**FINAL PROJECT PROPOSAL**

**GIMS Event HUB**

****

**Submitted By**

|  |  |
| --- | --- |
| RAB Nawaz | 20-Arid-1719 |
| Tayyab Fiaz | 20-Arid-1741 |
| Khalil Ahmad | 20-Arid-1694 |

**Supervisor**

Sir Sami Ullah

C0-Ordinator(CS)

**2023**

**Gujrat Institute of Management Sciences**  
**PMAS-Arid Agriculture University, Rawalpindi**

**GIMS Event HUB**

Project submitted to

Department of Computer Science

Gujrat Institute of Management Sciences

**By**

|  |  |
| --- | --- |
| RAB Nawaz | 20-Arid-1719 |
| Tayyab Fiaz | 20-Arid-1741 |
| Khalil Ahmad | 20-Arid-1694 |

In partial fulfillment of the requirements for the degree of

(BSCS (Hons))

**Gujrat Institute of Management Sciences**  
**PMAS-Arid Agriculture University, Rawalpindi**

**Members’ Detail**

|  |  |
| --- | --- |
| **Project ID** | GIMS-BSCS-F202016 |

|  |  |  |  |
| --- | --- | --- | --- |
| **GIMS Event HUB** | | | |
| **Group Leader: Tayyab Fiaz**  **Group Members:**3 | | | |
| Name | Registration # | Email Address | Class |
| Rab Nawaz | 20-Arid-1719 | [rabnawaz20arid1719@gmail.com](mailto:membermailid@gmail.com) | BSCS |
| TayyabFiaz | 20-Arid-1741 | [tayyabfiaz20arid1741@gmail.com](mailto:membermailid@gmail.com) | BSCS |
| KhalilAhmad | 20-Arid-1694 | [khalilhmad20arid1694@gmail.com](mailto:membermailid@gmail.com) | BSCS |

|  |  |
| --- | --- |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Mr. Awais Ilyas Baig**  PMO, GIMS |  |
| **Dated: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | |

# Dedication

“Start with the name of Allah Who created man from sounding clay, from mud molded into shape” I Dedicate this project to Allah Almighty my creator been the source of my strength, faith and inspiration throughout this program and on his wings only have I soared. I would like to express my sincere gratitude to our project manager Mr. Sami Ullah for kindly offering his valuable advice and suggestions throughout. Without her continues guidance this work would not have been completed. He provided us the courage and faith for completing this project. I would like to thank our parents for their endless love, efforts, encouragements they did for us and to.our loved ones, for their prayers, understandings, devotions for keeping motivate us.

# Project Summary

|  |  |
| --- | --- |
| **Project Title** | GIMS Event HUB |
| **Project ID** | GIMS-BSCS-F202016 |
| **Organization** | Gujrat Institute of Management Sciences |
| **Objective** | The Objectives are:   * To Develop online Event Management System for GIMS. * To Save the Time Of Gims Faculty. * To Develop a Secure System for Stuent and Faculty. * To Develop a User Friendly System. * To Provide Online Tracking Of Upcoming Events. * To keep the Proper and Organized Record Of Events. * To Reducing problems occurred in manual system . * To Rating to the Events and FYP Projects . * Analysis of Events by the Budgets and its Rating . * To Develop Online Event Ticket Purchasing System . |
| **Undertaken By** | Rab Nawaz (20-Arid-1719), Tayyab Fiaz (20-Arid-1741), and Khalil Ahmad (20-Arid-1694) |
| **Supervised By** | Sir Sami Ullah |
| **Date started** | November 20, 2023 |
| **Date Completed** | June 20, 2024 |
| **Technologies Used** | PHP (Laravel) , HTML , CSS, Bootstrap , Java Script, JQuery  , MySQL, Xamp Server |
| **System Used** | Window 10 pro |

**Proofreading Certificate**

**This is to acknowledge that the project entitled**

**GIMS Event HUB [GIMS-BSCS-F202016]**

**has been proofread**

**By**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Sir Sammi Ullah**

Co-Ordinator ( CS )

|  |
| --- |
| **Dated: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

# Declaration

We hereby declare that we developed this project and this report entirely on the basis of our personal efforts made under the sincere guidance of our project supervisor. We further declare that, the titled project and all associated documents, reports are submitted as partial requirements for the degree of “BS (Hons.) in Computer Sciences”.

|  |  |  |
| --- | --- | --- |
| **Members’ Name** | **Registration #** | **Signature** |
| Rab Nawaz | 20-Arid-1719 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Tayyab Fiaz | 20-Arid-1741 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Khalil Ahmad | 20-Arid-1694 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**Dated:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Acknowledgement

Our deepest gratitude goes to Allah who has provided all that was needed to complete this project and all program for which it was undertaken. Throughout this entire study, He took care of everything that would have stopped us in our tracks and strengthened us even through my worst difficult times. We would also like to express our deepest appreciation to all those who provided us the possibility to complete this report.

**Certificate**

This is to certify that **Rab Nawaz**, **20-Arid-1719**, **Tayyab Fiaz**, **20-Arid-1741**, and **Khalil Ahmad**, **20-Arid-1694** have successfully completed the final project titled:“**GIMS Event HUB**”, accepted by the Department of Computer Science and find satisfactory for the requirement of:

**Gujrat Institute of Management Sciences**

**PMAS-Arid Agriculture University Rawalpindi**

**For Award of the Degree**

**Bachelor’s in Computer Science**

|  |  |  |
| --- | --- | --- |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Supervisor  Mr Sami Ullah  Co-Ordinator(CS), GIMS |  |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Examiner 1 |  |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Examiner 2 |  |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Examiner 3 |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  HOD/CS-IT  GIMS |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Director General  GIMS |

**Dated:**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This final year project aimed to developing a system to manage the GIMS University Events and activitoes.The system enables efficient management of parties by providing a platform to organizye and oversee different types of indoor and outdoor events, such as bonfires, Final Year Project Exhibition, farewells, milled shareef, and departmental events including FYP Project. Users can easily Review the previous Events, access information about upcoming events, facilitating effective planning and participation., the system also includes departmental activities within the university.

It empowers administrators to oversee and manage various departmental events, ensuring smooth coordination and execution. This functionality enables efficient scheduling or arranging, and tracking of departmental activities, enhancing the overall departments within the organized manner in university.

The project contributes to the efficient management and organization of university events, by providing a centralized platform for events management, the system aims to enhance communication, engagement, and coordination among students, faculty, and event organizers.

# Table of Contents

[Dedication iii](#_Toc498001466)

[Project Summary iv](#_Toc498001467)

[Acknowledgement vii](#_Toc498001468)

[Abstract ix](#_Toc498001469)

[Table of Contents x](#_Toc498001470)

[List of Tables xii](#_Toc498001471)

[List of Figures xv](#_Toc498001472)

[List of Abbreviations xv](#_Toc498001473)i

[Chapter 1 Introduction 2](#_Toc498001474)

[1.1 Problem Statement 2](#_Toc498001475)

[1.2 Significance 3](#_Toc498001476)

[1.3 Objectives 4](#_Toc498001477)

[1.4 Limitations and Restrictions: 4](#_Toc498001478)

[1.5 Overview 5](#_Toc498001479)

[1.6 System Architecture 7](#_Toc498001480)

[1.7 Software/Hardware Requirements 7](#_Toc498001481)

[1.8 Implementation Tools and Technology 8](#_Toc498001482)

[1.9 Implementation Plan 8](#_Toc498001483)

[1.9.1 Deliverable Items 8](#_Toc498001484)

[1.9.2 Milestone Chart 9](#_Toc498001485)

[Chapter 2 Requirement Analysis 1](#_Toc498001486)0

[2.1 Functional Requirements: 1](#_Toc498001487)1

[2.2 Non Functional Requirement: 1](#_Toc498001488)3

[2.3 Use Cases 1](#_Toc498001489)3

2.3.1 Use Case for Admin Login...............................................................14

2.3.2 Use Case for Admin Add banner........................................................15

2.3.3 Use Case for Admin Accept Event.....................................................16

2.3.4 Use Case for Admin Reject Event.......................................................17

2.3.5 Use Case for Admin Update Event....................................................18

2.3.6 Use Case for Admin Check tickets....................................................19

2.3.7 Use Case for Admin Create Banner For Landing Page.....................20

2.3.8 Use Case for Event Manger Sign UP.................................................21

2.3.9 Use Case for Event Manger Log in....................................................22

2.3.10 Use Case for Event Manger Add Event.............................................23

2.3.11 Use Case for Event Manger Delete Event.........................................24

2.3.12 Use Case for Event Manger Event Update........................................25

2.3.13 Use Case for Event Manger Generate Ticket.....................................26

2.3.14 Use Case for Event Manger Post Event Data....................................27

2.3.15 Use Case for Event Manger ADD Event Expense………................28

2.3.16 Use Case for Student Sign UP..........................................................29

2.3.17 Use Case for Student Login..............................................................30

2.3.18 Use Case for Student View Upcoming Event...................................31

2.3.19 Use Case for Student View Prevous Evet Data Event......................32

2.3.20 Use Case for Student Purchase Event Ticket...................................33

2.3.21 Use Case for Student Check Ticket Generate.................................34

2.3.22 Use Case for Student Can Rate Events.............................................35

2.3.23 Use Case for Student .Upload FYP Project......................................36

[Chapter 3 DESIGN 37](#_Toc498001490)

[3.1 UML Diagrams 38](#_Toc498001491)

[3.2 Use-Case Diagram 3](#_Toc498001492)8

[3.2.2 Admin.....................................................................................................38](#_Toc498001494)

[3.2.2 Event Manger](#_Toc498001494) ........................................................................................39

[3.2.2 Student](#_Toc498001494)....................................................................................................40

[3.3 Class Diagram](#_Toc498001495) 41

[3.4 Sequence Diagram 4](#_Toc498001496)2

[3.4.1 Event Manager login 42](#_Toc498001497)

[3.4.2 Event Manger Create Event 43](#_Toc498001497)

[3.4.3 Event Manager Delete Event 44](#_Toc498001497)

[3.4.4 Event Manager Posting Event Detail 45](#_Toc498001497)

[3.4.5 Event Manager Expense Upload 46](#_Toc498001497)

[3.4.6 Event Manager Read Event 47](#_Toc498001497)

[3.4.7 Event Manager Ticket Check 47](#_Toc498001497)

[3.4.8 Event Manager Update Event 48](#_Toc498001497)

[3.4.9 Admin Accept or Reject Evnt 49](#_Toc498001497)

[3.4.10 Admin Manage Bannner Images 50](#_Toc498001497)

[3.4.11 Admin View Generated Tickets 51](#_Toc498001497)

[3.4.12 Admin View Event Expense 52](#_Toc498001497)

[3.4.13 Student Login 53](#_Toc498001497)

[3.4.14 Student Log out 54](#_Toc498001497)

[3.4.15 Student Filter UPcoming Events 55](#_Toc498001497)

[3.4.16 Student Rate Event 56](#_Toc498001497)

[3.4.17 Student Purchase Event 57](#_Toc498001497)

[3.4.18 Student Check Generated Ticket 58](#_Toc498001497)

[3.4.19 Student View FYP Project 59](#_Toc498001497)

[3.5 ERD 6](#_Toc498001498)2

[Chapter 4 Testing 23](#_Toc498001499)

[4.1 Sign Up Admin User & Worker 23](#_Toc498001500)

[Chapter 5 User Manual 25](#_Toc498001501)

[5.1 Home Page 25](#_Toc498001502)

[References: 26](#_Toc498001503)

# List of Tables

[Table 2.1: Use-Case 1 14](#_Toc498001506)

Table 2.3.1 Use Case for Admin Login...............................................................14

Table 2.3.2 Use Case for Admin Add banner........................................................15

Table2.3.3 Use Case for Admin Accept Event......................................................16

Table2.3.4 Use Case for Admin Reject Event.......................................................17

Table2.3.5 Use Case for Admin Update Event......................................................18

Table2.3.6 Use Case for Admin Check tickets.....................................................19

Table2.3.7 Use Case for Admin Create Banner For Landing Page......................20

Table 2.3.8 Use Case for Event Manger Sign UP..................................................21

Table 2.3.9 Use Case for Event Manger Log in....................................................22

Table2.3.10 Use Case for Event Manger Add Event............................................23

Table 2.3.11 Use Case for Event Manger Delete Event.......................................24

Table 2.3.12 Use Case for Event Manger Event Update......................................25

Table 2.3.13 Use Case for Event Manger Generate Ticket..................................26

Table 2.3.14 Use Case for Event Manger Post Event Data...................................27

Table 2.3.15 Use Case for Event Manger ADD Event Expense…........…….......28

Table2.3.16 Use Case for Student Sign UP..........................................................29

Table 2.3.17 Use Case for Student Login.............................................................30

Table 2.3.18 Use Case for Student View Upcoming Event.................................31

Table 2.3.19 Use Case for Student View Prevous Evet Data Event....................32

Table 2.3.20 Use Case for Student Purchase Event Ticket..................................33

Table 2.3.21 Use Case for Student Check Ticket Generate..................................34

Table 2.3.22 Use Case for Student Can Rate Events.............................................35

Table 2.3.23 Use Case for Student .Upload FYP Project......................................36

[Table 4.1: Sign up Test case 23](#_Toc498001507)

# List of Figures

[Figure 1.2: Gantt chart for milestones 9](#_Toc498001557)

[Figure 3.2: Use-Case Diagram for Admin 38](#_Toc498001559)

[Figure 3.2: Use-Case Diagram for Event Mamnger 3](#_Toc498001559)9

[Figure 3.2: Use-Case Diagram for Sutdent 40](#_Toc498001559)

[Figure 3.3 Class Diagram 41](#_Toc498001560)

[Figure 3.4: Sequence Diagram for User Authentication 20](#_Toc498001561)

[3.4.1 Event Manager login 42](#_Toc498001497)

[3.4.2 Event Manger Create Event 43](#_Toc498001497)

[3.4.3 Event Manager Delete Event 44](#_Toc498001497)

[3.4.4 Event Manager Posting Event Detail 45](#_Toc498001497)

[3.4.5 Event Manager Expense Upload 46](#_Toc498001497)

[3.4.6 Event Manager Read Event 47](#_Toc498001497)

[3.4.7 Event Manager Ticket Check 47](#_Toc498001497)

[3.4.8 Event Manager Update Event 48](#_Toc498001497)

[3.4.9 Admin Accept or Reject Evnt 49](#_Toc498001497)

[3.4.10 Admin Manage Bannner Images 50](#_Toc498001497)

[3.4.11 Admin View Generated Tickets 51](#_Toc498001497)

[3.4.12 Admin View Event Expense 52](#_Toc498001497)

[3.4.13 Student Login 53](#_Toc498001497)

[3.4.14 Student Log out 54](#_Toc498001497)

[3.4.15 Student Filter UPcoming Events 55](#_Toc498001497)

[3.4.16 Student Rate Event 56](#_Toc498001497)

[3.4.17 Student Purchase Event 57](#_Toc498001497)

[3.4.18 Student Check Generated Ticket 58](#_Toc498001497)

[3.4.19 Student View FYP Project 59](#_Toc498001497)

# Listof Abbreviations

|  |  |  |
| --- | --- | --- |
| GIMS |  | Gujrat Institute Management Of Sciences |
| HTML |  | Hypertext Markup Language |
| CSS |  | Cascading Style Sheets |
| PHP |  | Hypertext Preprocessor |
| HND |  | Human Nutrition & Dietetics |
| MySql |  | Structured Query Language |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Chapter 1

Introduction

# 

# Introduction

This Web based system aims to the management of parties and departmental events within the university environment. By providing a centralized platform, it offers a range of capabilities to enhance event planning, organization, and execution. The project focuses on two key aspects: party management and departmental event control. The system empowers the event organizers, administrators, and participants with a user-friendly interface to effectively manage various events. In terms of party management, the system offers a set of essential features. It enables event organizers to easily add new parties, including details such as event names, dates, and descriptions. Additionally, the system facilitates online ticket registration, allowing participants to purchase tickets through a secure online platform. Venue details are also incorporated, providing information on event locations and directions. The system further provides functionality for estimating expenses associated with each party, aiding in budget planning and financial management.

Furthermore, a ticket checker feature is available to validate tickets and ensure a smooth entry process for attendees. Moving on to departmental event management, the system provides comprehensive control over university departmental activities. It allows administrators to add various departmental events like FYP or Programming Competition. The system offers the flexibility to set up tournaments, allowing participants and spectators to stay informed. Finally, at the conclusion of the tournament or event, the system generates a winner, based on accumulated results, provided by Examiner a clear indication of the overall champion

## Problem

The currently university’s event management don’t have any centralized system to manage the Events and Activities. Event organizers face challenges in informing students about upcoming Events. Students face problem in getting the picture of Events as well as in purchasing Tickets of Events .:

## Significance

. Our project is designed to manage GIMS Events and its Departmental functions that facilitate the faculty and Society of GIMS to organize the Events in the proper ways it also empowers the student by giving it the authority of the rating the events and final year projects. Our project will also store the Information like the picture and videos about the previous events and admin will be able to the view the percentage of events successful by watching the budget of the event along with its rating by the student. It will carry significant role for enhancing the overall experience of university events successful that will brings the GIMS management and its student harmonious

A FYP Project has been supervised by Sir Bilal Tariq Butt named as GIMS Event Management System that provides only notification about the upcoming events and stores the pictures and videos about events

## Objectives

This system aims to reduce the Problem During Organizing any Event. And aim to Sure Updated to Universities events and Secure record of Previous Student.

The objectives are:

* To Save the Time Of Gims Faculty.
* To Develop a Secure System for Stuent and Faculty.
* To Develop a User Friendly System.
* To Provide Online Tracking Of Upcoming Events.
* To keep the Proper and Organized Record Of Events.
* To Reducing problems occurred in manual system .
* To Rating to the Events and FYP Projects .
* Analysis of Events by the Budgets and its Rating .
* To Develop Online Event Ticket Purchasing System

## Limitations and Restrictions:

* User Require Internet Connection and Browser to visit site
* User Required must have a Electronic device (Mobile phone, Laptop, Desktop.etc )
* Admin has to login with authentic Username or Email to manage services.
* Event Manger can’t organize the event without the permission of Admin.
* Only Student can purchase the Event ticket with their Student ID.

## Overview

|  |
| --- |
| **Project Goal:**  Project Goal: This Website is for creating event along with their Tickets to facilitate the university faculty to organize any an upcoming event effectively. University Faculty will be easily able to inform Students for upcoming events along with updation of location and time .   * To introduce the system to overcome the problems of Event management * Organize events in easy and well-mannered way * Updating Students with Event Location and date of up coming events.   Provide easy way for outsiders to access all information about university. |
| **Type of project:** 📺R&D 🗹Development |
| **Project Success Criteria:**   * Event Manger will be able to upload the information and data of the currently held Events * Event Manger will upload the data of Events after the permission of Admins. * This digital system will allows the Stuents to be Updated for real-time updates and notifications of upcoming events. * The system will stores records of previous events and FYP projects. * Student will have the ability to give rating events. |
| Risks of the Project:   |  |  |  |  | | --- | --- | --- | --- | | **(Please mark 🗹 where applicable)** | **Low** | **Medium** | **High** | | Technical risk |  | 🗹 |  | | Timing risk |  |  | 🗹 | | Budget risk | 🗹 |  |  | |
| **Target End users:**   * University Faculty * Students * Visitor |
| **Development Technology/ Languages:**   * PHP * BOOTSTRAP(CSS, JavaScript, J Query ) * MYSQL * XAMPP Server * LARAVEL * HTML * JAVA Script * CSS |
| **Platform:**  🗹Web based 📺Distributed 📺Setup Configurations  📺Desktop based 📺Android 📺iOS  📺Other\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| **Project Supervisor:** Sami Ullah |

## System Architecture

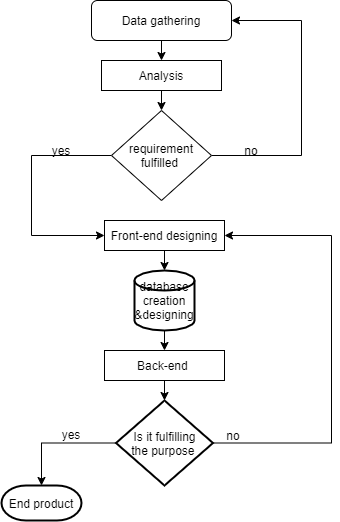


Figure 1.1: Model-View-Controller

## Software/Hardware Requirements

* i3 Processor Based Computer or higher
* Memory: 1 GB
* Hard Drive: 5 GB+
* Monitor
* Internet Connection

.

## 

## Implementation Tools and Technology

* Laravel Framework
* PHP
* Xampp
* Html
* CSS
* JavaScript
* Bootstrap

## Implementation Plan

### Deliverable Items

* Project Proposal
* Software Requirement Specification (SRS)
* Project Presentation and Final Documentation
* Executable Files
* CD
* User manual

### 

### Milestone Chart

**Gantt chart:**

|  |
| --- |
| C:\Users\Rabnawaz\Downloads\WhatsApp Image 2023-09-27 at 22.38.14.jpegWhatsApp Image 2023-09-27 at 22.38.14 |

Figure 1.2: Gantt chart for milestones

Chapter 2

Requirement Analysis

# 

# Requirement Analysis

## Functional Requirements

**For Student**

* **Sign up:**
* Manger can be sign up .
* **Login:**
* Studentcan be logged in by their email and password.
* **View Events:**
* Student can view all publish events / FYP projects.
* **Star rating:**
* Student can give star rating to Event.
* **Purchase Ticket:**
* Student can Purchase the ticket before the Deadline of the particular Event
* **Mange Fyp Project:**
* Student can mange own Fyp Project .
* **View Events History:**
* Studnent view events history that Evnts he Participated
* **Mange Profile:**
* Student can update manage profile .
* **Logout:**
* Student can update own profile .

**For Manager**

* **Sign up:**
* Manger can be sign up .
* **Login:**
* Manger can logged into by their Email and Password..
* **Manage Events :**
* Manger can create view and update delete Events.
* When Manger create event it send approval Request to admin.
* **Event Feedback:**
* Manger can view feedback of Event in rating.
* **Manage Profile:**
* Manger can manage own profile .
* **Event Expense :**
* Mange event expense.
* **Logout:**
* Manger can be logged out .
* **Ticket Managing:**
* Manger will upload design and itself ticket.
* Manager can see how many ticket sold on the base of event.

**For Admin**

* **Login:**
* Admin can be logged into by their Email and Password.
* **Manage Event:**
* Admin can view ,update accept or reject event purposed by the event manger.
* Admin can view all the event list.
* **Mange Mangers:**
* Admin can create view ,update ,delete, block and approve Manger’s request.
* **Manges Student:**
* Admin can create ,view, update, delete and block Student .
* **Manage profile:**
* Admin can manage own profile .
* **Event Feedbacks:**
* Admin can view Event Feedbacks .
* **Log out :**
* Admin can logout .
* **Ticket :**

Admin can see all ticket generated bases on the event.

## 

## Non Functional Requirement

All non-functional requirements of proposed system are as followed:

* Application availability will be 24/7 in a year. It means user can use this platform at any time anywhere.
* The system must be responsive and provide real-time updates to users.
* Response times for user interactions, such as event registration and match scheduling, should be within acceptable limits.
* User data, including personal information and payment details, must be securely stored and transmitted.
* Access controls and permissions should be robust to prevent unauthorized access to sensitive data and system functionalities.
* The system should have high availability to ensure users can access it when needed.
* Backup and disaster recovery procedures must be in place to minimize downtime and data loss in case of unexpected events.
* Continuous monitoring and automated error reporting should help identify and address issues promptly.

## 

## Use Cases

Table 2.3.1: Sign Up

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 1 | | **Use case Name :** | Sign Up | | **Level:** | System/User | | **Primary Actor :** | Admin/Event Manger/ Student | | **Goal :** | Sign Up Account | | **Pre-condition :** | Email Require | | **Post-condition :** | User Successfully sign up | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User reach sign up page 2. Fills all required fields. 3. Clicks on sign up button 4. user will be on Dashboard. 5. System verifies info and account created |
| **Alternative Path :**   1. If filled fields not correct. 2. System will generate invalid error. |

Table 2.3.2: Login

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | **2** | | **Use case Name :** | Login | | **Level:** | System/User | | **Primary Actor :** | Admin/Event Manger/ Student | | **Goal :** | Login Account | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. Fills all required fields. 2. Clicks on login button 3. Admin will be on Dashboard. 4. System verifies info 5. Admin will be on Dashboard. |
| **Alternative Path :**   1. If filled fields not correct. 2. System will generate invalid error. |

Table 2.3.3: Log out

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | **3** | | **Use case Name :** | Log out | | **Level:** | System/User | | **Primary Actor :** | Admin/Event Manger/ Student | | **Goal :** | Login Account | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. Clicks on log out button 2. Session will be destroy. 3. System Redirect to login |
| **Alternative Path :** |

Table 2.3.4: Manage profile

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | **4** | | **Use case Name :** | Manage profile | | **Level:** | User | | **Primary Actor :** | Admin/Event Manger/ Student | | **Goal :** | Manage profile | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on profile page. 2. Sytem redirect on profile window 3. User can make changes 4. Click on update profile button 5. Sytem verifies info and updata profile data. |
| **Alternative Path :**   1. Email or password isincorrect. 2. System cannot find user |

Table 2.3.5: view event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 5 | | **Use case Name :** | View Event | | **Level:** | Sysytem | | **Primary Actor :** | Event Manager/admin/ | | **Goal :** | Create Event | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User login account. 2. System will redirect to dashboard. 3. User click on event page 4. Sytem redirect to event page. |
| **Alternative Path :**   1. If user not login correctly 2. System fails to redirect event page. |

Table 2.3.6: Manager add event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 6 | | **Use case Name :** | Add Event | | **Level:** | Sysytem/Event Manager | | **Primary Actor :** | Event Manger | | **Goal :** | Create Event | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User login to account 2. System redirect user to dashboard 3. User click on add event page. 4. User click new event button. 5. Fill required fields. 6. User click on add event button 7. Sytem verifies info and store event |
| **Alternative Path :**   1. If fields not correct format 2. Sytem issue error for correct fields. |

Table 2.3.7: Manager delete event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 7 | | **Use case Name :** | Delete Event | | **Level:** | Sysytem/Event Manager | | **Primary Actor :** | Event Manger | | **Goal :** | Delete Event | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User login to account 2. System redirect user to dashboard 3. User click on event page. 4. User click delete button. 5. System will delete specific event. |
| **Alternative Path :** |

Table 2.3.8: Manager update event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 8 | | **Use case Name :** | update Event | | **Level:** | Sysytem/Event Manager | | **Primary Actor :** | Event Manger | | **Goal :** | update Event | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on event page. 2. User click update event button. 3. Fill required fields. 4. User click on update button 5. Sytem verifies info and store event. |
| **Alternative Path :**   1. If fields not correct format 2. Sytem issue error for correct fields. |

Table 2.3.9: Manager add expence

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 9 | | **Use case Name :** | Add expence | | **Level:** | Sysytem/Event Manager | | **Primary Actor :** | Event Manger | | **Goal :** | Add expence | | **Pre-condition :** | Must have account | | **Post-condition :** | Expence added | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on add expence page. 2. User click new event expence button. 3. Fill required fields. 4. User click on add new expence button 5. Sytem verifies info and store event |
| **Alternative Path :**   1. If fields not correct format 2. Sytem issue error for correct fields. |

Table 2.3.10: Manager update expence

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 10 | | **Use case Name :** | Update Expence | | **Level:** | Sysytem/Event Manager | | **Primary Actor :** | Event Manger | | **Goal :** | update Event | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on expence page. 2. User click update event expece button. 3. Fill required fields. 4. User click on update expence button 5. Sytem verifies info and store event. |
| **Alternative Path :**   1. If fields not correct format 2. Sytem issue error for correct fields. |

Table 2.3.11: Manager add ticket

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 11 | | **Use case Name :** | Add ticket | | **Level:** | Sysytem/Event Manager | | **Primary Actor :** | Event Manger | | **Goal :** | Add Ticket | | **Pre-condition :** | Must have account | | **Post-condition :** | Successfully Login | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on add ticket page. 2. User click new event icket button. 3. Fill required fields. 4. User click on add new ticket button 5. Sytem verifies info and store ticket |
| **Alternative Path :**   1. If fields not correct format 2. Sytem issue error for correct fields. |

Table 2.3.12: Admin Accept event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 12 | | **Use case Name :** | Accept event | | **Level:** | Sysytem/admin | | **Primary Actor :** | Admin | | **Goal :** | Accept Event | | **Pre-condition :** | Must have login to account | | **Post-condition :** | Admin access dashboard | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on Event page. 2. User click budget report of event. 3. System download budget report. 4. After reading the budget admin click on accept event button. 5. Sytem issue Success error. |
| **Alternative Path :** |

Table 2.3.13: Admin Reject event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 13 | | **Use case Name :** | Reject event | | **Level:** | Sysytem/admin | | **Primary Actor :** | Admin | | **Goal :** | Reject Event | | **Pre-condition :** | Must have login to account | | **Post-condition :** | Admin access dashboard | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on Event page. 2. User click budget report of event. 3. System download budget report. 4. After reading the budget admin click on reject event button. 5. Sytem issue Success error. |
| **Alternative Path :** |

Table 2.3.14: Admin Manage Managers

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | **14** | | **Use case Name :** | Aprove Manger | | **Level:** | Sysytem/admin | | **Primary Actor :** | Admin | | **Goal :** | Apreove Manager | | **Pre-condition :** | Must have login to account | | **Post-condition :** | Admin access dashboard | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on Managers page. 2. System will show all event manager details. 3. Admin can edit prove or reject. |
| **Alternative Path :** |

Table 2.3.15 Admin see feedback

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | **15** | | **Use case Name :** | Admin See feed back | | **Level:** | Sysytem/admin | | **Primary Actor :** | Admin | | **Goal :** | Admin will see feed back | | **Pre-condition :** | Must have login to account | | **Post-condition :** | Admin access dashboard | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on Feed back page. 2. System will redirect user to feedback page. 3. System automatically give there updated event. 4. User can fiter event to see feedback 5. System will show feedback on event filters |
| **Alternative Path :**   1. If user enter invalid event name 2. System will show error message |

Table 2.3.16: Admin see All tickets

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 16 | | **Use case Name :** | Admin See all tickets | | **Level:** | Sysytem/admin | | **Primary Actor :** | Admin | | **Goal :** | Admin will see all tickets generated | | **Pre-condition :** | Must have login to account | | **Post-condition :** | Admin access dashboard | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User click on Tickets page. 2. System will redirect user to tickets page. 3. System automatically giv details of updated event. 4. User can fiter event to see tickets 5. System will show all tickets for event filters |
| **Alternative Path :**   1. If user enter invalid event name 2. System will show error message |

Table 2.3.17: Student View upcoming event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 17 | | **Use case Name :** | Student view upcoming event | | **Level:** | Sysytem/Student | | **Primary Actor :** | Student | | **Goal :** | See upcomign Events | | **Pre-condition :** | User Having access to Web page. | | **Post-condition :** | User See events | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User Access web home page. 2. System will show all upcoming event. 3. User can filter the event according to date and time. 4. System will show data according to filte. |
| **Alternative Path :** |

Table 2.3.18: Student View past event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 18 | | **Use case Name :** | Student view past event | | **Level:** | Sysytem/Student | | **Primary Actor :** | Student | | **Goal :** | See past Events | | **Pre-condition :** | User Having access to Web page. | | **Post-condition :** | User See events | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User Access web home page. 2. System will show all details of event. 3. Admin click on event details. 4. System redirect the system to past event details. |
| **Alternative Path :** |

Table 2.3.19: Student rate event

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 19 | | **Use case Name :** | Student rate event | | **Level:** | Sysytem/Student | | **Primary Actor :** | Student | | **Goal :** | Rate Events | | **Pre-condition :** | User Having access to Web page. | | **Post-condition :** | User Successfully rate event | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User Access web home page. 2. System will show all details of event. 3. Admin click on event details. 4. System redirect the system to past event details. 5. User click on rate event page. 6. User fill required fields. 7. System verifies information and store rating 8. User get success message. |
| **Alternative Path :** |

Table 2.3.20: Student Purchase event ticket

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 20 | | **Use case Name :** | Student rate event | | **Level:** | Sysytem/Student | | **Primary Actor :** | Student | | **Goal :** | Purchase ticket | | **Pre-condition :** | User Having access to Web page. | | **Post-condition :** | Ticket Succesfully purchased | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User Access web home page. 2. System will show all details of event. 3. Admin click on event details. 4. System redirect the system to past event details. 5. User click on Purchase event ticket botton. 6. User fill required fields. 7. System verifies information and store ticket 8. User get success granted tickets. |
| **Alternative Path :** |

Table 2.3.21: Student upload fyp project

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Use case Number :** | 21 | | **Use case Name :** | Student upload fyp project | | **Level:** | Sysytem/Student | | **Primary Actor :** | Student | | **Goal :** | Upload fyp project | | **Pre-condition :** | User Having access to Web page. | | **Post-condition :** | Successfully upload project | | **Trigger :** | User | |
| **Main Success Scenario:**   1. User Access web home page. 2. User click on profile page. 3. User click upload Fyp project button. 4. System show form for fyp project. 5. User fill required fields. 6. System verifies information and store fyp project 7. User give messag successfully added project |
| **Alternative Path :** |

Chapter 3

Design

# DESIGN

## UML Diagrams

**FOR EXAMPLE:** UML, short for Unified Modeling Language, is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.

## Use-Case Diagram

### Admin

|  |
| --- |
| admin use case |

Figure 3.2: Use-Case Diagram for Admin

### Event Manager

|  |
| --- |
| event manager |

Figure 3.2: Use-Case Diagram for Event Manger

### Student

|  |
| --- |
| Student |

Figure 3.2: Use-Case Diagram for Student

## 

## Class Diagram

|  |
| --- |
| final class diagram geh |

Figure 3.3 Class Diagram

# Sequence Diagram

### Sign UP

|  |
| --- |
| 1 sign up |

Figure 3.4: Sign up

### Login

|  |
| --- |
| 2 login |

Figure 3.4: Login

### Logout

|  |
| --- |
| 3 log out |

Figure 3.4: log out

### Manage profile

|  |
| --- |
| 4 manage profile |

Figure 3.4: manage profile

### View Event

|  |
| --- |
| 5 view event |

Figure 3.4: View Event

### Manager add event

|  |
| --- |
| 6 manager add event |

Figure 3.4: Manager add event

### Delete Event

|  |
| --- |
| 7 delete event |

Figure 3.4: Delete Event

### Manager update event

|  |
| --- |
| 8 Manager update event |

Figure 3.4: Manager update event

### Manager add expence

|  |
| --- |
| 9 Manager add expence |

Figure 3.4: Manager add expence

### Manager update expence

|  |
| --- |
| 10 Manager update expence |

Figure 3.4:Manager update expence

### Manager add ticket

|  |
| --- |
| 11 Manager add ticket |

Figure 3.4: Manager add ticket

### Admin accept event

|  |
| --- |
| 12 Admin accept event |

Figure 3.4: Admin accept event

### 

### Admin reject event

|  |
| --- |
| 13 Admin reject event |

Figure 3.4: Admin reject event

### Admin Manage Managers

|  |
| --- |
| 14  Admin Manage Managers |

Figure 3.4: Admin Manage Managers

### Admin see feedback

|  |
| --- |
| 15 Admin see feedback |

Figure 3.4: Admin see feedback

### Admin see All tickets

|  |
| --- |
| 16 Admin see All tickets |

Figure 3.4: Admin see All tickets

### Student View upcoming event

|  |
| --- |
| 17 Student View upcoming event |

Figure 3.4: Student View upcoming event

### Student View past event

|  |
| --- |
| 18 Student View past event |

Figure 3.4: Student View past event

### Student rate event

|  |
| --- |
| 19 Student rate event |

Figure 3.4: Student rate event

### Purchase ticket

|  |
| --- |
| 20 Purchase ticket |

Figure 3.4: Purchase ticket

### Student upload fyp project

|  |
| --- |
| 21 Student upload fyp project |

Figure 3.4: Student upload fyp project

## 

## ERD

|  |
| --- |
| erd geh final |

Figure 3.5: ERD of HMP