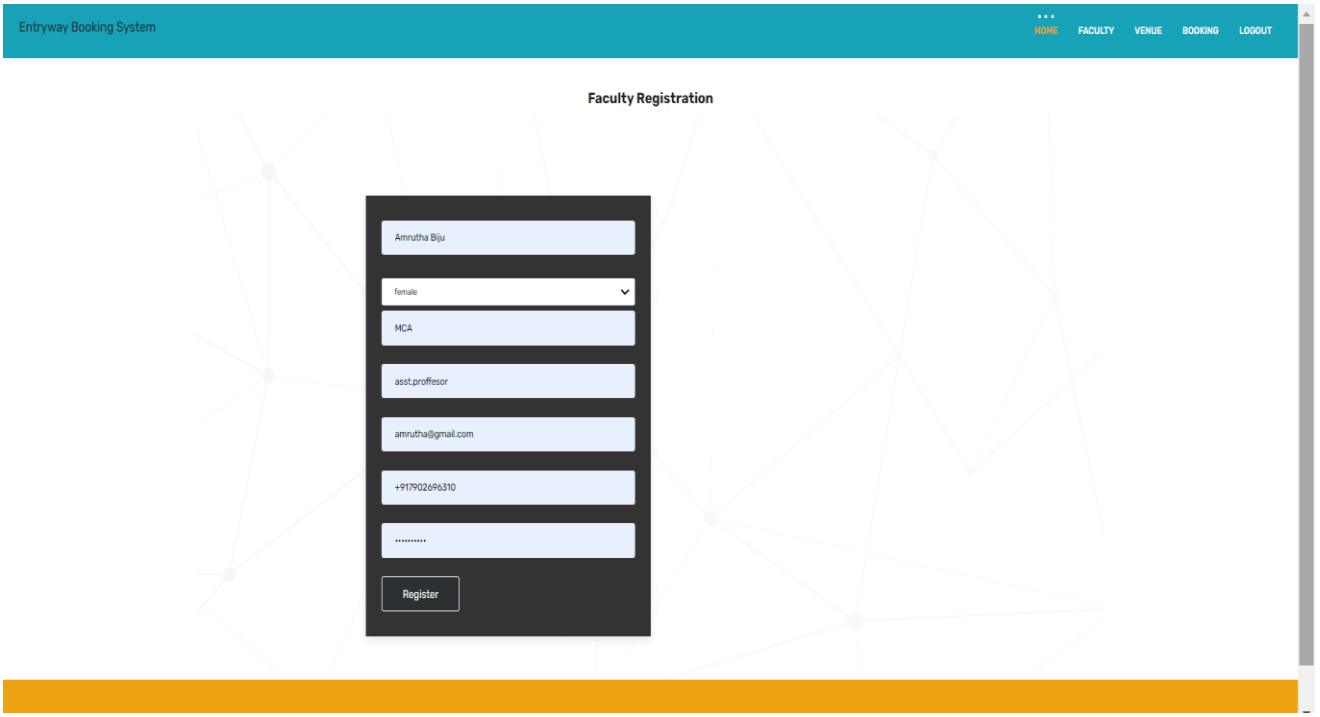
LOGIN



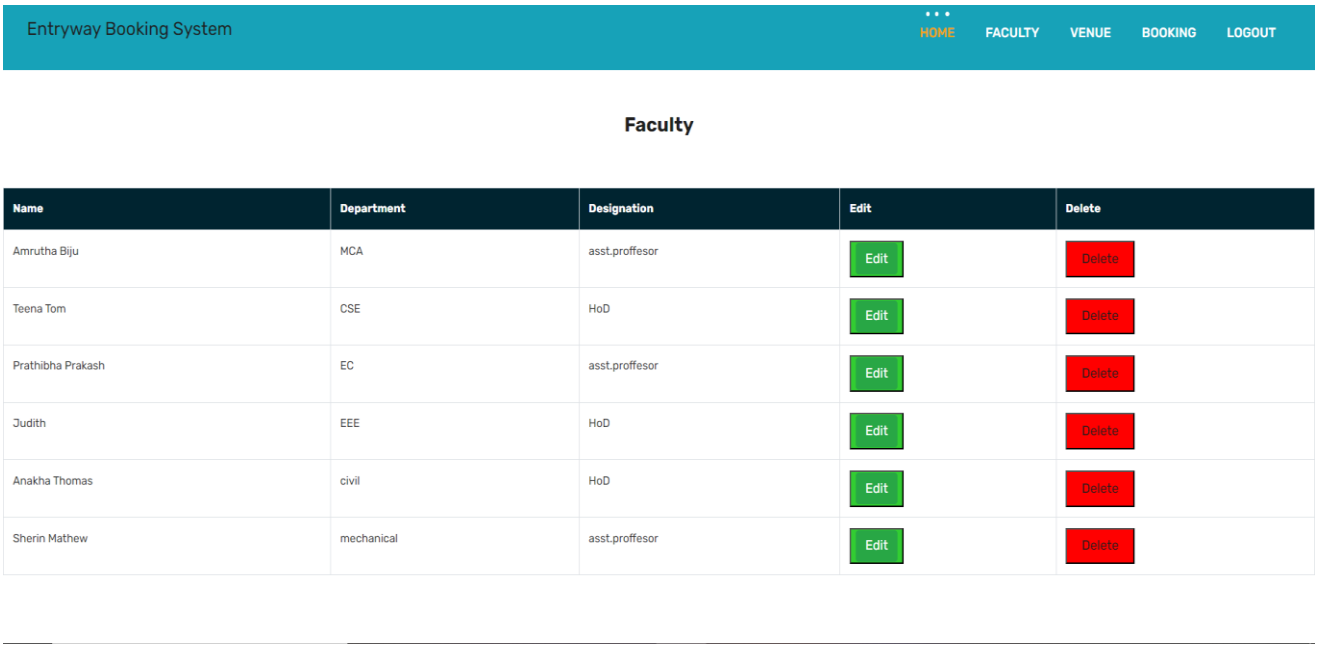
ADMIN-HOME



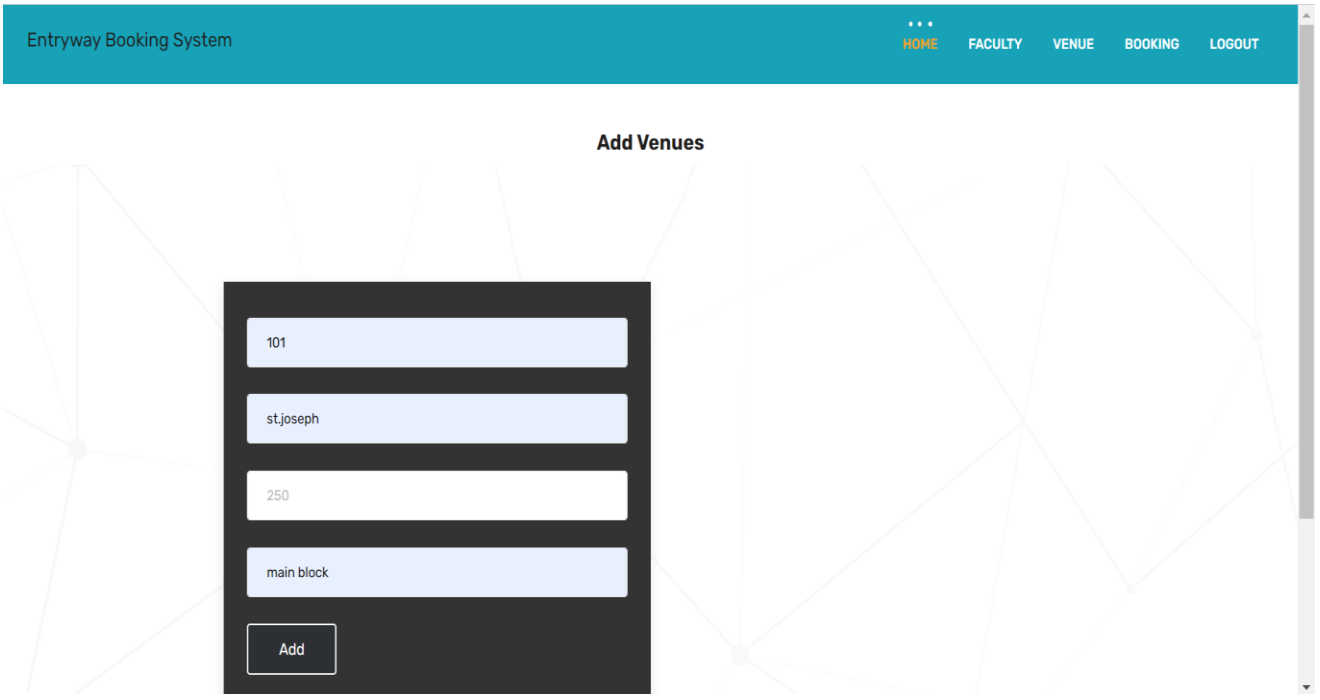
ADD FACULTY



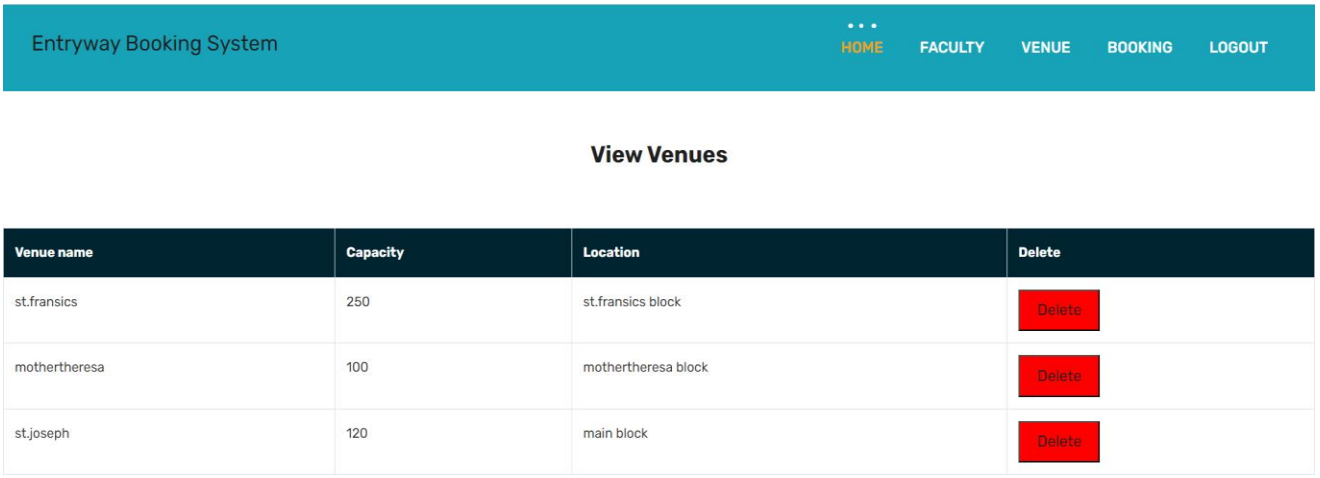
VIEW FACULTY



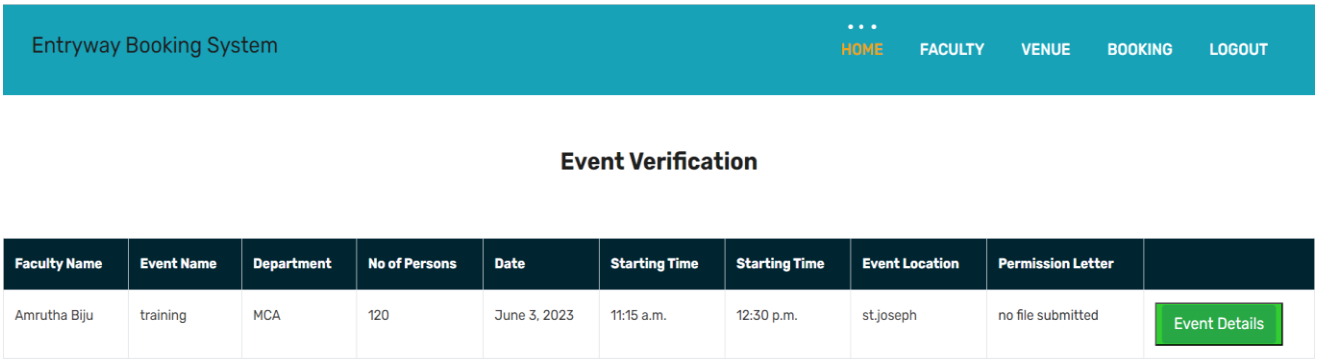
ADD VENUES

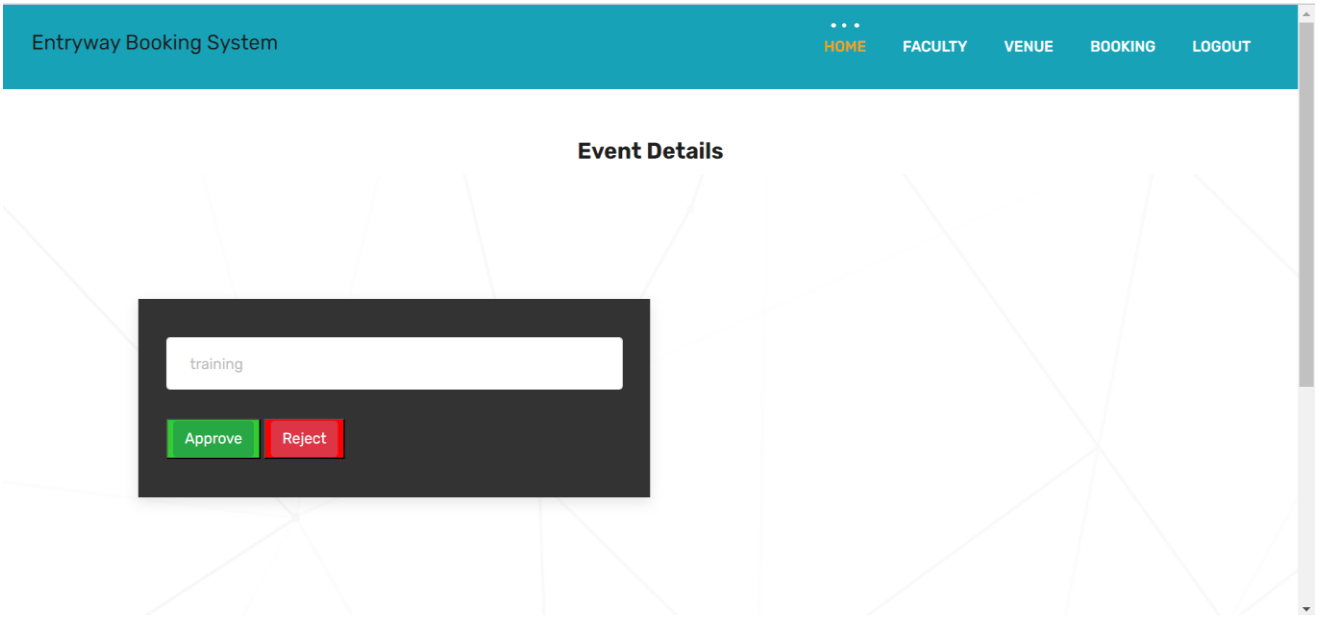


VIEW VENUES

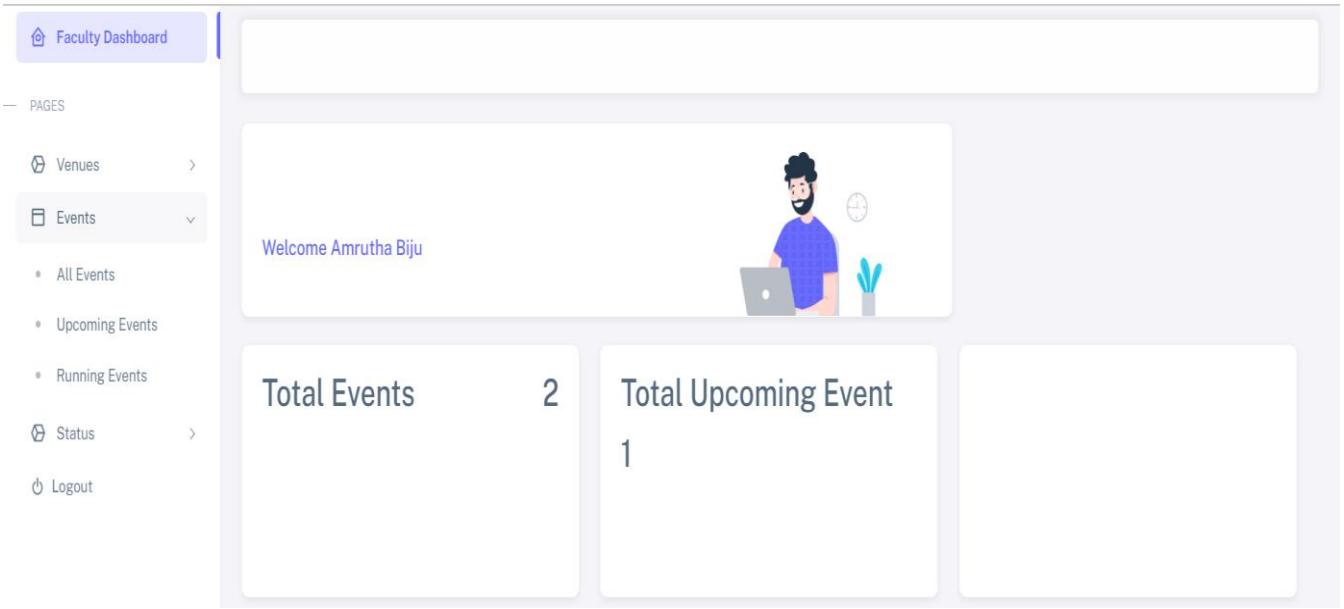


EVENT_APPROVAL

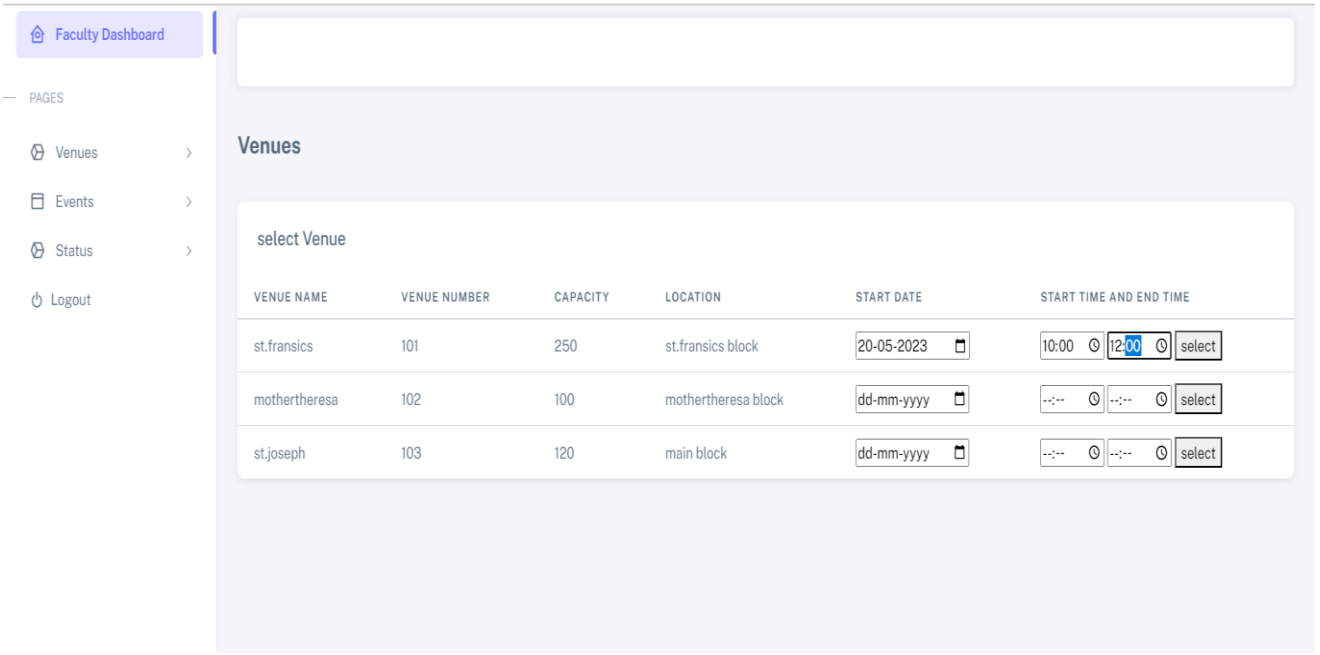




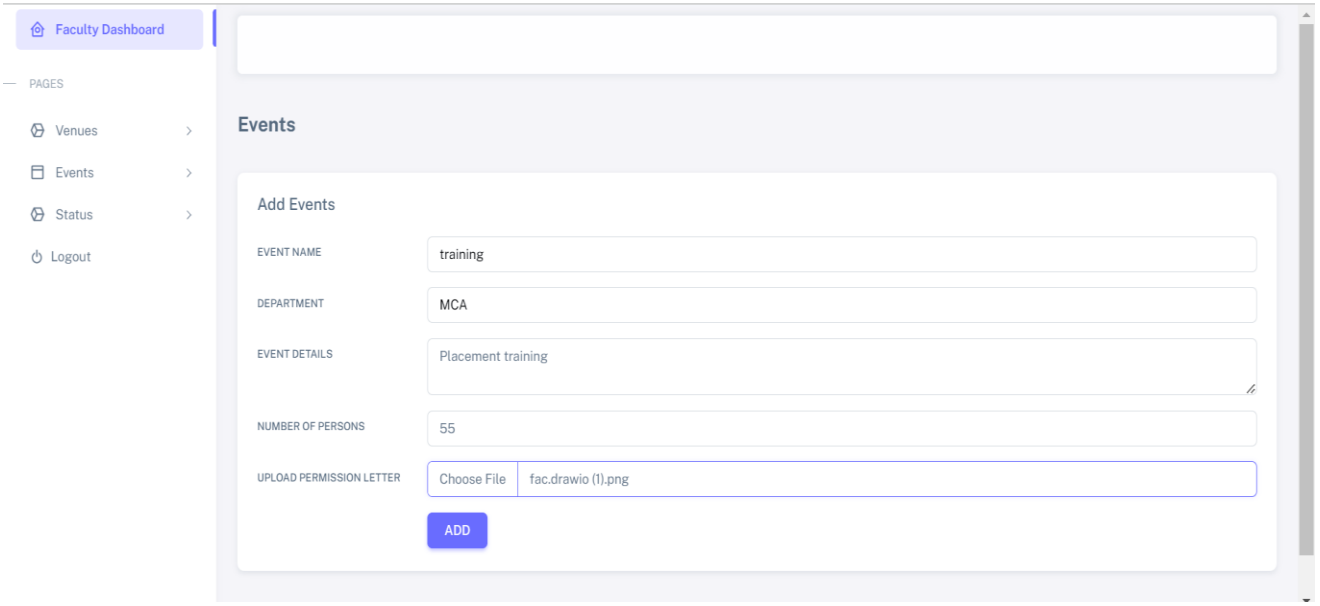
USER/FACULTY -DASHBOARD



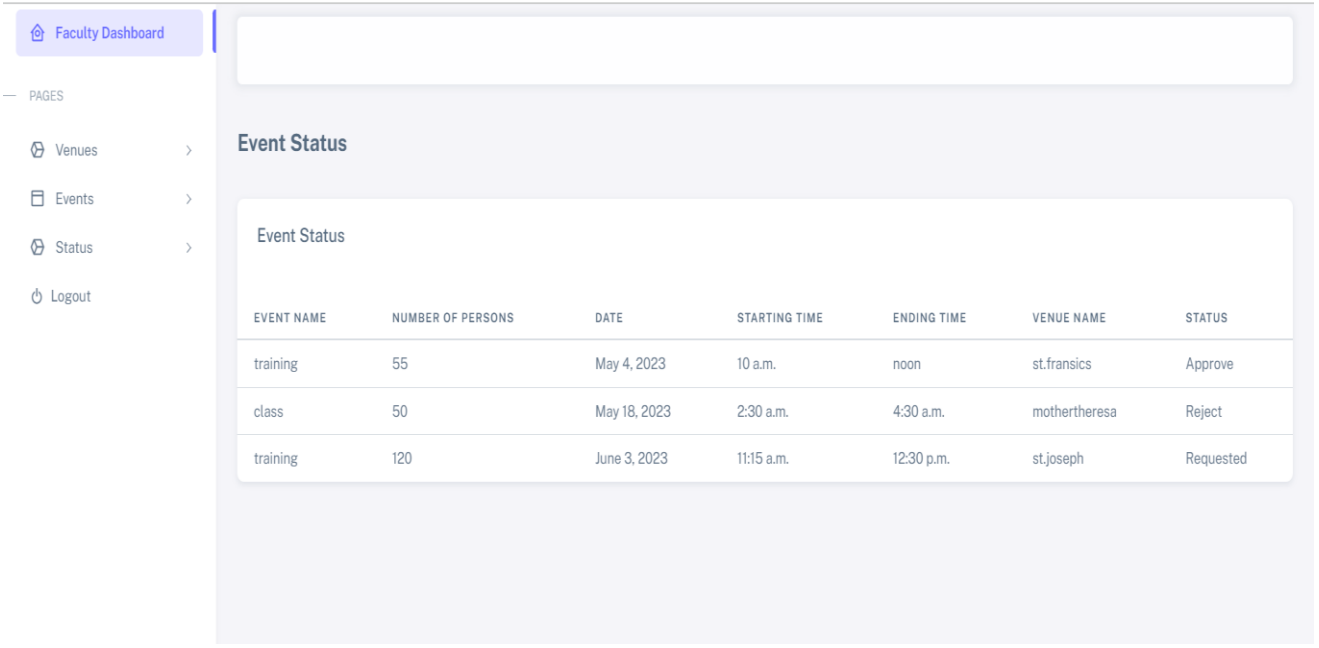
SELECT_VENUE



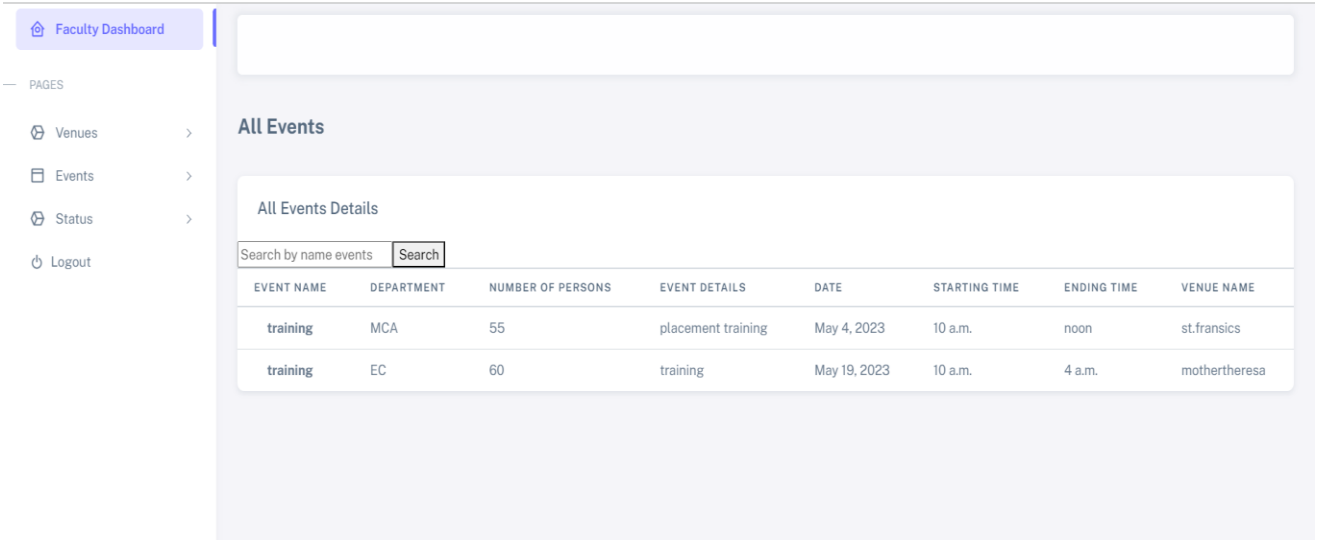
ADD_EVENTS/BOOK VENUE



STATUS OF BOOKING



VIEW ALL EVENTS



12.2 APPENDIX-B: SAMPLE CODE

ADMIN_VIEWS.PY

```
from django.contrib.auth.models import User
from django.shortcuts import render
from django.views import View
from django.views.generic import TemplateView
from SeminarApp.models import Faculty_Details, Hall_Name, Events_Details, UserType


class adminIndex(TemplateView):
    template_name = 'admin/index.html'


class User_Approvel(TemplateView):
    template_name = 'admin/user_vaify.html'
    def get_context_data(self, **kwargs):
        context = super(User_Approvel,self).get_context_data(**kwargs)
      user=Faculty_Details.objects.filter(user__last_name='0',user__is_staff='0',user__is_active='1')
        context['user'] = user
        return context
class user_view(TemplateView):
    template_name = 'admin/view_members.html'
    def get_context_data(self, **kwargs):
        context = super(user_view,self).get_context_data(**kwargs)
        user = Faculty_Details.objects.all()
```

```
   context['user'] = user
        return context
class ApproveView(View):
    def dispatch(self, request, *args, **kwargs):
        id = request.GET['id']
        user = User.objects.get(pk=id)
        user.last_name='1'
        user.save()
        return render(request,'admin/index.html',{'message':" Account Approved"})


class RejectView(View):
    def dispatch(self, request, *args, **kwargs):
        id = request.GET['id']
        user = User.objects.get(pk=id)
        user.last_name='0'
        user.is_active='0'
        user.save()
        return render(request,'admin/index.html',{'message':" Account Rejected"})


class Add_hall(TemplateView):
    template_name = 'admin/add_hall.html'
    def post(self, request, *args, **kwargs):
        hall = request.POST['hall']
```

```
no = request.POST['no']
        location = request.POST['location']
        capacity = request.POST['capacity']


 try:
            if Hall_Name.objects.filter(Hall_number=no):
                    return render(request, 'admin/add_hall.html', {'message': " Already added"})
            else:
                act = Hall_Name()
                act.capacity=capacity
                act.location=location
                act.Hall_name = hall
                act.Hall_number=no
                act.save()
                return render(request, 'admin/index.html', {'message': "Added"})
        except:
            act = Hall_Name()
            act.capacity = capacity
            act.location = location
            act.Hall_name = hall
            act.Hall_number = no
            act.save()
            return render(request, 'admin/index.html', {'message': "Added"})
```

```
class Event_Approvel(TemplateView):
    template_name = 'admin/event_approval.html'
    def get_context_data(self, **kwargs):
        context = super(Event_Approvel,self).get_context_data(**kwargs)
        user = Events_Details.objects.filter(status='Requested')
        context['user'] = user
        return context


class View_Booking(TemplateView):
    template_name = 'admin/view_booking.html'


    def get_context_data(self, **kwargs):
        context = super(View_Booking,self).get_context_data(**kwargs)
        user = Events_Details.objects.filter(status='Approve')
        context['user'] = user
        return context


class Details(TemplateView):
    template_name = 'admin/event_details.html'
    def get_context_data(self, **kwargs):
        id = self.request.GET['id']
        context = super(Details, self).get_context_data(**kwargs)
        single_view = Events_Details.objects.get(id=id)
```

```
 context['view'] = single_view
        return context


class Approve_Request(View):
    def dispatch(self, request, *args, **kwargs):
        id = self.request.GET['id']
        accept = Events_Details.objects.get(pk=id)
        accept.status = 'Approve'
        accept.save()
        return render(request, 'admin/index.html', {'message': "Approve"})
class Reject_Request(View):
    def dispatch(self, request, *args, **kwargs):
        id = self.request.GET['id']
        accept = Events_Details.objects.get(pk=id)
        accept.status = 'Reject'
        accept.save()
        return render(request, 'admin/index.html', {'message': "Reject"})
class UserReg(TemplateView):
    template_name = 'admin/reg.html'
    def post(self, request, *args, **kwargs):
        fullname = request.POST['name']
        email = request.POST['email']
        dept = request.POST['dept']
```

```
    desc = request.POST['desc']
        password = request.POST['password']
        if User.objects.filter(email=email):
            print('pass')
            return render(request, 'user_registration.html', {'message': "already added the username or email"})
        else:
          user=User.objects._create_user(username=email,password=password,first_name=fullname, email=email, last_name=1)
            user.save()
            student = Faculty_Details()
            student.user = user
            student.desc=desc
            student.dept=dept
            student.save()
            usertype = UserType()
            usertype.user = user
            usertype.type = "user"
            usertype.save()
            return render(request, 'admin/index.html', {'message': "successfully added"})


class Delete_Faculty(TemplateView):
    def dispatch(self,request,*args,**kwargs):
        id = request.GET['id']
```

```
        fa=User.objects.get(id=id)
        if request.method == 'POST':
            fa.delete()
            return render(request, 'admin/index.html',{'message': "Successfully Deleted"})
        context = {
            'fa': fa,
        }
        return render(request, 'admin/delete.html',context)


class Edit_faculty(TemplateView):
    template_name = 'admin/edit.html'


    def post(self, request, *args, **kwargs):
        id = self.request.GET['id']
        dept = request.POST['dept']
        desc = request.POST['desc']
        de = Faculty_Details.objects.get(pk=id)
        de.dept = dept
        de.desc=desc
        de.save()
        return render(request, 'admin/index.html', {'message': "Updated"})
```

ADMIN_URLS.PY

```
from django.urls import path

from SeminarApp.admin_views import adminIndex, User_Approvel, ApproveView, RejectView, Add_hall, Event_Approvel, Details, Approve_Request, Reject_Request, UserReg, View_Booking, user_view, Delete_Faculty, Edit_faculty

urlpatterns = [
    path('',adminIndex.as_view()),
    path('UserReg', UserReg.as_view()),
    path('user',User_Approvel.as_view()),
    path('Approve', ApproveView.as_view()),
    path('Reject', RejectView.as_view()),
    path('Add_hall',Add_hall.as_view()),
    path('Event_Approvel',Event_Approvel.as_view()),
    path('Details',Details.as_view()),
    path('app', Approve_Request.as_view()),
    path('rej',Reject_Request.as_view()),
    path('View_Booking',View_Booking.as_view()),
    path('user_view',user_view.as_view()),
    path('Delete',Delete_Faculty.as_view()),
    path('Edit',Edit_faculty.as_view())
]

def urls():
    return urlpatterns, 'admin','admin'
```

USER_VIEWS.PY

```
from django.core.files.storage import FileSystemStorage
from django.db.models import Q
from django.shortcuts import render
from django.views.generic import TemplateView
from datetime import date
from SeminarApp.models import Hall_Name, Faculty_Details, Events_Details
from datetime import datetime, timedelta


class userIndex(TemplateView):
    template_name = 'user/index.html'
    def get_context_data(self, **kwargs):
        today = date.today()
        pro = Events_Details.objects.filter(status='Approve')
        count=pro.count()
        prrr = Events_Details.objects.filter(date__gt=today, status='Approve')
        pr=prrr.count()
        context = {
            'pro': count,
            'pr':pr
        }
        return context
```

```
class all_events(TemplateView):
    template_name = 'user/all_events.html'
    def get_context_data(self, **kwargs):
        pro = Events_Details.objects.filter(status='Approve')
        context = {
            'pro': pro,
        }
        return context

    def post(self, request, *args, **kwargs):
        # template = loader.get_template('user/store.html')
        search = self.request.POST['search']
        pro = Events_Details.objects.filter(event_name__icontains=search)
        # return HttpResponse(template.render({"train": train}))
        return render(request, 'user/all_events.html', {'pro': pro})

class halls(TemplateView):
    template_name = 'user/halls.html'
    def get_context_data(self, **kwargs):
        pro = Hall_Name.objects.all()
context = {
            'pro': pro,
        }
```

```
   return context
    def post(self, request, *args, **kwargs):
        date = request.POST['date']
        stime = request.POST['sdate']
        etime = request.POST['edate']
        id=request.POST['id']
        if Events_Details.objects.filter(
                Q(starttime__range=(stime, etime)) |
                Q(endtime__range=(stime, etime)) |
                Q(starttime__lte=stime, endtime__gte=etime),date=date,hall_id=id
        ):
            return render(request, 'user/index.html',{'message': "Time period already Booked"})
        else:
            return render(request, 'user/add_event.html', {'sdate':stime,'edate':etime,'id':id,'date':date})


class add_event_details(TemplateView):
    template_name = 'user/add_event.html'
    def get_context_data(self, **kwargs):
        context = super(add_event_details, self).get_context_data(**kwargs)


   st = self.request.GET['sdate']
        et = self.request.GET['edate']
        date = self.request.GET['date']
```

```
  id = self.request.GET['id']
      context['id'] = id
      context['et'] = et
      context['date'] = date
      context['st'] = st
      return context
   def post(self, request, *args, **kwargs):
      st = request.POST['sdate']
      et = request.POST['edate']
      date = request.POST['date']
      id=request.POST['id']
      event_name = request.POST['name']
      department = request.POST['dept']
      event_details = request.POST['details']
      no_persons = request.POST['no']
      try:
         image=request.FILES['imagefield']
      except Exception as e:
         image=""
  if image is None:
         image=" "
      faculty = Faculty_Details.objects.get(user_id=self.request.user.id)
      add = Events_Details()
      add.event_name = event_name
```

```
        add.department = department
        add.event_details = event_details
        add.no_persons = no_persons
        add.starttime = st
        add.endtime = et
        add.date = date
        add.pletter=image
        add.hall_id =id
        add.faculty_id = faculty.id
        add.status = 'Requested'
        add.save()
        return render(request, 'user/index.html', {'message': "Added",'faculty':faculty})


class upcoming_events(TemplateView):
    template_name = 'user/upcoming_events.html'
    def get_context_data(self, **kwargs):
        today = date.today()
        pro = Events_Details.objects.filter(date__gt=today,status='Approve')


context = {
            'pro': pro,
        }
```

```
 return context
class running_events(TemplateView):
    template_name = 'user/upcoming_events.html'
    def get_context_data(self, **kwargs):
        today = date.today()
        pro = Events_Details.objects.filter(date=today,status='Approve')
        context = {
            'pro': pro,
        }
        return context


class status(TemplateView):
    template_name = 'user/my_status.html'
    def get_context_data(self, **kwargs):
        faculty = Faculty_Details.objects.get(user_id=self.request.user.id)
        pro = Events_Details.objects.filter(faculty_id=faculty.id)
        context = {
            'pro': pro,
        }
        return context
```

USER_URLS.PY

```
from django.urls import path

from . import user_views

from SeminarApp.user_views import userIndex, all_events, upcoming_events, running_events, halls, add_event_details, status


urlpatterns = [
    path('',userIndex.as_view()),
    # path('Add_events',Add_events.as_view()),
    path('all_events',all_events.as_view()),
    path('upcoming_events',upcoming_events.as_view()),
    path('running_events',running_events.as_view()),
    path('halls',halls.as_view()),
    path('add_event_details',add_event_details.as_view()),
    path('status',status.as_view()),
]
def urls():
    return urlpatterns, 'user','user'
```

VIEWS.PY

```
import io

from django.contrib.auth import login, authenticate

from django.contrib.auth.models import User
```

```
from django.shortcuts import render, redirect
from django.views.generic import TemplateView
from SeminarApp.models import UserType,Faculty_Details
# Create your views here.
class Index(TemplateView):
    template_name = 'index.html'
class Login(TemplateView):
    template_name = 'login1.html'
    def post(self, request, *args, **kwargs):
        email = request.POST['email']
        password = request.POST['password']
        user = authenticate(username=email, password=password)
        if user is not None:
            login(request, user)
            if user.last_name == '1':
                if user.is_superuser:
                    return redirect('/admin')
                elif UserType.objects.get(user_id=user.id).type == "user":
                    return redirect('/user')
            else:
                return render(request,'login1.htm', {'message': " User Account Not Authenticated"})
        else:
            return render(request,'login1.html', {'message': "Invalid Username or Password"})
```