

---

# **Software Requirements and Design Document**

for

**<Employee recruitment and  
management system>**

Prepared by <wajiha adnan,iffah naveed,eman shakeel>

<11/25/2024>

# Table of Contents

<b>Table of Contents</b>	<b>ii</b>
<b>1. Introduction</b>	<b>1</b>
1.1 Purpose	1
1.2 Product Scope	1
1.3 Title	1
1.4 Objectives	1
1.5 Problem Statement	1
<b>2. Overall Description</b>	<b>1</b>
2.1 Product Perspective	1
2.2 Product Functions	2
2.3 List of Use Cases	2
2.4 Extended Use Cases	2
2.5 Use Case Diagram	2
<b>3. Other Nonfunctional Requirements</b>	<b>2</b>
3.1 Performance Requirements	2
3.2 Safety Requirements	2
3.3 Security Requirements	2
3.4 Software Quality Attributes	2
3.5 Business Rules	3
3.6 Operating Environment	3
3.7 User Interfaces	3
<b>4. Domain Model</b>	<b>3</b>
<b>5. System Sequence Diagram</b>	<b>3</b>
<b>6. Sequence Diagram</b>	<b>3</b>
<b>7. Class Diagram</b>	<b>4</b>
<b>8. Package Diagram</b>	<b>4</b>
<b>9. Deployment Diagram</b>	<b>4</b>

# 1. Introduction

## 1.1 Purpose

The purpose of the **Employee Recruitment and Management System (ERMS)** is to provide a comprehensive, automated platform that streamlines and enhances the recruitment lifecycle for small to medium-sized tech enterprises. This system is designed to efficiently manage all stages of the recruitment process, from posting job openings to finalizing hiring decisions. By centralizing and automating key functions such as candidate management, evaluation, shortlisting, communication, and report generation, the ERMS enables recruiters and HR managers to make faster, more informed decisions.

The system also facilitates seamless communication between applicants, recruiters, and HR managers, ensuring that all stakeholders are kept up-to-date on the progress of applications. Additionally, the ERMS provides tools for managing applicant recommendations, generating detailed reports on recruitment activities, and handling contractual agreements with new hires.

Ultimately, the goal of the ERMS is to simplify and optimize the recruitment process for tech companies, allowing them to attract and retain top talent in a highly competitive industry, while improving efficiency, reducing human error, and enhancing overall user experience for both recruiters and applicants.

## 1.2 Product Scope

This project is focused on developing a **comprehensive Employee Recruitment and Management System (ERMS)**, tailored for small to medium-sized tech enterprises. The system offers a robust digital platform to manage the entire recruitment lifecycle, integrating applicant management, recruiter operations, and HR recommendations. Key processes include **job posting, candidate applications, written and interview evaluations, shortlisting, test administration, and hiring decisions**. Additionally, the system encompasses functionalities such as **recommendation management, contract management, report generation, and streamlined communication between applicants, recruiters, and HR managers**.

The ERMS is specifically designed to address the unique challenges of tech enterprises, where recruitment demands are fast-paced and competition for top talent is intense. By automating and integrating key recruitment processes, the ERMS aims to enhance decision-making, optimize resource allocation, and improve the overall hiring experience.

## 1.3 Title

Employee Recruitment and Management System for Tech Enterprises (ERMS).

## 1.4 Objectives

The project has the following objectives:

- To automate and digitize recruitment processes, reducing manual effort and human error.

- To enable seamless communication between applicants, recruiters, and HR managers.
- To streamline evaluations and shortlisting through written tests and interviews.
- To enhance the efficiency of recommendation-based hiring and contract management.
- To provide real-time insights and reports to stakeholders, aiding in data-driven decision-making.

## 1.5 Problem Statement

*Efficient recruitment and employee management are critical for tech enterprises to thrive in a highly competitive environment. Traditional recruitment practices, characterized by manual resume screening, interview scheduling, and paperwork, present significant challenges:*

1. **Inefficient Hiring Processes:** Outdated manual methods prolong hiring cycles, delaying the acquisition of critical talent needed for projects and business growth.
2. **High Operational Costs:** Manual and disjointed processes increase administrative overhead, consuming valuable resources that could be redirected to strategic initiatives.
3. **Loss of Top Talent:** Slow and inefficient recruitment processes often result in losing skilled candidates to competitors with streamlined hiring workflows.
4. **Lack of Centralized Communication:** Ineffective communication between applicants, recruiters, and HR leads to delays and potential misunderstandings during the recruitment lifecycle.

*The **ERMS** addresses these pain points by providing a fully integrated and automated solution that supports seamless collaboration, efficient evaluations, and data-driven hiring decisions.*

## 2. Overall Description

### 2.1 Product Perspective

*The ERMS is a **new, self-contained system** that integrates core recruitment processes with advanced post-hiring functionalities, designed for small to medium-sized tech enterprises. It bridges the gap between applicants, recruiters, and HR by providing specialized modules for their respective tasks.*

*The system serves as a digital replacement for traditional, manual HR methods. It is composed of interconnected subsystems:*

1. **Applicant Portal:** For candidates to manage their accounts, apply for jobs, view application status, and take evaluations.
2. **Recruiter Portal:** To manage job postings, applications, and test administration, and conduct evaluations.
3. **HR Module:** To manage recommendations, oversee hiring policies, and generate contracts.
4. **Admin Module:** For generating reports and overseeing system functionalities.

**Subsystem Interconnections:**

3. *The system supports communication between applicants and recruiters (e.g., messaging and job applications).*
4. *The HR module provides policy guidance and recommendations to recruiters.*
5. *The Admin module ensures system oversight and report generation.*

## 5.1 Product Functions

*The **Employee Recruitment and Management System (ERMS)** streamlines recruitment and employee management for tech enterprises. **Applicants** can manage profiles, view and apply for jobs, track application status, take evaluations, and communicate with recruiters. **Recruiters** handle job postings, applicant shortlisting, evaluations, job reports, and messaging. **HR managers** oversee recruitment policies, provide recommendations, and manage contracts, including salaries and benefits. **Admins** generate system-wide reports, such as hiring trends and applicant statistics. Cross-functional features include real-time data synchronization, messaging, and role-specific dashboards, ensuring efficient collaboration and seamless operations.*

## 5.2 List of Use Cases

**Applicant:**

- Manage account (update profile information).
- View and apply for available job postings.
- view contract.
- view portfolio.
- View application status.
- Communicate with recruiters and hr.

**Recruiter:**

- Manage account details.
- Post and manage job openings.
- Manage applicant shortlisting.
- Organize and conduct tests (written and interviews).
- book appointments
- Communicate with applicants .

**HR:**

- make contract.
- Provide recommendations for applicants.
- Communicate with applicants.

**Admin :**

- Generate reports for each actor.

### 5.3 Extended Use Cases

#### APPLICANT:-

Member name:-iffah naveed (22i0974)

#### USECASE1:-

Use case	Details
Use Case Name	Manage User Accounts
Use Case id	UC-001
Scope	Employee Recruitment and Management System
Level	User Goal
Actor	Applicant, Recruiter, HR Manager
Stakeholders and Interests	<b>-User (Applicant, Recruiter, HR Manager):</b> Needs to manage their account information, including creating a new account, updating existing information, or deleting their account. <b>- The System:</b> Must ensure that account information is stored securely, validated properly, and account actions (create, update, delete) are handled accurately.
Preconditions	- For account creation: The user is not yet registered in the system. - For account updates or deletion: The user must be logged in, and their account exists in the system.
Postconditions	- The user's account is created and securely stored in the system.

Main flow	<b>Actor(Action)</b>	<b>System(response)</b>
	1. User selects the option to create new account	
		2. Displays the registration form requesting necessary details (e.g., name, email, password, role).
	3. User fills out the registration form with required details.	
		4. Validates the entered information (e.g., email format, password strength, required fields).
	5.User submits the request.	
		6.Saves the account information in the database. Generates initial setup as needed for the user’s role.
	7. User receives confirmation of the successful action.	
		8. Sends confirmation of successful account creation (e.g., email or on-screen message).
Exceptions	<p>- <b>4a.</b> Validation fails (e.g., invalid email, missing required fields, weak password). Prompts the user to correct invalid or missing information.</p> <p>- <b>4b.</b>User corrects the errors and resubmits the form. - Validates the resubmitted form and proceeds to save the account if all inputs are valid.</p>	

**USECASE 2:-**

Use case	Details												
Use Case Name	View and apply for job												
Usecase id	UC-002												
Scope	Employee Recruitment and Management System												
Level	User Goal												
Actor	Applicant												
Stakeholders and Interests	<ul style="list-style-type: none"> <li>- <b>Applicant:</b> Wants to find and view relevant job postings quickly.</li> <li>- <b>Recruiter:</b> Needs their job postings to be visible and attract the right talent.</li> <li>- <b>HR Manager:</b> Wants to assess applicant interest in various job postings and ensure the postings attract suitable candidates.</li> <li>- <b>System:</b> Needs to ensure job postings are accurately displayed with filtering options for easier access.</li> </ul>												
Preconditions	The applicant is registered and logged into the system.												
Postconditions	The applicant has access to the list of job postings with relevant details.												
Main flow	<table border="1"> <thead> <tr> <th>Actor(Action)</th><th>System(response)</th></tr> </thead> <tbody> <tr> <td>1. Navigates to the "Job Listings" section.</td><td></td></tr> <tr> <td></td><td>2. Displays a list of job postings with ids.</td></tr> <tr> <td>3. Selects a job to view details.</td><td></td></tr> <tr> <td></td><td>4. presents the job details, including description and requirements.</td></tr> <tr> <td>5.Chooses to apply for the job or save it for later.</td><td></td></tr> </tbody> </table>	Actor(Action)	System(response)	1. Navigates to the "Job Listings" section.			2. Displays a list of job postings with ids.	3. Selects a job to view details.			4. presents the job details, including description and requirements.	5.Chooses to apply for the job or save it for later.	
Actor(Action)	System(response)												
1. Navigates to the "Job Listings" section.													
	2. Displays a list of job postings with ids.												
3. Selects a job to view details.													
	4. presents the job details, including description and requirements.												
5.Chooses to apply for the job or save it for later.													



		6. if requirements match with the job then application submits.
Exceptions	- 1a. If no jobs match the selected filters, the system displays a message indicating no available postings. - 3a. if the deadline has passed or the requirements don't match then error message is shown	

**USECASE 3:-**

Use case	Details
Use Case Name	view contract
Usecase id	UC-003
Scope	Employee Recruitment and Management System
Level	User Goal
Actor	Applicant
Stakeholders and Interests	- <b>Applicant:</b> Wants to view the contract if they are offered the job. - <b>HR Manager:</b> Needs to ensure the contract is available and accessible to the hired applicant. - <b>System:</b> Must securely store and provide the contract to successful applicants while maintaining confidentiality.
Preconditions	-The applicant has completed the application and evaluation process.  - The applicant has been offered the job.  - The HR has prepared and uploaded the contract.
Postconditions	The applicant successfully views or downloads their job contract.

Main flow		
	<b>Actor(Action)</b>	<b>System(response)</b>
	1.HR uploads the contract to the system for the selected applicant.	
		2.Stores the contract securely and links it to the applicant's profile.
	3.Applicant logs into the system.	
	4.Applicant navigates to the contract section.	
		5. Displays the uploaded contract for review.
Exceptions	<p>- 1a. HR has not uploaded the contract yet.</p> <p>Displays a message that the contract is not yet available and notifies HR.</p> <p>- 5a. Applicant encounters an error while viewing the contract.</p> <p>Logs the issue and prompts the user to retry or contact support.</p>	

**USECASE 4:-**

Use case	Details
Use Case Name	view appointment
Usecase id	UC-004
Scope	Employee Recruitment and Management System

Level	User Goal												
Actor	Applicant												
Stakeholders and Interests	<ul style="list-style-type: none"> <li>- <b>Applicant:</b> Wants to view and attend their scheduled appointment successfully.</li> <li>- <b>Recruiter:</b> Needs a reliable system to schedule appointments and communicate with applicants effectively.</li> <li>- <b>System:</b> Must facilitate appointment scheduling, notify relevant parties, and maintain records.</li> </ul>												
Preconditions	- The recruiter has scheduled an appointment for the applicant.												
Postconditions	-The applicant successfully views their scheduled appointment details.												
Main flow	<table border="1"> <thead> <tr> <th>Actor(Action)</th><th>System(response)</th></tr> </thead> <tbody> <tr> <td>1.Recruiter schedules an appointment for the applicant.</td><td></td></tr> <tr> <td></td><td>2. Saves the appointment details and links it to the applicant's profile.</td></tr> <tr> <td>3. Applicants log into the system to view their appointment.</td><td></td></tr> <tr> <td></td><td>4.Displays the appointment details on the applicant's dashboard.</td></tr> <tr> <td></td><td></td></tr> </tbody> </table>	Actor(Action)	System(response)	1.Recruiter schedules an appointment for the applicant.			2. Saves the appointment details and links it to the applicant's profile.	3. Applicants log into the system to view their appointment.			4.Displays the appointment details on the applicant's dashboard.		
Actor(Action)	System(response)												
1.Recruiter schedules an appointment for the applicant.													
	2. Saves the appointment details and links it to the applicant's profile.												
3. Applicants log into the system to view their appointment.													
	4.Displays the appointment details on the applicant's dashboard.												
Exceptions	<ul style="list-style-type: none"> <li>- 2a. Applicant does not receive the appointment.</li> </ul> <p>Allows the applicant to check their appointment manually in the system.</p>												

**USERCASE 5 :-**

Use case	Details
Use Case Name	Viewportfolio
Usecase id	UC-005
Scope	Employee Recruitment and Management System
Level	User Goal
Actor	applicant
Stakeholders and Interests	<p><b>Applicant:</b> Wants to view and track their profile information, including qualifications, CGPA, and HR recommendations.</p> <p>- <b>HR:</b> Wants to ensure that applicants have access to the relevant details, including recommendations, that will aid in their future applications.</p> <p>- <b>System:</b> Must allow applicants to view their personal information and HR recommendations in an organized manner.</p>
Preconditions	<p>- The applicant is logged into the system.</p> <p>- The applicant has a profile with relevant data (e.g., qualifications, CGPA) and HR recommendations.</p>
Postconditions	The applicant successfully views their profile details, qualifications, CGPA, and HR recommendations.

Main flow	<table><tr><th>Actor(Action)</th><th>System(response)</th></tr><tr><td>1. Applicant logs into the system.</td><td></td></tr><tr><td></td><td>2. System displays the applicant's dashboard.</td></tr><tr><td>3. Applicant selects "view portfolio" from the menu.</td><td></td></tr><tr><td></td><td>4.System retrieves and displays a list of all applications submitted by the applicant</td></tr><tr><td>5.Applicant selects a specific application to view details.</td><td></td></tr><tr><td></td><td>6.System displays the applicant's profile page, showing personal details such as name, description, qualifications, and CGPA.</td></tr><tr><td>7.Applicant selects "view Recommendations" .</td><td></td></tr><tr><td></td><td>8.System retrieves and displays any recommendations made by HR, along with the context .</td></tr></table>	Actor(Action)	System(response)	1. Applicant logs into the system.			2. System displays the applicant's dashboard.	3. Applicant selects "view portfolio" from the menu.			4.System retrieves and displays a list of all applications submitted by the applicant	5.Applicant selects a specific application to view details.			6.System displays the applicant's profile page, showing personal details such as name, description, qualifications, and CGPA.	7.Applicant selects "view Recommendations" .			8.System retrieves and displays any recommendations made by HR, along with the context .
	Actor(Action)	System(response)																	
	1. Applicant logs into the system.																		
		2. System displays the applicant's dashboard.																	
	3. Applicant selects "view portfolio" from the menu.																		
		4.System retrieves and displays a list of all applications submitted by the applicant																	
	5.Applicant selects a specific application to view details.																		
		6.System displays the applicant's profile page, showing personal details such as name, description, qualifications, and CGPA.																	
	7.Applicant selects "view Recommendations" .																		
	8.System retrieves and displays any recommendations made by HR, along with the context .																		
Exceptions	<p>- <b>5a.</b>If there is an issue retrieving HR recommendations from the database.</p> <p>System displays an error message and offers the applicant an option to retry viewing the recommendations.</p>																		

**USE CASE 6**

Use case	Details											
Use Case Name	View Application Status											
Usecase id	UC-006											
Scope	Employee Recruitment and Management System											
Level	User Goal											
Actor	applicant											
Stakeholders and Interests	<ul style="list-style-type: none"><li>- <b>Applicant:</b> Wants to stay informed about the status of their job application to understand where they stand in the hiring process.</li><li>- <b>Recruiters:</b> Benefit from having applicants who are engaged and informed, which can improve communication.</li><li>- <b>The System:</b> Must provide accurate and real-time updates regarding application statuses.</li></ul>											
Preconditions	The applicant has submitted an application for a job and is logged into the system.											
Postconditions	The applicant successfully views the current status of their application.											
Main flow	<table><tr><th>Actor(Action)</th><th>System(response)</th></tr><tr><td>1. Applicant logs into the system.</td><td></td></tr><tr><td></td><td>2. System displays the applicant's dashboard.</td></tr><tr><td>3. Applicant selects "view job status" from the menu.</td><td></td></tr><tr><td></td><td>4.System retrieves and displays a list of all applications submitted by the applicant</td></tr></table>		Actor(Action)	System(response)	1. Applicant logs into the system.			2. System displays the applicant's dashboard.	3. Applicant selects "view job status" from the menu.			4.System retrieves and displays a list of all applications submitted by the applicant
Actor(Action)	System(response)											
1. Applicant logs into the system.												
	2. System displays the applicant's dashboard.											
3. Applicant selects "view job status" from the menu.												
	4.System retrieves and displays a list of all applications submitted by the applicant											

	5.Applicant selects a specific application to view details.	
		6.System displays the current status of the selected application (e.g., under review, interviewed, rejected, etc.).
Exceptions	<ul style="list-style-type: none"> <li>- <b>4a.</b> If no applications have been submitted, the system displays a message indicating there are no applications to show.</li> <li>- <b>6a.</b> If the application status fails to load, the system notifies the applicant and suggests checking back later.</li> <li>- <b>7a.</b> If the applicant has questions regarding the status, the system allows them to send an inquiry to the recruiting team.</li> </ul>	

**RECRUITER :-**

Member name:-wajiha adnan (22i1263)

**USE CASE 7:-**

Use case	Details
Use Case Name	Job Post Operations
Usecase id	UC-007
Scope	Employee Recruitment and Management System
Level	User Goal
Actor	recruiter

Stakeholders and Interests	<ul style="list-style-type: none"><li>- <b>Recruiter</b>: Wants to add, update, or remove job posts easily to ensure accurate and timely job listings.</li><li>- <b>Applicant</b>: Interested in viewing and applying for available job postings that match their qualifications.</li><li>- <b>HR Manager</b>: Wants to monitor job postings to ensure they align with hiring policies and staffing needs.</li><li>- <b>System</b>: Needs to ensure that job posts are correctly added, updated, or removed while maintaining data integrity and visibility to applicants.</li></ul>																			
Preconditions	<ul style="list-style-type: none"><li>- The recruiter has an active profile in the system.</li><li>- For updates and removals, the job posting must already exist.</li></ul>																			
Postconditions	<ul style="list-style-type: none"><li>- Job post is successfully added, updated, or removed, and the system reflects the changes appropriately.</li></ul>																			
Main flow	<table><tr><th>Actor(Action)</th><th>System(response)</th></tr><tr><td>1. Recruiter selects "Post a Job" to add a new job</td><td></td></tr><tr><td></td><td>2 System displays job posting form.</td></tr><tr><td>3.Recruiter enters job details (title, description, requirements).</td><td></td></tr><tr><td></td><td>4.System validates the job details.</td></tr><tr><td>5. Recruiter submits the new job post.</td><td></td></tr><tr><td></td><td>6. System saves the job post and makes it visible to applicants</td></tr><tr><td>7. Recruiter views an existing job post to update.</td><td></td></tr><tr><td></td><td>8.System retrieves and displays the job details for editing</td></tr></table>		Actor(Action)	System(response)	1. Recruiter selects "Post a Job" to add a new job			2 System displays job posting form.	3.Recruiter enters job details (title, description, requirements).			4.System validates the job details.	5. Recruiter submits the new job post.			6. System saves the job post and makes it visible to applicants	7. Recruiter views an existing job post to update.			8.System retrieves and displays the job details for editing
Actor(Action)	System(response)																			
1. Recruiter selects "Post a Job" to add a new job																				
	2 System displays job posting form.																			
3.Recruiter enters job details (title, description, requirements).																				
	4.System validates the job details.																			
5. Recruiter submits the new job post.																				
	6. System saves the job post and makes it visible to applicants																			
7. Recruiter views an existing job post to update.																				
	8.System retrieves and displays the job details for editing																			



	9. Recruiter edits the job details and submits changes	
		10. System validates and updates the job post accordingly.
Exceptions	- <b>4a.</b> If validation fails (e.g., missing fields), the system prompts the recruiter to correct the information.	

**USECASE 8: -**

Use case	Details
Use Case Name	Manage Applications
Usecase id	UC-008
Scope	Employee Recruitment and Management System
Level	User Goal
Actor	Recruiter
Stakeholders and Interests	<ul style="list-style-type: none"> <li>- <b>Recruiter:</b> Needs to efficiently review, filter, and manage applicant profiles to find the best candidates for open positions.</li> <li>- <b>Applicant:</b> Wants their application to be reviewed thoroughly and fairly, and their profile to be viewed by recruiters.</li> <li>- <b>HR Manager:</b> Oversees the recruitment process and ensures the recruiter follows company policies in filtering and selecting candidates.</li> <li>- <b>System:</b> Must apply filtering criteria correctly, present relevant profiles quickly, and provide tools for managing candidates (e.g., shortlisting or rejecting).</li> </ul>
Preconditions	<ul style="list-style-type: none"> <li>- The recruiter is logged into the system.</li> <li>- Applicants have submitted their applications to job postings.</li> </ul>
Postconditions	- The recruiter successfully views and manages applicant profiles (e.g., shortlisting, rejecting) for further stages.

Main flow		
Exceptions		

**USECASE 9:-**

Use case	Details										
Use Case Name	Organize and Conduct Tests										
Use Case id	UC-009										
Scope	Employee Recruitment and Management System										
Level	User Goal										
Actor	Recruiter										
Stakeholders and Interests	<ul style="list-style-type: none"> <li>- <b>Recruiter</b>: Needs to organize and schedule tests efficiently to evaluate shortlisted applicants' skills.</li> <li>- <b>Applicants</b>: Expect to be informed about scheduled tests in a timely manner and want an easy and accessible way to take the tests.</li> <li>- <b>System</b>: Ensures smooth scheduling of tests, timely notifications to applicants, and tracking of test completion and results.</li> </ul>										
Preconditions	<ul style="list-style-type: none"> <li>- Job IDs have been created in the system</li> <li>- Recruiter is logged into the system.</li> </ul>										
Postconditions	<ul style="list-style-type: none"> <li>- Test details are successfully saved, tests are conducted, and applicants meeting the criteria are shortlisted.</li> </ul>										
Main flow	<table> <tr> <th>Actor(Action)</th><th>System(response)</th></tr> <tr> <td>1.Recruiter selects the job ID(s) for which the test will be created.</td><td></td></tr> <tr> <td>2.Recruiter enters test details: number of questions, minimum passing score.</td><td></td></tr> <tr> <td></td><td>3.saves the test configuration details and links it to the job ID(s).</td></tr> <tr> <td>4.Recruiter selects an applicant ID for a job and schedules the test.</td><td></td></tr> </table>	Actor(Action)	System(response)	1.Recruiter selects the job ID(s) for which the test will be created.		2.Recruiter enters test details: number of questions, minimum passing score.			3.saves the test configuration details and links it to the job ID(s).	4.Recruiter selects an applicant ID for a job and schedules the test.	
Actor(Action)	System(response)										
1.Recruiter selects the job ID(s) for which the test will be created.											
2.Recruiter enters test details: number of questions, minimum passing score.											
	3.saves the test configuration details and links it to the job ID(s).										
4.Recruiter selects an applicant ID for a job and schedules the test.											

	5. Recruiter conducts the test and enters the applicant's marks.	
		6. Saves the marks and checks if the score meets the minimum passing criteria.
	7. If the marks meet or exceed the minimum score, the applicant is shortlisted.	
		8. Updates the applicant's status to "shortlisted" and notifies the recruiter and applicant.
Exceptions	<b>- 2a.</b> Recruiter does not enter valid test configuration details. Prompts the recruiter to correct the errors (e.g., missing fields or invalid input).	

**USE CASE 10:-**

Use case	Details
Use Case Name	Shortlisting applicants
Usecase id	UC-010
Scope	Employee Recruitment and Management System
Level	User Goal
Actor	Recruiter, HR Manager

Stakeholders and Interests	<ul style="list-style-type: none"> <li>- <b>Recruiter:</b> Needs to shortlist qualified applicants and schedule interviews in an organized manner.</li> <li>- <b>Applicants:</b> Want fair consideration for interviews and expect timely interview scheduling.</li> <li>- <b>HR Manager:</b> Needs to ensure that interview scheduling and preparation align with internal hiring policies.</li> <li>- <b>System:</b> Handles shortlisting, scheduling interviews, and notifying applicants, ensuring the process is smooth and efficient.</li> </ul>										
Preconditions	<ul style="list-style-type: none"> <li>- The recruiter is logged into the system.</li> <li>- Applicants have completed tests or have been initially shortlisted based on applications.</li> </ul>										
Postconditions	<ul style="list-style-type: none"> <li>- Applicants are successfully shortlisted at two levels (Recruiter, HR Manager), and interviews are scheduled and communicated.</li> </ul>										
Main flow	<table border="1"> <thead> <tr> <th>Actor(Action)</th><th>System(response)</th></tr> </thead> <tbody> <tr> <td>1. Recruiter reviews applicant profiles and shortlists candidates for HR review.</td><td></td></tr> <tr> <td></td><td>2. Updates the applicants' status to "waiting for HR response" and notifies HR for further assessment</td></tr> <tr> <td>3. HR Manager reviews the recruiter's shortlisted applicants and finalizes candidates for interviews.</td><td></td></tr> <tr> <td></td><td>4. Updates the status of finalized applicants to "congrats you got the job".</td></tr> </tbody> </table>	Actor(Action)	System(response)	1. Recruiter reviews applicant profiles and shortlists candidates for HR review.			2. Updates the applicants' status to "waiting for HR response" and notifies HR for further assessment	3. HR Manager reviews the recruiter's shortlisted applicants and finalizes candidates for interviews.			4. Updates the status of finalized applicants to "congrats you got the job".
Actor(Action)	System(response)										
1. Recruiter reviews applicant profiles and shortlists candidates for HR review.											
	2. Updates the applicants' status to "waiting for HR response" and notifies HR for further assessment										
3. HR Manager reviews the recruiter's shortlisted applicants and finalizes candidates for interviews.											
	4. Updates the status of finalized applicants to "congrats you got the job".										
Exceptions	<ul style="list-style-type: none"> <li>- <b>1a.</b> Recruiter does not shortlist any applicants.</li> <li>- System notifies the recruiter of the error</li> <li>- <b>2a.</b> HR Manager does not shortlist the recruiter-shortlisted applicants.</li> <li>- System notifies the HR of the error</li> </ul>										

**USERCASE 11:-**

Use case	Details												
Use Case Name	Book appointment with applicant												
Usecase id	UC-011												
Scope	Employee Recruitment and Management System												
Level	User Goal												
Actor	Recruiter,applicant												
Stakeholders and Interests	<ul style="list-style-type: none"> <li>- <b>Recruiter</b>: Wants to schedule a discussion with a specific applicant efficiently.</li> <li>- <b>Applicant</b>: Wants to be notified of appointments in a timely manner and prepare accordingly.</li> <li>- <b>System</b>: Facilitates appointment scheduling, notifications, and confirmations between the recruiter and applicant.</li> </ul>												
Preconditions	<ul style="list-style-type: none"> <li>- The recruiter is logged into the system.</li> <li>- The applicant exists in the system with a valid ID.</li> </ul>												
Postconditions	- An appointment is successfully scheduled, and both the recruiter and applicant are notified.												
Main flow	<table> <tr> <th>Actor(Action)</th><th>System(response)</th></tr> <tr> <td>1. Recruiter navigates to the "Appointments" section.</td><td></td></tr> <tr> <td></td><td>2.System displays options for scheduling an appointment.</td></tr> <tr> <td>3. Recruiter specifies the appointment details: <b>date, time, purpose of discussion.</b></td><td></td></tr> <tr> <td></td><td>4.Saves the appointment details for applicant to view.</td></tr> <tr> <td>5.Applicant receives the appointment .</td><td></td></tr> </table>	Actor(Action)	System(response)	1. Recruiter navigates to the "Appointments" section.			2.System displays options for scheduling an appointment.	3. Recruiter specifies the appointment details: <b>date, time, purpose of discussion.</b>			4.Saves the appointment details for applicant to view.	5.Applicant receives the appointment .	
Actor(Action)	System(response)												
1. Recruiter navigates to the "Appointments" section.													
	2.System displays options for scheduling an appointment.												
3. Recruiter specifies the appointment details: <b>date, time, purpose of discussion.</b>													
	4.Saves the appointment details for applicant to view.												
5.Applicant receives the appointment .													

Exceptions	- <b>3a.</b> Recruiter enters an invalid Applicant ID or conflicting date/time. Prompts the recruiter to correct the errors and provides options to view the applicant's existing schedule.
------------	---

### HR MANAGER: -

Member name:-eman shakeel (22i1279)

#### USECASE 12:-

Use case	Details
Use Case Name	Make contract
Usecase id	UC-012
Scope	Employee Recruitment and Management System
Level	User Goal
Actor	HR Manager
Stakeholders and Interests	<ul style="list-style-type: none"> <li>- <b>HR Manager:</b> Needs to create and finalize contracts for selected job applicants efficiently.</li> <li>- <b>Applicant:</b> Wants to view and acknowledge the contract details for their job.</li> <li>- <b>System:</b> Ensures contracts are securely stored, properly validated, and accessible to both HR Managers and applicants.</li> </ul>
Preconditions	<ul style="list-style-type: none"> <li>- The HR Manager is registered and logged into the system.</li> <li>- The job posting and applicant IDs exist in the system.</li> </ul>
Postconditions	- Contracts are successfully created and applicants can view the final version.

Main flow	<b>Actor(Action)</b>	<b>System(response)</b>
	1. HR Manager navigates to the "Make Contract" section.	
	2.HR Manager selects a job ID and views the details of the selected posting.	
		3.Displays job details (e.g., title, description, selected applicant ID).
	4.HR Manager enters the contract details and submits the information.	
		5. System saves the contract for the applicant to view.
Exceptions	- <b>4a.</b> HR Manager enters incomplete or invalid details. Prompts the HR Manager to correct the errors before proceeding.	

**USERCASE8 13:-**

Use case	Details
Use Case Name	Provide applicant Recommendations
Usecase id	UC-013



Scope	Employee Recruitment and Management System													
Level	User Goal													
Actor	HR Manager													
Stakeholders and Interests	<ul style="list-style-type: none"><li>- <b>HR Manager</b>: Wants to provide meaningful recommendations to support employees transitioning to new positions or opportunities.</li><li>- <b>Employees</b>: Seek supportive recommendations that can enhance their prospects for future employment.</li><li>- <b>Recruiters</b>: Benefit from clear recommendations that give insights into an applicant's performance and suitability.</li><li>- <b>The System</b>: Must accurately record and display recommendations in the applicant's profile.</li></ul>													
Preconditions	The HR Manager is logged into the system and the employee has submitted a request for a recommendation.													
Postconditions	The recommendation is successfully added to the employee's profile and is visible to recruiters reviewing the profile.													
Main flow	<table><tr><th>Actor(Action)</th><th>System(response)</th></tr><tr><td>1. HR Manager selects "provide recommendations" from the dashboard.</td><td></td></tr><tr><td></td><td>2.System displays a list of applicants.</td></tr><tr><td>3. HR Manager selects applicantid to give recommendations.</td><td></td></tr><tr><td>4.HR Manager selects the recommendation type from the dropdown..</td><td></td></tr><tr><td></td><td>5.System saves it and update's the applicant's portfolio with recommendation</td></tr></table>		Actor(Action)	System(response)	1. HR Manager selects "provide recommendations" from the dashboard.			2.System displays a list of applicants.	3. HR Manager selects applicantid to give recommendations.		4.HR Manager selects the recommendation type from the dropdown..			5.System saves it and update's the applicant's portfolio with recommendation
Actor(Action)	System(response)													
1. HR Manager selects "provide recommendations" from the dashboard.														
	2.System displays a list of applicants.													
3. HR Manager selects applicantid to give recommendations.														
4.HR Manager selects the recommendation type from the dropdown..														
	5.System saves it and update's the applicant's portfolio with recommendation													

Exceptions	- <b>4a.</b> If the recommendation type or id is not selected then error message prompts
------------	--

**ADMIN: -**

Member name:-eman shakeel (22i1279)

**USERCASE 14:-**

Use case	Details
Use Case Name	Generate reports
Usecase id	UC-014
Scope	Employee Recruitment and Management System
Level	subfunction
Actor	HR Manager
Stakeholders and Interests	<ul style="list-style-type: none"> <li>- <b>HR Manager:</b> Wants to generate reports on applicants, recruiters, or HR activities for decision-making and performance analysis.</li> <li>- <b>Recruiter:</b> Needs reports to analyze the performance and effectiveness of job postings and recruitment processes.</li> <li>- <b>Applicant:</b> Interested in knowing the status of their application.</li> <li>- <b>System:</b> Must generate accurate reports on the selected actor's activities and performance.</li> </ul>

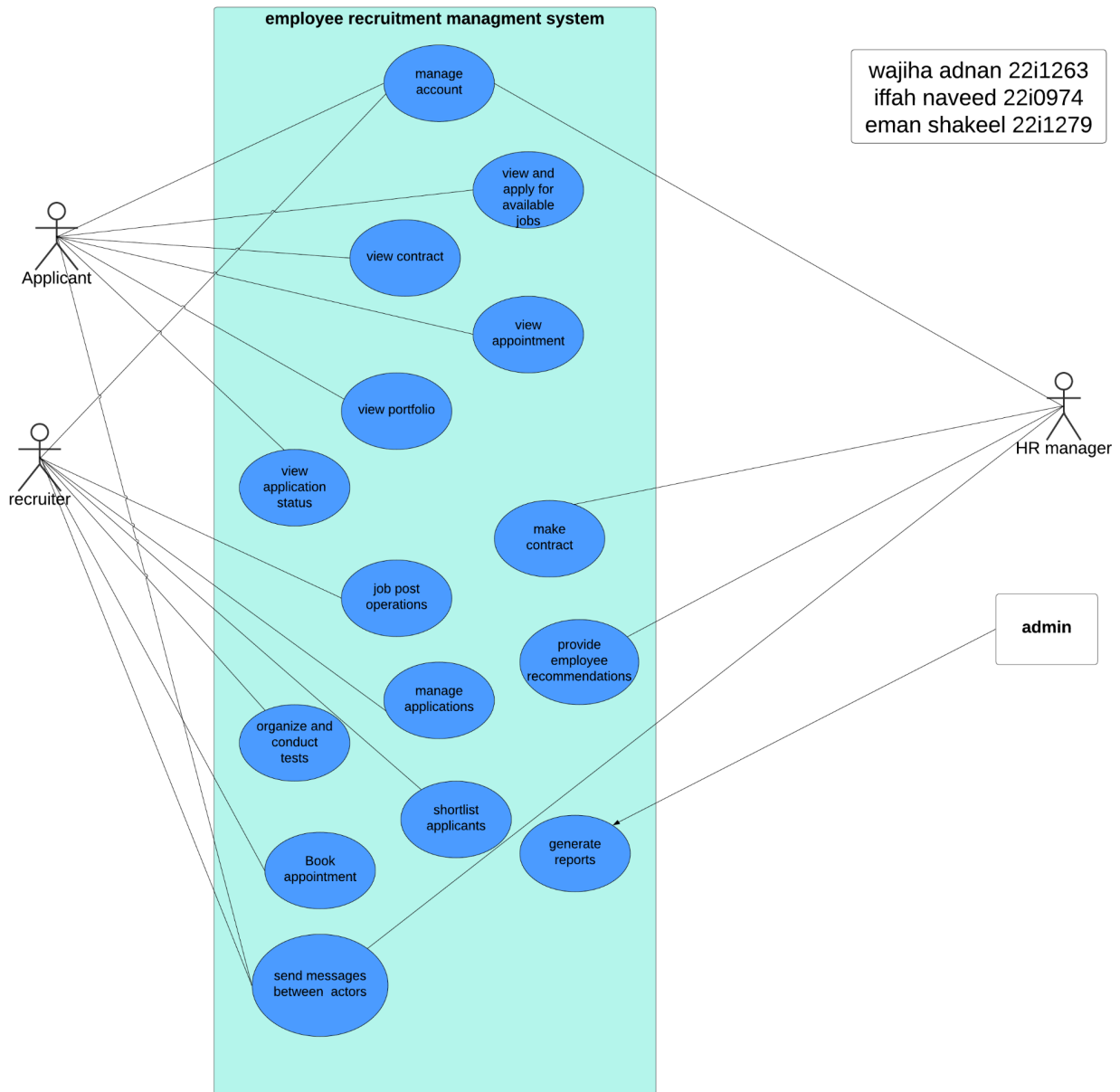
Preconditions	<ul style="list-style-type: none"> <li>- HR Manager is logged into the system.</li> <li>- The applicant, recruiter, or HR activities have data in the system that can be reported on.</li> </ul>										
Postconditions	The system successfully generates and displays the requested report.										
Main flow	<table> <tr> <th>Actor(Action)</th><th>System(response)</th></tr> <tr> <td>1. Admin selects the actor (Applicant, Recruiter, or HR) for which the report is to be generated..</td><td></td></tr> <tr> <td></td><td>2. The system displays ids of the actors.</td></tr> <tr> <td>3.Admin then selects the id for the actor</td><td></td></tr> <tr> <td></td><td>4. system then displays the report .</td></tr> </table>	Actor(Action)	System(response)	1. Admin selects the actor (Applicant, Recruiter, or HR) for which the report is to be generated..			2. The system displays ids of the actors.	3.Admin then selects the id for the actor			4. system then displays the report .
Actor(Action)	System(response)										
1. Admin selects the actor (Applicant, Recruiter, or HR) for which the report is to be generated..											
	2. The system displays ids of the actors.										
3.Admin then selects the id for the actor											
	4. system then displays the report .										
Exceptions	-3a. if no id selected then error message displays										

**USERCASE 15:-**

Use case	Details
Use Case Name	Send Messages Between Applicants and Recruiters or Applicant and HR
Usecase id	UC-015
Scope	Employee Recruitment and Management System
Level	User goal
Actor	Recruiter, applicant,HR
Stakeholders and Interests	<ul style="list-style-type: none"><li>- <b>Applicant:</b> Wants to communicate with the recruiter or HR to clarify questions about the job or application process.</li><li>- <b>Recruiter:</b> Aims to provide clear communication to applicants regarding their application status, job details, or interview feedback.</li><li>- <b>HR:</b> Needs to facilitate communication between the recruiter and applicant and address any HR-related concerns.</li><li>- <b>System:</b> Must enable secure, organized messaging between applicants, recruiters, and HR, ensuring both parties can send and receive messages efficiently.</li></ul>
Preconditions	<ul style="list-style-type: none"><li>- The recruiter or HR has selected an applicant and enabled messaging for that applicant.</li><li>- The applicant is registered and logged into the system.</li></ul>
Postconditions	Messages are successfully sent and received between the recruiter, applicant, and HR.

Main flow	<table> <tr> <th>Actor(Action)</th> <th>System(response)</th> </tr> <tr> <td>1. Actor selects an applicant to communicate with..</td> <td></td> </tr> <tr> <td></td> <td>2.System displays an option to send a message.</td> </tr> <tr> <td>3.Actor composes a message and sends it.</td> <td></td> </tr> <tr> <td></td> <td>4. System delivers the message to the applicant.</td> </tr> <tr> <td>5. Applicant receives the message</td> <td></td> </tr> <tr> <td></td> <td>6. System allows the applicant to read the message.</td> </tr> <tr> <td>7.Applicant composes a reply and sends it.</td> <td></td> </tr> <tr> <td></td> <td>8.System delivers the reply to the recruiter.</td> </tr> <tr> <td>9. Actor receives applicant's reply.</td> <td></td> </tr> <tr> <td></td> <td>10.System allows the applicant to read the response.</td> </tr> </table>	Actor(Action)	System(response)	1. Actor selects an applicant to communicate with..			2.System displays an option to send a message.	3.Actor composes a message and sends it.			4. System delivers the message to the applicant.	5. Applicant receives the message			6. System allows the applicant to read the message.	7.Applicant composes a reply and sends it.			8.System delivers the reply to the recruiter.	9. Actor receives applicant's reply.			10.System allows the applicant to read the response.
Actor(Action)	System(response)																						
1. Actor selects an applicant to communicate with..																							
	2.System displays an option to send a message.																						
3.Actor composes a message and sends it.																							
	4. System delivers the message to the applicant.																						
5. Applicant receives the message																							
	6. System allows the applicant to read the message.																						
7.Applicant composes a reply and sends it.																							
	8.System delivers the reply to the recruiter.																						
9. Actor receives applicant's reply.																							
	10.System allows the applicant to read the response.																						
Exceptions	<p><b>-4a.</b> If the message fails to send due to a technical issue, the system prompts an error message.</p>																						

## 5.4 Use Case Diagram



## 6. Other Nonfunctional Requirements

### Performance Requirements

- The system should support up to **100 users simultaneously** with minimal delay.
- Key operations like **viewing jobs** or **submitting applications** must complete within **2 seconds**.
- **Data synchronization** must be instant to keep information updated across all users.
- Operations like **searching jobs** or **generating reports** should take no longer than **5 seconds**.

### Safety Requirements

- Regular **automatic backups** will prevent data loss.
- Error-handling will ensure **data remains secure** and unaltered during failures.
- Fail-safe mechanisms will protect the integrity of applications and contracts during power or system failures.
- The system will follow safety standards to minimize risks.

### Security Requirements

- **User authentication** will require strong passwords and optional two-factor authentication.
- Data will be **encrypted** during storage and transfer to ensure privacy.
- The system will apply **role-based access control (RBAC)** to restrict access based on user roles (e.g., admin, HR, recruiter).

### Software Quality Attributes

- **Reliability:** The system will ensure **99.9% uptime**.
- **Maintainability:** Modular design will make updates and fixes easy.
- **Scalability:** The system can grow to handle more users, features, or integrations without slowing down.
- **Usability:** The interface will remain **simple and user-friendly**.
- **Testability:** The system will undergo detailed testing to ensure reliability.
- **Interoperability:** It will connect easily with tools like payroll and employee management software.

### Business Rules

- Only **logged-in users** can access specific modules.
- **Recruiters** can manage only their own job postings.
- **HR managers** can approve contracts but not modify applicant profiles.
- **Admins** can see all data but cannot apply for jobs or post openings.

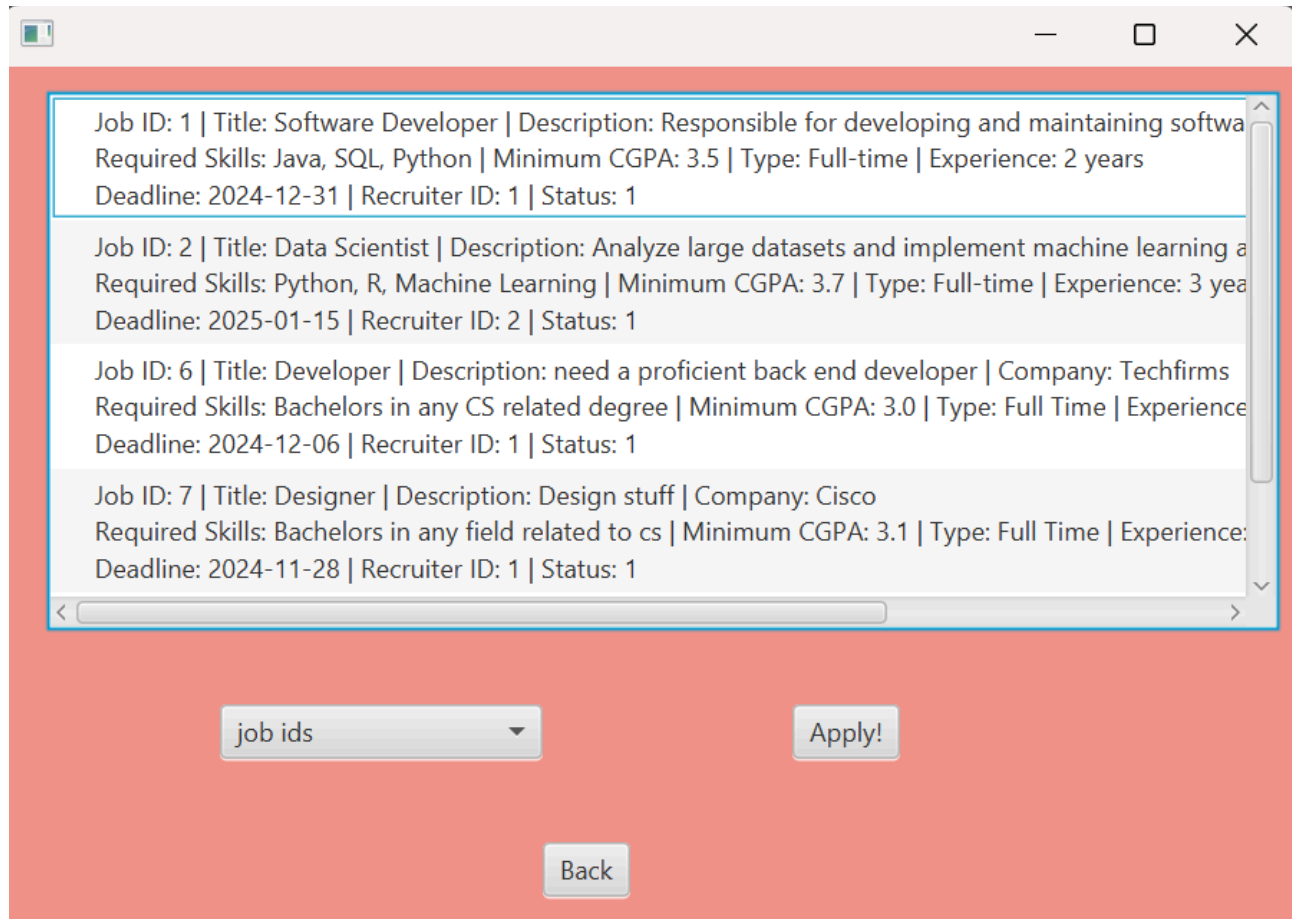
### Operating Environment

- The system will work on Windows, macOS, or Linux.
- It requires **Java 17, microsoft sql server**.

## **6.1 User Interfaces**

In this Employee Recruitment and Management System, the **system** is designed with **distinct user interfaces** for each actor (Admin, Applicant, Recruiter, and HR). Each interface is tailored to the specific needs and responsibilities of the actor, offering customized submenus, screens, and functionalities. This structured design ensures a seamless experience for users at different levels within the recruitment process. The system provides multiple screens for various functionalities and offers prompt messages to inform users about the success or failure of their actions, including error messages or confirmation of successful operations.

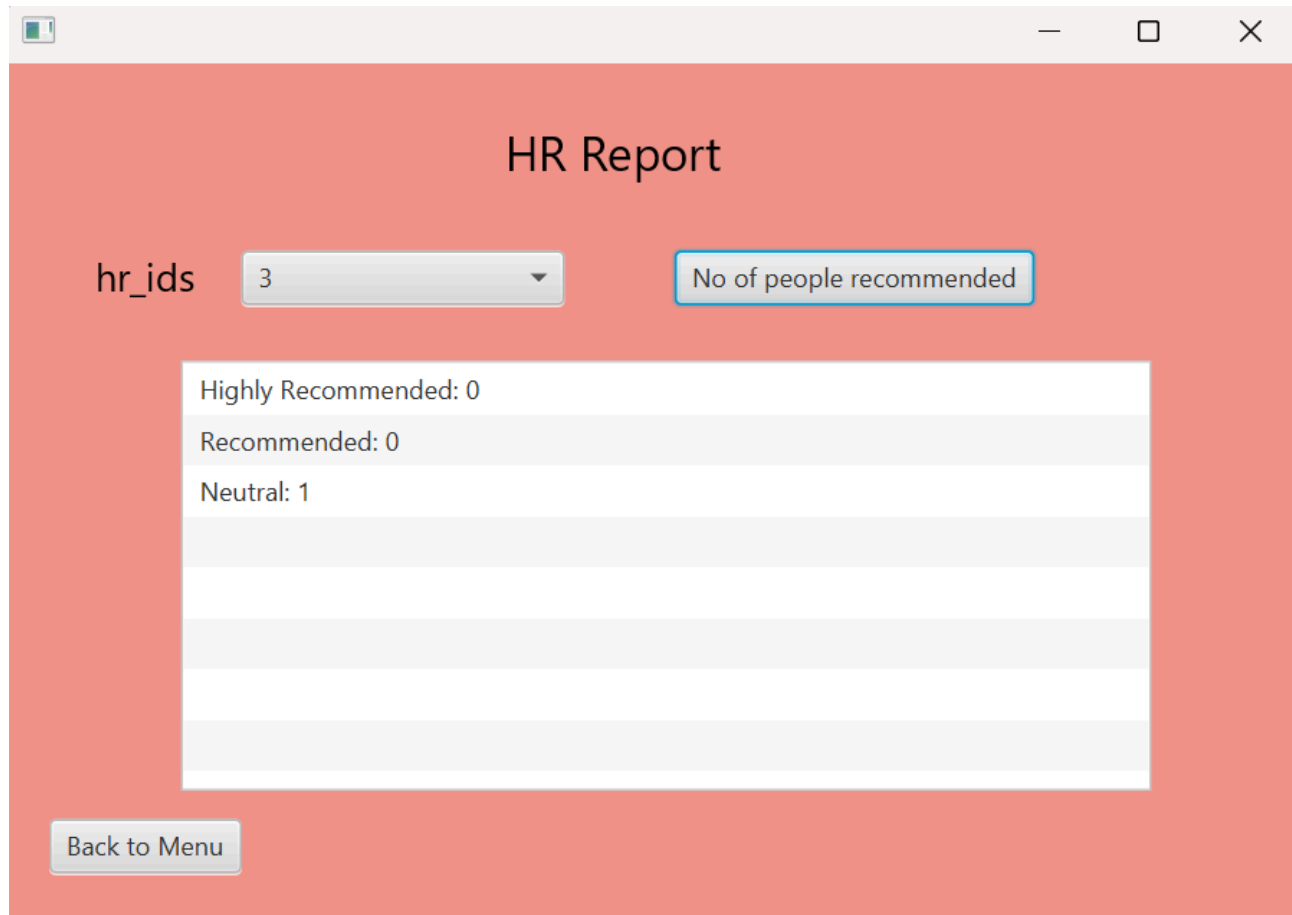




The screenshot displays a web application interface with a light red background. At the top, there is a window-like container with a title bar containing a green icon, a minus sign, a square icon, and a close 'X' icon. Inside this container is a scrollable list of four job postings, each with a light gray background and a blue border. The job postings are as follows:

- Job ID: 1 | Title: Software Developer | Description: Responsible for developing and maintaining software  
Required Skills: Java, SQL, Python | Minimum CGPA: 3.5 | Type: Full-time | Experience: 2 years  
Deadline: 2024-12-31 | Recruiter ID: 1 | Status: 1
- Job ID: 2 | Title: Data Scientist | Description: Analyze large datasets and implement machine learning algorithms  
Required Skills: Python, R, Machine Learning | Minimum CGPA: 3.7 | Type: Full-time | Experience: 3 years  
Deadline: 2025-01-15 | Recruiter ID: 2 | Status: 1
- Job ID: 6 | Title: Developer | Description: need a proficient back end developer | Company: Techfirms  
Required Skills: Bachelors in any CS related degree | Minimum CGPA: 3.0 | Type: Full Time | Experience: 2 years  
Deadline: 2024-12-06 | Recruiter ID: 1 | Status: 1
- Job ID: 7 | Title: Designer | Description: Design stuff | Company: Cisco  
Required Skills: Bachelors in any field related to cs | Minimum CGPA: 3.1 | Type: Full Time | Experience: 2 years  
Deadline: 2024-11-28 | Recruiter ID: 1 | Status: 1

Below the scrollable list, there is a horizontal scrollbar. At the bottom of the interface, there are three buttons: a dropdown menu labeled 'job ids' with a downward arrow, an 'Apply!' button, and a 'Back' button.



The image shows a window titled "HR Report" with a red background. At the top, there is a title bar with standard window controls (minimize, maximize, close). Below the title bar, the text "HR Report" is centered. On the left, there is a label "hr\_ids" followed by a dropdown menu showing the value "3". To the right of the dropdown is a button labeled "No of people recommended". Below these elements is a table with alternating light gray and white rows. The table contains the following data:

Highly Recommended: 0
Recommended: 0
Neutral: 1

At the bottom left of the window, there is a button labeled "Back to Menu".

back

Contract

Job id

salary

probabation period

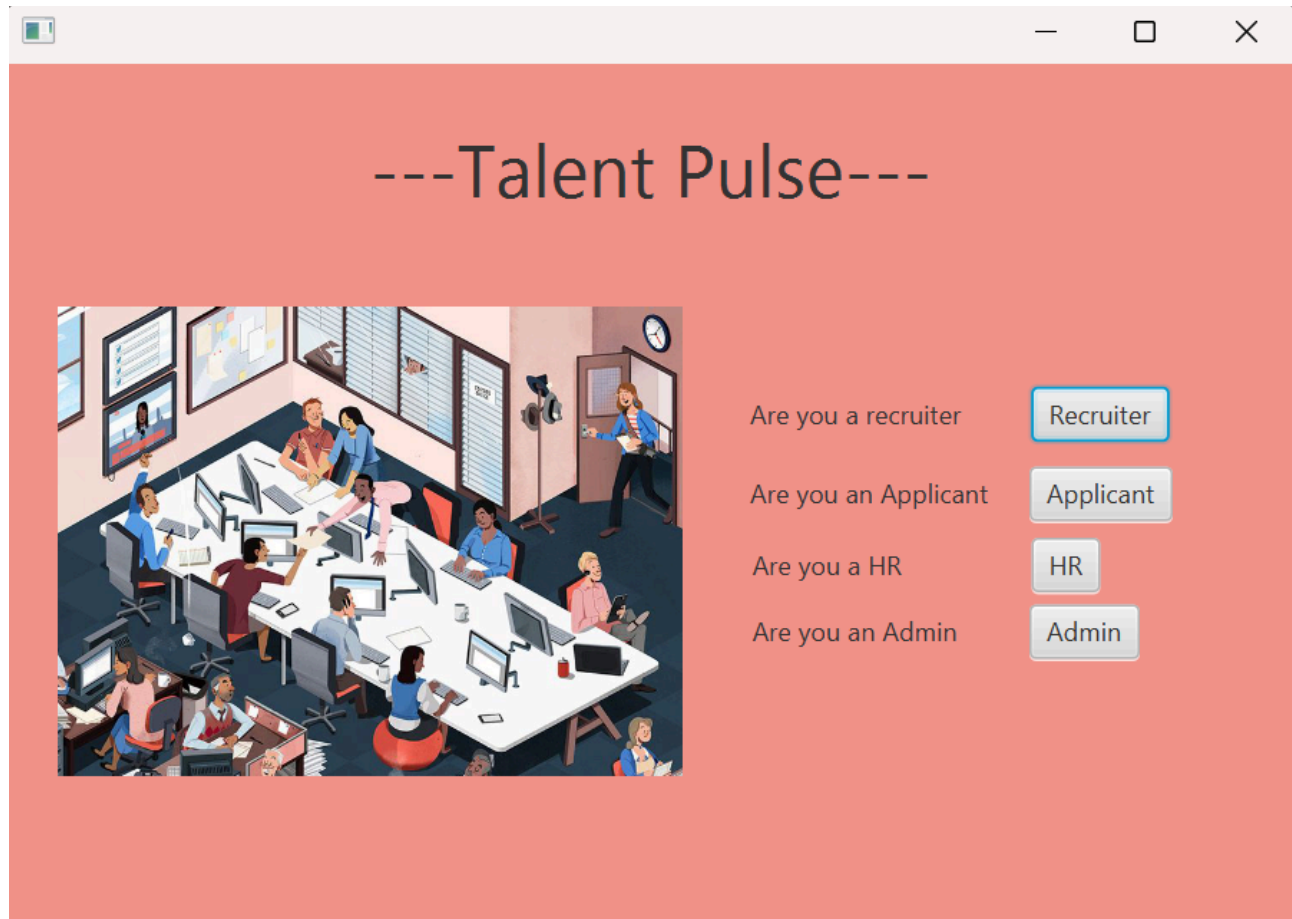
contract start date

contract end date

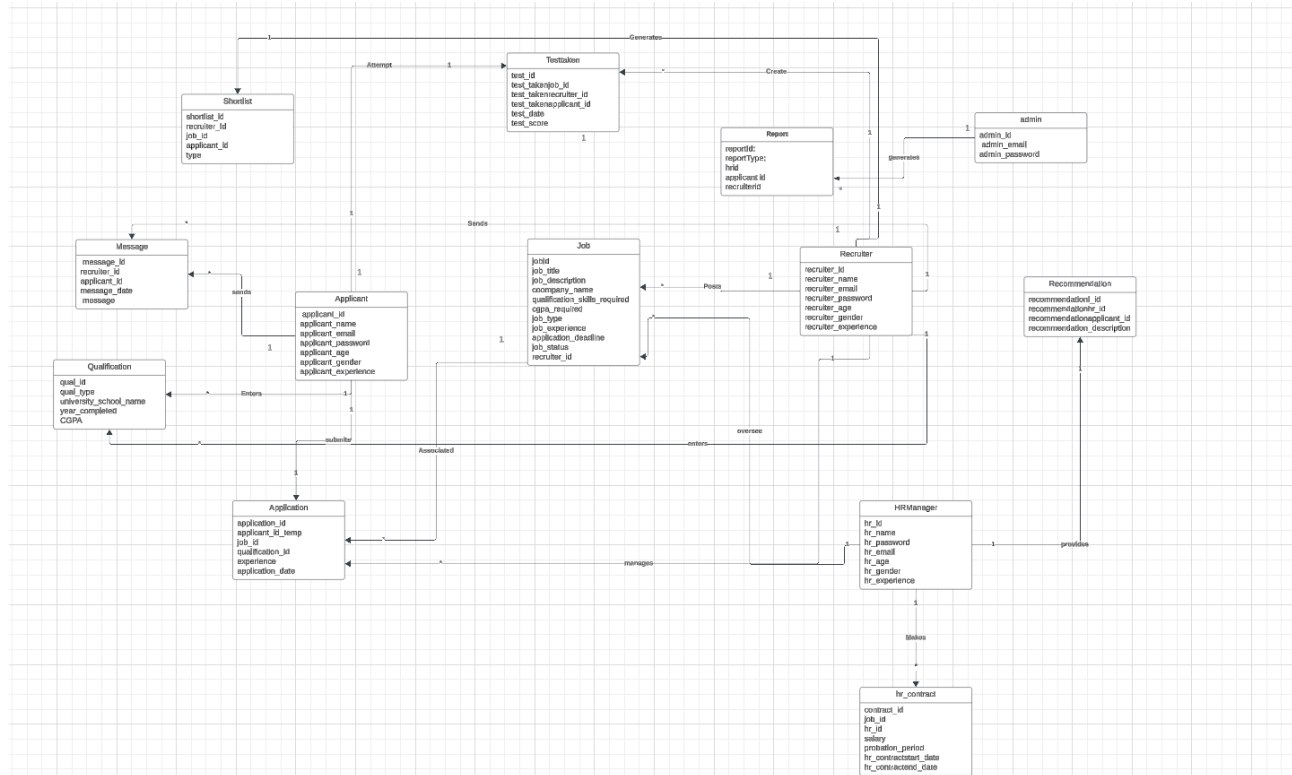
benefits

Job Details

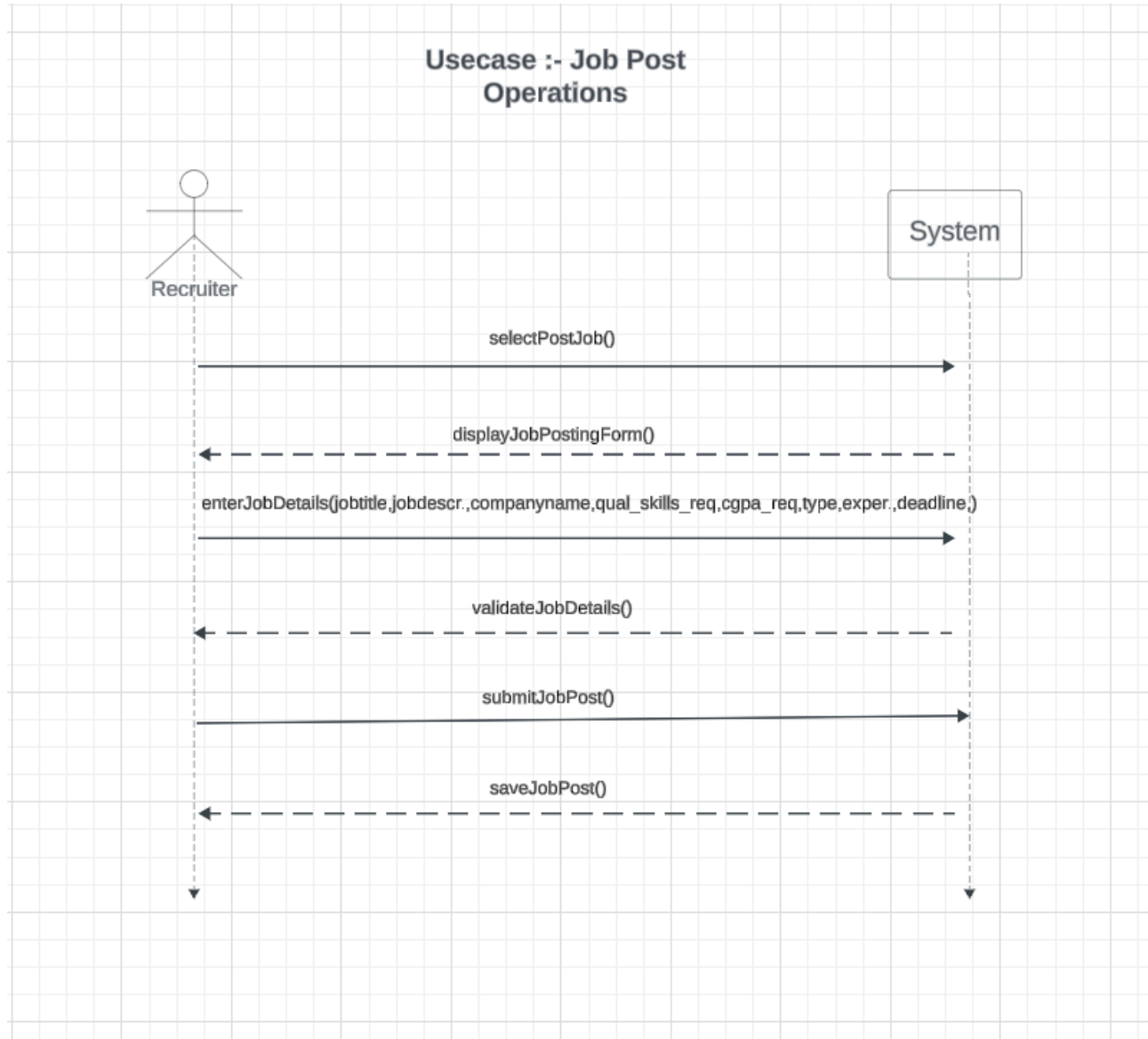
Button

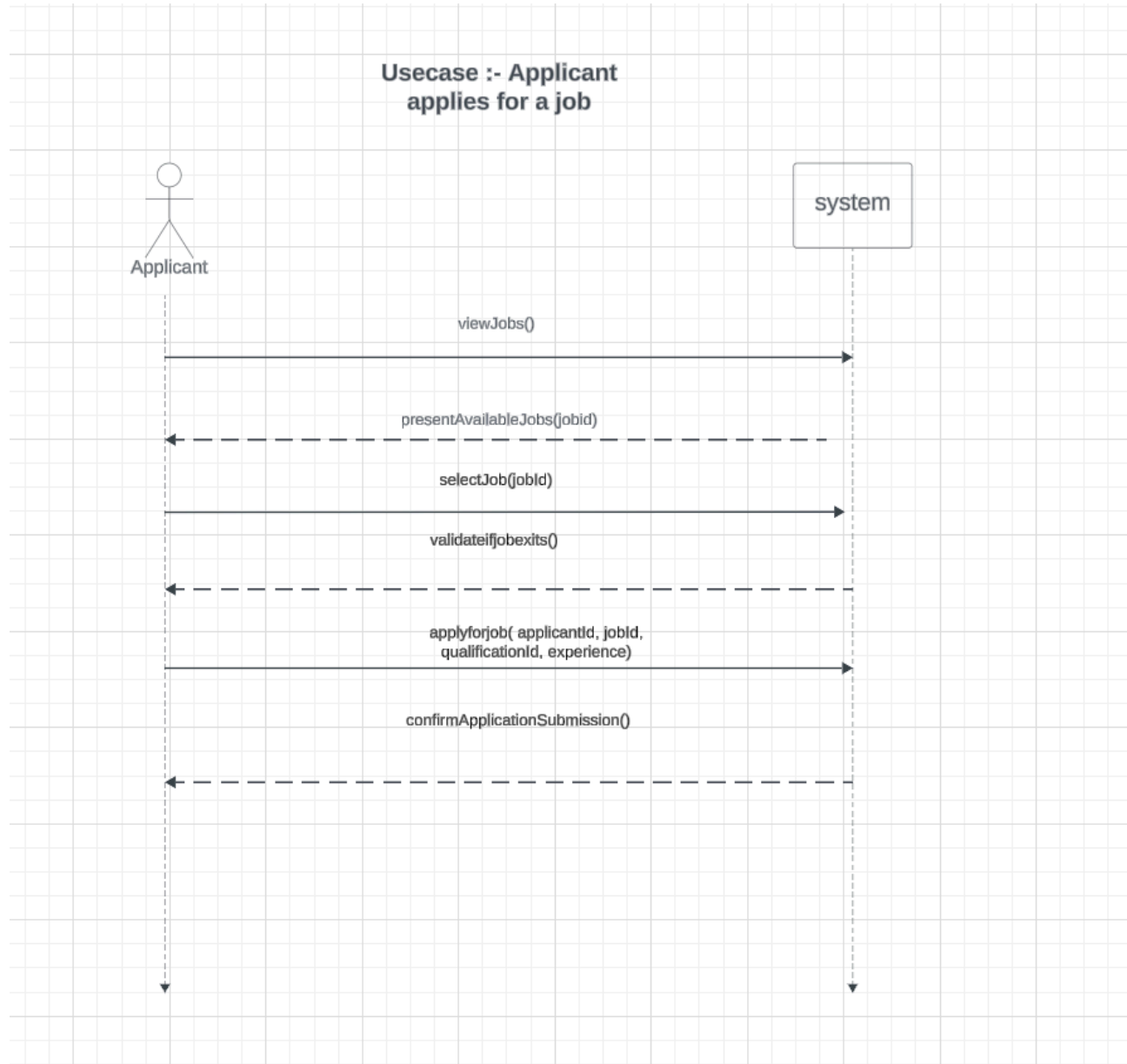


## 7. Domain Model



## **8. System Sequence Diagram**

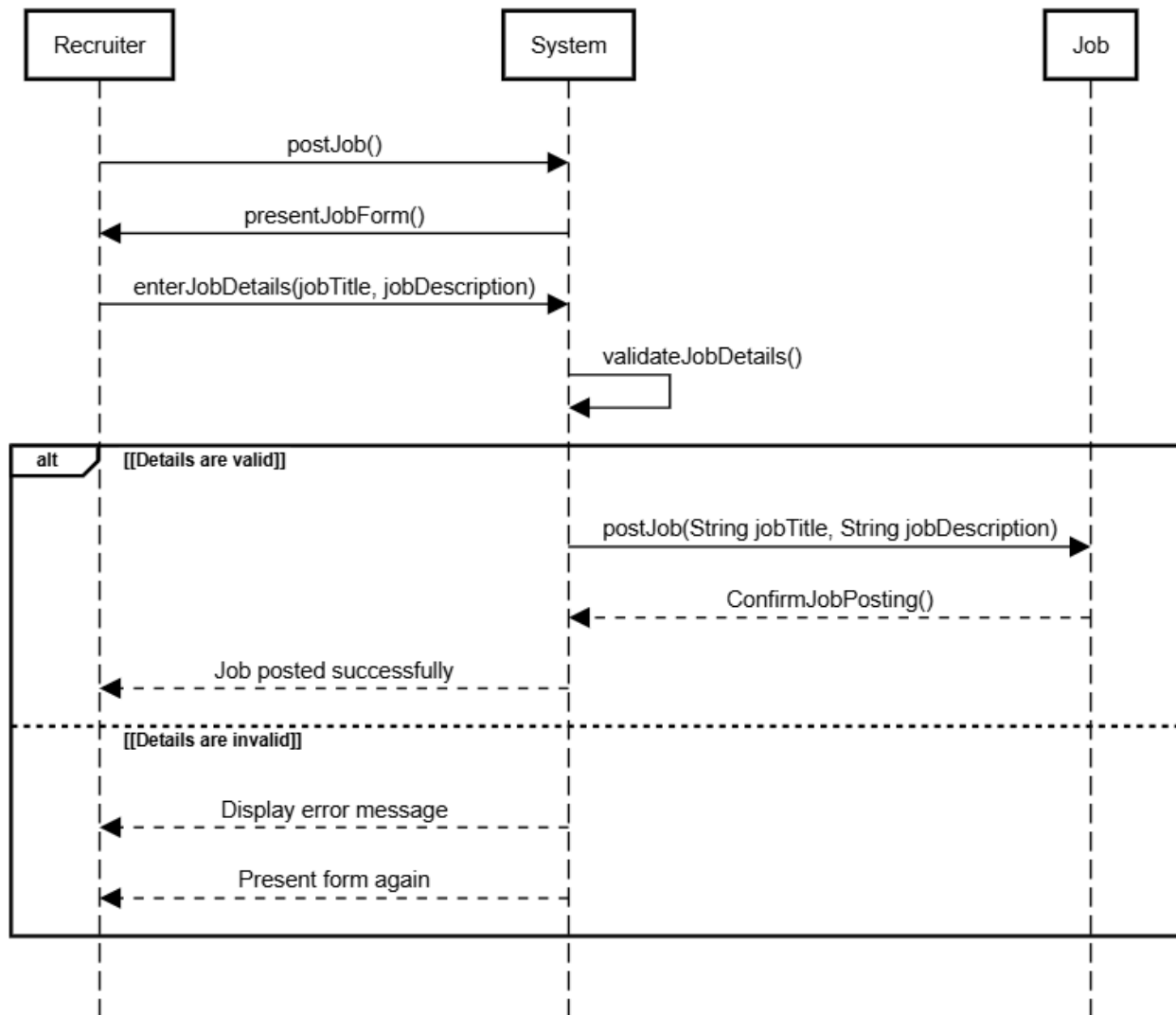




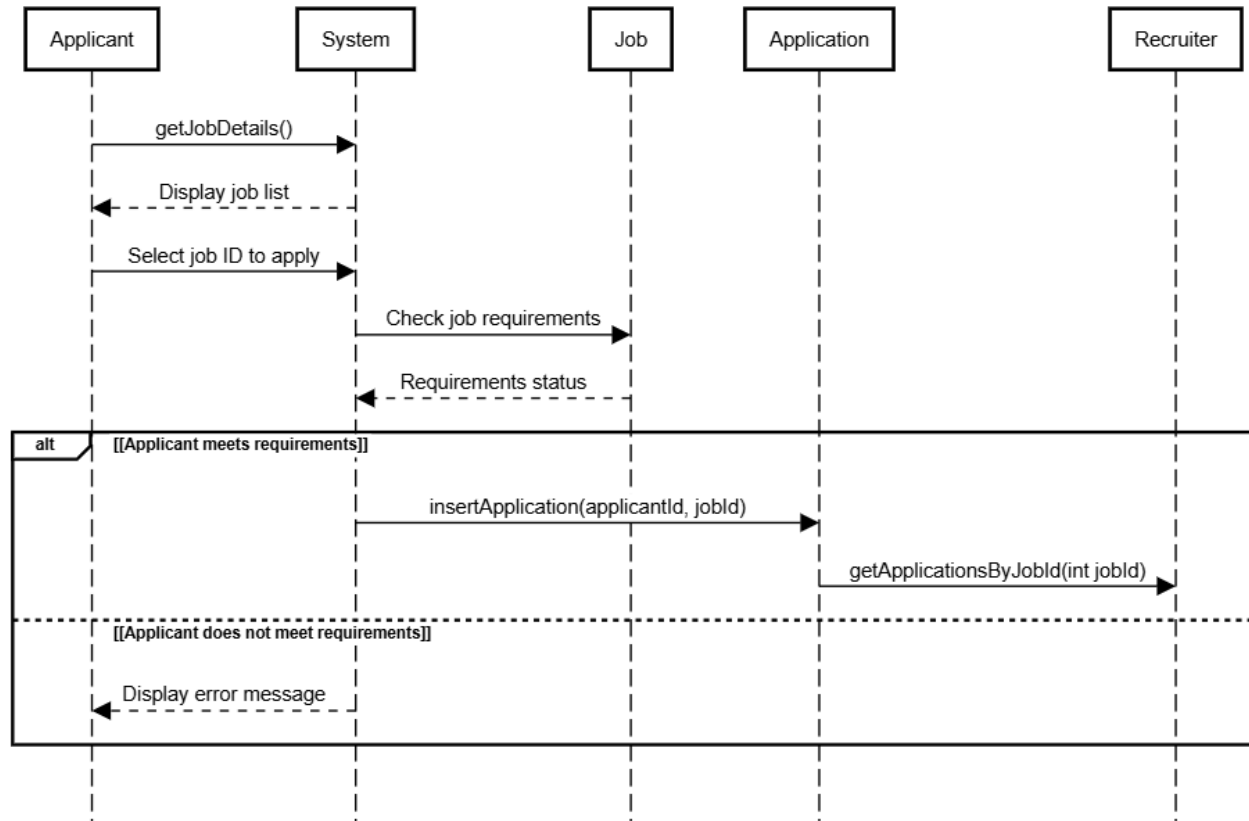
## 9. Sequence Diagram



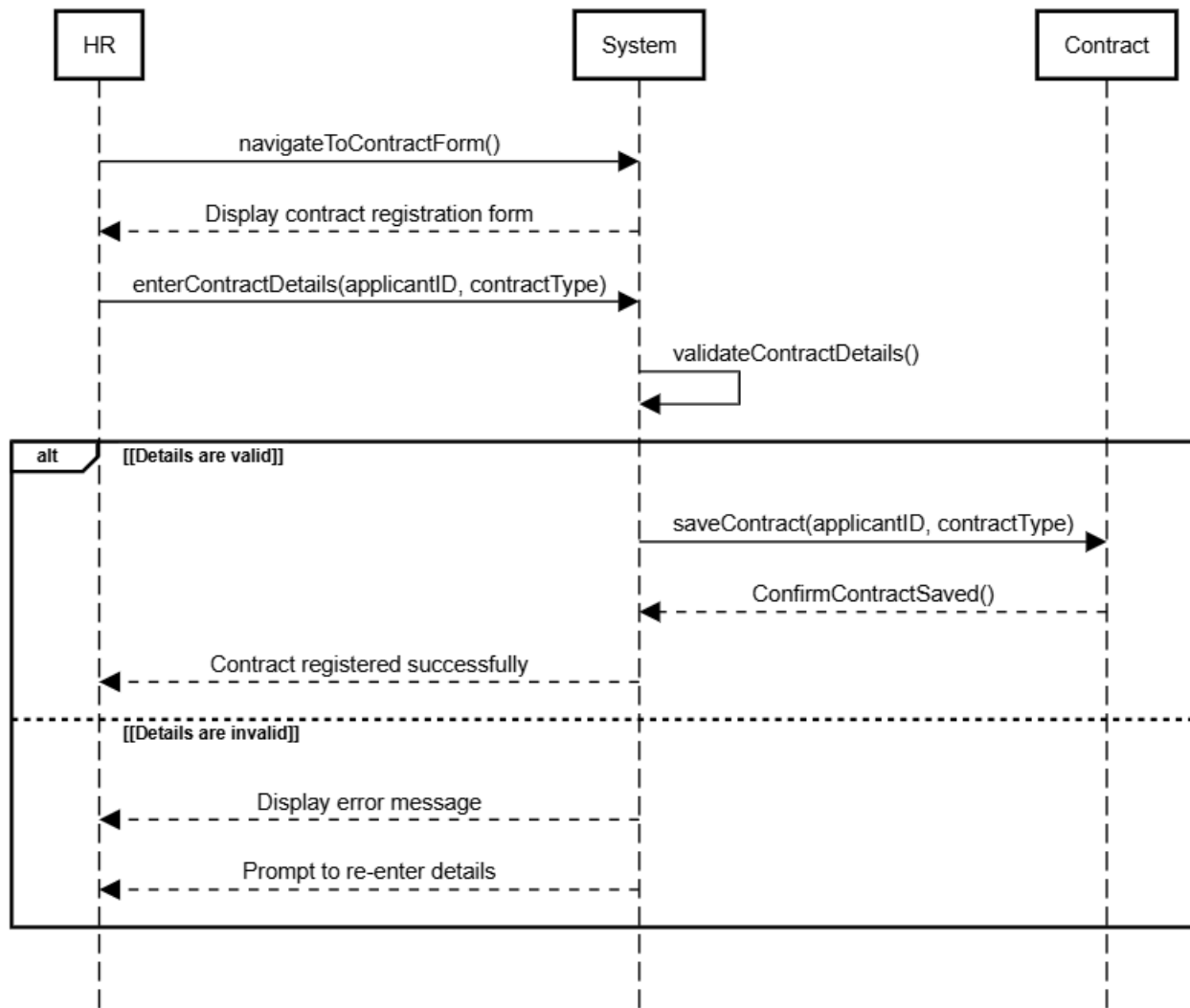
## Job Post Operation



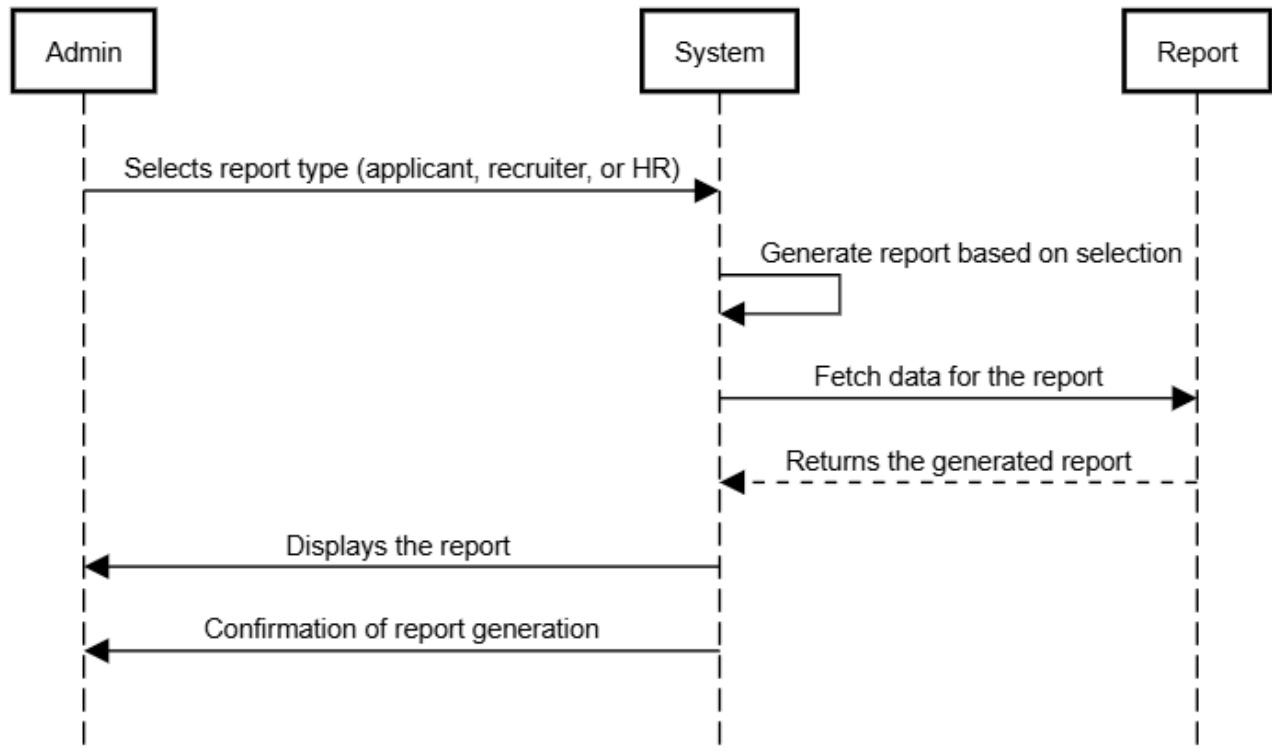
## Applicant Applies for a Job



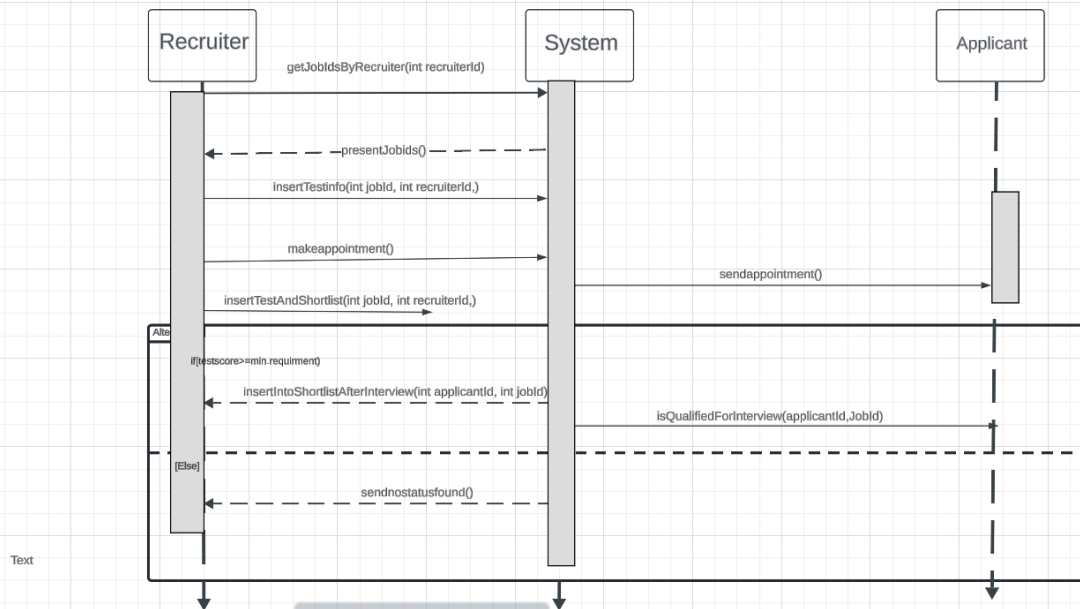
## HR Registers Contract Information

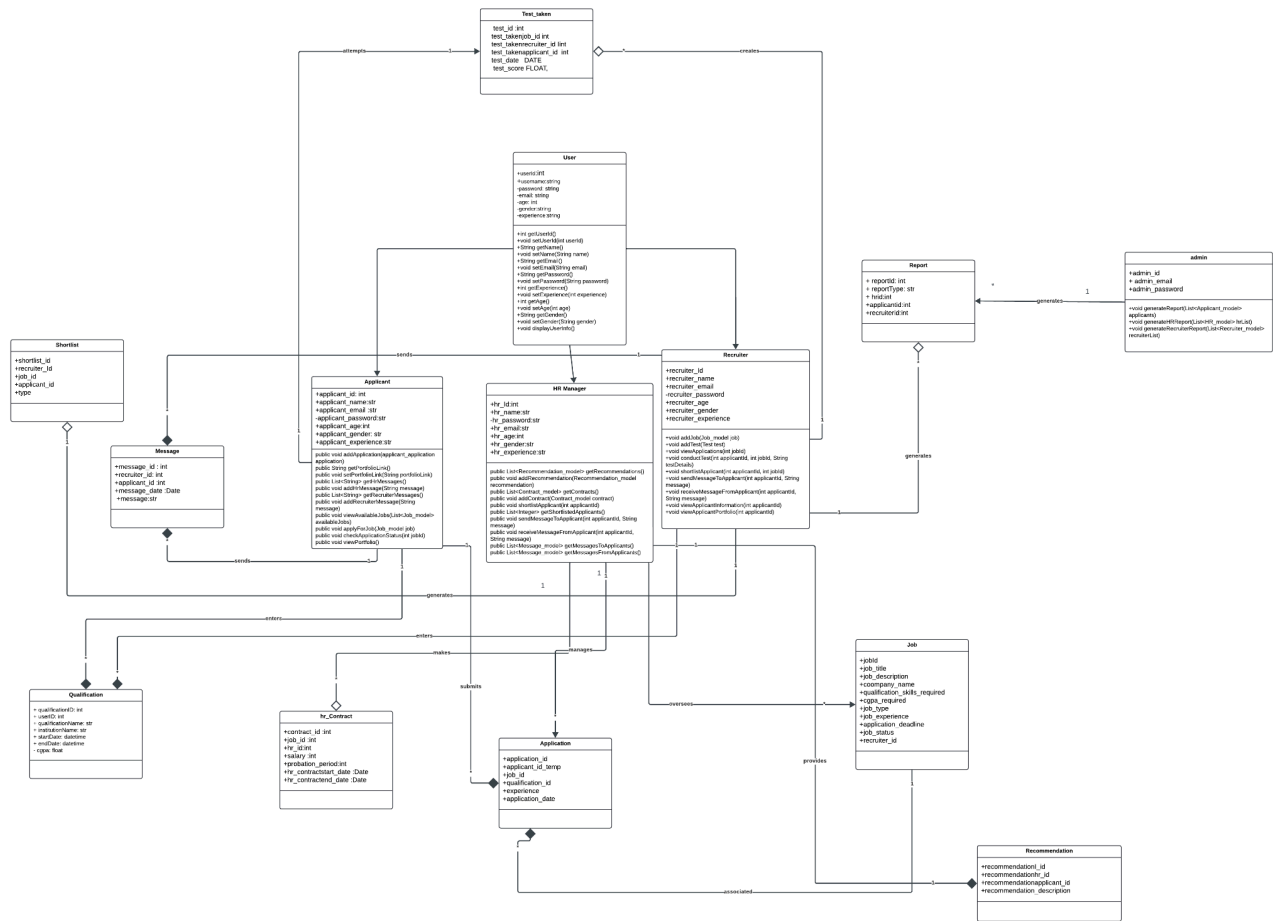


GENERATE REPORT

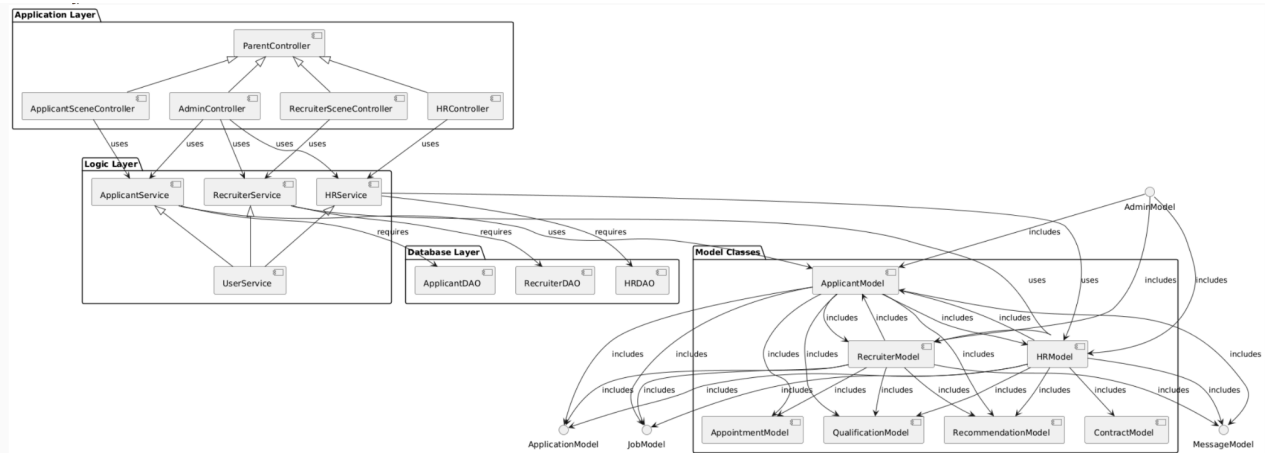


### Organize and conduct tests

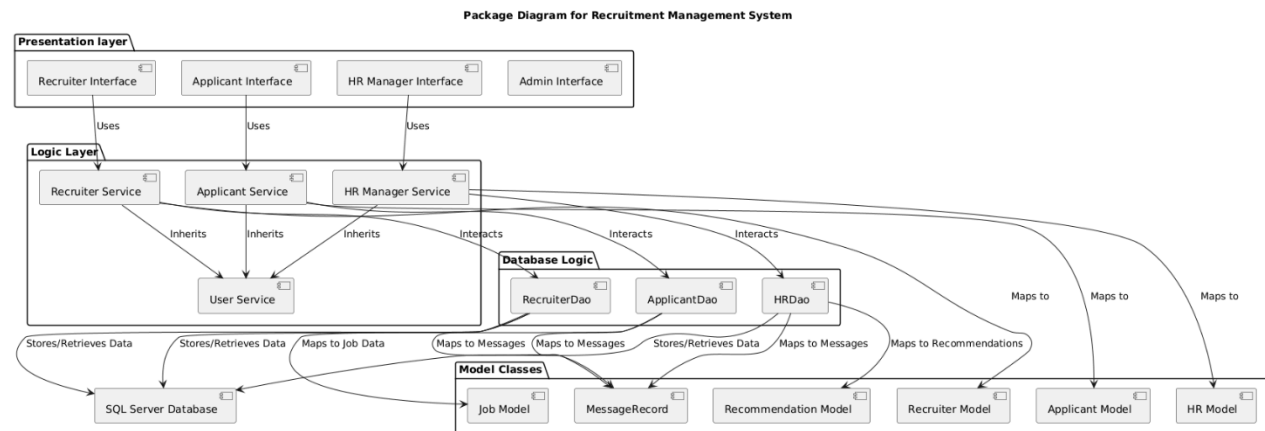




## 11. Component Diagram



## 12. Package Diagram



## 13. Deployment Diagram

