1. INTRODUCTION

1.1. Scope of The Project

The objective of this application is to develop a system that effectively manages all the data related to the various events that take place in an organization. The purpose is to maintain a centralized database of all event related information. The goal is to support various functions and processes necessary to manage the data efficiently.

1.2. Existing System

This existing system is not providing secure registration and profile management of all the users properly.This system is not providing on-line Help.This system doesn’t provide tracking of users activities and their progress.This manual system gives us very less security for saving data and some data may be lost due to mismanagement.This system is not providing event management through internet.This system is not providing proper events information.The system is giving manual information through the event management executer.

1.3. Proposed System

The development of this new system contains the following activities, which try to automate the entire process keeping in the view of database integration approach.This system maintains employee’s personal, address, and contact details.This system will provide on line help and search capabilities. User friendliness is provided in the application with various controls provided by system rich user interface.Authentication is provided for this application only registered users can access. event information files can be stored in centralized database which can be maintained by the system. This system provides the employees to manage the events systematically.

2. SYSTEM ANALYSIS

2.1 FEASIBILITY STUDY

A feasibility study is a high-level capsule version of the entire System analysis and Design Process. The study begins by classifying the problem definition. Feasibility is to determine if it's worth doing. Once an acceptance problem definition has been generated, the analyst develops a logical model of the system. A search for alternatives is analyzed carefully. There are 3 parts in feasibility study.

2.1.1 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 the users.
- Have the users been involved in planning and development of the Project.

2.1.2 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?
- The project is developed on vs code with 128 MB RAM.
- The environment required in the development of system is any windows platform.
- The observer pattern along with factory pattern will update the results eventually.
- The language used in the development is html , css and javascript for front end and php and mysql for backend.

2.1.2 Economical Feasibility

To decide whether a project is economically feasible, to consider various factors as cost benefit analysis, long-term returns and maintenance costs.

2.2 FUNCTIONAL REQUIREMENTS

Functional requirement defines a function of a software system or its component. A function is described as a set of inputs, the behavior, and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioral requirements describing all the cases where the system uses the functional requirements are captured in use cases.

NUMBER OF MODULES

The system after careful analysis has been identified to be presented with the following modules:
The Modules involved are

Event Management Module :

The employees who can use the application their personal information, contact information and other information etc…
This module consist of events information. All the information like type of the event, incharge of the event. These all information is maintained here.
Here in this module application is maintaining the total information of the event and the resources. Like type of the resource, resource details and resource management.

Event Task Manager Module :

The module is having the information of the events and their task manager details. Events task are maintained here, the total details of the events and incharge details maintained. In this module the information of the event and their details like, what are the events are there and from which event and to which event the movement is going on , which date , status of the event etc.

Scheduling :

This module consists of events information. All the information like type of the event, in charge of the event. These all information is maintained here.

Security & Authentication Module :

Security & Authentication module is main module which can provide security for entire processing of the system by using username, password, login, password modifications etc.

Reports Module :

In this module system can generate different type of the reports.

Payment Module:

In this module user can pay the amount.

2.3 NON-FUNCTIONAL REQUIREMENTS

Performance Requirements:

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely with the users of the existing system to give the requirement specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

The requirement specification for any system can be broadly stated as given below:

- The system should be able to interface with the existing system
- The system should be accurate
- The system should be better than the existing system

Reliability :

In this system reliability means the mail which is send by the source must reach the target user with any modification and accurate.

Security :

The web server and database server should be protected from hacking, virus etc

Portability :

The application will be developed using standard open source software (Except Oracle) like Java, tomcat web server, Internet Explorer Browser etc these software will work both on Windows and Linux o/s. Hence portability problems will not arise.

Availability :

This software will be available always.

Maintainability :

In this system the presentation layer is clearly separated from the service layer. So any modification in future will be done with less efforts. The database will be running at the server. Users access these forms by using the user-ids and the passwords.

2.4 HARDWARE REQUIREMENTS:

Processor	:	Intel P-IV based system
Processor Speed	:	2.0. GHz
RAM	:	1GB
Hard Disk	:	40GB to 80GB

2.5 SOFTWARE REQUIREMENTS:

Database	:	xampp
Server	:	Apache
Frontend	:	HTML
Scripting language	:	Java Script
Web Technologies	:	php and mysql
IDE	:	vs code
Technology	:	web development

2.6 DFD’s (Data Flow Diagrams)

Data Flow Diagram for Context Level

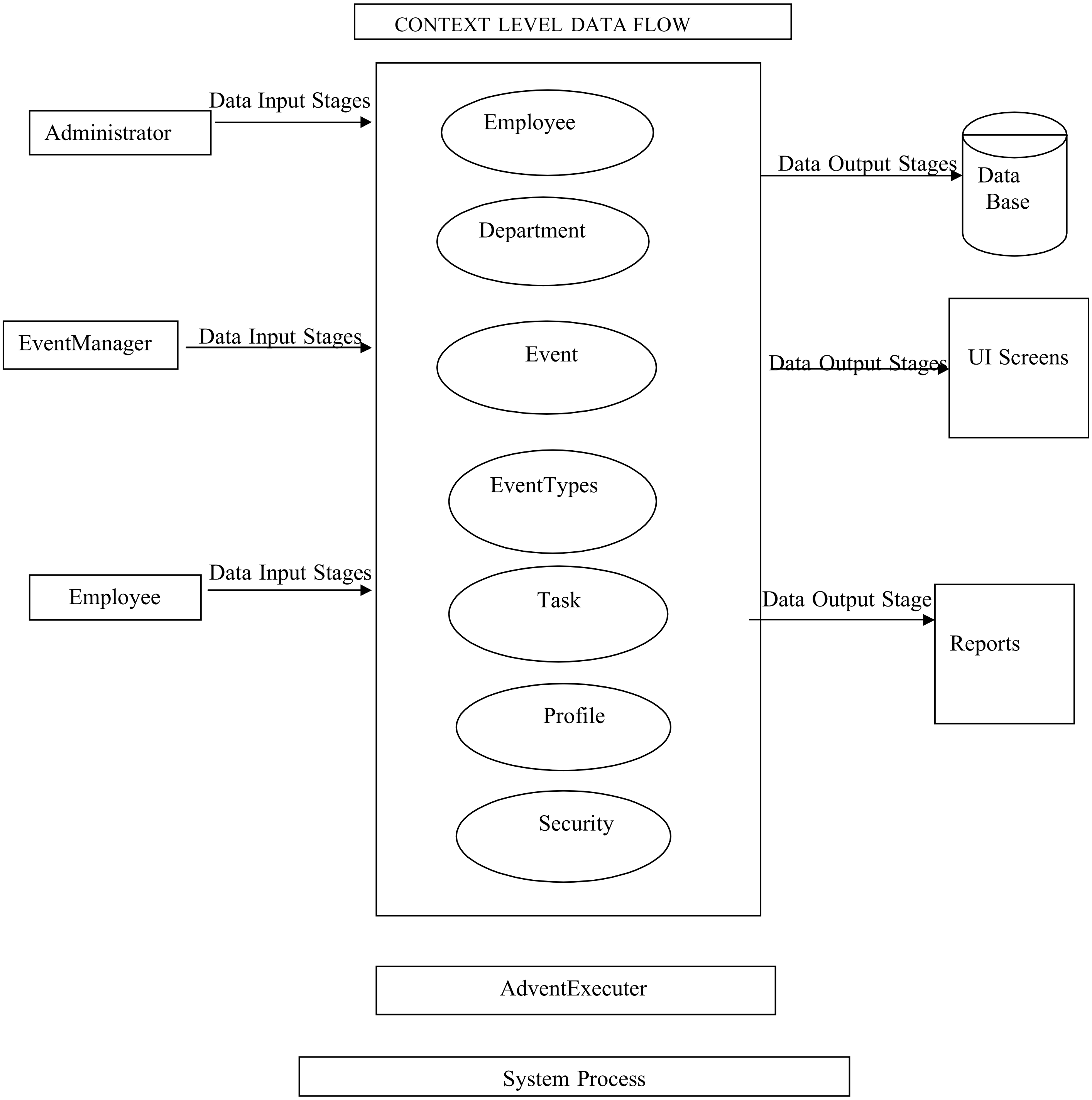


Fig: 2.6.1 Data flow diagram for Context Level

Data Flow Diagram for Level-1

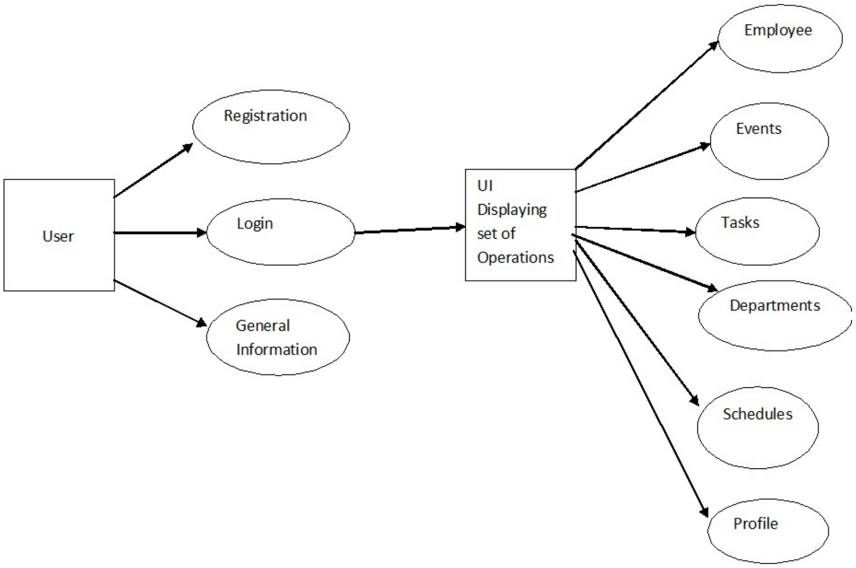


Fig: 2.6.2 Data Flow Diagram for Level-1

Data Flow Diagram for Administrator Context Level-1

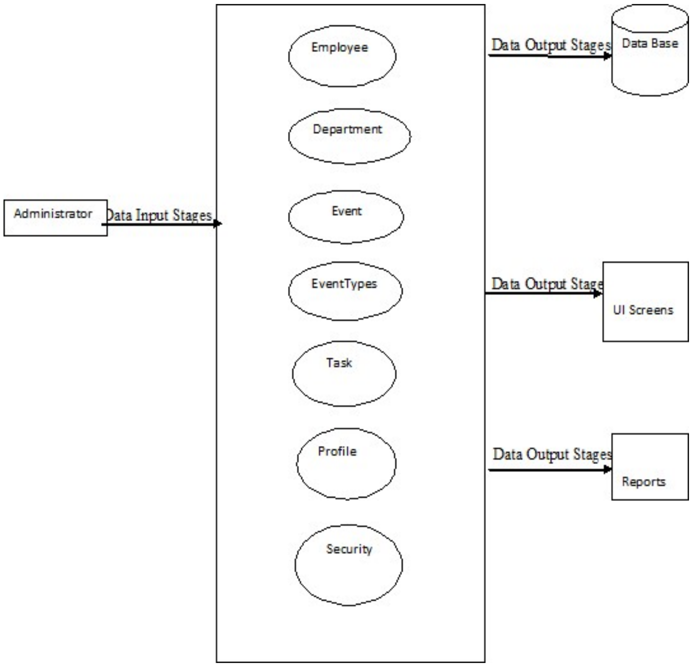


Fig: 2.6.3 Data Flow Diagram for Administrator Context Level-1

Data Flow Diagram for Level 1.0

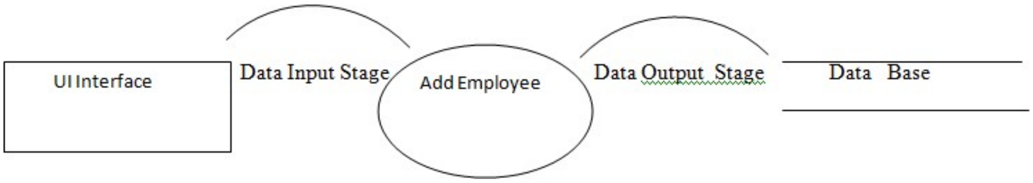


Fig: 2.6.4 Data Flow Diagram for Level 1.0

Data Flow Diagram for Level 1.0.1

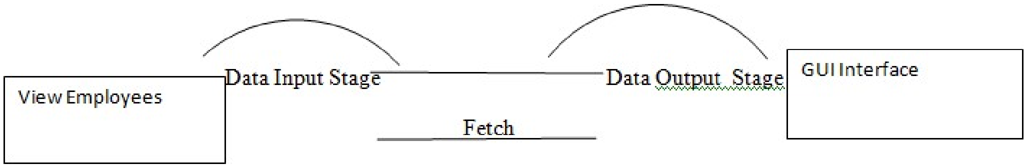


Fig: 2.6.5 Data Flow Diagram for Level 1.0.1

Data Flow Diagram for Level 1.1

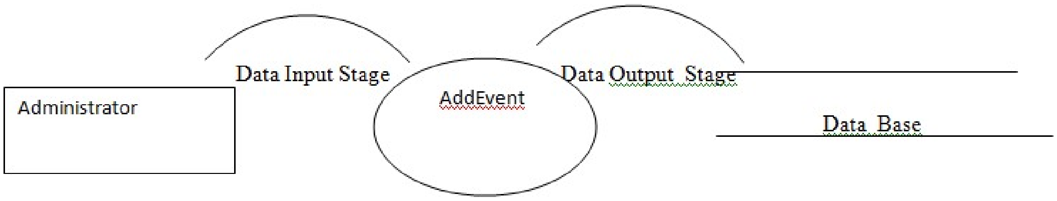


Fig: 2.6.6 Data Flow Diagram for Level 1.1

Data Flow Diagram for Level 1.1.0

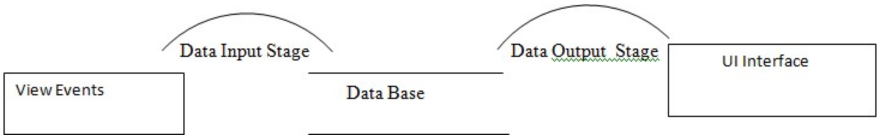


Fig: 2.6.7 Data Flow Diagram for Level 1.1.0

Data Flow Diagram for Level 1.2

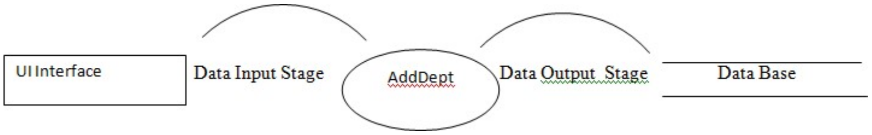


Fig: 2.6.8 Data Flow Diagram for Level 1.2

Data Flow Diagram for Level 1.3

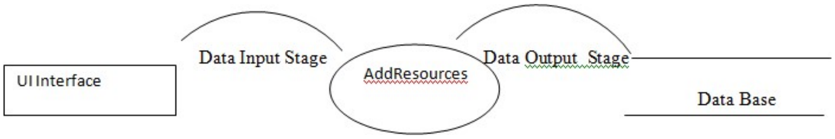


Fig: 2.6.9 Data Flow Diagram for Level 1.3

Data Flow Diagram for Level 1.4

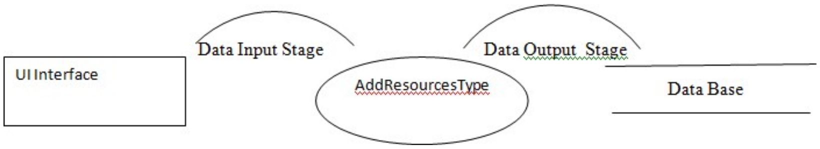


Fig: 2.6.10 Data Flow Diagram for Level 1.4

Data Flow Diagram for Level 1.5

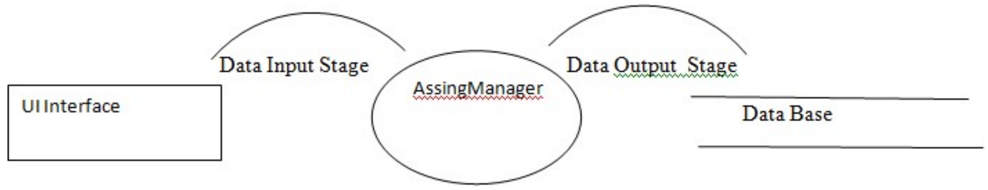


Fig: 2.6.11 Data Flow Diagram for Level 1.5

Data Flow Diagram for Event Manger Context Level-1

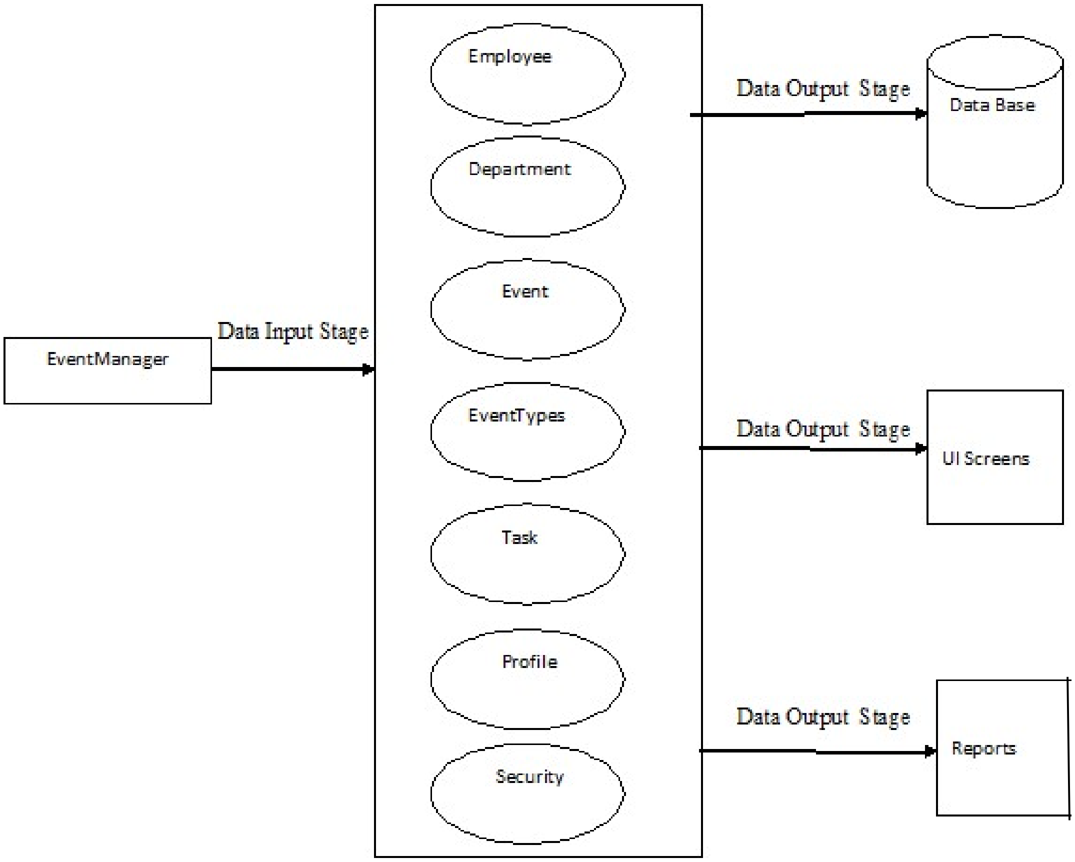


Fig: 2.6.12 Data Flow Diagram for Event Manger Context Level-1

Data Flow Diagram for Level 1.0

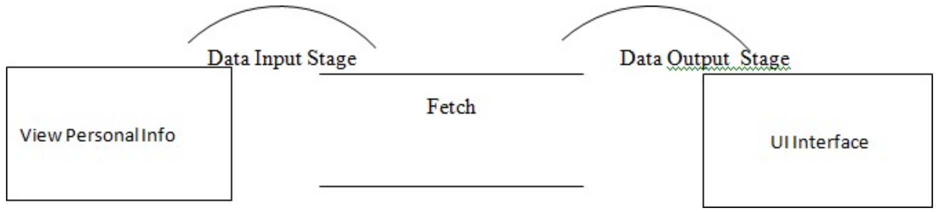


Fig: 2.6.13 Data Flow Diagram for Level 1.0

Data Flow Diagram for Level 1.1

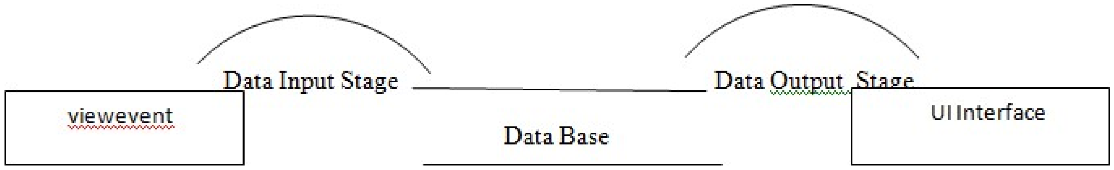


Fig: 2.6.14 Data Flow Diagram for Level 1.1

Data Flow Diagram for Level 1.2

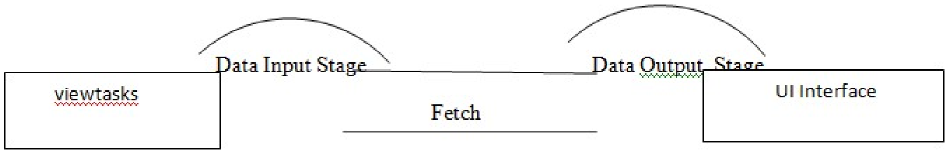


Fig: 2.6.15 Data Flow Diagram for Level 1.2

Data Flow Diagram for Level 1.3

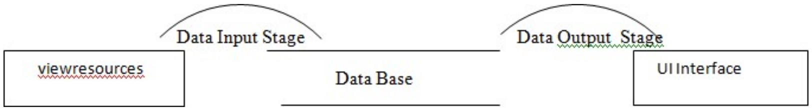


Fig: 2.6.16 Data Flow Diagram for Level 1.3

Data Flow Diagram for Level 1.4

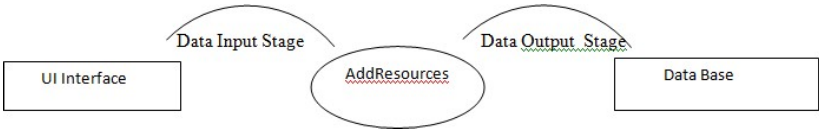


Fig: 2.6.17 Data Flow Diagram for Level 1.4

Data Flow Diagram for Employee Context Level-1

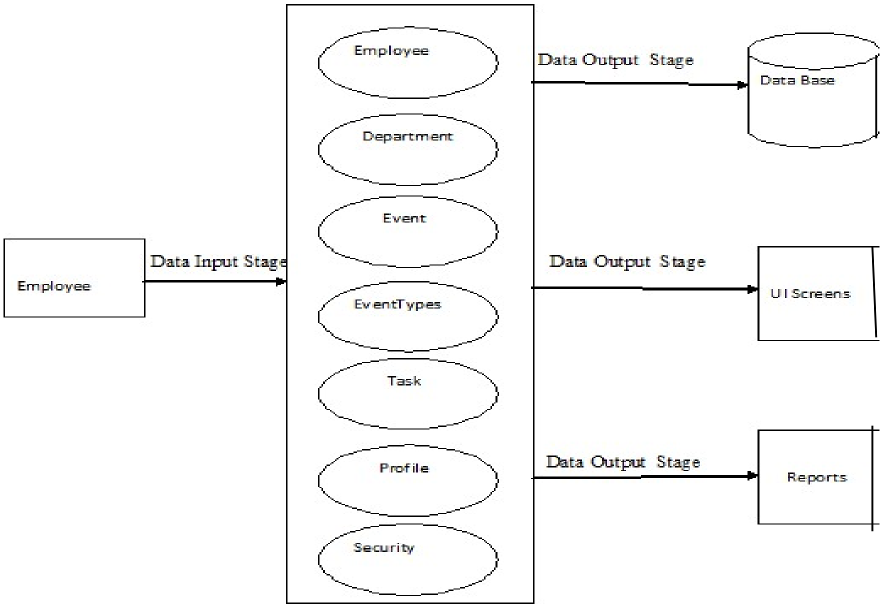


Fig: 2.6.18 Data Flow Diagram for Employee Context Level-1

Data Flow Diagram for Level 1.0

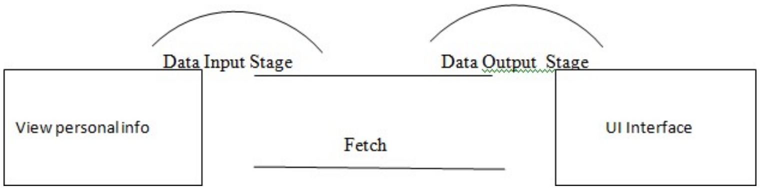


Fig: 2.6.19 Data Flow Diagram for Level 1.0

Data Flow Diagram for Level 1.1

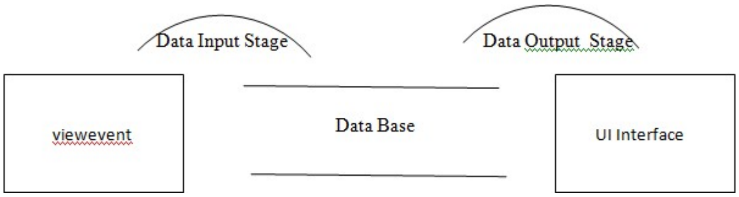


Fig: 2.6.20 Data Flow Diagram for Level 1.1

Data Flow Diagram for Level 1.2

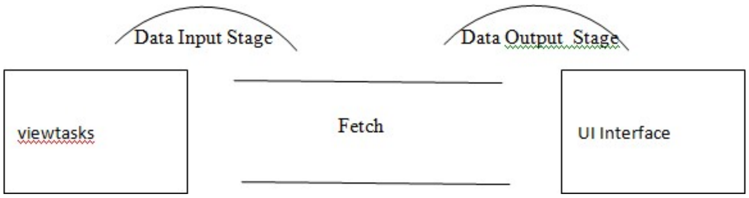


Fig: 2.6.21 Data Flow Diagram for Level 1.2

Data Flow Diagram for Level 1.3

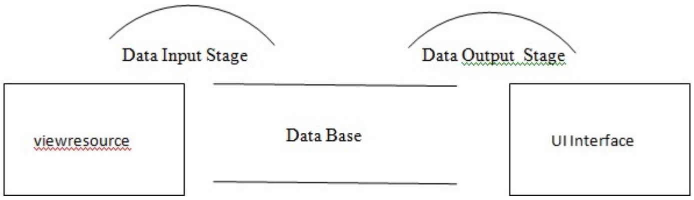


Fig: 2.6.22 Data Flow Diagram for Level 1.3

3. SYSTEM DESIGN

3.1 UML DIAGRAMS

The unified modeling language allows the software engineer to express an analysis model using the modeling notation that is governed by a set of syntactic semantic and pragmatic rules.

3.1.1 Class Diagram :

The class diagram shows a set of classes, interfaces, collaborations and their relationships.

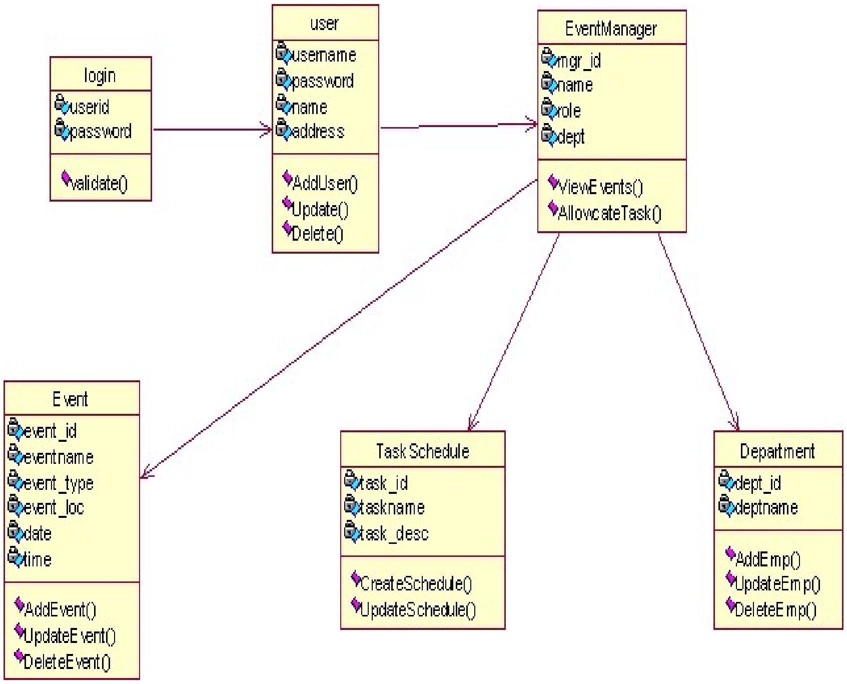


Fig: 3.1.1.1 Class diagram for event management

3.1.2 Use case diagrams:

Use case diagram consists of actors, use cases and their relationships. These diagrams are especially important in organizing and modeling the behaviors of a system.

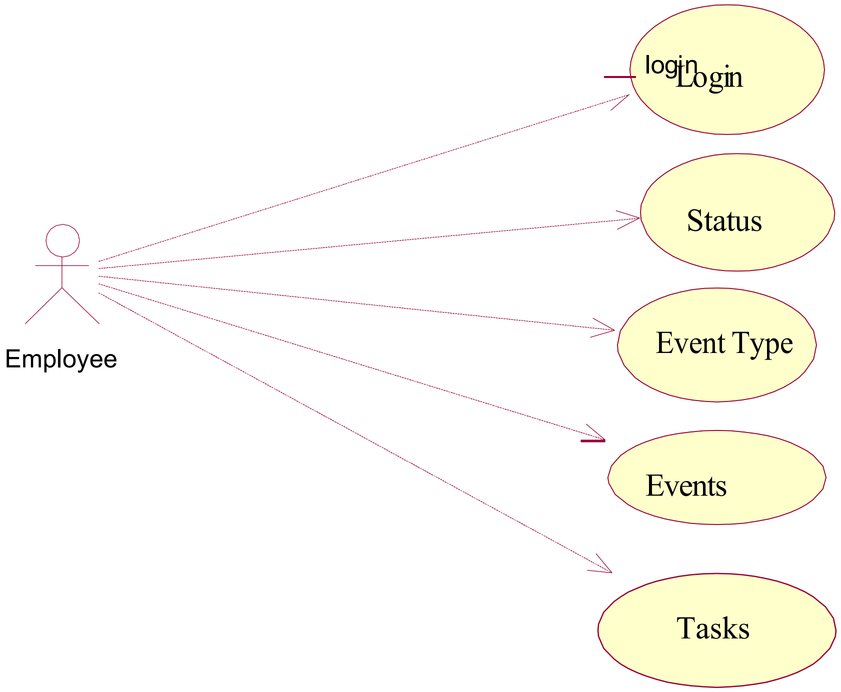


Fig: 3.1.2.1 Use case diagram for Employee

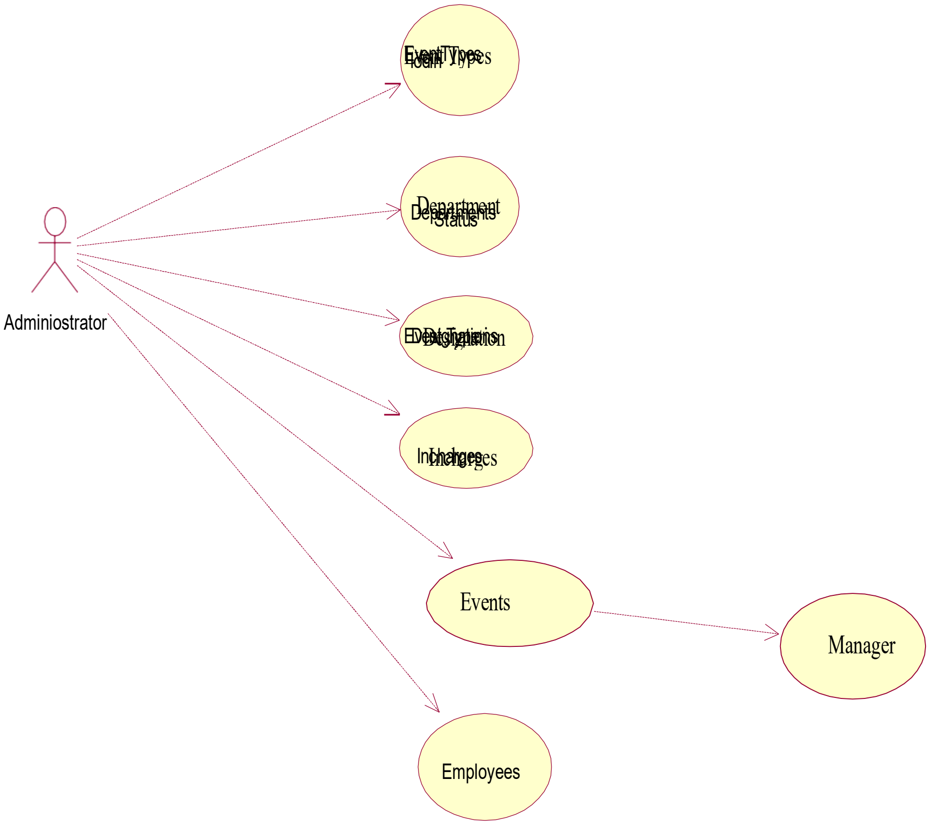


Fig: 3.1.2.2 Use case diagram for Administrator

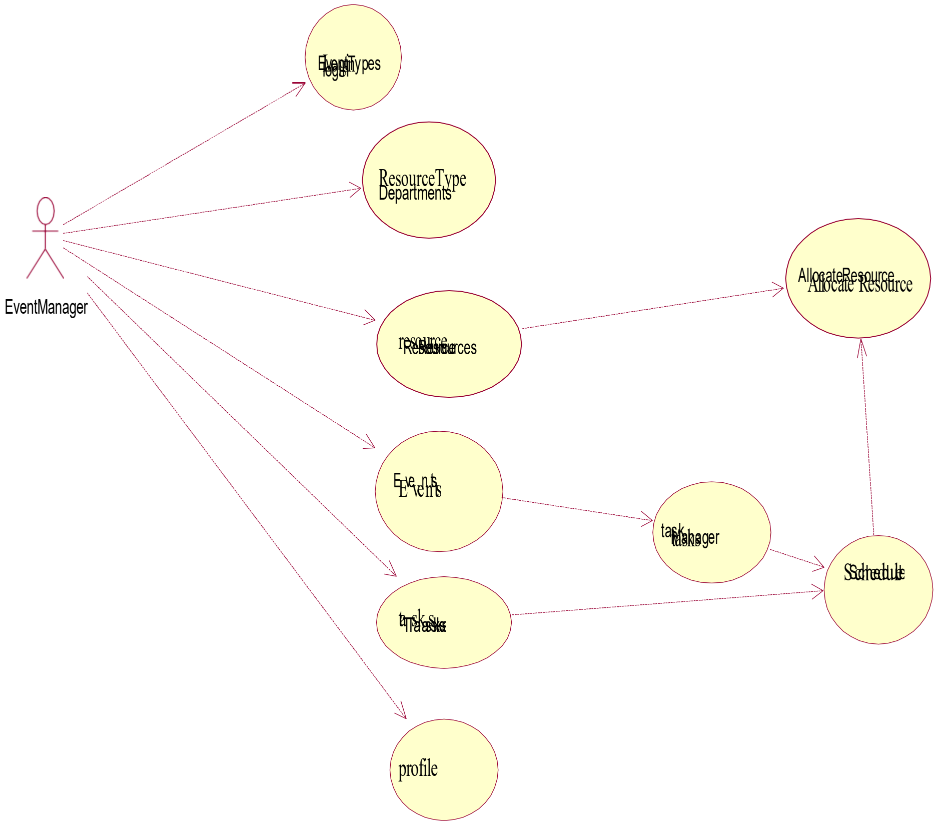


Fig: 3.1.2.3 Use case diagram for Task Manager

3.1.3 Sequence Diagram:

The Sequence diagram is an interaction diagram that emphasizes the time-ordering of messages.

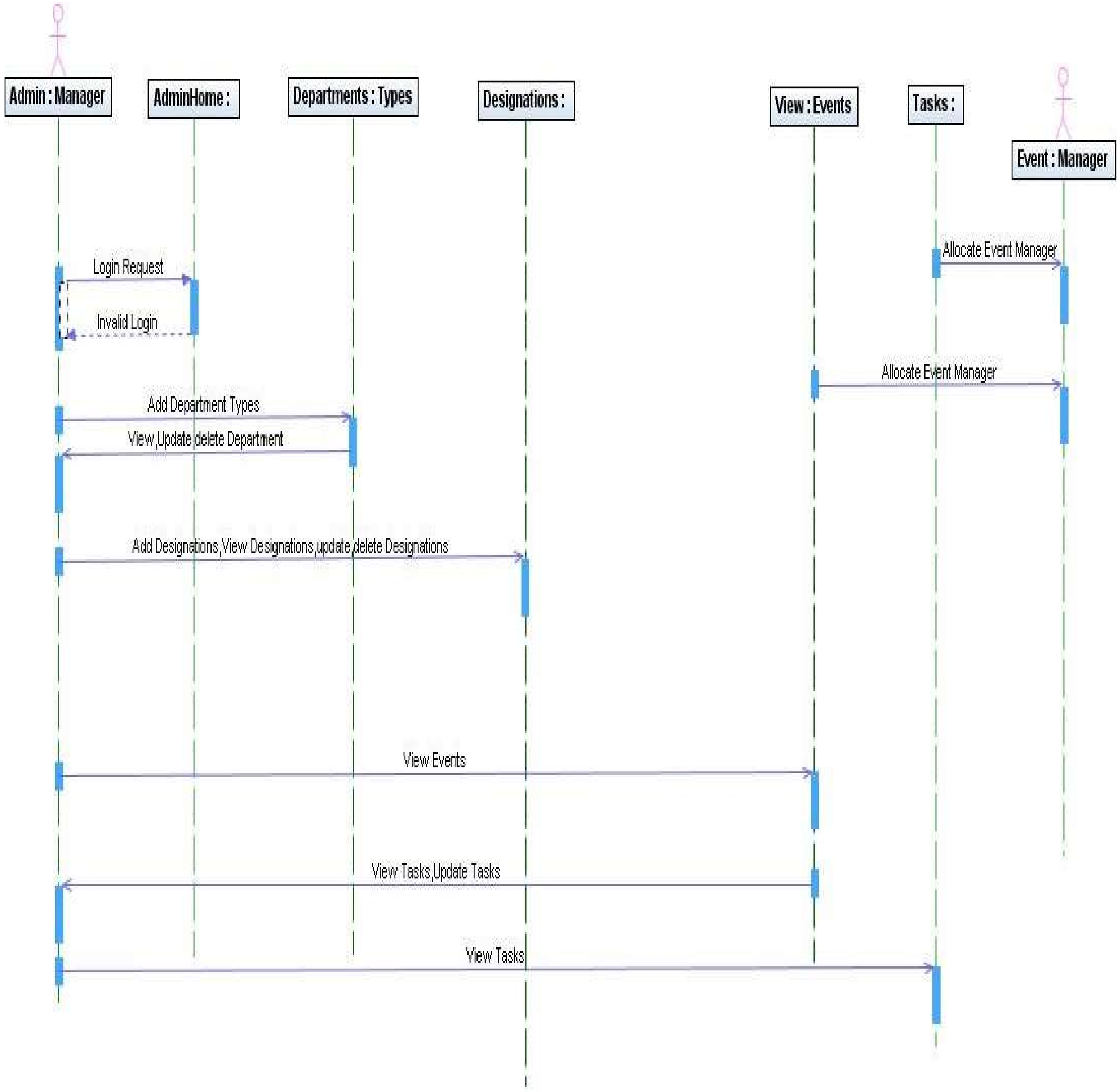


Fig: 3.1.3.1 Sequence diagram for Admin

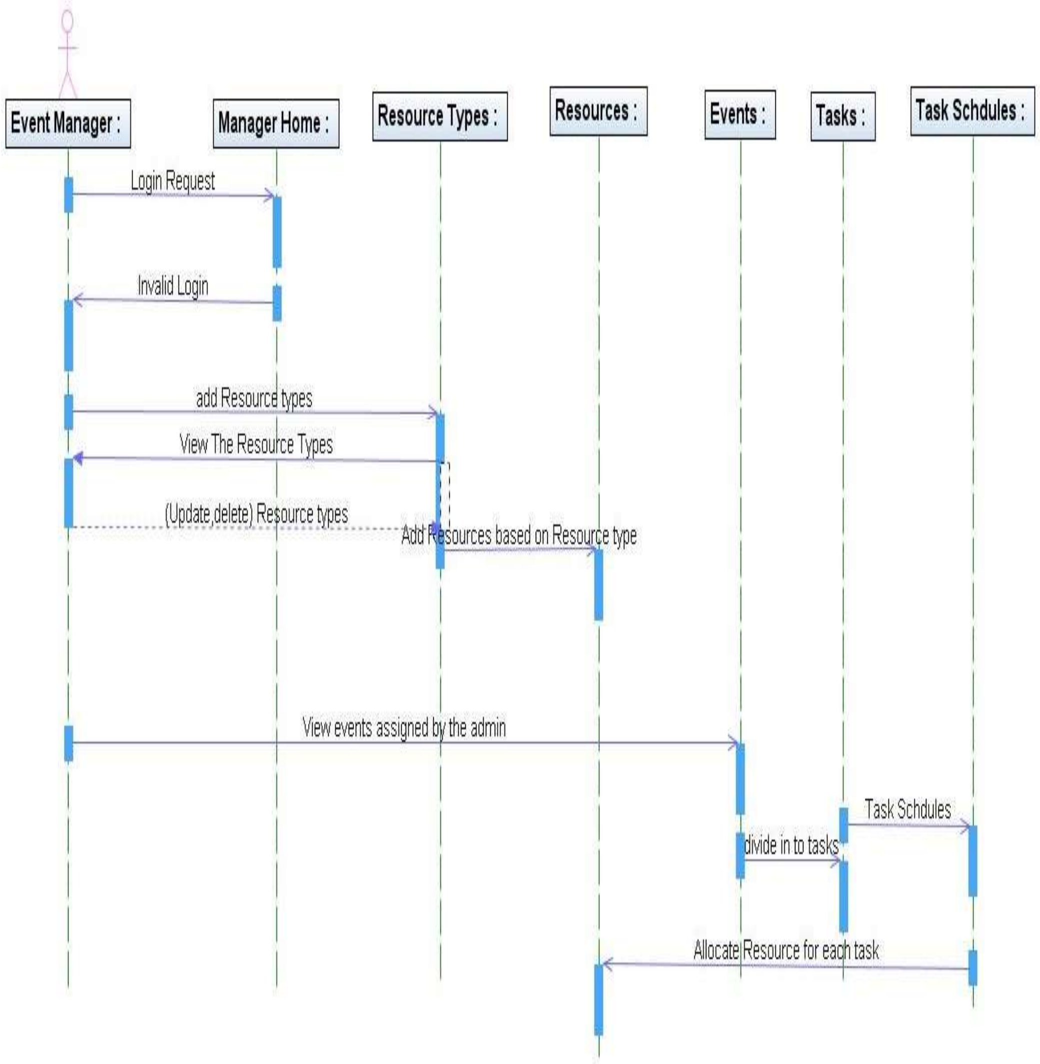


Fig: 3.1.3.2 Sequence diagram for Event Manager

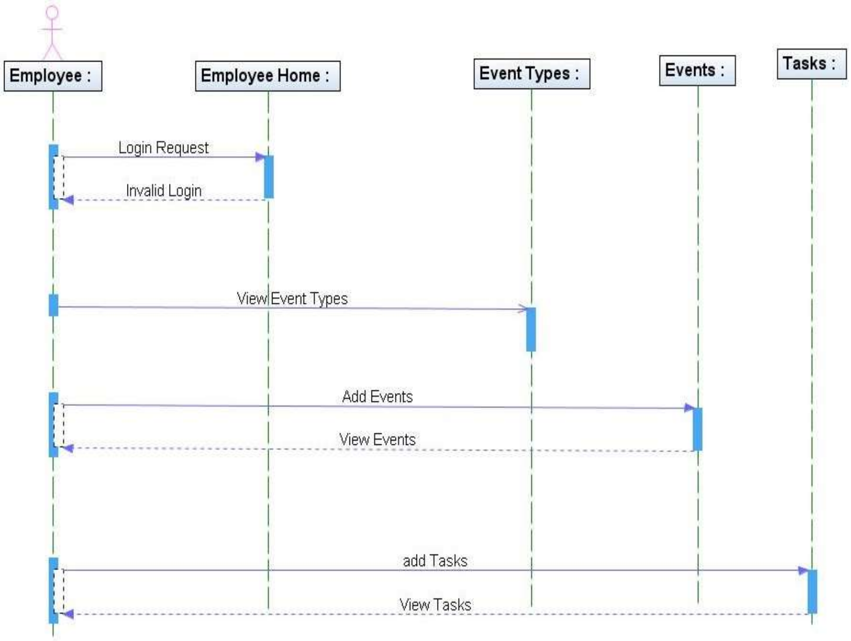


Fig: 3.1.3.3 Sequence diagram for Employee

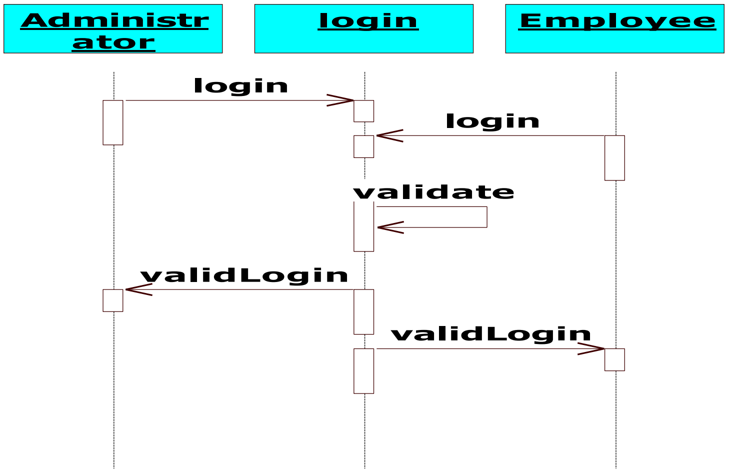


Fig: 3.1.3.4 Sequence diagram for Login

3.1.4 Activity Diagram

An Activity diagram is a special kind of a state chart diagram that shows the flow from activity to activity within a system.

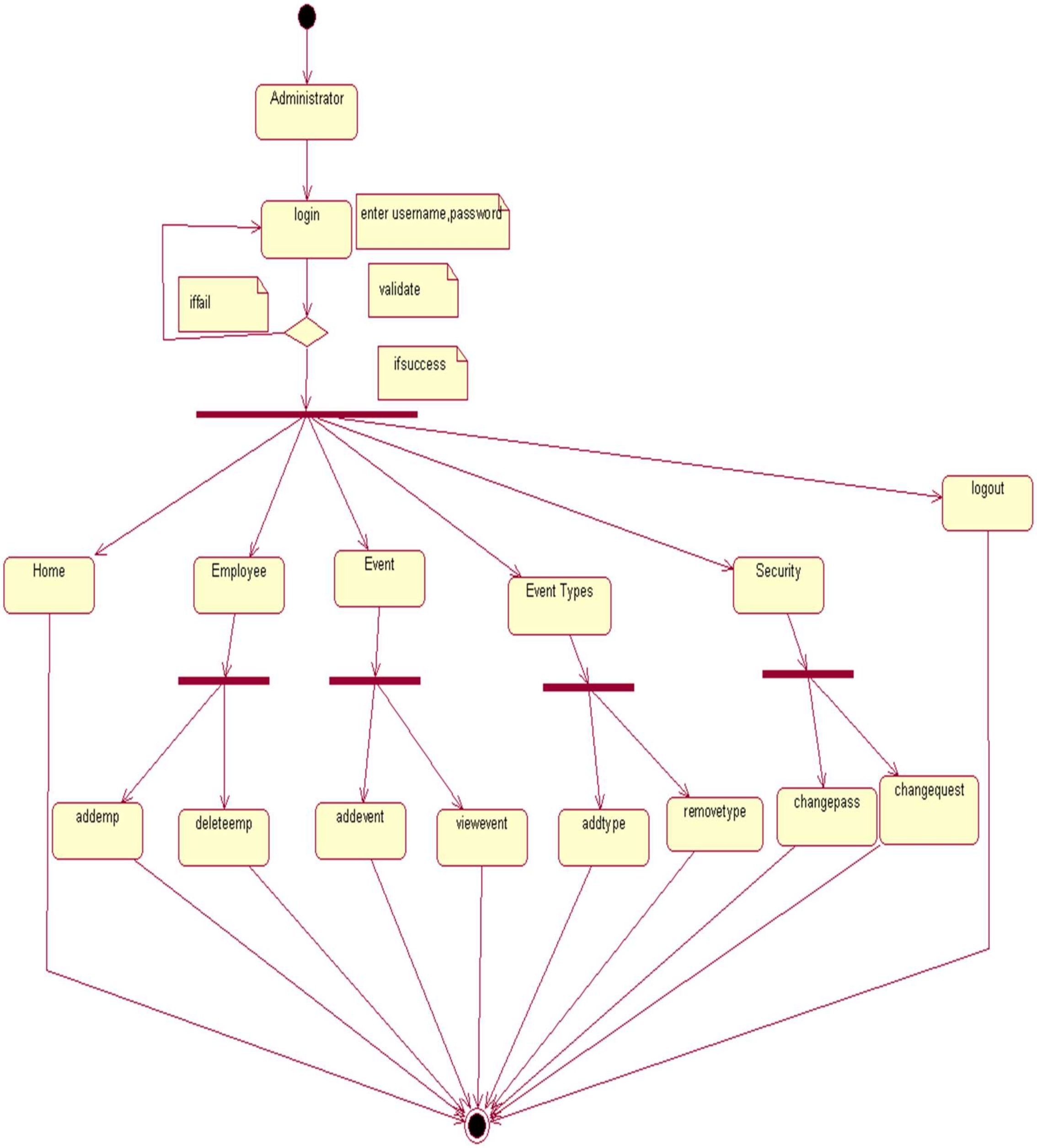


Fig: 3.1.4.1 Activity diagram for Admin

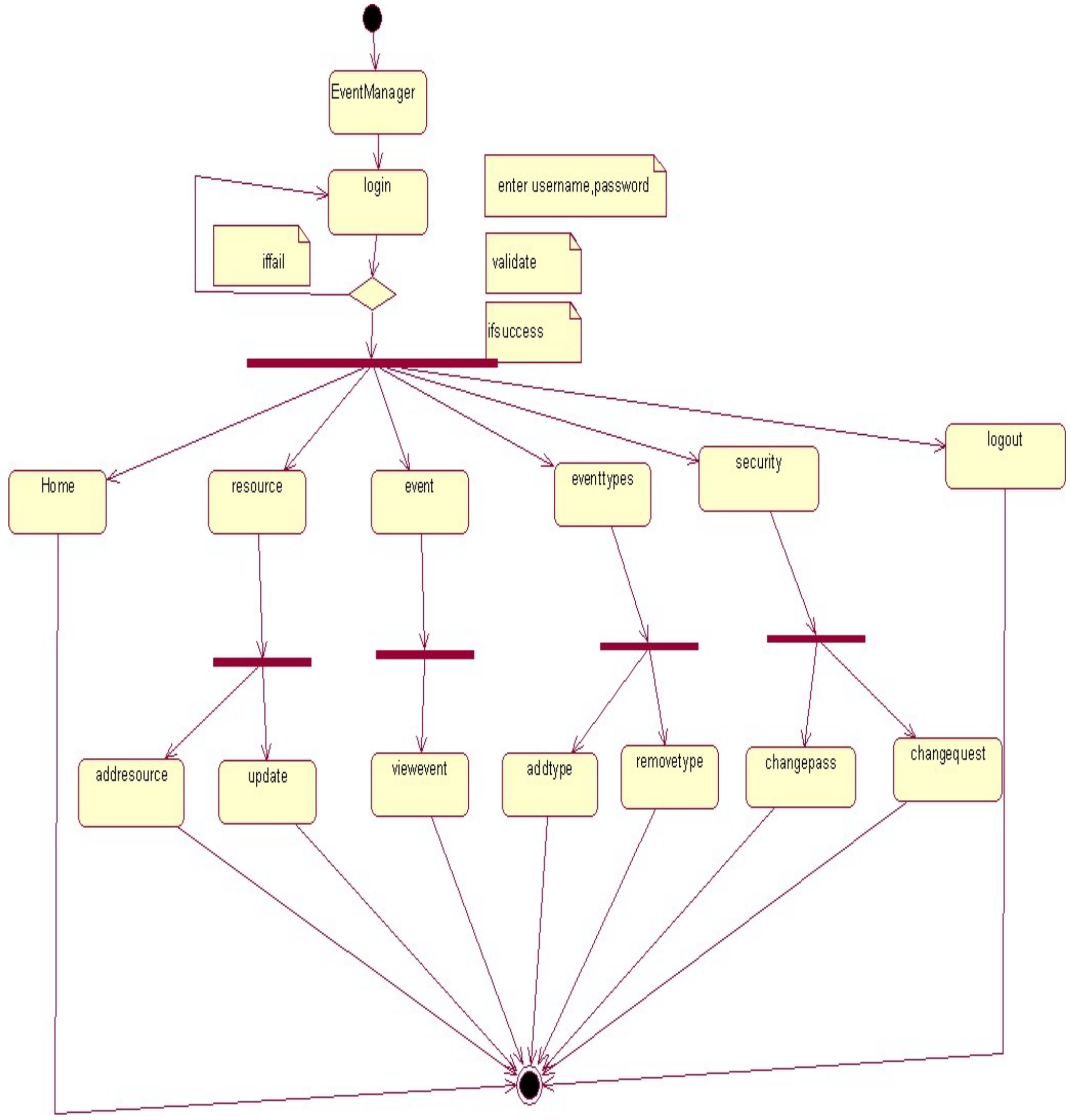


Fig: 3.1.4.2 Activity diagram for Task Manager

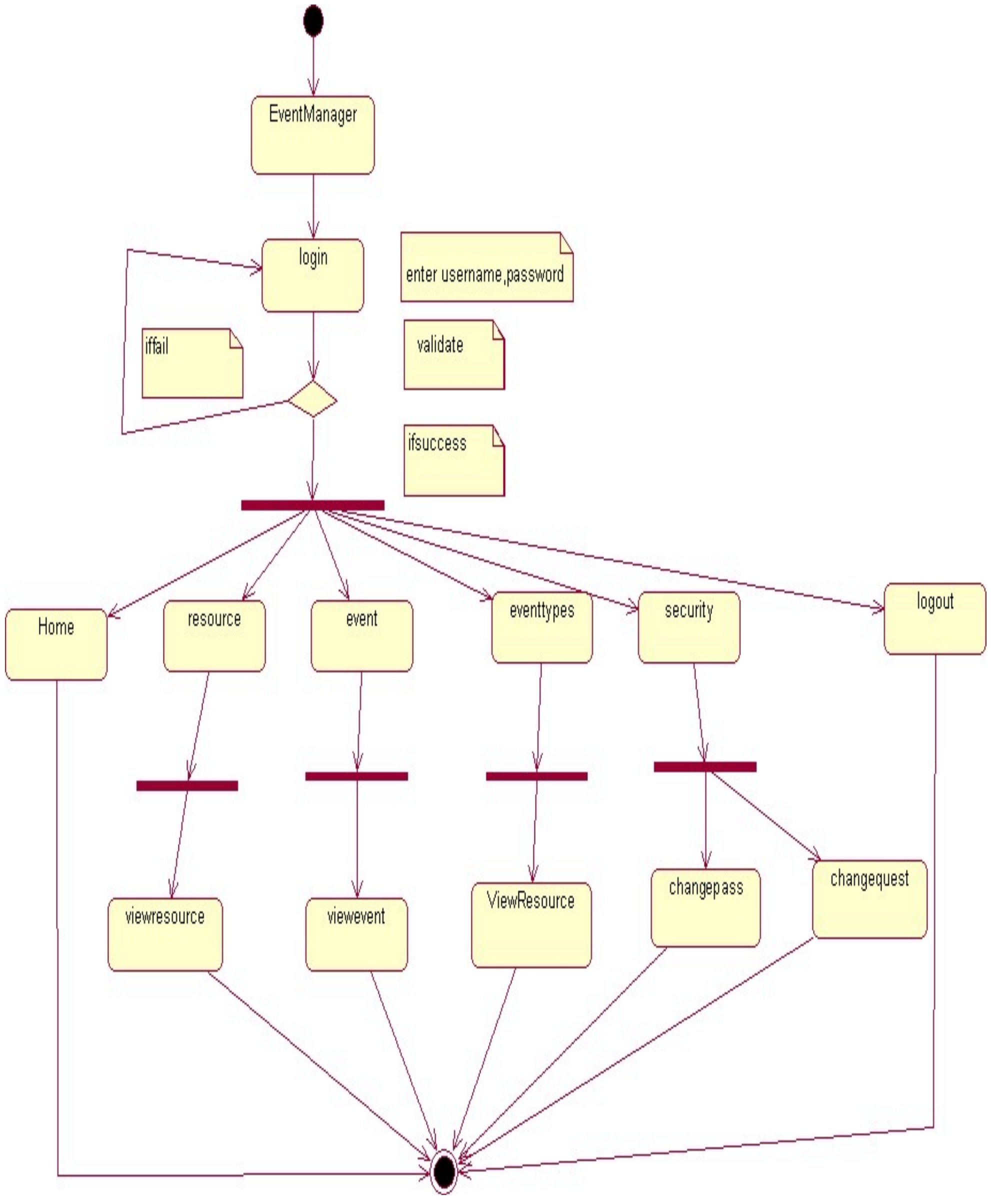


Fig: 3.1.4.3 Activity diagram for Employee

3.1.5 Collaboration Diagram

The second interaction diagram is collaboration diagram. It shows the object organization as shown below. Here in collaboration diagram the method call sequence is indicated by some numbering technique as shown below.

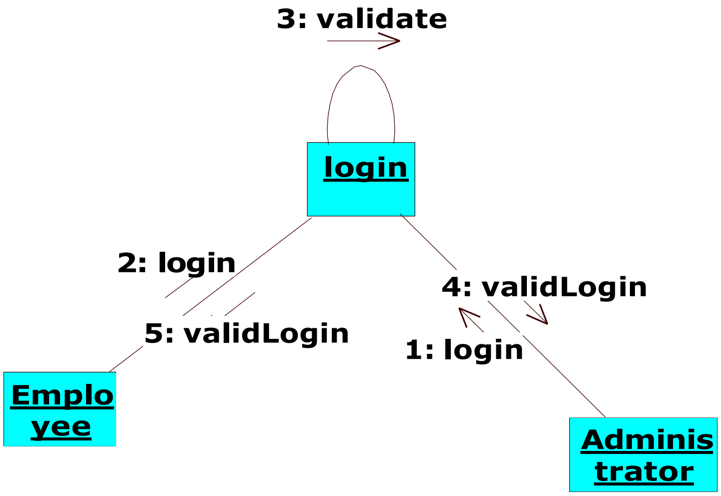


Fig: 3.1.5.1 Collaboration Diagram for Login

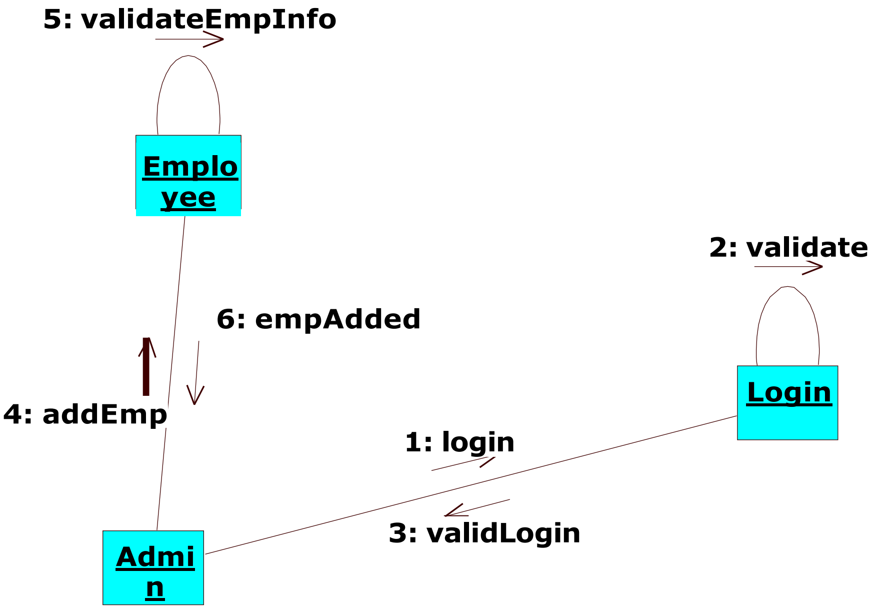


Fig: 3.1.5.2 Collaboration Diagram for Employee Registration

3.2 ER-DIAGRAM

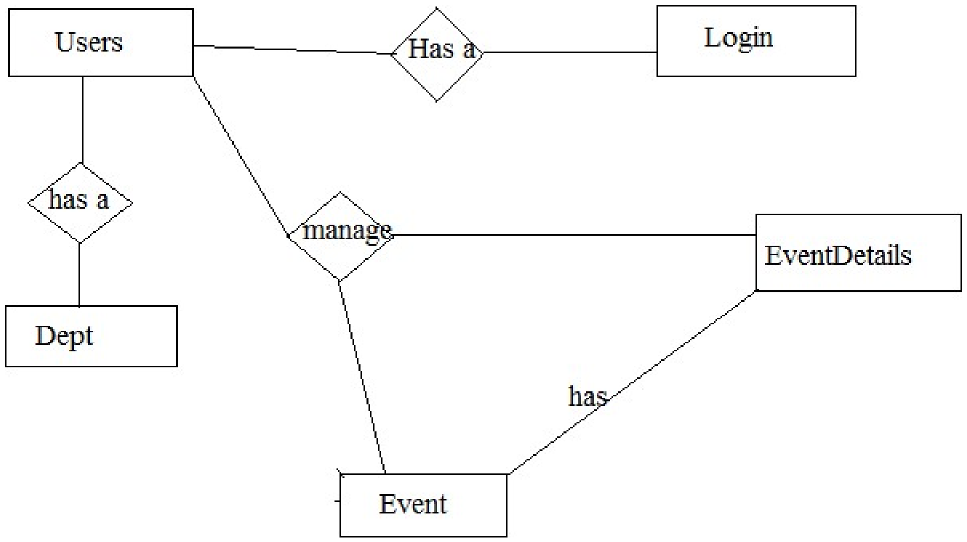


Fig: 3.1.3 ER diagram for event management

3.3 Tables

3.3.1 User Table

Name	Data Type	Size
username	Text	20
password	Text	20

3.3.2 Event Table

Name	Data Type	Size
eid	Number	14
customer_Name	Text	14
Event_Type	Text	14
Address	Text	14
cno	Number	14
Event_time	Text	14
event_date	Text	14
Total_amount	Number	14
Adv_amount	Number	14
status	Text	14
Rdate	Number	14

3.3.3 Pay Table

Name	Data Type	Size
cardno	Number	20
bankname	Text	20
cvvno	Number	20

3.4 SYSTEM ARCHITECTURE

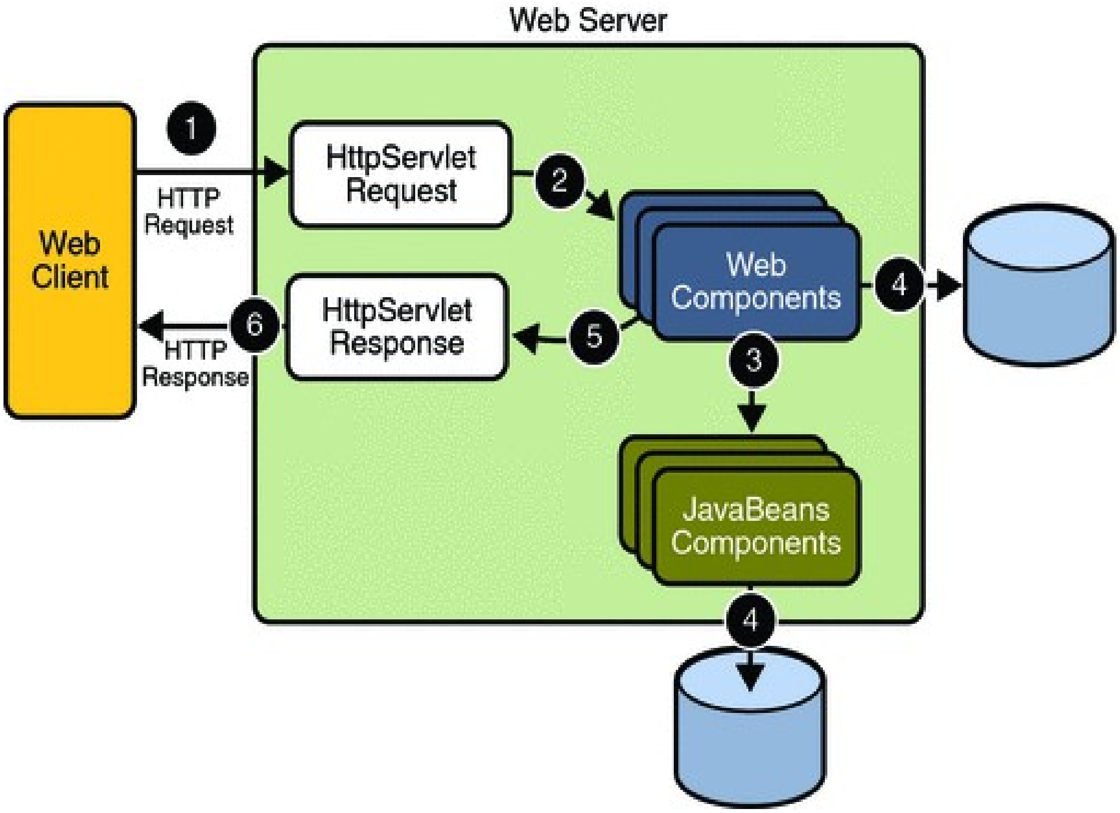


Fig 3.4.1 Architecture of Proposed System

4. IMPLEMENTATION

INTRODUCTION:

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

4.1 TECHNOLOGIES USED

Hardware/Software Used-

The implementation of a modern Blood Donation/Management Website represents a significant stride in leveraging technology to enhance the blood donation process and optimize the management of this critical healthcare resource. This section of the report outlines the practical implementation of the solution, highlighting the modern tools and technologies employed to create an efficient, user-friendly, and reliable platform.

Technological Stack:

Front-End Development:

HTML5, CSS3, JavaScript: These core technologies are used for creating the website's structure, design, and interactive elements. They ensure a visually appealing and responsive user interface.

Back-End Development:
PHP : You need a server to host your PHP application and MySQL database. The server can be physical or cloud-based. The hardware specifications will depend on the expected traffic and resource requirements of your application.

Database Management:
MySQL: MySQL is a relational database management system and is commonly used for PHP applications. You can install and configure MySQL on your server. You may use a MySQL client application to manage the database, such as phpMyAdmin or MySQL Workbench. These tools provide a graphical interface to interact with the database.

Mapping and Geolocation:
Google Maps API: Integration of the Google Maps API allows users to locate

nearby blood donation centres and events accurately. It enhances the accessibility of donation opportunities.

User Experience (UX) Enhancements:

Responsive Design: The website is designed to provide an optimal user experience across various devices, including desktops, tablets, and smartphones.

User-friendly Interface: Intuitive navigation, clear call-to-action buttons, and minimalistic design principles contribute to a user-friendly interface.

Personalized Profiles: Registered users have personalized profiles that showcase their donation history, impact, and upcoming donation appointments.

Notification System: Users receive notifications about upcoming donation drives, appointment reminders, and urgent blood requirements, enhancing engagement and participation.

- **Analysis –**

Code structure analysis*:* Examine the HTML, CSS, and PHP files of the e-voting website to identify how the code is structured. This includes looking at how the pages are organized, how the styling is implemented, and how the PHP scripts interact with the database. Use an HTML validator to ensure that the website's code follows the proper syntax and structure. Check the CSS files to ensure that the styling is consistent across all pages and that the website is responsive. Review the PHP scripts to verify that the code is well-organized and easy to understand.

Functionality analysis: Test the functionality of the Blood Donation System website to ensure that all features are working as expected. This includes testing the login and registration systems, verifying that users. Test the login and registration systems to ensure that users can create accounts and log in securely.

Security analysis: Check the website for potential security vulnerabilities. Make sure that the website is using secure encryption protocols and that sensitive data, such as user passwords, is stored securely. Use a web vulnerability scanner to identify any potential security vulnerabilities. Ensure that the website is using secure encryption protocols, such as HTTPS, to protect user data. Check that the website is using a secure method of storing user passwords, such as hashing and salting.

Usability analysis: Evaluate the usability of the website by testing it with users to ensure that it is easy to navigate and use. Look for ways to improve the user experience, such as adding clear instructions or simplifying complex processes. Conduct user testing to ensure that the website is easy to navigate and use. Look for areas where users may get confused or lost, and provide clear instructions or visual cues to guide them.

Optimize the website's design to make it visually appealing and easy to read.

Performance analysis: Check the website's performance, including load times and response times. Identify any bottlenecks or performance issues that may impact the user experience. Use a performance testing tool to identify any bottlenecks or performance issues that may affect the website's speed or responsiveness. Optimize the website's code and images to improve load times. Ensure that the website is compatible with all major browsers and devices.

Session Tracking:

HTTP is a stateless protocol, it provides no way for a server to recognize that a sequence of requests is all from the same client. This causes a problem for application such as shopping cart applications. Even in chat application server can’t know exactly who’s making a request of several clients.

The solution for this is for client to introduce itself as it makes each request, Each clients needs to provide a unique identifier that lets the server identify it, or it needs to give some information that the server can use to properly handle the request, There are several ways to send this introductory information with each request Such as:

USER AUTHORIZATION:

One way to perform session tracking is to leverage the information that comes with User authorization. When a web server restricts access to some of its resources to only those clients that log in using a recognized username and password. After the client logs in, the username is available to a servlet through getRemoteUser().

When use the username to track the session. Once a user has logged in, the browser remembers her username and resends the name and password as the user views new pages on the site. A servlet can identify the user through her username and they’re by Track her session.

The biggest advantage of using user authorization to perform session tracking is that it’s easy to implement. Simply tell the protect a set of pages, and use getRemoteUser() to identify each client.Another advantage is that the technique works even when the user accesses your site form or exists her browser before coming back.

The biggest disadvantage of user authrization is that it requires each user to register for an account and then log in in each time the starts visiting your site. Most users will tolerate registering and lagging in as a necessary evil when they are accessing sensitive information, but its all overkill for simple session tracking.Other problem with user authorization is that a user cannot simultaneously maintain more than one session at the same site.

Hidden Form Fields:

One way to support anonymous session tracking is to use hidden from fields. As the name implies, these are fields added to an HTML, form that are not displayed in the client's browser, They are sent back to the server when the form that contains them is submitted.

In a sense, hidden form fields define constant variables for a form. To a servlet receiving a submitted form, there is no difference between a hidden fields and a visible filed.

As more and more information is associated with a clients session . It can become burdensome to pass it all using hidden form fields. In these situations it's possible to pass on just a unique session ID that identifies as particular clients session.

That session ID can be associated with complete information about its session that is stored on the server.

The advantage of hidden form fields is their ubiquity and support for anonymity. Hidden fields are supported in all the popular browsers, they demand on special server requirements, and they can be used with clients that haven't registered or logged in.

The major disadvantage with this technique, however is that works only for a sequence of dynamically generated forms, The technique breaks down immediately with static documents, emailed documents book marked documents and browser shutdowns.

URL Rewriting:

URL rewriting is another way to support anonymous session tracking, With URL rewriting every local URL the user might click on is dynamically modified. or rewritten, to include extra, information. The extra information can be in the deform of extra path information, added parameters, or some custom, server-specific.URL change. Due to the limited space available in rewriting a URL, the extra information is usually limited to a unique session.Each rewriting technique has its own advantage and disadvantage

Using extra path information works on all servers, and it works as a target for forms that use both the Get and Post methods. It does not work well if the servlet has to use the extra path information as true path information.The advantages and disadvantages of URL.rewriting closely match those of hidden form fileds,The major difference is that

URL rewriting works for all dynamically created documents.

Persistent Cookies:

A fourth technique to perform session tracking involves persistent cookies. A cookie is a bit of information. sent by a web server to a browser that can later be read back form that browser. When a browser receives a cookie, it saves the cookie and there after sends the cookie back to the server each time it accesses a page on that server, subject to certain rules. Because a cookie's value can uniquely identify a client, cookies are often used for session tracking.

Persistent cookies offer an elegant, efficient easy way to implement session tracking. Cookies provide as automatic an introduction for each request as we could hope for. For each request, a cookie can automatically provide a client's session ID or perhaps a list of clients performance. The ability to customize cookies gives them extra power and versatility.

The biggest problem with cookies is that browsers don't always accept cookies sometimes this is because the browser doesn't support cookies. More often its because

The browser doesn't support cookies. More often its because the user has specifically configured the browser to refuse cookies.

The power of serves:

The power of servlets is nothing but the advantages of servlets over other approaches, which include portability, power, efficiency, endurance, safety elegance,integration,extensibility and flexibility.

5. TESTING

Introduction

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionalities of components, sub assemblies, and/or a finished product it is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

5.1 Types of Testing

5.1.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing we have is white box oriented and some modules the steps are conducted in parallel.

5.1.2. Integration Testing

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system will operate successfully in all its aspects conditions. Thus the system testing is a confirmation that all is correct and an opportunity to show the user that the system works.The purpose of integration testing is to verify functional, performance and reliability requirements placed on major design items. These "design items", i.e. assemblages (or groups of units), are exercised through their interfaces using black box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-process communication is tested and individual subsystems are exercised through their input interface.

Test cases are constructed to test that all components within assemblages interact correctly, for example across procedure calls or process activations, and this is done after testing individual modules, i.e. unit testing.

5.1.5 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

This type of testing ensures that

- All independent paths have been exercised at least once
- All logical decisions have been exercised on their true and false sides
- All loops are executed at their boundaries and within their operational bounds
- All internal data structures have been exercised to assure their validity.

To follow the concept of white box testing we have tested each form .we have created independently to verify that Data flow is correct, All conditions are exercised to check their validity, All loops are executed on their boundaries.

Basic Path Testing

Established technique of flow graph with Cyclometer complexity was used to derive test cases for all the functions. The main steps in deriving test cases were:

Use the design of the code and draw correspondent flow graph.

Conditional Testing

In this part of the testing each of the conditions were tested to both true and false aspects. And all the resulting paths were tested. So that each path that may be generate on particular condition is traced to uncover any possible errors.

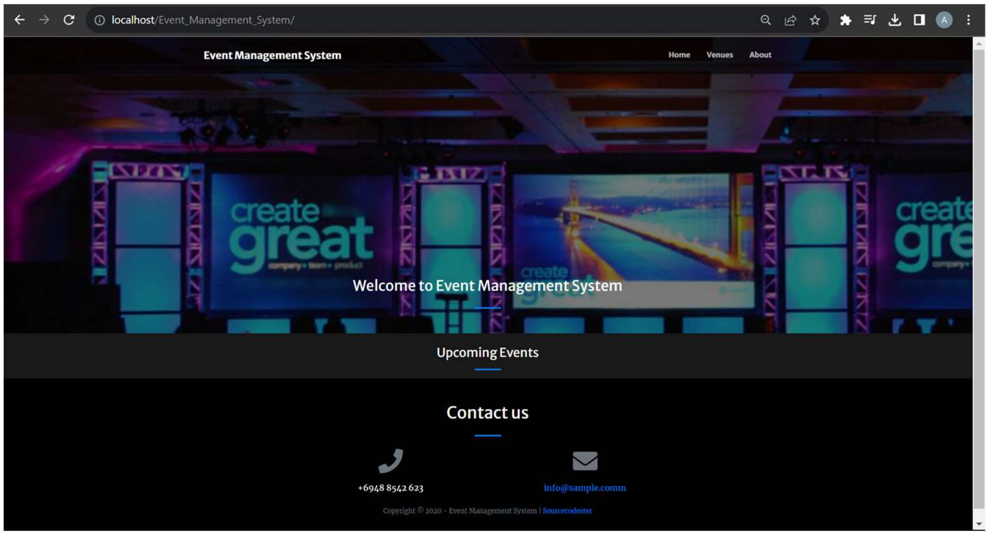
Data Flow Testing

This type of testing selects the path of the program according to the location of definition and use of variables. This kind of testing was used only when some local variable were declared. The definition-use chain method was used in this type of testing. These were particularly useful in nested statements.

Loop Testing

In this type of testing all the loops are tested to all the limits possible. The following exercise was adopted for all loops:All the loops were tested at their limits, just above them and just below them.All the loops were skipped at least once.For nested loops test the inner most loop first and then work outwards.For concatenated loops the values of dependent loops were set with the help of connected loop.Unstructured loops were resolved into nested loops or concatenated loops and tested as above.Each unit has been separately tested by the development team itself and all the input have been validated.

6. OUTPUT SCREENSHOTS



Welcome to Event Management System

Upcoming Events

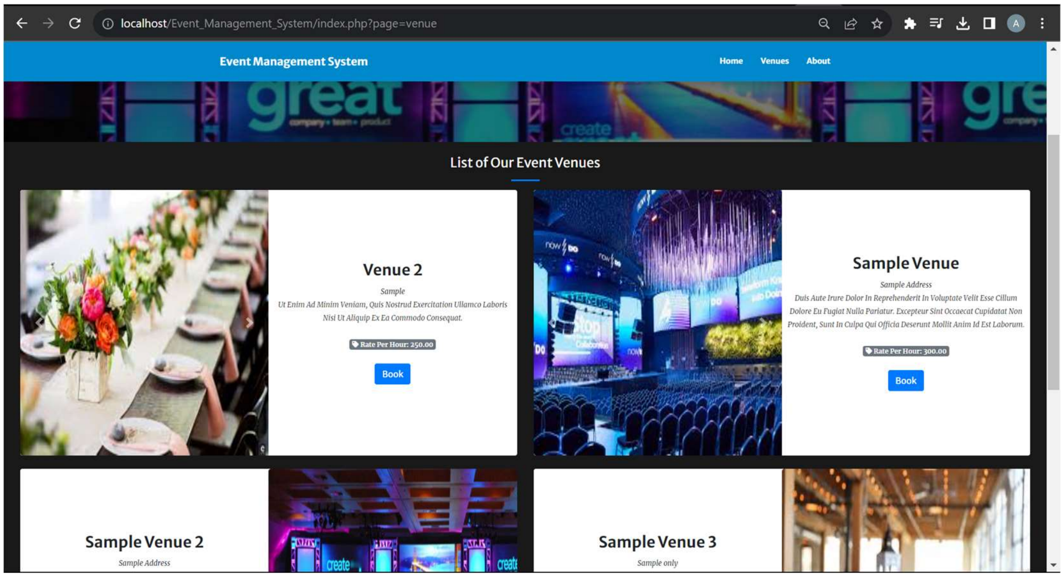
Contact us

+6948 8542 623

info@sample.comm

Copyright © 2020 - Event Management System | Sourcecodester

Fig: 6.1 Home page of event management



List of Our Event Venues

Venue 2

Sample

Ut Enim Ad Minim Veniam, Quis Nostrud Exercitation Ullamco Laboris Nisi Ut Aliquip Ex Ea Commodo Consequat.

Rate Per Hour: 250.00

Sample Venue

Sample Address

Duis Aute Irure Dolor In Reprehenderit In Voluptate Velit Esse Cillum Dolore Eu Fugiat Nulla Pariatur. Excepteur Sint Occaecat Cupidatat Non Proident, Sunt In Culpa Qui Officia Deserunt Mollit Anim Id Est Laborum.

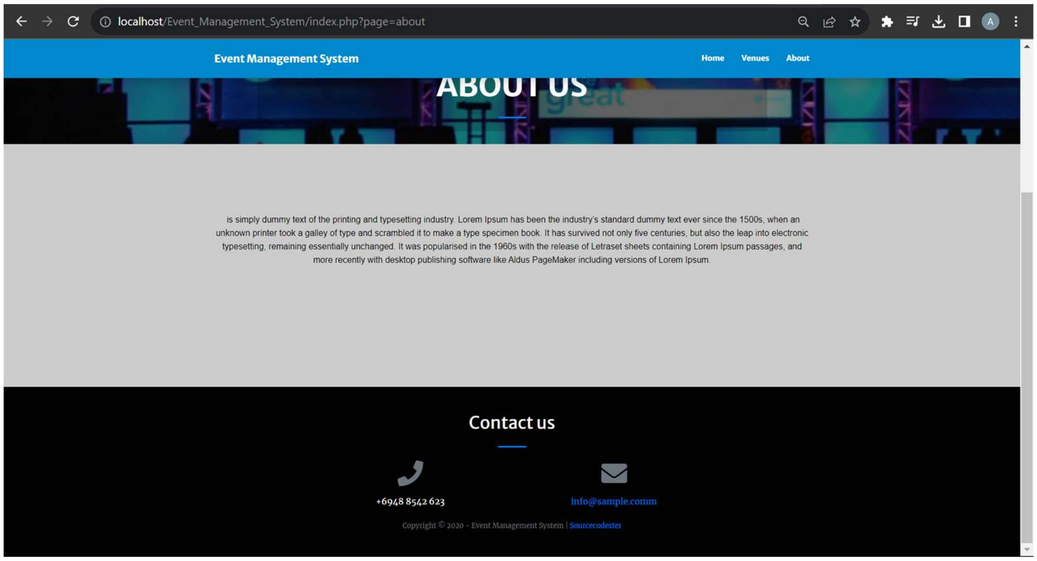
Rate Per Hour: 300.00

Sample Venue 2

Sample Address

Sample Venue 3

Sample only



is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Contact us

+6948 8542 623

info@sample.comm

Copyright © 2020 - Event Management System | Sourcecodester

The above screen is the main screen for this project. It contains the details about the event management system. In this page it contains Home, Admin and Update.

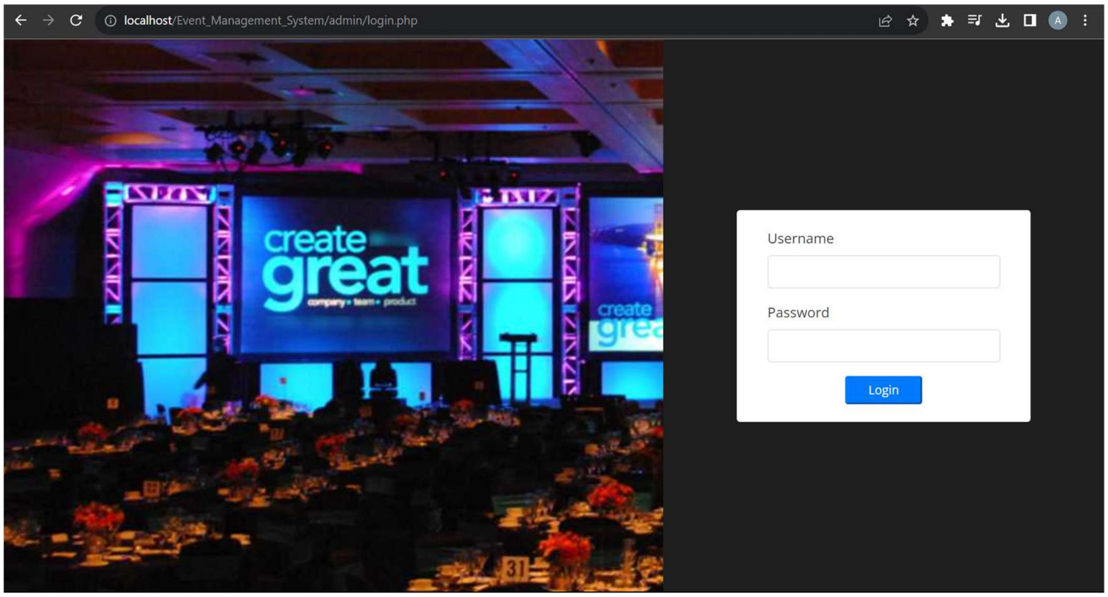
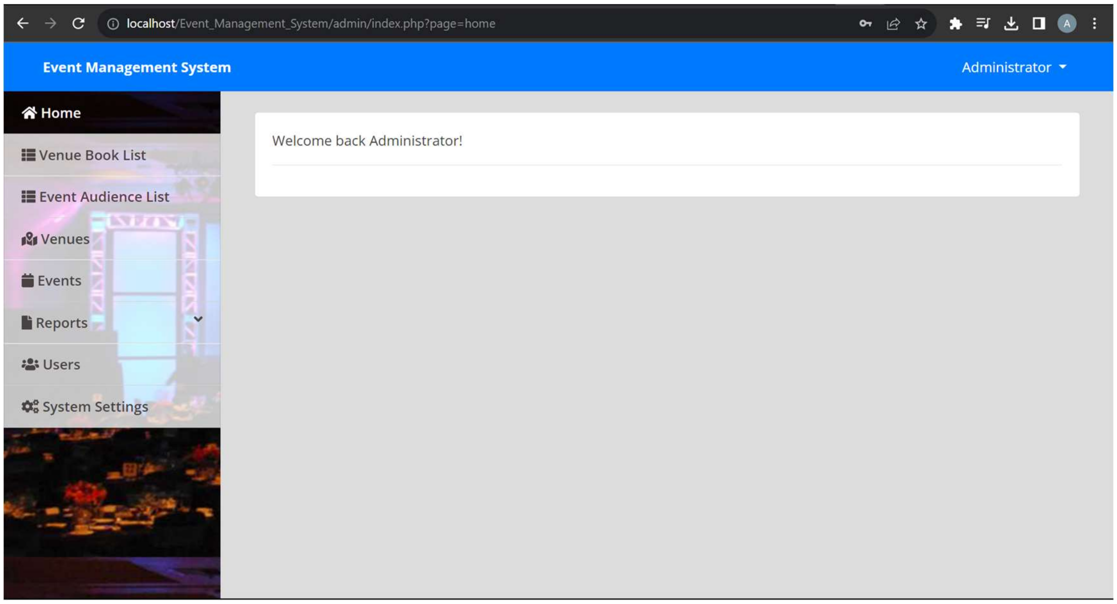


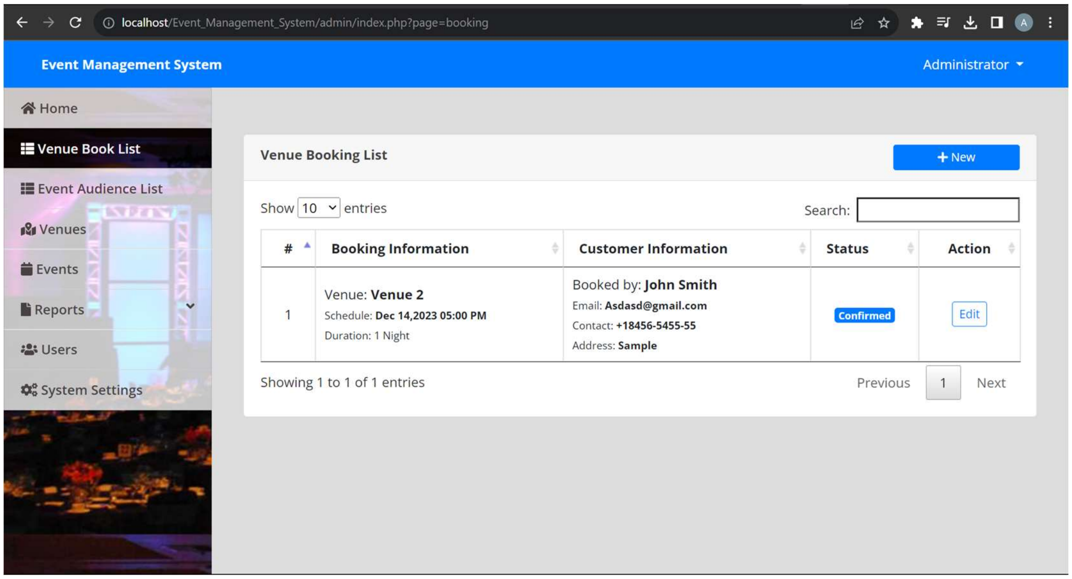
Fig: 6.2 Login page of event management

The above screen shows the details of Login page. Here user can view details and view the events.



Welcome back Administrator!

Fig: 6.3 dashboard of event management



Venue Booking List

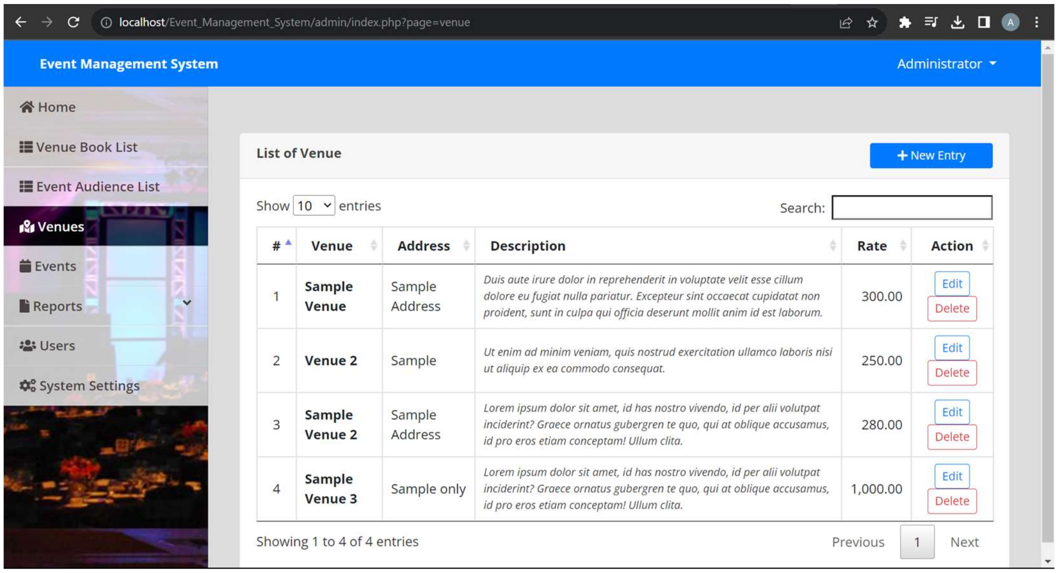
Show 10 entries

Search:

#	Booking Information	Customer Information	Status	Action
1	Venue: **Venue 2** Schedule: **Dec 14,2023 05:00 PM** Duration: 1 Night	Booked by: **John Smith** Email: **Asdasd@gmail.com** Contact: **+18456-5455-55** Address: **Sample**	Confirmed	Edit

Showing 1 to 1 of 1 entries

Fig: 6.4 venue booking of event management



localhost/Event_Management_System/admin/index.php?page=venue

List of Venue

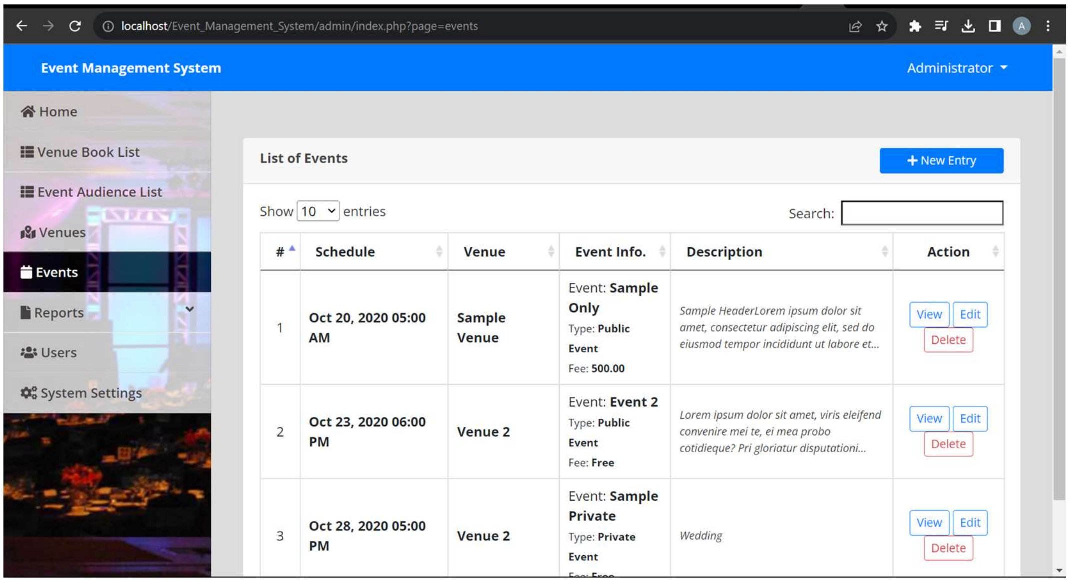
Show 10 entries

Search:

#	Venue	Address	Description	Rate	Action
1	**Sample Venue**	Sample Address	*Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.*	300.00	Edit Delete
2	**Venue 2**	Sample	*Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.*	250.00	Edit Delete
3	**Sample Venue 2**	Sample Address	*Lorem ipsum dolor sit amet, id has nostro vivendo, id per alii volutpat inciderint? Graece ornatus gubergren te quo, qui at oblique accusamus, id pro eros etiam conceptam! Ullum clita.*	280.00	Edit Delete
4	**Sample Venue 3**	Sample only	*Lorem ipsum dolor sit amet, id has nostro vivendo, id per alii volutpat inciderint? Graece ornatus gubergren te quo, qui at oblique accusamus, id pro eros etiam conceptam! Ullum clita.*	1,000.00	Edit Delete

Showing 1 to 4 of 4 entries

Fig: 6.5 lists of venue of event management



localhost/Event_Management_System/admin/index.php?page=events

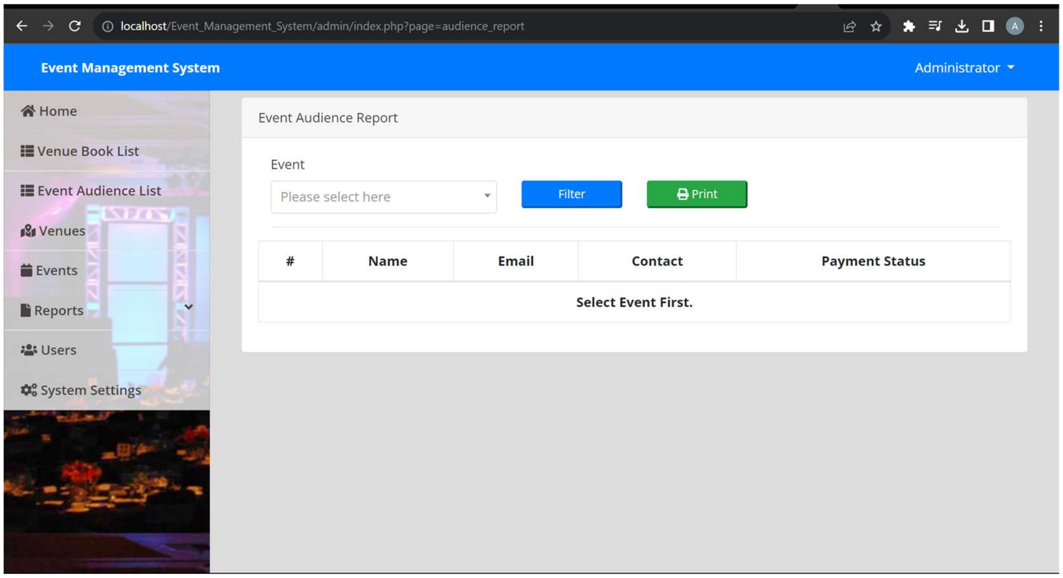
List of Events

Show 10 entries

Search:

#	Schedule	Venue	Event Info.	Description	Action
1	**Oct 20, 2020 05:00 AM**	**Sample Venue**	Event: **Sample Only** Type: **Public Event** Fee: **500.00**	*Sample HeaderLorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et...*	View Edit Delete
2	**Oct 23, 2020 06:00 PM**	**Venue 2**	Event: **Event 2** Type: **Public Event** Fee: **Free**	*Lorem ipsum dolor sit amet, viris eleifend convenire mei te, ei mea probo cotidieque? Pri gloriatur disputationi...*	View Edit Delete
3	**Oct 28, 2020 05:00 PM**	**Venue 2**	Event: **Sample Private** Type: **Private Event** Fee: **Free**	*Wedding*	View Edit Delete

Fig: 6.6 List of events of event management



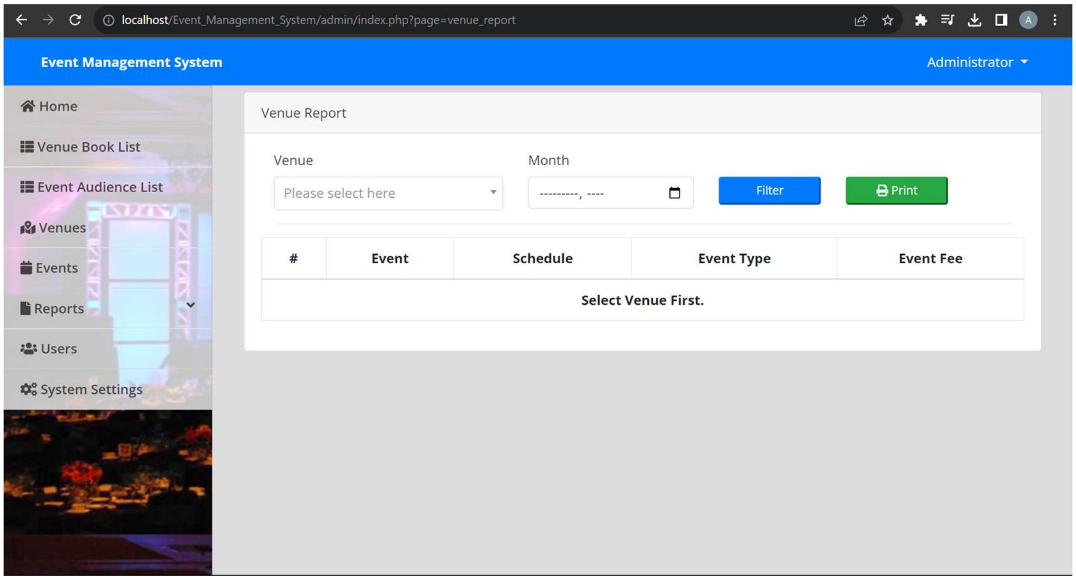
Event Audience Report

Event

Please select here

#	Name	Email	Contact	Payment Status
Select Event First.				

Fig: 6.7 event audience report of event management



Venue Report

Venue

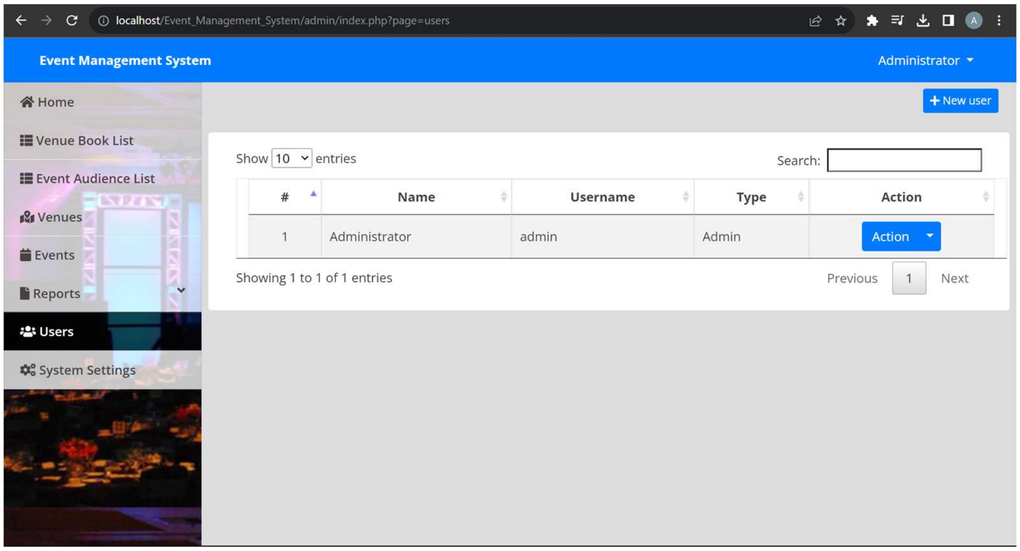
Please select here

Month

---------, ----

#	Event	Schedule	Event Type	Event Fee
Select Venue First.				

Fig: 6.8 venue report of event management



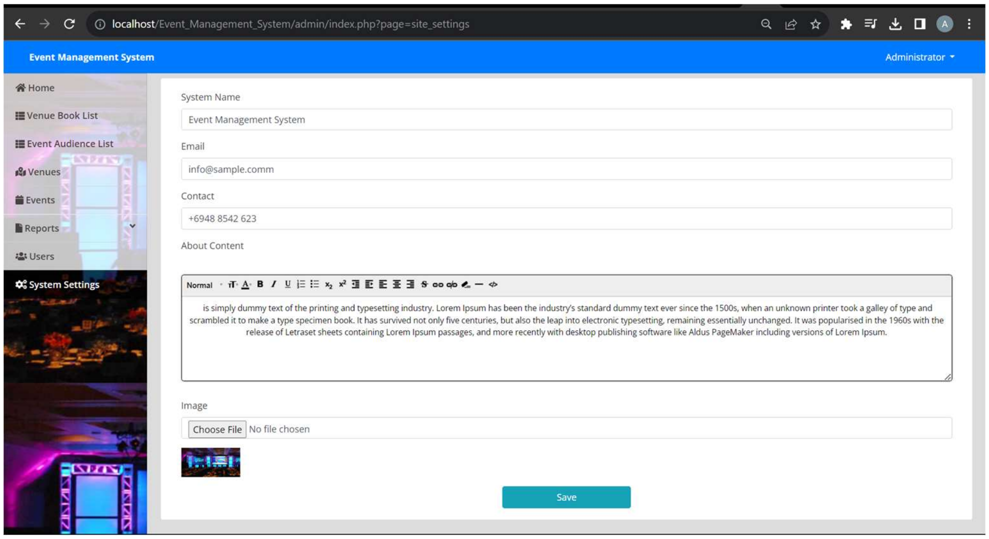
Show 10 entries

Search:

#	Name	Username	Type	Action
1	Administrator	admin	Admin	Action

Showing 1 to 1 of 1 entries

Fig: 6.9 user control of event management



System Name: Event Management System

Email: info@sample.comm

Contact: +6948 8542 623

About Content:

is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Image: No file chosen

Fig: 7.0 System settings of event management

7. CONCLUSION

The **“Event Management”** was successfully designed and is tested for accuracy and quality.During this project we have accomplished all the objectives and this project meets the needs of the organization. The developed will be used in searching, retrieving and generating information for the concerned requests.

GOALS ACHIVIED

- ✓ Reduced entry work
- ✓ Easy retrieval of information
- ✓ Reduced errors due to human intervention
- ✓ User friendly screens to enter the data
- ✓ Portable and flexible for further enhancement
- ✓ Web enabled.
- ✓ Fast finding of information request

REFERENCES

(1) Wikipedia, URL: http://www.wikipedia.org.

(2) Answers.com, Online Dictionary, Encyclopedia and much more,
URL: http://www.answers.com

(3) Google, URL: http://www.google.co.in

(4) Project Management URL: http://www.startwright.com/project.htm