****

**Project Report on**

**Admission portal**

**Submitted to**

**VISHWAKARMA UNIVERSITY, PUNE**

**Under the Initiative of**

**Contemporary Curriculum, Pedagogy, and Practice**

**(C2P2)**

**PROJECT BASED LEARNING - PYTHON**

**By**

|  |  |
| --- | --- |
| **Name of the Student** | **: Aishwarya Zinjurte** |
| **SRN. No.** | **: 31231743** |
| **Class/Div.** | **: SY / C** |
| **Roll no.** | **: 41** |

**Faculty In-charge: Prof. Jayendra Jadhav**

**Department of Artificial Intelligence**

**Faculty of Science & Technology**

**Academic Year**

**2024-2025 Term-II**

1. ProblemStatement
   1. Problem Statement Sr. No:
   2. Project Title:
   3. Problem Statement:
2. Introduction
3. Methodology
4. Implementation and Results
5. Outcomeof the Project
6. Conclusion
7. References

1. **Problem Statement**

1.1. **Problem Statement Sr. No:** 41

1.2. **Project Title:** **College Admission Portal**

1.3. **Problem Statement:**

Educational institutions require a web-based system to manage student admissions efficiently. The system needs to handle online application submissions, document uploads, and application status tracking. It should provide role-based access for students and administrators, with features for application processing, status updates, and secure data management.

1. **Introduction**

The College Admission Portal is a web-based Java application designed to modernize and streamline the college admission process. This application provides a user-friendly interface for students to submit applications online and for administrators to manage the admission process. The system is built using Java Servlets and JSP for the web interface and implements MVC architecture for efficient data management and processing.

1. **Methodology**

The project follows a structured development approach:

1. Analysis Phase:

• Requirements gathering

• System architecture design

• Database design planning

2. Design Phase:

• Class design (User, Application, Document)

• Web interface design

• Database schema design

3. Implementation Phase:

• Frontend development using JSP and Bootstrap

• Backend development using Java Servlets

• Database implementation using MySQL

• Testing and deployment

1. **Implementation and Results**

The implementation includes:

**1. Core Classes:**

a) User.java:

```java  
public class User {  
 private int id;  
 private String email;  
 private String password;  
 private String role;  
 private Timestamp createdAt;  
  
 // Constructor  
 public User(String email, String password, String role) {  
 this.email = email;  
 this.password = password;  
 this.role = role;  
 }  
  
 // Getters and Setters  
 public int getId() { return id; }  
 public void setId(int id) { this.id = id; }  
 // ... other getters and setters  
}  
```

b) Application.java:

```java  
public class Application {  
 private int id;  
 private int userId;  
 private String firstName;  
 private String lastName;  
 private Date dateOfBirth;  
 private String phone;  
 private String address;  
 private String program;  
 private String status;  
 private Timestamp createdAt;  
 private Timestamp updatedAt;  
  
 // Constructor and methods  
 // ... implementation details  
}  
```

c) ApplicationServlet.java:

```java  
@WebServlet("/application/\*")  
@MultipartConfig  
public class ApplicationServlet extends HttpServlet {  
 private ApplicationDAO applicationDAO;  
  
 @Override  
 public void init() throws ServletException {  
 applicationDAO = new ApplicationDAO();  
 }  
  
 protected void doPost(HttpServletRequest request,   
 HttpServletResponse response) throws ServletException, IOException {  
 // Implementation details  
 }  
}  
```

**Results:**

**Results Section**

The College Admission Portal has been successfully implemented with the following key features and results:

1. **User Authentication System**

* Secure login and registration for students and administrators
* Role-based access control with differentiated features for admins and students
* Session-based authentication to maintain user state

1. **Application Submission**

* Intuitive form interface for students to submit applications
* Document upload functionality for required credentials (marksheet, entrance scores)
* Real-time validation to ensure complete application submissions

1. **Application Management**

* Admin dashboard to view and manage all applications
* Application status transitions (Pending → Under Review → Accepted/Rejected)
* Document verification system for admins

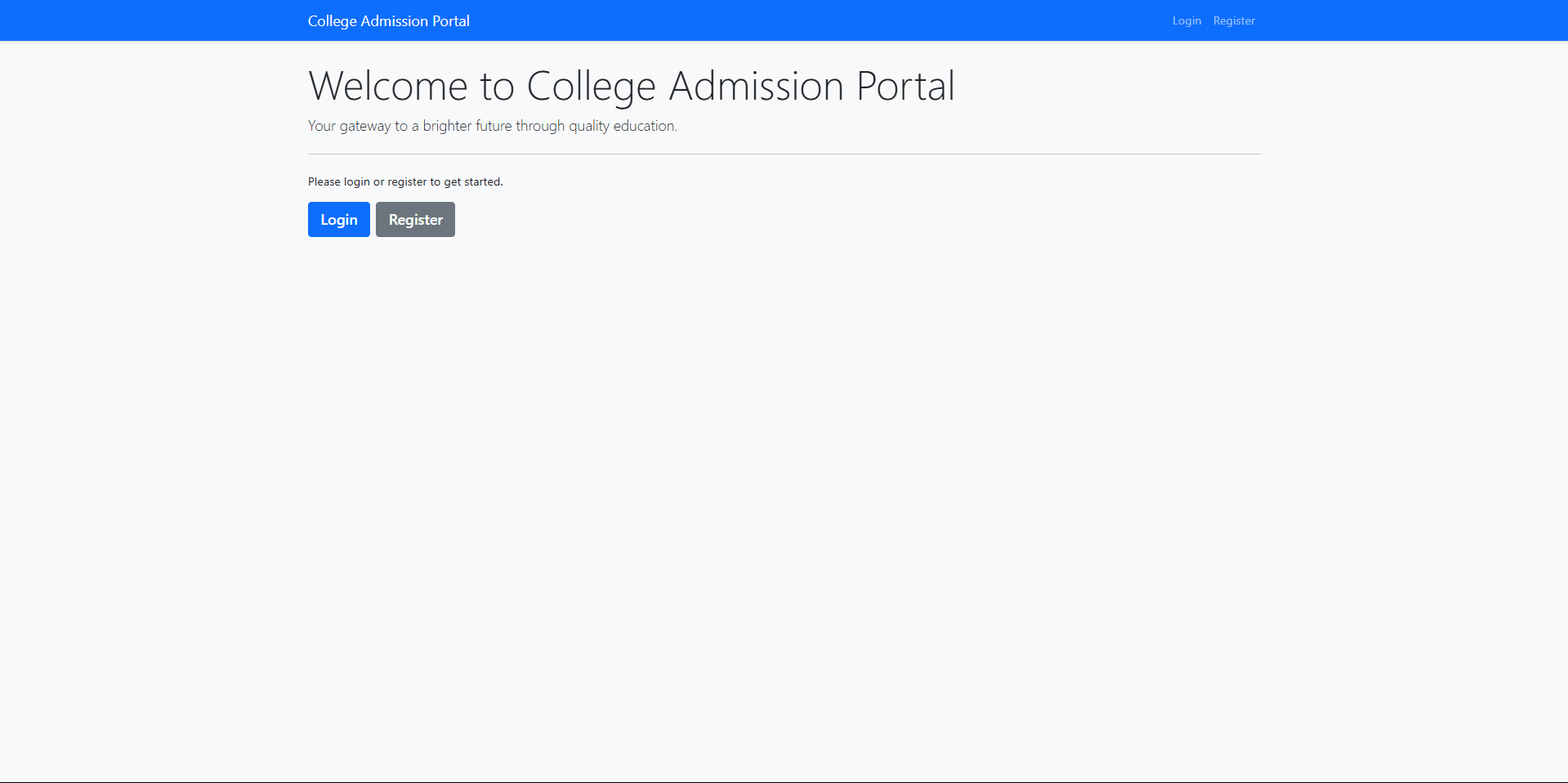
1. **Communication System**

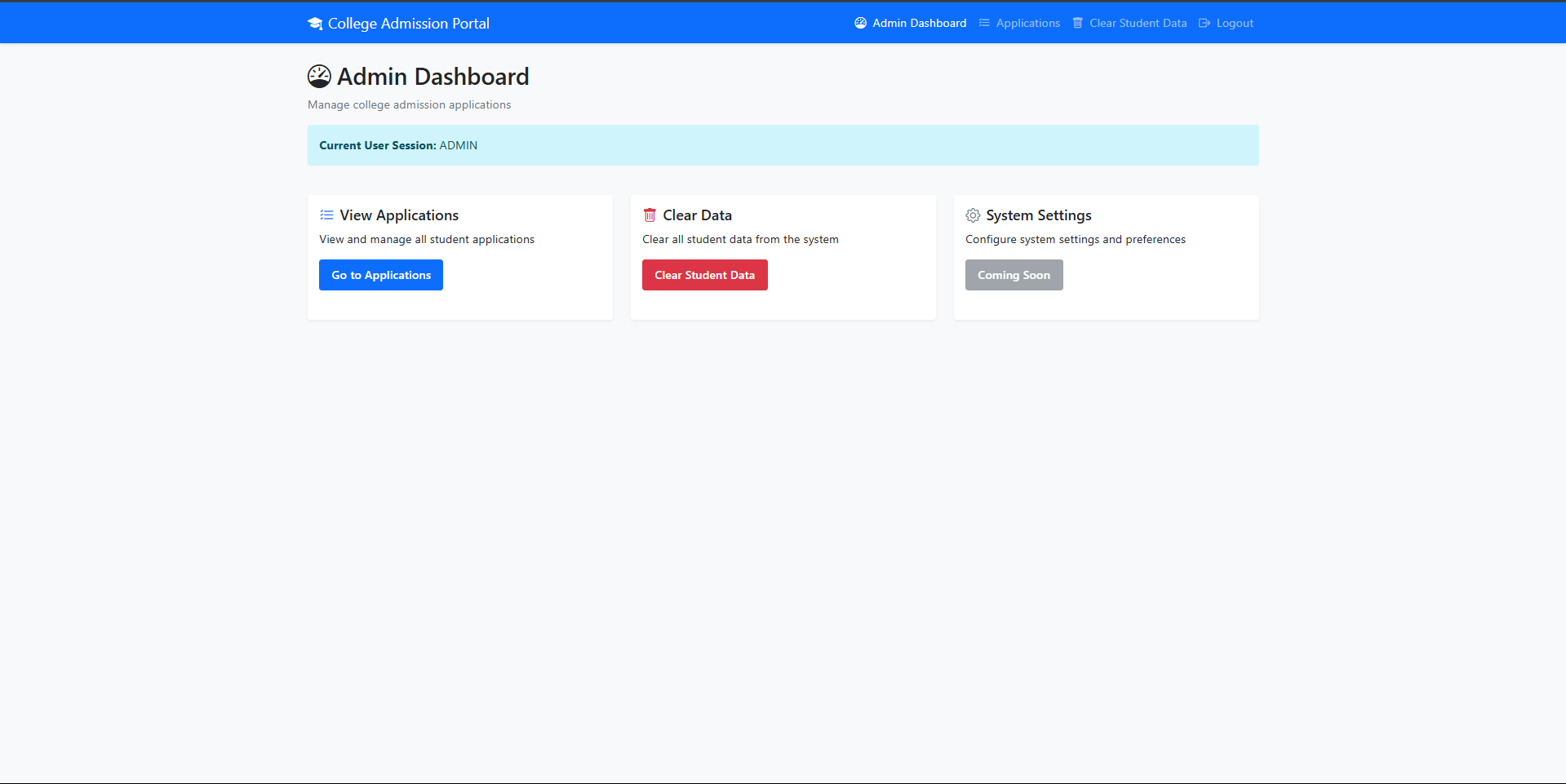
* Integrated messaging between applicants and administrators
* Real-time updates and notifications
* Message history for tracking communication threads

1. **Data Management**

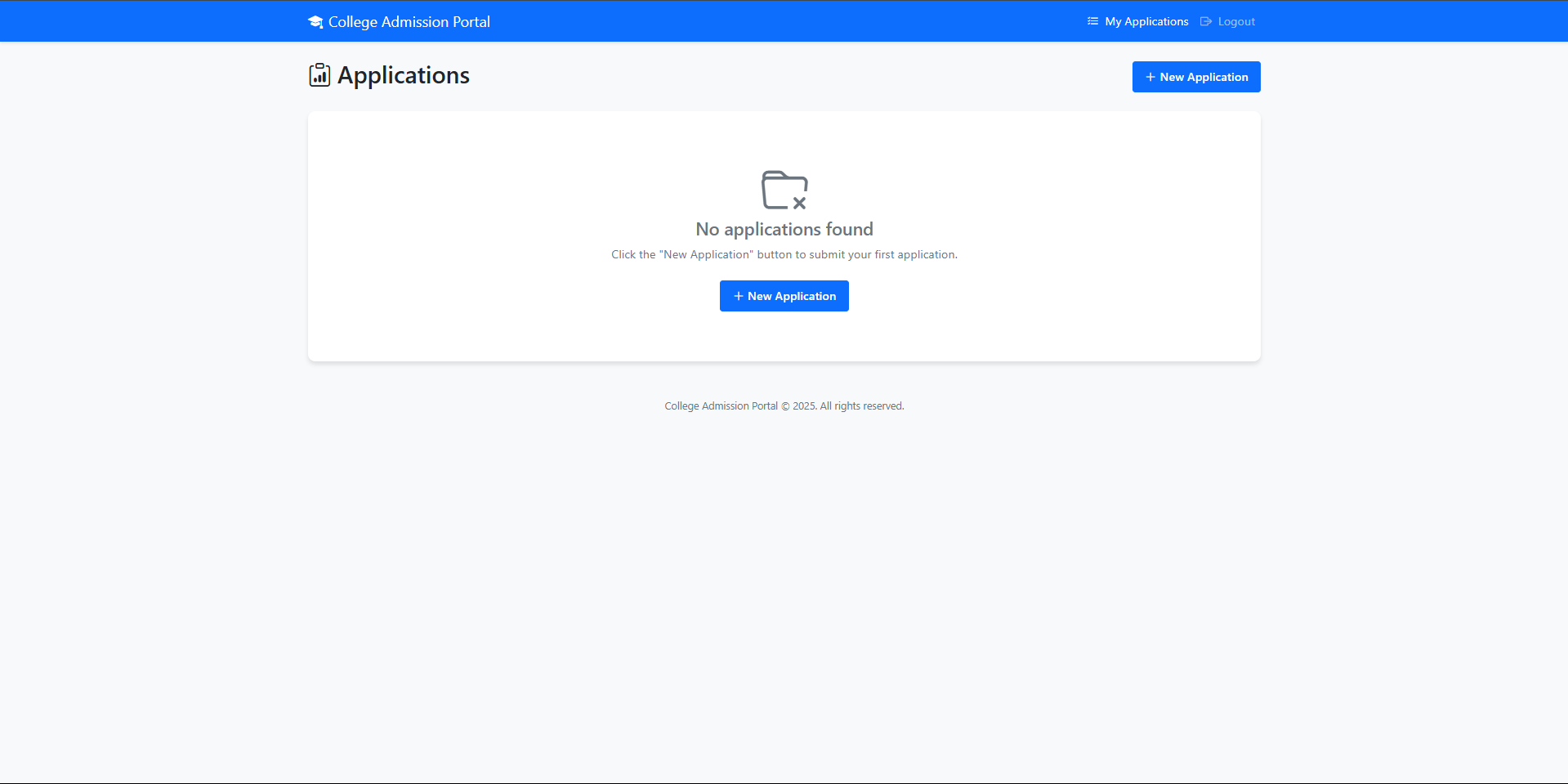
* Admin tool to clear student data while preserving admin accounts
* Secure document storage and retrieval
* Database integrity with proper foreign key relationships

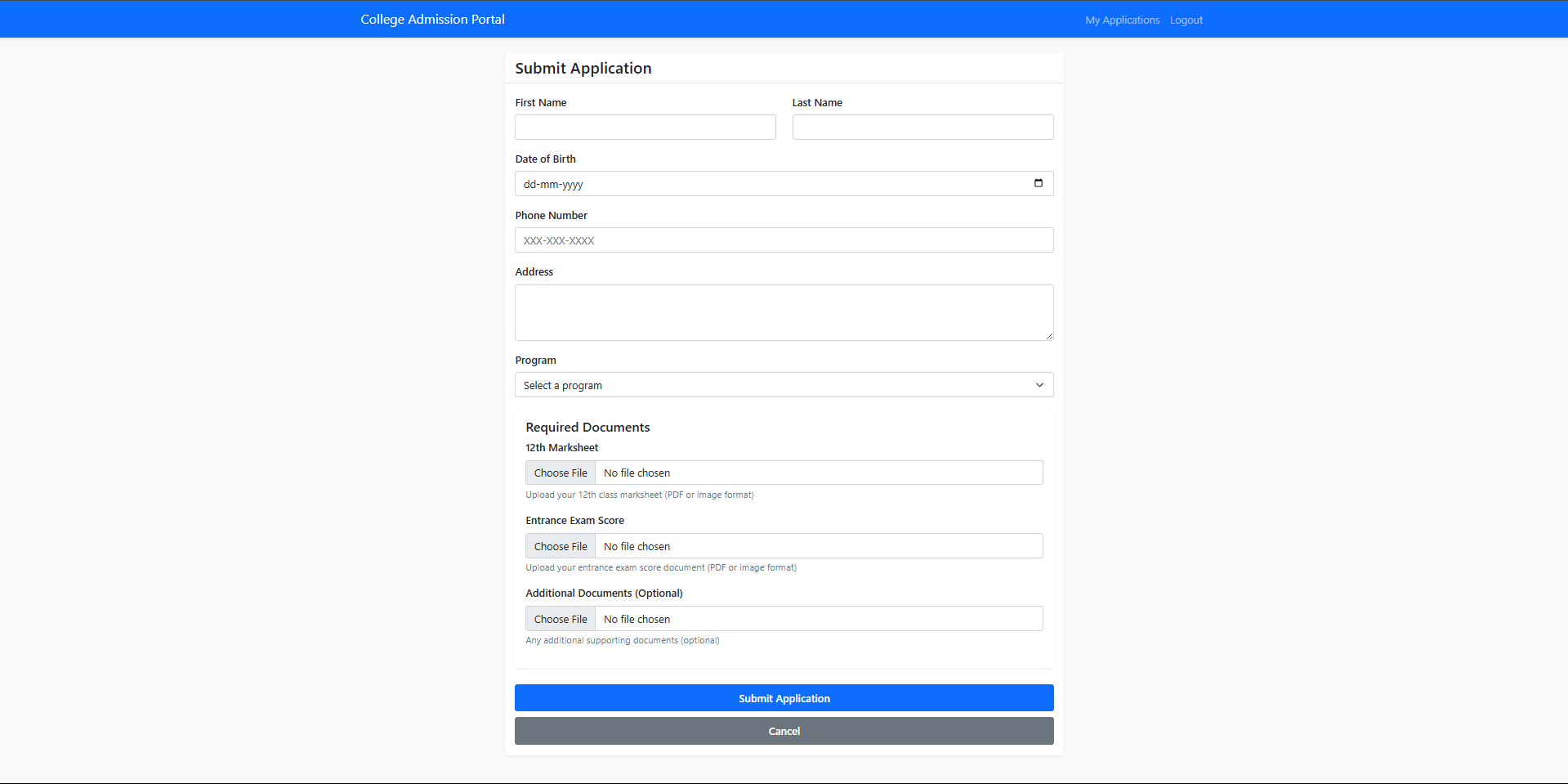
**WEBPAGE SCREENSHOTS:**

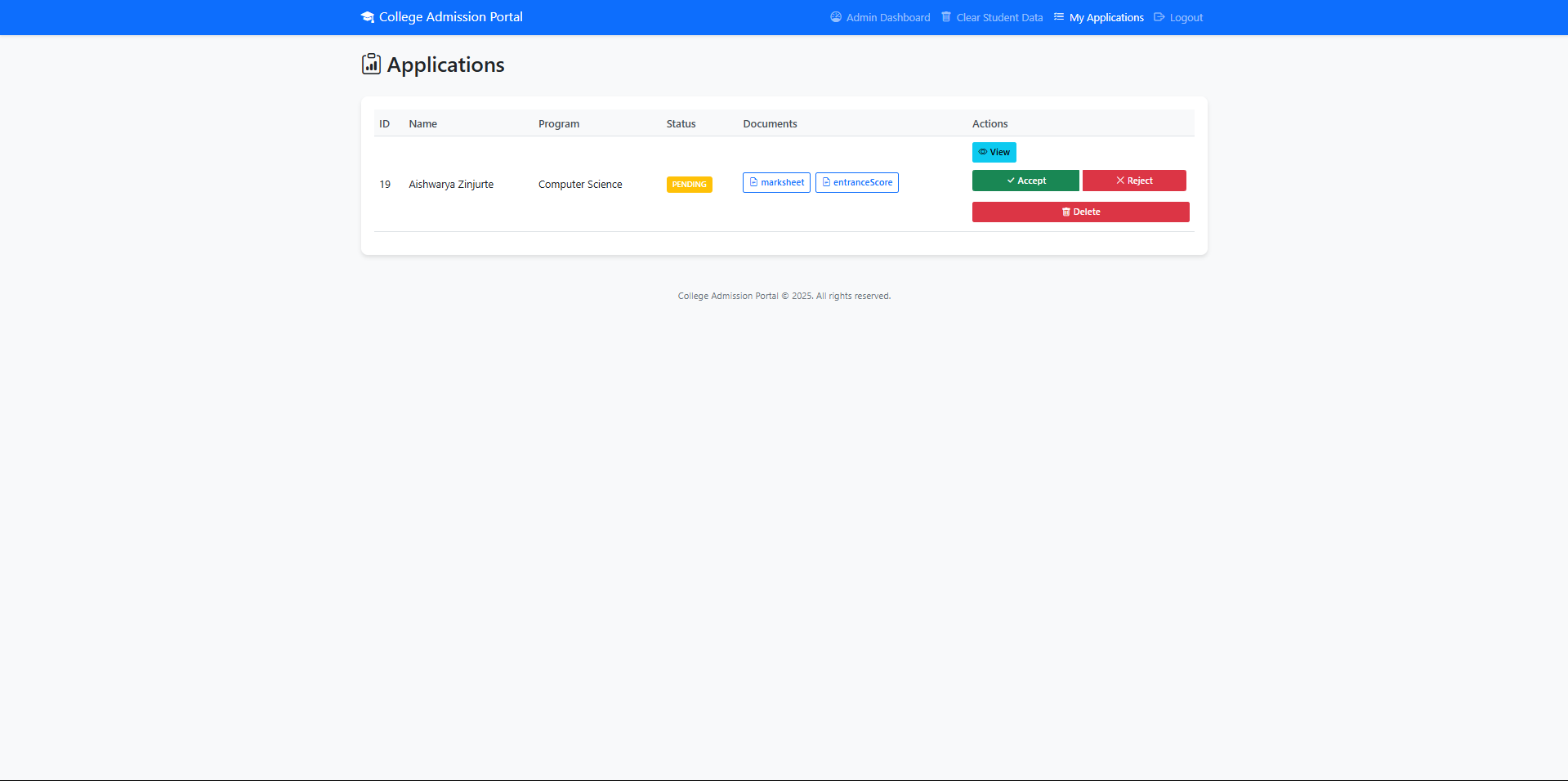
1. **Login page**
2. **Admin panel**

****

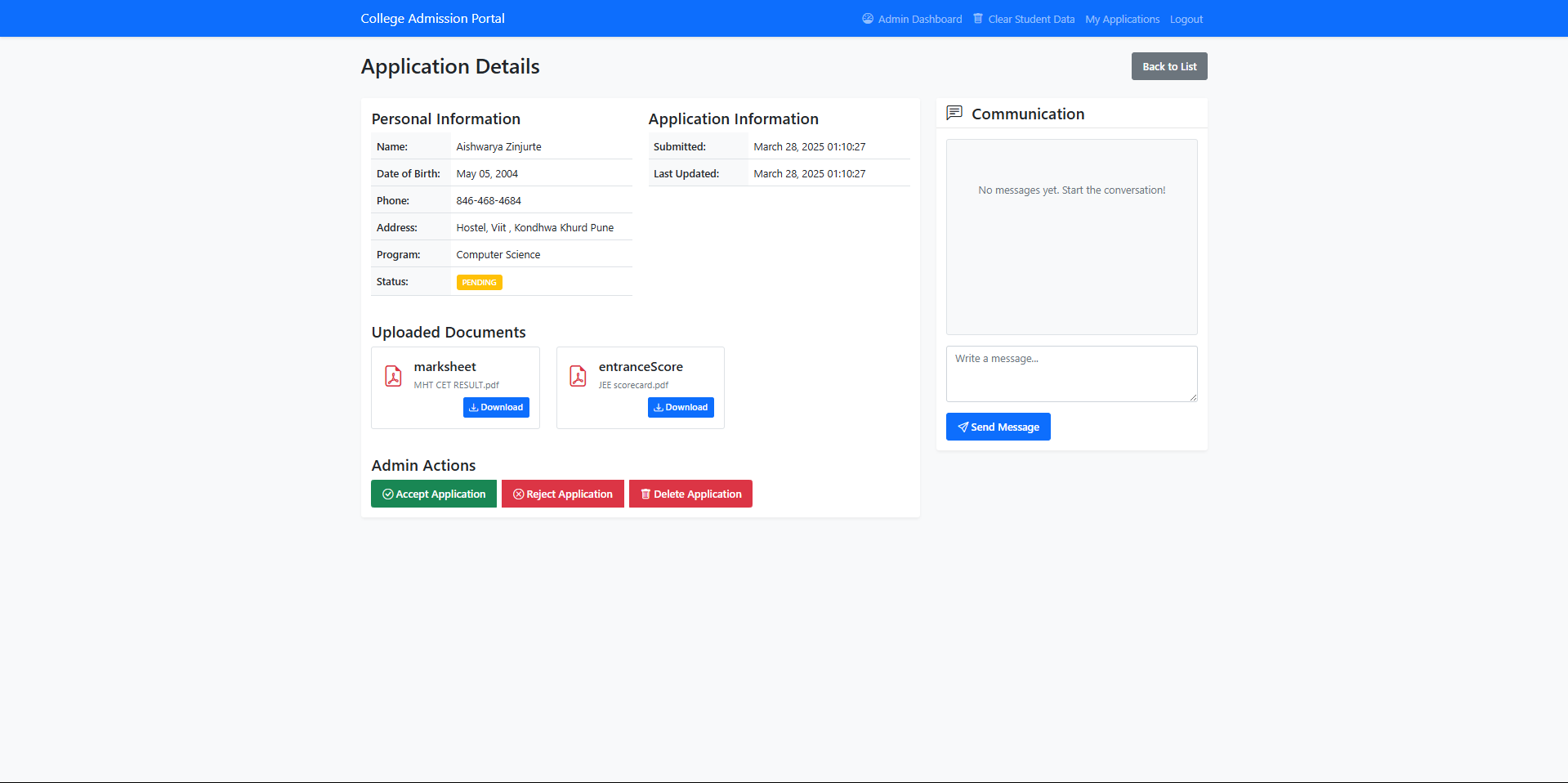
1. **Student Homepage**



1. **Application Page**
2. **Application panel as admin**

****

1. **Application Details as admin**

****

1. **Outcome of the project**

The project successfully achieved:

1. Functional Outcomes:

• Automated admission process

• Online application submission

• Document management

• Status tracking system

2. Technical Outcomes:

• Implementation of MVC architecture

• Secure user authentication

• Database integration

• Responsive web design

3. Business Outcomes:

• Streamlined admission process

• Reduced paperwork

• Improved efficiency

• Better applicant experience

1. **Conclusion**

The College Admission Portal successfully addresses the need for a modern, digital admission management system. The application demonstrates the practical application of Java web technologies and provides a user-friendly solution for both applicants and administrators. The implementation of MVC architecture, security features, and responsive design showcases the integration of technical skills with real-world requirements.

**7. References**

1. Java Servlet Programming by Jason Hunter

2. Head First Servlets and JSP by Kathy Sierra

3. MySQL Documentation

4. Bootstrap Documentation

5. Apache Tomcat Documentation

6. Maven Documentation

7. Web Application Security Guidelines

8. Database Design Principles

9. Web Development Best Practices

10. Software Engineering Standards