**Back-End Development**

Block 6

**Student Name:** M Talha Imran

**ID:** L39531244

GITHUB: <https://github.com/talhaa587/Project1.git>

Contents

[Introduction 4](#_Toc172536932)

[Configuration with Database 4](#_Toc172536933)

[ Classes: 4](#_Toc172536934)

[ Pupils: 4](#_Toc172536935)

[ Parents/Guardians: 4](#_Toc172536936)

[ Teachers: 4](#_Toc172536937)

[ Staff: 4](#_Toc172536938)

[Deployment 4](#_Toc172536939)

[ Setting up the EC2 Instance: 5](#_Toc172536940)

[ Installing Apache: 5](#_Toc172536941)

[ Installing MySQL: 5](#_Toc172536942)

[ Installing PHP: 5](#_Toc172536943)

[ Deploying the Application: 5](#_Toc172536944)

[Login and Signup 5](#_Toc172536945)

[ Login: 5](#_Toc172536946)

[ Signup: 5](#_Toc172536947)

[Manage Class 5](#_Toc172536948)

[ Class Name: 5](#_Toc172536949)

[ Capacity: 5](#_Toc172536950)

[Manage Staff 5](#_Toc172536951)

[ Name: 5](#_Toc172536952)

[ Address: 5](#_Toc172536953)

[ Phone Number: 6](#_Toc172536954)

[ Annual Salary: 6](#_Toc172536955)

[ Background Check Status: 6](#_Toc172536956)

[Manage Pupil 6](#_Toc172536957)

[ Name: 6](#_Toc172536958)

[ Address: 6](#_Toc172536959)

[ Medical Information: 6](#_Toc172536960)

[ Class Enrollment: 6](#_Toc172536961)

[Manage 6](#_Toc172536962)

[ Name: 6](#_Toc172536963)

[ Address: 6](#_Toc172536964)

[ Phone Number: 6](#_Toc172536965)

[ Annual Salary: 6](#_Toc172536966)

[ Background Check: 6](#_Toc172536967)

[Manage 6](#_Toc172536968)

[ Name: 6](#_Toc172536969)

[ Address: 7](#_Toc172536970)

[ Email: 7](#_Toc172536971)

[ Telephone: 7](#_Toc172536972)

[ER Diagram 7](#_Toc172536973)

[ Classes to Pupils: 7](#_Toc172536974)

[ Pupils to Parents: 7](#_Toc172536975)

[ Teachers to Classes: 7](#_Toc172536976)

[Why These Technologies 7](#_Toc172536977)

[ HTML and CSS: 7](#_Toc172536978)

[ PHP: 7](#_Toc172536979)

[ MySQL: 7](#_Toc172536980)

[ AWS: 7](#_Toc172536981)

[Conclusion 7](#_Toc172536982)

[References 8](#_Toc172536983)

# Introduction

In this report, I have documented the whole implementation process and how I have developed this website. This system will replace the school’s old paper-based system with a digital web solution that will efficiently manage everything. The technologies that I have used in this project are HTML, CSS, PHP, MySQL, and AWS.

# Configuration with Database

Firstly, I configured the database then I drew an Entity-Relationship (ER) diagram in order to design a database that accurately represents the requirements. Moreover, we all know that the ER diagram is an essential tool for visualizing the data structure and relationships between different entities.

The entities defined include:

* Classes: This entity contains information about each class (class name, capacity).
* Pupils: This entity store pupil details like name, address, medical information and class enrollment.
* Parents/Guardians: This entity keeps records of parents or guardians with information like name, address, email and telephone.
* Teachers: This entity contain teacher details like name, address, phone number, annual salary and background check status.
* Staff: This entity records information about other school staff members.

This ER diagram will make sure that each entity is properly linked and maintaining data properly and providing efficient data retrieval. For example, pupils and classes relationship makes sure that each pupil is associated with only one class at a time, and each class can have multiple pupils up to its capacity.

I implemented the database using MySQL which is a widely-used database management system known for its good performance. MySQL’s support for SQL (Structured Query Language) makes sure that database operates perfectly.

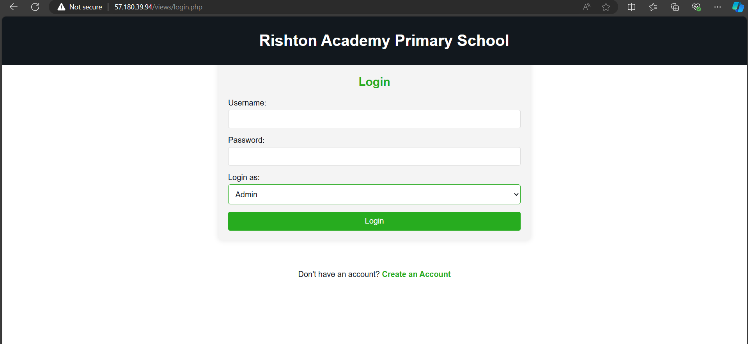
# Deployment

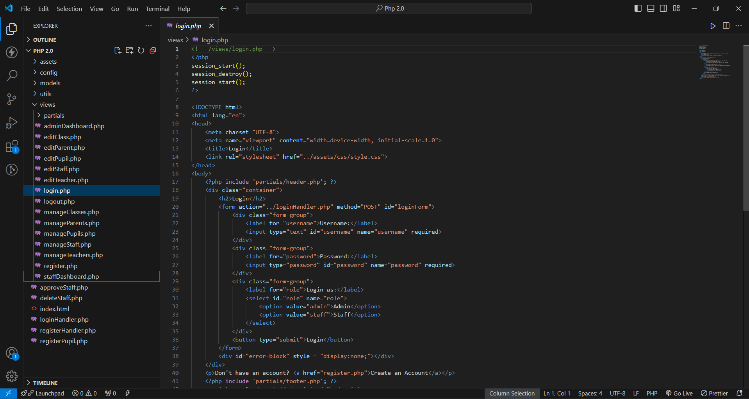
I deployed this website on AWS. I choose AWS because it has extensive services and a global structure.

Steps involved in deployment:

* Setting up the EC2 Instance: I launched EC2 instance with an Amazon AMI (Amazon Machine Image) and then security groups were configured to allow HTTP and SSH access.
* Installing Apache: I installed and configured Apache to serve the web application. It is a power full web server
* Installing MySQL: I installed MySQL in order to manage the database. Proper validations were done to make sure the database security.
* Installing PHP: I installed PHP and then integrated it with Apache and MySQL just to handle the server-side scripting and database interactions.
* Deploying the Application: At last, the PHP files, HTML, and CSS were uploaded to the EC2 instance. Configuration files were edited to ensure proper connection to the MySQL database.

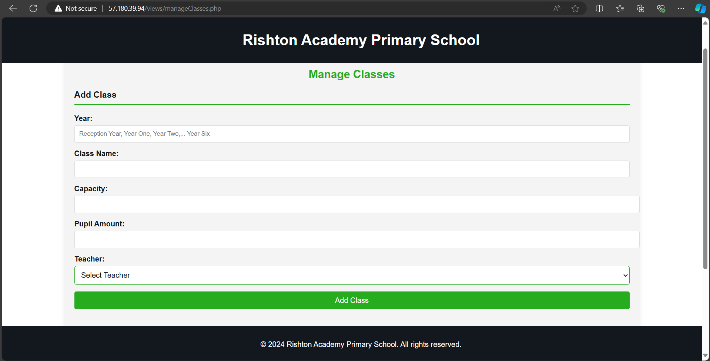
# Login and Signup

I have made sure that application have a secure user authentication system that includes login and signup functions. This makes sure that no one can access the system illegally.

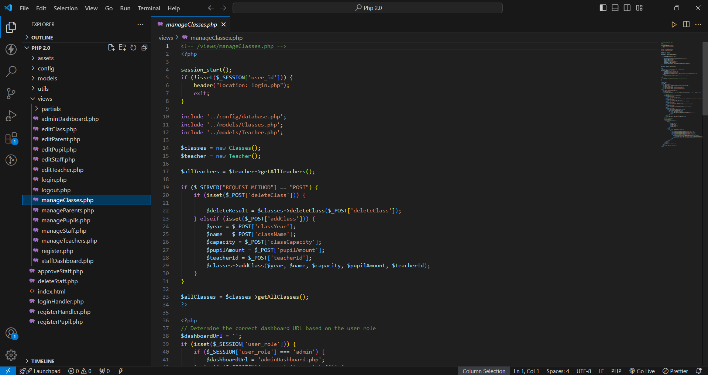
* Login: I implemented the login functionality with the help of PHP. It verifies the data from the database.
* Signup: The signup process allows all of the new users to have accounts. User input is checked and to avoid any type of SQL injection a.

----------------------------------------

# Manage Class

The "Manage Class" module allows admin to access and edit class records. This includes functions like add, edit, and delete. Each class record consists of:

* Class Name: The name of the class (e.g., Reception Year, Year One).
* Capacity: The maximum number of pupils.

I implemented this module by creating the data forms for data entry and by doing validation using HTML and PHP. I used SQL queries to insert, update and delete records in the MySQL database.

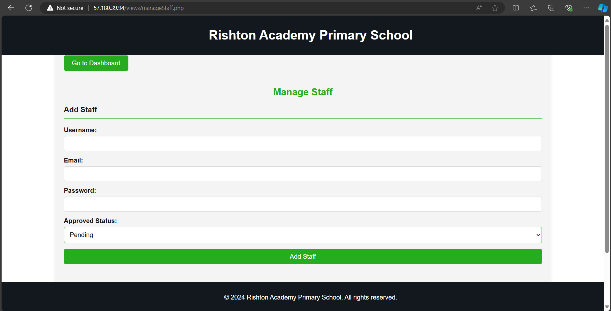
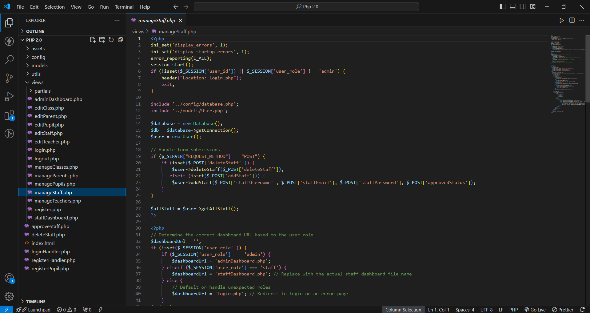
# 

------------------------------------------

# Manage Staff

The "Manage Staff" module provides functions to manage staff members which include teachers and other school staff. I have made this module support adding, editing and deleting the records. Each staff record includes:

* Name: The staff member’s name.
* Address: Contact address.
* Phone Number: Contact number.
* Annual Salary: Salary information.
* Background Check Status: Information about the completion of background checks.

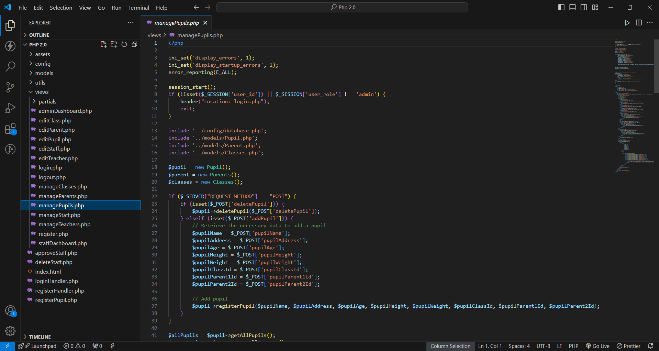
I have created the forms to handle data entry and then I used PHP code to handle submissions

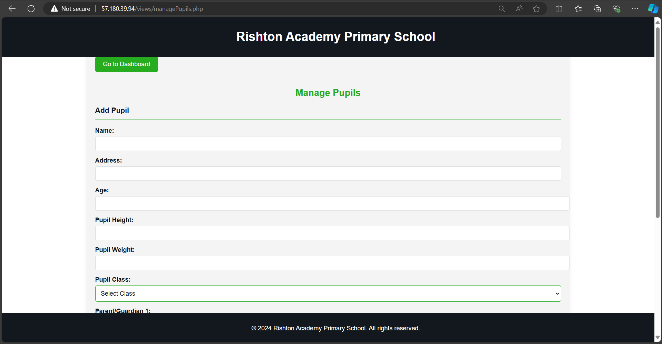
# 

---------------------------------------------------------------------------------------

# Manage Pupil

The "Manage Pupil" module allows the administrator to edit pupil records. This includes adding new pupils, editing existing information and deleting pupil records. Each pupil record contains:

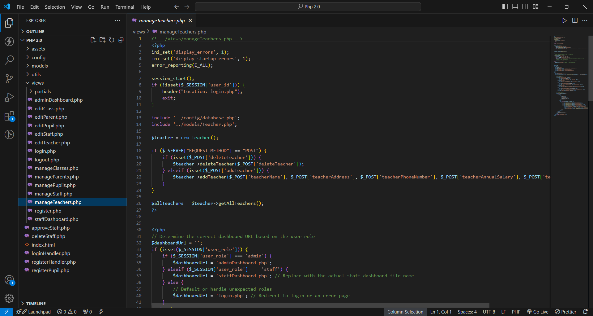
* Name: Pupil’s name.
* Address: Contact address.
* Medical Information: Relevant medical details.
* Class Enrollment: The class in which the pupil is enrolled.

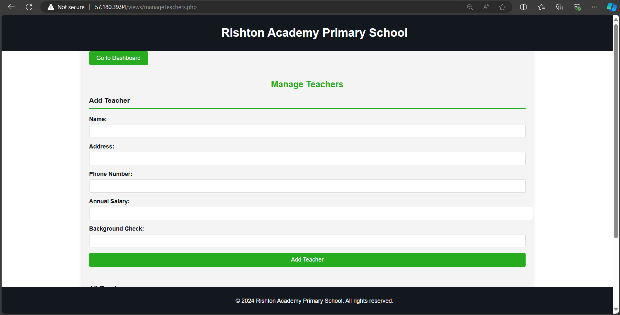
In this module I have data Validation and again PHP is used to handle submissions.

---------------------------------------------------

ManageTeachers

I have designed the "Manage Teachers" module to manage teacher records. Administrators can add, edit and delete teacher records. Each record includes:

* Name: Teacher’s name.
* Address: Contact address.
* Phone Number: Contact number.
* Annual Salary: Salary details.
* Background Check: Status of background checks

.

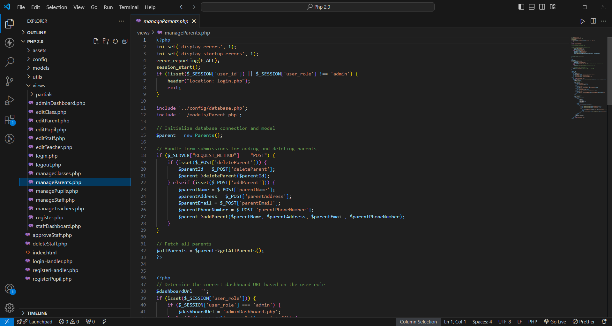
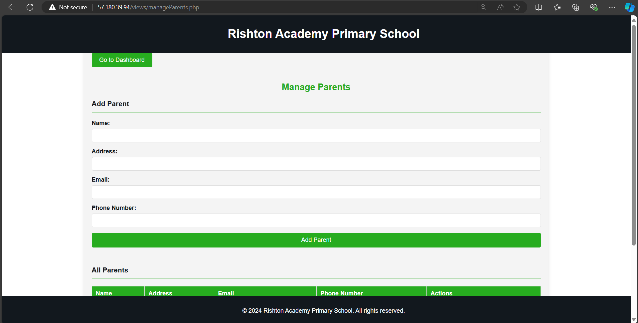
Like all other PHP is again used for handling submissions and SQL is used for database.

---------------------------------------------------------------------------------------

ManageParents

The "Manage Parents" module makes sure that he management of parent’s records is done properly. Administrators can add new parents, edit existing records and delete parent records. Each parent record includes:

* Name: Parent or guardian’s name.
* Address: Contact address.
* Email: Email address.
* Telephone: Contact number.

In this system I have allowed the parents to be associated with multiple pupils in order to facilitate the management of sibling information

# ER Diagram

I have created an ER diagram using Lucid Charts which serves as a blueprint for the database design. It outlines the entities, their attributes and the relationships between them. The diagram makes sure a normalized database structure, ensure data integrity. The key entities in the diagram include Classes, Pupils, Parents/Guardians, Teachers, and Staff, with relationships such as:

* Classes to Pupils: One-to-many relationship, where a class can have many pupils.
* Pupils to Parents: Many-to-many relationship, allowing pupils to have multiple parents and vice versa.
* Teachers to Classes: One-to-one relationship, where each teacher is assigned to one class.

# Why These Technologies

The choice of technologies was based on several factors, including ease of use, compatibility, and industry standards:

* HTML and CSS: These tools are one of the main tools of the web development. I used the HTML for creating a good structure and then I used the CSS for styling
* PHP: I used PHP as it can be easily connected with the MySQL. It is also well known in the world of Back-end development.
* MySQL: I used MySQL because it is an efficient database management system. In my view it is one of the most reliable systems.
* AWS: I used AWS because it offers scalability, reliability, and high availability. These things made it ideal for me.

# Conclusion

This report has outlined the development process efficiently. Each module is designed to perform specific functions ensuring an efficient management system. The choice of technologies was driven by their compatibility, ease of use and industry standards, ensuring a scalable and maintainable solution.

# References

1. Welling, L., & Thomson, L. (2008). **PHP and MySQL Web Development**. Addison-Wesley Professional.
2. Ben-Natan, R. (2014). **Implementing Database Security and Auditing: Includes Examples for Oracle, SQL Server, DB2 UDB, Sybase**. Digital Press.
3. Hughes, J., & Mohr, J. (2016). **Practical PHP and MySQL Website Databases: A Simplified Approach**. Apress.
4. AWS Documentation. (n.d.). **Amazon Web Services (AWS)**. Retrieved from [AWS Documentation](https://aws.amazon.com/documentation/)
5. Silberschatz, A., Korth, H. F., & Sudarshan, S. (2010). **Database System Concepts**. McGraw-Hill.