Software Requirements Specification Documents

Written By
Srikar(21MCME19), Akash(21MCME28).
Under the guidance of
Prof. Salman Abdul Moiz, Placement Head, UoH.

1. INTRODUCTION

1.1 Purpose

1.2 Scope

1.3 Definitions, Acronyms, Abbreviations

1.4 References

1.5 Overall Description

2. OVERALL DESCRIPTION

2.1 Product perspective

2.2 Product functions

2.3 User characteristics

2.4 User constraints

2.5 Assumptions & dependencies

2.6 Apportioning requirements

3. SPECIFIC REQUIREMENTS

3.1 Interface requirements

3.1.1 External Interface

3.1.2 Hardware Interface

3.1.3 Software Interface

3.1.4 Communication Interface

3.2 Functional requirements

3.2.1 Use case model

3.2.2 Use Case Specifications

3.2.3 Analysis Classes

3.3 Performance requirements

3.4 Logical database requirements

3.5 Design Constraints

3.6 Software System attributes

3.6.1 Reliability

3.6.2 Availability

3.6.3 Security

3.6.4 Maintainability

3.6.5 Portability

4. SUPPORTING INFORMATION

1. Introduction

1.1 Purpose

The Placement Management System aims to streamline the recruitment process for students and facilitate efficient management of job placements within the university. The primary purpose of this system is to provide a centralized platform where students can create profiles, view job opportunities, and apply for positions, while enabling placement coordinators to post job openings and access applicant details.

Additionally, the system allows the chairman to generate comprehensive reports and statistics related to placements across various academic years, departments, schools, streams, and program tenures.

The target audience for the Placement Management System includes:

1. Students: The system empowers students to create accounts, maintain resumes, explore job opportunities, and apply for positions that align with their interests and qualifications.

2. Placement Coordinators: Coordinators can leverage the system to post job openings, access applicant details, and streamline the recruitment process for various organizations.

3. Chairman/TPO: The system provides the chairman with valuable insights by generating detailed reports and statistics on placements, enabling data-driven decision-making and analysis.

1.2 Scope

The Placement Management System aims to achieve the following:

1. Student Profile Management: Enable students to create and maintain comprehensive profiles, explore job opportunities, and apply through a user-friendly interface.

2. Job Opportunity Management: Allow placement coordinators to post job openings with details like descriptions, eligibility criteria, and deadlines, and facilitate the application process.

3. Applicant Tracking and Management: Provide a centralized platform for placement coordinators to view and manage applications received for each job opening, including applicants' resumes/CVs.

4. Offer Letter Management: Enable students to upload and store offer letters received from organizations.

5. Reporting and Analytics: Generate comprehensive reports and statistics for the chairman, including overall placement statistics, department/school/stream-wise statistics, and program tenure-based statistics.

6. User Authentication and Authorization: Implement secure user authentication and authorization mechanisms for data privacy and access control.

1.3 Definitions, Acronyms, Abbreviations

Definitions:

Job Opportunity: A vacant position or role offered by an organization for which candidates can apply.

Applicant: A student who has submitted an application for a particular job opportunity.

Offer Letter: A formal document issued by an organization to a selected applicant, offering them employment and specifying the terms and conditions.

Placement: The process of securing employment for students through the university's placement program.

Acronyms and Abbreviations:

PMS	Placement Management System
CV	Curriculum Vitae
CGPA	Cumulative Grade Point Average
NIRF	National Institutional Ranking Framework
TPO	Training Placement officer

1.4 References

- "Software Requirements Specification (SRS) Document" by Tutorialspoint (https://www.tutorialspoint.com/software_engineering/software_requirements_specification.htm)
- "IEEE Recommended Practice for Software Requirements Specifications" by IEEE Computer Society (https://ieeexplore.ieee.org/document/720574)

1.5 Overview

Introduction

- Purpose: Outlines the intentions and target audience of the system.
- Scope: Defines the goals, objectives, and capabilities of the system.
- Definitions, Acronyms, and Abbreviations: Explains the terms and abbreviations used throughout the document.

Overall Description

- Product Perspective: Describes the context and interfaces of the system.
- Product Functions: Outlines the key features and functionalities of the system.
- User Characteristics: Defines the expected user types and their roles.
- Constraints and Assumptions: Lists any constraints, dependencies, or assumptions related to the system.

Specific Requirements

- Interface Requirements: Specifies the external interfaces, hardware, software, and communication interfaces.
- Functional Requirements: Details the use cases, information flows, data models, and process descriptions.
- Non-functional Requirements: Covers performance, database, design constraints, reliability, availability, security, maintainability, and portability requirements.

Supporting Information

Includes any additional information, such as diagrams, models, or references, that support the requirements specified in the document.

2. Overall Description

2.1 Product Perspective

PMS is an web-based application that will be developed specifically for the University of Hyderabad. It will provide a centralized platform for students, placement coordinators, the TPO , and the chairman to manage various aspects of the placement process.

2.2 Product Functions

The primary functions of the PMS include:

- Student profile management and job application

- Job opportunity posting and applicant tracking by placement coordinators

- Offer letter management for students

- Comprehensive reporting and analytics for the chairman.

2.3 User Characteristics

The PMS will have three main user groups:

1. Students: Enrolled in various programs at the university, with basic computer literacy and internet access.

2. Placement Coordinators: Faculty members or staff responsible for managing the placement process, with proficient computer skills and familiarity with recruitment processes.

3. Chairman/TPO: The administrative head overseeing the placement activities, with a good understanding of data analysis and decision-making processes.

2.4 User Constraints

- The system must be accessible through web browsers on various devices (desktops, laptops).

- User interfaces should be intuitive and user-friendly for all user groups.

- Data security and privacy measures must be implemented to protect sensitive information.

2.5 Assumptions and Dependencies

- The university will provide the necessary infrastructure (servers, databases, and network connectivity) to host and run the PMS.

- Students, placement coordinators, and the chairman,TPO should have access to the internet and compatible web browsers.

- Relevant data (student information, job opportunities, etc.) should be available for initial setup and population.

2.6 Apportioning Requirements

- The system should allow placement coordinators to select and shortlist students for specific job opportunities based on predefined criteria (CGPA, department, program tenure, etc.).

- Placement coordinators should be able to mark students as "placed" once they receive an offer letter, facilitating tracking and reporting.

- An confirmation email should be sent after a student tries to SignUp, and a forgot password mechanism has to be implemented.

3.1 Interface Requirements

3.1.1 External Interface

The Placement Management System (PMS) will have the following external interfaces:

1. User Interface: The system will provide a web-based user interface accessible through standard web browsers on various devices (desktops, laptops). The user interface will be responsive and user-friendly, with separate views and functionalities for students, placement coordinators, the chairman, and TPO.

3.1.2 Hardware Interface

The Placement Management System will be a web-based application, and therefore, it will not have any direct hardware interfaces.

The hardware requirements for accessing the PMS will be minimal, as long as the devices have a compatible web browser and an internet connection.

3.1.3 Software Interface

The Placement Management System will interact with the following software components:

1. Web Server: The PMS will be hosted on a web server, which will serve the application to clients (web browsers) over the internet or the university's intranet.

2. Database Management System (DBMS): The system will require a DBMS to store and manage data related to student profiles, job opportunities, applications, and other relevant information.

3. Web Browsers: The PMS will be accessible through compatible web browsers on various devices, such as Google Chrome, Mozilla Firefox, Microsoft Edge, Safari, and others.

4. Operating Systems: The PMS should be compatible with various operating systems, including Windows, macOS, and Linux as long as the devices have a compatible web browser.

3.1.4 Communication Interface

The Placement Management System will rely on the following communication interfaces:

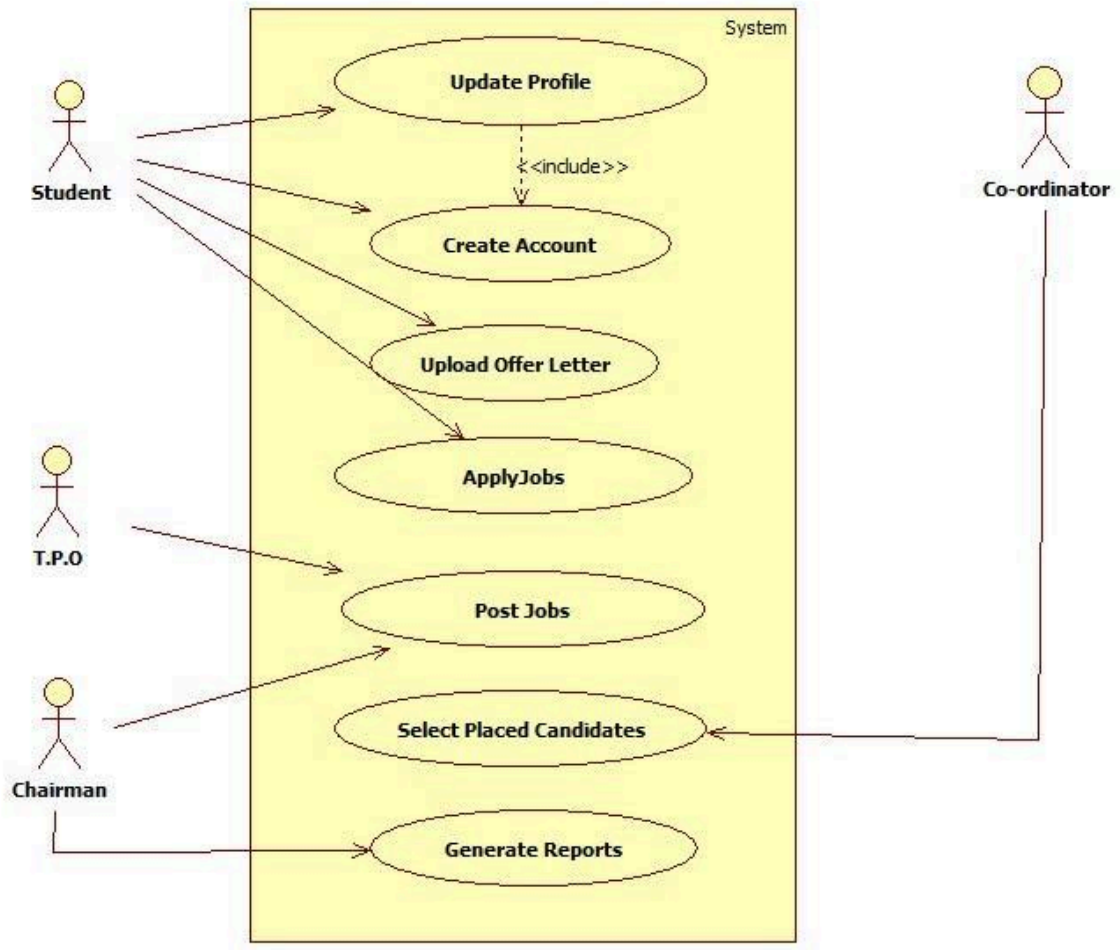
1. Internet/Intranet: The PMS will be accessible over the internet or the university's intranet, depending on the deployment strategy. Users will access the system through their web browsers using a secure connection (HTTPS).

2. Email Integration (Optional): The system may integrate with email services to facilitate communication between stakeholders, such as sending notification emails to students and placement coordinators.

3. Messaging Integration (Optional): The system may optionally integrate with messaging services (e.g., SMS, push notifications) to send alerts and notifications to users.

3.2 Functional Requirements

3.2.1 Use Case Model



3.2.2 Use Case Specifications

USE CASE TEMPLATE

Use Case ID:	1		
Use Case Name:	Post Job opportunity		
Created By:	Srikar , Akash	Last Updated By:	Srikar,Akash
Date Created:	20-02-24	Date Last Updated:	06-03-24

Actors:	Chairman, TPO
Description:	This Use case models the process of recruiter posting jobs
Preconditions:	Users should be logged in
Post conditions:	After successful verification, the job details are stored in the system.
Normal Flow:	1. The User opens the GUI for entering job details.The system displays input fields for Required Qualification, Number of Openings, Job Description, etc. 2. User enters Required Qualifications(A1) 3. User enters Number of Openings(A2) 4. User fills all other details and submits for verification.(A3) 5. If all details are valid, the system confirms the successful submission.
Alternative Flows:	A1.1. If the user enters an invalid qualification,the system displays an error message. A2.1 If the user enters an invalid number of openings, the system displays an error message. A3.1 System displays an error message if an error occurs in another field.
Exceptions:	-
Includes:	-
Priority:	High

Frequency of Use:	High
Business Rules:	-
Assumptions:	-

USE CASE TEMPLATE

Use Case ID:	2		
Use Case Name:	Upload Offer Letter		
Created By:	Srikar , Akash	Last Updated By:	Srikar,Akash
Date Created:	01-03-24	Date Last Updated:	06-03-24

Actors:	Student
Description:	This Use case models the process of students uploading the offer letter received from the companies
Preconditions:	The student is placed in a certain company.
Post conditions:	The system stores the uploaded offer letter.
Normal Flow:	1. Select the job application they are placed into. 2. User uploads the offer letter(A1). 3. User submits the uploaded offer letter.
Alternative Flows:	A1.1 User uploads files with extensions other than pdf. Check and re-upload the file.
Exceptions:	-
Includes:	-
Priority:	Low
Frequency of Use:	Low
Business Rules:	-
Assumptions:	-

USE CASE TEMPLATE

Use Case ID:	3		
Use Case Name:	Create Account		
Created By:	Srikar , Akash	Last Updated By:	Srikar,Akash
Date Created:	23-02-24	Date Last Updated:	06-03-24

Actors:	Student
Description:	This Use case models the process of users creating accounts on placement management portal.
Preconditions:	-
Post conditions:	Account Successfully Created.
Normal Flow:	1. The User opens the GUI for creating the account 2. User enters details and submits the form (A1) 3. User will prompted whether account creation is successful or unsuccessful.
Alternative Flows:	A1.1 User enters invalid details. Check and refill form
Exceptions:	-
Includes:	-
Priority:	High
Frequency of Use:	High
Business Rules:	-
Assumptions:	-

USE CASE TEMPLATE

Use Case ID:	4		
Use Case Name:	Select Placed Candidates		
Created By:	Srikar , Akash	Last Updated By:	Srikar,Akash
Date Created:	01-03-24	Date Last Updated:	06-03-24

Actors:	Coordinator
Description:	This Use case models the process of the coordinator selecting all the candidates who have obtained an offer letter
Preconditions:	-
Post conditions:	The system stores the updated placement information
Normal Flow:	1. The User opens the GUI for selecting placed students. 2. User marks the students who have uploaded the offer letters from the list of applied students provided by the company. 3. User confirms the selection of placed students. 4. The system records the updated placement status in the database.
Alternative Flows:	-
Exceptions:	-
Includes:	-
Priority:	Low
Frequency of Use:	Medium
Business Rules:	-
Assumptions:	-

USE CASE TEMPLATE

Use Case ID:	5		
Use Case Name:	Update profile		
Created By:	Srikar , Akash	Last Updated By:	Srikar,Akash
Date Created:	01-03-24	Date Last Updated:	06-03-24

Actors:	Student
Description:	This Use case models the process of the students updating the personal details into the system.
Preconditions:	The student should have an account in the system.
Post conditions:	The system updates the student's profile with the modified information.
Normal Flow:	1. The system displays the student's current details such as name, ID number, email, average CGPA, resume/CV, department, school, etc. 2. User select the option to modify their profile. 3. User modifies the particular details(A1) 4. User submit the modified information to the system.
Alternative Flows:	A1.1 User enters invalid details. Check and refill form
Exceptions:	-
Includes:	-
Priority:	Medium
Frequency of Use:	Low
Business Rules:	-
Assumptions:	-

USE CASE TEMPLATE

Use Case ID:	6		
Use Case Name:	Generate Reports		
Created By:	Srikar , Akash	Last Updated By:	Srikar,Akash
Date Created:	01-03-24	Date Last Updated:	06-03-24

Actors:	Chairman/TPO
Description:	This Use case models the process of the chairman generating reports which contains all the details such as median,max,min etc.
Preconditions:	The user is a valid user of the system
Post conditions:	The system stores the updated placement information
Normal Flow:	1. User has an interface/GUI for generating placement reports. 2. User chooses a specific school (e.g., School of Social Sciences) from the available options. 3. If applicable, User selects a specific department within the chosen school (e.g., SCIS).If not, the user skips this step. 4. If an applicable user selects the tenure of the program (e.g., five-year or six-year programs).If not, the user skips this step. 5. The user initiates the report generation process based on the selected criteria. 6. The system displays the generated statistical report to the user.
Alternative Flows:	-
Exceptions:	-
Includes:	-
Priority:	Medium

Frequency of Use:	Medium
Business Rules:	-
Assumptions:	-

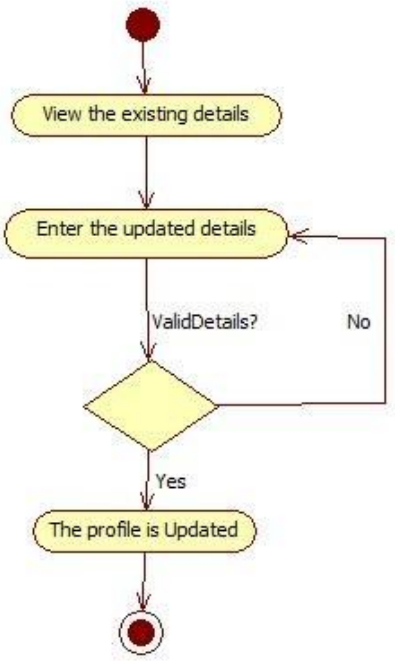
USE CASE TEMPLATE

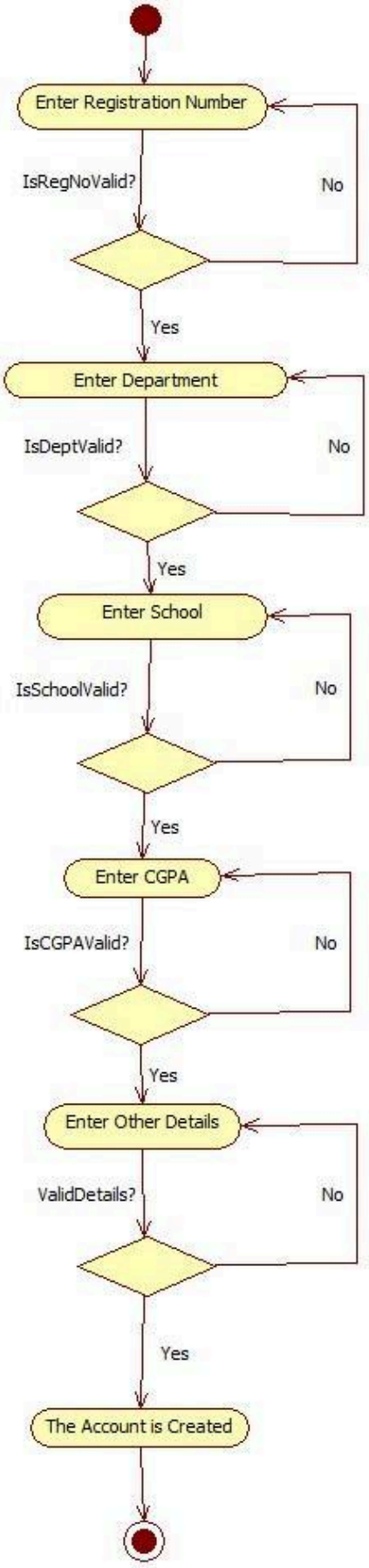
Use Case ID:	7		
Use Case Name:	Apply Jobs		
Created By:	Srikar , Akash	Last Updated By:	Srikar,Akash
Date Created:	01-03-24	Date Last Updated:	06-03-24

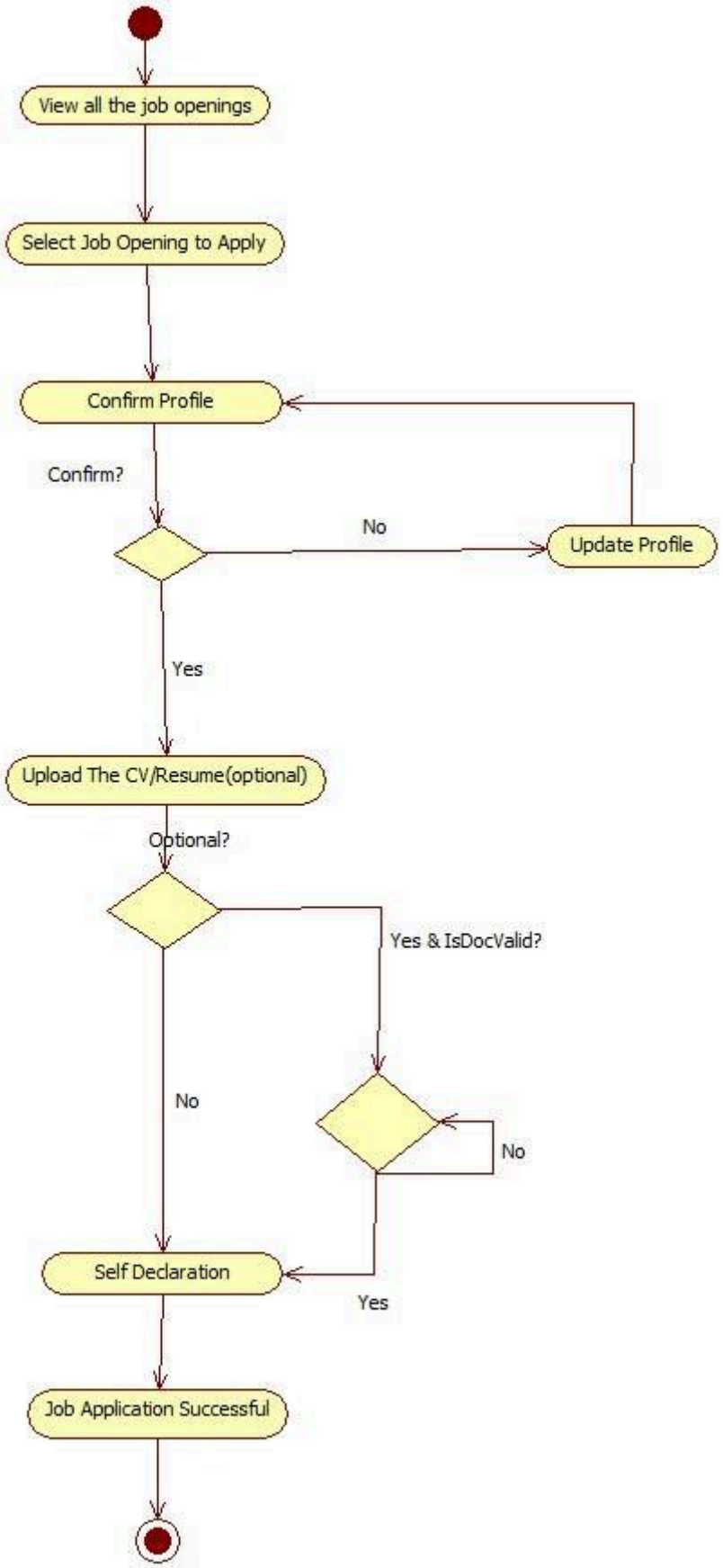
Actors:	Student
Description:	This Use case models the process of the students applying the job opportunities available.
Preconditions:	The user should have an account in the system.
Post conditions:	The system stores the the application of the student for the job
Normal Flow:	1. User has an interface/GUI for applying to the job opportunities. 2. The user selects the job visible on their dashboard. 3. The user fills the necessary details for the application.(A1) 4. The user uploads the resume/CV if needed. 5. The user provides the declaration of sincerity.(A2)
Alternative Flows:	A1.1 User enters invalid details. Check and refill form. A2.1 The application won't be accepted if this is not provided. The User is prompted with the same.
Exceptions:	-

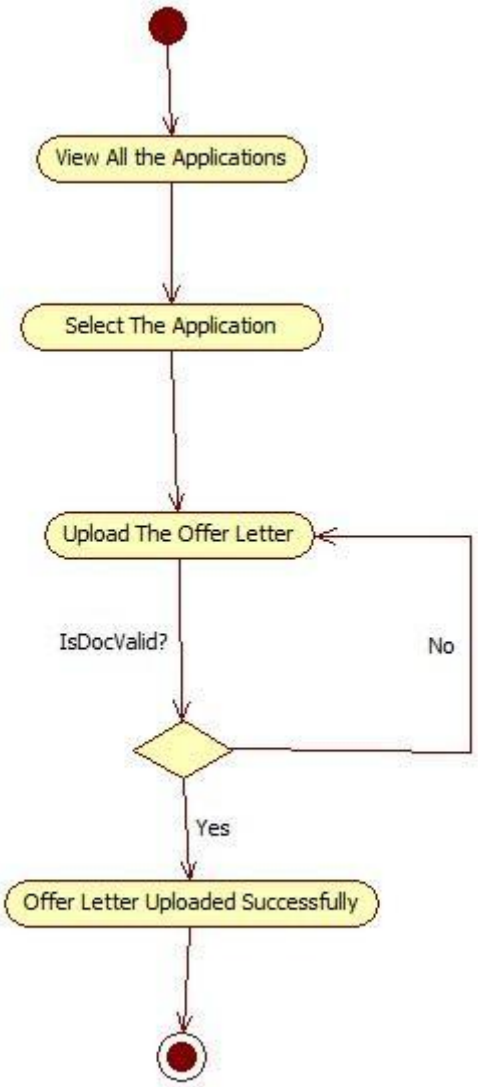
Includes:	-
Priority:	High
Frequency of Use:	High
Business Rules:	-
Assumptions:	-

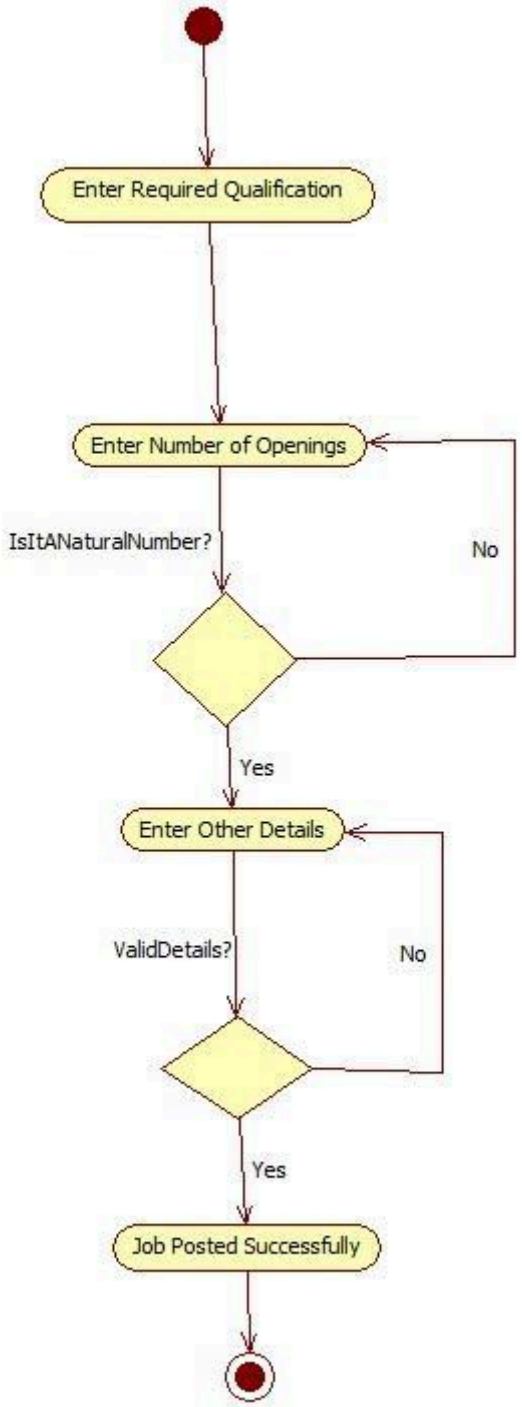
3.2.3 Activity Diagram

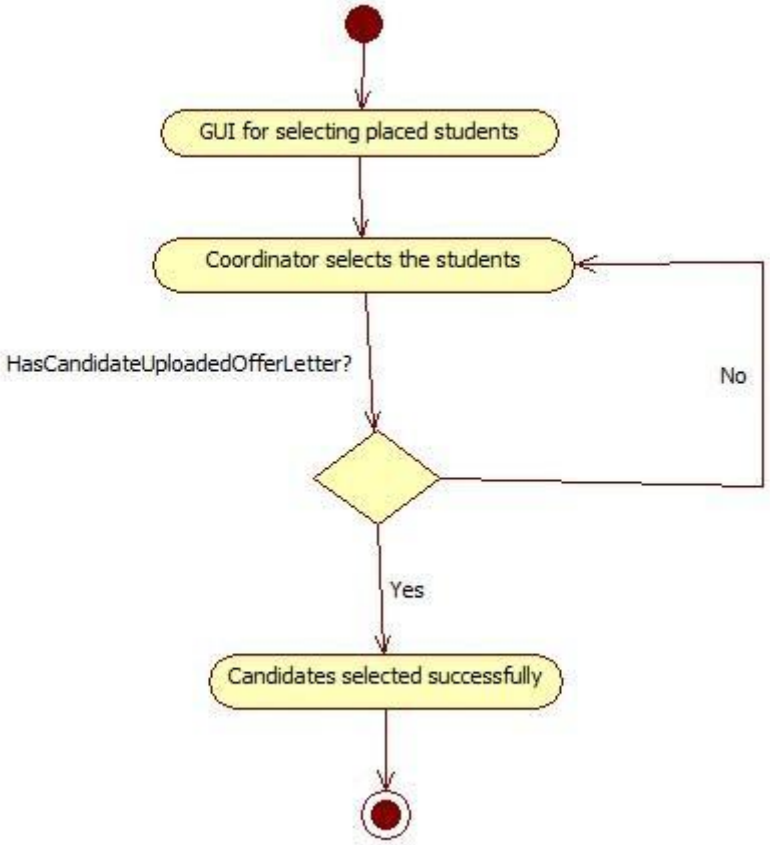


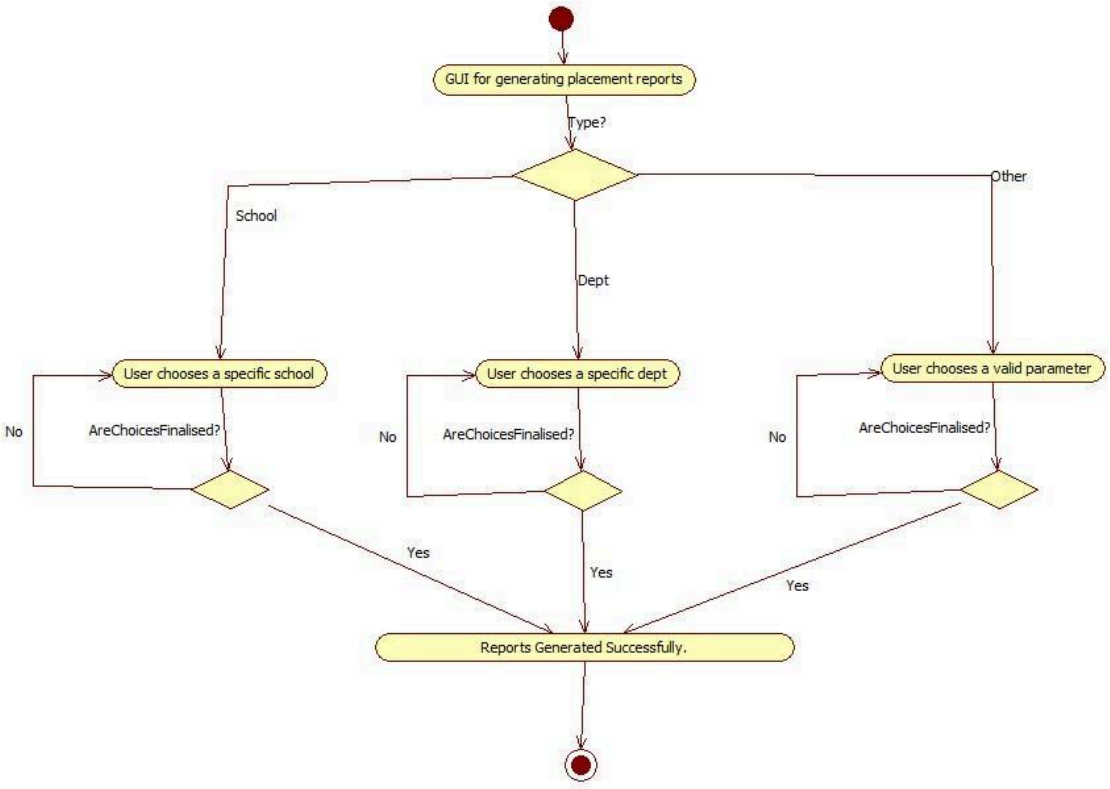

Enter Registration Number
IsRegNoValid?
No
Yes
Enter Department
IsDeptValid?
No
Yes
Enter School
IsSchoolValid?
No
Yes
Enter CGPA
IsCGPAValid?
No
Yes
Enter Other Details
ValidDetails?
No
Yes
The Account is Created


View all the job openings
Select Job Opening to Apply
Confirm Profile
Confirm?
No
Update Profile
Yes
Upload The CV/Resume(optional)
Optional?
Yes & IsDocValid?
No
No
Self Declaration
Yes
Job Application Successful


View All the Applications
Select The Application
Upload The Offer Letter
IsDocValid?
No
Yes
Offer Letter Uploaded Successfully


Enter Required Qualification
Enter Number of Openings
IsItANaturalNumber?
No
Yes
Enter Other Details
ValidDetails?
No
Yes
Job Posted Successfully


GUI for selecting placed students
Coordinator selects the students
HasCandidateUploadedOfferLetter?
No
Yes
Candidates selected successfully


GUI for generating placement reports
Type?
School
Dept
Other
User chooses a specific school
User chooses a specific dept
User chooses a valid parameter
No
AreChoicesFinalised?
No
AreChoicesFinalised?
No
AreChoicesFinalised?
Yes
Yes
Yes
Reports Generated Successfully.

3.2.4 Analysis Classes

1. Student

- Attributes: StudentID, Name, Email, Password, Contact Information

- Relationships:

  - Students have a many-to-many relationship with Jobs

  - TPOs, Chairman, and Coordinators may have specific permissions and responsibilities

2. Job

- Attributes: JobID, Title, Description, Requirements, Company, Location, Salary, Date Posted

- Relationships:

  - Jobs are created by TPOs and Chairman

  - Job has a one-to-many relationship with Applications

3. Application

- Attributes: ApplicationID, StudentID, JobID, Status (e.g., Pending, Accepted, Rejected, Selected), Date Submitted

- Relationships:

  - Application belongs to a Student and a Job

  - Applications are created by Students and managed by Coordinators

4. Coordinator

- Attributes: CoordinatorID, Email

- Relationships:

  - Coordinator is responsible for managing and updating the status of job applications and offer letters uploaded by students

3.3 Performance Requirements

1. Concurrent Users: Handle minimum 100 concurrent users during peak periods, scalable for increasing users.

2. Data Storage and Retrieval: Maximum 4 second query response time, optimized for fast data retrieval, especially for reports and statistics.

3. File Upload/Download: Minimum 1 Mbps speed, reliable and secure process with error handling and validation.

4. System Availability: 95% uptime during business hours, scheduled maintenance communicated in advance, minimized downtime.

3.4 Logical Database Requirements

Database Structure

1. Jobs Table

- Stores job opening details

- Attributes: JobID, OrgName, OrgDesc, JobName, JobDesc, Qualification, JobLocation, NumberOfOpenings, SkillsRequired, LastDateToApply, GraduationYear, Departments

2. Student Table

- Contains student information

- Attributes: StudentID, ContactNumber, Email, DepartmentID, Password, SchoolID, Gender, AverageCGPA, Course, StartYear, EndYear

3. Department Table

- Stores department details

- Attributes: DepartmentID, DepartmentName, SchoolID

4. Coordinator Table

- Contains coordinator login credentials

- Attributes: Email, Password, DepartmentID

5. TPO Table

- Stores TPO login credentials

- Attributes: Email, Password

6. Chairman Table

- Contains chairman login credentials

- Attributes: Email, Password

7. School Table

- Stores school or faculty information

- Attributes: SchoolID, SchoolName

8. Application Table

- Tracks job applications submitted by students

- Attributes: ApplicationID, StudentID, JobID, AppliedOn, URL, Status

9. Accepted Table

- Stores details about accepted job offers

- Attributes: ApplicationID, Department, School, CTC (Cost To Company), Duration, Placed_Year

Relationships

- One-to-Many (Jobs to Applications)

- Many-to-One (Students to Applications)

- One-to-Many (Departments to Coordinators)

- Many-to-One (Departments to Students)

- Many-to-One (Departments to Schools)

- One-to-One (TPO, Chairman to Departments)

3.5 Design Constraints

1. Web-based Architecture: The Placement Management System should be developed as a web-based application to ensure cross-platform compatibility and ease of access.

2. Compatibility: The system should be compatible with the latest versions of popular web browsers, such as Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari.

3.6 Software System Attributes

3.6.1 Reliability

- The system should have a high level of reliability, with minimal downtime and data loss.

- Appropriate error handling and validation mechanisms should be implemented to ensure the integrity of data and prevent system failures..

3.6.2 Availability

- The system should be available, with a minimum uptime of 95% during regular business hours.

- Scheduled maintenance windows should be communicated to users in advance, and downtime should be minimized.

3.6.3 Security

- The system should implement secure authentication and authorization mechanisms to protect user data and prevent unauthorized access.

3.6.4 Maintainability

- The system should be designed with a modular and extensible architecture to facilitate future enhancements and modifications.

- Proper documentation and coding standards should be followed to ensure the system's maintainability.

3.6.5 Portability

- The system should be portable and able to run on different operating systems and hardware platforms, as long as the necessary software/hardware dependencies are met.

4. Supporting Information

- UOHYD Brochure 2023-2024 (http://acad.uohyd.ac.in/download.html)
- UOHYD portal (https://uohyd.ac.in/)