

Project Name: ScoreZone (Football Tournament Manager)

Branch PG-DAC Sept-2022

Documentation On

**“ScoreZone (Football Tournament Manager)”**

PG-DAC SEPT 2022

**Guided By: Bakul Joshi**

Submitted By:

**Group No: 05**

* **Parth Patil 220943020062**
* **Shantanu Patil 220943020066**
* **Soham Ghatpande 220943020100**

**Table of Contents**

**1. Introduction 4**

1.1 Document Purpose **4**

1.2 Project Background **4**

1.3 Aim & Objectives **4**

**2. Business Requirements Overview**…………………………………………………………**5**

**3.Functional Requirements** **6**

3.1 Admin Module **6**

3.2 Tournament Manager Module **6**

3.3 Team Manager Module **6**

3.4 Match Updater Module **6**

3.5 Viewer Module **6**

**4.** **Non-Functional Requirement………………………………………………………………7**

**5. Use Case Diagram 8**

5.1 Admin **6**

5.2 Tournament Manager **6**

5.3 Team Manager **6**

5.4 Match Updater **6**

5.5 Viewer **6**

**6. Database Design 12**

1 Users **12**

2 Tournaments **12**

3 Teams **12**

4 Matches **13**

5 Players **13**

6 User Types **13**

6 Tournament Teams **13**

7 Match Updater Tournament **13**

**7. E-R Diagram………………………………………………………………………………………14**

**8. Snapshots 15**

**9. Conclusion 20**

**List of Figures**

Use Case Diagrams 17

Fig 1 Admin 8

Fig 2 Student 9

Fig 3 Teacher 10

Fig 4 ER Diagram 14

1. **Introduction:**
   1. **Document Purpose:**

This document communicates the business requirements and scope for developing **ScoreZone**. The scope of this document is to define the functional and nonfunctional requirements, business rules, and other constraints.

## Project Background:

Currently, many football teams in many cities of India play tournaments organized by various organizers. These organizers do not have a proper and efficient way of managing these tournaments. Many talented people do not receive the recognition they deserve because tournament organizers lack the resources to keep track of the results.

Instead of having the benefits of a digital world where they can manage all of their data digitally, these organizers have to do a lot of paperwork to keep the information related to these tournaments.

.

## Aim & Objectives:

The main objective of this project is to build a website for these organizers, who belong to various cities, to organize tournaments in their areas by managing tournaments through our web application, and they can also manage and maintain all the data related to these tournaments. This will also allow the viewers to view the live scores, past scores, browse through different tournaments and players. Team Managers also get a platform to represent their teams in various tournaments.

1. **Business Requirements Overview:**

* Score Zone is a public web application.
* There are mainly 5 types of users:

1. Admin
2. Tournament Manager
3. Match Updater
4. Team Manager
5. Viewers
6. Tournament Organizer (Virtual User)

* Tournament managers creates a tournament on their portal, which will be visible to all the team managers.
* Team managers get data of upcoming tournaments where they can register their teams.
* Tournament managers get the team details submitted by the team managers.
* The tournament manager is in charge of choosing the date and location of the match and generating matches of the registered teams.
* The Match Updater keep the scores up to date for each match and declare the result of the matches.
* An administrator looks after Score Zone.

# Functional Requirements Overview:

Score Zone consists of five modules, described below

* 1. Admin Module
  2. Tournament Manager Module
  3. Match Updater Module
  4. Team Manager Module
  5. Viewer Module

**1. Admin Module**

* + Admin has all the rights to modify and access all the functions in Score Zone after successful login authentication.
  + Authorization of new Tournament Managers and new Team Managers after their registration is complete.
  + Adding new venues for different tournaments on proper communication with different Tournament Organizers.
  + Report generation of each tournament when completed.

**2. Tournament Manager Module**

* + Tournament Manager requests to create(register) his or her own account.

# To generate tournament by providing necessary details.

# **Tournament Level Tasks:**

* + TM generates matches for the teams that are registered in the current tournament.

# **Match Level Tasks:**

* + TM sends fixture details like opposition team, date, time and venue to TEM before every match.
  + TM will create account for Match Updater.

**3. Match Updater Module**

* + MU is able to update the score of individual matches.
  + MU declares the result of every match.
  + MU update player performance after every match.

**4. Team Manager Module**

* + TEM requests to create(register) his or her own account.

# **Tournament Level Tasks:**

# To register team details (player information in the form of player name, player position, player age, etc.) in proper file format provided by the TM.

# To participate in the tournament scheduled by the TM.

# **Match Level Tasks:**

* + To view the fixtures scheduled by TM and selecting players for that particular fixture.
* To view other team’s data in the form of player stats, past results, upcoming matches.

**5. Viewer Module**

# **Tournament Level Features:**

# To view Team performance

* + To view player performance of different teams.
  + To view Team Fixtures of all teams participating in tournament.

# **Match Level Features:**

* + To view live scores of the ongoing matches.
  + Will allow the viewers to predict the winners during live matches.

1. **Non-Functional Requirement:**

* The website should use professional design, look and feel and color scheme.
* Being a public website, the site must follow general usability guidelines for menus, navigation, colors, links and other actions provided on the screens.
* The system should be designed in such a manner that user will be able to complete tasks in minimum number of steps.

.

## 5. Use-Case Diagram

**5.1 Admin:**

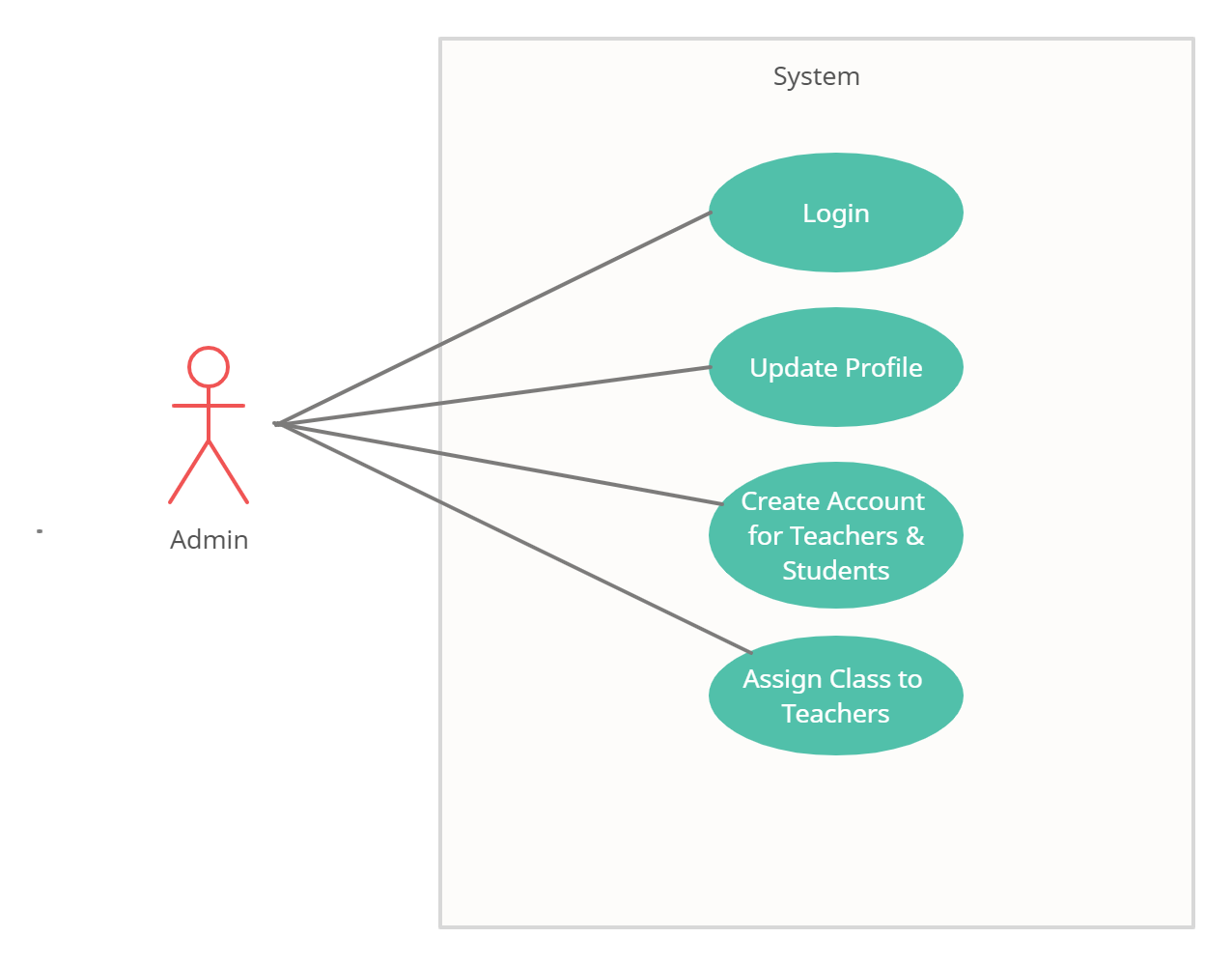
****

Fig. Use-Case Diagram for Admin

## 5.2 Student:

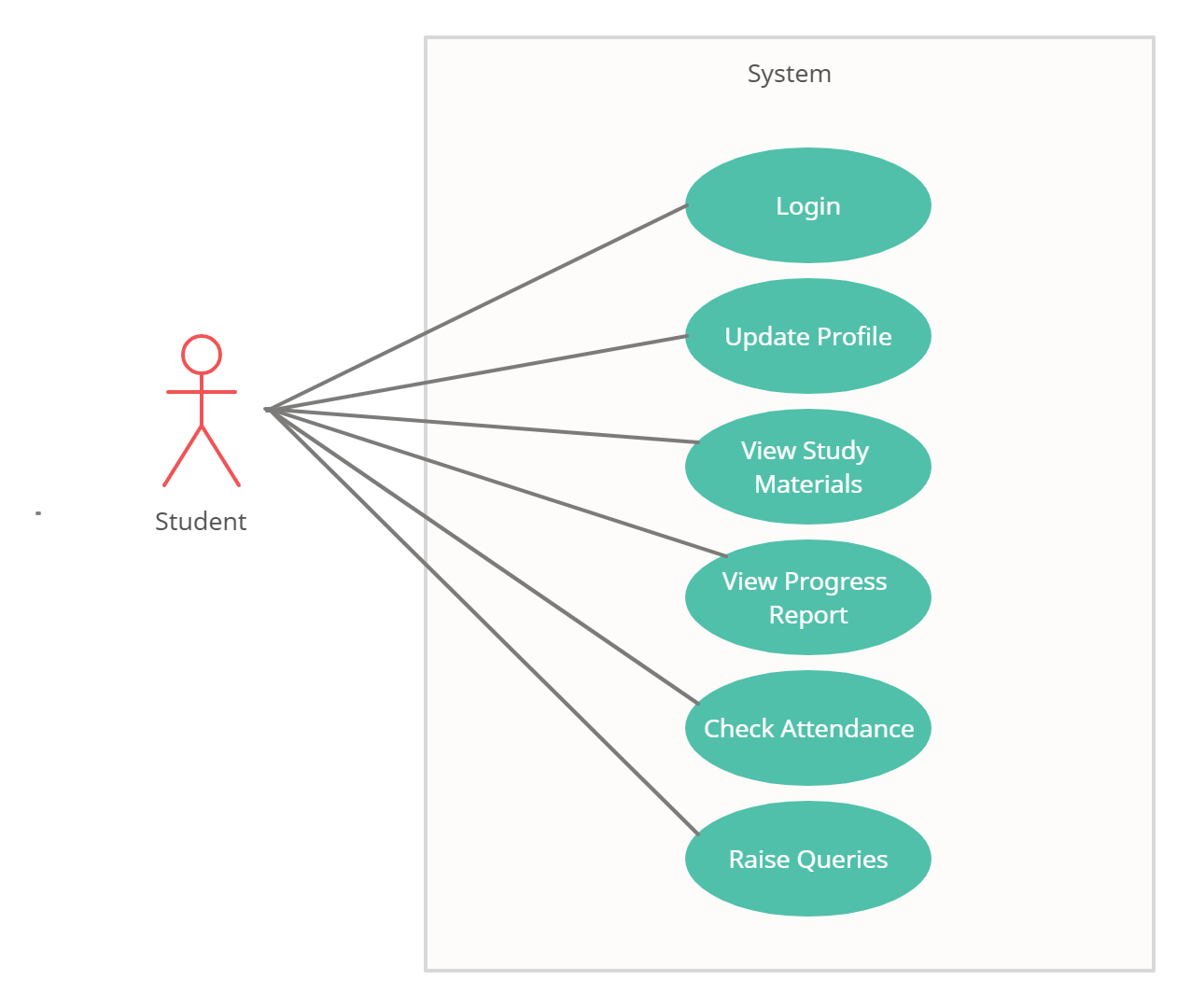
****

Fig. Use-Case Diagram for Student

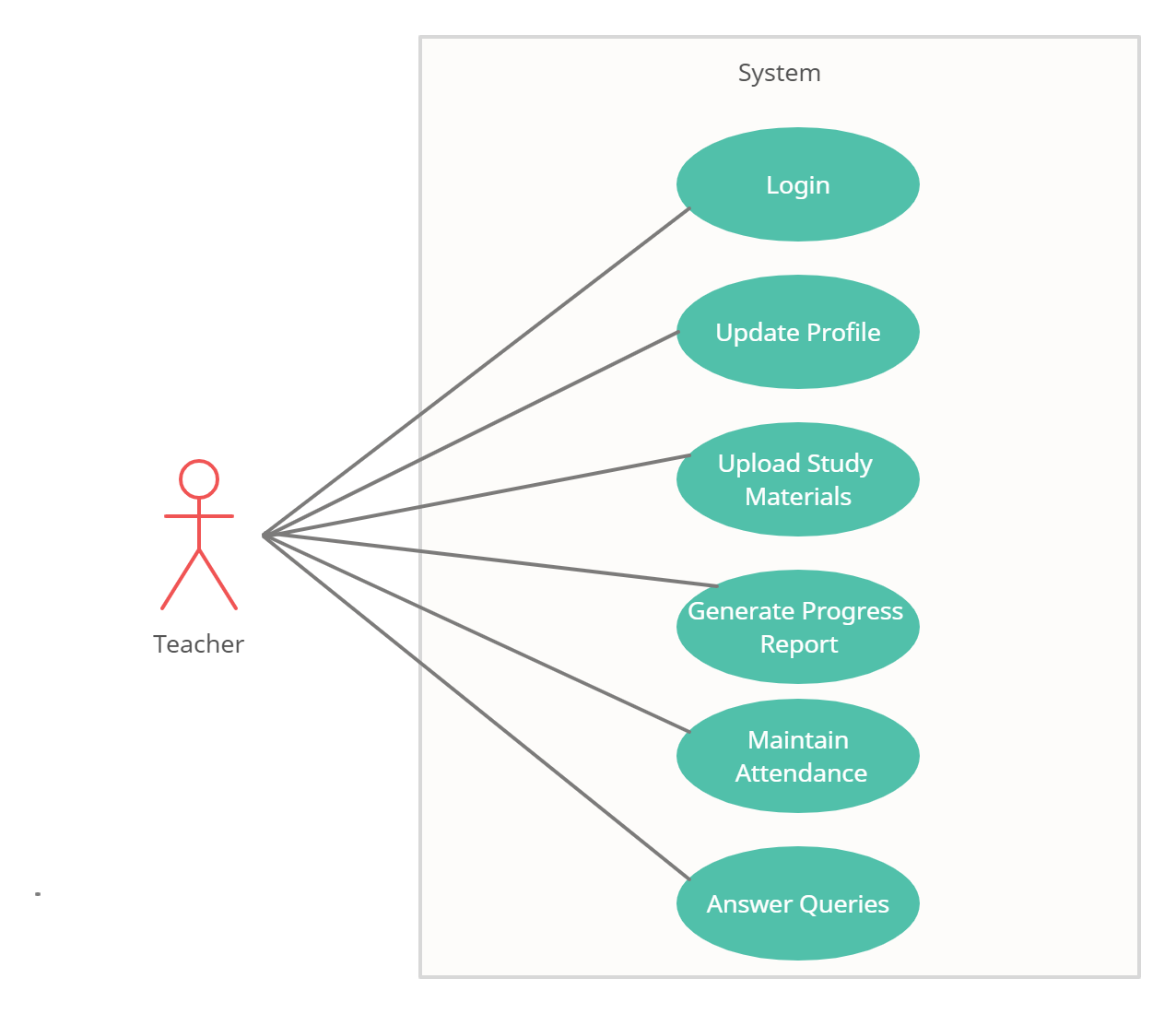
**5.3 Teacher:**  


Fig. Use-Case Diagram for Teacher

**6. Database Design:**

**1] student\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| s\_id | Integer | No | PK | AI | Student ID |
| user\_id | Integer | No |  |  | Student ID-2021001 |
| first\_name | Varchar(45) | No |  |  | First Name |
| middle\_name | Varchar(45) | No |  |  | Middle Name |
| last\_name | Varchar(45) | No |  |  | Last Name |
| email | Varchar(45) | No |  |  | Email ID of Parent |
| contact\_no | Varchar(45) | No |  |  | Contact No. of Parent |
| address | Varchar(100) | No |  |  | Permanent Address |
| dob | Date | No |  |  | Date of Birth |
| class\_id | Integer | No | FK |  | Class ID |

**2] teacher\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| t\_id | Integer | No | PK | AI | Teacher ID |
| user\_id | Integer |  |  |  | Teacher ID-1001 |
| first\_name | Varchar(45) | No |  |  | First Name |
| last\_name | Varchar(45) | No |  |  | Last Name |
| email | Varchar(45) | No |  |  | Email ID of Teacher |
| contact\_no | Varchar(45) | No |  |  | Contact no of Teacher |
| qualification | Varchar(45) | No |  |  | Qualification of Teacher |
| address | Varchar(100) | No |  |  | Address |

**3] login\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| login\_id | Integer | No | PK | AI | Login ID-1 |
| user\_id | Integer | No | FK |  | User ID |
| password | Varchar(45) | No |  |  | Password |
| user\_type | Varchar(45) | No |  |  | User Type |

**4] class\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| class\_id | Integer | No | PK |  | Class ID- 10,20….. |
| standard | Varchar(15) | No |  |  | Standard-I,II,III,IV…… |
| t\_id | Integer | No | FK |  | Teacher ID |

**5] subject\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| subject\_id | Integer | No | PK | AI | Subject ID-101,102….. |
| subject\_name | Varchar(45) | No |  |  | Subject Name |
| class\_id | Integer | No | FK |  | Class ID |

**6] chapter\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| c\_id | Integer | No | PK | AI | Chapter ID |
| chapter\_id | Integer |  |  |  | Chapter ID |
| chapter\_name | Varchar(45) | No |  |  | Chapter Name |
| content | Varchar(2000) | No |  |  | Chapter Content |
| subject\_id | Integer | No | FK |  | Subject ID |

**7] progress\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| progress\_id | Integer | No | PK | AI | Progress Report ID |
| comment | Varchar(45) | No |  |  | Comment/Suggestions |
| s\_id | Integer | No | FK |  | Student ID |

**8] marks\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| mark\_id | Integer | No | PK | AI | Marks ID |
| out\_of\_marks | Integer | No |  |  | Out of Marks |
| obtain\_marks | Integer | No |  |  | Obtain Marks |
| subject\_id | Integer | No | FK |  | Subject ID |
| progress\_id | Integer | No | FK |  | Progress ID |

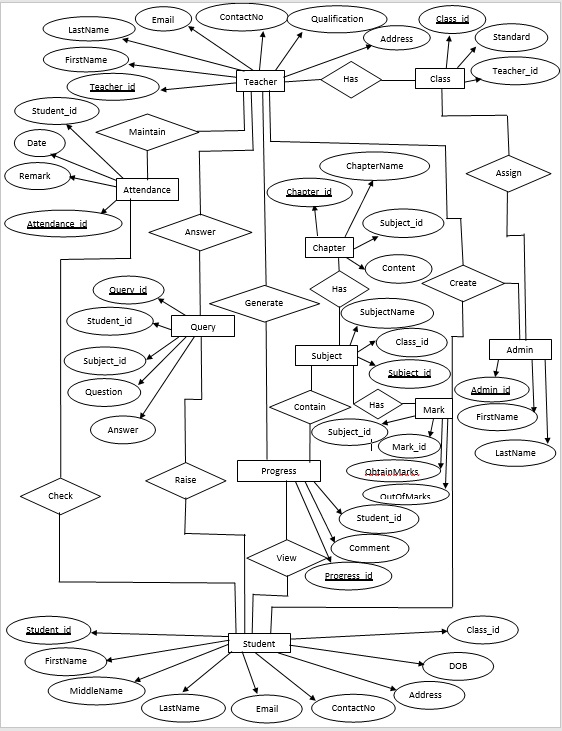
**9] attendance\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| attendance\_id | Integer | No | PK | AI | Attendance ID |
| date | Date | No |  |  | Date |
| remark | Varchar(10) | No |  |  | Remarks(P/A) |
| s\_id | Integer | No | FK |  | Student ID |

**10] query\_master table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| query\_id | Integer | No | PK | AI | Query ID |
| question | Varchar(100) | No |  |  | Question raise by Student |
| answer | Varchar(500) | No |  |  | Answer given by Teacher |
| subject\_id | Integer | No | FK |  | Subject ID |
| s\_id | Integer | No | FK |  | Student ID |

## 7. ER-Diagram:



**E-R diagram shows database of Paperless School Management System**

**8. Snapshots:**

**8.1 Home Page:**

Following snapshot shows the Home page for Paperless School Management System

****

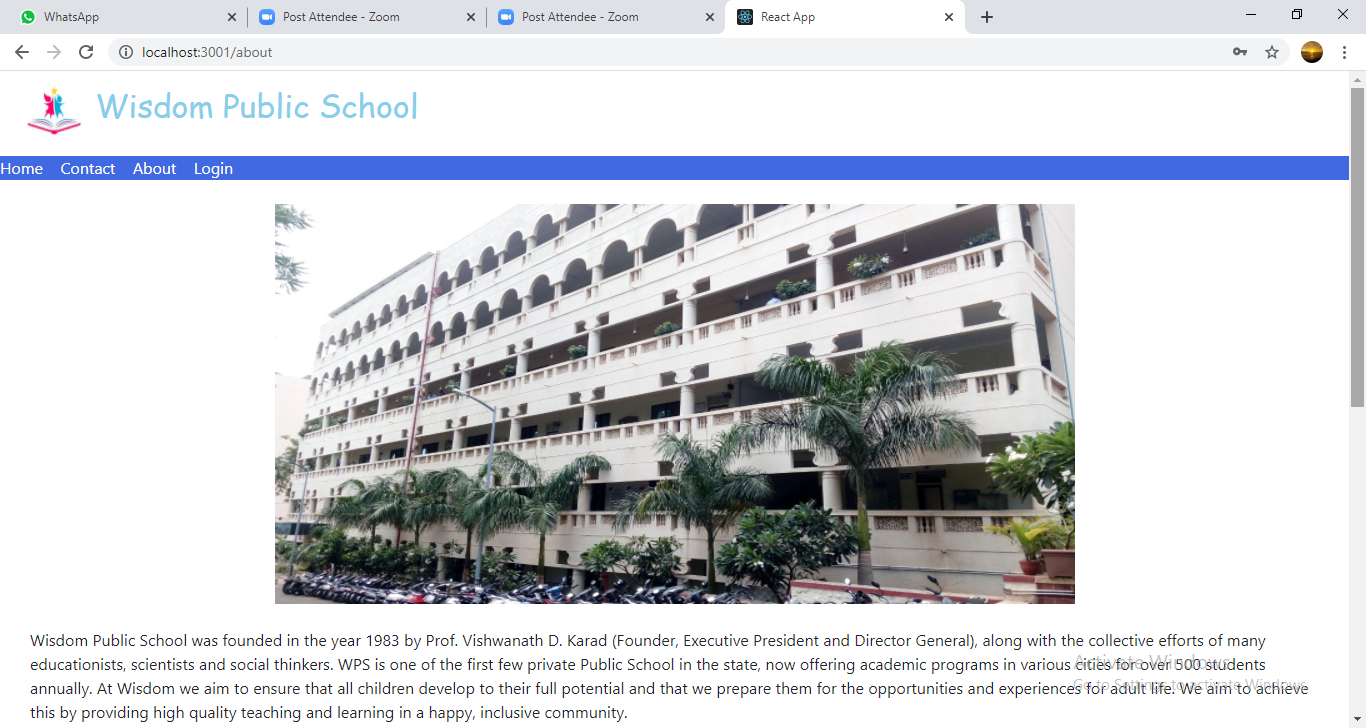
This page contains following controls

* Home
* Contact
* About
* Login Button

**8.2 Contact Page**

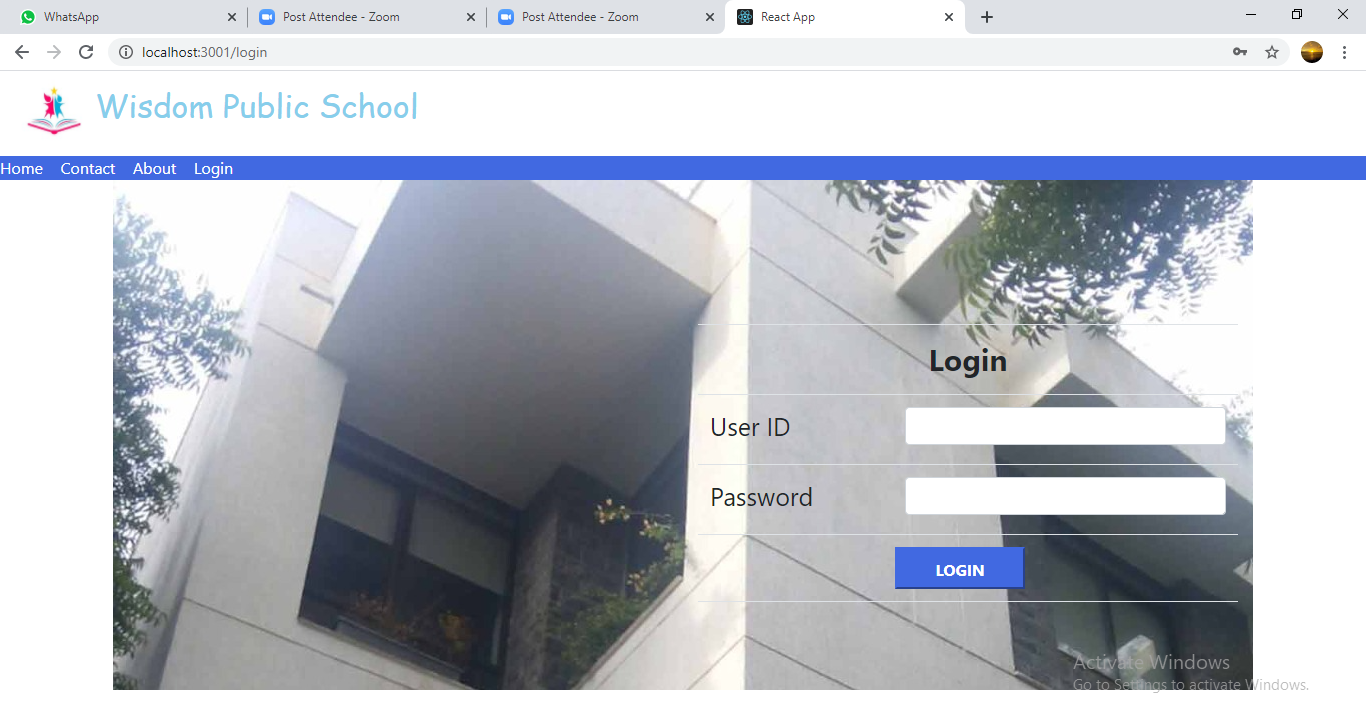


**8.3 About Page**



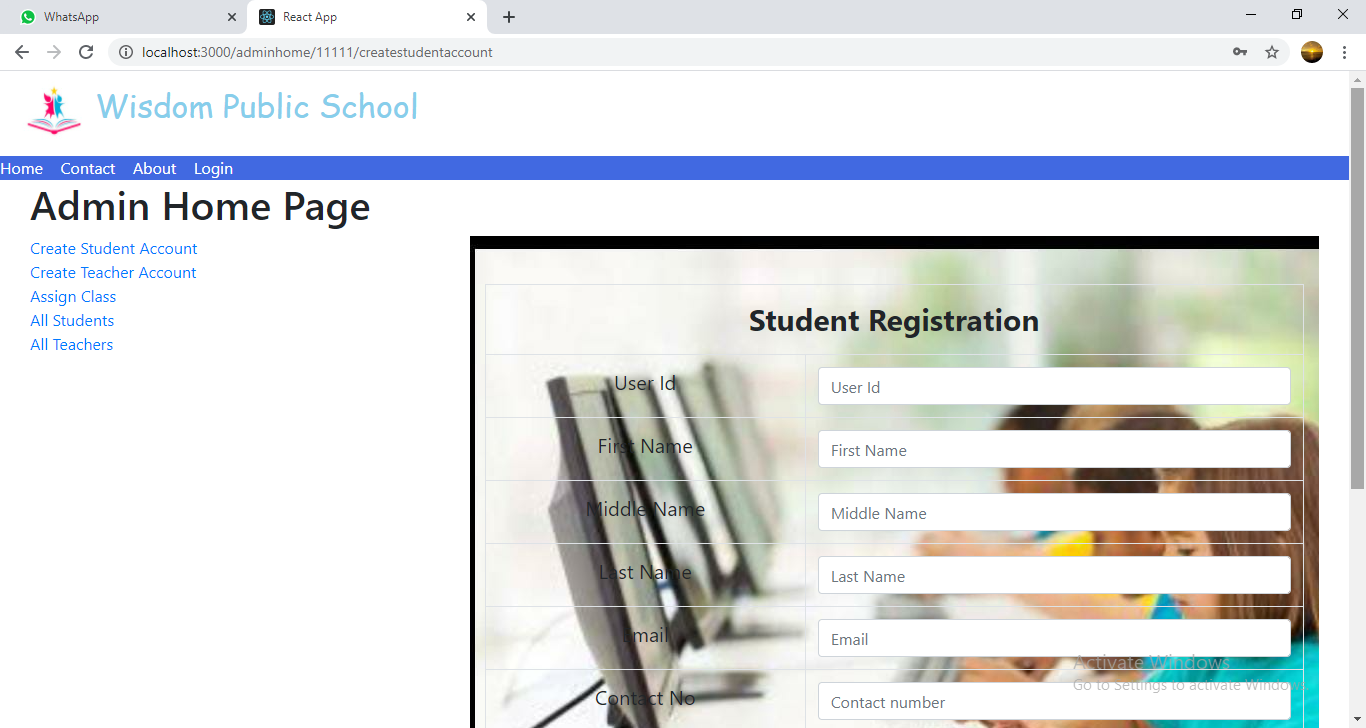
**8.4 Login Page**

Following snapshot shows the Login page for Paperless School Management System

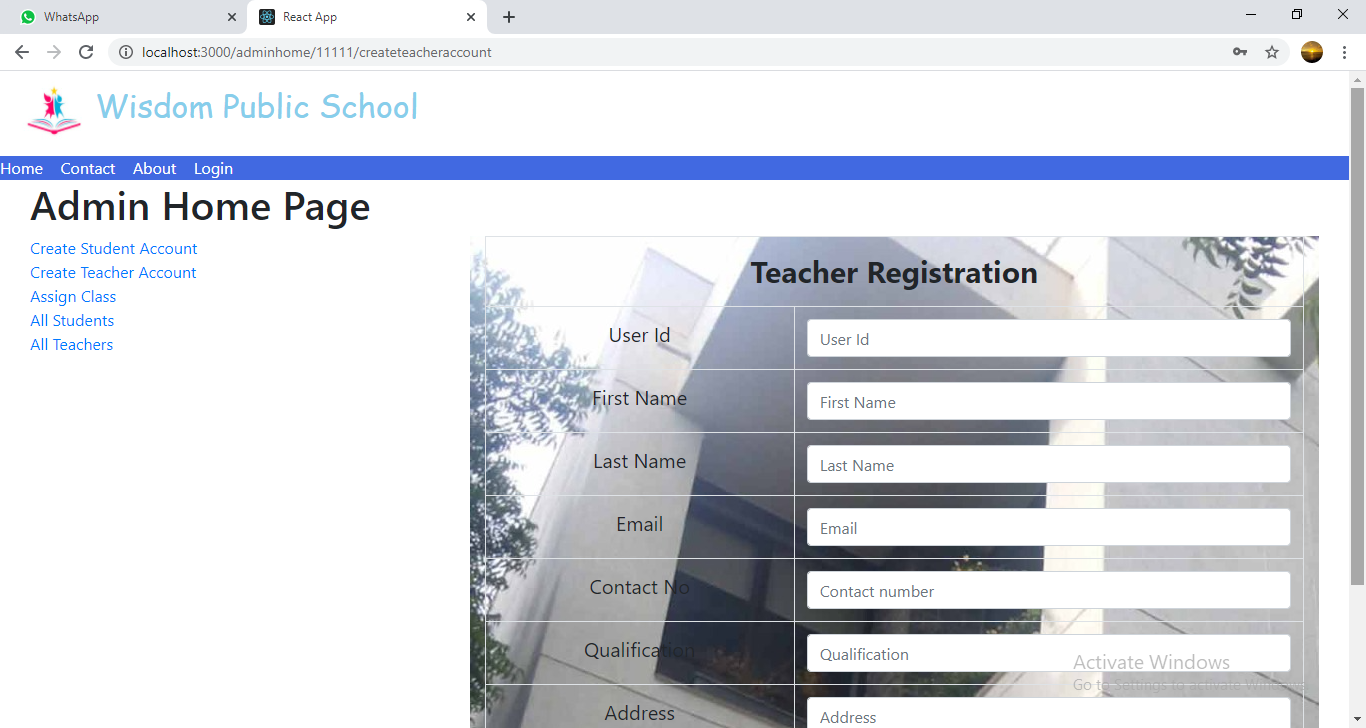


**8.5 Admin Home Page**

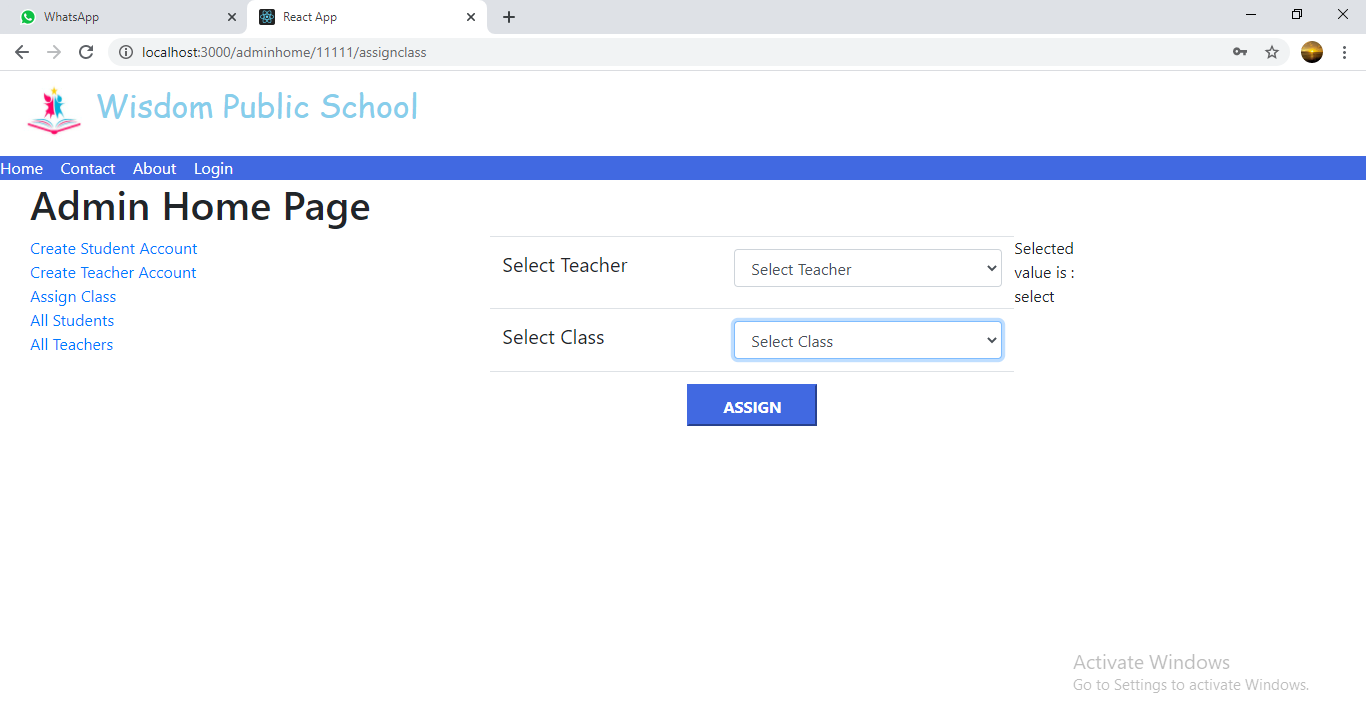
**8.5.1 Student Registration Page**



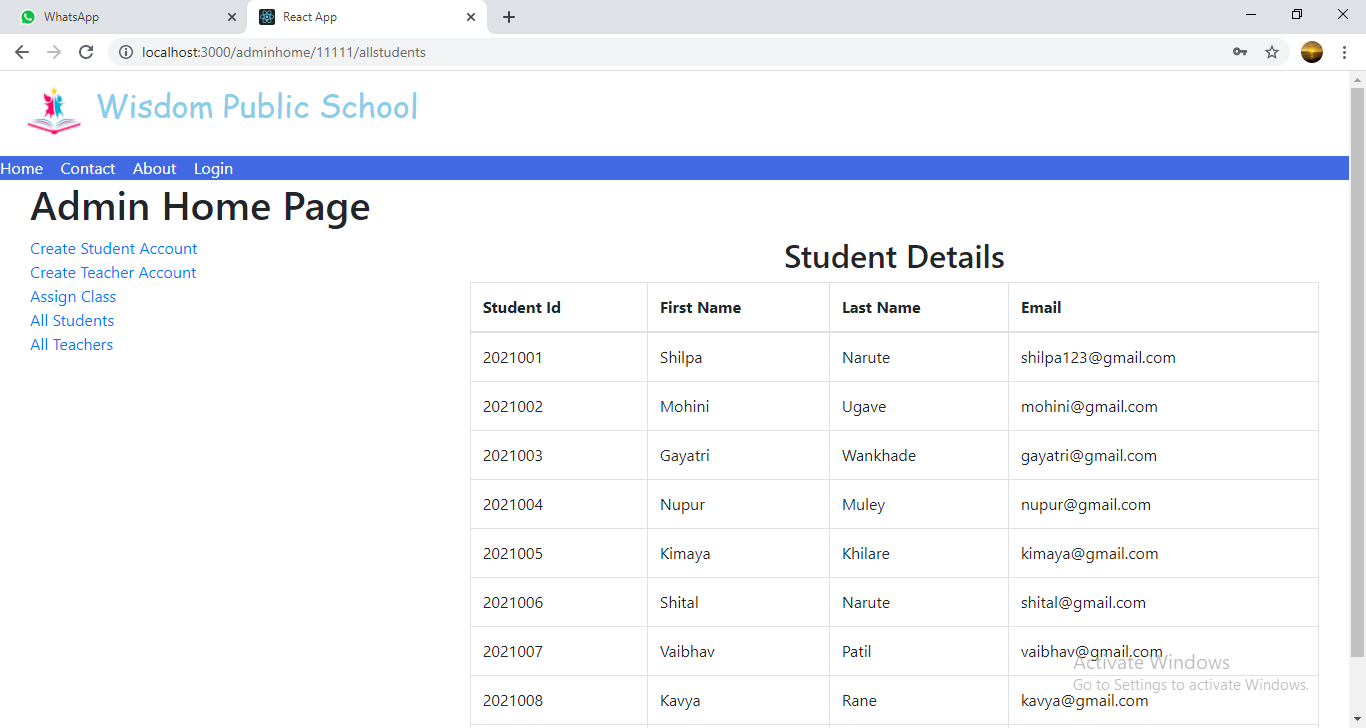
**8.5.2 Teacher Registration Page**



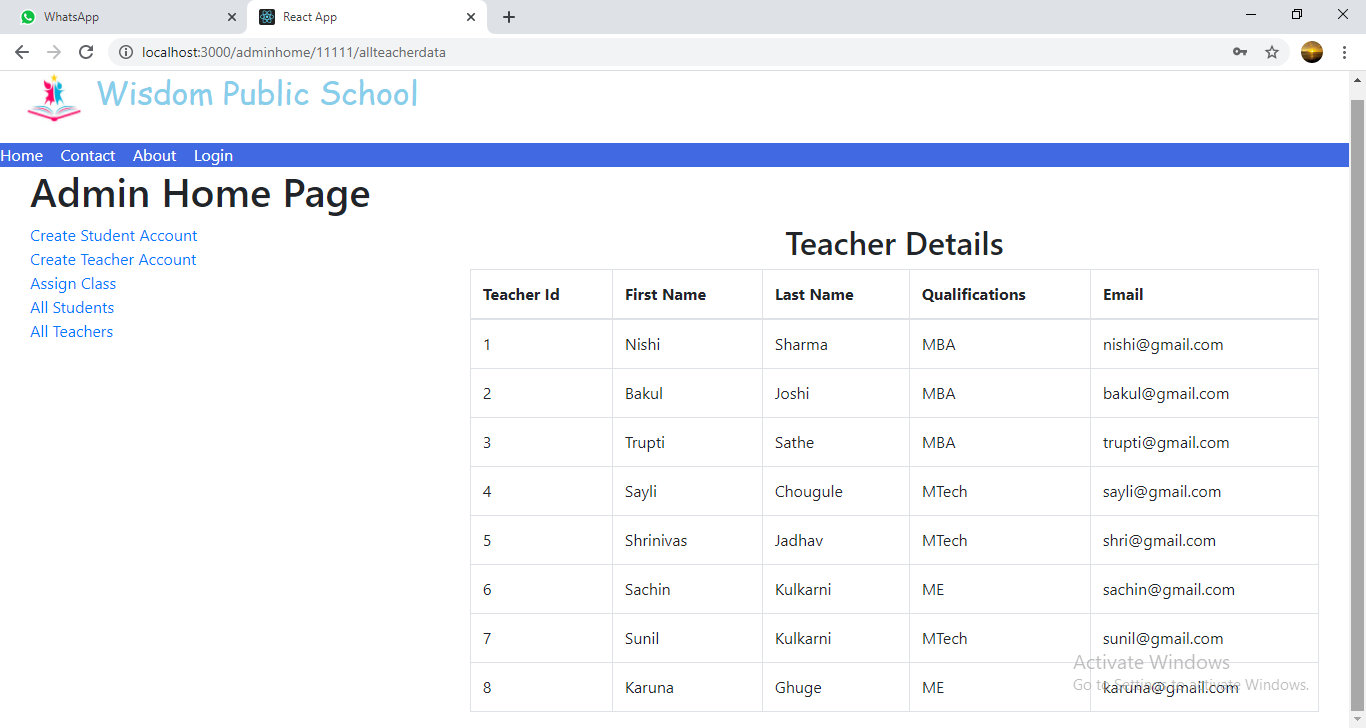
**8.5.3 Assign Class Page**



**8.5.4 All Students List**

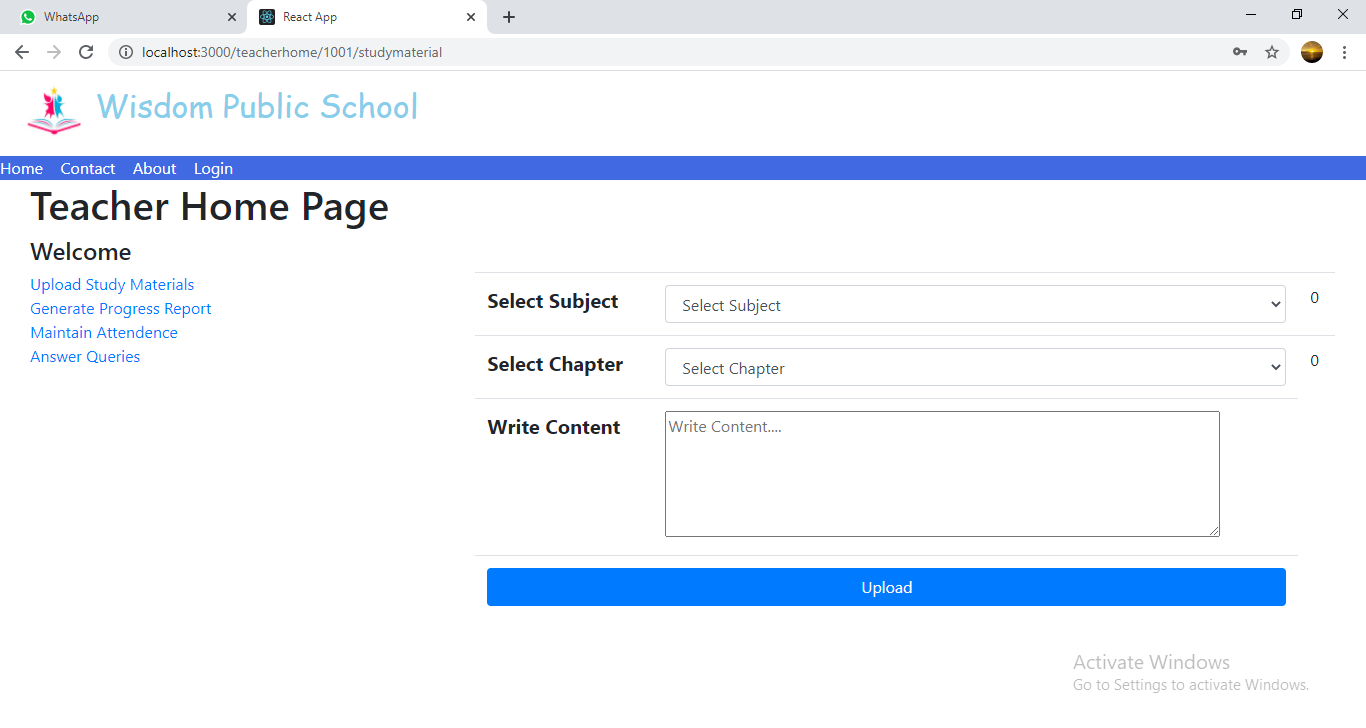


**8.5.5 All Teachers List**

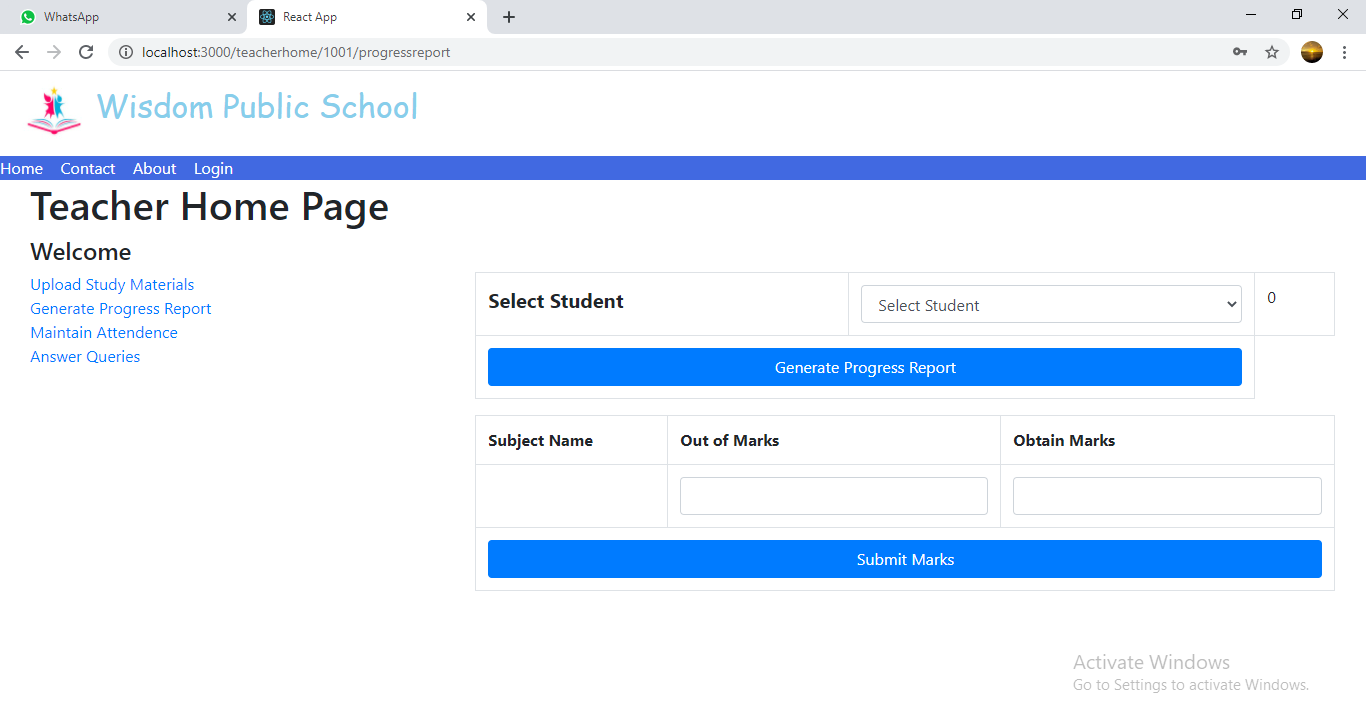


**8.6 Teacher Home Page**

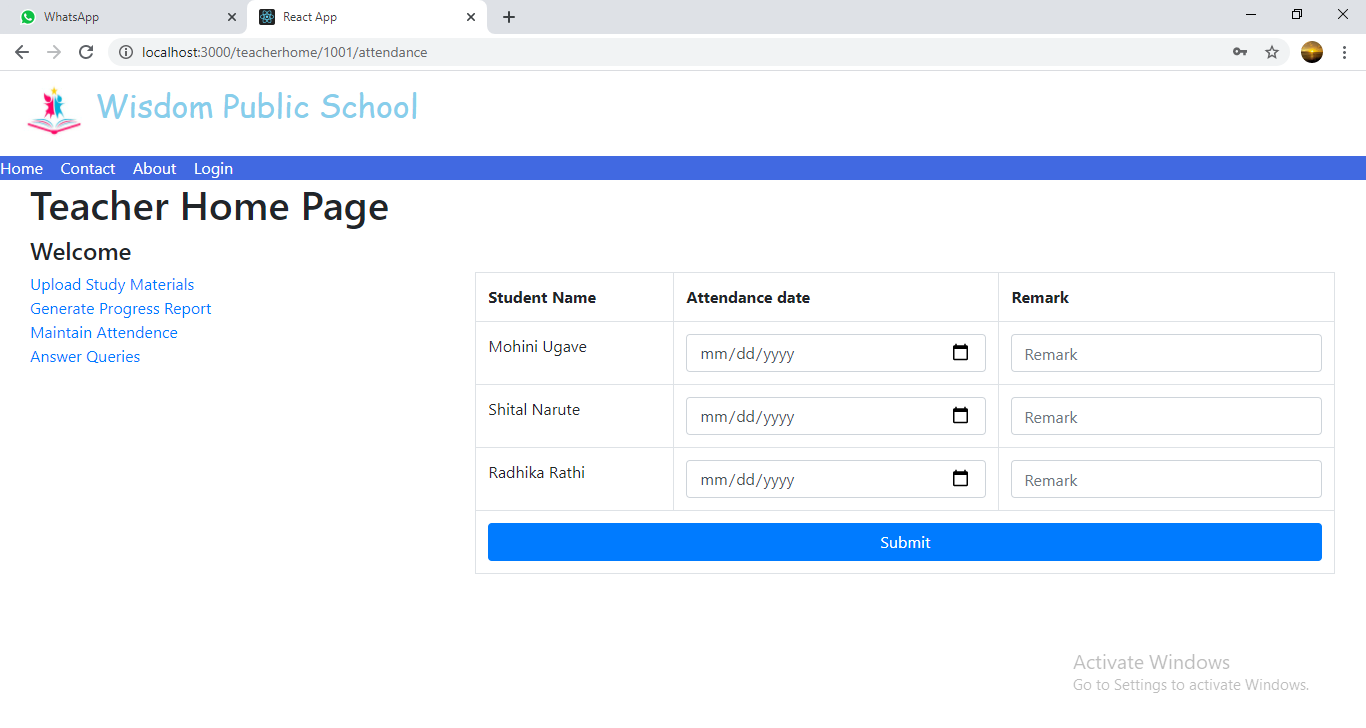
**8.6.1 Upload Study Material Page**

****

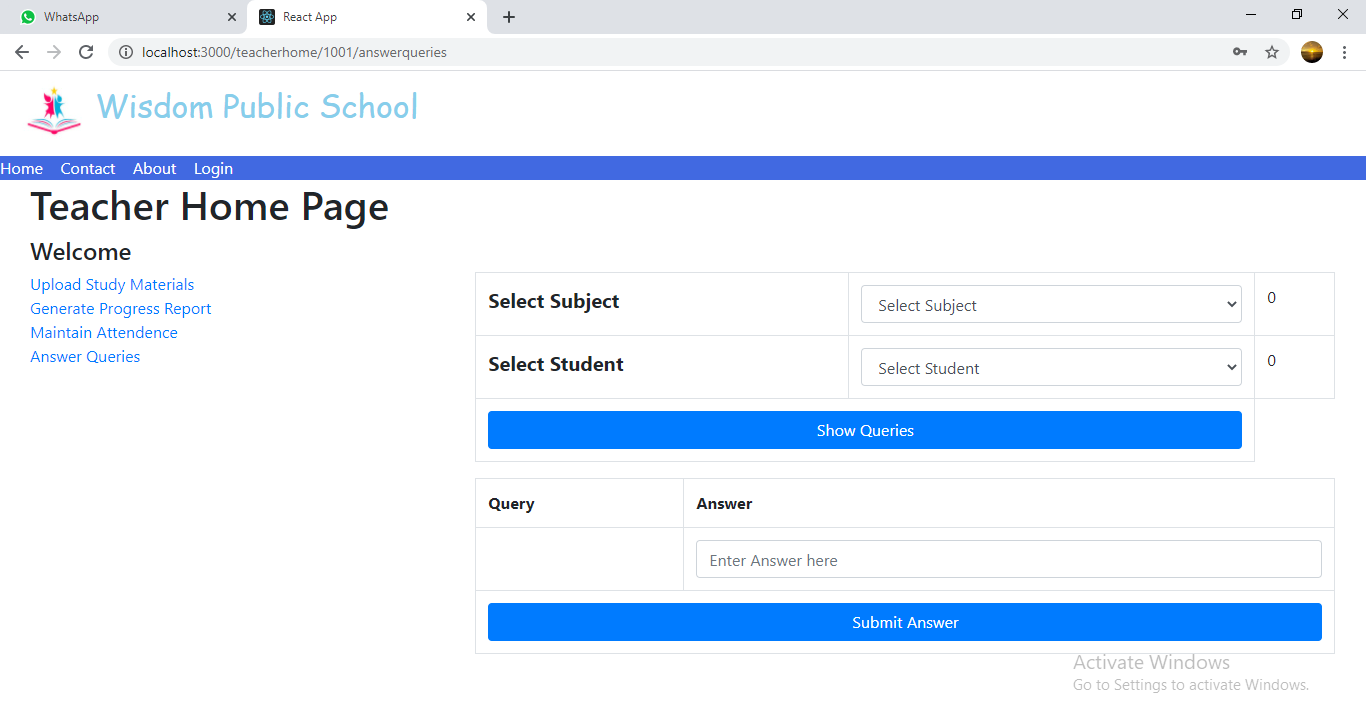
**8.6.2 Generate Progress Report Page**



**8.6.3 Maintain Attendance Page**

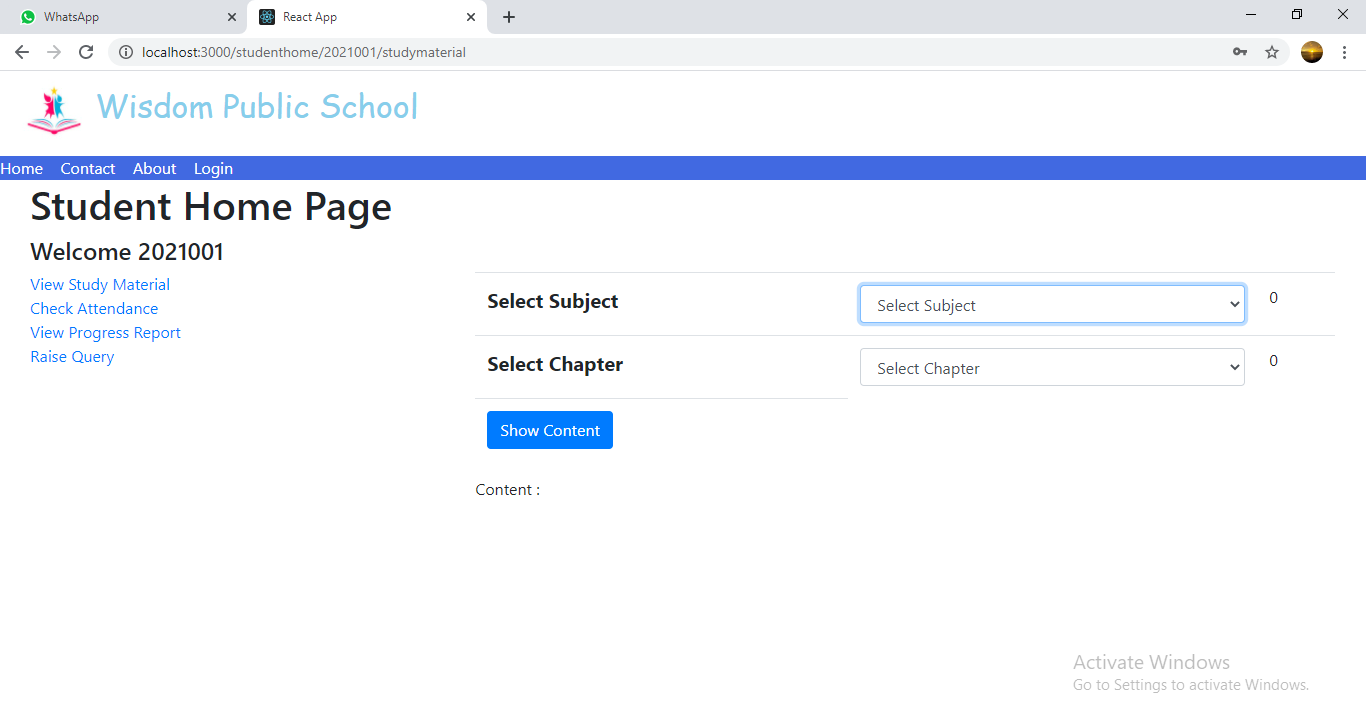


**8.6.4 Answer Queries Page**

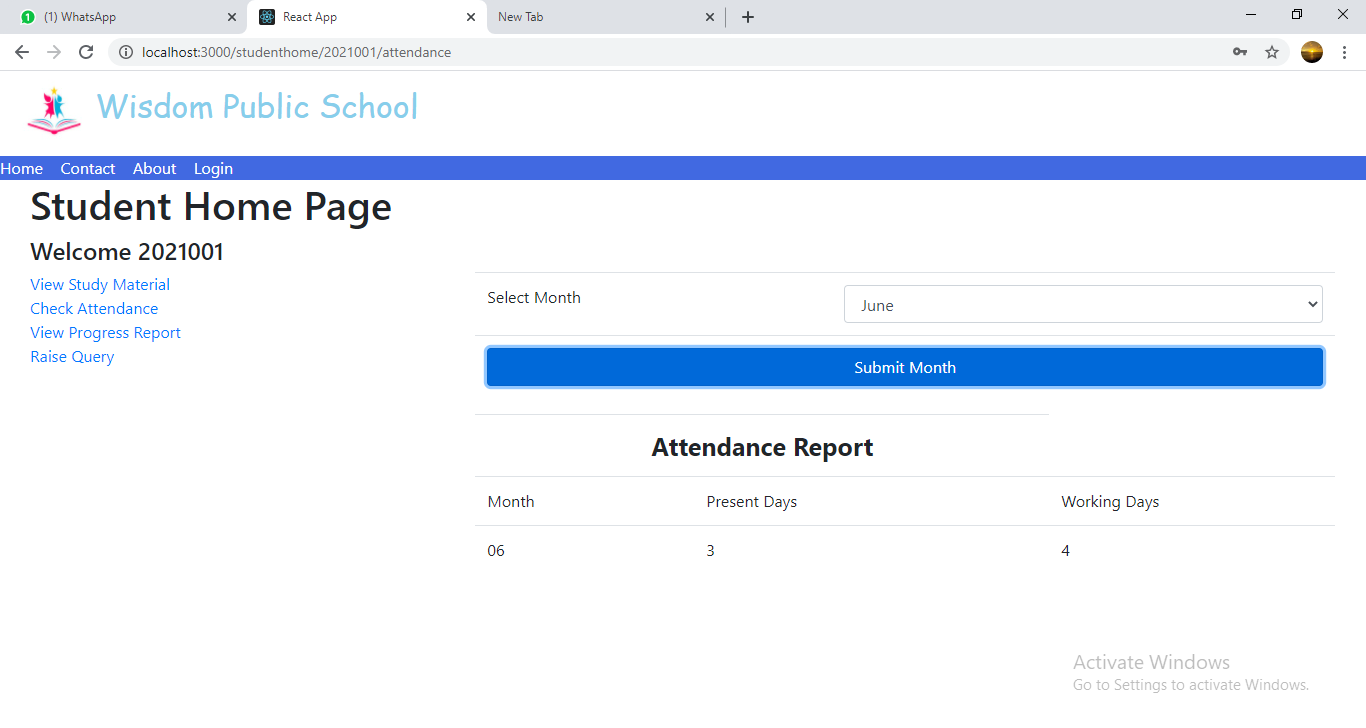


**8.7 Student Home Page**

**8.7.1 View Study Material Page**

****

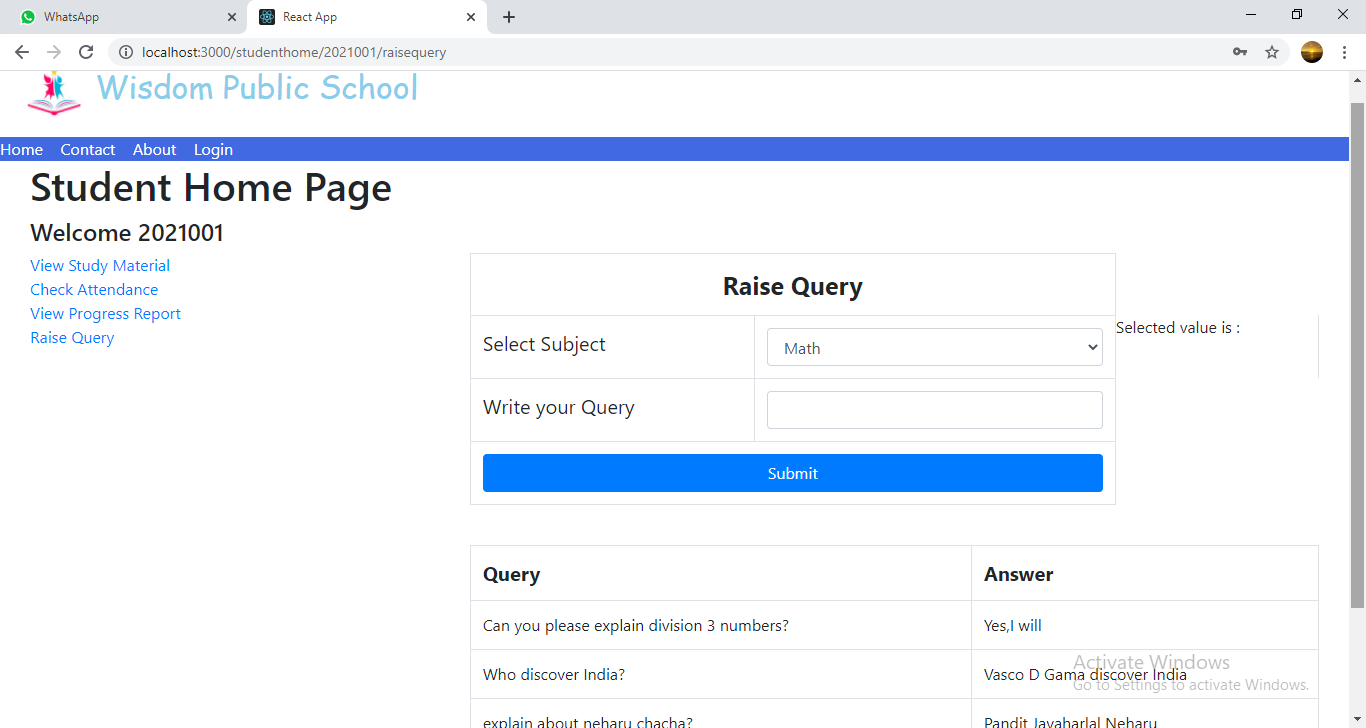
**8.7.2 Check Attendance Page**



**8.7.3 View Progress Report Page**



**8.7.4 Raise Query Page**



**9. CONCLUSION AND FUTURE SCOPE**

Paperless School Management System provides better education and platform to get connected with Teachers and Students efficiently.

Our System provides a very user-friendly platform where Student can easily read the study materials, check the attendance, raise the queries and view progress report, also Teacher will upload study materials, maintain attendance, answer queries and generate progress report.

Our system is aimed efficient management of various tasks like Generating time table for students on daily basis, Managing fees payment, Examination management. Digitally monitor daily activities along with managing all the resources and information on a single platform.