****

**COMSATS University Abbottabad Campus**

**Project Proposal**

**UNIVERSITY EVENT MANAGEMENT SYSTEM**

**Group Members**

**Hamza Fareed FA21-BSE-056**

**Abdul Muhamin FA21-BSE-061**

**Ahmad Faraz FA21-BSE-047**

**Submitted Date 5/04/2023**

Table of Contents

[**Chapter 1 Project Proposal** 1](#_Toc132229629)

[**Introduction:** 1](#_Toc132229630)

[**Objectives:** 2](#_Toc132229631)

[**Scope:** 2](#_Toc132229632)

[**Benefits:** 2](#_Toc132229633)

[**Conclusion:** 2](#_Toc132229634)

[**Use Case model** 2](#_Toc132229635)

[**1. Functional Requirements** 2](#_Toc132229636)

[**2. Use cases** 4](#_Toc132229637)

[**3. 10% use case** 4](#_Toc132229638)

[**Supplementary Specification** 4](#_Toc132229639)

[**Risk List and Risk Management Plan** 5](#_Toc132229640)

[**Chapter 2 Use Cases** 7](#_Toc132229641)

[**Use case diagram** 7](#_Toc132229642)

[**Use Case Distribution** 7](#_Toc132229643)

[**Fully Use Dressed Case** 8](#_Toc132229644)

[Hamza Fareed 8](#_Toc132229645)

# **Chapter 1 Project Proposal**

## **Introduction:**

Organizing events is a crucial aspect of university life. It requires a lot of planning and management to ensure that the events are successful. Therefore, we propose the development of a University Event Management System that will assist in planning, organizing, and managing events within the university.

## **Objectives:**

The objective of the University Event Management System is to create a platform that will simplify the process of event management, enabling easy planning and execution of events. The system will be designed to meet the following objectives:

* Provide an online platform for event planning and management.
* Facilitate easy event scheduling and booking.
* Manage event logistics and resources.
* Track event attendance and participation.
* Generate reports for event analysis and future planning.

## **Scope:**

The University Event Management System will be a solution that covers all aspects of event management within the university. The system will be designed to cater to all types of events, including conferences, student week, seminars, workshops, and cultural events.

## **Benefits:**

* The University Event Management System will provide the following benefits:
* Simplify the event planning and management process.
* Eliminate manual processes, reducing the likelihood of errors and inefficiencies.
* Increase productivity and efficiency in event management.
* Improve communication and collaboration among event stakeholders.
* Provide data-driven insights for future event planning.

## **Conclusion:**

The University Event Management System will be a valuable tool for managing events within the university. It will provide a comprehensive solution for event planning and management, streamlining the process and increasing productivity and efficiency. The system will improve communication and collaboration among event stakeholders, enabling easy scheduling, booking, and tracking of events. The benefits of the system will be significant, including increased efficiency and improved communication.

## **Use Case model**

## **1. Functional Requirements**

**For User:**

* Registration: The system should allow users to register on the platform by providing basic personal information.
* Login: The system should allow users to log in using their registered credentials.
* View event details: The system should display event details such as date, time, location, and description.
* Register for an event: Users should be able to register for an event through the platform.
* Search events: The system should provide a search function that allows users to search for events based on various criteria such as date, location, and type of event.
* Provide feedback: Users should be able to provide feedback on an event they have attended.

**For Organizer:**

* Add an event: The system should allow organizers to add new events by providing basic event information.
* Cancel an event: The system should allow organizers to cancel an event if necessary.
* Organize an event: The system should provide the ability to book venues, arrange for food services, manage parking, and provide security and advertising for events.
* Verify an event: The system should send email notifications to organizers to verify their event details before it is posted on the platform.
* View event details: The system should allow organizers to view event details, registration lists, and feedback.

**For Admin:**

* Manage users: The system should provide admins with the ability to manage users and their information.
* Manage events: The system should allow admins to manage events by adding, modifying, or deleting events.
* Verify events: The system should verify events submitted by organizers before they are posted on the platform.
* Generate reports: The system should generate reports on user registrations, events, and feedback.
* Contact us: The system should provide an option for users to contact admin for any queries.
* About us: The system should provide information about the university and the event management system.

## **2. Use cases**

* Registration
* Login
* Details of event
* Registration for event
* Search event
* Add event
* Cancel event
* Organizing event(Book venues ,Food services, Security advertisement, Manage parking)
* Verify event(Send email)
* Feedback event
* Contact us
* About us

## **3. 10% use case**

* User registers on the platform by providing basic personal information
* User logs in to the system using their registered credentials
* User views event details such as date, time, location, and description
* User registers for an event through the platform
* User searches for events based on various criteria such as date, location, and type of event
* User provides feedback on an event they have attended
* Organizer adds a new event by providing basic event information
* Organizer cancels an event if necessary
* Organizer organizes an event by booking venues, arranging for food services, managing parking, providing security and advertising for events
* Admin manages users and their information

## **Supplementary Specification**

**Performance:**

The system should be able to handle a large number of users and events simultaneously without any performance issues.

**Security:**

The system should ensure that user information is protected and not accessible by unauthorized personnel.

**Scalability:**

The system should be designed to be scalable to accommodate future growth in the number of users and events.

**User interface:**

The user interface should be intuitive, easy to use, and aesthetically pleasing.

**Accessibility:**

The system should be designed to be accessible to users with disabilities.

**Mobile Responsiveness:**

The system should be designed to be responsive to mobile devices to allow users to access the system from anywhere.

**Analytics and Reporting:**

The system should provide analytics and reporting features to track user behaviour, event success, and generate reports.

**Email and Notification Integration:**

The system should integrate with email and notification services to keep users informed about event updates and promotions.

## **Risk List and Risk Management Plan**

1. **Risk List**

**Security Risks:**

Security risks such as hacking, data breaches, and unauthorized access to user information.

**Technical Risks:**

Technical risks such as system failures, crashes, and downtime.

**Performance Risks:**

Performance risks such as slow system response times, delays in event registration, and user frustration.

**User Acceptance Risks:**

User acceptance risks such as low user adoption, user dissatisfaction, and negative feedback.

**Legal Risks:**

Legal risks such as non-compliance with data privacy laws, intellectual property infringement, and contractual disputes with vendors.

**Financial Risks:**

Financial risks such as cost overruns, unexpected expenses, and revenue loss due to low event attendance

1. **Risk Management Plan**

**Security Risks:**

Implement strong password policies and two-factor authentication for user accounts.

**Technical Risks:**

* Perform regular backups of data to prevent data loss in case of system failures.
* Regularly monitor system performance and take proactive measures to prevent downtime and slow response times.

**Performance Risks:**

Implement load testing and performance monitoring to ensure that the system can handle a high volume of users and events.

**User Acceptance Risks:**

* Engage with users during the development process to ensure that the system meets their needs and expectations.
* Provide training and support to users to ensure that they understand how to use the system effectively.

**Legal Risks:**

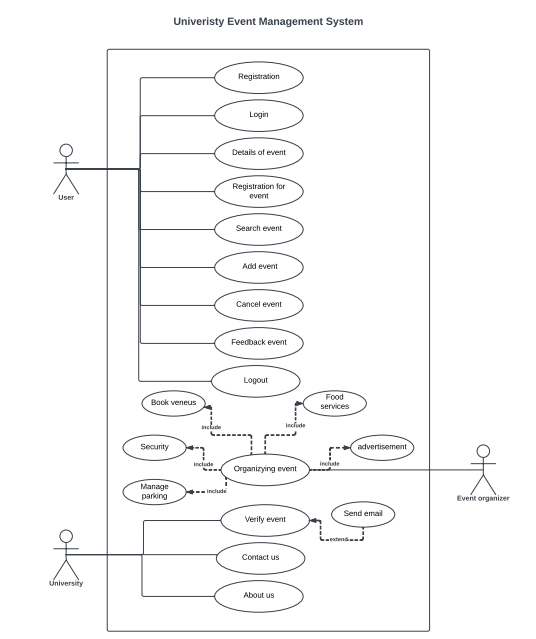
* Stay up to date with relevant data privacy laws and regulations to ensure compliance.
* Implement data protection policies and procedures to reduce the risk of data breaches and non-compliance.

**Financial Risks:**

Develop and regularly update a detailed budget to reduce the risk of cost overruns and unexpected expenses.

# **Chapter 2 Use Cases**

## **Use case diagram**



|  |  |
| --- | --- |
| **Group Members** | **Use Cases** |
| Hamza Fareed  FA21-BSE-056 | * Registration * Login * Details of event * Registration for event * Search event * Add event * Cancel event |
| Ahmad Faraz  FA21-BSE-047 | * Feedback event * Logout * Organizing event(Book venues, Food services, Security, advertisement, Manage parking) |
| Abdul Muhaimin  FA21-BSE-061 | * Verify event(Send email) * Contact us * About us |

## **Use Case Distribution**

|  |  |
| --- | --- |
| Hamza Fareed (FA21-BSE-056) | * Registration * Login * Details of event * Registration for event * Search event * Add event * Cancel event |
| Abdul Muhaimin (FA21-BSE-061) | * Verify event (send Email) * Contact us * About us |
| Ahmad Faraz (FA21-BSE-047) | * Feedback Event * Logout * Organizing Event(Book venues, Food services, Security advertisement , Manage parking) |

## **Fully Use Dressed Case**

### Hamza Fareed

#### **Use Case: Register**

|  |  |
| --- | --- |
| **Title** | Register |
| **Description:** | This use case describes the process of registering for a student account on the University Event Management System website. |
| **Level:** | User goal |
| **Primary Actor:** | User |
| **Preconditions:** | * The student must have access to the Internet. * The student must have a valid email address. |
| **Post conditions:** | * The student has successfully registered for an account on the University Event Management System website. * The student is able to log in to their account and access event registration functionality. |
| **Basic Flow:** | * The student navigates to the University Event Management System website. * The student clicks on the "Register" button on the website homepage. * The system displays the registration form for a student account. * The student fills out the registration form with their personal information, including their name, email address, and a chosen password. * The student submits the registration form. * The system verifies the email address is valid and has not been used to register another account. * The system creates a new student account for the email address provided and stores the student's personal information in the system's database. * The system sends a confirmation email to the email address provided during registration. * The student receives the confirmation email and clicks on the link provided to verify their email address. * The system verifies the email address and activates the student's account.   . |
| **Alternative Flows:** | * If the email address provided is already associated with an account: 1. the system displays an error message and prompts the student to log in or use a different email address. 2. The use case returns to step 4 in the basic flow. * If the student does not provide a valid email address: 1. the system displays an error message and prompts the student to provide a valid email address. 2. The use case returns to step 4 in the basic flow. |

#### **Use case: Login**

|  |  |
| --- | --- |
| **Title** | Login |
| **Description:** | This use case describes the process of logging into the University Event Management System website. |
| **Level:** | User goal |
| **Primary Actor:** | User |
| **Preconditions:** | * The student must have a registered account on the University Event Management System website. * The student must have access to the Internet. * The student must know their email address and password associated with their account. |
| **Post conditions:** | * The student is logged into their account on the University Event Management System website. * The student is able to access event registration functionality. |
| **Basic Flow:** | * The student navigates to the University Event Management System website. * The student clicks on the "Login" button on the website homepage. * The system displays the login form for the student account. * The student enters their email address and password associated with their account. * The student submits the login form. * The system verifies the email address and password provided match a registered account in the system's database. * The system logs the student into their account. * The system redirects the student to the homepage of their account on the University Event Management System website.   . |
| **Alternative Flows:** | * If the email address and password provided do not match a registered account: 1. the system displays an error message and prompts the student to try again or reset their password. * If the student forgets their password: 1. the student clicks on the "Forgot Password" link on the login form. 2. The system prompts the student to enter their email address associated with their account. 3. The system sends a password reset link to the email address provided. 4. The student receives the password reset link and clicks on it. 5. The system verifies the password reset link is valid and prompts the student to create a new password. 6. The student creates a new password and submits the password reset form. 7. The system updates the student's password in the system's database. |
| **Exceptions:** | * If the system is unavailable or experiencing technical difficulties: 1. the system displays an error message and prompts the student to try again later. 2. The use case ends. * If the student's email address is invalid or not recognized by the system: The system displays an error message and prompts the student to contact the system administrator for assistance. |

#### **Use case: Details of events**

|  |  |
| --- | --- |
| **Title** | Details of events |
| **Description:** | This use case describes the process of viewing the details of an event on the University Event Management System website. |
| **Level:** | User goal |
| **Primary Actor:** | User |
| **Preconditions:** | * The student must be logged into their account on the University Event Management System website. * The student must have access to the Internet. |
| **Post conditions:** | * The student is able to view the details of the selected event. |
| **Basic Flow:** | * The student navigates to the homepage of their account on the University Event Management System website. * The student clicks on the "Events" tab on the top navigation bar. * The system displays a list of upcoming events. * The student selects an event from the list by clicking on the event name or image. * The system displays the details page for the selected event, which includes: * Event title * Event description * Event date, time, and location * Event organizer information * List of speakers or performers * The student can view and read all of the information about the event on this page. |
| **Exceptions:** | * If the system is unavailable or experiencing technical difficulties: The system displays an error message and prompts the student to try again later. |

#### **Use Case: Registration for event**

|  |  |
| --- | --- |
| **Title** | Registration for event |
| **Description:** | This use case describes the process of registering for an event on the University Event Management System website. |
| **Level:** | User goal |
| **Primary Actor:** | User |
| **Preconditions:** | * The student must be logged into their account on the University Event Management System website. * The student must have viewed the details of the event they wish to register for. |
| **Post conditions:** | * The student is registered for the selected event. * The student is able to view the event in their list of registered events. |
| **Basic Flow:** | * The student navigates to the details page of the event they wish to register for. * The student clicks on the "Register" button on the event details page. * The system displays the registration form for the event, which includes: Student name and email address * The student completes the registration form and submits it. * The system verifies that the student has completed all required fields on the registration form. * The system checks if there are any conflicts between the selected event |
| **Exceptions:** | * If the system is unavailable or experiencing technical difficulties: 1. the system displays an error message and prompts the student to try again later. 2. The use case ends. |
| **Alternative flow** | * If there are scheduling conflicts between the selected event and the student's schedule: 1. the system displays a message indicating that there are scheduling conflicts. 2. The system prompts the student to confirm that they still wish to register for the event. 3. If the student confirms, the use case returns to step 6 in the basic flow. |

#### **Use Case: Search Event**

|  |  |
| --- | --- |
| **Title** | Search Event |
| **Description:** | This use case describes the process of searching for an event on the University Event Management System website. |
| **Level:** | User goal |
| **Primary Actor:** | User |
| **Preconditions:** | * The student must be logged into their account on the University Event Management System website. * The student must have access to the Internet. |
| **Post conditions:** | * The system displays a list of events that match the search criteria. * The student is able to view the details of the selected event |
| **Basic Flow:** | * The student clicks on the "Events" tab on the top navigation bar. * The system displays a search bar for events. * The student enters the search criteria in the search bar, such as keywords or event type. * The student selects an event from the list by clicking on the event name or image. * The system displays the details page for the selected event, which includes: * Event title * Event description * Event date, time, and location * Event organizer information * The student can view and read all of the information about the event on this page. |
| **Alternative Flow** | * If there are no events that match the search criteria: 1. the system displays a message indicating that there are no events that match the search criteria. 2. The use case ends. * If the student is unable to attend any of the events: 1. the student clicks on the "Back" button to return to the list of events. 2. The use case ends. |
| **Exception** | * If the system is unavailable or experiencing technical difficulties: 1. the system displays an error message and prompts the student to try again later. |

#### **Use Case: Add Event**

|  |  |
| --- | --- |
| **Title** | Add Event |
| **Description:** | This use case describes the process of a student selecting to add an event to the University Event Management System website. |
| **Level:** | User goal |
| **Primary Actor:** | User |
| **Preconditions:** | * The student must be logged into their account on the University Event Management System website. * The student must have access to the Internet. |
| **Post conditions:** | * The selected event is added to the student's list of saved events on the University Event Management System website. |
| **Basic Flow:** | * The student searches for an event using the search bar or by browsing through the list of events. * The student selects an event they are interested in by clicking on the event name or image. * The system displays the details page for the selected event * The student clicks on the "Add Event" button. * The system adds the event to the student's list of saved events on the University Event Management System website. * The system displays a confirmation message that the event has been successfully added to the student's list of saved events. |
| **Alternative Flow** | * If the student is not logged into their account on the University Event Management System website: the system prompts the student to log in or create an account. 2. The use case returns to step 1 in the basic flow. * A2. If the system is unable to add the event to the student's list of saved events due to technical difficulties: the system displays an error message and prompts the student to try again later. |
| **Exception** | * If the student has already added the selected event to their list of saved events: The system displays a message indicating that the event is already on the student's list of saved events. |