



**Daffodil**  
*International*  
**University**

**“SWE TOUR GUIDE”**  
**A Web Application for Online Software Department Tour Event**

**Supervised By:**

MD. Shohel Arman

lecturer

Department of Software Engineering

Daffodil International University

**Submitted By:**

MD.Mahdi Hasan Mahin

ID: 152-35-1300

BSc in Software Engineering

Daffodil International University

This Project report has been submitted in fulfillment of the requirements for the  
Degree of Bachelor of Science in Software Engineering.

© All right Reserved by Daffodil International University

## **APPROVAL**

This **Project** titled “**SWE SPORT EVENT REMINDER**”: A Web Application for **Online Reminder of Software Department Tour Event**, submitted by **MD.Mahdi Hasan Mahin, ID: 152-35-1300** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of BSc. in Software Engineering and approved as to its style and contents.

## **BOARD OF EXAMINERS**

-----  
**Dr. Touhid Bhuiyan**  
**Professor and Head**

Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

**Chairman**

-----  
**Md. Hasan Sharif**  
**Lecturer**

Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

**Internal Examiner**

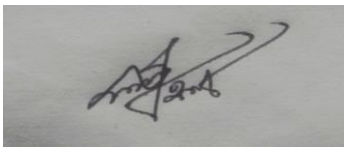
-----  
**Mr. Shafiul Hasan**  
**Managing Director**

Vivacom Solution, Dhaka

**External Examiner**

## DECLARATION

We hereby declare that, this project has been done by me under the supervision of **Md.Shohel Arman, Assistant Professor, Department of Software Engineering, Daffodil International University**. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.



---

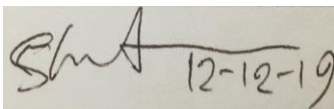
**MD.Mahdi Hasan Mahin**

ID: 152-35-1300

Department of Software Engineering

Daffodil International University

**Certified by:**



---

**MD.Shohel Arman**

Senior lecturer

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

## ACKNOWLEDGEMENT

First of all I express my heartiest thanks, and gratefulness to Almighty Allah for His divine blessing makes us possible to complete the final year Project Successfully. We really grateful and wish our profound our indebtedness to **Md Shohel Arman, Assistant Professor, Department of Software Engineering, Daffodil International University**. Deep Knowledge & keen interest of our supervisor in the field of “**A Web Application for Online Reminder of Software Department Tour Event**” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

I would like to express our heartiest gratitude to **Pro. Dr. Touhid Bhuiyan, Professor, and Head, Department of SWE**, for his kind help to finish our project and also to other faculty member and the staff of SWE department of Daffodil International University.

I would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work. Finally, we must acknowledge with due respect the constant support and patients of our parents.

## **EXECUTIVE SUMMARY**

Our project titled “Tour guide: A Web Application for Online Event registration and informing its user through message, notification, post about different event about the Tour. The main aim of this free Tour guide project is make information dissemination much easier in a paperless community as the world tends to interact with the Tour Event Reminder facility as an project, Tour Event Reminder, admin can send the notification to the particular players SWE Events Sport Reminder work generally intends to act as a support system for the all users.

We have used Bootstrap, HTML, CSS and Java Script for front-end design and validations and PHP programming language for back-end design. We also used My SQL Server as our main database.

After finishing all the task and test process this application proved to be working effectively.

## Table of Contents

CHAPTER 1 .....	11
INTRODUCTION .....	11
1.1 Project Overview: .....	11
1.2 Project Purpose: .....	11
1.2.1 Background: .....	11
1.2.2 Benefits: .....	12
1.2.3 Goals: .....	12
1.3 Stakeholders: .....	12
1.4 Proposed System Model (Block Diagram): .....	13
1.5 Project Schedule: .....	14
1.5.1 Gantt Chart: .....	14
1.5.2 Milestone: .....	14
CHAPTER 2 .....	15
SOFTWARE REQUIREMENT SPECIFICATION .....	15
2.1 .....	15
Functional Requirements: .....	15
2.2 Performance Requirements: .....	16
2.2.1 Speed and Latency Requirements: .....	16
2.2.2 Accuracy Requirements: .....	17
2.2.3 Capacity Requirements: .....	17
2.3 Dependability Requirements: .....	18
2.3.1 Reliability Requirements: .....	18
2.3.2 Availability Requirements: .....	18
2.4 Maintainability and supportability requirements: .....	18
2.5 Maintenance Requirements: .....	19
2.5.1 Supportability Requirements: .....	19
2.6 Security requirements: .....	19
2.6.1 Access Requirements: .....	19
2.6.2 Integrity Requirements: .....	19
2.6.3 Privacy Requirements: .....	19
2.7 Usability and human-interaction requirements: .....	19
2.7.1 Ease of Use Requirements: .....	20
2.7.2 Personalization and Internationalization Requirements: .....	20
2.7.3 Understandability and politeness Requirements: .....	20
2.7.4 Accessibility Requirements: .....	20
2.7.5 User Documentation Requirements: .....	20
2.7.6 Training Requirements: .....	20

2.8	Look and feel requirements .....	20
2.8.1	Appearance Requirements .....	20
2.8.2	Style Requirements .....	20
CHAPTER 3 .....		21
SYSTEM ANALYSIS .....		21
3.1	Use Case Diagram: .....	21
3.2	Use Case Description: .....	22
3.2.1	User Authentication: .....	22
3.2.2	Manage User: .....	23
3.2.3	Manage Notice: .....	24
3.2.4	Manage Account: .....	25
3.2.5	Event Registration:.....	26
3.3	Activity Diagram: .....	27
3.3.1	Activity Diagram for Tour guide: .....	27
CHAPTER 4 .....		28
SYSTEM DESIGN SPECIFICATION.....		28
4.1	Class Responsibilities Collaboration (CRC) Cards: .....	28
4.2	Sequence Diagram: .....	29
4.3	Entity Relationship Diagram:.....	30
4.4	Development Tools & Technology:.....	31
4.4.1	User Interface Technology:.....	31
CHAPTER 5 .....		32
SYSTEM TESTING .....		32
5.1	Testing Features: .....	32
5.1.1	Features to be tested:.....	32
5.1.2	Features not to be tested:.....	32
5.2	Testing strategies: .....	32
5.2.1	Test Approach: .....	32
5.2.2	Pass/Fail criteria:.....	33
5.2.3	Suspension & Resumption: .....	33
5.3	.....	33
Testing Environment:.....		33
5.4	Test Case Description: .....	33
5.4.1	Test Case for Admin Login:.....	33
5.4.2	Test Case for Event Entry: .....	34
5.4.3	Test Case for User Login: .....	34
5.4.4	Test Case for Posting Notice:.....	35
CHAPTER 6 .....		36
USER MANUAL.....		36

6.1	Home Page for Authenticate User: .....	36
6.2	Home Page for Guest User: .....	37
6.3	Admin Dashboard: .....	37
6.3.1	Manage User .....	38
6.3.2	Manage User .....	38
6.3.3	Create Event: .....	39
6.3.4	All Event: .....	39
6.3.5	Update Event:.....	40
6.3.6	Delete Event:.....	40
6.3.7	Create Notice: .....	41
6.3.8	All Notice:.....	41
6.3.9	Update Notice: .....	42
6.3.10	All Pending Users: .....	42
6.3.11	Approved User List:.....	43
6.3.12	Update Password:.....	43
6.4.	User Registration: .....	44
6.4.1	User Login: .....	44
6.4.3	User Profile: .....	45
6.4.4	Update Profile: .....	45
6.4.5	Check Notice:.....	46
6.4.6	Update Password:.....	46
6.5	Committee:.....	47
6.5	About: .....	47
CHAPTER 7 .....		48
PROJECT SUMMARY .....		48
7.1	Critical Evolution:.....	48
7.2	Limitations: .....	48
7.3	Obstacles & Achievements: .....	48
7.4	Future Scope: .....	48
7.5	Discussion and Conclusion: .....	49
REFERENCES .....		50



## LIST OF FIGURES

Figure 1.1:	Registrar Block Diagram-----	3
Figure 1.2:	Gantt chart -----	4
Figure 3.1:	Use Case Diagram -----	11
Figure 3.2:	Activity Diagram -----	17
Figure 4.1:	Sequence Diagram -----	23
Figure 4.2:	Entity Relation Ship Diagram -----	24
Figure 6.1:	Home Page1 -----	30
Figure 6.2:	Home Page2-----	31
Figure 6.3:	Admin Dashboard-----	31
Figure 6.4:	Manage User -----	32
Figure 6.5:	Delete User-----	32
Figure 6.6:	Create Event -----	33
Figure 6.7:	All Event--- -----	33
Figure 6.8:	Update Event-----	34
Figure 6.9:	Delete Event -----	34
Figure 6.10:	Create Notice-----	35
Figure 6.11:	All Notice-----	35
Figure 6.12:	Update Notice -----	36
Figure 6.13:	Pending User List-----	36
Figure 6.14:	Approved User list-----	37
Figure 6.15:	Update Password -----	37
Figure 6.16:	User Registration-----	38
Figure 6.17:	User Login -----	38
Figure 6.18:	User Profile-----	39
Figure 6.19:	Update User Profile-----	39
Figure 6.20:	Check Notice-----	40
Figure 6.21:	User Pass Update -----	40
Figure 6.22:	Committee-----	41
Figure 6.23:	About-----	41

## LIST OF TABLES

Table 1.1:	Milestone -----	4
Table 2.1:	Functional Requirement -----	5
Table 2.2:	Speed and Latency Requirements -----	6
Table 2.3:	Accuracy Requirements -----	7
Table 2.4:	Capacity Requirements -----	7
Table 2.5:	Reliability Requirements -----	8
Table 2.6:	Availability Requirements -----	8
Table 2.7:	Availability Requirements -----	8
Table 3.1:	Use Case Description of User Authentication -----	12
Table 3.2:	Use Case Description of Manage User-----	13
Table 3.3:	Use Case Description of Notice Management-----	14
Table 3.4:	Use Case Description of Managing Account -----	15
Table 3.5:	Use Case Description of Registering process of Event -----	16
Table 4.1:	Admin Responsibilities and collaboration-----	18
Table 5.1:	Test Case for Admin Login-----	27
Table 5.1:	Test Case for Admin Login -----	27
Table 5.2:	Test Case for Event Entry -----	28
Table 5.3:	Test Case for User Login-----	28
Table 5.4:	Test Case for Posting Notice-----	29

# CHAPTER 1

## INTRODUCTION

### 1.1 Project Overview:

A web application that provides SWE tour event schedule and provide notice to its users. Tour guide is a place where people can leave any types of messages and notifications The main aim of this free Tour guide project is make information dissemination much easier in a paperless community as the world tends to interact with the tour Event Reminder facility as an project, Tour Event Reminder, admin can send the notification to the particular players SWE Events Sport Reminder work generally intends to act as a support system for the all users. Admin will maintain the system like as: creating event, sending notification, approving people, removing people, managing user. User will be able to create account in order to get all information and for tour event entry. Room allocation for accommodation, without an account a user cannot accessed any event. So, it's an easier way to inform the user.

### 1.2 Project Purpose:

The main purpose of your project is to make a digital Tour event reminder system. Where users can get notify before any departmental tour information of Software Department.

#### 1.2.1 Background:

Tour guide is a place where people can leave any types of messages and notifications The main aim of this free Tour guide project is make information dissemination much easier in a paperless community as the world tends to interact with the Tour Event Reminder facility as an project, Tour Event Reminder, admin can send the notification to the particular players SWE Events Sport Reminder work generally intends to act as a support system for the all users.

### 1.2.2 Benefits:

There are several benefits that I have listed below.

- ✓ This is a user-friendly digitalized system
- ✓ Student can entry to the event through the system.
- ✓ Everyone will be notified before the event start.
- ✓ Admin can announced anything through the system

### 1.2.3 Goals:

We want to make this web application. This web application will have contained these features.

- ✓ To keep all fundamental information about the SWE Tour Events.
- ✓ To notify everyone about the event who are associated with the event.
- ✓ To provide online registration.
- ✓ To provide online notice.

## 1.3 Stakeholders:

Every system is developed by targeting some stakeholders. A system has no value without stakeholders. Our system also has some stakeholders:

- **Students:** Students are the main internal stakeholder of our system. Because they are the participants of the Tour event and spectators.
- **Teacher:** Though the events is under the department so teacher are also internal stake holders. They will manage the event.
- **Admin:** Admin will be assigned by the university authority. Admin has the authority or power to control the whole system.

## 1.4 Proposed System Model (Block Diagram):

This is the block diagram of the whole system of Tour guide. When a digital complex system is shown through a diagram in a schematic way from the general arrangement, then it is called a Block Diagram [5].

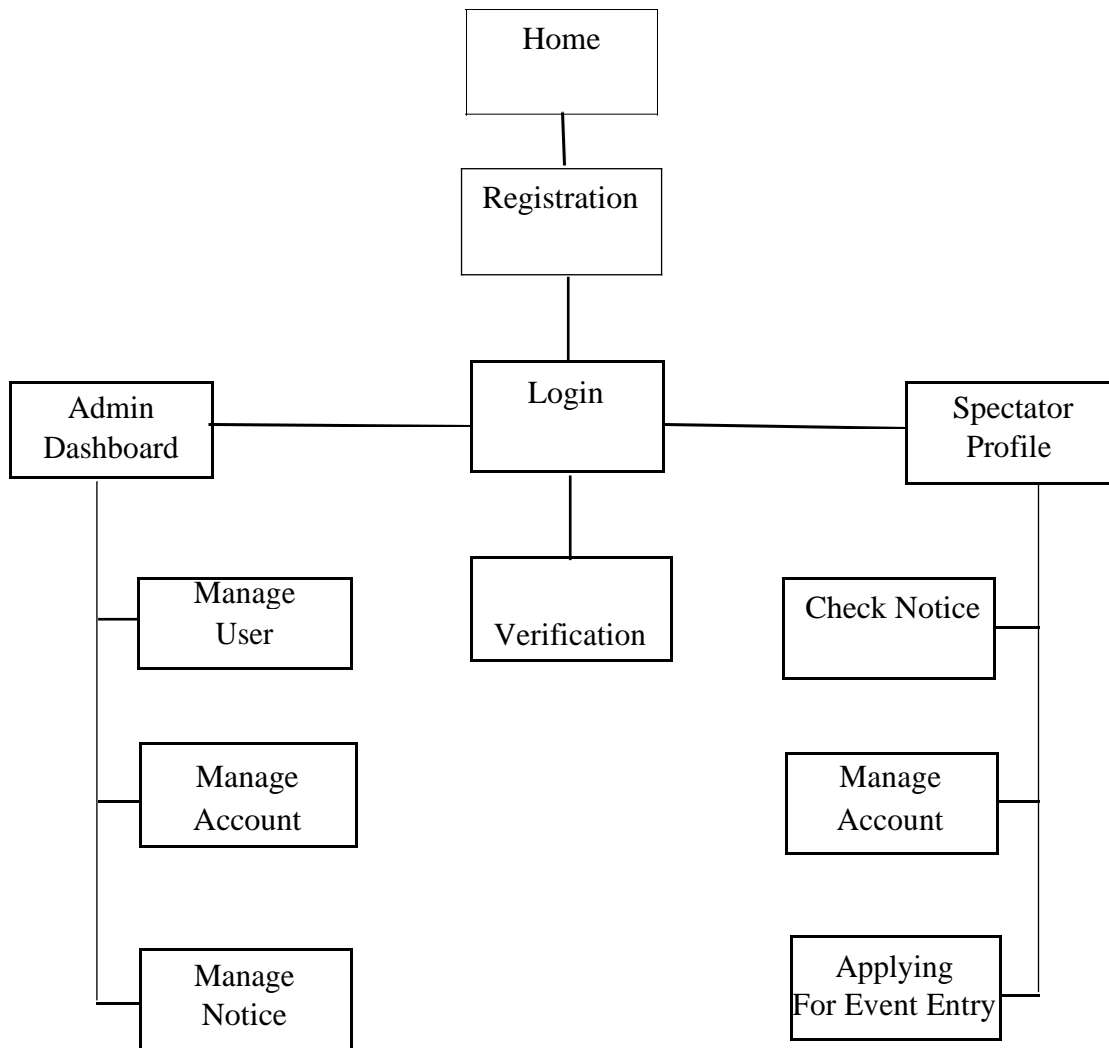


Figure 1.1: Registrar Block Diagram

## 1.5 Project Schedule:

To complete the project, I have maintained a schedule. So, it was easier for me to see the continuous progress of myself in order to complete the project.

### 1.5.1 Gantt Chart:

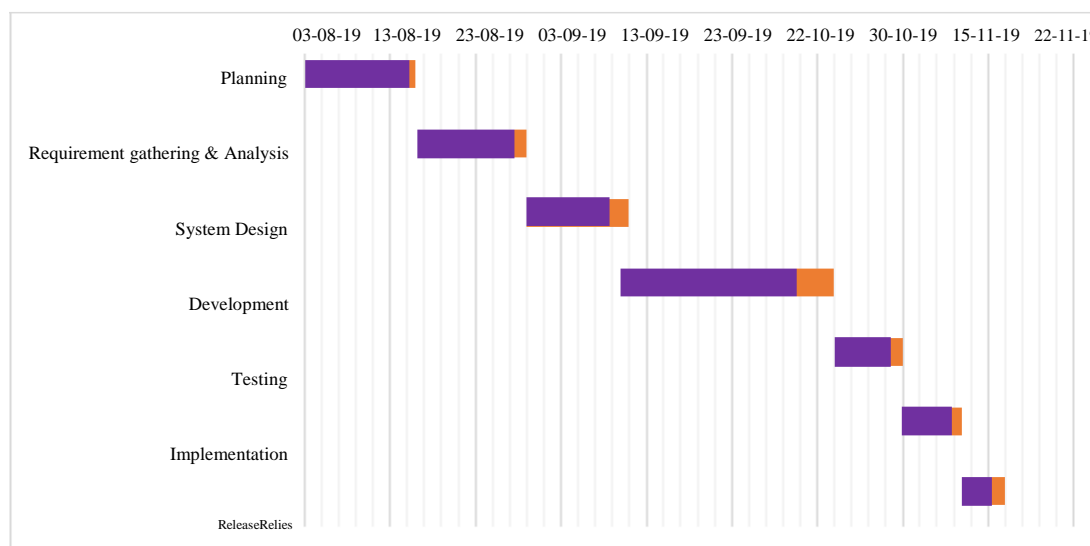


Figure 1.2: Gantt chart

### 1.5.2 Milestone:

Milestone, it is a time frame of project. That will define the project task. Project milestones are as follows:

Table 1.1: Milestone

Task no.	Task Name	Start Date	End Date	Duration
01	Planning	03-08-19	16-08-19	13 days
02	Requirements Gathering & Analysis	17-08-19	28-08-19	12 days
03	System Design	29-08-19	15-09-19	17 days
04	Development	16-09-19	22-10-19	37 days
05	Testing	23-10-19	30-10-19	07 days
06	Implementation	01-11-19	10-11-19	10 days
07	Release	11-11-19	22-11-19	11 days
	<b>Total</b>			<b>107 days</b>

## CHAPTER 2

### SOFTWARE REQUIREMENT SPECIFICATION

Requirement analysis is the first step of identifying user needs in order to satisfy the users. This is the most important part of any software development.

#### 2.1 Functional Requirements:

System's Functional Requirements are listed below:

---

Table 2.1: Functional Requirement

Requirement Id	Requirement Name	Description	Functional/Non Functional	Priority
01	Sign Up	Users should have to fill up email, password, name, mobile number, image, gender and hobbies for being authenticate	Functional	High
02	Log in	Users must have valid email and password to get into the system	Functional	High
03	Add User	Admin can add new user	Functional	Medium
04	Delete User	Admin can delete user	Functional	High
05	Add notice	Admin can add notice	Functional	High

<b>Requirement Id</b>	<b>Requirement Name</b>	<b>Description</b>	<b>Functional/Non-Functional</b>	<b>Priority</b>
07	Send Notice	Admin can send notice to the specific user	Functional	High
08	Update Password	Admin and user can update their own password	Functional	High
09	Check Notification	User can check notification	Functional	High
10	Update Profile	User can update their profile	Functional	Medium

## 2.2 Performance Requirements:

It's very necessary to sustain the performance of the project. To assure the better performance, this project has to meet some requirements which will provide the better performance.

### 2.2.1 Speed and Latency Requirements:

While user will run our project in the browser, system needs a minimum amount of speed to perform the task.

Table 2.2: Speed and Latency Requirements

SLR-01	Processing speed of the system will be faster.
Description	When user will run our system speed will be dependent on their internet speed & the server Bandwidth speed.
Stakeholders	Users as players/viewers, Admin etc.



### 2.2.2 Accuracy Requirements:

System have to confirm the Legibility and Accuracy of the data.

Table 2.3: Accuracy Requirements

AR-01	Required proper information of every user
Description	The input data should be correct and right pattern data, otherwise the input field show error message, like personal information, phone number, email address, password etc. the input information is not Valid, the data never saved. Or the input data pattern is not match, then users will not be able to go Forward step with the system.
Stakeholders	Users as players/viewers, Admin etc.

### 2.2.3 Capacity Requirements:

The system should maintain all inserting data

Table 2.4: Capacity Requirements

CR-01	Manages all the data in database properly
Description	All data like, Users registration data, Tour Event Registration data, notice etc. are Stored in the database.
Stakeholders	Users as players/viewers, Admin etc.

## 2.3 Dependability Requirements:

Dependability means, it measures of a system reliability, availability, robustness, safety etc. Here, dependability means the running time of this project.

### 2.3.1 Reliability Requirements:

Table 2.5: Reliability Requirements

RR-01	The system is reliable
Description	Four event information, user information, time schedule and others information will be safe and Secured in our system.
Stakeholder	Users/viewers, Admin etc.

### 2.3.2 Availability Requirements:

Table 2.6: Availability Requirements

AR-01	The system will be available 24x7.
Description	It is available 24 hours in a day and 7 days in a week The system will be updated regularly.
Stakeholder	Users as players/viewers, Admin etc.

### 2.3.3 Robustness or Fault Tolerance requirements:

Table 2.7: Availability Requirements

FTR-01	Well robustness of the system
Description	If any problem occurs the system will show error message and the fault tolerance is handled properly.
Stakeholder	Users as players/viewers, Admin etc.

## 2.4 Maintainability and supportability requirements:

Authority would have assigned some Admins and specialist for the maintenance of the system and support the system.

## **2.5 Maintenance Requirements:**

Authority would have to assign some specialists for maintaining the system. System can produce wrong results and the specialists must be able to reproduce the data flow through the system.

### **2.5.1 Supportability Requirements:**

Authority would have to recruit some admins for supportability of the system.

## **2.6 Security requirements:**

There are two security to get access to this system or a specific module of the system must provide an authentication mechanism.

### **2.6.1 Access Requirements:**

This system provides accesses the different module, by access in an authentication way to the authentic user.

### **2.6.2 Integrity Requirements:**

Credential information of users' will be kept safe.

### **2.6.3 Privacy Requirements:**

All the passwords and personal information will be kept safe and securely in the system.

## **2.7 Usability and human-interaction requirements:**

This system is very user friendly with simple user-interface.

### **2.7.1 Ease of Use Requirements:**

Users can use the system very easily with satisfaction and efficiency.

### **2.7.2 Personalization and Internationalization Requirements:**

I have developed this system in order to help the Software department Tour event. So, there are no internationalization requirements.

### **2.7.3 Understandability and politeness Requirements:**

This system is very easy to use and understand with simple user-interface.

### **2.7.4 Accessibility Requirements:**

This system is very easily accessible for definite user in definite module.

### **2.7.5 User Documentation Requirements:**

Events information and user records will be stored and kept in the system properly.

### **2.7.6 Training Requirements:**

There is not needed any extensive training for operating this digital system.

## **2.8 Look and feel requirements**

According to client's expectation look means the graphical user interface, design, layout, shapes, colors etc. and feel means the behavior of different elements like menu bar, buttons etc.

### **2.8.1 Appearance Requirements**

Needed knowledge about PHP, My SQL and algorithm.

### **2.8.2 Style Requirements**

Knowledge about HTML, CSS, JavaScript and bootstrap.

# CHAPTER 3

## SYSTEM ANALYSIS

### 3.1 Use Case Diagram:

Use case Diagram is the simplest way of representing in a graphical representation of a complex system's user interactions among the elements of the system. Marriage Registrar's use case diagram is given below:

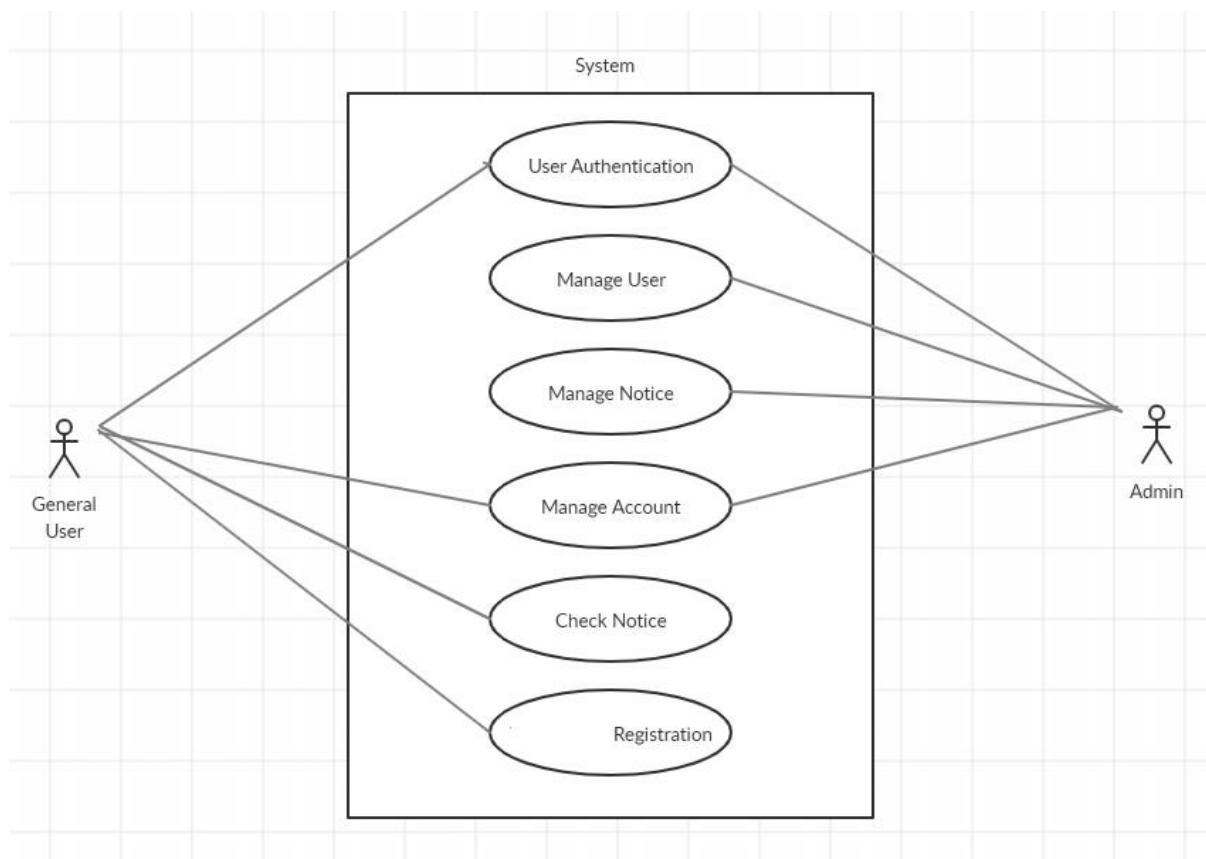


Figure 3.1 Use Case Diagram

### 3.2 Use Case Description:

This part will describe about the task briefly about how the users needed to interact with the system.

#### 3.2.1 User Authentication:

Table 3.1: Use Case Description of User Authentication

Use case name	User Authentication	
Scenario	Authenticate the users through registration and log in process	
Brief	To get access to the system a user must be authenticate. He must create an account. After the creation of the account the user will get access to his authorized resources.	
Actor	Admin, Viewers/Players	
Preconditions	Name, Email, Mobile Number etc. Must exist for Creating an account.	
Post conditions	New User must be added.	
Flow of events	Actor	System
	Users	Account Created successfully
Failure results	If system not response then show an error message.	

### 3.2.2 Manage User:

Table 3.2: Use Case Description of Manage User

Use case name	Manage User	
Scenario	Admin have to manage the user list.	
Brief description	Admin will have the privileges to manage the user such as check the user list, delete /remove user from the list, block user etc.	
Actor	Admin	
Preconditions	Admin must exist.	
Post conditions	User list will be displayed	
Flow of events	Actor	System
	Enter email And password	Response with displaying Proper information.
Failure results	If system not response then show an error message.	

### 3.2.3 Manage Notice:

Table 3.3: Use Case Description of Notice Management

Use case name	Manage Notice	
Scenario	Admin will manage the notice	
Brief description	Admin will send notice to the user. He will also be able to modify and delete the notice.	
Actor	Admin,	
Preconditions	Admin Must exist	
Post conditions	A successful message is shown by the system.	
Flow of events	Actor	System
	Enter email And pass for investigating the notice	Response with displaying a successful message.
Failure results	If system not response then show an error message.	



### 3.2.4 Manage Account:

Table 3.4: Use Case Description of Managing Account

Use case name	Manage Account	
Scenario	User can manage his own account	
Brief description	When a user creates an account, a profile is created. Day by day it's needed to be updated so here user can Update his information	
Actor	Admin, Participants	
Preconditions	Must have an account	
Post conditions	After all changes a successful message will be showed	
Flow of events	Actor	System
	Enter valid information	Response with displaying proper information.
Failure results	If system not response then show an error message.	

### 3.2.5 Event Registration:

Table 3.5: Use Case Description of Registering process of Event

Use case name	Event Registration	
Scenario	When a Event /Tour event is posted user can applying through the system.	
Brief description	When a Event is arranged a notice will be created by the admin and the willing participant can make registration through the system	
Actor	Participants	
Preconditions	Must have an account	
Post conditions	When all valid information is provided then a successful Message will be showed	
Flow of events	Actor	System
	Enter player List number for Registration.	Response with displaying Success message.
Failure results	If system not response then show an error message.	

### 3.3 Activity Diagram:

In Unified Modeling Language Activity diagram is an important diagram. Basically, it is a flow chart. It can describe the flow of the activity in a graphical representation [6].

#### 3.3.1 Activity Diagram for Tour guide:

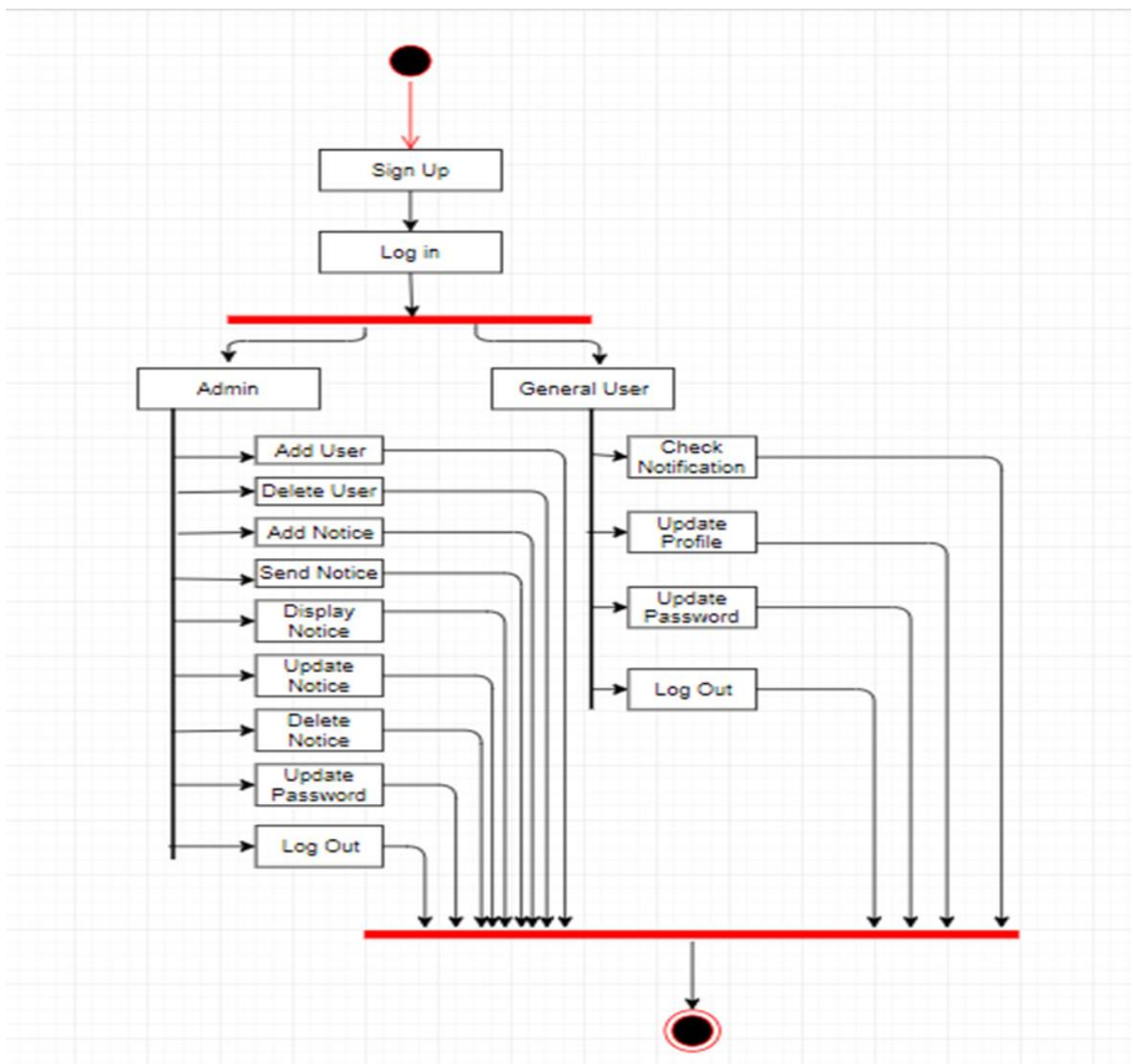


Figure 3.2 Activity Diagram

# CHAPTER 4

## SYSTEM DESIGN SPECIFICATION

### 4.1 Class Responsibilities Collaboration (CRC) Cards:

It's a brainstorming tool used to design any kind of object-oriented software. Following tables are given below:

#### **Admin:**

Table 4.1: Admin Responsibilities and collaboration

Responsibilities	Collaboration
Monitoring the system Enquiry about regular tasks	Send Notifications, Post Events, Manage Users, Manage Account

#### **User/Participants:**

Table 4.2: User Responsibilities and collaboration

Responsibilities	Collaboration
Log in with proper Information. Check notification, Event Registration	Authentication Process Check Notification  Event Registration

## 4.2 Sequence Diagram:

Sequence diagram is another very important UML diagram of a software. Sequence diagram can describes the working process of a system as how it works and what order is following it to complete a process [3].

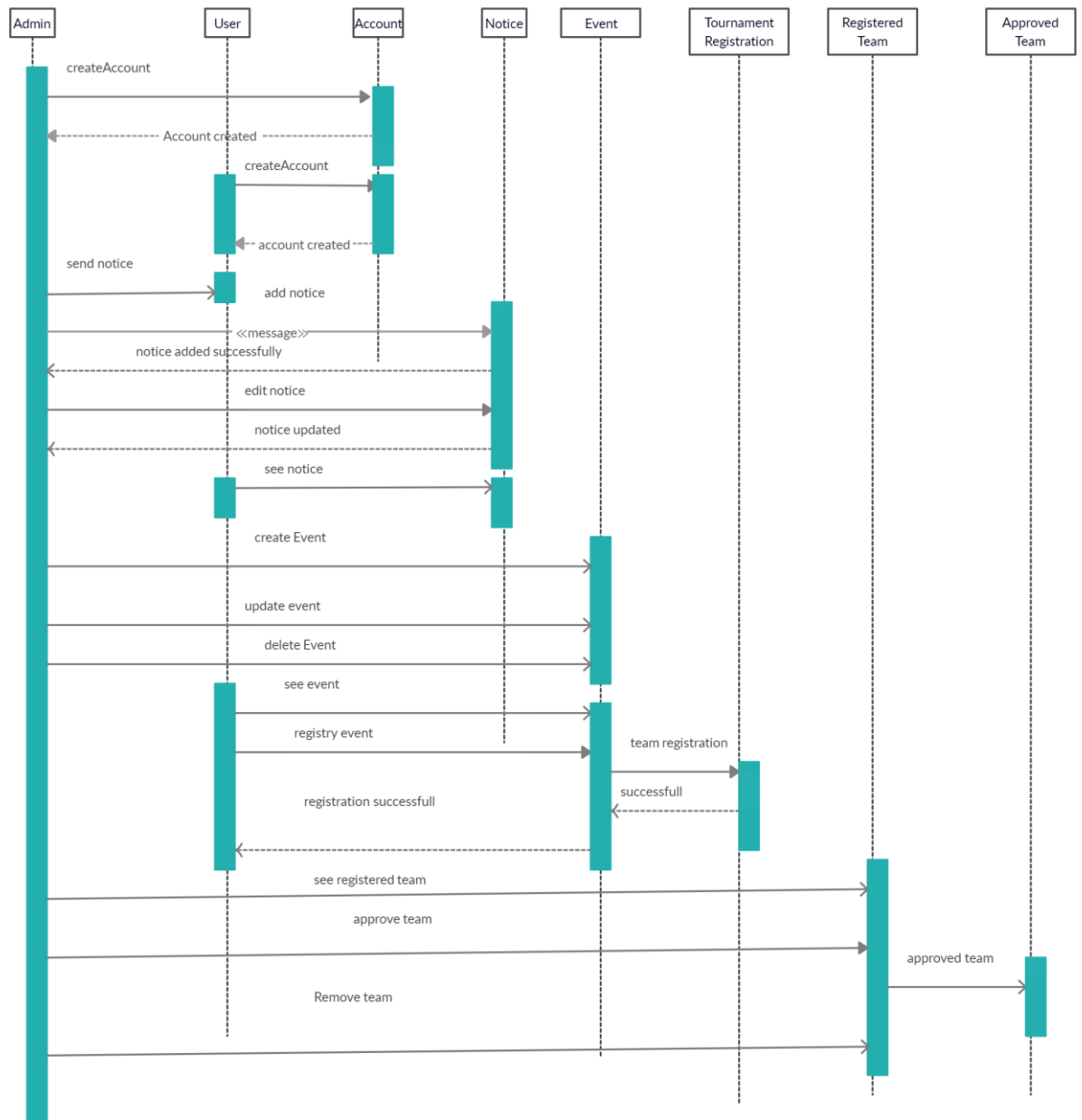


Figure 4.1: Sequence Diagram

### 4.3 Entity Relationship Diagram:

Database is the most important part of any system. We have tried to design a dynamic database for our system [9]. Our database design diagram is given below:

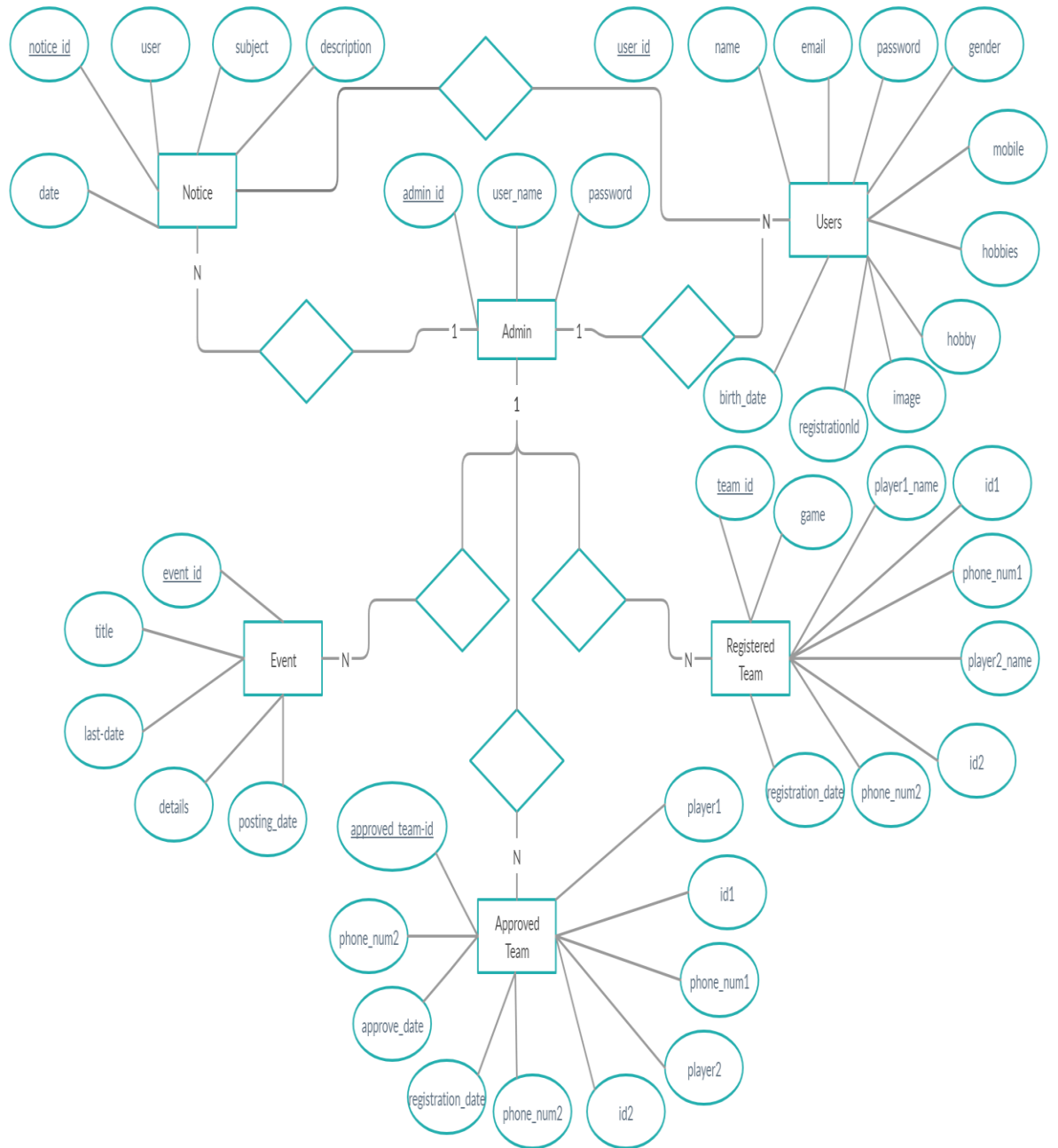


Figure 4.2: Entity Relation Ship Diagram

## 4.4 Development Tools & Technology:

For developing a software system, selecting development tools and technology is a vital fact and challenge.

Development tools:

- Development IDE: Sublime Text 3
- Server: MY SQL Server, Xampp Serve
- Operating System: Windows 10

Development technology:

- Programming language: PHP

### 4.4.1 User Interface Technology:

- HTML
- CSS
- Bootstrap
- JavaScript

# CHAPTER 5

## SYSTEM TESTING

### 5.1 Testing Features:

Testing is one of the most important part for checking the functionality of a software or system. Testing every features of a software we can ensure that the software or system is ready to provide its service [8].

#### 5.1.1 Features to be tested:

There is given a list of the features we have tested

+

- ✓ Admin Login
- ✓ User Login
- ✓ Post Notice
- ✓ Manage User
- ✓ Event Registration

#### 5.1.2 Features not to be tested:

We have tested every field of our system.

### 5.2 Testing strategies:

There is given the strategies of our system or software testing.

- Quality test
- Measure test

#### 5.2.1 Test Approach:

To implementation a project test approach is very important. Test approach has two types of techniques:

- **Proactive** - An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.
- **Reactive** - An approach in which the testing is not started until after design and coding are completed.



### 5.2.2 Pass/Fail criteria:

- When the expected result is come after testing then it is called pass
- When unexpected result is come during testing then it is called fail.

### 5.2.3 Suspension & Resumption:

Our system is very smooth and easy to use and maintain. There is no suspension and resumption problem.

### 5.3 Testing Environment:

- Testing IDE : Sublime Text3
- Browser : Google chrome
- Server : My SQL Server,Xampp
- Operating System : Windows 10
- Possessor : Core I 5

### 5.4 Test Case Description:

We have tested our system many times. Now we are focusing on that our system will execute properly.

#### 5.4.1 Test Case for Admin Login:

Table 5.1: Test Case for Admin Login

Test Scenario:	Admin Login	Test Case ID:	TC01		
Test Case Description:	Login test case	Test Priority:	High		
Pre-Requisite:	A valid user account	Post-Requisite:	NA		
<b>Test Execution Steps:</b>					
SL. No	Action	Inputs	Expected Result	Actual Result	Test Result
01	Enter correct Email and password	Email: tanjim@gmail.com Password: *****	Login Successful	Login Successful	Pass

### 5.4.2 Test Case for Event Entry:

Table 5.2: Test Case for Event Entry

Test Scenario:	Event Registration	Test Case Id:	TC02
Test Case Description:	Event entrance Registration	Test Priority:	High
Pre-Requisite:	An authenticate User	Post-Requisite:	Na

#### Test Execution Steps:

SL. No	Action	Inputs	Expected Result	Actual Result	Test Result
01	Enter Correct Name, Email, Mobile	Name: Rahman Email: Mahin1424@gmail.com  Mobile No: 01628190891	Registration Successful	Registration Assign Successful	Pass

### 5.4.3 Test Case for User Login:

Table 5.3: Test Case for User Login

Test Scenario:	User Login	Test Case ID:	TC03
Test Case Description:	Login test case	Test Priority:	High
Pre-Requisite:	A valid user account	Post-Requisite:	NA

#### Test Execution Steps:

SL. No	Action	Inputs	Expected Result	Actual Result	Test Result
01	Enter correct email and password	NID: <a href="mailto:Mahin1424@gmail.com">Mahin1424@gmail.com</a> Password: *****	Login Successful	Login Successful	Pass

#### 5.4.4 Test Case for Posting Notice:

Table 5.4: Test Case for Posting Notice

Test Scenario:	Notice Management	Test Case ID:	TC04		
Test Case Description:	Posting notice and management of notice	Test Priority:	High		
Pre-Requisite:	An authenticated Admin	Post-Requisite:	NA		
<b>Test Execution Steps:</b>					
SL. No	Action	Inputs	Expected Result	Actual Result	Test Result
01	Enter Title Con tent	Tour Tomorrow xxxxxx xxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxx	Successful	Successful	Pass

# CHAPTER 6

## USER MANUAL

### 6.1 Home Page for Authenticate User:

This is the home page of SWE Tour guide. Through home page user will get different event information and they can register for the entrance of the Event.

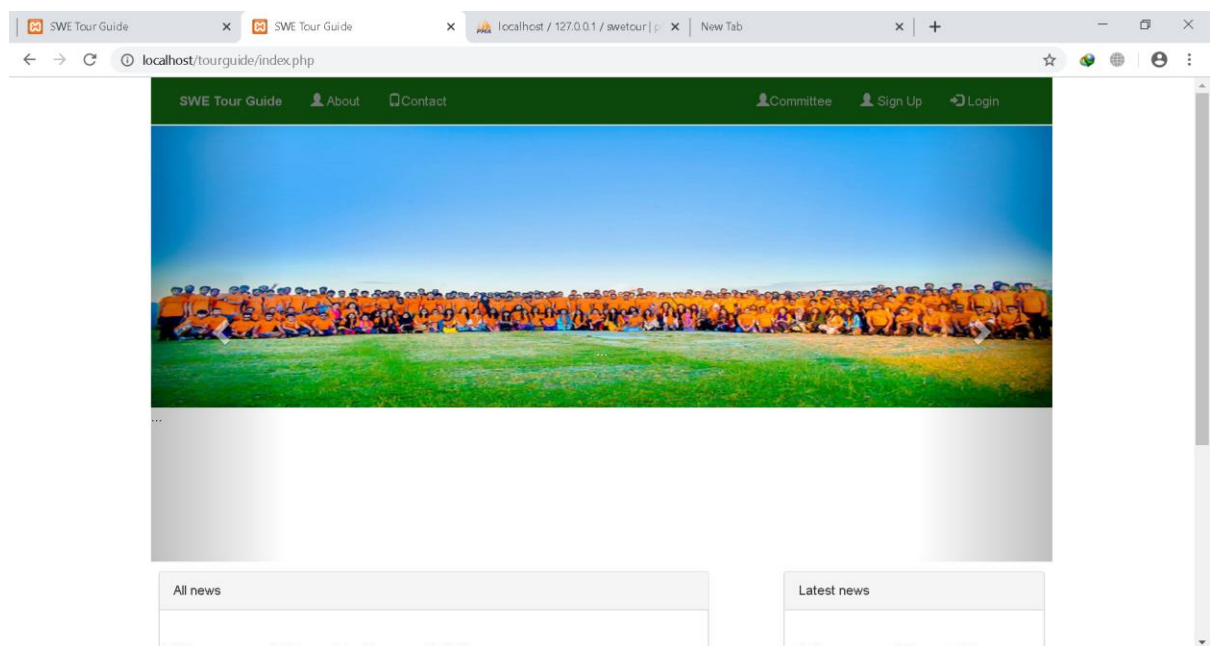


Figure 6.1: Home Page1

## 6.2 Home Page for Guest User:

Guest user will not get the button to register the event. In order to register the Event, he has to create an account

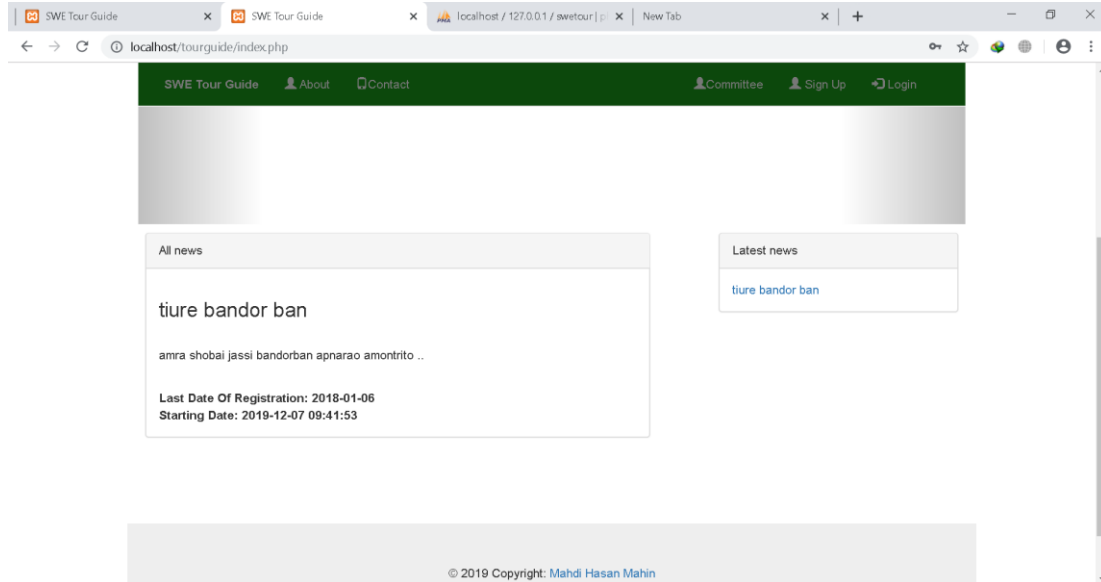


Figure 6.2: Home Page2

## 6.3 Admin Dashboard:

This is the admin dashboard. From here admin control the whole system.

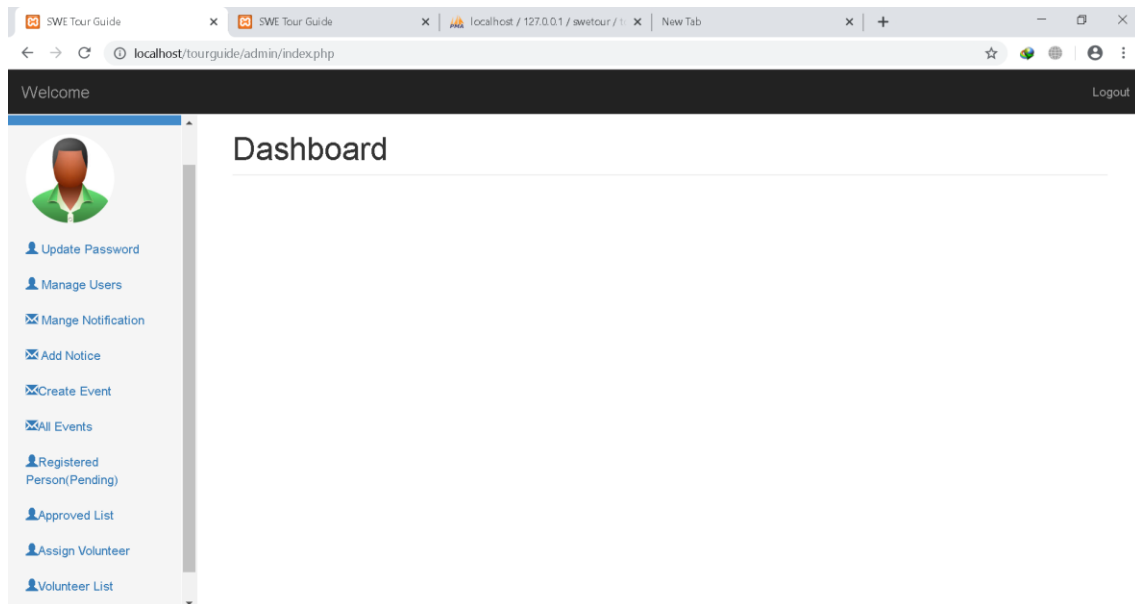


Figure 6.3: Admin Dashboard

### 6.3.1 Manage User

Admin here will be able to see the user list and he has the right to remove a user

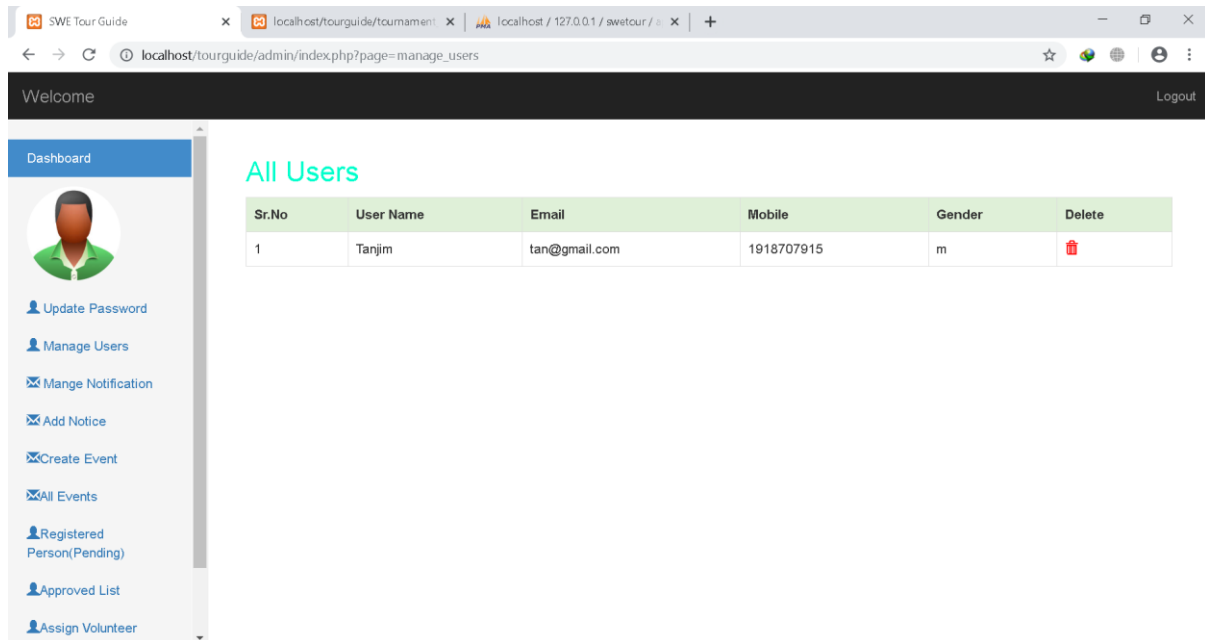


Figure 6.4: Manage User

### 6.3.2 Manage User

Admin here will be able to see the user list and he has the right to delete a user. When he tries to delete a user it will show an warning message to confirm.

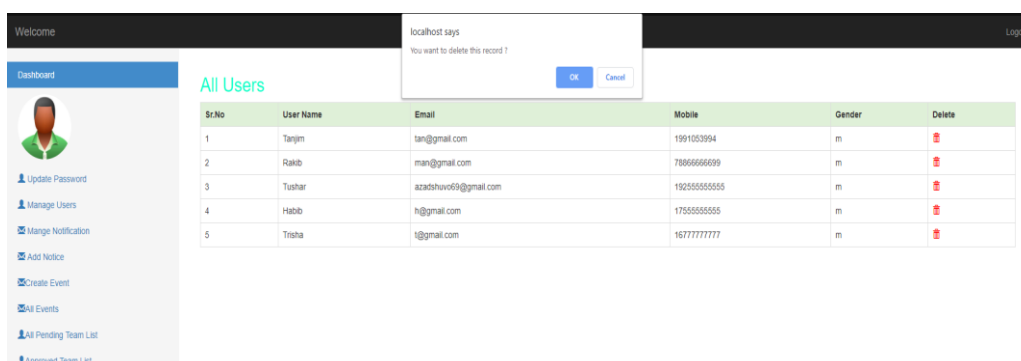


Figure 6.5: Delete User

### 6.3.3 Create Event:

Admin can create event by providing the valid information through the page.

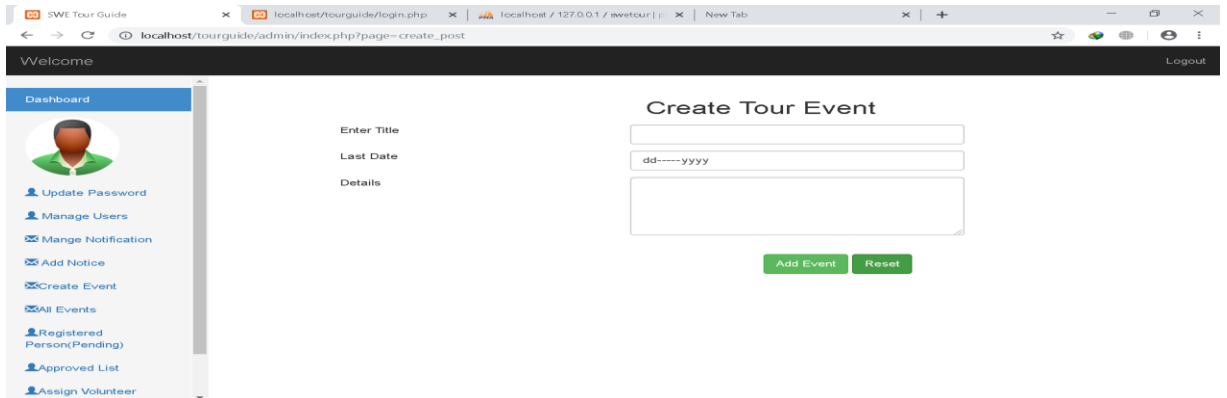


Figure 6.6: Create Event

### 6.3.4 All Event:

Admin can see the all created event from this page.

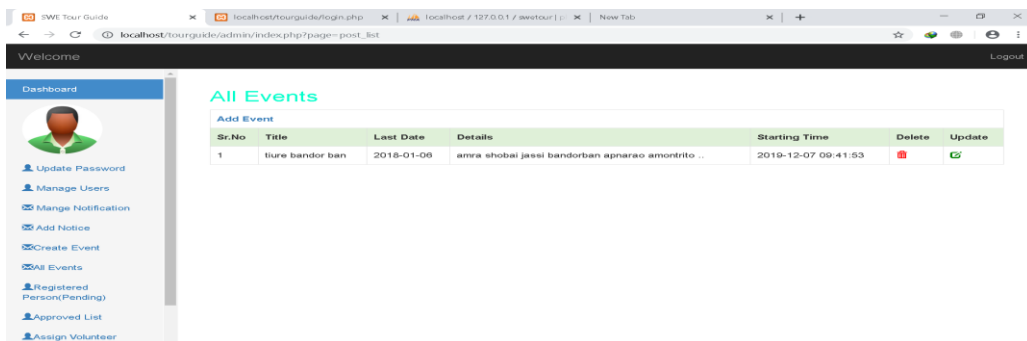


Figure 6.7: All Event

### 6.3.5 Update Event:

An event can be updatable by the admin through the page. In the page of all event list an update sign is given.

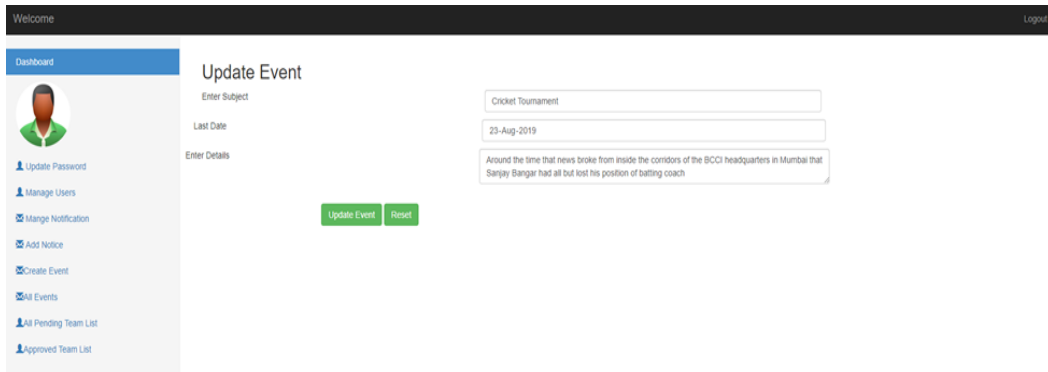


Figure 6.8: Update Event

### 6.3.6 Delete Event:

An event can be deletable by the admin through the page. In the page of all event list a delete sign is given.

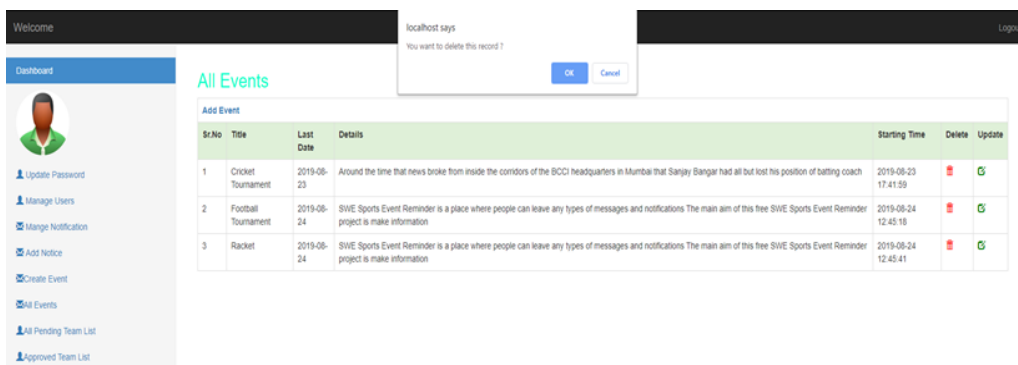


Figure 6.9: Delete Event



### 6.3.7 Create Notice:

Here admin can create a notice and can be able to send the notice to the specific user of the system.

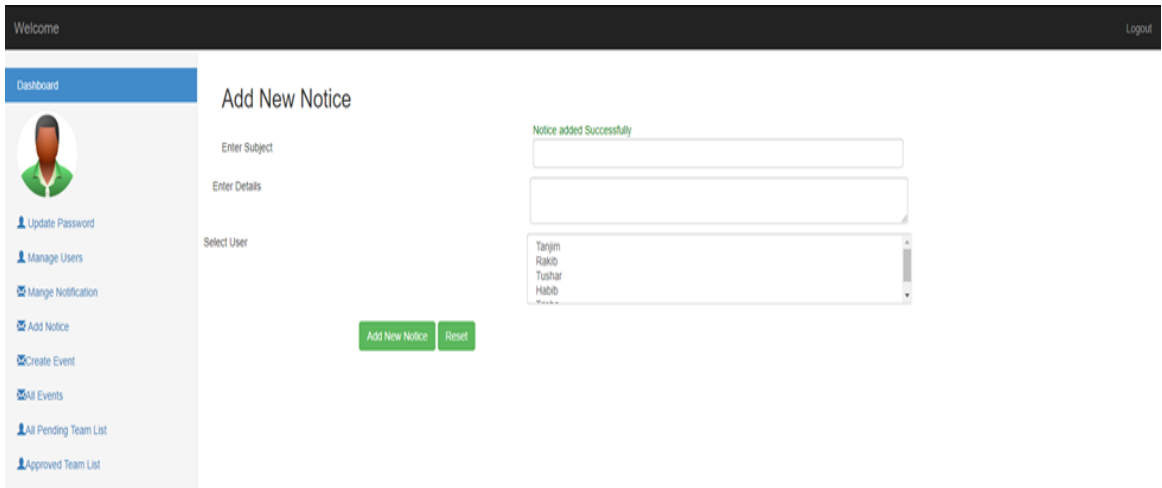


Figure 6.10: Create Notice

### 6.3.8 All Notice:

Here admin can see all the notice that he has been sent and manage it through update or delete.

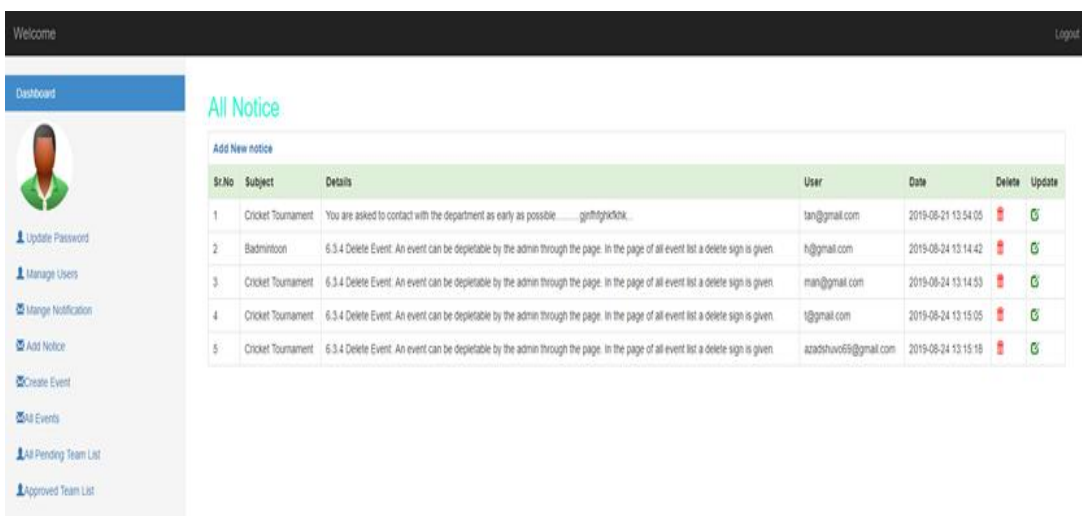


Figure 6.11: All Notice

### 6.3.9 Update Notice:

Here admin update any notice.

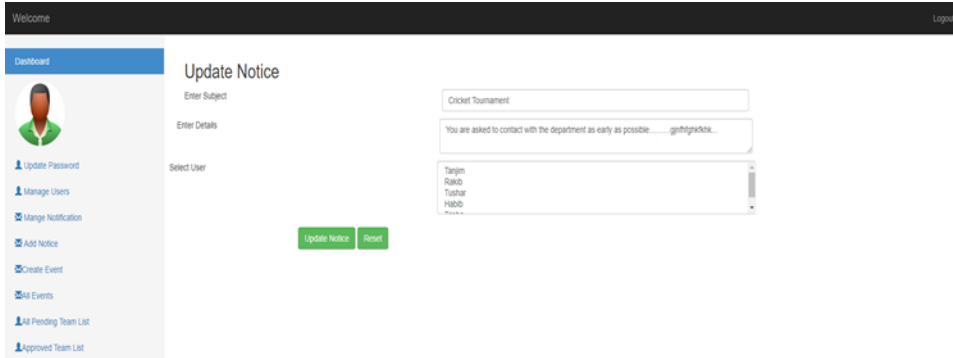


Figure 6.12: Update Notice

### 6.3.10 All Pending User:

Here admin can see the list of registered student who are waiting for approval. From here admin can approve a user or remove a user. When admin approve it, it will go the approved list.

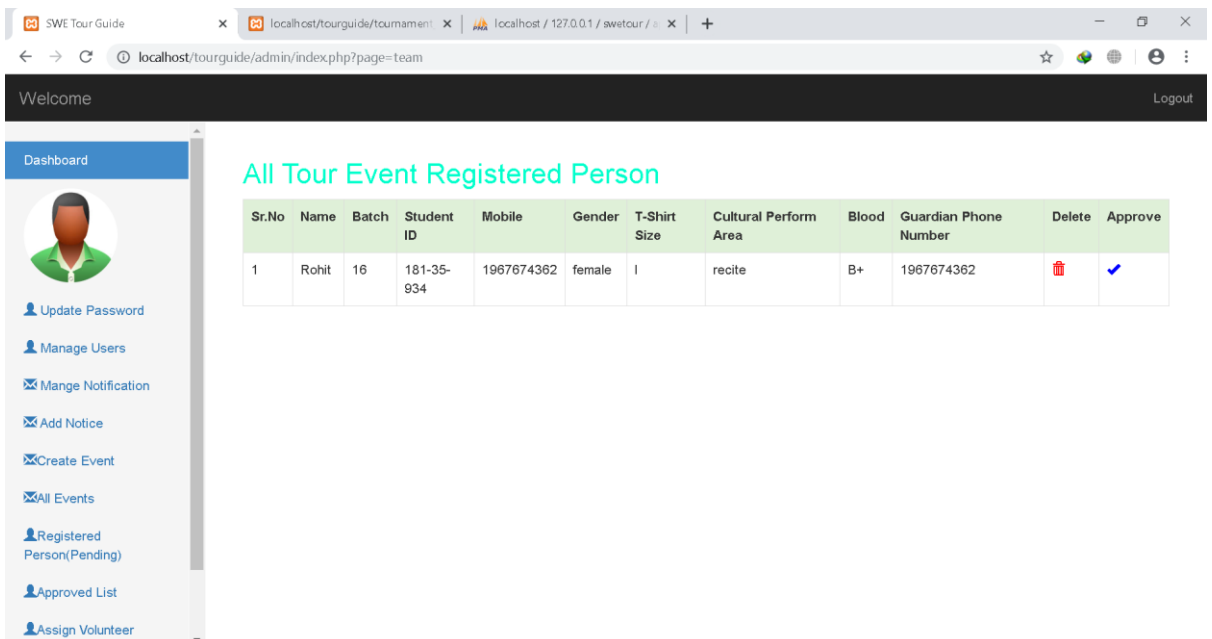


Figure 6.13: Pending user List

### 6.3.11 Approved User List:

Here admin can see the list of approved users and admin has the right to delete user from the approved list.

the approved user list.

Dashboard

Welcome Logout

#### All Approved Registered Person(Male)

Sr.No	Name	Batch	Student ID	Mobile	Gender	T-Shirt Size	Cultural Perform Area	Blood	Guardian Phone Number	Delete
1	MAHMUDA ZONI	25	181-35-936	1967674362	male	I	recite	B+	1967674362	

#### All Approved Registered Person(Female)

Sr.No	Name	Batch	Student ID	Mobile	Gender	T-Shirt Size	Cultural Perform Area	Blood	Guardian Phone Number	Delete
1	Jenny	25	181-35-934	1967674362	female	I	recite	B+	1967674362	

- Update Password
- Manage Users
- Manage Notification
- Add Notice
- Create Event
- All Events
- Registered Person(Pending)
- Approved List
- Assign Volunteer

Figure 6.14: Approved Users List

### 6.3.12 Update Password:

Here admin can update his password by providing the valid information.

Welcome Logout

Dashboard

#### Update Password

Enter Your Old

Enter Your New Password

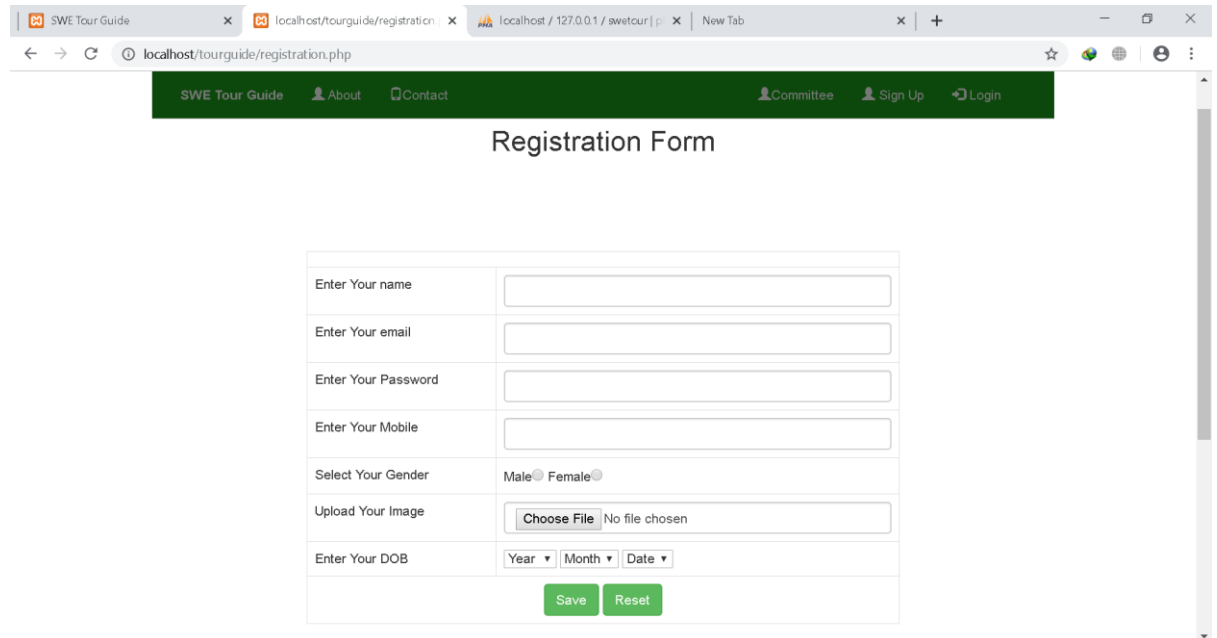
Enter Your Confirm Password

- Update Password
- Manage Users
- Manage Notification
- Add Notice
- Create Event
- All Events
- All Pending Team List
- Approved Team List

Figure 6.15: Update Password

## 6.4. User Registration:

Here a user can create his account through registration by providing the valid information.



The screenshot shows a web browser window with the URL `localhost/tourguide/registration.php`. The page has a green header with navigation links: "SWE Tour Guide", "About", "Contact", "Committee", "Sign Up", and "Login". The main heading is "Registration Form". The form contains the following fields:

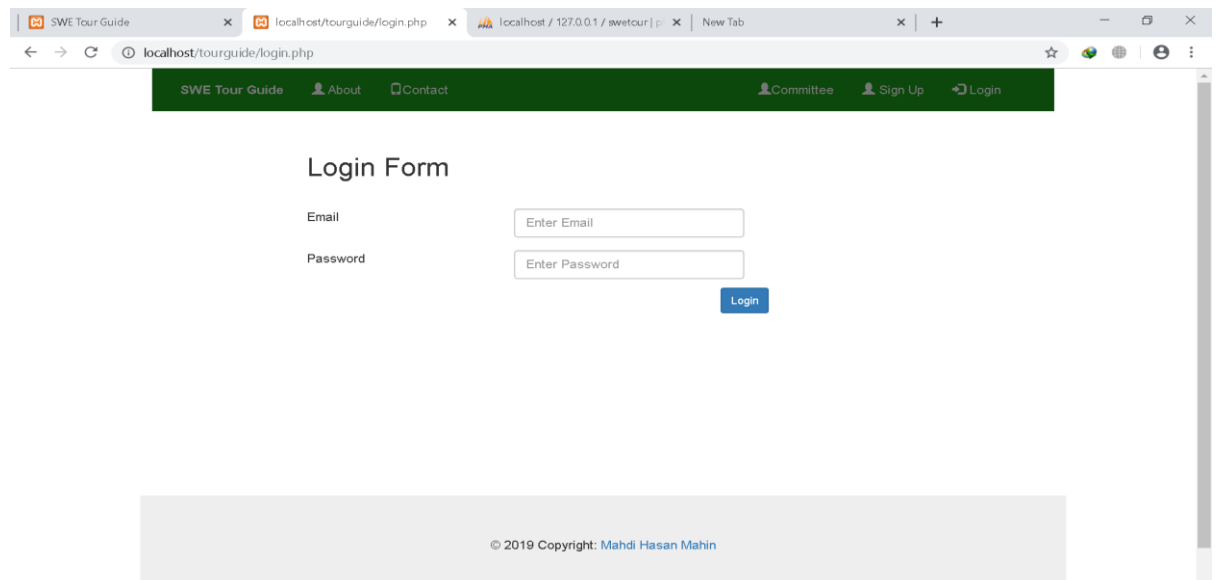
- Enter Your name:
- Enter Your email:
- Enter Your Password:
- Enter Your Mobile:
- Select Your Gender: Male  Female
- Upload Your Image:  No file chosen
- Enter Your DOB: Year  Month  Date

At the bottom of the form are two green buttons: "Save" and "Reset".

Figure 6.16: User Registration

### 6.4.1 User Login:

Here a user can login to the system if he has an account



The screenshot shows a web browser window with the URL `localhost/tourguide/login.php`. The page has a green header with navigation links: "SWE Tour Guide", "About", "Contact", "Committee", "Sign Up", and "Login". The main heading is "Login Form". The form contains the following fields:

- Email:
- Password:

Below the password field is a blue "Login" button. At the bottom of the page, there is a footer: "© 2019 Copyright: Mahdi Hasan Mahin".

Figure 6.17: User Login

### 6.4.3 User Profile:

This is the user profile. When a user creates an account that's time a profile is created.

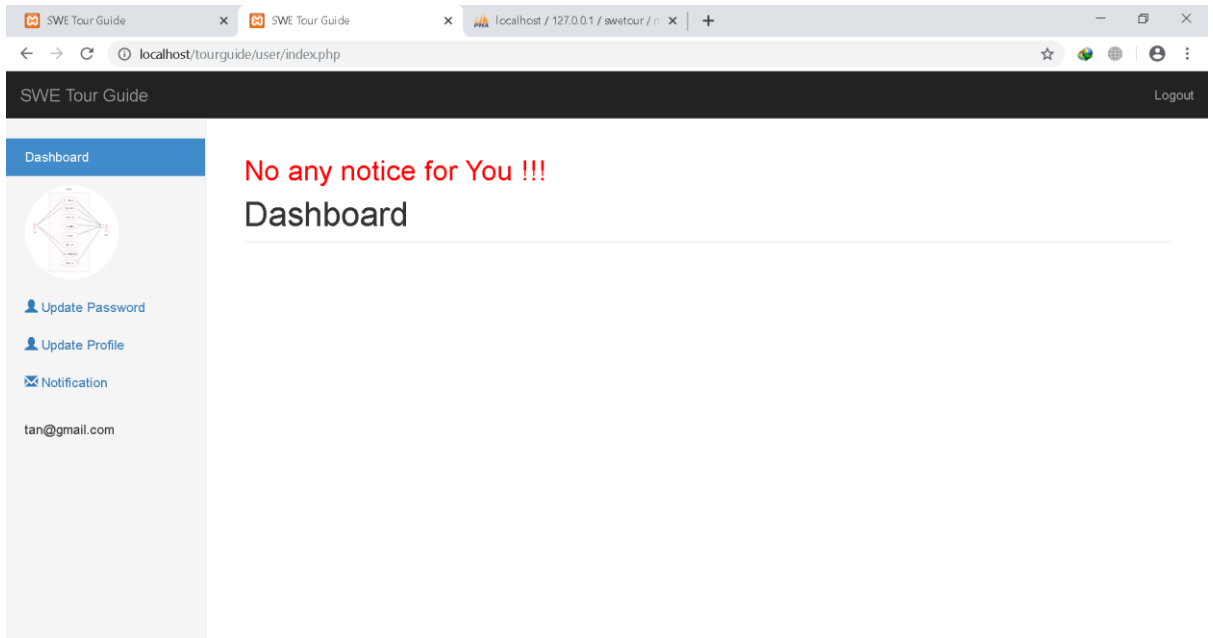


Figure 6.18: User Profile

### 6.4.4 Update Profile:

User can update his personal information.

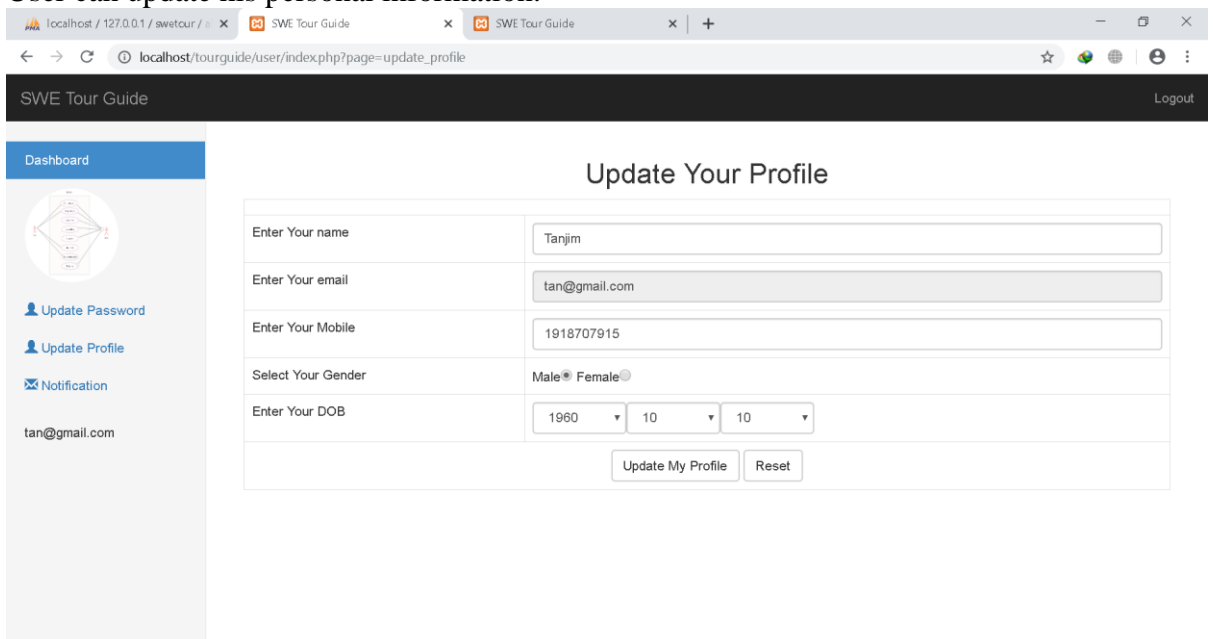


Figure 6.19: Update Profile

### 6.4.5 Check Notice:

User can see the personal notice.

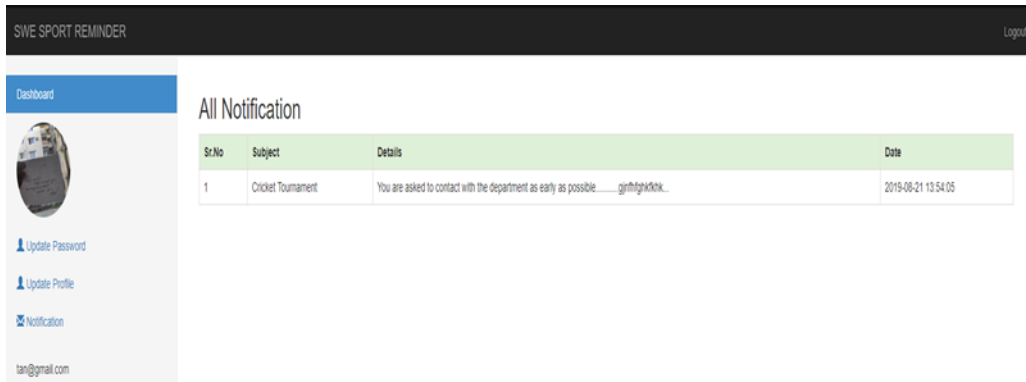


Figure 6.20: Check Notice

### 6.4.6 Update Password:

User can update his old password by providing the previous one.

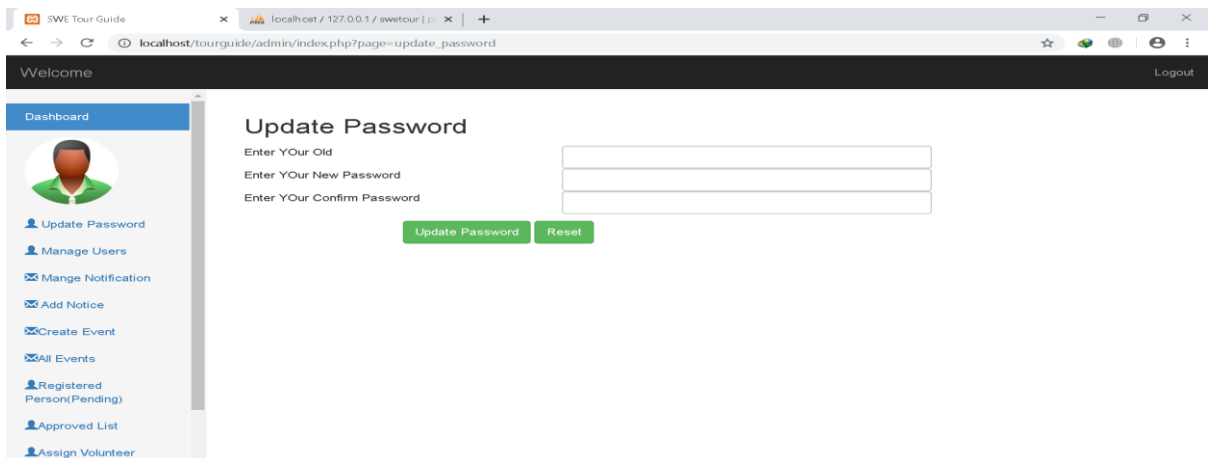


Figure 6.21: Update Password

## 6.5 Committee:

This page describes the persons who are connect to the Tour guide and manage the Tour.

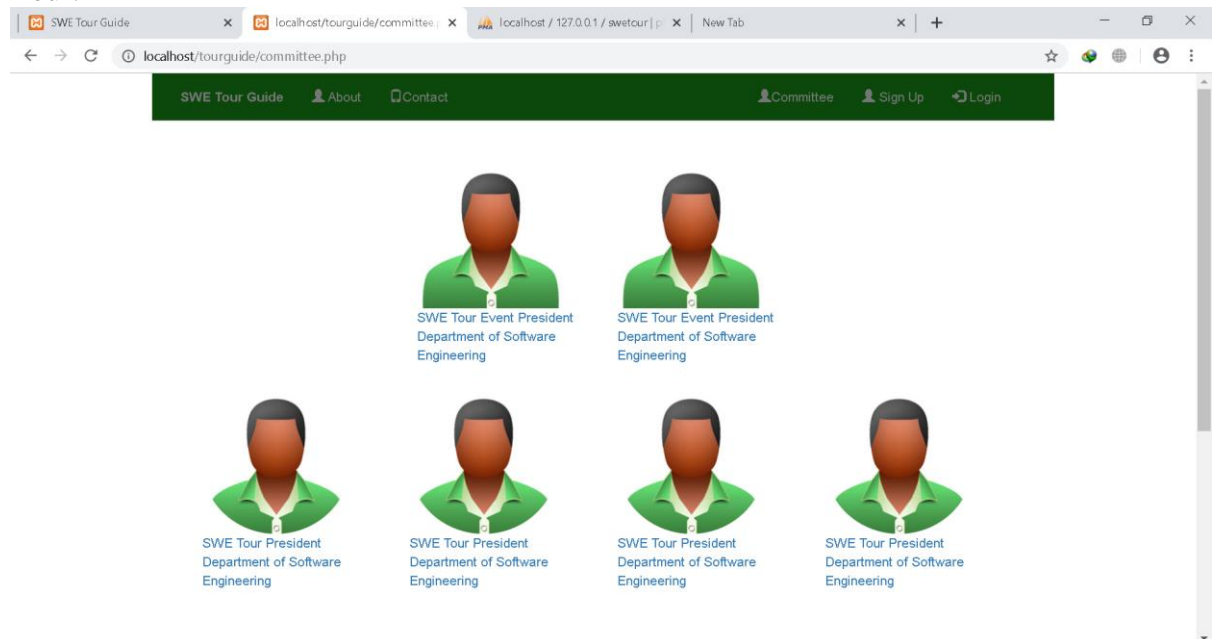
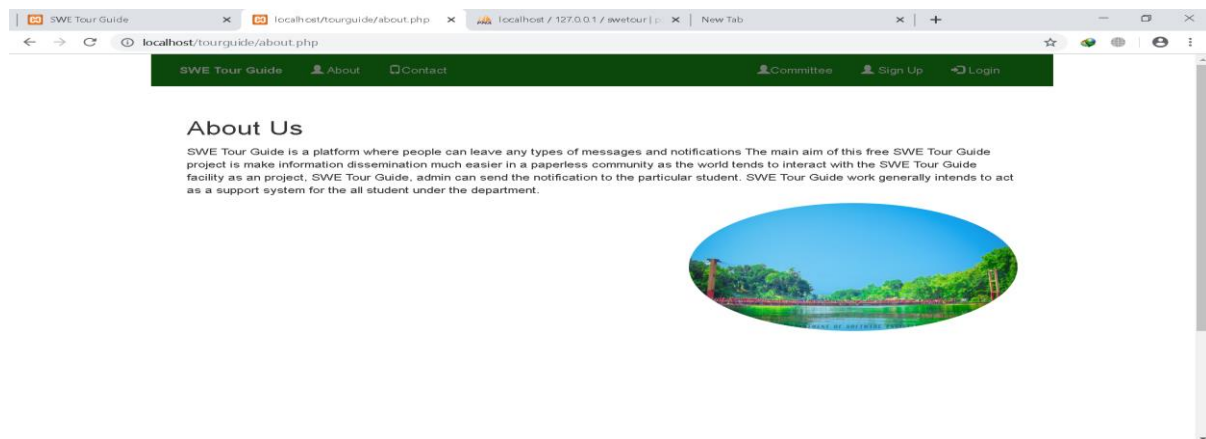


Figure 6.22: Committee

## 6.5 About:

People will get informed about our activities through this page. Figure 6.23: About



# CHAPTER 7

## PROJECT SUMMARY

### 7.1 Critical Evolution:

Requirement engineering was a challenging part. Our system is not available for all department till now.

### 7.2 Limitations:

Every system has some limitations. Because a system is not built for all services or not for all over the world. Like that our system has some limitations given below.

- Our project is based on only for our department Tour event perspective, rules and regulations.
- Our project is only for Software Department of DIU
- This is a beta version.

### 7.3 Obstacles & Achievements:

We have to take challenge to think or do something new. Because we have to face some difficulties as obstacles to do something new. When we can complete the new thing then we can learn something new and can overcome the obstacles then these is our achievements.

- ✓ Requirement analysis is a great challenge.
- ✓ After completing this project, we have learned many things.

### 7.4 Future Scope:

We cannot implement some important feature in our app. But our exertion won't be stop. We want to add some more features that will bring benefit to the SWE Sport Event Reminder and our faculty. In future we will provide an android app that will fetch data from the website and informed the user. Then the players /spectators will never miss any event or notice.



## **7.5 Discussion and Conclusion:**

Now-a-days maintain security in an area is a big problem. Our application is based on some data. In virtual world data is asset. SWE Sport Event reminder will remind its user about the event schedule.

If the user fills free to use the system and accept our work in a cordial heart then the implementation of the application will successful.

## REFERENCES

- [1]. Block Diagram, <https://www.smartdraw.com/block-diagram>, last accessed: Nov 23, 2018.
- [2]. Activity Diagram, <https://www.smartdraw.com/activity-diagram>, last accessed: Nov 24, 2018.
- [3]. UML Sequence Diagram Tutorial, <https://www.lucidchart.com/pages/uml-sequence-diagram>, last accessed: Nov 25, 2018.
- [4]. 7 software testing Principles: Learn with case study, <https://www.guru99.com/software-testing-seven-principles.html>, last accessed: Nov 20, 2018.
- [5]. Learn UI Design, <https://learnui.design/>, last accessed: Dec 8, 2018.