# Software Requirements Specification (SRS)

**For:** College Admission Management Portal

**Prepared by:** Group03

**Date:** 12th September, 2025

# 1. Introduction

## 1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the requirements for the development of the **College Admission Portal (CAP)**, an online system designed to streamline and automate the admission process for higher education institutions. The system is intended to replace traditional paper-based admission methods, minimize administrative overhead, and provide a transparent, efficient, and user-friendly platform for prospective students, staff, and management.

This document provides a detailed account of the functional and non-functional requirements of the system. It serves as a formal agreement between stakeholders, including college administrators, office staff, financial staff, system developers, and end-users (students and guardians).

The **primary objectives** of this document are:

- To clearly specify the **scope, goals, and constraints** of the College Admission Portal.
- To provide a **common understanding** of system features and operations among stakeholders.
- To act as a **reference document** for developers during the design, implementation, and testing phases.
- To establish a **baseline** for validation and verification of the final product.
- To assist in the future **maintenance and enhancement** of the system.

## 1.2 Scope

The College Admission Portal (CAP) is a **web-based application** designed to facilitate admission-related activities for students and administrators. The system will enable applicants to **register online, submit application forms, upload required documents, track application status, and receive admission decisions** digitally.

For administrators, including **Office Staff** and **Financial Staff**, the portal provides tools to **verify applications, manage applicant records, track fee payments, generate merit lists, allocate seats, and generate reports**. It will also include communication features to notify applicants about admission status, payment reminders, and important deadlines.

Key features of the system include:

- Online student registration and login.
- Application form submission with document upload.
- Online fee payment and receipt generation.

- Automated verification by Office Staff.
- Fee tracking and management by Financial Staff.
- Merit list generation and seat allocation.
- Admission confirmation and issuance of admission letters.
- Real-time updates and notifications to applicants.
- Reporting and analytics for administrators.

The system aims to **reduce manual workload, eliminate data redundancy, minimize human errors, and provide transparency** in the admission process. It will be accessible through web browsers on desktops, laptops, and mobile devices, ensuring convenience for all stakeholders.

### 1.3 Definitions, Acronyms, and Abbreviations

- **CAP** – College Admission Portal
- **Applicant/Student** – A prospective student applying for admission.
- **Office Staff** – Administrative staff responsible for verifying student details and processing applications.
- **Financial Staff** – Staff responsible for managing admission-related financial transactions such as fee payments.
- **Admin** – Superuser with oversight of Office and Financial Staff, may generate system-wide reports.
- **Application ID** – Unique identifier assigned to each applicant.
- **Merit List** – A list generated based on applicant performance and eligibility criteria.
- **Payment Gateway** – Secure third-party system integrated to facilitate online fee transactions.
- **UI/UX** – User Interface / User Experience.
- **NFR** – Non-Functional Requirement.
- **FR** – Functional Requirement.

### 1.4 References

The system design and requirements have been gathered from:

1. IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.
2. College Admission Guidelines by [University Grants Commission (UGC), India].
3. Online Admission Portals of reputed universities.
4. Payment gateway integration documentation (e.g., Razorpay, Paytm, Stripe).
5. Academic project requirements and institutional guidelines.

### 1.5 Overview

The rest of this SRS document is organized as follows:

- **Section 2: General Description** – Provides a high-level overview of the product perspective, functionalities, user characteristics, and constraints.
- **Section 3: Specific Requirements** – Describes in detail the functional and non-functional requirements of the system.
- **Section 4: Use Cases** – Provides detailed use-case models including <<include>> and <<extend>> relationships.
- **Section 5: Appendices** – Contains supporting information such as glossary, data flow diagrams, and references.

This document aims to present a **clear, unambiguous, and detailed description** of the College Admission Portal so that both technical and non-technical stakeholders can understand and contribute effectively.

## 2. General Description

### 2.1 Product Perspective

The College Admission Portal (CAP) is designed as a **standalone web-based system** that integrates seamlessly with existing institutional infrastructure such as student records, payment gateways, and reporting systems. It acts as a **bridge between applicants and college administration**, providing a unified and digital admission workflow.

The system will be hosted on a secure web server and made available via the institution's official domain. It can be accessed through standard web browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge, without requiring additional software installations. The design will follow a **modular architecture** with distinct modules for registration, document verification, fee management, merit list generation, and reporting.

**System Interfaces**

- **External Systems:**
    - **Payment Gateway** – The portal will integrate with trusted payment gateways (e.g., Razorpay, Paytm, Stripe) for processing admission and tuition fees.
    - **Email/SMS Notification System** – To send automated updates and alerts to applicants.
    - **College ERP (if available)** – For synchronization with the main student information system.
- **User Interfaces:**
    - **Applicant Interface** – Designed for prospective students to register, submit applications, upload documents, track admission status, and make payments.
    - **Office Staff Interface** – Provides tools for verifying applications, validating documents, and approving or rejecting candidates.
    - **Financial Staff Interface** – Used to manage payment confirmations, handle refunds, generate fee reports, and monitor outstanding balances.
    - **Admin Dashboard** – Enables system-wide reporting, configuration, and monitoring of admission activities.

**Hardware and Software Requirements**

- **Server Side:**
    - Operating System: Linux/Windows Server
    - Web Server: Apache or Nginx
    - Database: MySQL or PostgreSQL

- Backend Framework: Node.js / Django / Spring Boot
- Payment Gateway API integration

- **Client Side:**
  - Device: Laptop, desktop, or mobile phone
  - Browser: Any modern browser (latest version recommended)
  - Internet connectivity (minimum 1 Mbps for smooth experience)

### 2.2 Product Functions

The primary functions of the College Admission Portal are summarized below. These will later map into detailed functional requirements (FRs) in Section 3.

1. **User Registration & Authentication**
   - Students create accounts, verify via OTP/email, and log in securely.
   - Password recovery and two-factor authentication options provided.
2. **Application Form Submission**
   - Students fill out personal, academic, and course preference details.
   - Upload scanned documents such as mark sheets, identity proof, and photographs.
   - Application form validation with error prompts.
3. **Document Verification (Office Staff)**
   - Staff reviews submitted documents.
   - Approve, reject, or request resubmission of documents.
   - Maintain digital records of verification history.
4. **Fee Management (Financial Staff)**
   - Online payment through integrated payment gateway.
   - Issue digital receipts.
   - Refund management (if required).
   - Reports of pending, completed, and failed transactions.

5. **Merit List Generation**
    - Automatic calculation based on predefined rules (e.g., marks, reservation policies, entrance exam scores).
    - Ability to generate multiple merit lists (rounds of admission).
6. **Seat Allocation & Admission Confirmation**
    - Allocation of seats as per merit list and availability.
    - Students confirm admission by paying fees.
    - Issuance of admission confirmation letter digitally.
7. **Communication & Notifications**
    - Automated alerts via email/SMS about deadlines, results, and payments.
    - Dashboard notifications for applicants.
8. **Reporting and Analytics**
    - Reports on admission statistics, fee collection, course-wise enrollment.
    - Export functionality (CSV, PDF, Excel).

### 2.3 User Characteristics

The portal is designed for three major user categories:

1. **Applicants/Students**
    - Typically aged between 17–25.
    - Basic knowledge of computers and smartphones.
    - Familiar with online forms and digital payments.
    - Expect a simple and intuitive interface.
2. **Office Staff**
    - Experienced with administrative work, but not always technically skilled.
    - Require efficient tools for verifying large volumes of applications.
    - Need secure access with different privileges.

3. **Financial Staff**
    - Skilled in accounting and fee management.
    - Require accurate and real-time transaction data.
    - Need integration with institutional financial systems.
4. **Administrator (Optional)**
    - Possesses technical knowledge for managing the system.
    - Responsible for monitoring, system maintenance, and overall reporting.

### 2.4 General Constraints

The system must comply with various operational and regulatory constraints:

1. **Regulatory Constraints**
    - Must follow **UGC guidelines** for admissions.
    - Must comply with **IT Act, 2000** for digital transactions in India.
    - Adherence to **data privacy laws** (GDPR principles, if applicable).
2. **Technical Constraints**
    - Web-based only; offline access not available.
    - System must support at least **10,000 concurrent applicants**.
    - Availability of at least **99.5% uptime** during admission season.
3. **Operational Constraints**
    - Staff must be trained before full deployment.
    - College must have a reliable internet connection.
    - Payment gateways must support major Indian banks and wallets.
4. **Design Constraints**
    - User Interface must follow **WCAG 2.1 accessibility standards**.
    - Responsive design for mobile and desktop.
    - Role-based authentication for different users.

### 2.5 Assumptions and Dependencies

The design of CAP is based on certain assumptions and dependencies, including:

**Assumptions**

- Applicants have access to internet-enabled devices.
- Applicants are capable of making digital payments.
- The institution provides necessary technical infrastructure for hosting.
- Admission rules and merit list criteria are provided beforehand by the institution.

**Dependencies**

- **Payment Gateway Dependency** – If the external payment gateway experiences downtime, payments may fail.
- **Email/SMS Service Dependency** – Notifications depend on third-party services.
- **Browser Compatibility** – System performance may vary depending on browser version.
- **Database Server Availability** – Continuous server uptime is required to prevent data loss.

## 3. Specific Requirements

### 3.1 Functional Requirements (FRs)

The functional requirements define the **services and behaviors** the College Admission Portal (CAP) must provide. These requirements are grouped according to the major system modules.

#### 3.1.1 User Registration and Authentication

- **FR-1:** The system shall allow new applicants to register by providing personal details (name, email, phone number, date of birth).
- **FR-2:** The system shall send an OTP/email verification code to validate the applicant's identity.
- **FR-3:** The system shall allow applicants to set a password that complies with security standards (minimum 8 characters, including numbers and symbols).
- **FR-4:** The system shall provide login functionality using username/email and password.
- **FR-5:** The system shall support password recovery through OTP/email verification.
- **FR-6:** The system shall lock accounts temporarily after 5 failed login attempts to prevent brute-force attacks.

#### 3.1.2 Application Form Management

- **FR-7:** The system shall provide an online application form with sections for personal, academic, and course preference details.
- **FR-8:** The system shall validate inputs in real time (e.g., date format, marks range, mandatory fields).
- **FR-9:** The system shall allow applicants to upload supporting documents (e.g., certificates, ID proof, photos).
- **FR-10:** The system shall assign a unique **Application ID** to each applicant.
- **FR-11:** The system shall allow applicants to preview the completed form before final submission.
- **FR-12:** The system shall allow applicants to edit the form until the submission deadline.
- **FR-13:** The system shall lock the application after submission and generate a digital acknowledgment receipt.

### 3.1.3 Document Verification (Office Staff)

- **FR-14:** The system shall allow office staff to log in with staff-specific credentials.
- **FR-15:** The system shall present staff with a dashboard of pending applications requiring verification.
- **FR-16:** The system shall allow staff to view and verify uploaded documents.
- **FR-17:** The system shall allow staff to approve or reject an application.
- **FR-18:** The system shall allow staff to request resubmission of incorrect or unclear documents.
- **FR-19:** The system shall maintain logs of staff verification actions.

### 3.1.4 Fee Management (Financial Staff)

- **FR-20:** The system shall allow applicants to pay fees through an integrated online payment gateway.
- **FR-21:** The system shall generate a digital receipt upon successful payment.
- **FR-22:** The system shall notify applicants of failed or pending payments.
- **FR-23:** The system shall allow financial staff to verify fee transactions.
- **FR-24:** The system shall generate a list of applicants who have paid fees successfully.
- **FR-25:** The system shall support refund processing where applicable.
- **FR-26:** The system shall generate financial reports of daily and cumulative transactions.

### 3.1.5 Merit List and Seat Allocation

- **FR-27:** The system shall allow office staff to input eligibility and merit criteria (e.g., marks, entrance scores).
- **FR-28:** The system shall automatically generate a merit list based on defined rules.
- **FR-29:** The system shall allow staff to review and approve the merit list before publishing.
- **FR-30:** The system shall publish the merit list on the applicant’s dashboard.
- **FR-31:** The system shall allow multiple merit list rounds until seats are filled.
- **FR-32:** The system shall allocate available seats to students based on rank and preferences.
- **FR-33:** The system shall notify students of their seat allocation status.

### 3.1.6 Admission Confirmation

- **FR-34:** The system shall allow selected students to confirm admission by paying admission fees.
- **FR-35:** The system shall issue a digital admission letter to confirmed students.
- **FR-36:** The system shall update the student record in the institutional database after confirmation.

### 3.1.7 Notifications and Communication

- **FR-37:** The system shall send email/SMS notifications for important events (form submission, payment confirmation, merit list publication).
- **FR-38:** The system shall display notifications on applicant dashboards.
- **FR-39:** The system shall allow staff to send custom notifications to applicants.

### 3.1.8 Reporting and Analytics

- **FR-40:** The system shall generate reports on total applications received, verified, and admitted.
- **FR-41:** The system shall generate course-wise, category-wise, and gender-wise admission reports.
- **FR-42:** The system shall generate fee collection reports for financial staff.
- **FR-43:** The system shall allow exporting reports in PDF, Excel, and CSV formats.

### 3.1.9 System Administration

- **FR-44:** The system shall allow an administrator to manage user accounts (students, staff).
- **FR-45:** The system shall allow role-based access control (student, office staff, financial staff, admin).
- **FR-46:** The system shall provide a system health monitoring dashboard (uptime, database status).
- **FR-47:** The system shall allow admin to configure merit rules, deadlines, and notification templates.

### 3.2 Non-Functional Requirements (NFRs)

Each NFR is linked to corresponding FR(s).

#### 3.2.1 Performance Requirements

- **NFR-1 (FR-1 to FR-47):** The system shall support at least **10,000 concurrent users** during peak admission season.
- **NFR-2 (FR-27 to FR-33):** Merit list generation shall complete within **2 minutes** for up to 100,000 applications.
- **NFR-3 (FR-20 to FR-26):** Payment processing must provide a response within **10 seconds**.

#### 3.2.2 Security Requirements

- **NFR-4 (FR-1, FR-2, FR-6):** All user authentication must follow strong password policies and 2FA.
- **NFR-5 (FR-9, FR-16):** All uploaded documents must be encrypted (AES-256) when stored.
- **NFR-6 (FR-20 to FR-26):** Payment transactions must comply with **PCI-DSS** standards.
- **NFR-7 (FR-44, FR-45):** Role-based access control must prevent unauthorized access.

#### 3.2.3 Usability Requirements

- **NFR-8 (FR-7 to FR-12):** Application form must have an intuitive interface with real-time validation.
- **NFR-9 (FR-37, FR-38):** Notifications must be concise and available in both English and regional languages.
- **NFR-10 (FR-1 to FR-47):** System must support **WCAG 2.1 accessibility guidelines** for disabled users.

#### 3.2.4 Reliability Requirements

- **NFR-11 (FR-1 to FR-47):** The system shall provide at least **99.5% uptime** during the admission cycle.
- **NFR-12 (FR-10, FR-34):** All application and admission data must be backed up every **6 hours**.
- **NFR-13 (FR-27 to FR-36):** In case of system crash, merit lists and admission status must be recoverable within **30 minutes**.

### 3.2.5 Maintainability Requirements

- **NFR-14 (FR-44 to FR-47):** The system shall allow updates without downtime using rolling deployments.
- **NFR-15 (FR-1 to FR-47):** Source code must follow modular design for easy maintenance.

### 3.2.6 Portability Requirements

- **NFR-16 (FR-1 to FR-47):** The system must be compatible with Windows, Linux, and Mac servers.
- **NFR-17 (FR-1 to FR-47):** The client interface must be responsive and work on desktop and mobile devices.

## 3.3 Interface Requirements

### 3.3.1 User Interface

- Clean and simple design with college branding.
- Responsive design for desktops, laptops, tablets, and mobile devices.
- Color contrast and font size adjustable for accessibility.

### 3.3.2 Hardware Interface

- Server hardware with minimum: 8-core CPU, 32 GB RAM, 1 TB SSD.
- Client hardware: any device with at least 2 GB RAM and modern browser.

### 3.3.3 Software Interface

- Operating System: Linux (preferred), Windows Server supported.
- Database: MySQL/PostgreSQL.
- API: RESTful APIs for integration with ERP and Payment Gateway.

### 3.3.4 Communication Interface

- HTTPS protocol with TLS 1.3 for secure communication.
- SMTP integration for email.
- SMS gateway API for mobile notifications.

## 4. Appendices

### 4.1 Glossary

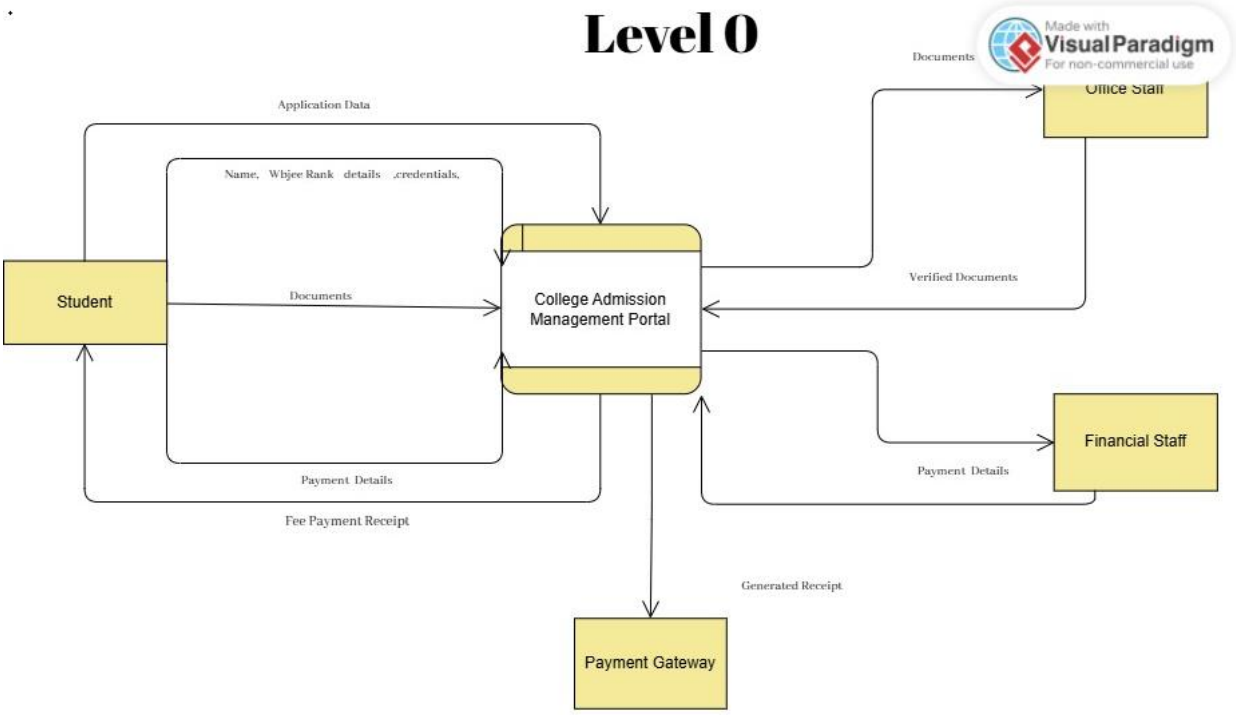
- **Admission Confirmation** – Final acceptance of a student after payment and seat allocation.
- **Admin** – Superuser responsible for overseeing Office Staff and Financial Staff operations.
- **Applicant/Student** – A prospective student applying through the portal.
- **Application ID** – Unique identifier for each submitted application.
- **CAP (College Admission Portal)** – The system being developed.
- **ERP (Enterprise Resource Planning)** – Existing institutional software where admission data may sync.
- **Financial Staff** – College staff responsible for handling fee-related transactions.
- **Merit List** – System-generated ranked list of eligible students.
- **NFR (Non-Functional Requirement)** – Requirements that specify quality attributes (e.g., security, performance).
- **Office Staff** – Administrative staff responsible for verifying applications.
- **Payment Gateway** – External service provider enabling online fee transactions.
- **Role-Based Access Control (RBAC)** – Security feature ensuring only authorized staff perform specific actions.
- **Student Dashboard** – Personalized web interface where applicants track their application status.
- **Two-Factor Authentication (2FA)** – An extra security step requiring OTP/email verification.

### 4.2 Supporting Diagrams

Below are the supporting diagrams of the College Admission Portal. These diagrams reinforce the functional and architectural understanding of the system.

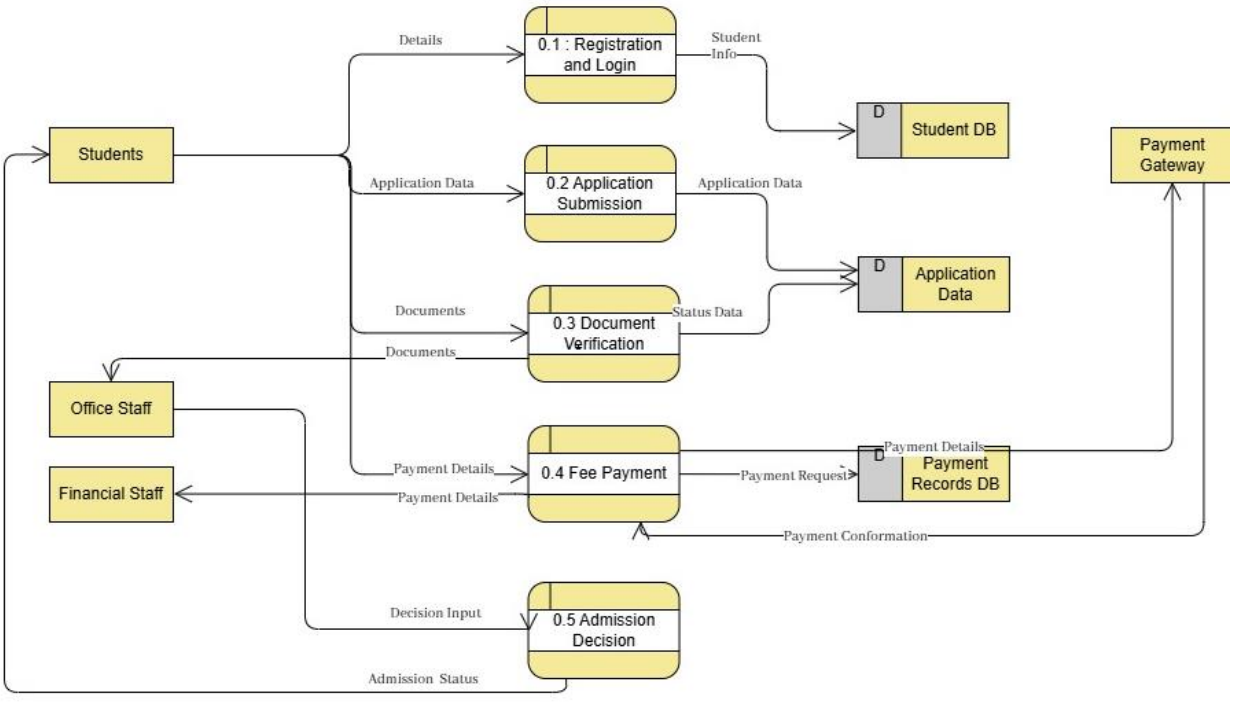
### 4.2.1 Context Level DFD (Level 0)

This shows the overall interaction of external entities (Student, Office Staff, Financial Staff) with the system.



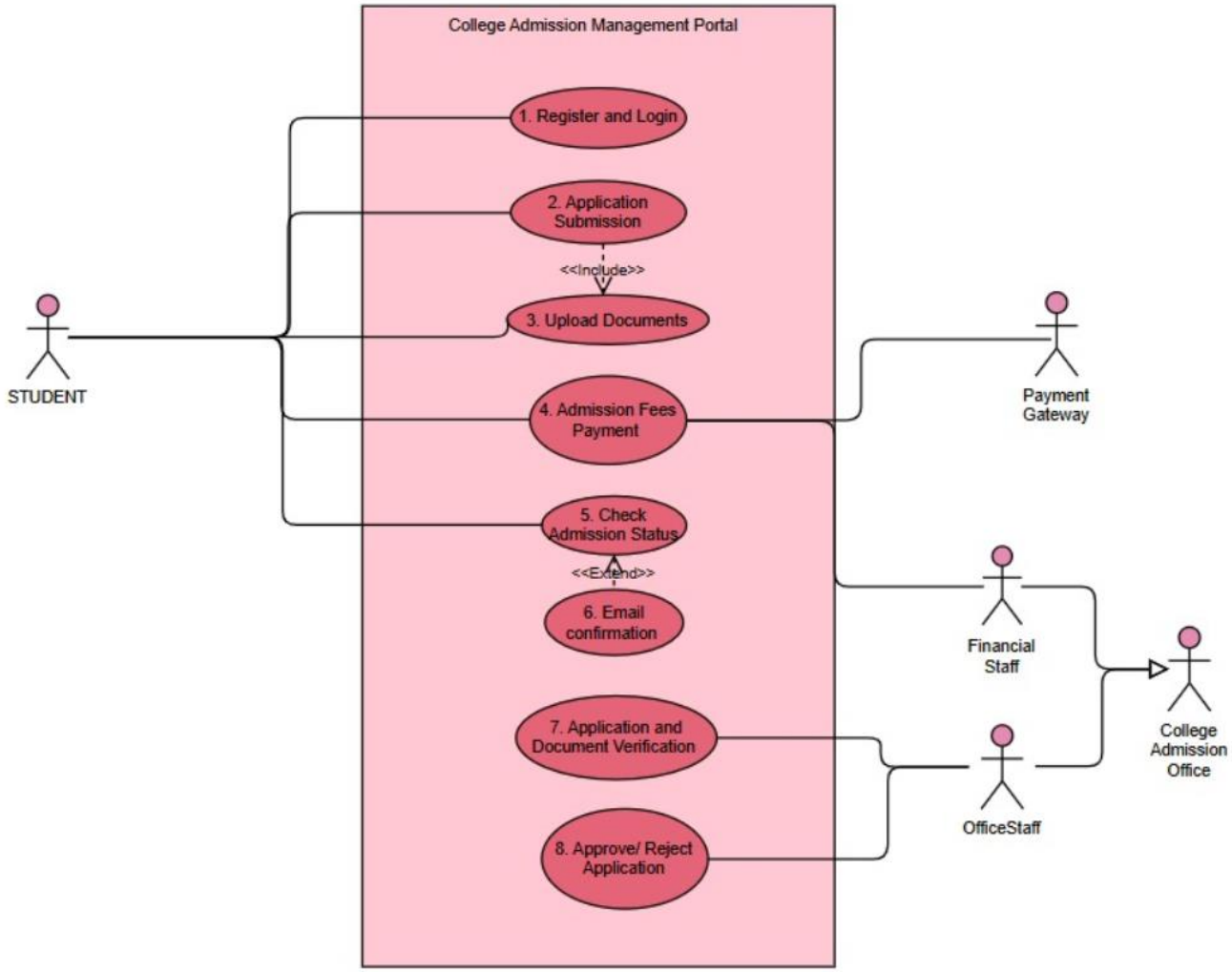

### 4.2.2 Level 1 DFD

Breakdown of main processes: Registration, Application Submission, Document Verification, Fee Management, Merit List, and Admission Confirmation.

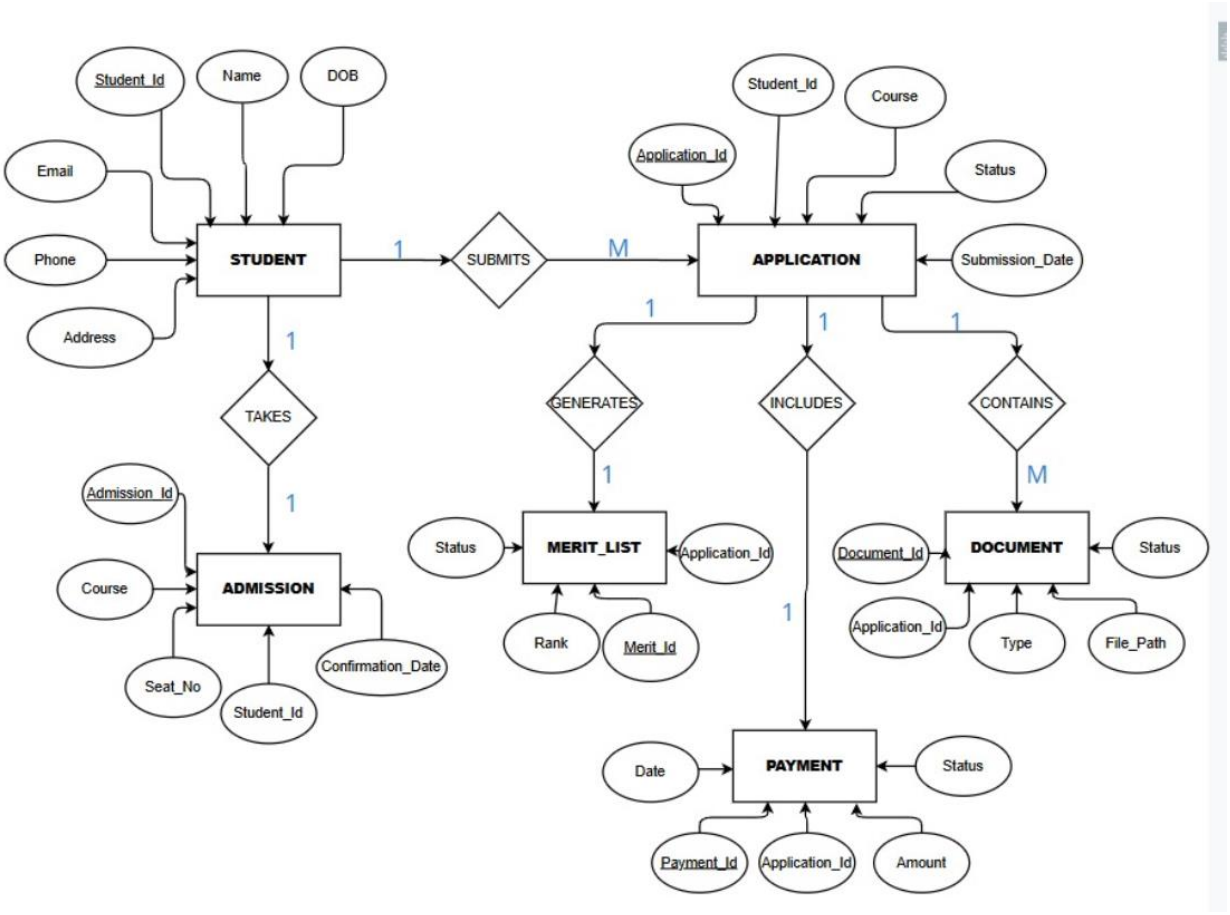


### 4.2.4 Use Case Diagram

Actors: **Student, Office Staff, Financial Staff**
Use cases: Register, Submit Application, Verify Documents, Process Payments, Generate Merit List, Allocate Seats, Confirm Admission, Notifications, Reporting.
Includes: <<include>> for mandatory steps (e.g., Login included in every operation).
Extends: <<extend>> for optional flows (e.g., Refund extends Payment).



### 4.2.5 ER Diagram (Entity Relationship Model)




### 4.2.6 Sequence Diagram (Optional – Advanced)

**Scenario: Student Admission**

1. Student logs in.
2. Submits application form.
3. System verifies form.
4. Office Staff verifies documents.
5. Student makes payment.
6. Financial Staff confirms transaction.
7. System generates merit list.
8. Student receives seat allocation and admission letter.

### 4.3 Index