Event Management System

for

Video ADS Company (Pvt) Ltd.

W.D.I.C. Wevita

Event Management System

for

Video ADS Company (Pvt) Ltd.

W.D.I.C. Wevita

BIT Registration Number: R100498

Index Number: 1004982

Supervisor: Mr. Isuru Vincent

2017

This dissertation is submitted in partial fulfillment of the requirement of the Degree of Bachelor of Information Technology (external)

of the University of Colombo School of Computing

Declaration

"I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for degree or diploma in any university and to the best of my knowledge and brief, It does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for interlibrary loans and for the title and abstract to be made available to outside organization."

Signature of candidate: .... Date: 2017/11 /06

Name of the candidate: W. D. I. C. Wevita

Countersigned by:

Signature of supervisor: .... Date: 2017/11 /06

Name of supervisor: V.P. IsuruPraboth Vincent

Abstract

Video ADS Company (Pvt) Ltd is a growing and well known company in the industry of Video Production. The existing system that the company currently uses to manage their company's operations is defective, inefficient, time consuming and a manual system. It was realized that an electronic information system could have be used instead of current manual system to reduce defects and increase the efficiency. As a solution, an Event Management System was implemented in the client environment.

The implemented system has provided many facilities such as storing customer details, employee details, event details, providing good overview of available video packages, assigning employees for events effectively, reminding the management about the upcoming events and reminding the customer about the due payments etc.

The Event Management System has been developed based on Rational Unified Process (RUP) framework. PHP is an Object Oriented language and has been selected as the programming language to implement this system. Further MySQL server database has been selected as the database to store the data of the system.

The requirements for the new system were gathered by number of interviews with the top management and the end users. Also the documentation used in the current manual system was examined in order to collect all the requirements. Many types of UML diagrams including use case and ER were used to illustrate the proposed information system in order to define the functional and non functional requirements, flow of data among uses of the system and define the database structure etc. Database has been designed with the suitable relationships among the tables in order to minimize the data redundancy. Also the user interfaces have been designed to make the proposed system more user friendly. Further this system has provided automation of the processes and has become an aid to face the competition of the industry.

Acknowledgement

I wish to express my sincere gratitude to the academic staff who are assigned to BIT degree in University of Colombo School of Computing, especially for the project coordinator for giving us continues guidance to make this project success.

I sincerely thank my project supervisor, Mr.IsuruProboth for his guidance and encouragement in carrying out this project work.

My heartiest gratitude should go to my client, Mr. Anslem Dilip, Proprietor of Video ADS Company (Pvt) Ltd and his staff for the great support provided by dedicating their valuable time for my project.

Table of Content

List of Figures

List of Tables

List of Acronyms

ADSL	**- Asymmetric Digital Subscriber Line**
AJAX	**- Asynchronous JavaScript and XML**
CD ROM	**- Compact Disc Read Only Memory**
CSS	**- Cascading Style Sheet**
DBMS	**- Database Management Systems**
GB	**- Gigabyte**
GHz	**- Gigahertz**
HTML	**- Hypertext Mark- up Language**
JSON	**- JavaScript Object Notation**
MB	**- Megabyte**
MD5	**- Message Digest Algorithm 5**
NIC	**- National Identity Card**
OOAD	**- Object Orient Analysis and Design**
PDF	**- Portable Document Format**
PHP	**- Hypertext Pre- processor**
RAM	**- Random Access Memory**
RUP	**- Rational Unified Process**
SQL	**- Structured Query Language**
OOMS	**- Online Operations Management System**
UML	**- Unified Modeling Language**
URL	**- Uniform Resource Locator**
WWW	**- World Wide Web**
XAMPP	**- Windows/Linux Apache MySQL PHP Perl**

Chapter 01 – INTRODUCTION

Video Production is capturing of moving images (video), reducing parts of captured video and combining video parts together (video editing) to produce the required video output. Development of Video Production industry is highly combined with Information Technology.

The manual system used by the Video ADS Company has become a barrier to step forward beyond their competitors in the industry of Video Production. This new computer based information system will break the barriers to their success.

1.1 Motivation for the Project

Lots of manual methods are been used by the Client Company to accomplish their company work currently. Also the officers and the management of the company used to keep many important details of events in mind instead of having proper documentation. This has caused many problems and it has affected to the quality of their service.

The following problems have been occurred in the existing manual system.

1. There is no proper mechanism to manage the event details, employee details and customer details.
2. There is no proper mechanism to assign employees for an event.
3. There is no proper system to select the suitable video solutions for the customer requirements.
4. The manual methods currently used to remind the customer about the due payments are inefficient.
5. Problems occurred because of no proper preparation for events.

Event Management System for the Video ADS Company was developed to overcome the above mentioned problems of their current manual system.

1.2 Scope of the Proposed Project

Scope of the proposed Event Management system for Video ADS Company is as follows.

- System has facilitated to store Event details, Customer details, Payment details, Employee details, Video package details and Event team details.
- System has provided information to the management to decide where it is possible to accept a event considering the other bookings.
- System has provided a clear picture of video packages for the customer to select the suitable video solution for their video requirement.
- System has facilitated to assign employees for an event effectively.
- System has provided a reminding mechanism to management about upcoming events.

1.3 Objectives of the System

The main objective of the system is to minimize the problems occurred in organizing process of a video coverage event. The proposed system will increases the efficiency and productivity of the company. Following are the objectives of the system.

- Provide a clear idea about each video package to select the suitable one for customer requirements.
- Increase the efficiency of managing event details, employee details and customer details.
- Increase the effectiveness of assigning the employees for an event.
- Increase the trust of customers by starting early preparation for events.

1.4 Dissertation Overview

The dissertation provides a clear picture of the Event Management System for Video ADS Company. The structure of dissertation is as follows.

Chapter 02-Analysis

This chapter explains the requirement gathering techniques, details of the manual system, functional non-functional requirements and details of the existing systems.

Chapter 03-Design

Use case diagram of the proposed system, database design of the system and main interfaces of the system are included in this chapter.

Chapter 04-Implementation

This chapter explains the hardware software requirements, development tools which is used for system.

Chapter 05-Evaluation

The techniques of testing, details of software testing, high level test plan of the system and client evaluation of the system are described in this chapter.

Chapter 06-Conclusion

This chapter explains the future enhancements of the system and lesson learnt of the overall project work.

Appendixes

These are provided further details about the content of the dissertation chapters which were not included in the chapters.

Chapter 02 – ANALYSIS

System Analysis is the process of studying procedures in order to identify their goals for finding out an efficient way to achieve them. This study provides a clear picture of the system and finalizes the scope of the project.

2.1 Fact Gathering Techniques

Gathering requirements of the users was difficult because the end users were incapable of expressing their requirement for the new system. Various requirement gathering techniques were carried out in order to clarify their requirements.

- **Interviews**
  The proposed system will not be able to satisfy the stakeholders without knowing their expectations and goals. The information taken by the interviews are used to understand the perspectives of every stakeholder. This technique was used to identify the end user requirements by interviewing the users several times whenever a conflict has been occurred. Interviews were carried out with the users who are willing to use the proposed system and the other stakeholders such as expecting reports from the system.

- **Observation**
  Goals of using the existing system were studied by observing users, flow of communication and documentation. Documents which has been used by the manual system were very useful to resolve conflicts.

2.2 Analyzing the Current Manual System

2.2.1 Business Procedure

Main business process of the client’s company is reserving dates for video coverage of the events for customer requests, assign a video team for each event and provide their service at the requested date according to customer requirements.

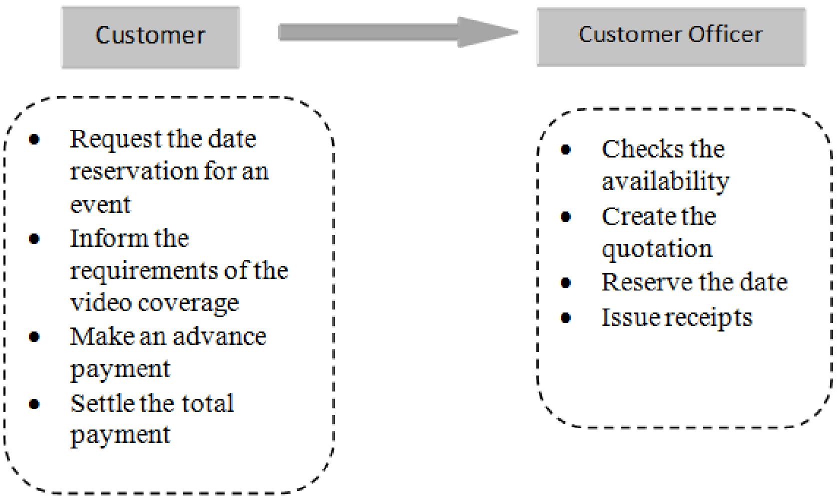


Figure 2.1 - Business Procedure

When a customer asks for a reservation, customer officer checks for the availability of the date before accepting the event by checking their log book. If the date is available, quotation is created according to the customer requirements including the total payment. If the customer agreed with the quotation, an advance payment should be made in order to reserve the date. After making the advanced payment the reservation will be added to the log book. The total payment should be settled before the event date. If a customer requests to cancel a reservation, customer officer de-allocate the date from the log book. Positive communication flow of this process is illustrated in figure 2.1 above.

The customer officer checks the log book on daily basis for reserved dates for coming week. Customers are informed about the due payments one week in advance for the event date by the customer officer. The Manager also informed about the events of the coming week for early preparation. The Manager assigns a video team for the each event on coming week and the employees are informed about the event. If the customer settled the full payment before the event date, the service will be provided at the event. Communication flow of this is presented in figure 2.2 above.

All the organization information and reports regarding the reservation activities quotations and income status are recorded and granted manually with paper documents and log books.

2.2.1 Problems Exist in the Manual System

The client organization had no proper system to handle reservation activities. Due to the lack of proper reservation tracking system, the customer office and the manager had a huge stress in reminding reservations. Hence, the lack of a management tool had created extra work for the customer office and the manager which slows down their efficient performance as mentioned below.

- Checking the availability of the requested date is hard and time consuming.
- Manual calculation of the payments is unreliable.
- Event details, Customer details and Employee details are not stored in a proper manner for future use.
- The process of reminding the customer about the due payments and the process of reminding the manager for early preparation for the event at the correct time are unreliable.
- The process of assigning the team members for an event is defective.

Due to the lack of standard and centralized system to handle all the reservation details, the business process had become hard and time wasting. Further there is no proper report generating process about the events.

2.3 Current Manual System

Figure 2.3 shows the use case diagram of the current manual system. This system consists of several modules. They are listed with brief descriptions.

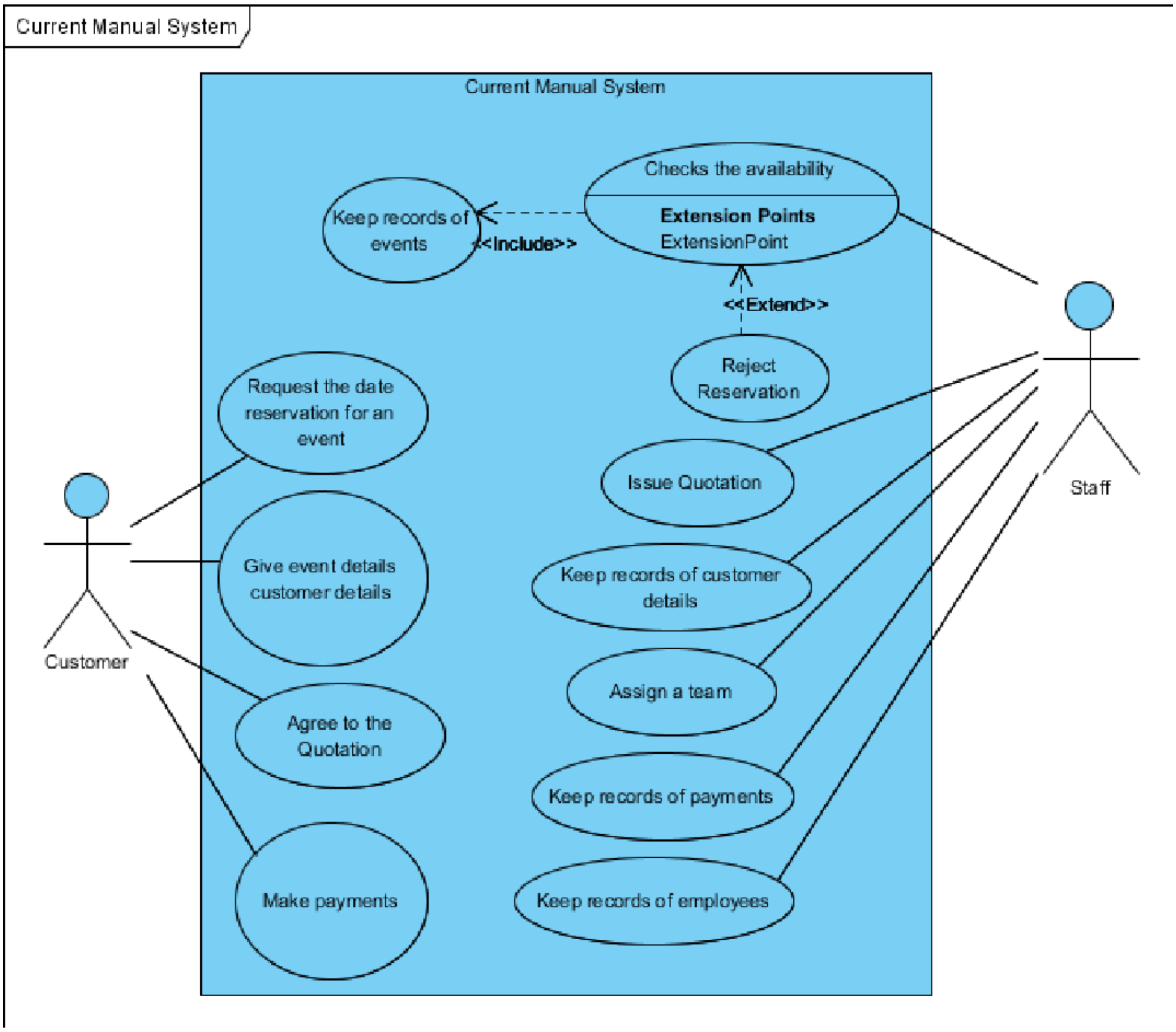


Figure 2.2-Use case diagram of the current manual system

2.3.1 Customer Registration

Mainly they use a manual way to store details about customers who are dealing with the company frequently, such as customer name and contact details.

2.3.2 Event Registration

When customers request for an event date reservation and the date is available event details are stored manually. Since the current system is a manual system, they have to waste lot of time to store these details. More time consuming when finding history details of events. Mainly they keep records of customer contact details, event details and payment details of one event at one place. System has provided facilities to manage these event details effectively and has removed the heavy paper based work.

2.3.3 Quotation Issuing

Quotations are issued according to the customer requirements for the event. But quotation details are not stored in the current manual system. If the customer agreed to the quotation event details are stored manually.

2.3.4 Manage Payment Details

Payment details management is one of the critical parts of this company. Customer payments for events are calculated and stored manually. Proposed system has given a solution to store these details and manage them efficiently.

2.3.4 Team Allocation

Assigning team of employees for an event is done by the manager after the event date is reserved for an event. But there is no proper process for team allocation in the current manual system. New system provides an automated process for team allocation, which is more accurate and efficient.

2.3.4 Manage Employee Details

Mainly they store personal details about employees such as full name, NIC, address and contact details. Currently they use manual way to store these details.

2.4 Functional Requirements

The basic functionalities that are expected from the new system are functional requirements. Following are the functional requirements that have been identified through the system analysis stage.

2.4.1 User Management Module

- Users can log into the system securely by validating username and password.
- System has provided facilities to create, edit, view, delete new user.
- System has provided facilities to assign user privileges based on the user level.
- System has provided facilities to change password function.
- User can be logout from the system.

2.4.2 Customer Management Module

- User can be able to insert, delete, update, view and search customer details.
- User can be able to generate reports of all information.

2.4.3 Event Details Management Module

- User can be able to check availability before reserve dates for events.

- When customers give their event details, system allows the user to insert event details such as Customer name, Event Name, Event date, Starting time, End time and Number of cameras. Further can select package type based on customer requirements.
- If the event is from a regular customer user can be able to select the customer name without entering the customer details again in to the system.
- System allows the user to cancel the reserved event date based on the customer consent.
- System has facilitated to update the event details
- User can be able to search event details considering the reserved date and event name.

2.4.4 Quotation Management Module

- User can be able to create quotation for an event.
- System generates the camera charges based on the number of cameras and selected video package.
- User can be able to insert transport charges, other expenses and discount to create the quotation.
- User can be able to delete, update, view and search quotation details.

2.4.5 Payment Details Management Module

- User can be able to insert, delete, update, view and search payment details.
- Payment can be made under the issued quotation.

2.4.6 Package Details Management Module

- User can be able to insert, delete, update, view the package details.
- Package details are available to the system user in order to help the customers to select the suitable video package for their requirements.

2.4.7 Employee Details Management Module

- User can be able to insert, delete, update, view and search employee details.
- User can be able to generate reports of all information.

2.4.8 Team Management Module

- System has provided facilities to automatically assign employees for events.
- User is capable of inserting, deleting, updating and viewing team details.

2.4.9 Reminding Massages Management Module

- System has provided facilities to send reminding emails to the customer to remind about due payments.
- System has provided facilities to send reminding emails to the manager to remind about the upcoming events.

2.4.9 Notification Module

- System has provided facility to remind, before one week for an event.
- These notifications are disabled when required task is completed.

2.4.10 Report Generation Module

- System has facilitated to generate pay slips for each and every driver.
- System has facilitated to generate monthly reports for fuel expenses, maintenance, expenses, other company expenses,
- System has facilitated to generate monthly income reports.
- System has facilitated to generate report for monthly reservation reports etc.
- System can generate payment slips when customers do payments.
- System can generate quotations according to the customer requirements.

2.5 Non Functional Requirements

Non Functional requirements are very important fact in a software system. Non-functional requirements describe the constraints of the system. These constraints are applied to the whole system, not only for the individual part of the system. Following are the Non-functional requirements of the system.

2.5.1 Security

Security is very important feature as the system contains information such as payment details, customer details and employee details. Misplacing or misusing of such information will be a huge problem to the future of the company. Information entered to the system will be secured because of the user management features.

2.5.2 Usability

Usability requirement has been achieved by using various techniques such as easy menu navigation, searching options, attractive interfaces and use of matching colors.

2.5.3 Accuracy

Accuracy is achieved by several techniques throughout the system. By applying validation techniques to the user input data and applying accurate calculations at the required points have lead to the accurate output of the system.

2.6 Existing Similar Systems

Following are identified as existing similar systems and they were compared with each other to identify positive and negative facts.

- **Tradify**

  It is a browser based job tracking and managing software which capable of generating quotation and invoice. It runs on both Windows and Mac platforms and on mobile devices. Main interface shown in figure 2.4

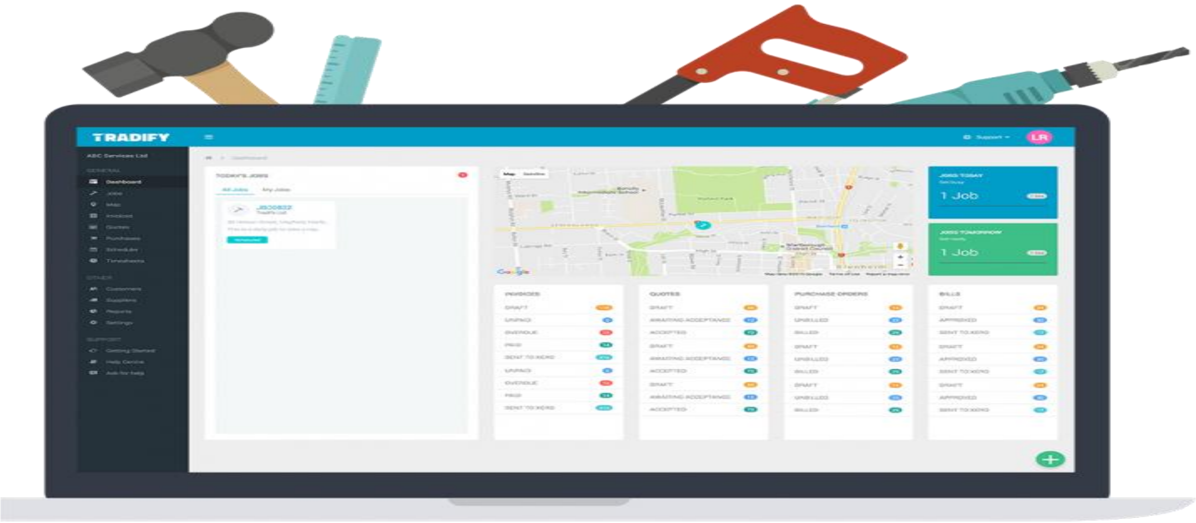


Figure 2.3- Main interface of Tradify

- **Zoho Projects**

Zoho Projects has incredible Gantt chart options, allowing users to set complicated tasks and milestones. It also offers timesheets and detailed reporting features (and for those who don’t want to make their own reports, it has 50 pre-made templates to choose). Main interface shown in figure 2.5

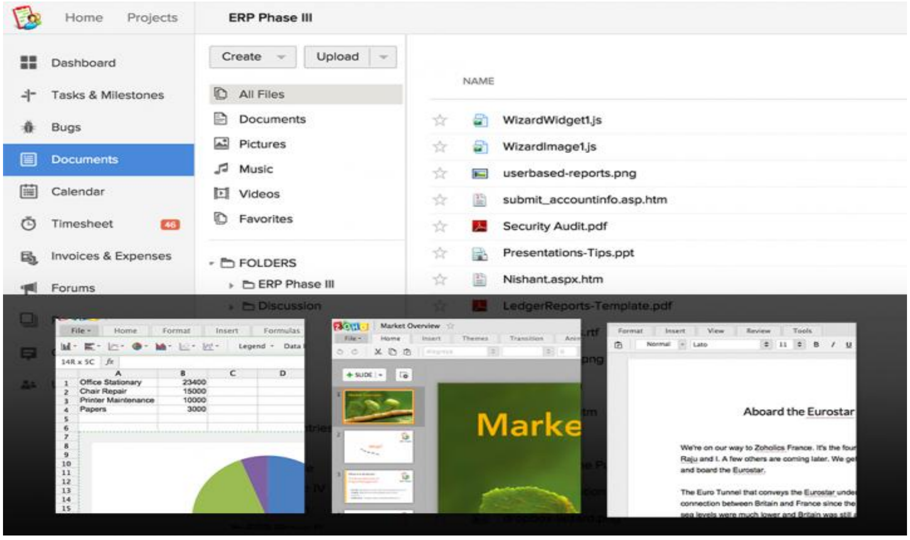


Figure 2.4- Main interface of ***Zoho Projects***

There is lots of Event Management Software which are created by various organizations in worldwide for their own business process, and then enhanced it as readymade products. Some of them are high cost and also consist of additional features which are not suit to the client's organizational needs. Maintain those additional features are overhead to the company and expand the learning curve of the system users.

This Event Management System developed for client's company is a low cost simple custom software solution for this client organization providing all the necessary facilities in the domain of event management.

Chapter 03 – DESIGN

Software design is an interactive process through which requirements are translated into a blueprint for constructing the software. Initially blueprint depicts the holistic view of software. [1]

The quality of the entire system will depend on this stage. Selecting a methodological approach is very important task for effective development of the system.   System designers use various system design approaches such as modern structured design, information engineering, prototyping, JAD, RAD and object-oriented design. Object oriented approach considers every part of the system as a Object has its own properties, behaviors and relationships. This approach has been selected to develop the proposed system.

3.1 System Development Life Cycle

The systems development life cycle, or SDLC, is a planning tool used by developers to plan, build and maintain high-quality products. Steps in the SDLC move teams through planning, development, construction and deployment of new software or platforms. [2]

There are various SDLC methodologies/models such as waterfall, Agile, incremental etc.

3.2 Methodology for the Proposed System

Rational Unified Model(RUP) is used as a software process model to develop Event Management System for Video ADS Company. RUP is a software development methodology which is very much suit for the systems with vague requirements.

This methodology was chosen for developing the proposed system as it had some unclear and unstable user requirements. Main phases of RUP are briefly described below.

- Inception Phase

Identify the initial scope of the project, a potential architecture for the system and obtain user acceptance.

- Elaboration Phase

In this phase create the project plan and construct an architectural baseline that implements a working application with limited functionality.

- Construction Phase

Build working software on a regular, incremental basis which meets the highest priority needs of the client.

- Transition Phase

Validate and deploy the system into the production environment.

3.3 Object Oriented Designing

The purpose of Object-Oriented (OO) design is to define the classes (and their relationships) that are needed to build a system that meets the requirements. This technique was employed with UML visual models. This approach increases the modeling capability and with the use of basic concepts of Object Oriented Design.

3.3.1 High Level Use Case Diagram for the Event Management System

Figure 3.1 shows the high level use case diagram of the system

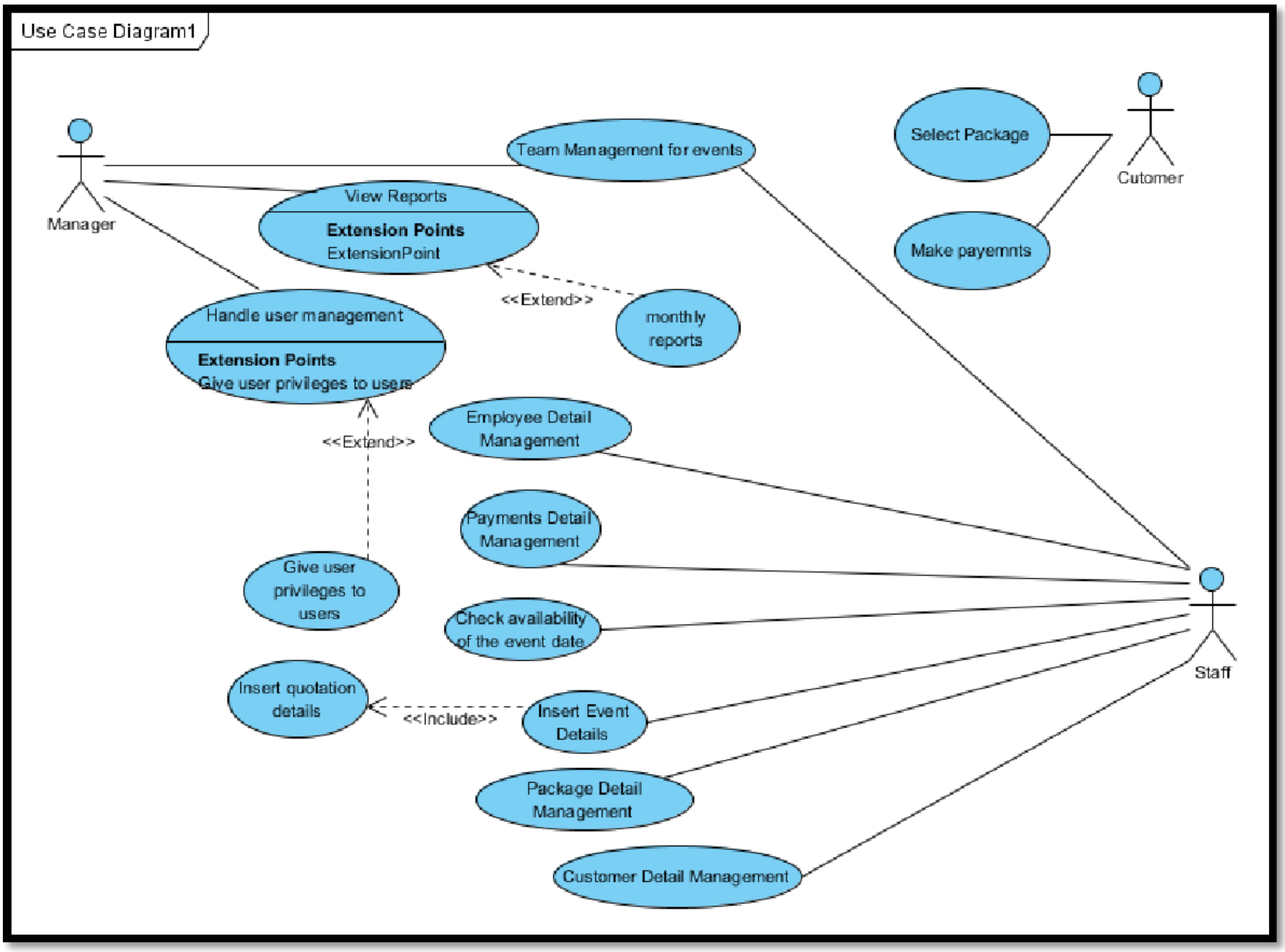


Figure 3.1 -High level use case diagram of the system

3.3.2 Class Diagram for the Event Management System

Figure 3.2 shows the class diagram of the system

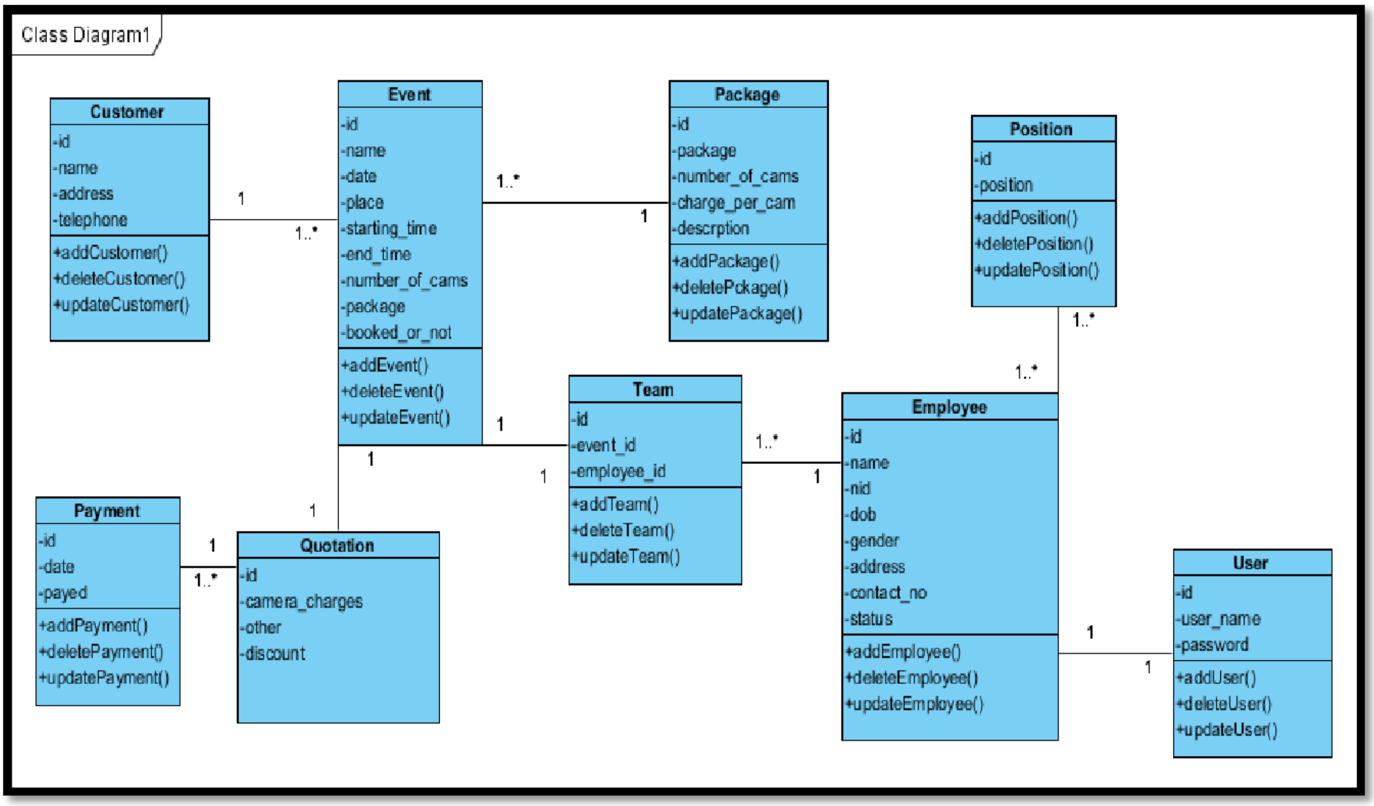


Figure 3.2 -Class diagram of the system

3.4 Database Design

Database design is the process of producing a detailed data model of a database [3].

Figure 3.3 represent the table structure of the database for proposed system is given below.

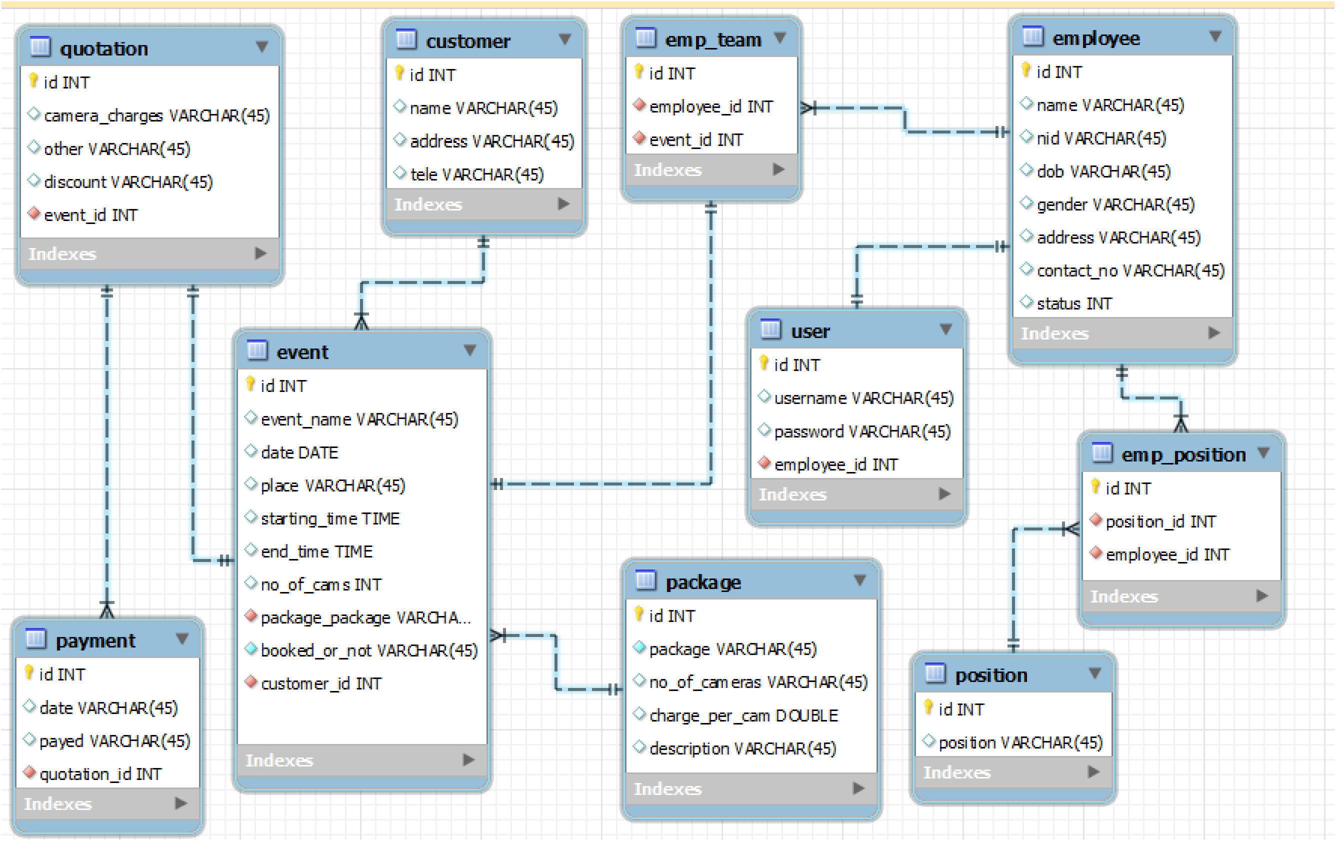


Figure 3.3-Table structure of the database for proposed system

3.4.1 Database Normalization

Database normalization is the process of organizing the fields and tables of a relational database to minimize redundancy and dependency. Normalization usually involves dividing large tables into smaller (and less redundant) tables and defining relationships between them [WWW4]

- **First Normal Form (1NF)**

In the first normal form the repeating groups and multi valued columns are removed and arranged them into one table, and the primary key is defined for identifying each related attribute.

- **Second Normal Form (2NF)**

In the first normal form partial dependencies are removed and separate tables are created. Then relationships among the tables are created with a foreign key.

- **Third Normal Form (3NF)**

In the first normal form transitive dependencies are removed.

3.5 Interface Design

Interfaces exist to enable interaction between humans and our world. They can help clarify, illuminate, enable, show relationships, bring us together, pull us apart, manage our expectations, and give us access to services. The act of designing interfaces is not Art. Interfaces are not monuments unto themselves. Interfaces do a job and their effectiveness can be measured. They are not just utilitarian, however. The best interfaces can inspire, evoke, mystify, and intensify our relationship with the world. [5]

System is covered by a good user friendly interface in order to interact with the user. Following are the actions taken to ensure the user friendliness of the interfaces of the entire system.

- Eye friendly colours were used for the interface.(e.g. Navy blue and sky blue were used)

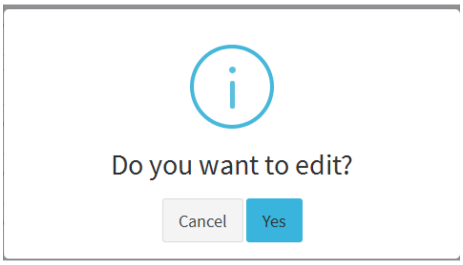
- Tab indexes were used for easily navigation.

- Enter button of keyboard has used for the enter details instead of mouse click in order to increase the effectiveness of the system.

- Meaningful names and texts were used for button, forms and labels for increase the effectiveness of the system.

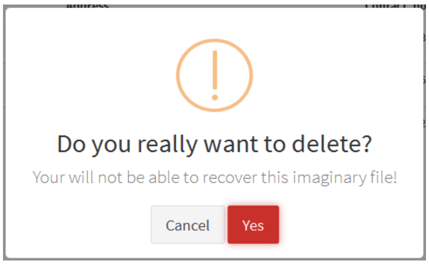
- After saving, updating and deleting record, every form should show the successful message. Figure 3.4 and Figure 3.5 shows the message used to inform inserting and deleting of record.

- Before saving, updating, deleting record, system should take permission from the user. Following figure 3.4 and figure 3.5 Shows the confirmation message which is displaying before editing and deleting record.



Do you want to edit?

Figure 3.4 confirmation message before editing



Do you really want to delete?

Your will not be able to recover this imaginary file!

Figure 3.5 confirmation message before deleting

3.5.1 User Login Interface

The common interface for all users to log into the system is as follows. Authorized users can log into the system and error messages are popup if the invalid user log into the system. Following figure 3.6 shows login interface of the system

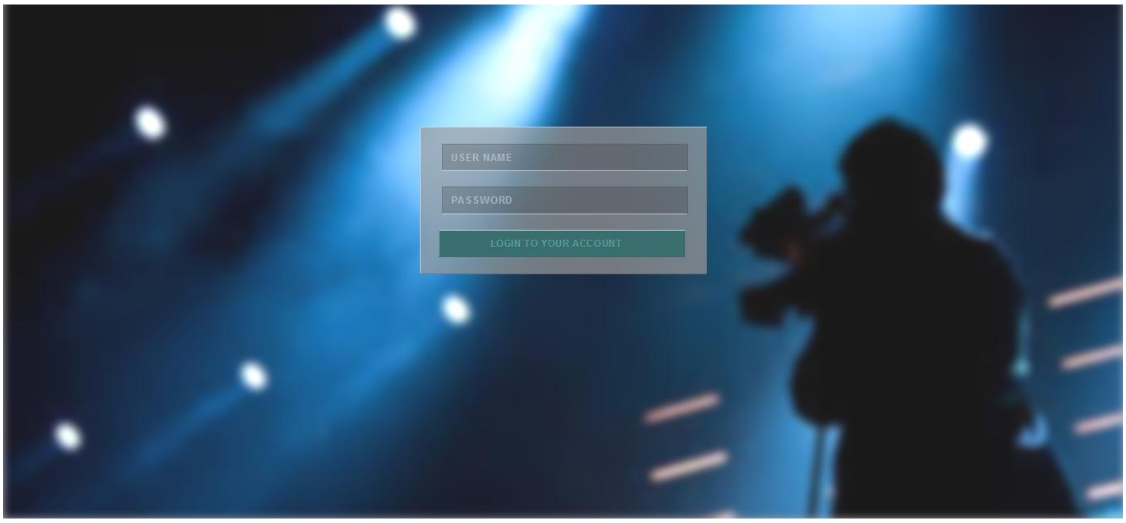
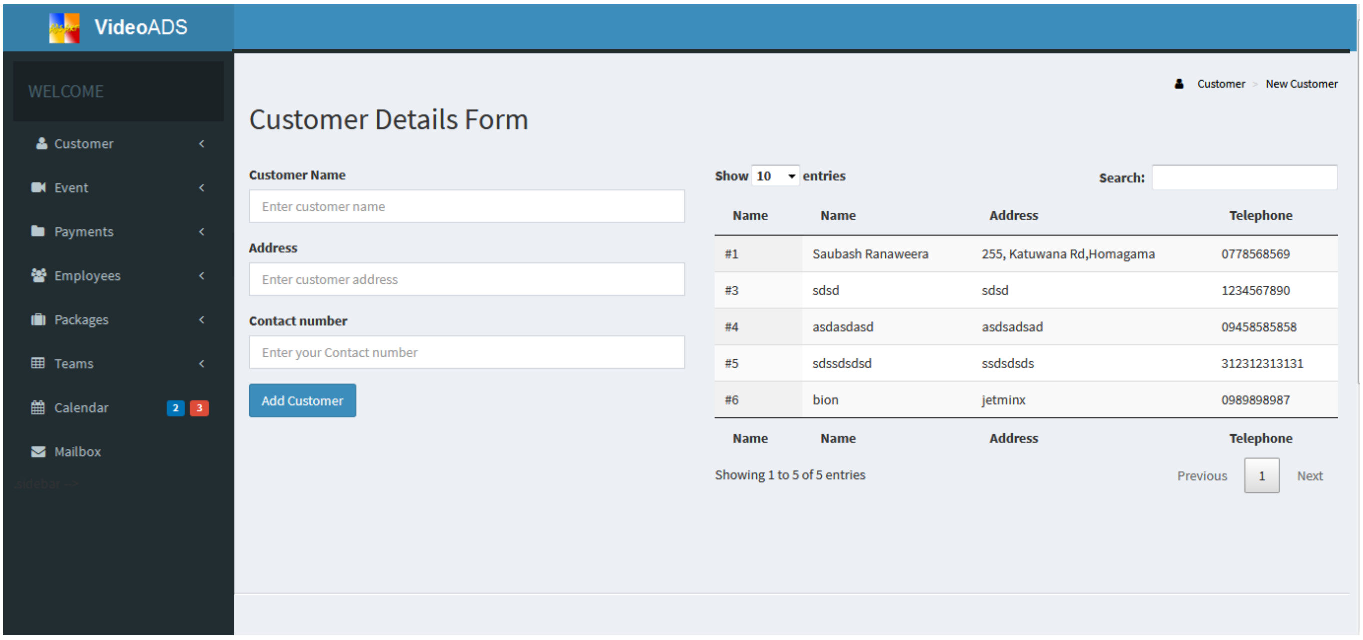


Figure 3.6 - User Login Interface

3.5.2 Customer Registration Interface

Figure 3.7 represents the customer registration form. New customers are registered in the system before inserting the event details. Registering the same customer for multiple events is not required.



Customer Details Form

Customer Name: Enter customer name

Address: Enter customer address

Contact number: Enter your Contact number

Show 10 entries

Search:

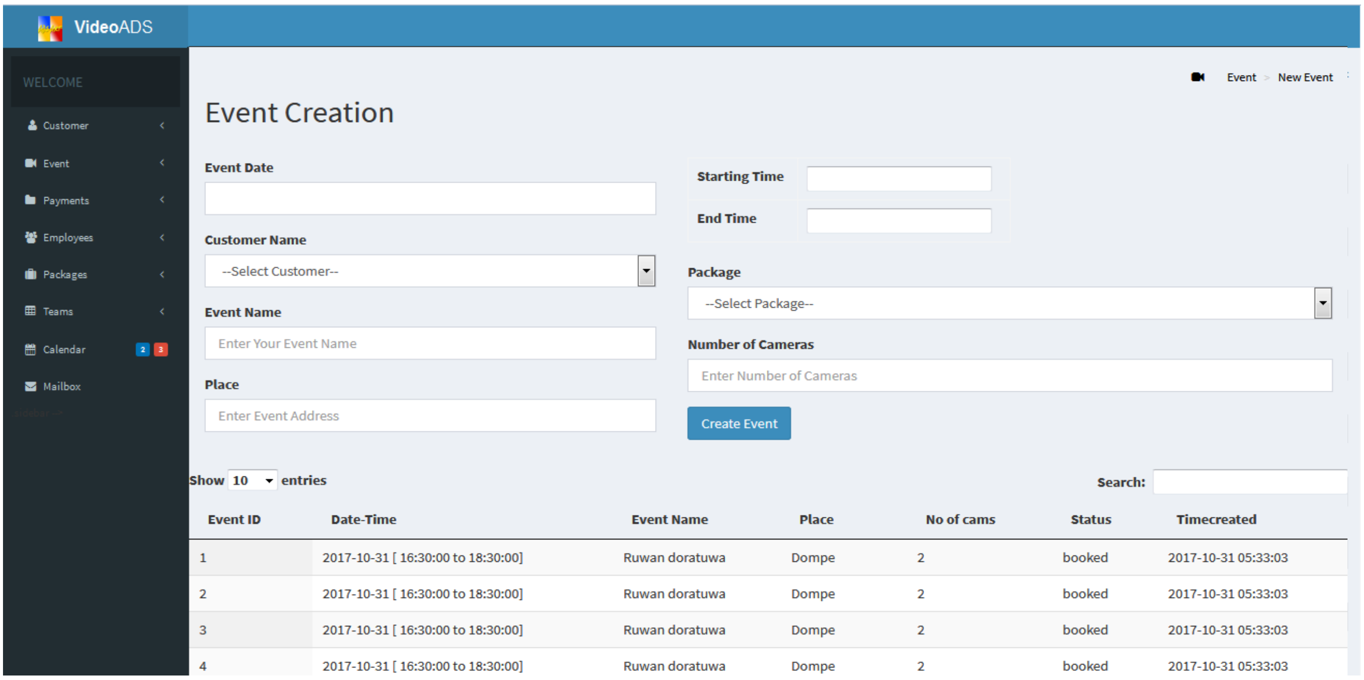
Name	Name	Address	Telephone
#1	Saubash Ranaweera	255, Katuwana Rd,Homagama	0778568569
#3	sdsd	sdsd	1234567890
#4	asdasdasd	asdsadsad	09458585858
#5	sdssdsdsd	ssdsdsds	312312313131
#6	bion	jetminx	0989898987
Name	Name	Address	Telephone

Showing 1 to 5 of 5 entries

Figure 3.7 Customer registration form

3.5.3 Event Date Reservation Form

Figure 3.8 represent the event date reservation form. Customer name should be selected from the drop down list. This feature allow user to enter event details of regular customers, without going to the customer registration again. Package which is selected by the customer should be selected from another drop down list. From the button on the bottom of the form allow user to go to another form ‘Create Quotation’. Both passing the event data to the database and loading the Create Quotation Form are done by the same button. This feature ensures the quotations for all events that are inserted to the system.



Event Creation

Event Date

Starting Time

End Time

Customer Name

--Select Customer--

Package

--Select Package--

Event Name

Enter Your Event Name

Number of Cameras

Enter Number of Cameras

Place

Enter Event Address

Show 10 entries

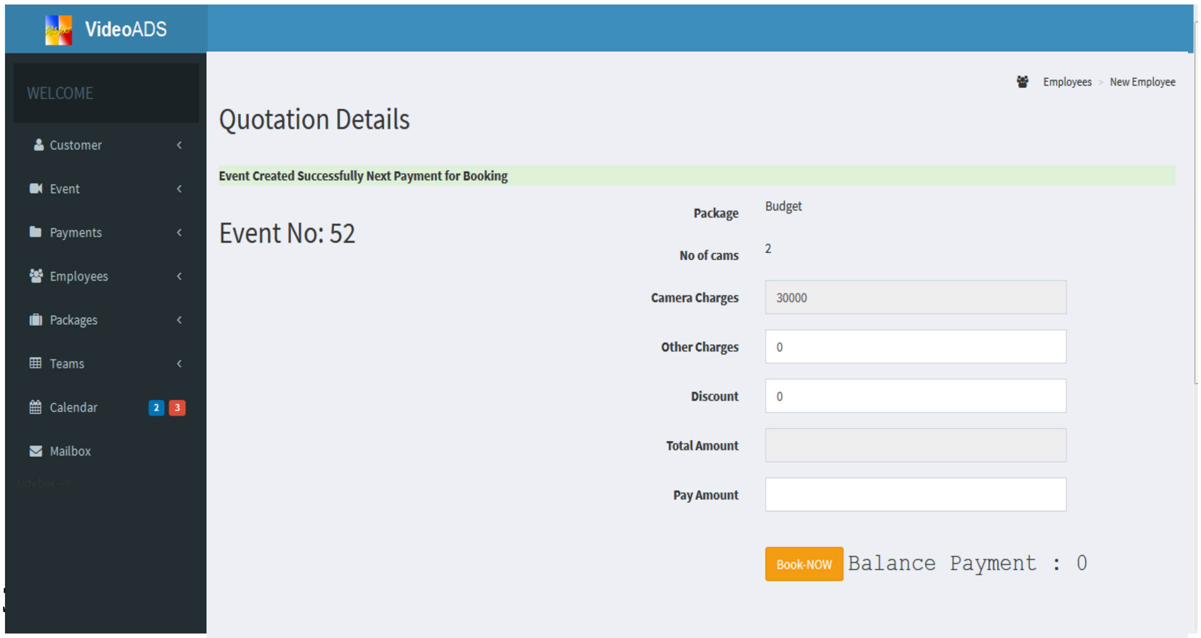
Search:

Event ID	Date-Time	Event Name	Place	No of cams	Status	Timecreated
1	2017-10-31 [16:30:00 to 18:30:00]	Ruwan doratuwa	Dompe	2	booked	2017-10-31 05:33:03
2	2017-10-31 [16:30:00 to 18:30:00]	Ruwan doratuwa	Dompe	2	booked	2017-10-31 05:33:03
3	2017-10-31 [16:30:00 to 18:30:00]	Ruwan doratuwa	Dompe	2	booked	2017-10-31 05:33:03
4	2017-10-31 [16:30:00 to 18:30:00]	Ruwan doratuwa	Dompe	2	booked	2017-10-31 05:33:03

Figure 3.8 –Event Creation form

3.5.4 Create Quotation Form

Figure 3.9 represents the create quotation form which allow the user to enter the essential data for generating the quotation.



Quotation Details

Event Created Successfully Next Payment for Booking

Event No: 52

Package	Budget
No of cams	2
Camera Charges	30000
Other Charges	0
Discount	0
Total Amount	
Pay Amount	

Balance Payment : 0

Figure 3.9-Create quotation form

Payment Insertion Form

Figure 3.10 represents the payment insertion form which allows the user to insert a new payment regarding to a previously issued quotation. Receipt for the payment could be printed by clicking on the ‘Print Receipt’ button.

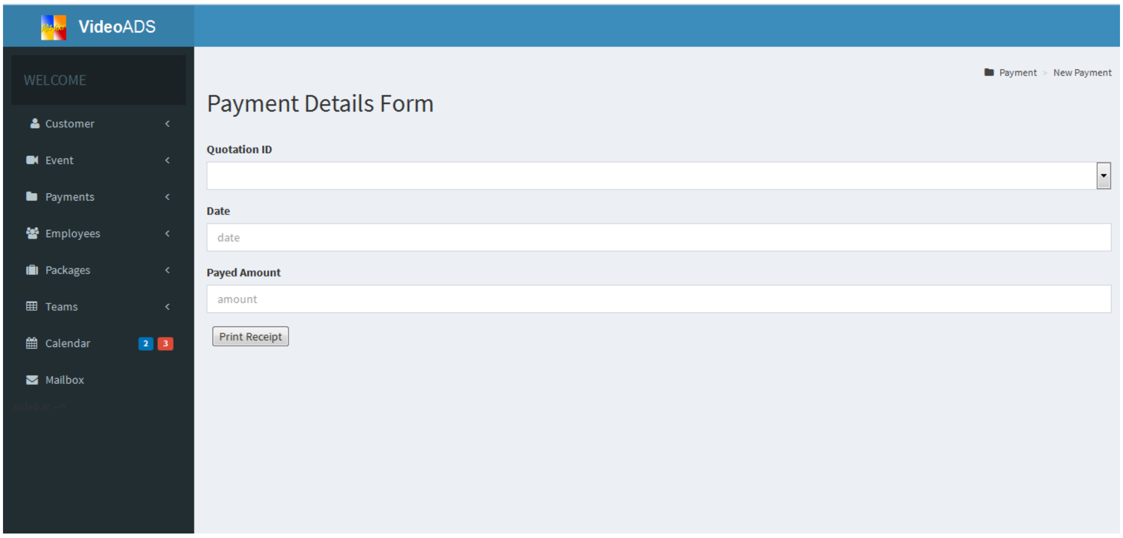


Figure 3.10 -Payment insertion form

3.5.6 Package Insertion Form

Figure 3.11 represents the package insertion form which allows the user to insert a new video package into the system.

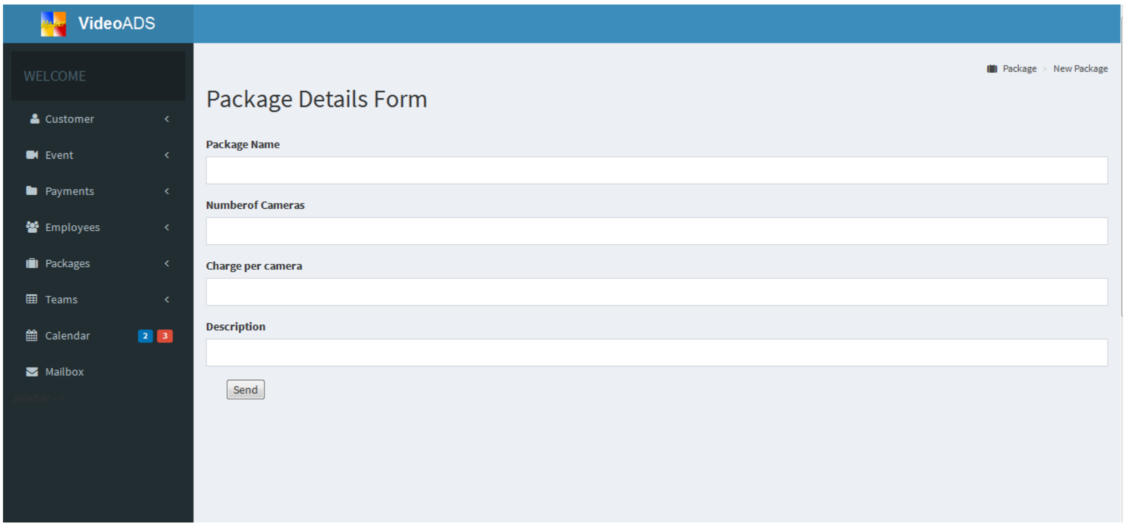
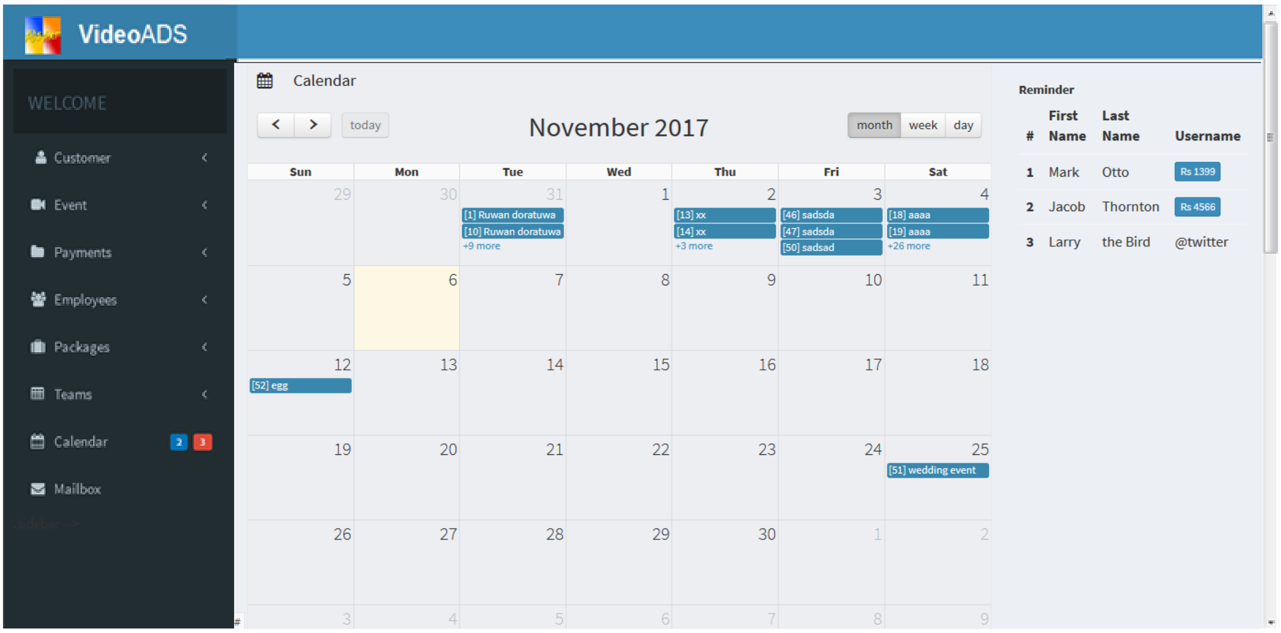


Figure 3.11 -Package insertion form

3.5.7 Calendar

Figure 3.12 represents the Calendar. It facilitates the user to check the availability of the requested date. If the date is available user can accept the new event by entering the event details through ‘Event Details Form’. Events which are already booked by paying an advance payment, represented in red colour. Events which are not booked but issued the quotations, represented in blue colour.



November 2017

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	31 [1] Ruwan doratuwa [10] Ruwan doratuwa +9 more	1	2 [13] xx [14] xx +3 more	3 [46] sadsda [47] sadsda [50] sadsad	4 [18] aaaa [19] aaaa +26 more
5	6	7	8	9	10	11
12 [52] egg	13	14	15	16	17	18
19	20	21	22	23	24	25 [51] wedding event
26	27	28	29	30	1	2
3	4	5	6	7	8	9

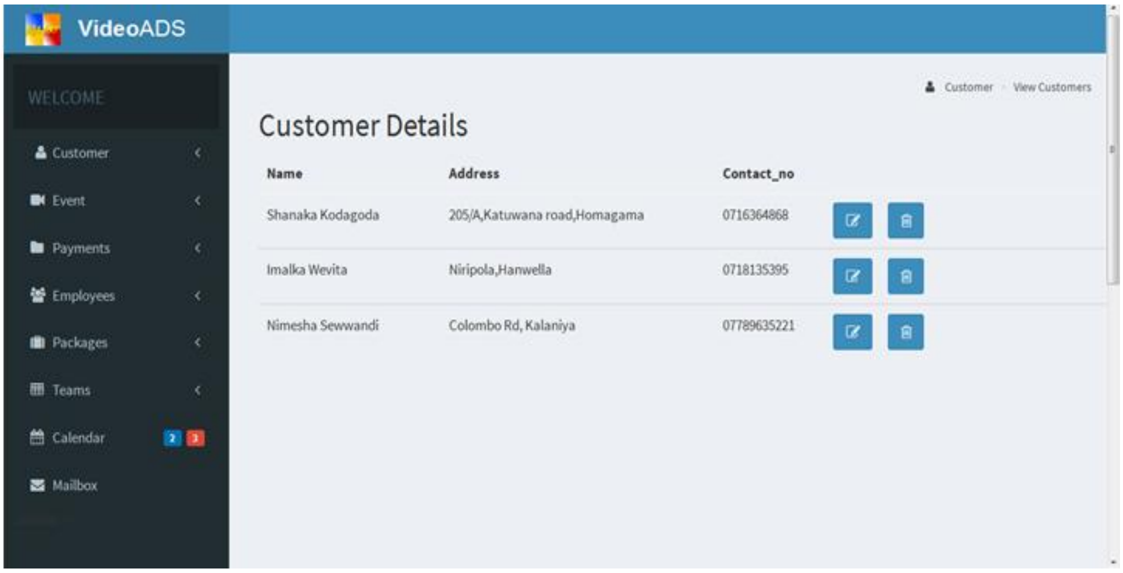
Reminder

#	First Name	Last Name	Username
1	Mark	Otto	Rs 1399
2	Jacob	Thornton	Rs 4566
3	Larry	the Bird	@twitter

Figure 3.12-Calendar

3.5.8 Customer Details

Figure 3.13 represents the Customer Details View which shows the details of all registered customers. This view facilitates the user to edit or delete customer recodes.



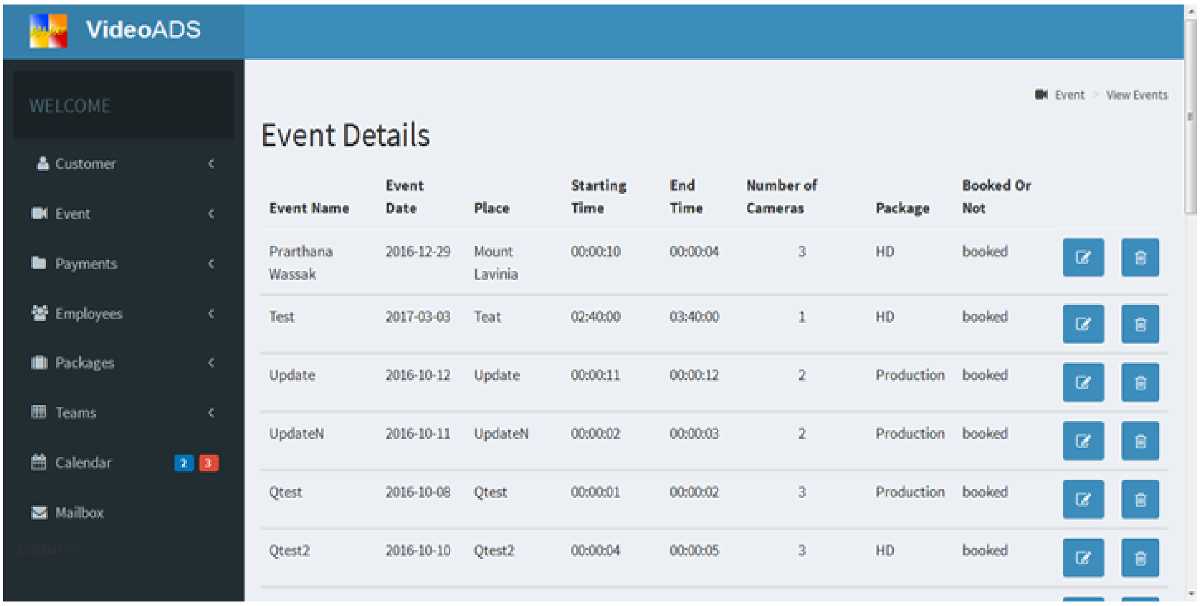
Customer Details

Name	Address	Contact_no		
Shanaka Kodagoda	205/A,Katuwana road,Homagama	0716364868		
Imalka Wevita	Niripola,Hanwella	0718135395		
Nimesha Sewwandi	Colombo Rd, Kalaniya	07789635221		

Figure 3.13- Customer Details View

3.5.9 Event Details

Figure 3.14 represents the Event Details View which shows the details of all events. This view facilitates the user to edit or delete event recodes.



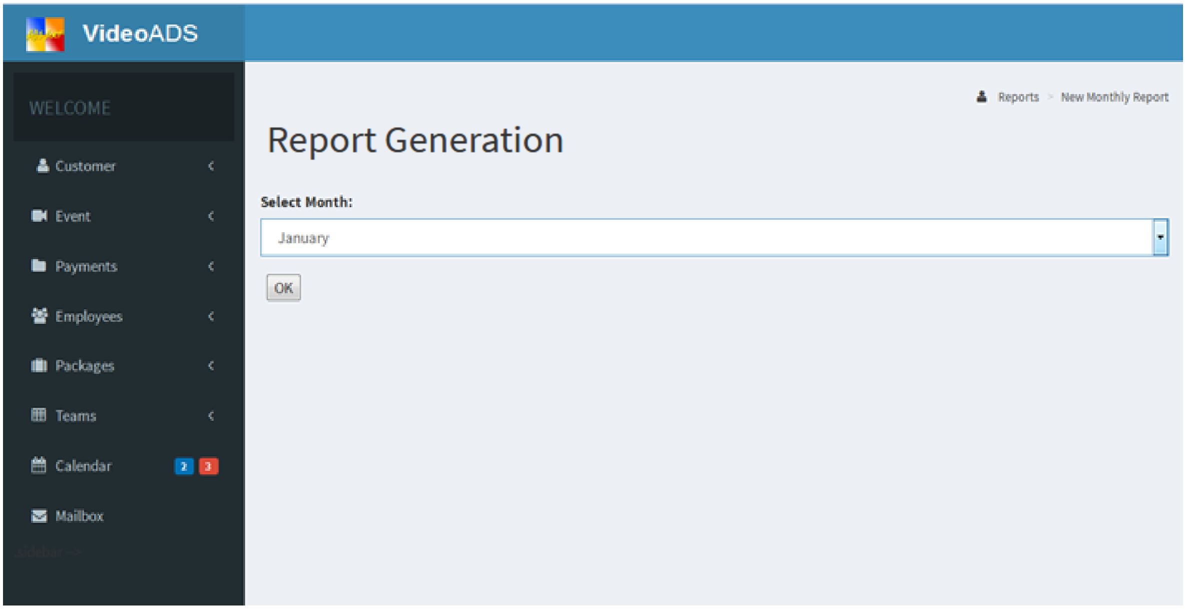
Event Details

Event Name	Event Date	Place	Starting Time	End Time	Number of Cameras	Package	Booked Or Not		
Prarthana Wassak	2016-12-29	Mount Lavinia	00:00:10	00:00:04	3	HD	booked		
Test	2017-03-03	Teat	02:40:00	03:40:00	1	HD	booked		
Update	2016-10-12	Update	00:00:11	00:00:12	2	Production	booked		
UpdateN	2016-10-11	UpdateN	00:00:02	00:00:03	2	Production	booked		
Qtest	2016-10-08	Qtest	00:00:01	00:00:02	3	Production	booked		
Qtest2	2016-10-10	Qtest2	00:00:04	00:00:05	3	HD	booked		

Figure 3.14 -Event Details View

3.5.10 Report Generation Form

Figure 3.15 shows the report form which can generate monthly reports. Month should be selected from the drop down list in order to generate the report of monthly handled events.



Report Generation

Select Month: January

Figure 3.15 -Report form

Chapter 04 – IMPLEMENTATION

Implementation is the third step of Software Development Life Cycle. Main objective of this phase is to convert the result of the design phase into programming codes using a suitable programming language. This is the most time consuming phase in SDLC. Time spend on the implementation phase to develop a proper set of codes can help to reduce the effort required in testing and maintenance phases.

The selected programming language to develop this system was PHP which is an object oriented language. When developing this system the time spent on the implementation phase was longer than the other phases, in order to ensure the maintainability, reusability, understandability and accuracy of the system codes, by following good programming practices such as use of comments and proper naming conventions etc.

4.1 Hardware and Software Requirements

4.1.1 Hardware Requirements

- Intel Core i5 Processor 2.50GHz
- 500 GB Hard Disk
- 4GB RAM
- Printer
-

4.1.2 Software Requirements

- Micrososft Windows 7 Home Primium
- WAMP Server 3.0.6
- MySQL 5.6.19
- PHP Version 5.4.6
- Mozilla Firefox Version 49.0.1

4.2 System Developed Tools and Technologies

The tools and techniques used in developing the system are listed below with brief descriptions.

- PHP

  This is the programming language used to develop the system which is an object oriented server side scripting language.

- MySQL

  This is the database software which used to the database of the system.

- Code Igniter

  This is an open source PHP framework that gives more supportive background when implementing the system.

- Visual Paradigm for UML

  This software was used to create the diagrams for system analysis.

- HTML

  This is the basic web related language which helps keep the system structure clear.

- Java Script

  This is a server side scripting language which helps to make dynamic system components.

- CSS

  This is a technique of adding styles for the system components.

- JSON

  JSON is syntax for storing and exchanging information.[WWW6]

- JQuery

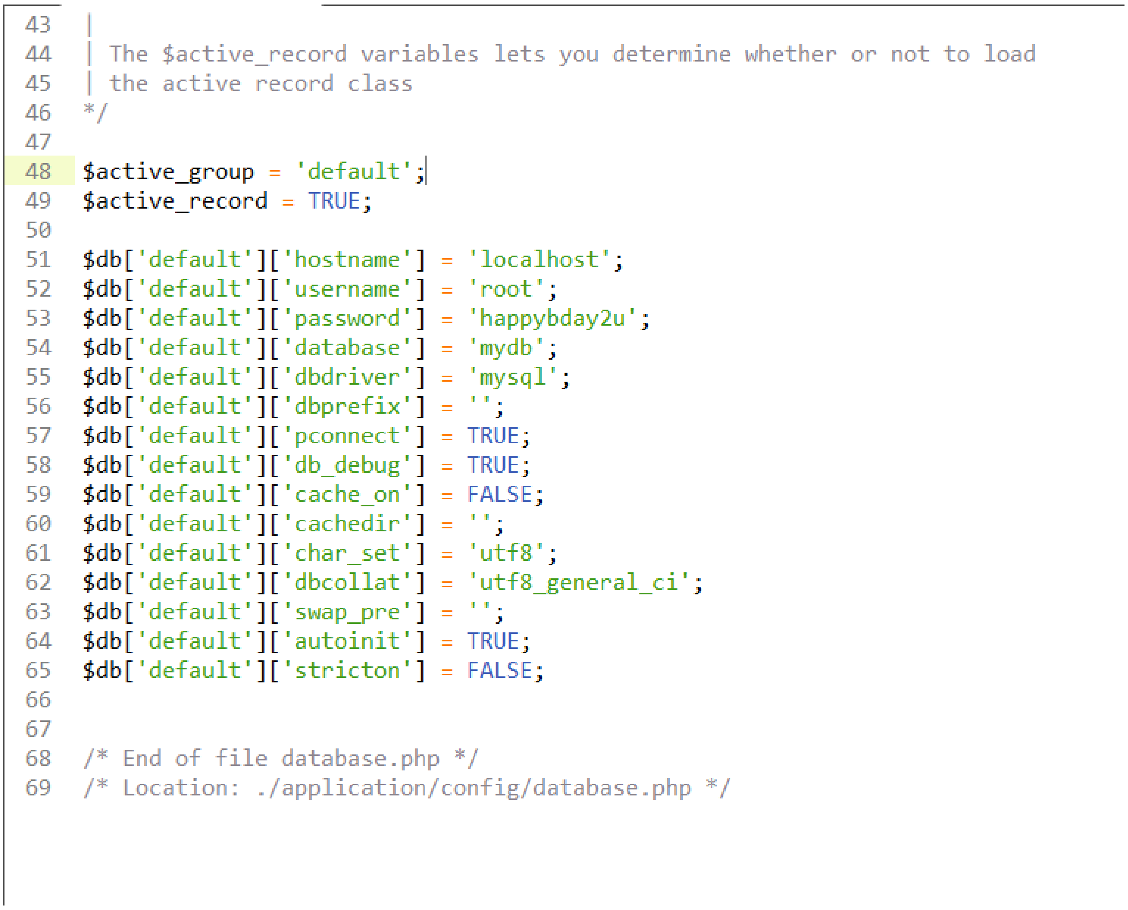
  This is a Java Script library which simplifies the client side scripting of HTML.

4.3 Major Code Segments

The main code modules developed in the system are briefly described below. Comments have been included in order to understand the function of codes.

- **Database Connection Page**

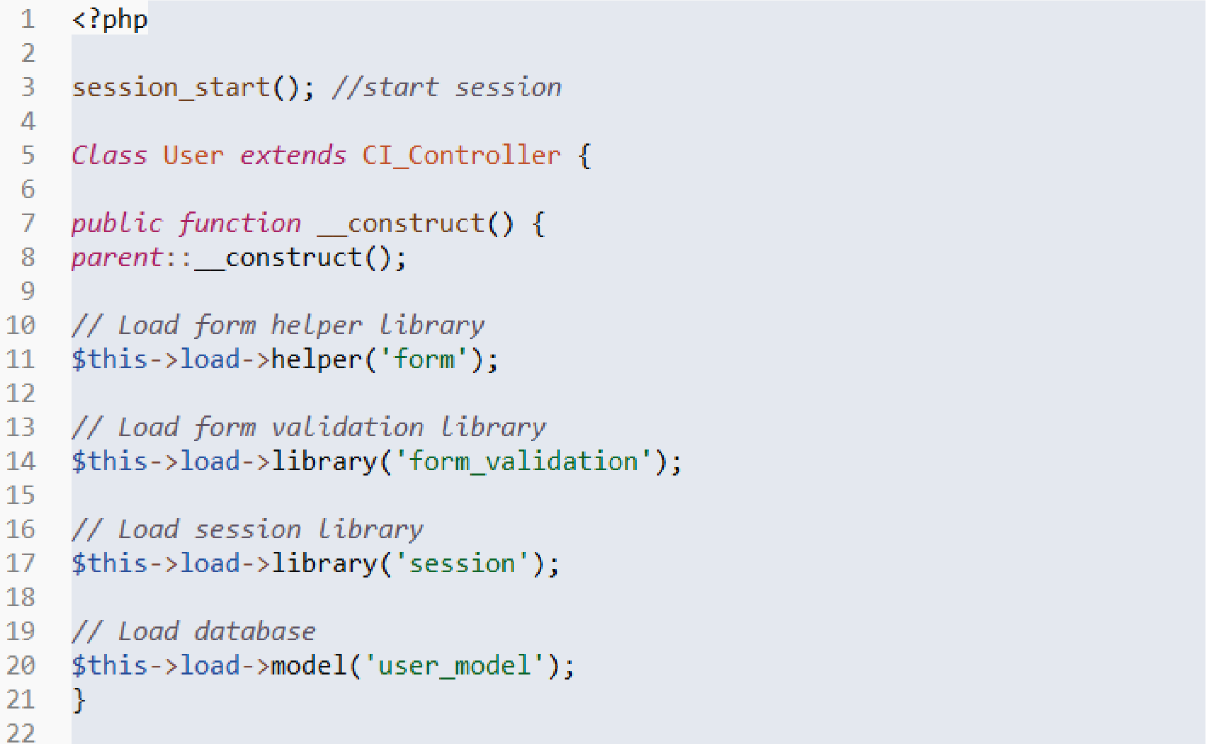
Connecting to a database is an essential part of the system coding. Every operation preformed on the system such as insertion, deletion, updating are required to deal with the related database. Code segment given below links the system to its database.



```
|
| The $active_record variables lets you determine whether or not to load
| the active record class
*/

$active_group = 'default';
$active_record = TRUE;

$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'root';
$db['default']['password'] = 'happybday2u';
$db['default']['database'] = 'mydb';
$db['default']['dbdriver'] = 'mysql';
$db['default']['dbprefix'] = '';
$db['default']['pconnect'] = TRUE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = '';
$db['default']['char_set'] = 'utf8';
$db['default']['dbcollat'] = 'utf8_general_ci';
$db['default']['swap_pre'] = '';
$db['default']['autoinit'] = TRUE;
$db['default']['stricton'] = FALSE;


/* End of file database.php */
/* Location: ./application/config/database.php */
```

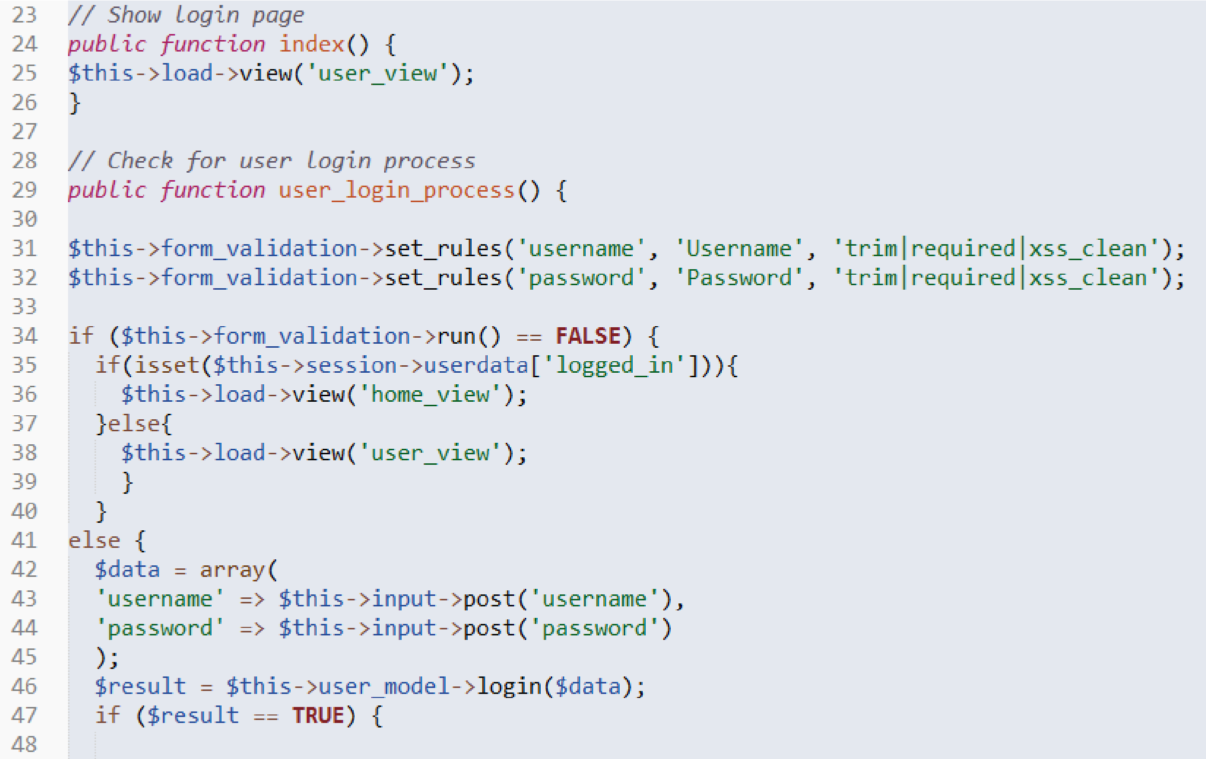
- **Login Handling Page**

Following code segment contains the codes to start the session, load the necessary libraries.

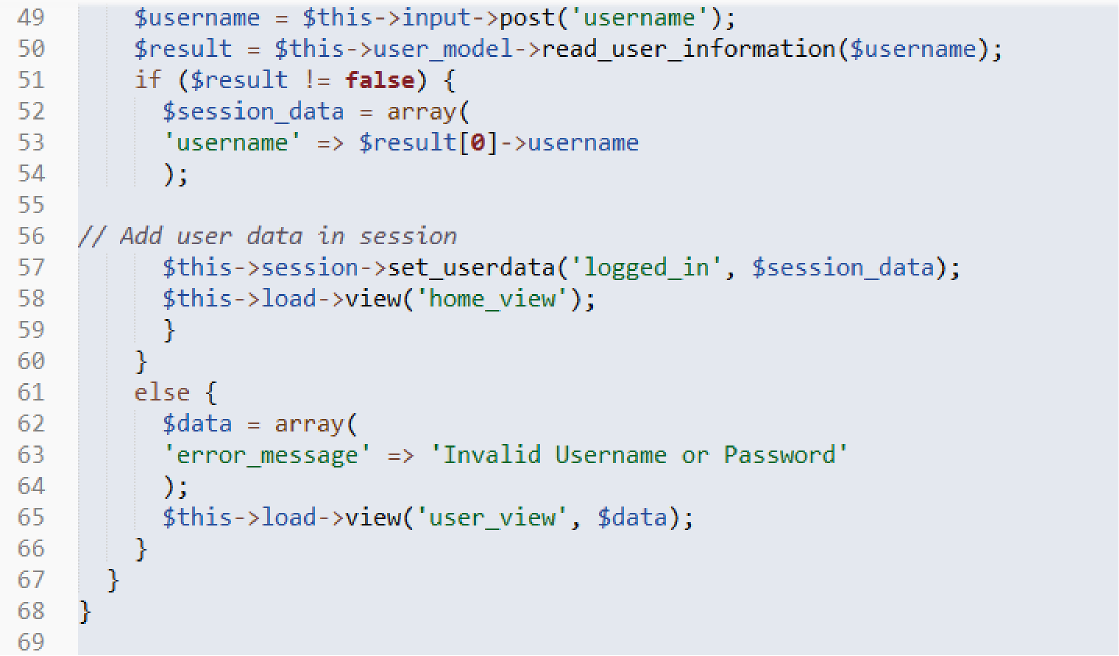


```php
<?php

session_start(); //start session

Class User extends CI_Controller {

public function __construct() {
parent::__construct();

// Load form helper library
$this->load->helper('form');

// Load form validation library
$this->load->library('form_validation');

// Load session library
$this->load->library('session');

// Load database
$this->load->model('user_model');
}

```

The following code segment contains the codes for and checking for the user login process and adding user data in session.

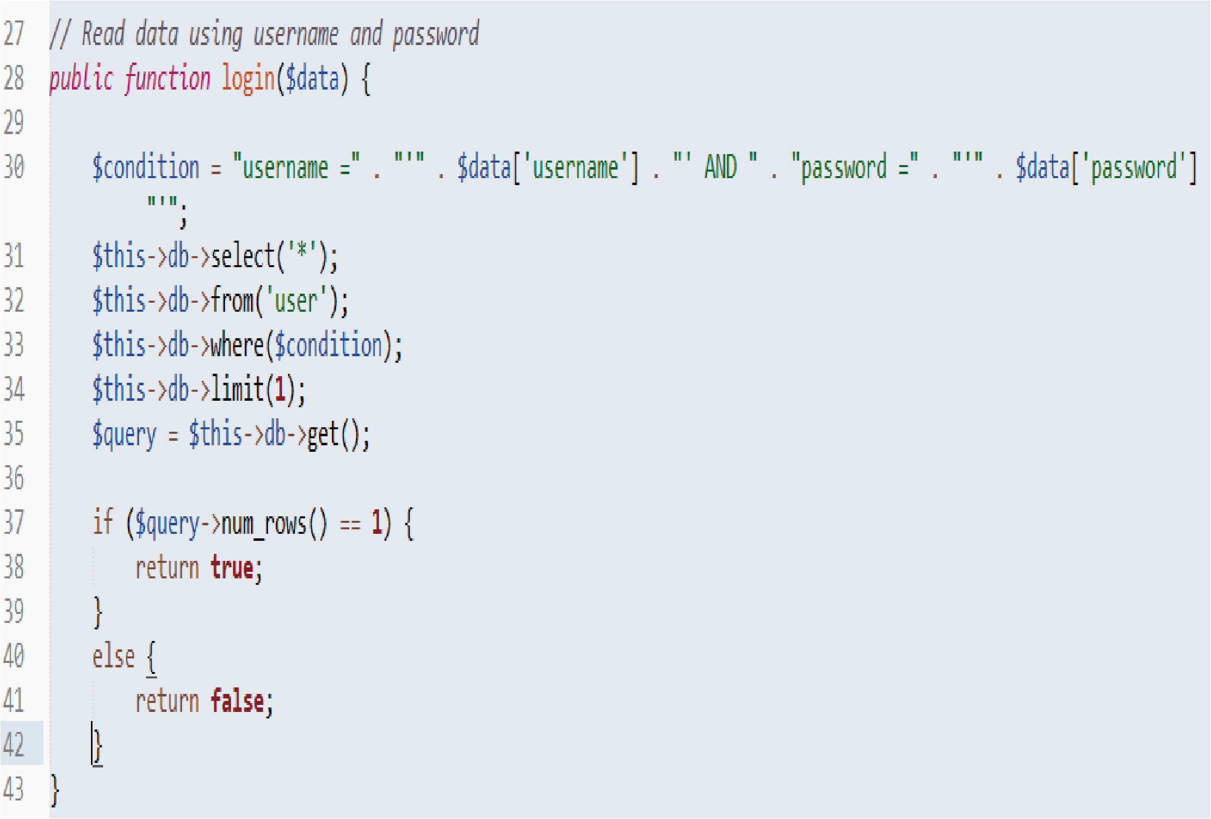


```php
// Show login page
public function index() {
$this->load->view('user_view');
}

// Check for user login process
public function user_login_process() {

$this->form_validation->set_rules('username', 'Username', 'trim|required|xss_clean');
$this->form_validation->set_rules('password', 'Password', 'trim|required|xss_clean');

if ($this->form_validation->run() == FALSE) {
  if(isset($this->session->userdata['logged_in'])){
    $this->load->view('home_view');
  }else{
    $this->load->view('user_view');
    }
  }
else {
  $data = array(
  'username' => $this->input->post('username'),
  'password' => $this->input->post('password')
  );
  $result = $this->user_model->login($data);
  if ($result == TRUE) {

```



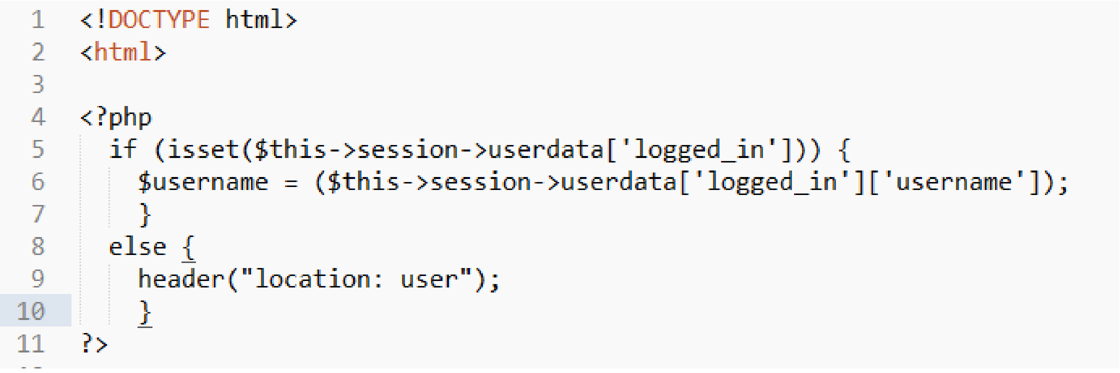
```
  $username = $this->input->post('username');
  $result = $this->user_model->read_user_information($username);
  if ($result != false) {
   $session_data = array(
   'username' => $result[0]->username
   );

// Add user data in session
   $this->session->set_userdata('logged_in', $session_data);
   $this->load->view('home_view');
   }
  }
  else {
   $data = array(
   'error_message' => 'Invalid Username or Password'
   );
   $this->load->view('user_view', $data);
  }
 }
}
```

Login details are checked with the data in the database by the following codes.



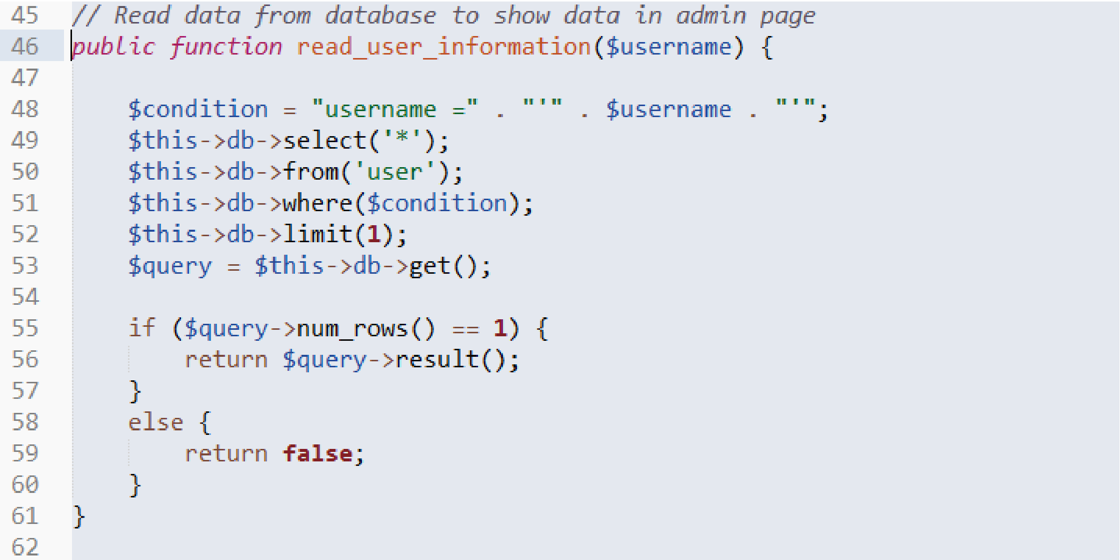
```
// Read data using username and password
public function login($data) {

    $condition = "username =" . "'" . $data['username'] . "' AND " . "password =" . "'" . $data['password']
        "'";
    $this->db->select('*');
    $this->db->from('user');
    $this->db->where($condition);
    $this->db->limit(1);
    $query = $this->db->get();

    if ($query->num_rows() == 1) {
        return true;
    }
    else {
        return false;
    }
}
```

If the user login is succeeded admin page is loaded. Following code segment is contained in the user admin page.



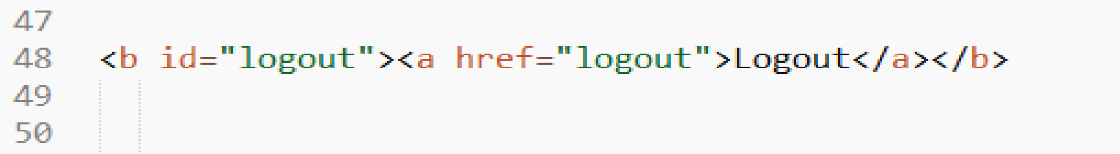
```
<!DOCTYPE html>
<html>

<?php
  if (isset($this->session->userdata['logged_in'])) {
    $username = ($this->session->userdata['logged_in']['username']);
    }
  else {
    header("location: user");
    }
?>
```

Following code segment is contained codes to read the data from the database to show in the admin page.



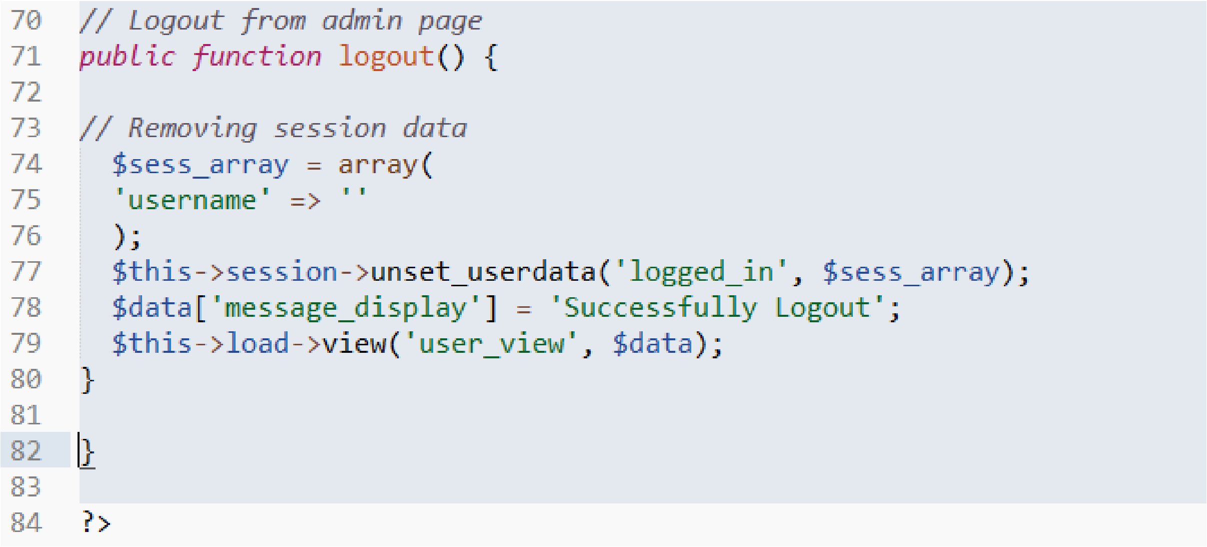
```
// Read data from database to show data in admin page
public function read_user_information($username) {

    $condition = "username =" . "'" . $username . "'";
    $this->db->select('*');
    $this->db->from('user');
    $this->db->where($condition);
    $this->db->limit(1);
    $query = $this->db->get();

    if ($query->num_rows() == 1) {
        return $query->result();
    }
    else {
        return false;
    }
}

```

If user wants to log out from the account, user has to follow link created by the following code segment.



```

  <b id="logout"><a href="logout">Logout</a></b>


```

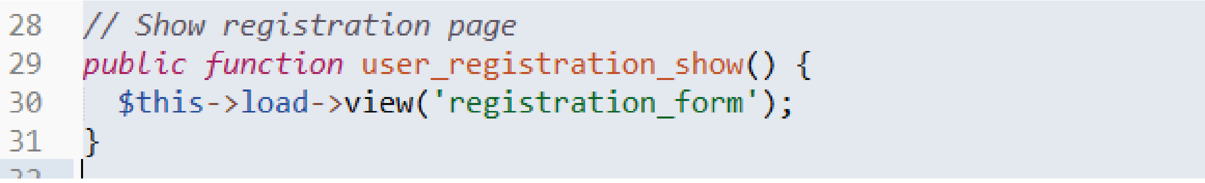
The following code segment contains the codes for the logout process.



```
// Logout from admin page
public function logout() {

// Removing session data
  $sess_array = array(
  'username' => ''
  );
  $this->session->unset_userdata('logged_in', $sess_array);
  $data['message_display'] = 'Successfully Logout';
  $this->load->view('user_view', $data);
}

}

?>
```

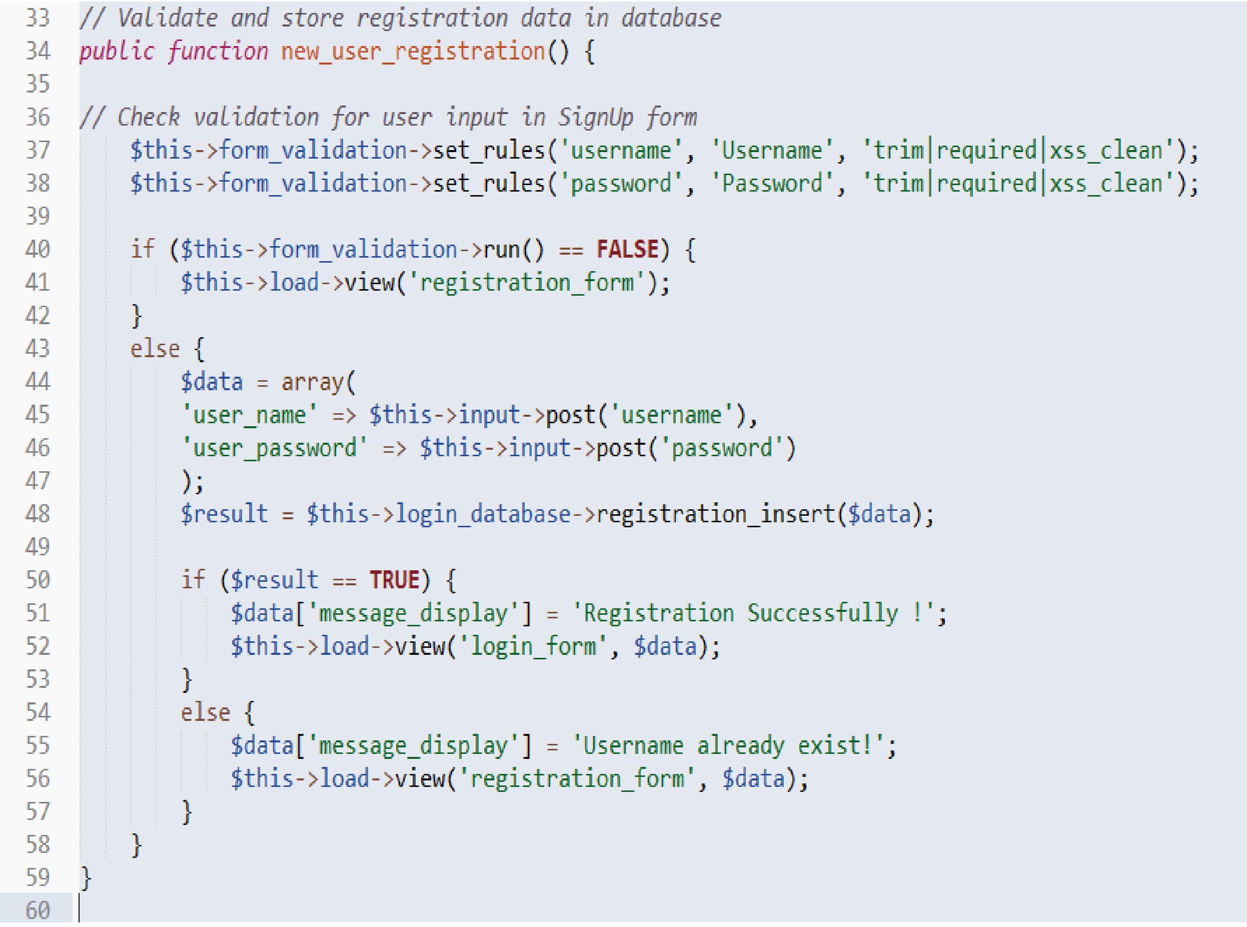
- **New user registration**

Creating a new user could be done by the system administrator by clicking on the ‘new user’ link on his admin page. The following code segment contains the codes for go to process of new user creation.



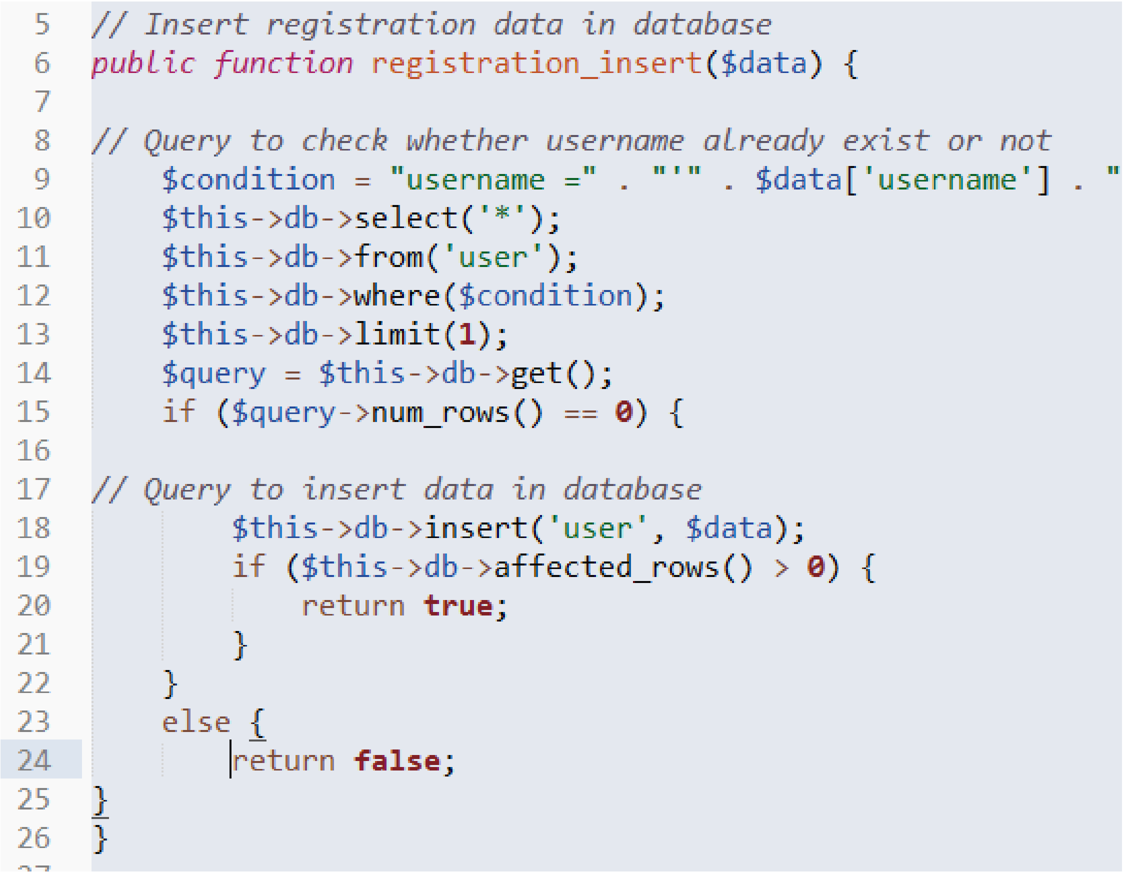
```
// Show registration page
public function user_registration_show() {
  $this->load->view('registration_form');
}
```

The following code segment contains the codes for validating and storing new user registration data in the database.



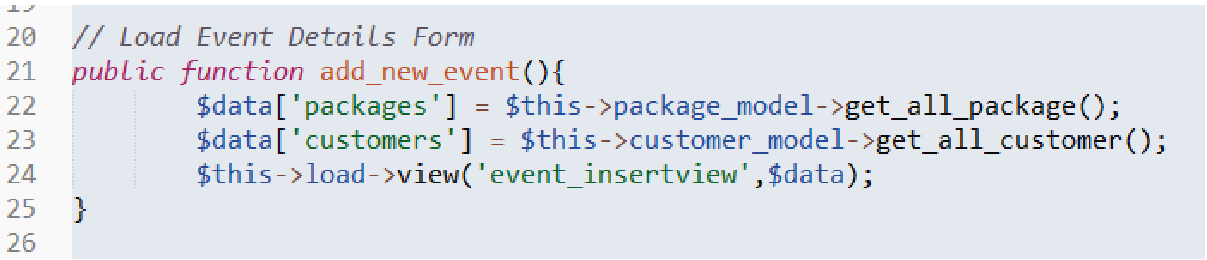
```php
// Validate and store registration data in database
public function new_user_registration() {

// Check validation for user input in SignUp form
    $this->form_validation->set_rules('username', 'Username', 'trim|required|xss_clean');
    $this->form_validation->set_rules('password', 'Password', 'trim|required|xss_clean');

    if ($this->form_validation->run() == FALSE) {
        $this->load->view('registration_form');
    }
    else {
        $data = array(
        'user_name' => $this->input->post('username'),
        'user_password' => $this->input->post('password')
        );
        $result = $this->login_database->registration_insert($data);

        if ($result == TRUE) {
            $data['message_display'] = 'Registration Successfully !';
            $this->load->view('login_form', $data);
        }
        else {
            $data['message_display'] = 'Username already exist!';
            $this->load->view('registration_form', $data);
        }
    }
}
```

Inserting registration data to the database is showed by the following code segment.



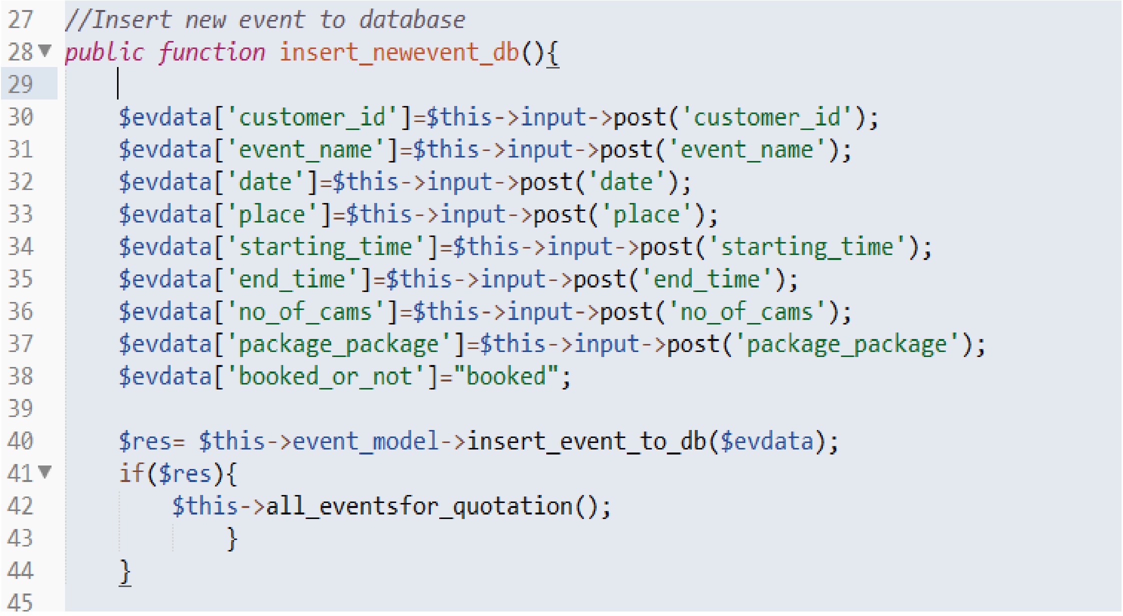
```php
// Insert registration data in database
public function registration_insert($data) {

// Query to check whether username already exist or not
    $condition = "username =" . "'" . $data['username'] . "
    $this->db->select('*');
    $this->db->from('user');
    $this->db->where($condition);
    $this->db->limit(1);
    $query = $this->db->get();
    if ($query->num_rows() == 0) {

// Query to insert data in database
        $this->db->insert('user', $data);
        if ($this->db->affected_rows() > 0) {
            return true;
        }
    }
    else {
        return false;
}
}
```

- **Event Details Management**
  Inserting, deleting, updating and viewing event details are done by the following given codes.
  Loading the event details form for inserting a new event is done by the following code segment. Package and the Customer name should be selected by the user from a drop down list. Codes for passing of package data and customer data to the Event Details Form are also included as follows.



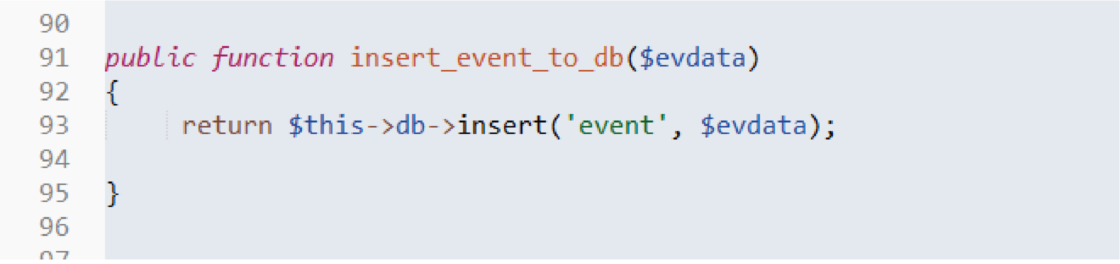
```
// Load Event Details Form
public function add_new_event(){
        $data['packages'] = $this->package_model->get_all_package();
        $data['customers'] = $this->customer_model->get_all_customer();
        $this->load->view('event_insertview',$data);
}

```

Following code segment contains the codes for passing of user input data to the database.



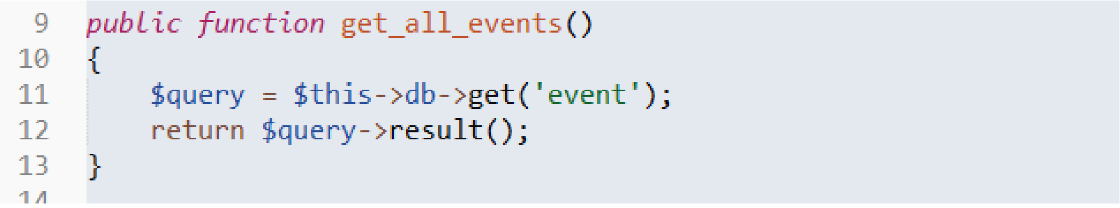
```
//Insert new event to database
public function insert_newevent_db(){

    $evdata['customer_id']=$this->input->post('customer_id');
    $evdata['event_name']=$this->input->post('event_name');
    $evdata['date']=$this->input->post('date');
    $evdata['place']=$this->input->post('place');
    $evdata['starting_time']=$this->input->post('starting_time');
    $evdata['end_time']=$this->input->post('end_time');
    $evdata['no_of_cams']=$this->input->post('no_of_cams');
    $evdata['package_package']=$this->input->post('package_package');
    $evdata['booked_or_not']="booked";

    $res= $this->event_model->insert_event_to_db($evdata);
    if($res){
        $this->all_eventsfor_quotation();
            }
    }

```

Storing of event data in the database is done by the following code segment.



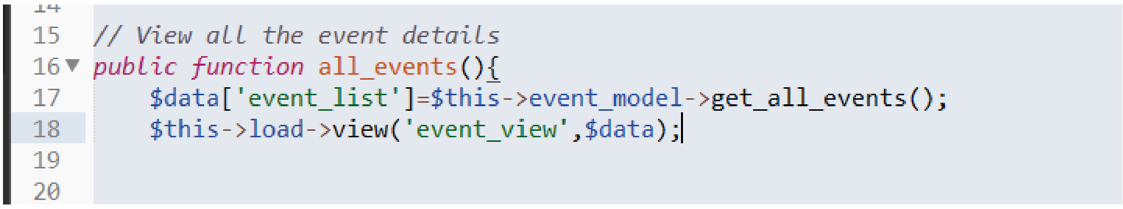
```
public function insert_event_to_db($evdata)
{
    return $this->db->insert('event', $evdata);

}
```

User is allowed to view all the event details stored in the database. Retrieving of all the event details from the database are given in following code segment.



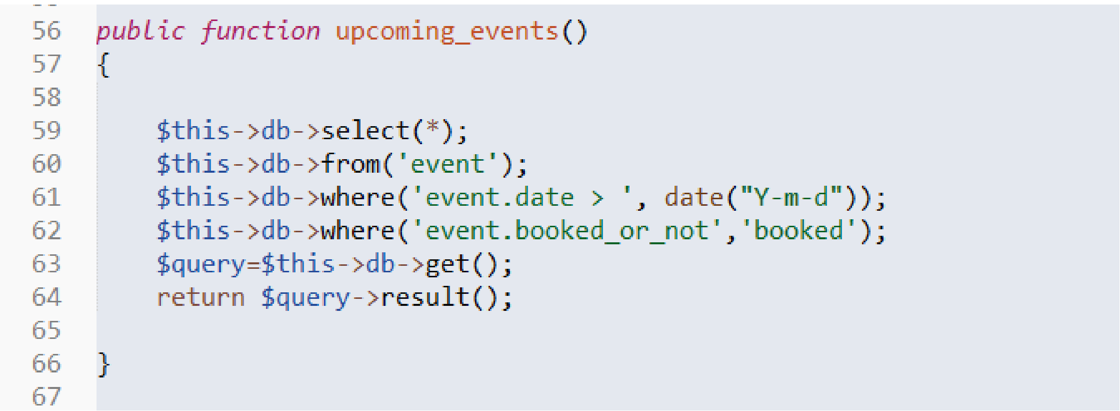
```
public function get_all_events()
{
    $query = $this->db->get('event');
    return $query->result();
}
```

Retrieved data of all the events are passed to the Event details page by the following codes.



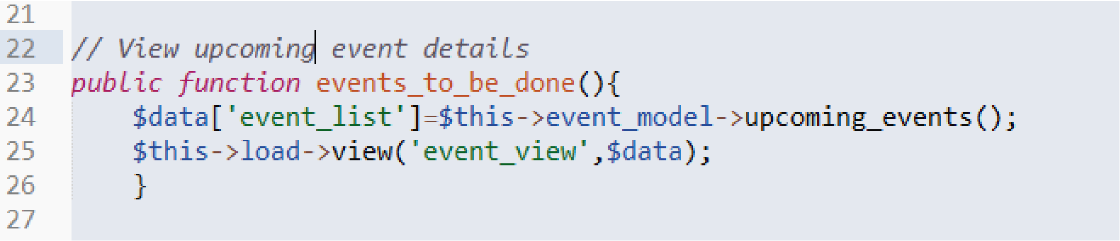
```
// View all the event details
public function all_events(){
    $data['event_list']=$this->event_model->get_all_events();
    $this->load->view('event_view',$data);
```

User is allowed to check only the upcoming events which have been booked by the customer. The following segment of codes contains the function of retrieving data to facilitate it.



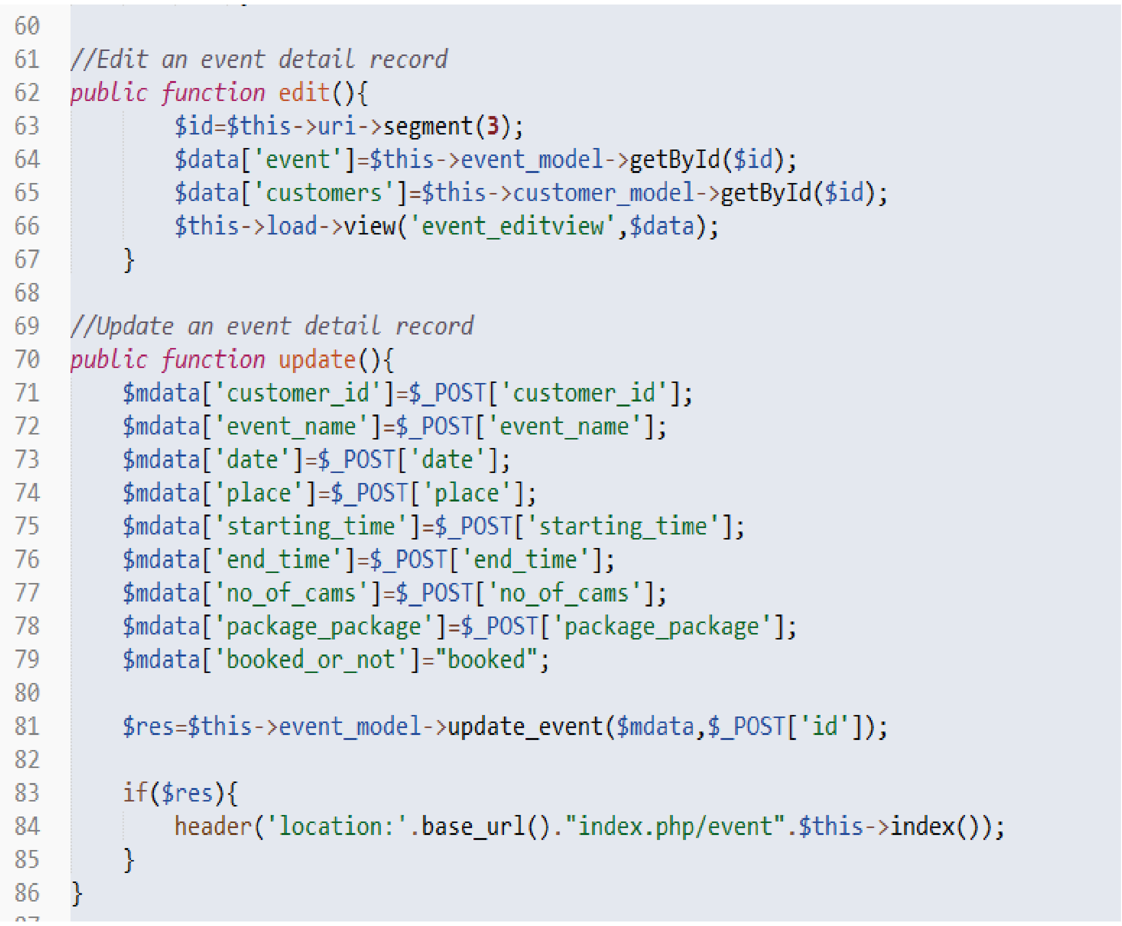
```
public function upcoming_events()
{

    $this->db->select(*);
    $this->db->from('event');
    $this->db->where('event.date > ', date("Y-m-d"));
    $this->db->where('event.booked_or_not','booked');
    $query=$this->db->get();
    return $query->result();

}
```

Retrieved data are passed to the view as follows.



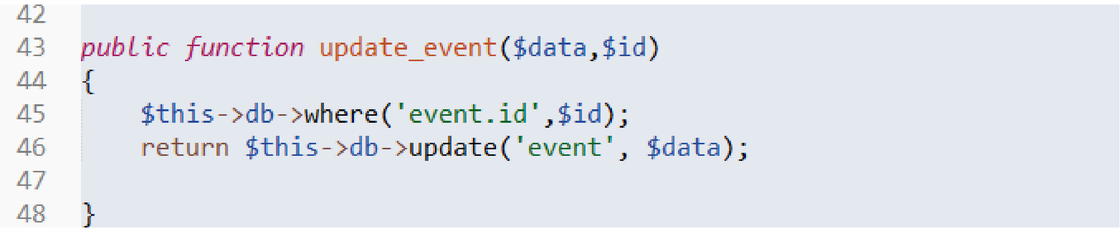
```php

// View upcoming event details
public function events_to_be_done(){
    $data['event_list']=$this->event_model->upcoming_events();
    $this->load->view('event_view',$data);
    }

```

User is allowed to edit and update the event data showed in the Event Details page by the following codes.



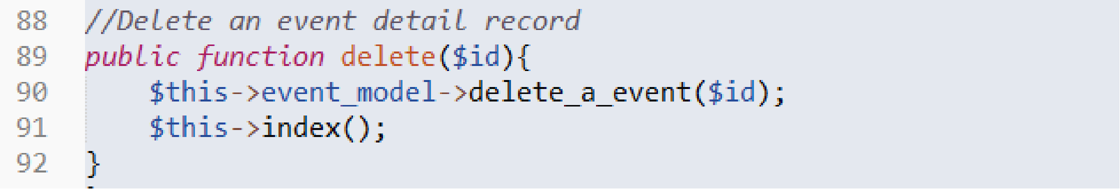
```php

//Edit an event detail record
public function edit(){
        $id=$this->uri->segment(3);
        $data['event']=$this->event_model->getById($id);
        $data['customers']=$this->customer_model->getById($id);
        $this->load->view('event_editview',$data);
    }

//Update an event detail record
public function update(){
    $mdata['customer_id']=$_POST['customer_id'];
    $mdata['event_name']=$_POST['event_name'];
    $mdata['date']=$_POST['date'];
    $mdata['place']=$_POST['place'];
    $mdata['starting_time']=$_POST['starting_time'];
    $mdata['end_time']=$_POST['end_time'];
    $mdata['no_of_cams']=$_POST['no_of_cams'];
    $mdata['package_package']=$_POST['package_package'];
    $mdata['booked_or_not']="booked";

    $res=$this->event_model->update_event($mdata,$_POST['id']);

    if($res){
        header('location:'.base_url()."index.php/event".$this->index());
    }
}
```

Updating of database for the edited event is done by the following segment of codes.



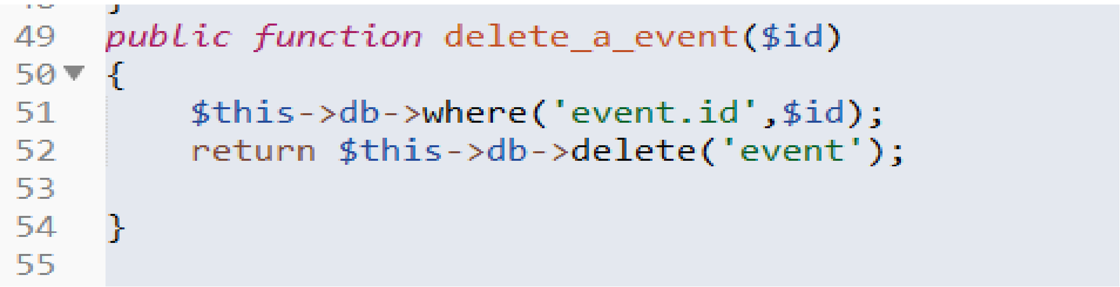
```
public function update_event($data,$id)
{
    $this->db->where('event.id',$id);
    return $this->db->update('event', $data);

}
```

Further user is facilitated to delete an event showed on the Event Details page by the following code segment.



```
//Delete an event detail record
public function delete($id){
    $this->event_model->delete_a_event($id);
    $this->index();
}
```

Deleting the database recode of the event is done by the following code segment.



```
public function delete_a_event($id)
{
    $this->db->where('event.id',$id);
    return $this->db->delete('event');

}
```

4.4 Reused Existing Codes

Some codes were found by referring internet and used in this project. Extracted codes were customized and well tested. Following codes are extracted from the websites.

- Codes for validating and storing login details were written by referring internet.
- Codes for displaying data on the calendar view were written by referring internet.
- Some CSS styles were taken from the internet to design the interfaces.

Chapter 05 – EVALUATION

Testing is the process of evaluating a system or its component with the intent to find whether it satisfies the specified requirements or not. Further testing is executing a system in order to identify any gaps, errors or missing requirements in contrary to the actual desire or requirements [7]

5.1 Software Testing

Software testing is a task that should be performed during the development. Verification and validation are done in this process.

Verification is the process to make sure the product satisfies the conditions imposed at the start of the development phase. In other words, to make sure the product behaves the way we want it to.[8]

Validation is determining if the system complies with the requirements and performs functions for which it is intended and meets the organization's goals and user needs.

[9]

5.2 Techniques of Software Testing

There are two techniques of software testing.

1. Black box Testing

Black box testing is a testing technique that ignores the internal mechanism of the system and focuses on the output generated against any input and execution of the system. It is also called functional testing.[10]

2. White box Testing

White box testing is a testing technique that takes into account the internal mechanism of a system. It is also called structural testing and glass box testing.

Black box testing is often used for validation and white box testing is often used for verification. [10]

5.3Types of Testing

5.3.1 Unit Testing

Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that the unit he/she has implemented is producing expected output against given input.[8]

5.4.2 Integration Testing

Integration testing is testing in which a group of components are combined to produce output. Also, the interaction between software and hardware is tested in integration testing if software and hardware components have any relation. It may fall under both white box testing and black box testing. [8]

5.4.3 System Testing

System testing is the testing to ensure that by putting the software in different environments (e.g., Operating Systems) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing. [8]

5.4.4 Acceptance Testing

Acceptance testing is often done by the customer to ensure that the delivered product meets the requirements and works as the customer expected. It falls under the class of black box testing.[8]

5.4.5 Regression Testing

Regression testing is the testing after modification of a system, component, or a group of related units to ensure that the modification is working correctly and is not damaging or imposing other modules to produce unexpected results. It falls under the class of black box testing..[8]

5.4 Test Plan and Test Cases

Test plan is detail information about the scope and schedule of the testing, test deliverables etc. It gives how the testing will proceed, who will do the testing, what will be tested, in how much time the test will take place, and to what quality level the test will be performed. Following table show high level test plan of the transport management system.[10]

Module Name	**Function Name**	**Test Priority**
User Management Module	Insert new user	High
	Delete user	High
	Confirm deletion	High
	Edit User	High
	Confirm editing of user	High
	View user	High
	Login user	Medium
	Logout user	High
Customer Management Module	Insert new customer	High
	Delete customer	High
	Confirm deletion	High
	Edit customer	High
	Confirm editing of customer	High
	View customers	High
	Search customer	Medium

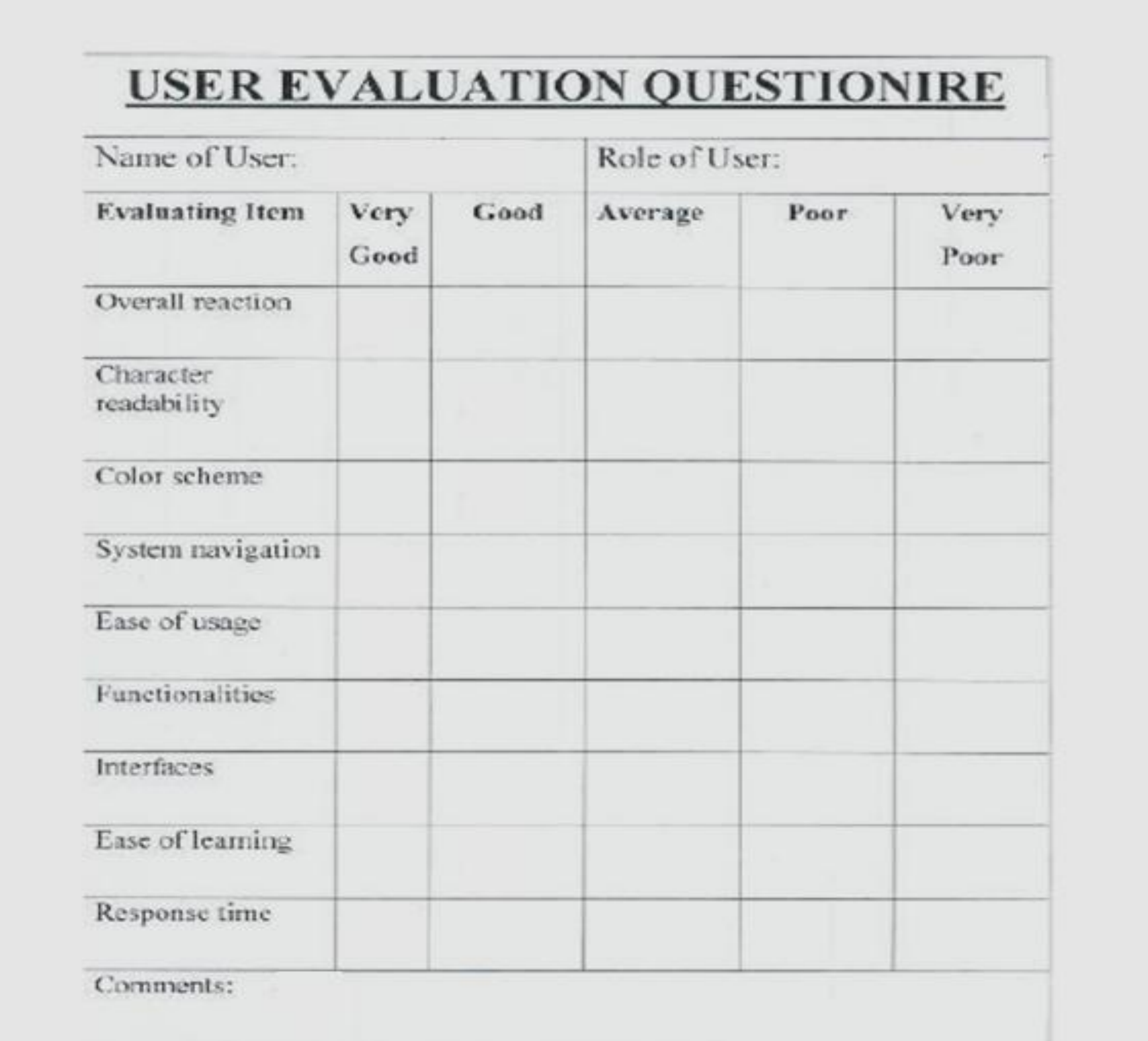
Employee Details Management Module	Insert new employee	High
	Delete employee	High
	Confirm deletion	High
	Edit employee	High
	Confirm editing of employee	High
	View employees	High
	Search employee	Medium
Event Details Management Module	Insert new event	High
	Delete event	High
	Confirm deletion	High
	Edit event	High
	Confirm editing of event	High
	View events	High
	Search event	High
Quotation Management Module	Create new quotation	High
	Delete quotation	High
	Confirm deletion	High
	Edit quotation	Medium
	Confirm editing of quotation	High
	View quotations	High
	Search quotation	Low

Payment Details Management Module	Insert new payment	High
	Delete payment	High
	Confirm deletion	High
	Edit payment	High
	Confirm editing of payment	High
	View payments	High
Package Details Management Module	Insert new package	High
	Delete package	High
	Confirm deletion	High
	Edit package	High
	Confirm editing of package	High
	View packages	High
Team Management Module	Assign a team	Low
	View team	High
	Edit team	Low
	Confirm editing of team	High
	Delete team	High
	Confirm deletion	High

Reminding Massages Management Module	Creating Reminding Email	Low
	Sending Reminding Email	Low
Notification Module	View Notifications	High
Report Generation Module	Generating monthly event details report	High
	Generating monthly income report	Medium
	Generating customer-event details report	Low

Table 5. 1-Test Plan

5.5 User Evaluation



USER EVALUATION QUESTIONIRE

Name of User:

Role of User:

Evaluating Item	Very Good	Good	Average	Poor	Very Poor
Overall reaction					
Character readability					
Color scheme					
System navigation					
Ease of usage					
Functionalities					
Interfaces					
Ease of learning					
Response time					

Comments:

Figure 5.1-Uuser evaluation questionnaire

Above figure 5.1 shows user evaluation questionnaire.

In this System the user evaluation was done by selection different users. In the client’s company Manager has been taken as an administrator of the system and other users has taken as normal users with different privileges. User evaluation questionnaire was given to the users and results has summarized.

Chapter 06 – CONCLUSION

Video ADS (pvt) Ltd is a developing video production company in the country. They were in need of enhancing their business with new technology. The manual system they used earlier was time consuming unreliable process. Now with the new Event Management System, they are carrying out their event booking and other related tasks effectively. Many difficulties they faced earlier mentioned in analysis chapter, have been solved by the new system. Since all the day to day activities have automated through this system, now they can enhance their business easily. With the use of report generation process of this new system, they can analyze their business status and forecasting about future business.

6.1 Lessons Learnt

This project was a vast practical experience in many fields. . Applications of programming theories are practically understood throughout this project. Managing the pressure with the time constraint was a good challenge which has been overcome at the end. This was a good life lesson which helped to improve the self confident.

6.2 Critical Assessment of the System

Comparison with existing similar systems

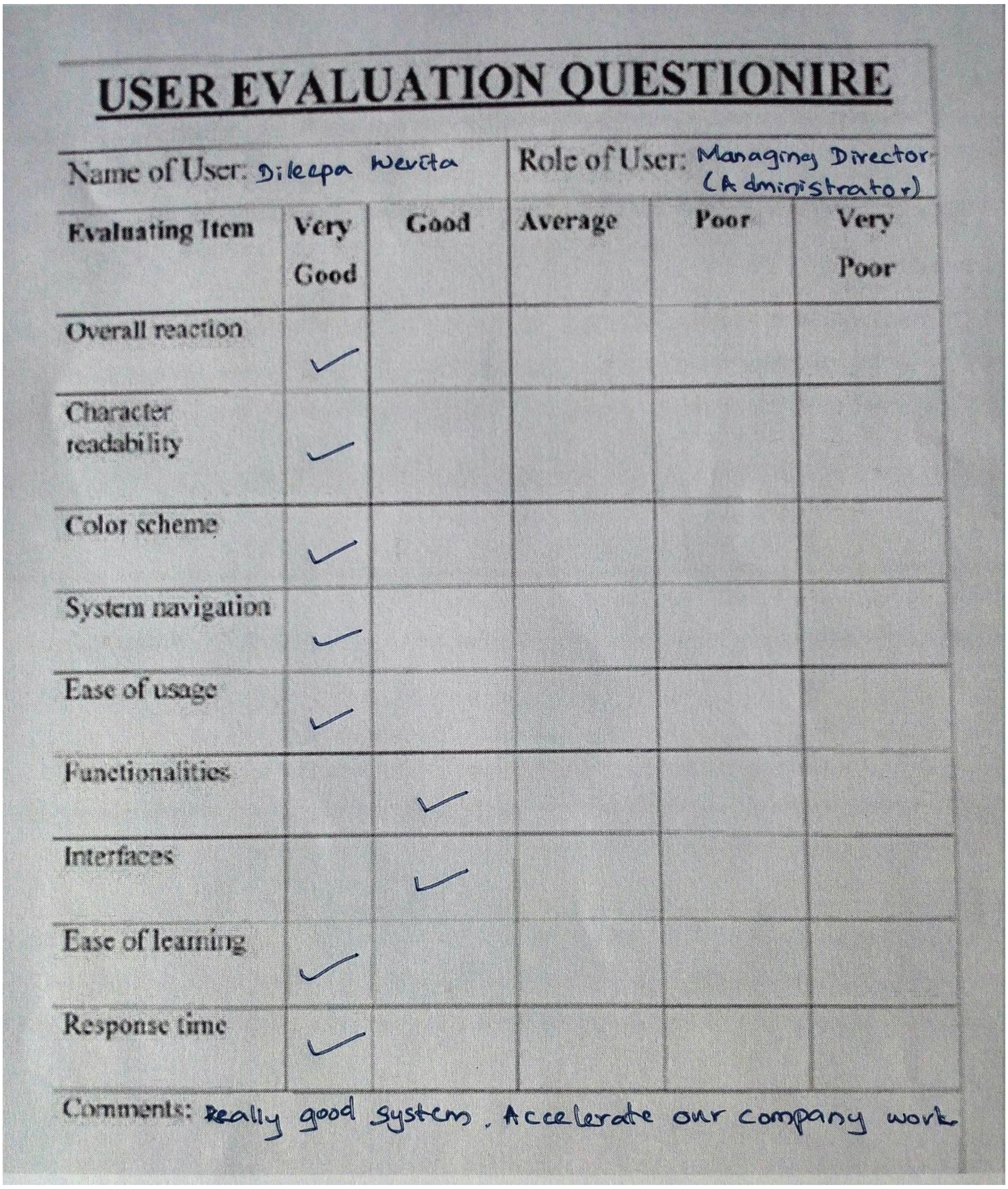
Event Management System for Video ADS Company		Existing similar systems	
		Tradify	Zoho Projects
Similarities	Manage event details	Manage event details	Manage event details
	Show events on calendar	Show events on calendar	Show events on calendar
	Allow the user to insert payments for events and print quotation for each event	Allow the user to insert payments for events and print quotation for each event	Allow the user to insert payments for events and print quotation for each event
	Facilitating different user levels	Facilitating	Facilitating

		different user levels	different user levels
	Generate event report for given period of time	Generate event report for given period of time	Generate event report for given period of time
Differences	Manage customer details	Not managing customer details	Not managing customer details
	Clearly depicts payment status of the events from the calendar	Not showing payment status of the events from the calendar	Not showing payment status of the events from the calendar
	Give specific details of an event such as number of cameras by clicking on the event on the calendar	Not giving the most relevant details of an event such as number of cameras by clicking on the event	Not giving the most relevant details of an event such as number of cameras by clicking on the event
	Give number of available cameras by clicking on a calendar date	Not showing number of available cameras by clicking on a calendar date	Not showing number of available cameras by clicking on a calendar date
	Allow the user to insert an employee to the system by inserting specific employee details such as employee position which can be selected	Not allowing the user to insert an employee to the system by inserting specific employee details such as employee position	Not allowing the user to insert an employee to the system by inserting specific employee details such as employee position
	Allow the user to insert and view package details which are specific for the company	Cannot insert package details	Cannot insert package details
	Allow the user to allocate employees to create a team for an event	Not facilitating for team management	Not facilitating for team management
	Generate employee event report for given period of time	No employee event report is available	No employee event report is available
	Generate package report for given period of time	No package report is available	No package report is available

Table 6. 1- Comparison with existing similar systems

User Evaluation

User evaluation has done by selecting different users of the system. In this Event management system Managing Director has been taken as an administrator of the system and other users have been taken as normal users with different privileges. User evaluation questionnaire was given to target population and results has summarized. Following figure 6.1, 6.2 and 6.3 show user evaluation questionnaires.



USER EVALUATION QUESTIONIRE

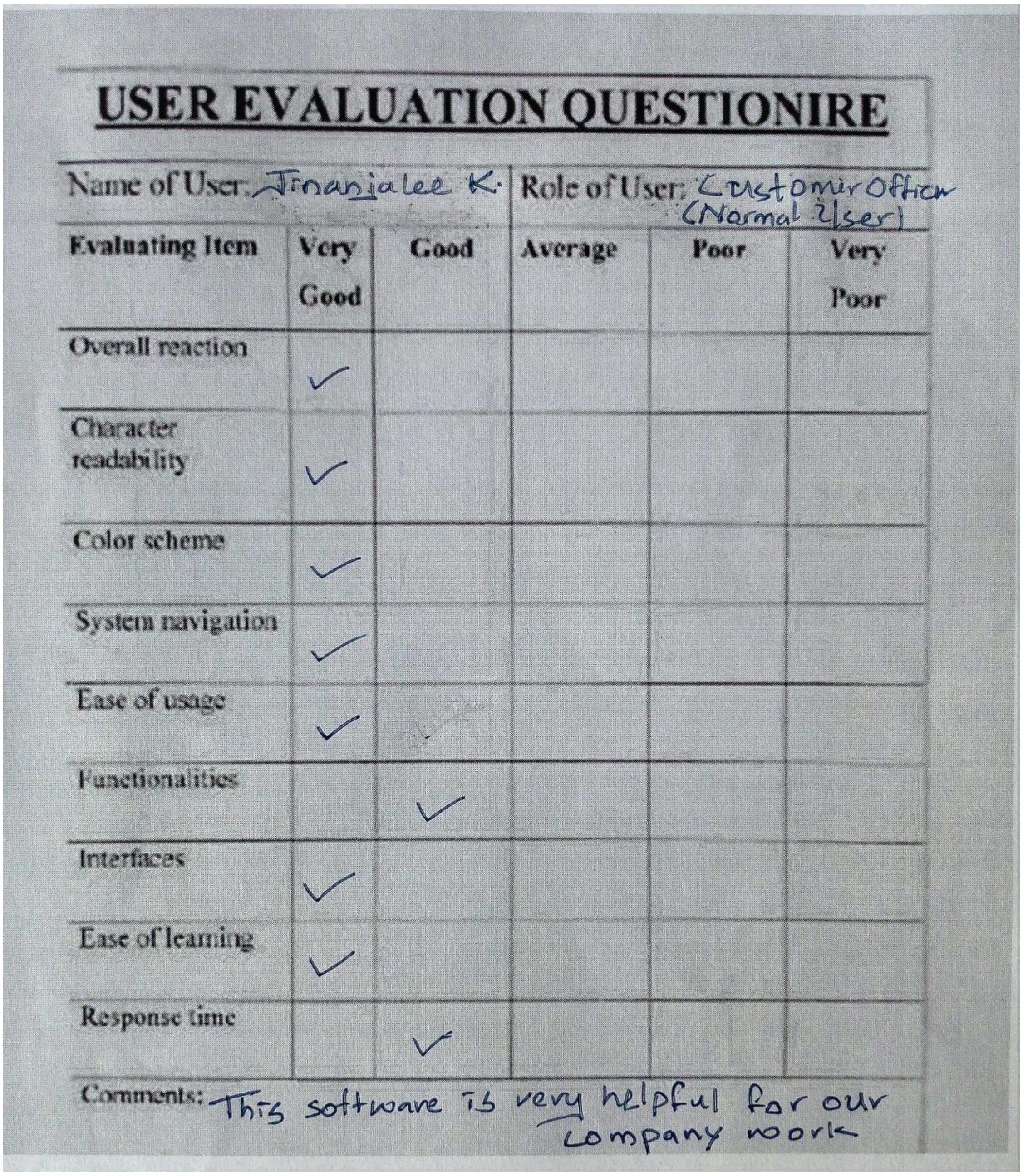
Name of User: Dilepa Werita

Role of User: Managing Director (Administrator)

Evaluating Item	Very Good	Good	Average	Poor	Very Poor
Overall reaction	✓				
Character readability	✓				
Color scheme	✓				
System navigation	✓				
Ease of usage	✓				
Functionalities		✓			
Interfaces		✓			
Ease of learning	✓				
Response time	✓				

Comments: Really good system . Accelerate our company work

Figure 6.1-Uuser evaluation questionnaire given to an Administrator



USER EVALUATION QUESTIONIRE

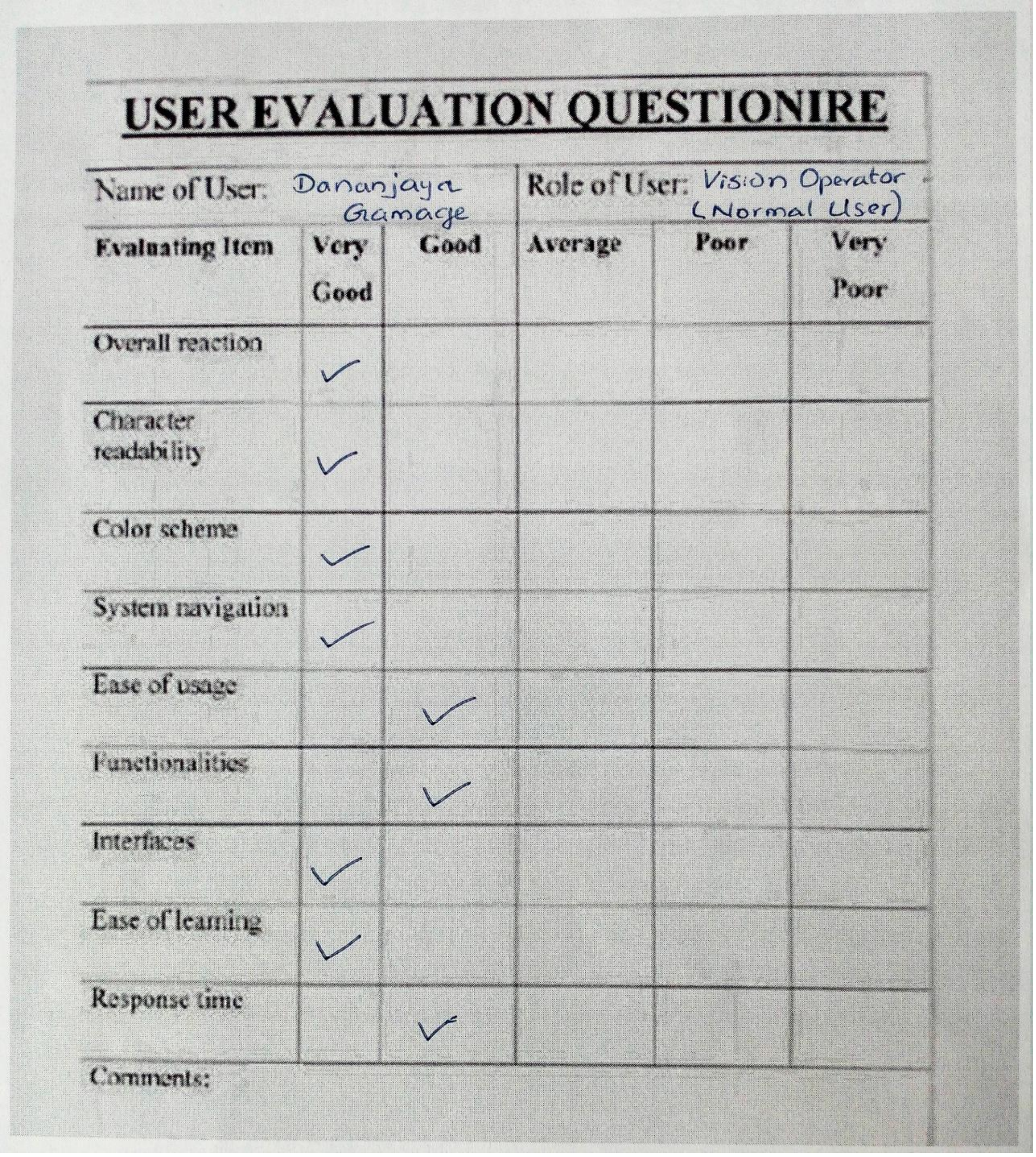
Name of User: Jinanjalee K.

Role of User: Customer Officer (Normal User)

Evaluating Item	Very Good	Good	Average	Poor	Very Poor
Overall reaction	✓				
Character readability	✓				
Color scheme	✓				
System navigation	✓				
Ease of usage	✓				
Functionalities		✓			
Interfaces	✓				
Ease of learning	✓				
Response time		✓			

Comments: This software is very helpful for our company work

Figure 6.2-Uuser evaluation questionnaire given to a normal user



USER EVALUATION QUESTIONIRE

Name of User: Dananjaya Gamage

Role of User: Vision Operator - (Normal User)

Evaluating Item	Very Good	Good	Average	Poor	Very Poor
Overall reaction	✓				
Character readability	✓				
Color scheme	✓				
System navigation	✓				
Ease of usage		✓			
Functionalities		✓			
Interfaces	✓				
Ease of learning	✓				
Response time		✓			

Comments:

Figure 6.3-Uuser evaluation questionnaire given to another normal user

When considering all the user evaluation results it is reasonable to mention that the overall system performance is well suit to the Video ADS Company and it satisfies the user requirements of the system successfully.

6.3 Future Enhancement

This Event Management System was developed within a limited period and with the functionalities which were agreed by the client. But it is open to further enhancement with many new features that support the business process more than it does. Following are the some of the future enhancement of the system.

- Use of Google maps to find the time taken to travel for event and find out the most convenient rout to travel.
- Conversion of generating emails into SMS which is more efficient way to give a massage.
- Use of bar code reading system to use when they take out the equipments for events to ensure the security of the system.
- Allow to book multiple events on a day.

References

- [1] System Design http://www.slideshare.net/syedmuhammadhammad/software-design-13430869 [June 2017]
- [2] System Development Life Cycle https://www.techwalla.com/articles/steps-in-the-system-development-life-cycle [June 2017]
- [3] Database design http://en.wikipedia.org/wiki/Database_design [June 2017
- [4] Database Normalization https://en.wikipedia.org/wiki/Systems_design [June 2017]
- [5] Interface Design http://bokardo.com/principles-of-user-interface-design/ [July 2017]
- [6] JSON http://www.w3schools.com/js/js_json_intro.asp [July 2017]
- [7] Evaluation http://www.tutorialspoint.com/software_testing/ [July 2017]
- [8] Software testing types http://www.codeproject.com/Tips/351122/What-is-software-testing-What-are-the-different-ty [July 2017]
- [9] Validation http://istqbexamcertification.com/what-is-validation-in-software-testing-or-what-is-software-validation/ [July 2017]
- [10] Test plan http://testingsoftware.blogspot.com/2005/11/what-is-test-plan_30.html
- [July 2017]

Appendix A–SYSTEM DOCUMENTATION

System Manual

Step1:-

Install the Wamp Server 3.0.6

Download and install WAMP for Windows from http://www.wampserver.com .

Install web browser

Install web browser Google Chrom Version 54.0.2840.71m or Mozilla Firefox Version 49.0.1

Step2 :-

Open the CD and copy the EMS folder and paste it to the directory path "C:\wamp\www".

Step 3:-

Database Installation

Open the web browser and type the URL http://localhost/phpmyadmin/ and enter username and password if you have set before.

Create empty database by providing name as 'mydb'.

Navigate to the 'Import' tab and click on 'Browse' button on the 'File to import' section.

Then browse the CD and select the file 'DB.sql' in the 'Database' folder.

Then click on 'Go' button located on the bottom of the page.

Launching the System

Open the installed web browser and type the URL http://localhost/EMS and press 'Enter' button to access the system.

Appendix B – DESIGN DOCUMENTATION

Use case diagrams and descriptions

- Use case diagram of the Customer Management module

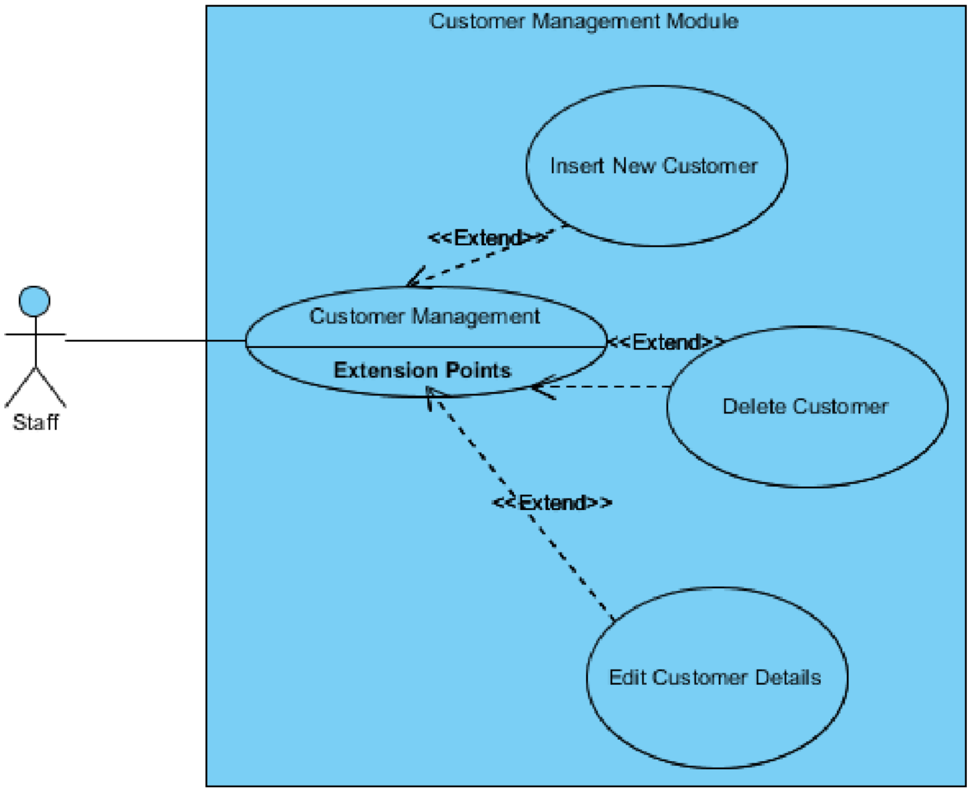


Figure B. 1 -Use case diagram of the Customer Management module

Use Case Name	Add New Customer	
Actor	Staff	
Description	Staff inserts the new customer details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Enter the customer Details	
	Click on Submit Button	Display all customer details
Post Condition	Customer data stored in the database.	

Table B. 1- Use case descriptions Add New Customer

Use Case Name	Delete Customer	
Actor	Staff	
Description	Staff deletes the customer details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Delete button	Confirmation box will appear.
	Confirms the deletion.	
Post Condition	Customer data is deleted from the database.	

Table B. 2- Use case description Delete Customer

Use Case Name	Edit Customer	
Actor	Staff	
Description	Staff edits the customer details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Edit button	Confirmation box will appear.
	Confirms the edition.	
Post Condition	Customer data is updated in the database.	

Table B. 3- Use case description Staff

- Use case diagram of the Event Details Management module

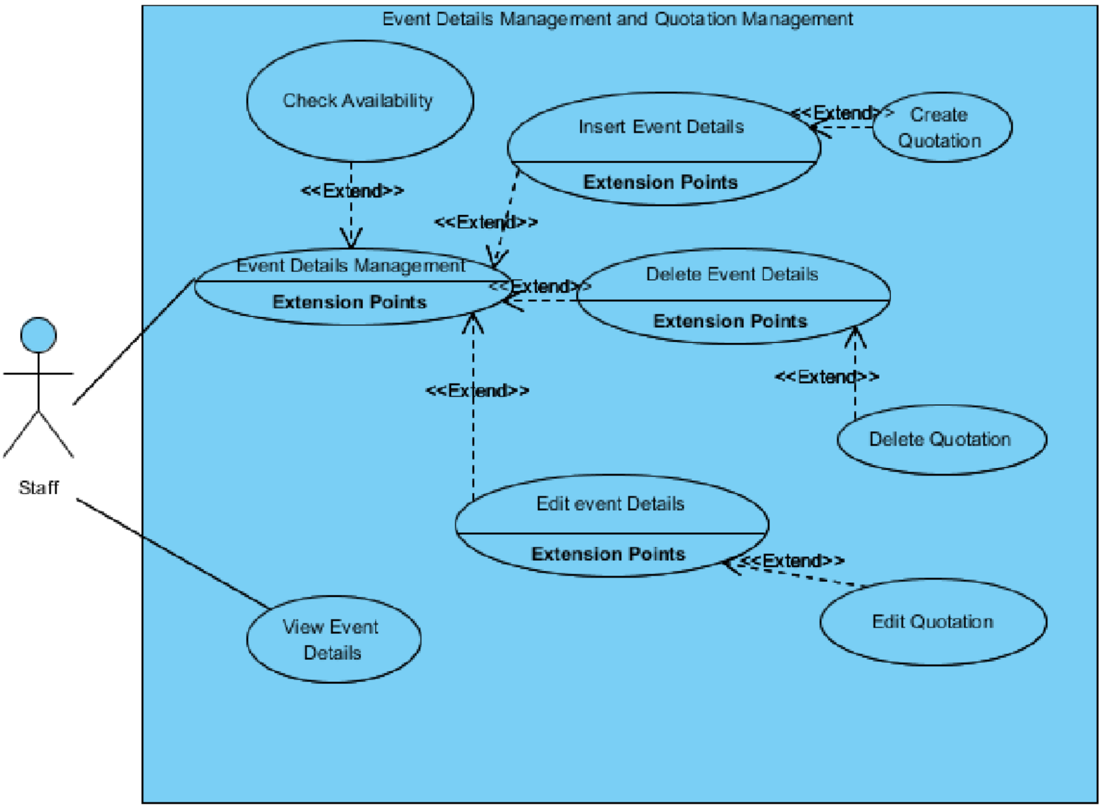


Figure B. 2 -Use case diagram of the Event Details Management module

Use Case Name	Check Availability	
Actor	Staff	
Description	Staff checks the availability of the customer requested date.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
		Display the event calendar.
Post Condition		

Table B. 4- Use case description Check Availability

Use Case Name	Insert Event Details	
Actor	Staff	
Description	Staff inserts the event details.	
Pre-conditions	System user should be logged into the system Customer details should be entered to the system.	
Typical course of events	**Actor Action**	**System Response**
	Select the customer name.	
	Enter event details.	
	Click on Submit button.	Quotation Details form will appear.
	Enter quotation details.	
	Click on Submit button.	Quotation will appear.
	Click on Print button.	Quotation will be printed.
Post Condition	Event data is stored in the database. Quotation table is up dated.	

Table B. 5- Use case description Insert Event Details

Use Case Name	Delete Event Details	
Actor	Staff	
Description	Staff deletes the event details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Delete button	Confirmation box will appear.
	Confirms the deletion.	
Post Condition	Customer data is deleted from the database. Quotation data is deleted in the database.	

Table B. 6- Use case description Delete Event Details

Use Case Name	Edit Event	
Actor	Staff	
Description	Staff edits the event details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Edit button	Confirmation box will appear.
	Confirms the edition.	
Post Condition	Event data is updated in the database. Quotation data is updated in the database.	

Table B. 7- Use case description Edit Event

- Use case diagram of the Payment Management module

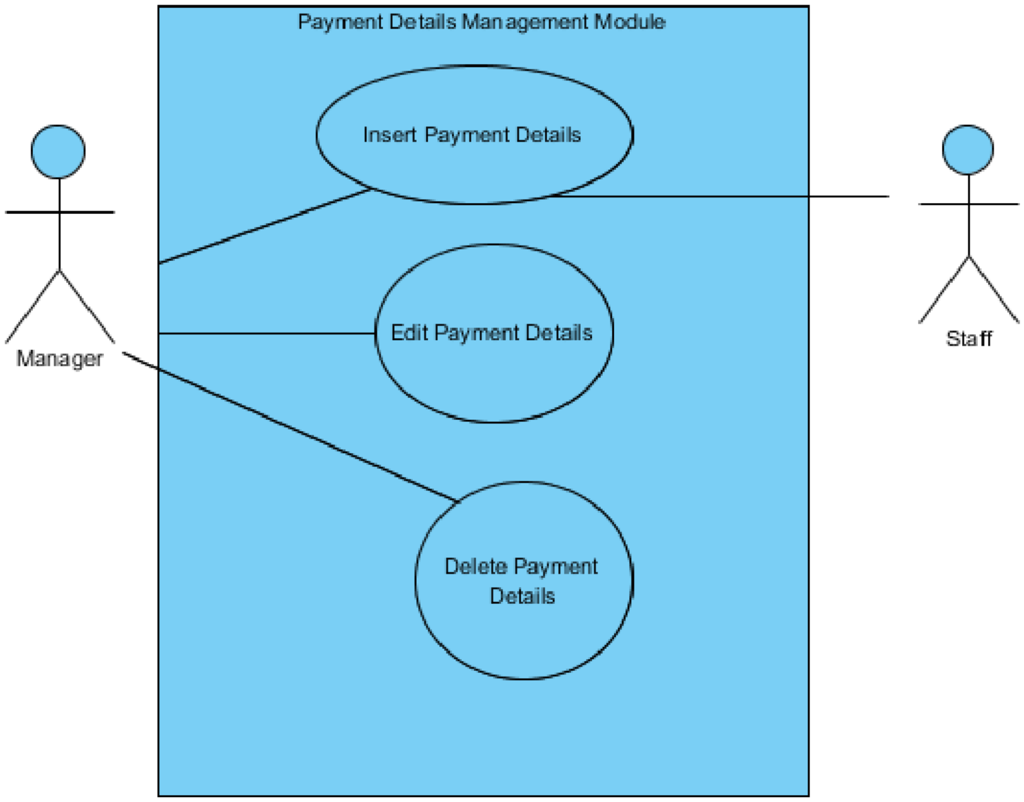


Figure B. 3-Use case diagram of the Payment Management module

Use Case Name	Insert Payment Details	
Actor	Staff	
Description	Staff inserts payment details.	
Pre-conditions	Staff should be logged into the system.	
Typical course of events	**Actor Action**	**System Response**
	Enter payment details.	
	Click on Submit button.	
Post Condition	Payment data is updated in the database.	
Use Case Name	Delete Payment Details	
Actor	Manager	
Description	Manager deletes the payment details.	

Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Delete button	Confirmation box will appear.
	Confirms the deletion.	
Post Condition	Payment data is deleted from the database.	

Table B. 8- Use case description Insert Payment Details

Use Case Name	Edit Payment	
Actor	Manager	
Description	Manager edits the payment details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Edit button	Confirmation box will appear.
	Confirms the edition.	
Post Condition	Payment data is updated in the database.	

Table B. 9 -Use case description Edit Payment

- Use case diagram of the Employee Management module

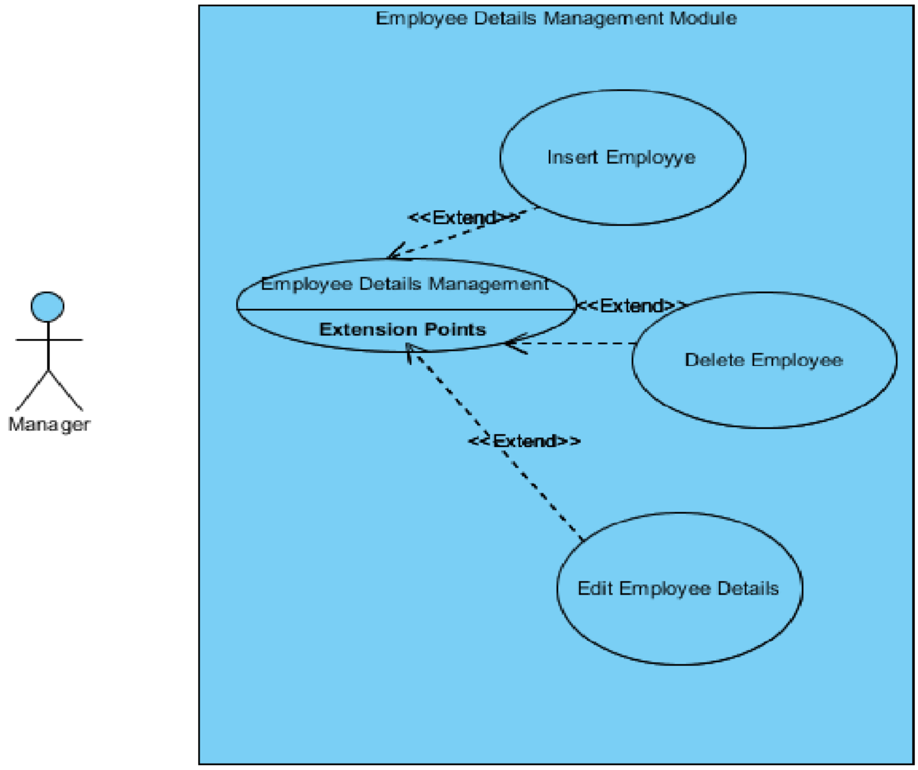


Figure B. 4 -Use case diagram of the Employee Management module

Use Case Name	Add New Employee	
Actor	Manager	
Description	Manager inserts the new employee details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Enter the Employee Details	
	Click on Submit Button	Display all employee details
Post Condition	Employee data stored in the database.	

Table B. 10-Use case description Add New Employee

Use Case Name	Delete Employee	
Actor	Manager	
Description	Manager deletes the employee details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Delete button	Confirmation box will appear.
	Confirms the deletion.	
Post Condition	Employee data is deleted from the database.	

Table B. 11-Use case description Delete Employee

Use Case Name	Edit Payment	
Actor	Staff	
Description	Staff edits the employee details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Edit button	Confirmation box will appear.
	Confirms the edition.	
Post Condition	Employee data is updated in the database.	

Table B. 12-Use case description Edit Payment

- Use case diagram of the Package Management module

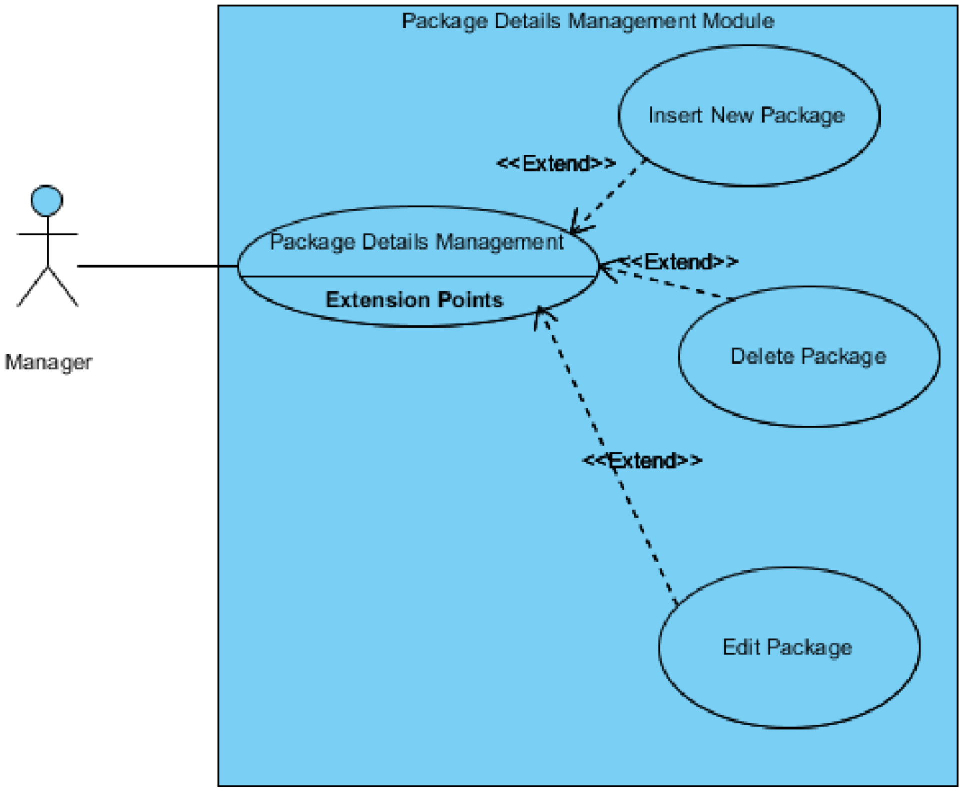


Figure B. 5 -Use case diagram of the Package Management module

Use Case Name	Add New Package	
Actor	Staff	
Description	Staff inserts the new package details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Enter the Package Details	
	Click on Submit Button	Display all employee details
Post Condition	Package data stored in the database.	

Table B. 13-Use case description Add New Package

Use Case Name	Delete Package	
Actor	Staff	
Description	Staff deletes the package details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Delete button	Confirmation box will appear.
	Confirms the deletion.	
Post Condition	Package data is deleted from the database.	

Table B. 14-Use case description Delete Package

Use Case Name	Edit Package	
Actor	Staff	
Description	Staff edits the package details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Edit button	Confirmation box will appear.
	Confirms the edition.	
Post Condition	Package data is updated in the database.	

Table B. 15-Use case description Edit Package

- Use case diagram of the User Management module

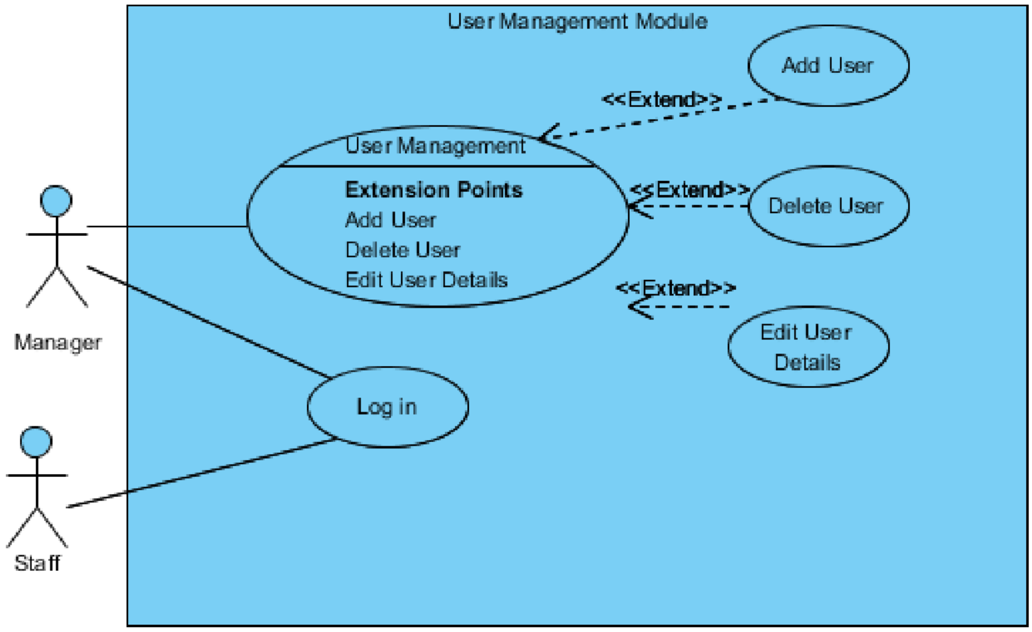


Figure B. 6 -Use case diagram of the User Management module

Use Case Name	Add New User	
Actor	Manager	
Description	Manager inserts the new user details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Enter the user Details	
	Click on Submit Button	Display alert box
Post Condition	User data stored in the database.	

Table B. 16-Use case description Add New User

Use Case Name	User Login	
Actor	Staff	
Description	User log in to the system.	
Pre-conditions	User should have the username and password or user should have the verification email.	
Typical course of events	**Actor Action**	**System Response**
	If user has the username and password, user enters them to system.	Log in to the system.
	If the user has the verification email at the first time he log on to the system, user enters the verification code.	Allows the user to change the password.
Post Condition	Show the home page of the user. Store the password in the database.	

Table B. 17-Use case description User Login

Use Case Name	Delete User	
Actor	Manager	
Description	Manager deletes the user details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Delete button	Confirmation box will appear.
	Confirms the deletion.	
Post Condition	User data are deleted from the database. User is no more allowed to access the system.	

Table B. 18-Use case description Delete User

Use Case Name	Edit User	
Actor	Manager	
Description	Manager edits the user details.	
Pre-conditions	System user should be logged into the system	
Typical course of events	**Actor Action**	**System Response**
	Click on the Edit button	Confirmation box will appear.
	Confirms the edition.	
Post Condition	User data is updated in the database.	

Table B. 19-Use case description Edit User

Activity Diagrams

- Activity Diagram for Event Details Management Module

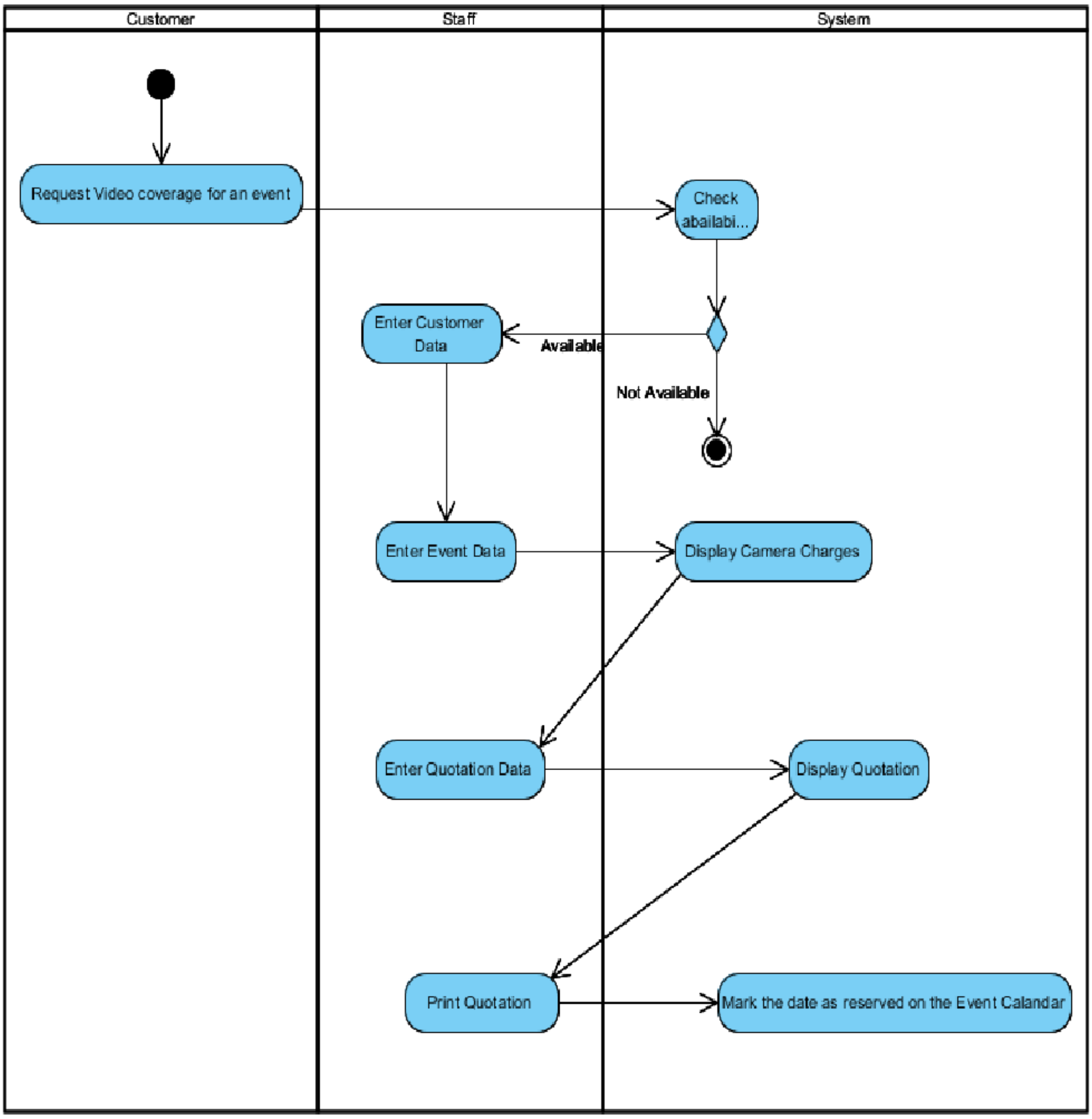


Figure B. 7 - Activity Diagram for Event Details Management Module

Sequence diagram

- Sequence diagram for Reserving an event date for a new customer.

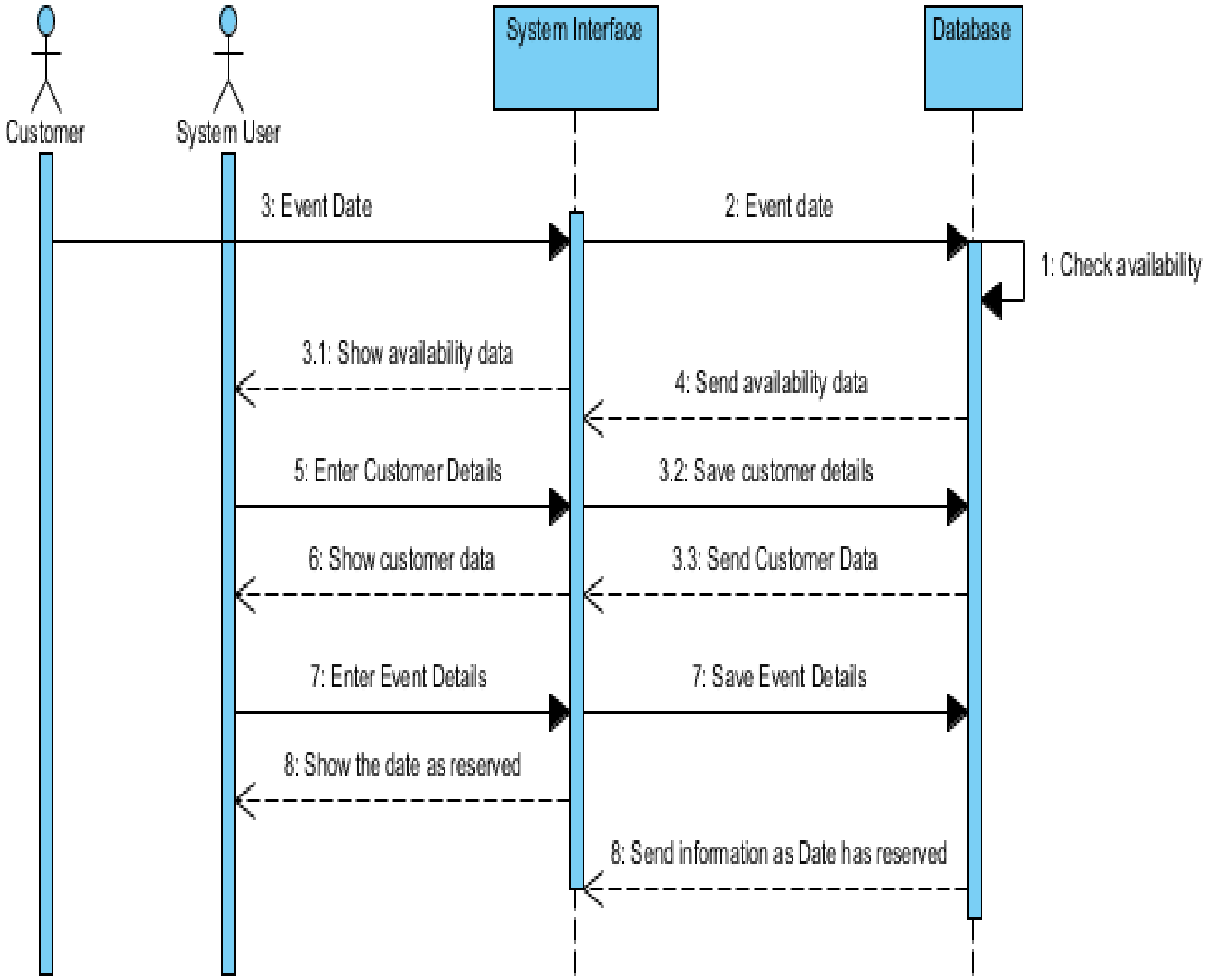


Figure B. 8 - Sequence diagram for Reserving an event date for a new customer

Appendix C – USER DOCUMENTATION

- Login form
  Open the installed web browser and type the URL http://localhost/EMS “Enter” button to access the system. You will get the following form.

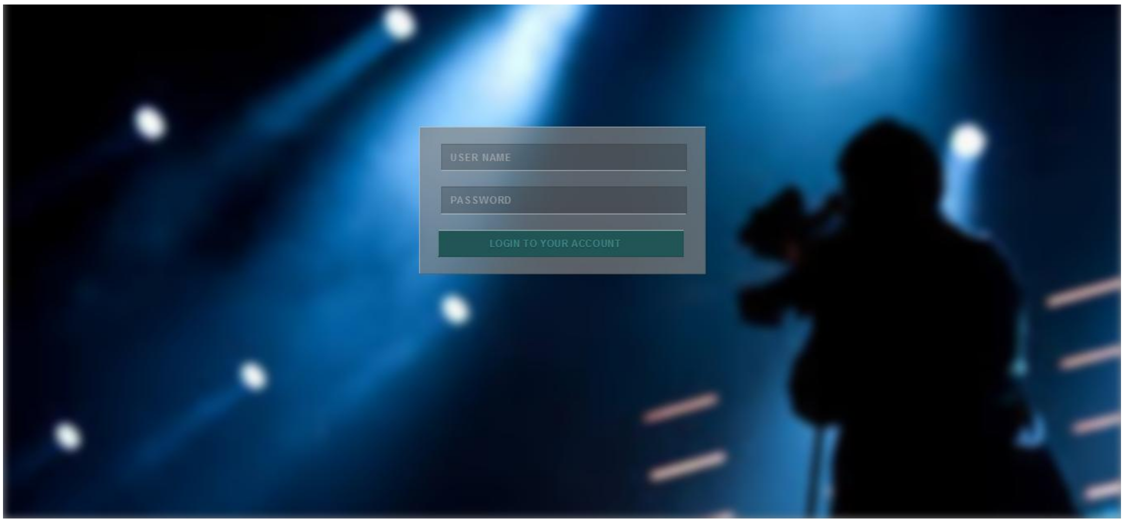


Figure C. 1- Login form

- You can access the system by entering correct username and password. In the main window many options are existing as follows.

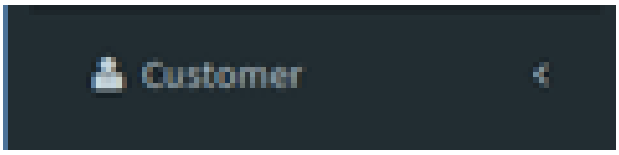


Figure C. 2-Customer menu

- Above option customer will give the following options.

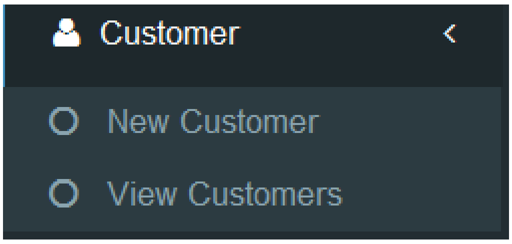


Figure C. 3-Customer menu options

- The Event option will give the options as follows

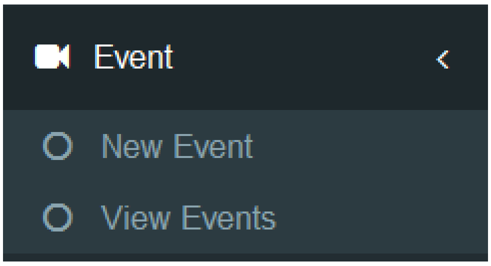


Figure C. 4-Event menu

- The Payment option will give the options as follows

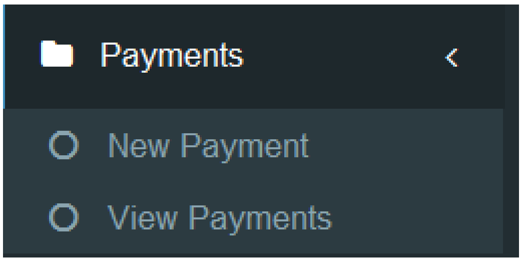


Figure C. 5-Package menu

- The Employees option will give the options as follows

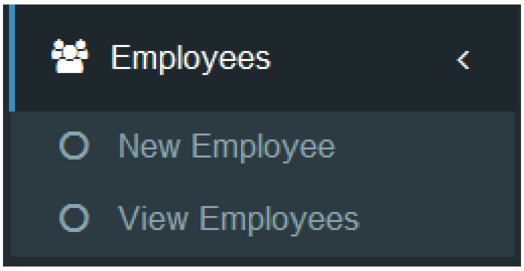


Figure C. 6-Employee menu

- The Packages option will give the options as follows

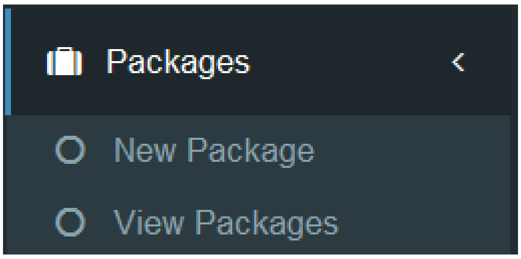


Figure C. 7-Package menu

- The Employees option will give the options as follows

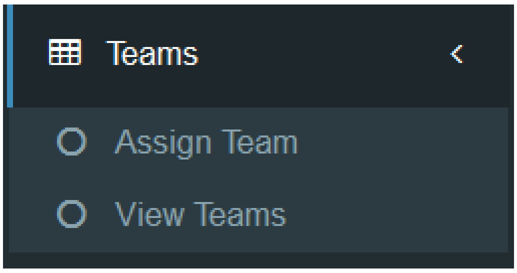


Figure C. 8-Team menu

- Calendar and Mail box options also exists.

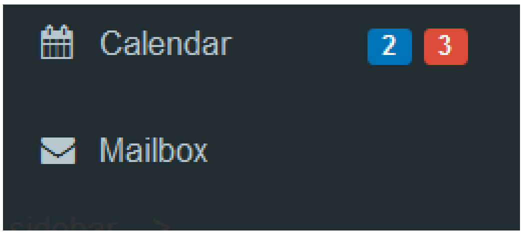
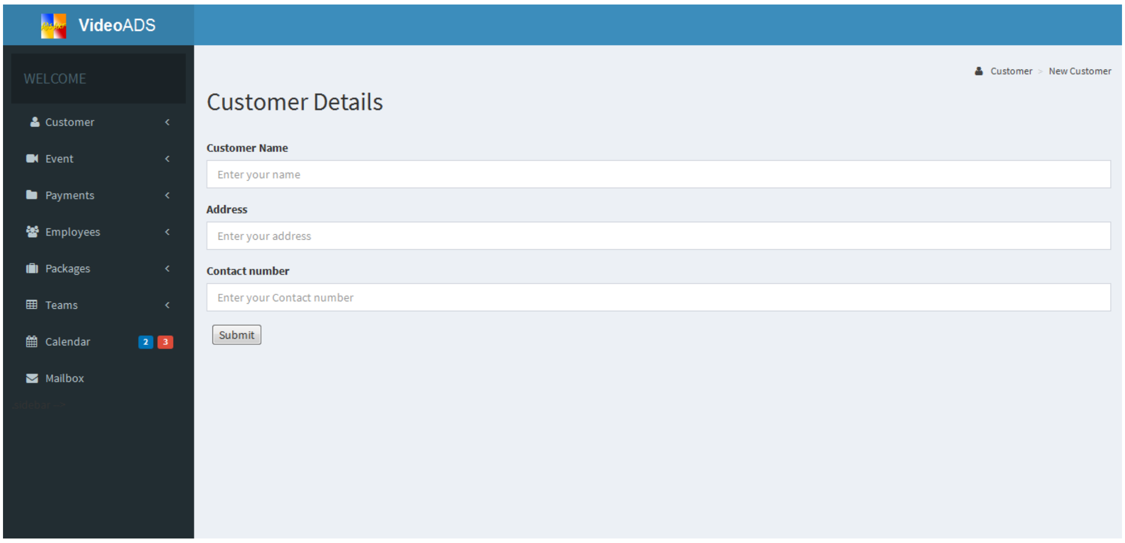


Figure C. 9-Calendar notifications

- New customer option will give the Customer Details form.



Customer Details

Customer Name

Enter your name

Address

Enter your address

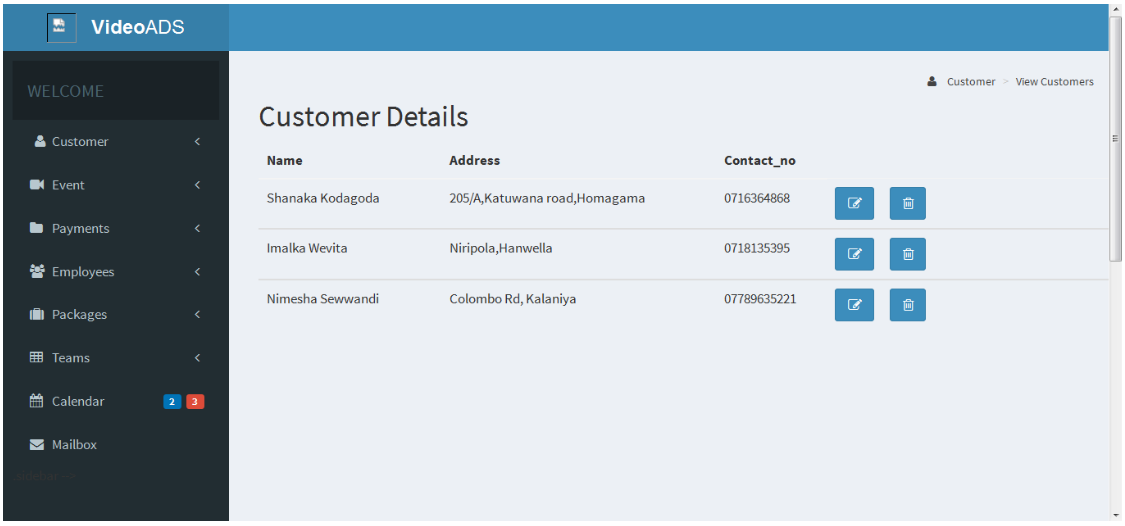
Contact number

Enter your Contact number

Submit

Figure C. 10- Customer Details form

- View customer option will give the Customer Details page.

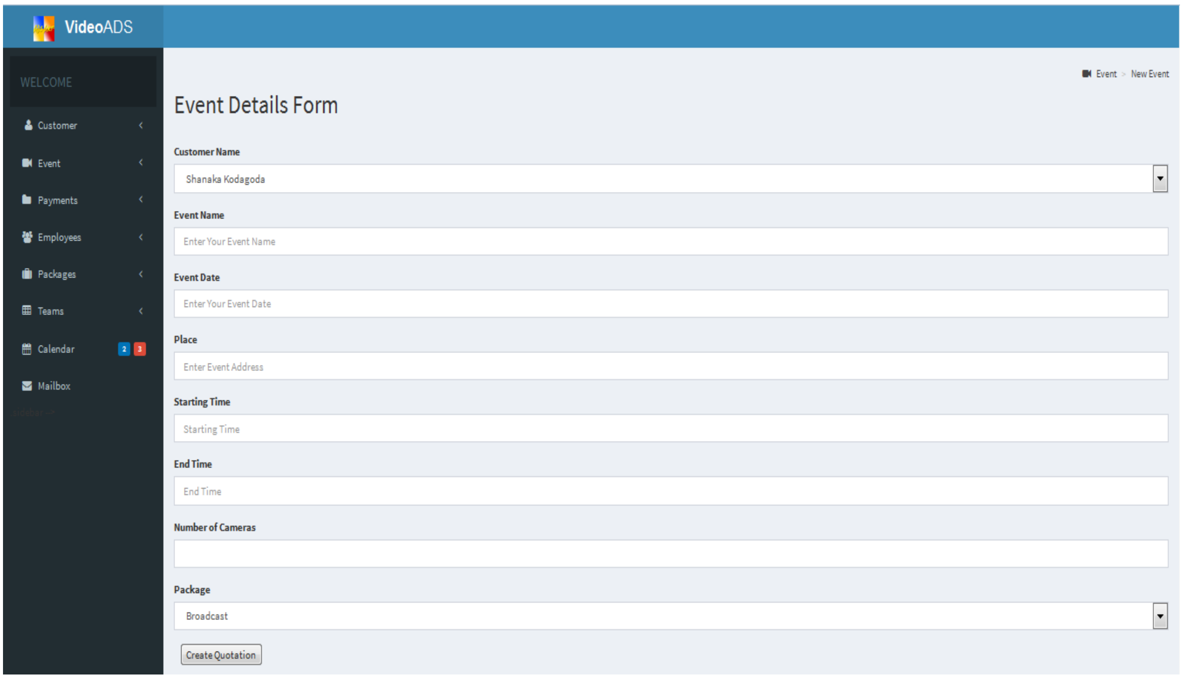


Customer Details

Name	Address	Contact_no
Shanaka Kodagoda	205/A,Katuwana road,Homagama	0716364868
Imalka Wevita	Niripola,Hanwella	0718135395
Nimesha Sewwandi	Colombo Rd, Kalaniya	07789635221

Figure C. 11- Customer Details page

- New Event option will give the Event Details form.

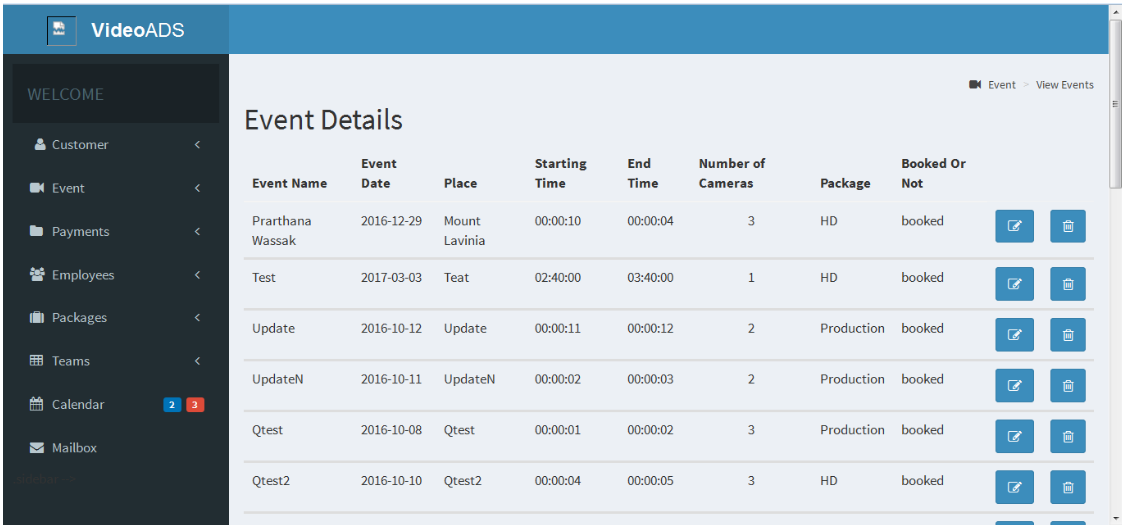


Event Details Form

Customer Name	Shanaka Kodagoda
Event Name	Enter Your Event Name
Event Date	Enter Your Event Date
Place	Enter Event Address
Starting Time	Starting Time
End Time	End Time
Number of Cameras	
Package	Broadcast

Figure C. 12- Event Details form

- View Events option will give the Event Details page.



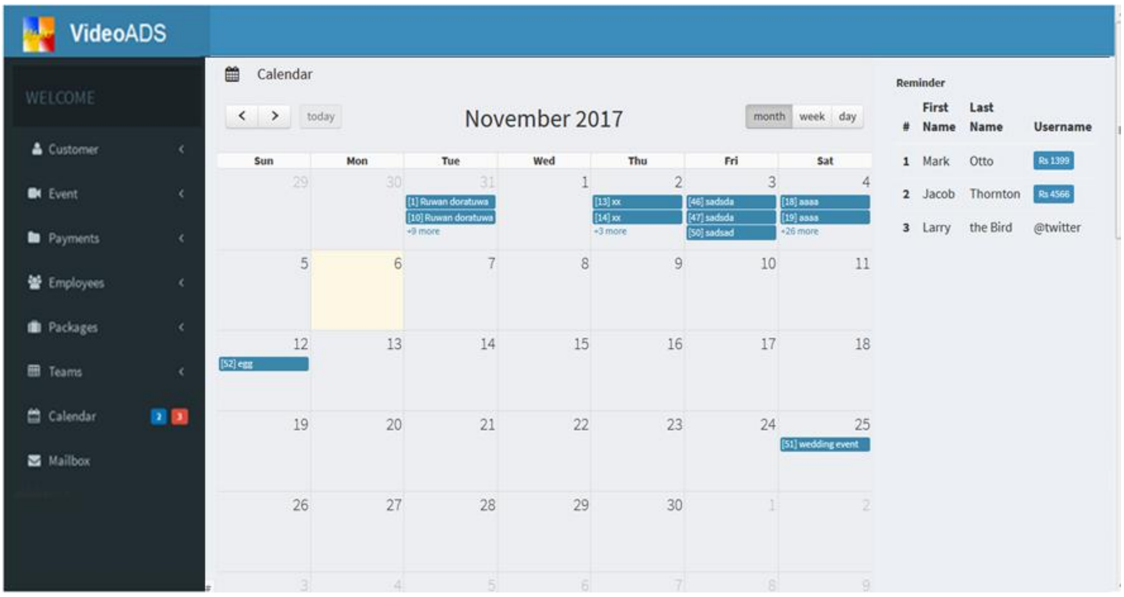
Event Details

Event Name	Event Date	Place	Starting Time	End Time	Number of Cameras	Package	Booked Or Not
Prarthana Wassak	2016-12-29	Mount Lavinia	00:00:10	00:00:04	3	HD	booked
Test	2017-03-03	Teat	02:40:00	03:40:00	1	HD	booked
Update	2016-10-12	Update	00:00:11	00:00:12	2	Production	booked
UpdateN	2016-10-11	UpdateN	00:00:02	00:00:03	2	Production	booked
Qtest	2016-10-08	Qtest	00:00:01	00:00:02	3	Production	booked
Qtest2	2016-10-10	Qtest2	00:00:04	00:00:05	3	HD	booked

Figure C. 13- Event Details page

Other options such as Package, Payment, Employee, Team also gives the pages like the above.

- Calendar option gives the following page.



November 2017

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	31 [1] Ruwan doratuwa [10] Ruwan doratuwa +9 more	1	2 [13] xx [14] xx +3 more	3 [46] sadsda [47] sadsda [50] sadsad	4 [18] aaaa [19] aaaa +26 more
5	6	7	8	9	10	11
12 [52] egg	13	14	15	16	17	18
19	20	21	22	23	24	25 [51] wedding event
26	27	28	29	30	1	2
3	4	5	6	7	8	9

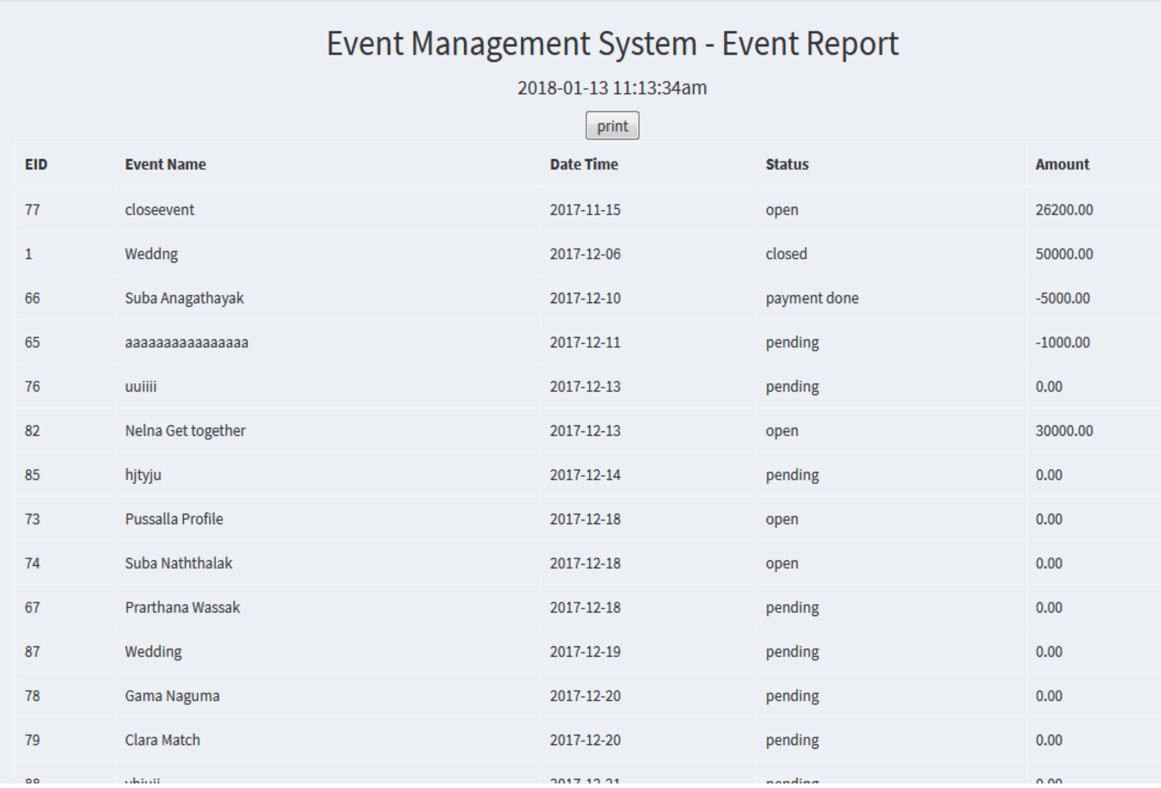
Reminder

#	First Name	Last Name	Username
1	Mark	Otto	Rs 1399
2	Jacob	Thornton	Rs 4566
3	Larry	the Bird	@twitter

Figure C. 14- Calendar

Appendix D – MANAGEMENT REPORTS

- Event Report displays all the events with their details for given period of time



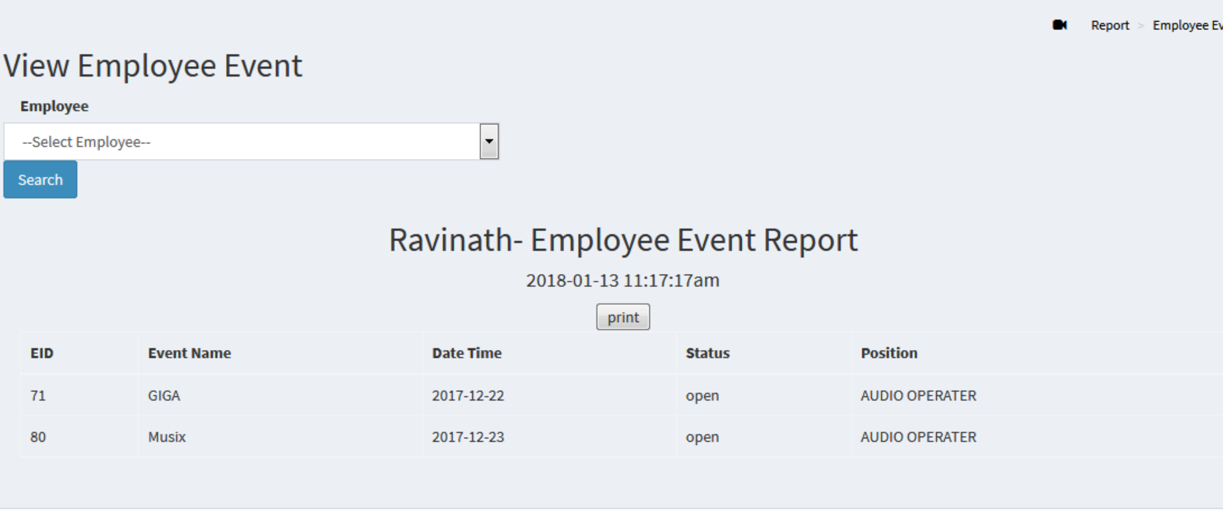
Event Management System - Event Report

2018-01-13 11:13:34am

EID	Event Name	Date Time	Status	Amount
77	closeevent	2017-11-15	open	26200.00
1	Weddng	2017-12-06	closed	50000.00
66	Suba Anagathayak	2017-12-10	payment done	-5000.00
65	aaaaaaaaaaaaaaaa	2017-12-11	pending	-1000.00
76	uuiiii	2017-12-13	pending	0.00
82	Nelna Get together	2017-12-13	open	30000.00
85	hjtyju	2017-12-14	pending	0.00
73	Pussalla Profile	2017-12-18	open	0.00
74	Suba Naththalak	2017-12-18	open	0.00
67	Prarthana Wassak	2017-12-18	pending	0.00
87	Wedding	2017-12-19	pending	0.00
78	Gama Naguma	2017-12-20	pending	0.00
79	Clara Match	2017-12-20	pending	0.00
88	vhjuii	2017-12-21	pending	0.00

Figure D. 1- Event Report

- Employee Report display all the events that particular employee is allocated for given period of time



View Employee Event

Employee: --Select Employee--

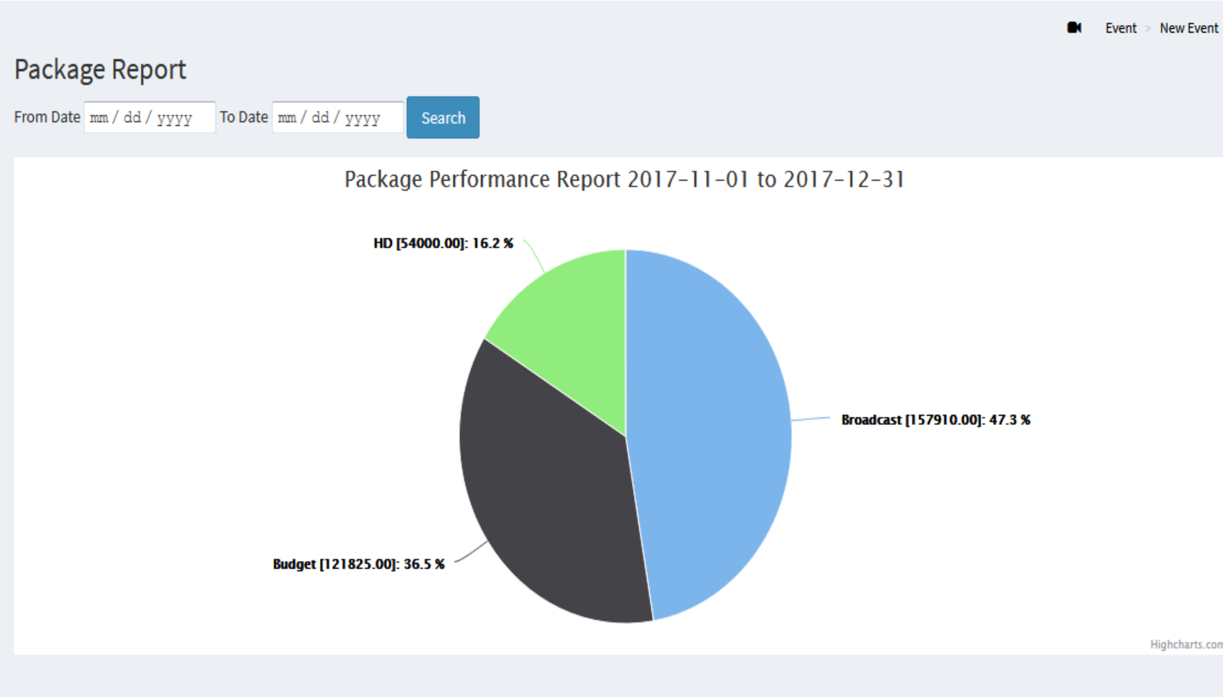
Ravinath- Employee Event Report

2018-01-13 11:17:17am

EID	Event Name	Date Time	Status	Position
71	GIGA	2017-12-22	open	AUDIO OPERATER
80	Musix	2017-12-23	open	AUDIO OPERATER

Figure D. 2- Employee Report

- Package Report display income as a percentage for every package for given period of time



Package Report

From Date mm / dd / yyyy To Date mm / dd / yyyy

Package Performance Report 2017-11-01 to 2017-12-31

Figure D. 3- Package Report

Appendix E – TEST RESULTS

- Test result for User Authentication**.**

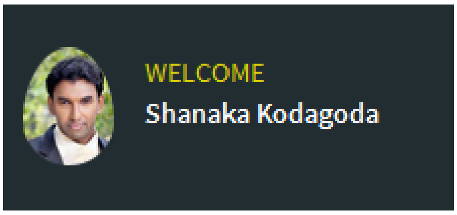
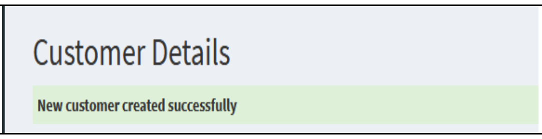
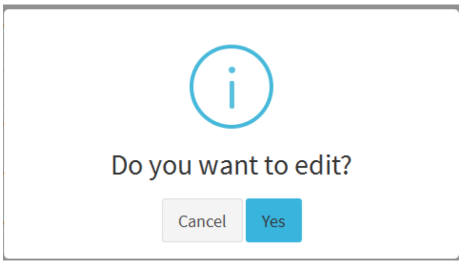
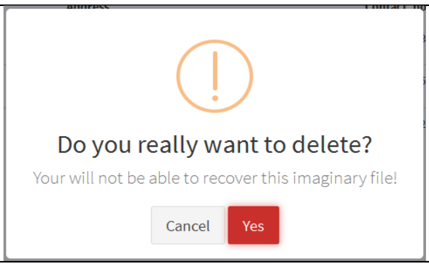
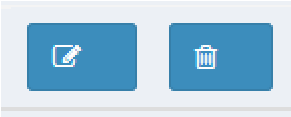
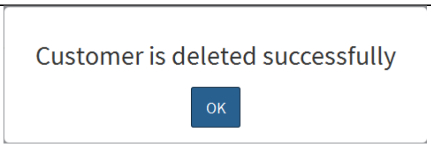
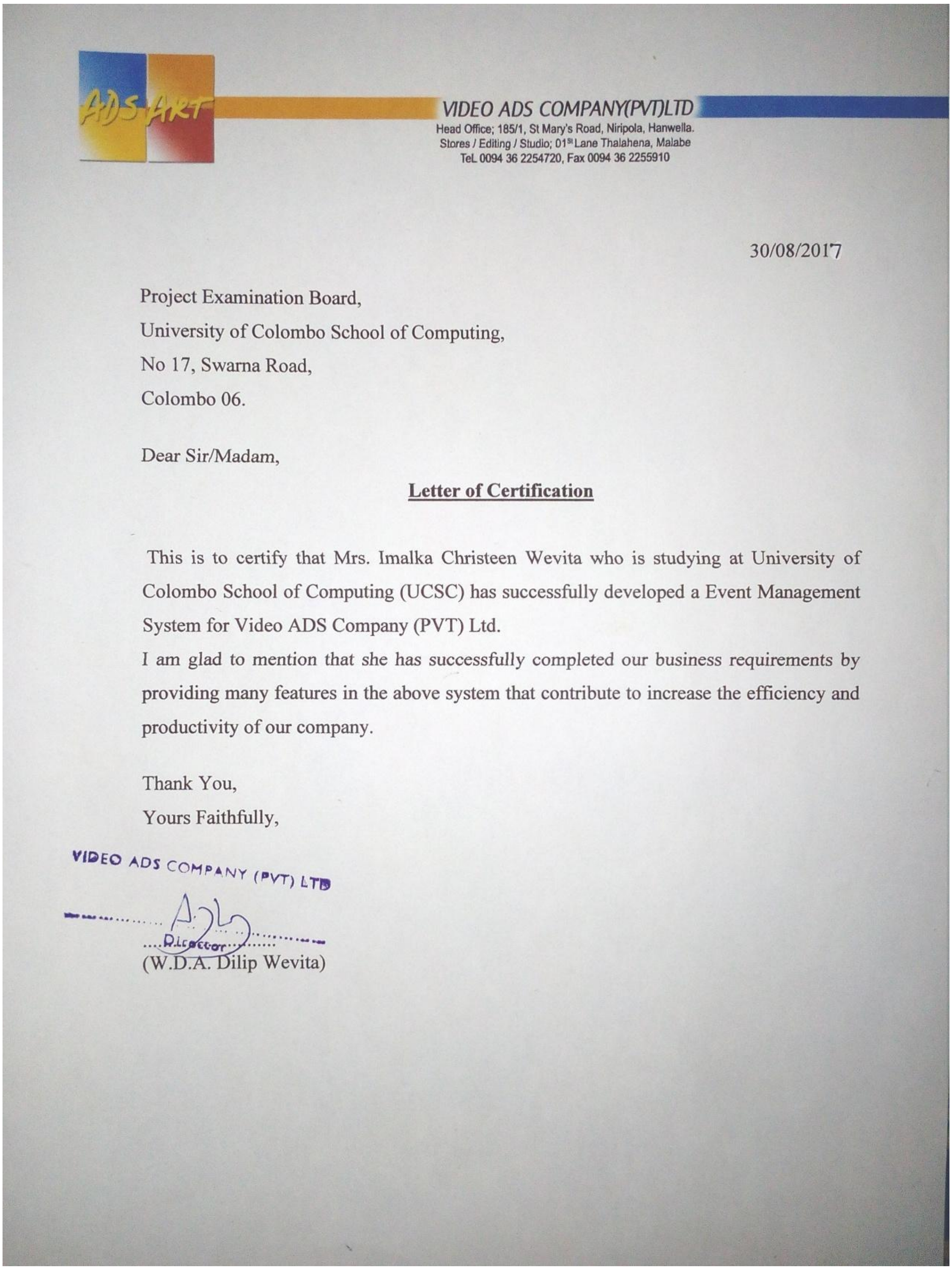
Test No	Step to Test	Expected Result	Status
1	Enter correct username and correct password Enable account status	Successfully Login into the System.  WELCOME Shanaka Kodagoda	✓
2	Enter a new customer	 Customer Details New customer created successfully	✓
3	When the edit button is clicked to edit any record such as customer, Event etc.	 Do you want to edit? 	✓
4	When the delete button is clicked to delete any record such as customer, Event etc.	 Do you really want to delete? Your will not be able to recover this imaginary file! 	✓
5	Each recode contains Edit and Delete buttons to facilitate Editing and deleting data.		✓
6	Deleting of a record completed	 Customer is deleted successfully 	✓

Table E. 1- Test result for User Authentication

Appendix F – CLIENT CERTIFICATE



ADS ART

VIDEO ADS COMPANY(PVT)LTD
Head Office; 185/1, St Mary's Road, Niripola, Hanwella.
Stores / Editing / Studio; 01st Lane Thalahena, Malabe
Tel. 0094 36 2254720, Fax 0094 36 2255910

30/08/2017

Project Examination Board,
University of Colombo School of Computing,
No 17, Swarna Road,
Colombo 06.

Dear Sir/Madam,

Letter of Certification

This is to certify that Mrs. Imalka Christeen Wevita who is studying at University of Colombo School of Computing (UCSC) has successfully developed a Event Management System for Video ADS Company (PVT) Ltd.

I am glad to mention that she has successfully completed our business requirements by providing many features in the above system that contribute to increase the efficiency and productivity of our company.

Thank You,
Yours Faithfully,

VIDEO ADS COMPANY (PVT) LTD
....
Director
(W.D.A. Dilip Wevita)

Figure F. 1- Client Certificate

Glossary

AJAX – Stands for Asynchronous JavaScript and XML. Combine collection of technologies. Running on Client side and helps to develop interactive web applications

Apache – Secure web server developed by Apache Software foundation.

CSS – Stands for Cascading Style Sheets. Use to apply styles for Markup languages such as HTML, XML.

HTML – Stands for Hyper Text Markup Language. Use to build Web pages.

Interface – Interconnect web system with the user.

JavaScript – Use to dynamic web applications. Develop by Netscape.

MySQL – One of most popular Database management system can handle big amount of data related to different types.

PHP – Stand for PHP Hypertext Preprocessor. Object oriented supported server side scripting language.

RUP – Stands for Rational Unified Process. Iterative software development methodology. Develop by Rational Software Co-operation.

WAMP – Open source Bundled software package. Include Apache, PHP and MYSQL.

JQuery – It is a JavaScript library. It was designed to simplify the client-side scripting of HTML.

JSON – Stands for Java Script Object Notation. Light weight data exchange format. Have similar characteristics of XML

SQL - Stands for Structured Query Language. Help to retrieve data base details.

MD5 - (Message Digest Algorithm) Use when converting data into unreadable format.

UML - Stands for Unified Modeling Language. Developed by Ivar Jacobson, James Rumbaugh, and Grady Booch at Rational Software. It is a modeling language.

Index