COMSATS University Islamabad,
COMSATS Road, off GT Road, Sahiwal, Pakistan

SOFTWARE REQUIREMENTS SPECIFICATION

(SRS DOCUMENT)

for

Batch Advisory

Version 1.0

By

CUI/SP21-BSE-004/SWL
CUI/SP21-BSE-029/SWL
CUI/SP21-BSE-045/SWL

Supervisor

Ms. Mubeen Javid

Bachelor of Science in Computer Science (2021-2025)

Table of Contents

Revision History

Name	Date	Reason for changes	Version

Application Evaluation History

Comments (by committee) *include the ones given at scope time both in doc and presentation	Action Taken

Supervised by

Ms. Mubeen Javid

Signature_____

1 INTRODUCTION

The requirements for creating a batch advisory system are outlined in detail in the Software.

1.1 Purpose

The purpose of the batch advisory system is to streamline course allocations for academic advisors and provide students with a convenient platform to submit applications for academic requests and issues.

1.2 Scope

By giving batch advisors the resources, they need to effectively assign courses to students and oversee online applications, the student advisory system seeks to expedite academic advising procedures. To guarantee efficient communication between advisors and student, the system will incorporate functions like application submission, course allocation, and notification viewing for students.

2 OVERALL DESCRIPTION

2.1 Product perspective

The purpose of the stand-alone student advisory system is to improve academic advising procedures in educational establishments. This advanced technology is the next generation, designed to replace antiquated manual procedures with an automated and intuitive alternative.

2.2 Operating environment

The system shall operate correctly with the following web browsers: Windows Internet Explorer versions 7, 8, and 9; Firefox versions 12 through 26; Google Chrome (all versions); and Apple Safari versions 4.0 through 8.0.

2.3 Design and Implementation Constraints

- The system must be developed using different programming languages to integrate seamlessly with existing academic management systems.
- Integration with email services like Gmail for notifications must comply with security and privacy regulations.

3 Requirement Identifying Technique

We are using Use Case diagrams to elicit user requirements that is an effective technique for interactive and end-user applications.

3.1 Use Case Diagram

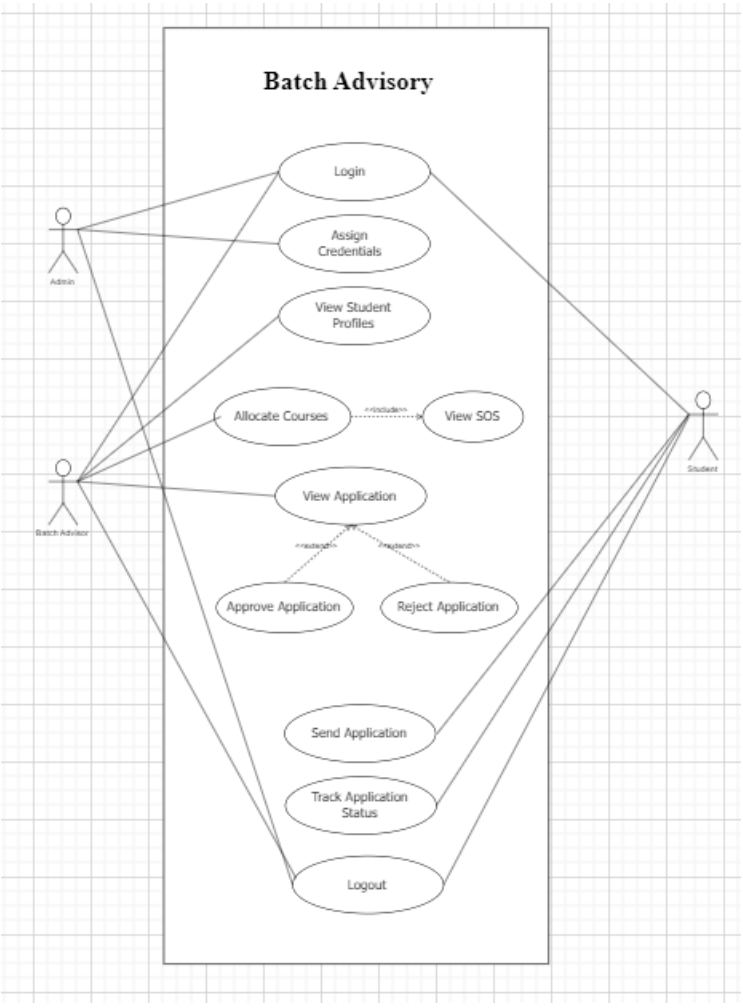

Figure 1: Use Case

Use case description.

Table 1: Use Case Description

Use Case Name:	Automate Batch Advisory
Actors:	• Primary Actor: Advisors and Students • Secondary Actors: Admin and University existing system
Description:	The use case diagram represents the interactions between advisory and student within the Course Allocation System and the Online Application Submission.

Preconditions:	PRE-1. Batch Advisors, Students, and University Administration actors are registered and authenticated within the system. PRE-2. The Course must be allocated based on the provided SOS.
Postconditions:	POST-1. Batch Advisors can successfully view student profiles, allocate courses, and send notifications related to course allocations. POST-2. Students can use a user-friendly interface to submit applications and track its status. POST-3. University Administration can track and monitor applications submitted by students.
Normal Flow:	**Batch Advisor View Student Profiles:** - Batch Advisor logs into the Course Allocation System. - Batch Advisor navigates to the "View Student Profiles" use case. - System displays a list of student profiles. - Batch Advisor selects a student profile to view. - System retrieves and displays the student's academic status, including completed courses, grades, and program progress. **Batch Advisor Allocate Courses:** - Batch Advisor navigates to the "Allocate Courses."

	- Batch Advisor selects individual students or groups based on program requirements and academic status. - Batch Advisor assigns appropriate courses to selected students. **Student Use User-Friendly Interface:** - Student logs into the Online Application Submission System. - Student navigates to the user-friendly interface for submitting applications. - System presents options for selecting application categories. - Student selects the appropriate category (e.g., academic, personal, administrative). **Student Track and Monitor Applications:** - Student navigates to the "Track and Monitor Applications" use case. - System displays a list of submitted applications and their status (e.g., pending, in progress, approved). - Student selects an application to view detailed information and progress updates.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	**Batch Advisor Allocate Courses:** • *Alternate Flow 1:* If a student requests a specific course assignment not typically included in their program requirements, the Batch Advisor may need to seek special approval from relevant stakeholders. **Student Track and Monitor Applications:** • Alternate Flow 2: If the Student needs to provide additional information or follow-up on a pending application, they may need to initiate direct communication with advisors through the system.
Exceptions:	1.0.*E1:* If a user forgets their password, they will need to contact their administrator. 2.0.*E2:* If the system encounters technical issues or downtime, Batch Advisors may be unable to access student profiles until the issue is resolved.

4 Specific Requirements

4.1 Functional Requirements

FR-1 Authentication

A valid email and login are required to login.

Credentials will be given by the Admin.

FR-2 View Student Profiles

Batch advisors should be able to access:

- student profiles and
- view their current academic status, including completed courses, grades, and program progress.

FR-3 Course Allocation

Batch advisors must have options to select and assign appropriate courses to individual students or groups based on their program requirements, academic status, and Scheme of Study (SOS).

FR-4 Select Application Category

Students should be able to select categories or types of issues (e.g., academic, personal, administrative) to streamline the submission process.

FR-5 Track Application Status

- Advisors should have tracking and monitoring functionalities to manage and prioritize pending applications efficiently.
- Student may also track the application status whether it is approved or rejected.

FR-6 Notify Users

- The system should send notifications to students and advisors about course allocations and any changes made to their study plans.

4.2 Non-Functional Requirements

Non-functional requirements may be more critical than functional requirements. If these are not met, the system is useless.

Developing an online batch advisory system using React.js for the front end and Node.js for the back end, the non-functional requirements to ensure the system's effectiveness, scalability, security, reliability, and usability are:

4.2.1 Availability

- Application must be responsive and available at every time. Availability of internet connection is a major requirement of the application.

4.2.2 Reliability

- **Error Handling**: Implement robust error handling mechanisms in both the frontend and backend to gracefully handle exceptions and provide meaningful error messages to users.
- **Logging and Monitoring**: Set up logging and monitoring tools to track system performance, detect errors, and troubleshoot issues proactively.

4.2.3 Maintainability

- **Modular Architecture**: Design the application with a modular architecture to facilitate code organization, reusability, and maintainability.
- **Documentation**: Provide comprehensive documentation for the front-end and backend codebases, including APIs, dependencies, and deployment instructions, to assist developers in understanding and maintaining the system.

4.2.4 Usability

- **Responsive Design:** Develop a responsive and mobile-friendly user interface using React.js to ensure usability across various devices and screen sizes.
- **Intuitive User Experience:** Design an intuitive user interface with clear navigation, consistent design patterns, and user-friendly workflows to enhance user satisfaction and productivity.

4.2.5 Portability

It's a web application that can be run on all android phones. So, the system is portable.

4.2.6 Performance

- **Optimized Client-Side Rendering:** Ensure efficient rendering of React components to minimize page load times and enhance user experience.
- **Server-Side Rendering (SSR):** Implement SSR to improve initial page load performance and facilitate search engine optimization (SEO).
- **API Performance:** Optimize API endpoints in Node.js to handle requests efficiently and minimize response times.

4.2.7 Scalability:

- **Horizontal Scaling:** Design the Node.js backend to be horizontally scalable, allowing the system to handle increasing loads by adding more server instances.
- **Database Scalability:** Choose a scalable database solution (e.g., MongoDB, PostgreSQL) and implement sharing or clustering to handle growing data volumes.

5 External Interface Requirements

5.1 Software Requirements

- Windows 8 or higher
- Visual studio 4.0.5
- Mongo Db Database

6 References

1. Soleh, O., Ariessanti, H. D., & Haryono, G. F. (2017, October). Wedding innovative application as a container to provide wedding preparation service: Development and application. In 2017 International Seminar on Application for Technology of Information and Communication (iSemantic) (pp. 121-125). IEEE.

2. Kumar, A., & Jain, R. (2015, October). Sentiment analysis and feedback evaluation. In 2015 IEEE 3rd International Conference on MOOCs, Innovation and Technology in Education (MITE) (pp. 433-436). IEEE.
3. Rogerson, J. M., & Wolfaardt, Z. (2016). Wedding tourism in South Africa: An exploratory analysis.
4. Farooq Butt, H., Akhtar, F., & Shaukat, T. (2017). Online Wedding

   Arrangement (Doctoral dissertation, University of Management and Technology).
5. Booth, J. R. (2010). U.S. Patent No. 7,725,402. Washington, DC: U.S. Patent and Trademark Office.