



National University
Of Computer and Emerging Sciences

Assignment 3

“Final deliverable”

Software Engineering

Member names:-

**wajiha adnan(22i1263) , eman shakeel(22i1279) ,iffah
naveed(22i0974)**

Presented to:-

Mr Basharat hussain

Dated:-4/27/2025

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 Intended Audience and Reading Suggestions

This document is intended for multiple stakeholders, each with a different level of involvement in the project. The suggested reading order for each audience is outlined below:

- **Developers & Engineers:** Focus on Sections **2 (Overall Description)** and **3 (Specific Requirements)** to understand system functionalities, constraints, and technical implementation.
- **Project Managers:** Start with Sections **1 (Introduction)** and **2 (Overall Description)** to grasp the project scope, objectives, and system workflow.
- **HR & Recruiters:** Review Section **2 (Overall Description)** and **4 (User Requirements)** to understand how the system will impact recruitment operations.
- **Testers & QA Team:** Read Section **3 (Specific Requirements)** and **5 (Verification & Validation)** to ensure system functionalities align with expected outcomes.
- **Documentation Writers:** Use Section **1 (Introduction)** as a reference when preparing user manuals and training guides.

1.3 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.

2. Functional Requirements and non Functional Requirements

Applicant Functional Requirements

- **REQ-1:** The system shall allow applicants to view available job postings.
- **REQ-2:** The system shall allow applicants to apply for available job postings.
- **REQ-3:** The system shall allow applicants to view their employment contract details.
- **REQ-4:** The system shall enable applicants to communicate with recruiters and HR through a messaging system.
- **REQ-5:** The system shall allow applicants to view their portfolio.
- **REQ-6:** The system shall allow applicants to view the status of their applications.

Recruiter Functional Requirements

- **REQ-7:** The system shall allow recruiters to manage their account details.
- **REQ-8:** The system shall allow recruiters to post new job openings.

- **REQ-9:** The system shall allow recruiters to shortlist applicants based on criteria.
 - **REQ-10:** The system shall allow recruiters to schedule assessments such as written tests and interviews.
 - **REQ-11:** The system shall allow recruiters to conduct assessments (upload test results).
 - **REQ-12:** The system shall allow recruiters to approve or reject job applications.
 - **REQ-13:** The system shall allow recruiters to manage appointment schedules (set dates/times with applicants).
 - **REQ-14:** The system shall enable recruiters to communicate with applicants.
-

HR Functional Requirements

- **REQ-15:** The system shall allow HR to generate and manage employment contracts.
- **REQ-16:** The system shall allow HR to provide recommendations for applicants to recruiters.
- **REQ-17:** The system shall enable HR to communicate with both applicants and recruiters.

Admin Functional Requirements

- **REQ-18:** The system shall allow Admin to generate reports on hr ,applicant and recruiter .

NON FUNCTIONAL:

1. Performance Requirements

- Page load time shall not exceed 2 seconds under normal conditions.
- The system shall process job applications and updates within 1 second of submission.
- The system shall support data retrieval from the database in less than 500 milliseconds.

2. Safety Requirements

- The system shall implement error handling and fail-safe mechanisms to prevent crashes.
- The system shall provide warnings and confirmation prompts before performing critical actions

3. Security Requirements

- Only verified users connect with the system.

4. Software Quality Attributes

- **Reliability:** The system shall handle failures without losing user data.
- **Maintainability:** The codebase shall follow modular design patterns to allow easy updates and enhancements.
- **Usability:** The user interface shall be intuitive and accessible to users.
- **Testability:** The system shall support automated testing and continuous integration.

5. Business Rules

- Only recruiters and HR managers can post job listings and manage candidate assessments.
- Applicants can only view and apply for jobs that match their profile and qualifications.
- Only HR managers can generate employment contracts.
- Users must verify their email addresses before applying for jobs.

3. User stories (Total- 15- 5 by each member)

Eman (HR)

Story ID:001	Story Title:Employment Contract Generation
<p>User Story:</p> <p>As an: <i>HR manager</i>,</p> <p>I want: to generate employment contracts for selected candidates</p> <p>So that: I can formalize their hiring.</p>	
<p>Acceptance Criteria:</p> <p>And I know I am done when:</p> <p>HR can create and save contracts</p> <p>Given the HR user is logged into the system, When they generate a new contract for a selected candidate, Then the system should allow them to save the contract successfully.</p> <p>Contracts are linked to selected candidates</p> <p>Given a contract has been created, When the HR user finalizes it, Then the contract should be linked to the corresponding candidate's profile.</p>	

Importance: [5/5]	Estimate: [4 hrs]
Type: <input type="checkbox"/> Search <input type="checkbox"/> Workflow <input checked="" type="checkbox"/> Manage Data <input type="checkbox"/> Payment <input type="checkbox"/> Report/View	

Story ID:002	Story Title:Provide Recommendations for Applicants
User Story: As an: <i>HR manager</i> I want: <i>to review applications and provide recommendations,</i> So that: <i>recruiters can make informed hiring decisions.</i>	
Acceptance Criteria: And I know I am done when: HR can add recommendations to applicant profiles Given the HR user is logged into the system, When they navigate to an applicant's profile and enter a recommendation, Then the system should allow them to save the recommendation successfully. Recommendations remain editable before final submission Given a recommendation has been added, When the HR user wants to make changes before final submission, Then the system should allow them to edit the recommendation.	
Importance: [4/5]	Estimate: [3 hrs]

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

Story ID:003	Story Title:Generate Recruitment Reports
<p>User Story:</p> <p>As a:HR manager</p> <p>I want:to generate recruitment reports</p> <p>So that: I can analyze hiring trends and optimize recruitment.</p>	
<p>Acceptance Criteria:</p> <p>And I know I am done when:</p> <p>HR can access reports section</p> <p>Given the HR user is logged into the system,</p> <p>When they navigate to the reports section,</p> <p>Then they should be able to generate reports based on filters.</p> <p>Reports display recruitment insights</p> <p>Given the HR user has selected a report type,</p> <p>When they generate the report,</p> <p>Then it should include relevant recruitment data.</p>	
Importance:	Estimate:
[4/5]	[4 hrs]

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

Story ID:004	Story Title:Generate HR, Recruiter, and Applicant Reports
<p>User Story:</p> <p>As an:Admin</p> <p>I want: generate comprehensive reports on HR staff, recruiters, and applicants</p> <p>So that: I can access key information for decision-making and analysis.</p>	
<p>Acceptance Criteria:</p> <p>And I know I am done when:</p> <p>Admin can Generate HR Personnel Reports</p> <p>Given the admin is logged in ,</p> <p>When I go to the Reports section and select HR</p> <p>Then I can generate reports showing HR profiles (name, role, department etc)</p> <p>Admin can Generate Recruiter Reports</p> <p>Given the admin is logged in ,</p> <p>When I go to the Reports section and select recruiter</p> <p>Then I can generate reports showing recruiter profiles (name, role, department etc)</p> <p>Admin can Generate ApplicantReports</p> <p>Given the admin is logged in ,</p> <p>When I go to the Reports section and select applicant</p> <p>Then I can generate reports showing applicant profiles (name, role, department etc)</p>	
Importance:	Estimate:
[4/5]	[5 hrs]

Type:

- ✓ Search
- ✓ Workflow
- ✓ Manage Data
- ☐ Payment
- ✓ Report/View

Wajiha (Recruiter)

Story ID:005	Story Title: Job Posting System
<p>User Story:</p> <p>As a: recruiter</p> <p>I want: I want to create, edit, and delete job postings.</p> <p>So that: So that I can attract suitable candidates.</p>	
<p>Acceptance Criteria:</p> <p>And I know I am done when:</p> <p>Given the recruiter is logged into the system,</p> <p>When they navigate to the "Post a Job" section and enter job details (e.g., title, description, location, requirements),</p> <p>Then the system should allow them to publish the job post.</p> <p>2. Job posts are visible to applicants</p> <p>Given a job post has been created by a recruiter,</p> <p>When an applicant visits the job listings page,</p> <p>Then they should be able to view the posted job with all relevant details.</p>	
Importance:	Estimate:
[5/5]	[6 hrs]

Type:

- ☐ Search
- ☐ Workflow
- ☒ Manage Data
- ☐ Payment
- ☐ Report/View

Story ID:006

Story Title: Shortlist Candidates

User Story:

As a: recruiter

I want: shortlist candidates based on predefined criteria

So that: So that I can focus on the most qualified applicants.

Acceptance Criteria:

And I know I am done when:

The system allows recruiters to filter applicants

Given the recruiter is logged into the system,

When they apply filters based on criteria such as skills, experience, or location,

Then the system should display only the applicants matching the selected criteria.

Shortlisted candidates are stored in a separate list

Given the recruiter has selected candidates for shortlisting,

When they mark an applicant as shortlisted,

Then the system should store the shortlisted candidates in a separate list.

Importance:

[5/5]

Estimate:

[4 hrs]

Type:

- ☐ Search
- ☒ Workflow
- ☐ Manage Data
- ☐ Payment
- ☐ Report/View

Story ID:007	Story Title: Schedule and Conduct Tests
<p>User Story:</p> <p>As a: recruiter</p> <p>I want: I want to schedule and conduct written tests.</p> <p>So that: So that I can assess candidates efficiently.</p>	
<p>Acceptance Criteria:</p> <p>And I know I am done when:</p> <p>The recruiter can schedule assessments</p> <p>Given the recruiter is logged into the system, When they select a candidate and schedule an assessment, Then the system should save the scheduled assessment with the date, time, and details.</p> <p>Test results are recorded</p> <p>Given a candidate has completed the assessment, When the test results are available, Then the system should record the results.</p>	
<p>Importance:</p> <p>[5/5]</p>	<p>Estimate:</p> <p>[6 hrs]</p>
<p>Type:</p> <p><input type="checkbox"/> Search</p> <p><input checked="" type="checkbox"/> Workflow</p> <p><input type="checkbox"/> Manage Data</p> <p><input type="checkbox"/> Payment</p> <p><input type="checkbox"/> Report/View</p>	

Story ID:008	Story Title: Interview Scheduling and conduction
<p>User Story:</p> <p>As a: recruiter</p>	

I want:I want to schedule interview appointments with applicants So that: I can ensure a smooth interview process.	
Acceptance Criteria: And I know I am done when: The recruiter can choose interview slots Given the recruiter is logged into the system, When they select a candidate and choose an available interview slot, Then the system should save the scheduled interview with the date and time.	
Importance: [5/5]	Estimate: [3 hrs]
Type: <input type="checkbox"/> Search <input checked="" type="checkbox"/> Workflow <input type="checkbox"/> Manage Data <input type="checkbox"/> Payment <input type="checkbox"/> Report/View	

Story ID:009	Story Title:Approve or Reject Job Applications
User Story: As a: recruiter I want: to approve or reject job applications So that:I can efficiently manage the hiring pipeline.	
Acceptance Criteria: And I know I am done when: Recruiters can review applications Given the recruiter is logged into the system, When they access the job applications list,	

<p>Then they should be able to review submitted applications.</p> <p>Recruiters can approve or reject applications</p> <p>Given a recruiter has reviewed an application,</p> <p>When they choose to approve or reject,</p> <p>Then the system should update the application status accordingly.</p>	
<p>Importance:</p> <p>[5/5]</p>	<p>Estimate:</p> <p>[3 hrs]</p>
<p>Type:</p> <p><input type="checkbox"/> Search</p> <p><input checked="" type="checkbox"/> Workflow</p> <p><input type="checkbox"/> Manage Data</p> <p><input type="checkbox"/> Payment</p> <p><input type="checkbox"/> Report/View</p>	

Story ID:010	Story Title:View Applications for the job
<p>User Story:-</p> <p>As a: recruiter</p> <p>I want:to view applications submitted for a job posting</p> <p>So that: I can review applicants' details and shortlist suitable candidates efficiently</p>	
<p>Acceptance Criteria:</p> <p>And I know I am done when:</p> <p>User authentication and access</p> <p>Given the recruiter is registered in the system,</p> <p>When they log into their account,</p> <p>Then they should be able to access the job applications section successfully.</p> <p>Application details are displayed correctly</p> <p>Given there are applications submitted for a job,</p>	

<p>When the recruiter views the applications list,</p> <p>Then the system should display applicant details such as name, qualifications, CGPA, experience, and HR recommendations accurately.</p> <p>User-friendly interface for viewing applications</p> <p>Given the recruiter is logged into the system,</p> <p>When they navigate to the job applications section,</p> <p>Then the interface should be structured, clear, and easy to use for reviewing applications.</p>	
<p>Importance:</p> <p>[4/5]</p>	<p>Estimate:</p> <p>[5 hrs]</p>
<p>Type:</p> <p><input type="checkbox"/> Search</p> <p><input type="checkbox"/> Workflow</p> <p><input checked="" type="checkbox"/> Manage Data</p> <p><input type="checkbox"/> Payment</p> <p><input checked="" type="checkbox"/> Report/View</p>	

Story ID:011	Story Title: Book Appointments
<p>User Story:</p> <p>As a: recruiter,</p> <p>I want: to schedule and book appointments with applicants,</p> <p>So that: I can efficiently arrange meetings or interviews.</p>	
<p>Acceptance Criteria:</p> <p>Selecting a Date and Time for an Appointment</p> <p>Given the recruiter wants to schedule an appointment,</p> <p>When he access the appointment scheduling system,</p> <p>Then they should be able to select an available date and time for the meeting.</p> <p>Notification for Scheduled Appointment</p>	

Given an appointment has been scheduled,
When the applicant logs into their account,
Then they can view the appointment scheduled by the recruiter

Successful Appointment Booking

Given the recruiter completes the scheduling process,
When they submit the appointment details,
Then the appointment should be successfully saved in the system.

Importance:

[4]

Estimate:

[6 hrs]

Type:

- ☐ Search
- ☒ Workflow
- ☒ Manage Data
- ☐ Payment
- ☒ Report/View

Story ID:012

Story Title: Communicate with Applicants

User Story:

As an: recruiter

I want: to communicate with applicants,

So that: I can provide updates, schedule interviews, and clarify information.

Acceptance Criteria:

Given the recruiter is logged into their account,
When they access the messaging feature,
Then the system should display a list of applicants available for communication.

Given the recruiter selects an applicant,
When they compose and send a message,
Then the system should successfully deliver the message to the applicant's inbox.

Given the applicant responds, When the recruiter opens the conversation thread, Then the system should correctly display the full conversation history.	
Importance: [4]	Estimate: [6 hrs]
Type: <input checked="" type="checkbox"/> Search <input checked="" type="checkbox"/> Workflow <input checked="" type="checkbox"/> Manage Data <input type="checkbox"/> Payment <input checked="" type="checkbox"/> Report/View	

Iffah (Applicant)

Story ID:013	Story Title:Apply for Jobs
User Story: As an: applicant I want: I want to browse and apply for job openings So that:So that I can secure a suitable position.	
Acceptance Criteria: And I know I am done when: System displays a list of available job postings Given the user is on the job listings page, When they access the job board, Then the system should display a list of available job postings. User can filter and search for jobs Given the user is on the job listings page, When they apply filters or enter search keywords, Then the system should update the job list to match the search criteria.	

Application is successfully submitted Given the user has selected a job and filled out the required application details, When they submit the application, Then the system should process and save the application successfully.	
Importance: [5/5]	Estimate: [4 hrs]
Type: <input checked="" type="checkbox"/> Search <input checked="" type="checkbox"/> Workflow <input type="checkbox"/> Manage Data <input type="checkbox"/> Payment <input type="checkbox"/> Report/View	

Story ID:014	Story Title:view Employment Contract
User Story: As an: applicant I want: I want to review my employment contract So that: I can understand my job terms before accepting	
Acceptance Criteria: And I know I am done when: Scenario: User can view contract details Given the user has the necessary permissions, When they navigate to the contract details page, Then the system should display the contract details. Scenario: Contract is displayed in a readable format Given the user is viewing a contract, When the system renders the contract details, Then the contract should be displayed in a structured and readable format.	

Importance: [4/5]	Estimate: [3 hrs]
Type: <input type="checkbox"/> Search <input type="checkbox"/> Workflow <input type="checkbox"/> Manage Data <input type="checkbox"/> Payment <input checked="" type="checkbox"/> Report/View	

Story ID:015	Story Title:Track Application Progress
User Story: As an:Applicant I want: I want to track my application status So that: I know if I have progressed in the hiring process.	
Acceptance Criteria: And I know I am done when: The system updates application status in real-time Given the user has submitted a job application, When there is a change in the application status. Then the system should update the status immediately.	
Importance: [5/5]	Estimate: [2 hrs]
Type: <input type="checkbox"/> Search <input type="checkbox"/> Workflow <input type="checkbox"/> Manage Data <input type="checkbox"/> Payment <input checked="" type="checkbox"/> Report/View	

Story ID:016	Story Title:Communicate with Recruiters and HR
User Story: As an applicant, I want: to send and receive messages from recruiters and HR, So that: I can ask questions and receive updates on my application status.	
Acceptance Criteria: User Authentication and Access to Messaging Given the applicant is registered in the system, When they log into their account, Then they should be able to access the messaging feature successfully. Initiate Conversation with Recruiter or HR Given the applicant is logged into the system, When they navigate to the messaging section and select a recruiter or HR, Then they should be able to send an initial message. Secure and Private Communication Given an applicant is using the messaging feature, When they send or receive a message, Then the conversation should be encrypted and accessible only to the involved parties.	
Importance: [5]	Estimate: [6]
Type: <input checked="" type="checkbox"/> Search <input checked="" type="checkbox"/> Workflow <input checked="" type="checkbox"/> Manage Data <input type="checkbox"/> Payment <input type="checkbox"/> Report/View	
Story ID:017	Story Title: View Appointments and Contracts

User Story:

As an: applicant

I want: to view my scheduled appointments and employment contracts,

So that: I can stay informed about upcoming meetings, interviews, and contract details.

Acceptance Criteria:

Accessing Scheduled Appointments,contracts

Accessing Scheduled Appointments:

Given the user (applicant or recruiter) is logged into their account,

When they navigate to the "View Appointments" section,

Then they should see a list of all their upcoming appointments.

Viewing Appointment Details:

Given the user has scheduled appointments,

When they select a specific appointment,

Then they should be able to view details

Accessing Contracts:

Given the user (applicant) is logged into their account,

When they navigate to the "view Contracts" section,

Then they should see a list of all their employment contracts.

Viewing Contract Details:

Given the user has employment contracts available,

When they select a specific contract,

Then the system should display the details

Importance:

[4]

Estimate:

[2 hrs]

Type:

☐ Search

☒ Workflow

☒ Manage Data

☐ Payment

☒ Report/View

Story ID:018

Story Title: View Applicant Portfolio

User Story:

As an: applicant

I want: to view my portfolio,

So that: I can see my personal and professional details clearly.

Acceptance Criteria:

Given the applicant is logged into their account,

When they navigate to the portfolio section,

Then the system should display the following details:

- Username
- Email
- Age
- Gender
- Experience

Given the applicant has updated any of these details (where allowed),

When they view their portfolio,

Then the portfolio should reflect the latest, accurate information.

Importance:

[5]

Estimate:

[1 hrs]

Type:

- ☐ Search
- ☐ Workflow
- ☐ Manage Data
- ☐ Payment
- ☒ Report/View

4. Product Backlog

1. Applicant Features

1.2 View and Apply for Available Job Postings

Priority: High

User Story: As an applicant, I want to browse and apply for jobs so that I can secure employment.

1.4 Review Employment Contract Details

Priority: High

User Story: As an applicant, I want to review my employment contract so that I can understand my employment terms.

1.4 View Appointments and Contracts

Priority: High

User Story: As an applicant i want to view my scheduled appointments and employment contracts, So that I can stay informed about upcoming meetings, interviews, and contract details.

1.5 View Portfolio

Priority: Medium

User Story: As an applicant, I want to view my portfolio so that I can showcase my work to recruiters.

1.6 Track Application Progress

Priority: High

User Story: As an applicant, I want to track my job application status so that I can stay updated on my recruitment process.

1.7 Communicate with Recruiters and HR

Priority: High

User Story: As an applicant, I want to chat with recruiters and HR so that I can get updates and clarifications.

2. Recruiter Features

2.1 Post and Manage Job Openings

Priority: High

User Story: As a recruiter, I want to post and manage job listings so that I can find suitable candidates.

2.2 Manage Applicant Shortlisting

Priority: High

User Story: As a recruiter, I want to shortlist candidates so that I can proceed with hiring the best talent.

2.3 Schedule and Conduct Assessments (Written Tests & Interviews)

Priority: High

User Story: As a recruiter, I want to schedule and manage assessments so that I can evaluate candidates.

2.4 Approve or Reject Job Applications

Priority: High

User Story: As a recruiter, I want to approve or reject applications so that I can finalize candidate selection.

2.5 Communicate with applicants

Priority: High

User Story: As a recruiter, I want to approve or reject applications so that I can finalize candidate selection.

2.6 Book and Manage Appointments

Priority: Medium

User Story: As a recruiter, I want to book and manage appointments so that I can coordinate interviews effectively.

3. HR Features

3.1 Generate Employment Contracts

Priority: High

User Story: As an HR manager, I want to create employment contracts so that I can formalize job offers.

3.2 Provide recommendations to applicants

Priority: High

User

Story: As an HR, I want to provide recommendations for applicants so that recruiters can make better hiring decisions.

3.3 Communicate with applicants

Priority: High

User Story: As an HR, I want to communicate with applicants so that we can keep them informed and engaged throughout the hiring process.

4. Admin Features

4.1 Generate Detailed Recruitment Reports for All Actors

Priority: High

User Story: As an admin, I want to generate recruitment reports so that I can analyze hiring trends.

5. Sprint1 and sprint 2 backlog

Sprint 1:

View and Apply for Available Job Postings

Priority: High

User Story: As an applicant, I want to browse and apply for jobs so that I can secure employment.

Manage Account Details

Priority: High

User Story: As a recruiter, I want to update my profile details so that my information is accurate.

Post and Manage Job Openings

Priority: High

User Story: As a recruiter, I want to post and manage job listings so that I can find suitable candidates.

Generate Employment Contracts

Priority: High

User Story: As an HR manager, I want to create employment contracts so that I can formalize job offers.

Sprint 2:

View Portfolio

Priority: Medium

User Story: As an applicant, I want to view my portfolio so that I can showcase my work to recruiters.

Track Application Progress

Priority: High

User Story: As an applicant, I want to track my job application status so that I can stay updated on my recruitment process.

Manage Applicant Shortlisting

Priority: High

User Story: As a recruiter, I want to shortlist candidates so that I can proceed with hiring the best talent.

Provide recommendations to applicants

Priority: High

User Story: As an HR, I want to provide recommendations for applicants so that recruiters can make better hiring decisions.

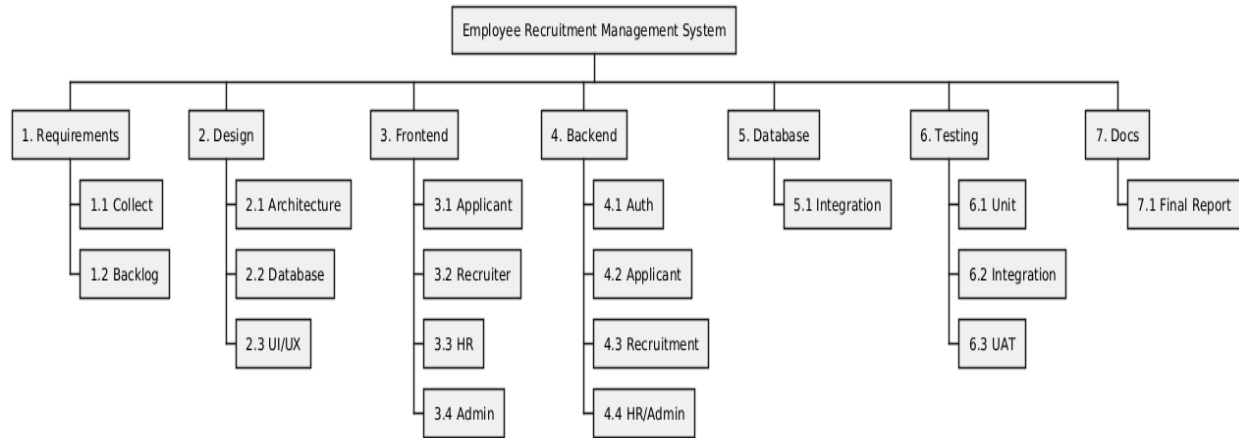
Schedule and Conduct Assessments (Written Tests & Interviews)

Priority: High

User Story: As a recruiter, I want to schedule and manage assessments so that I can evaluate candidates.

6. Project Plan

Work Breakdown Structure (WBS) of the System



Level 1: Project: Employee Recruitment Management System (ERMS)

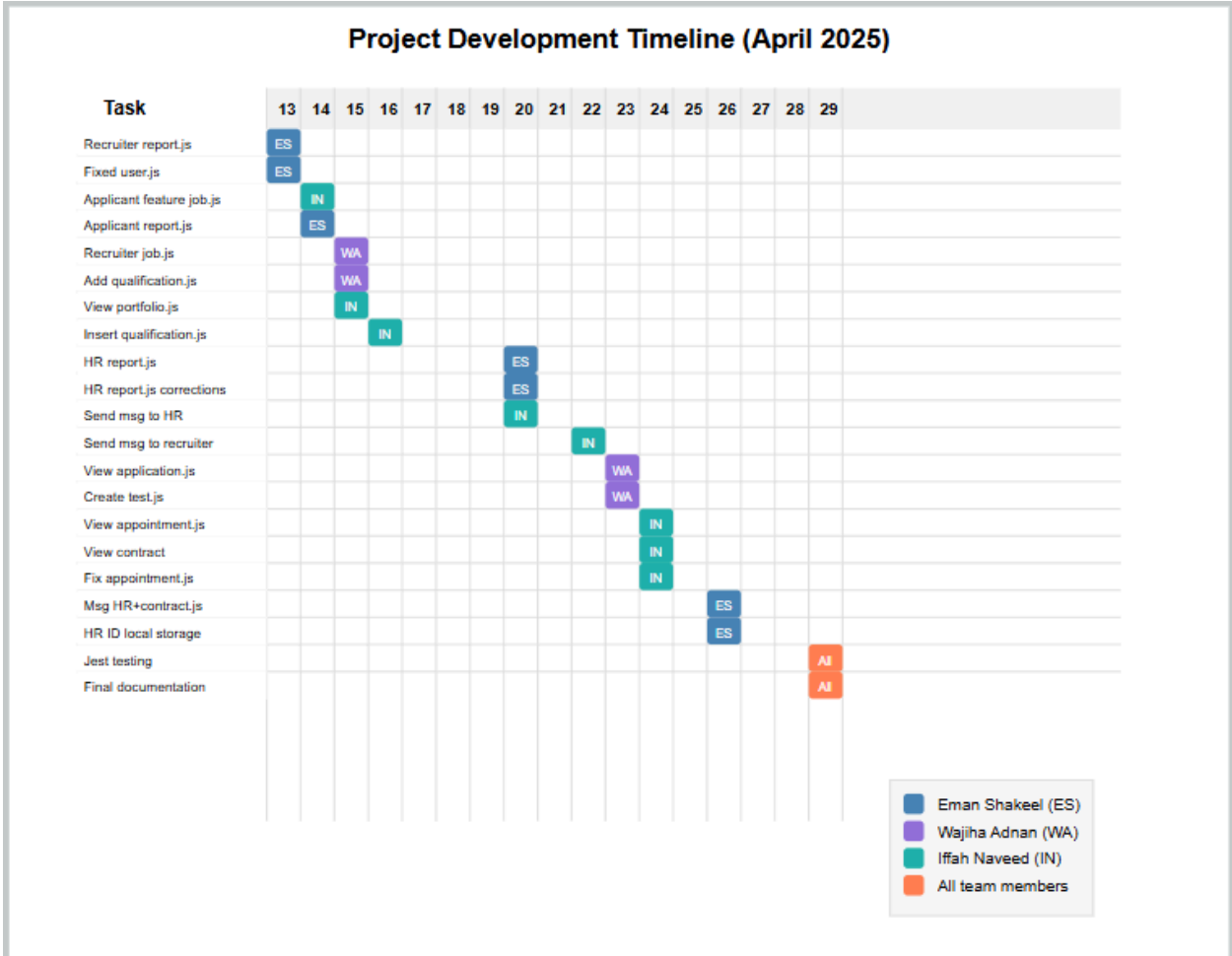
Level 2	Level 3	Tasks
1. Requirements Analysis	1.1 Collect Requirements	Gather applicant, recruiter, HR, and admin requirements.
	1.2 Define Product Backlog	Create user stories and prioritize features.
2. System Design	2.1 Design Architecture	Create Deployment, Component, and Sequence Diagrams.
	2.2 Database Design	Design tables for users, jobs, applications, contracts, portfolios, etc.
	2.3 UI/UX Design	Create wireframes and mockups for frontend (Applicant, Recruiter, HR, Admin).

3. Frontend Development	3.1 Applicant Module	Develop interfaces to apply jobs, track applications, view appointments, chat.
	3.2 Recruiter Module	Develop job posting, shortlisting, scheduling assessment UI.
	3.3 HR Module	Develop employment contract generation, recommendations UI.
	3.4 Admin Module	Develop recruitment report generation UI.
4. Backend Development	4.1 Auth Module	Implement signup, login, role management.
	4.2 Applicant Service	Implement application tracking, communication with HR/Recruiter.
	4.3 Recruitment Service	Implement job posting, assessment scheduling, shortlisting.
	4.4 HR/Admin Service	Implement contract generation, applicant communication, reports.
5. Database Integration	5.1 Connect Backend with Database	Set up PostgreSQL tables and connect APIs.
6. Testing	6.1 Unit Testing	Test individual modules (applicant, recruiter, HR, admin).

	6.2 Integration Testing	Test end-to-end functionality between frontend and backend.
	6.3 User Acceptance Testing (UAT)	Verify system meets initial requirements.
8. Documentation and Final Report	8.1 Prepare Documentation	Write user manual, final project report, and diagrams

Date	Task	file/module	Change	done by
13/4/2025	front end +backend	recruiter report.js	N/A	Eman Shakeel
13/4/2025	fixed user.js	recruiter report.js	corrected queries for retrieving the data	Eman Shakeel
14/4/2025	applicant feature	job.js	retrieved the available jobs for applicant	iffah naveed
14/04/2025	backend	applicant report.js	corrected errors of missing fields	Eman Shakeel
15/04/2025	front end +backend	recruiter job.js	implemented the logic of adding a job	wajiha adnan
15/04/2025	front end +backend	recruiter add_qualification.js	implemented the logic of adding a qualification and linking it to contains table	wajiha adnan
15/04/2025	frontend+backend	viewportfolio.js	implemented logic for viewing portfolio for applicant	iffah naveed
16/04/2025	backend	insertqualification.js	implemented qualification insertion for applicant	iffah naveed
19/04/2025	front end +backend	hrreport.js	implemented logic for displaying details of hr	Eman Shakeel
19/04/2025	backend	hrreport.js	corrected queries	Eman Shakeel

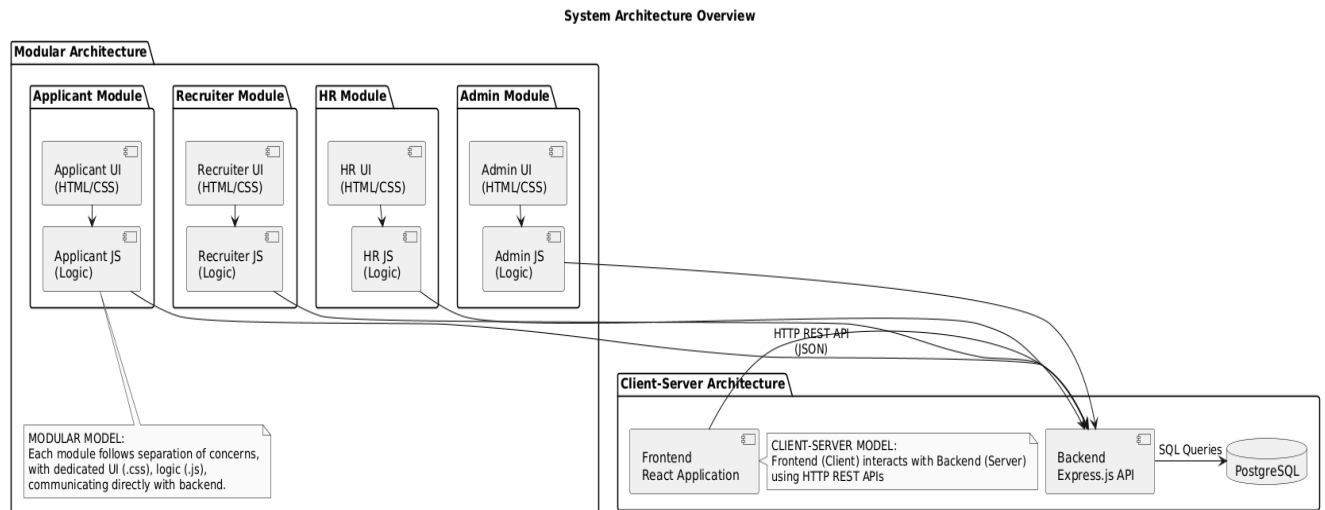
21/04/2025	frontend+backend	sendmsgrec.js + recmsgrec.js	implemented logic for it	iffah naveed
22/04/2025	frontend+backend	view_application.js	implemented the insertion logic for it	wajiha adnan
22/04/2025	frontend+backend	create_test.js	implemented the insertion logic for it	wajiha adnan
23/04/2025	frontend+backend	viewappointment.js	implementation	iffah naveed
23/04/2025	frontend+backend	viewcontract	implementation	iffah naveed
23/04/2025	backend	viewappointment.js	fixed the fetch logic	iffah naveed
25/04/2025	frontend+backend	msg_hr_appl.js +contract.js	implementation	Eman Shakeel
25/04/2025	frontend+backend	msg_hr_appl.js +contract.js	fetching hrid from local storage	Eman Shakeel
28/04/2025	testing	jest	implementing whitebox testing	all group members
28/04/2025	documentation	final report	working on documentation	



<https://docs.google.com/spreadsheets/d/1K-ZOFCLP0Ssw0Ghk1jtV7o990Z1Xy0Vf/edit?gid=1211392741#gid=1211392741>

7. Architecture Diagram

• Architecture Styles



We used a combination of these two architectural styles:

1. Modular Architecture

- You divided your system into separate modules (Applicant, Recruiter, HR/Admin).
- Each team member worked independently on one module.
- Each module had its own responsibility (Single Responsibility Principle).
- This makes the system easier to develop, test, and maintain.

Key points:

- Code for different roles is organized into different parts.
- Changes in one module (e.g., Applicant) don't break the others.
- Modules communicate with each other only where needed, mainly through APIs.

2. Client-Server Architecture

- You used React for the client (frontend).
- Express.js + Database (PostgreSQL) handled the server (backend).
- The client (browser) made API calls (HTTP requests) to the server.
- Server responded with data (e.g., successful signup/login responses).

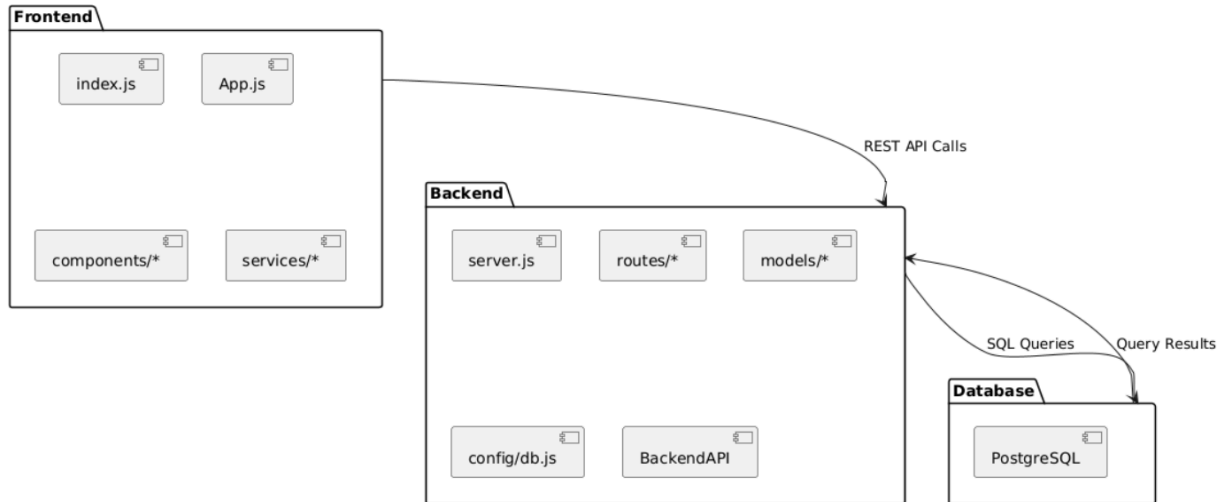
Key points:

- Client (React app) and Server (Express backend) are separate.
- They communicate over a network using HTTP APIs.
- Server handles data storage, validation, business logic.
- Client handles displaying UI and sending user inputs.

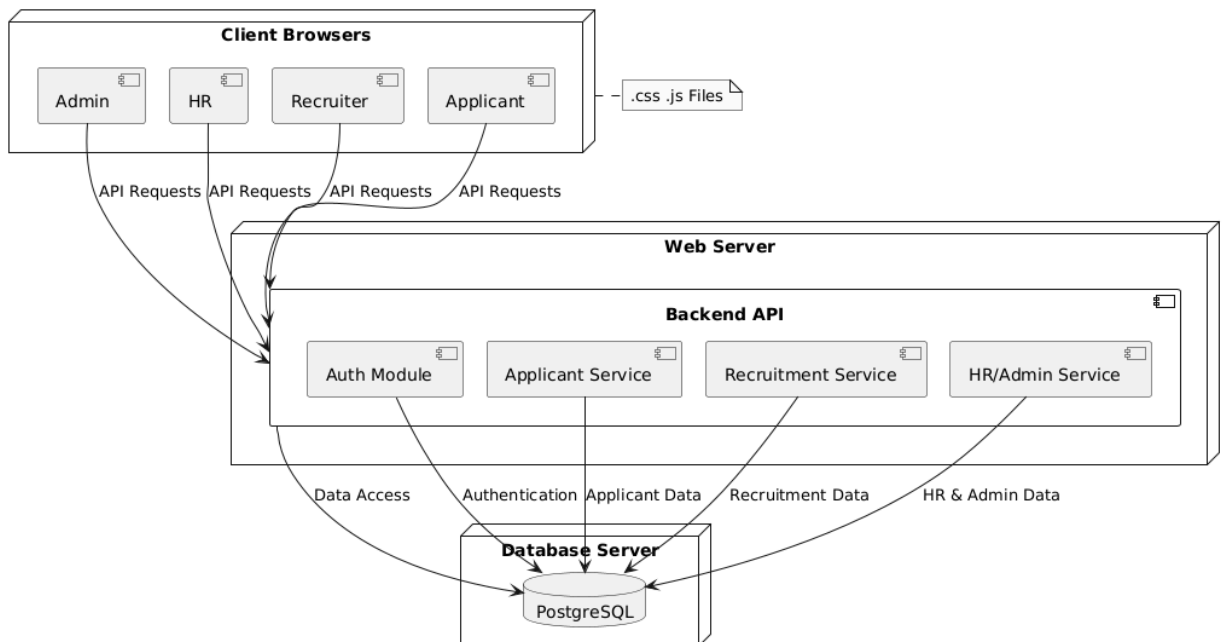
• Identifying Subsystems

Subsystem	Responsibilities
Applicant	Apply to Jobs, View status,getting jobs based on qualifications/tests
Recruiter	Post Jobs, View Applicants, Shortlist, Schedule Interviews
HR/Admin	Manage Recruiters/Applicants, handle reports and contracts, provides recommendations

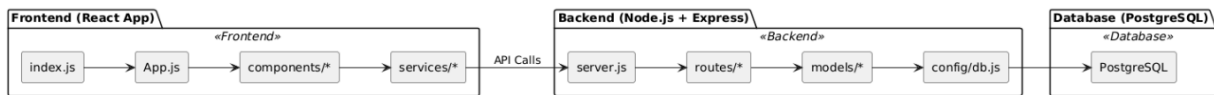
- Draw and describe UML package diagram



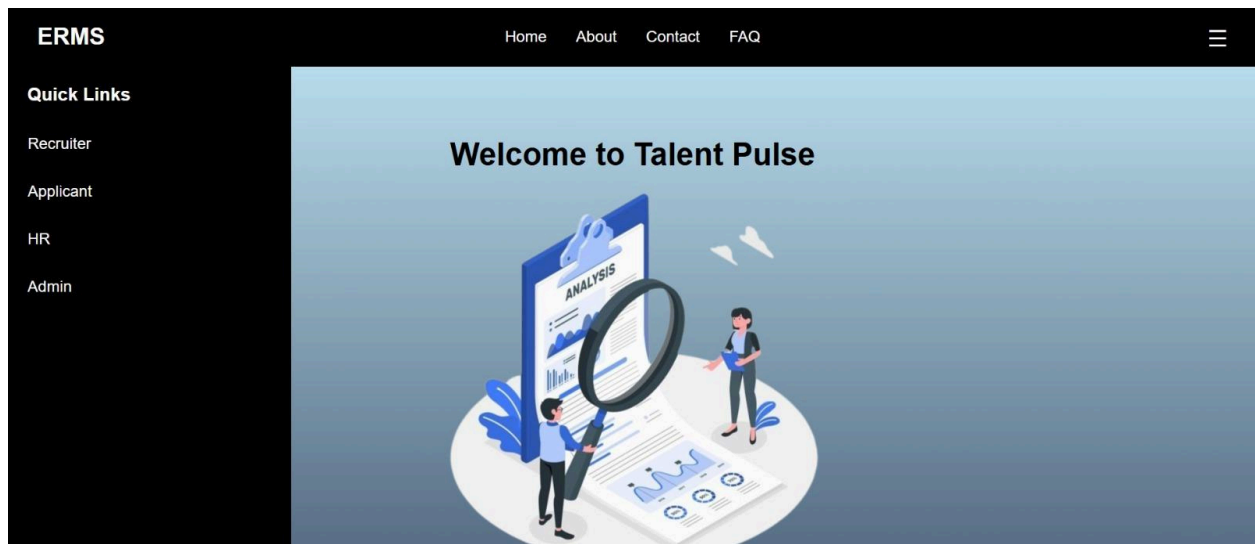
- Deployment diagram for client deployments




- Component diagrams



8. Actual implementation screenshots



Applicant ID	Qualification ID	Experience	GPA	Application Date	Action
1	1	2 yrs	3.5	2025-04-09T19:00:00.000Z	<button>Chat</button>


Chat with Applicant ID: 1

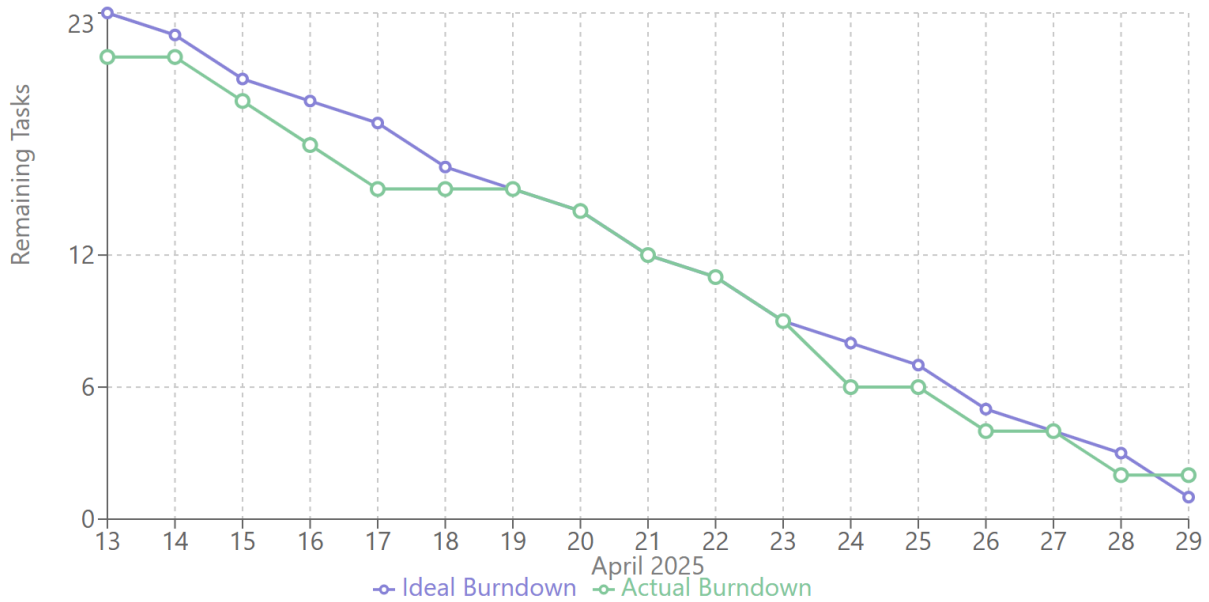
hello how are you
Invalid Date
hello kese ho
Invalid Date
Hello Recruiter, test message from applicant.
Invalid Date
Hello Recruiter, test message from applicant.
Invalid Date
Hello Recruiter, test message from applicant.
Invalid Date
Hello Recruiter, test message from applicant.
Invalid Date

Available Jobs

Job Title	Company	Skills Required	Job Type	Experience	Deadline	Action
frontend developer	cisco	bachelors in cs	Full	1 years	4/22/2025	<button>Apply</button>
cisoc	cisco	bachelor in cs	Full time	1 years	4/30/2025	<button>Apply</button>
full stack developer	wajiha's company	Bachelors in cs	Full-time	1 years	4/22/2025	<button>Apply</button>

9. Product Burn down chart for the project

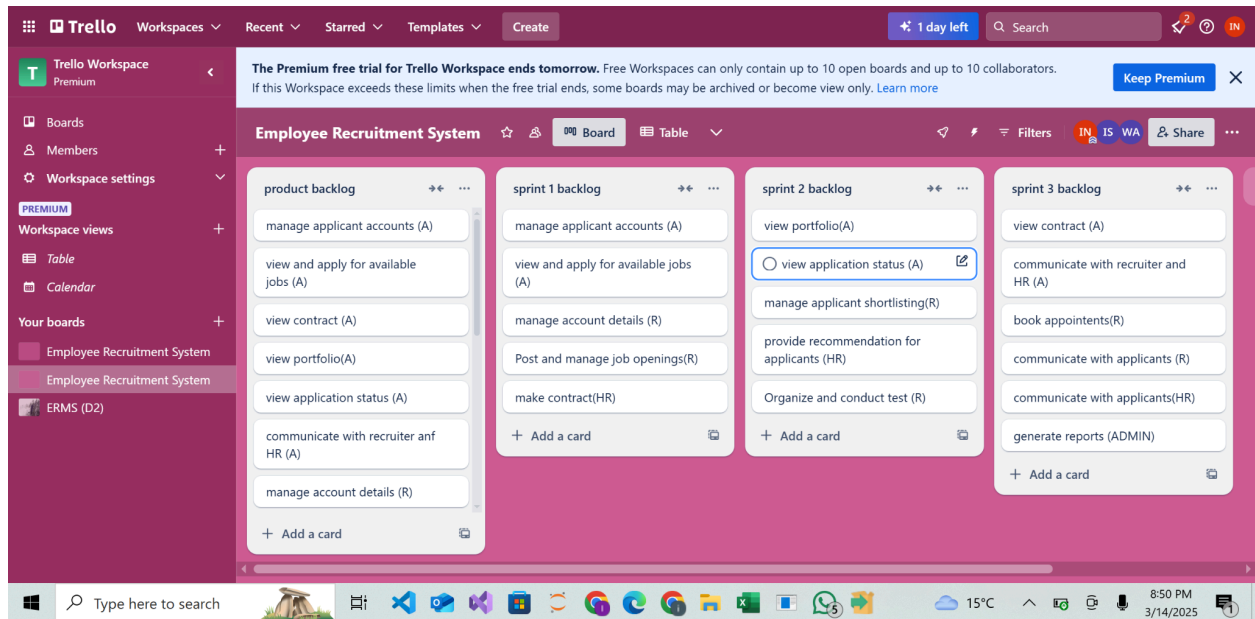
Project Burndown Chart (April 2025)



Burndown Chart Analysis:

- Total tasks: 23
- Project duration: 17 days (April 13-29, 2025)
- All tasks completed by project end: Yes
- Team members: Eman Shakeel (ES), Wajha Adnan (WA), Iffah Naveed (IN), All team members

10. Trello board screenshots



11. Black box Testing

Hr:

TestCase ID	Description	Input	Expected Output	Actual Output	Pass/Fail
TC01	Login with valid credentials (ECP)	Valid email, valid password (>=8 chars)	Login successful, redirect to HR menu	Login successful, redirect to HR menu	Pass
TC02	Login with invalid email format (BVA)	Invalid email,password(missing @)	"Please enter a valid email address"	"Please enter a valid email address"	Pass

TC03	Login with short password (BVA)	Valid email, password (7 chars)	"Password must be at least 8 characters long"	"Password must be at least 8 characters long"	Pass
TC04	Login with empty fields (BVA)	Empty email, empty password	"Please enter both email and password"	"Please enter both email and password"	Pass
TC05	Login with non-existent email (ECP)	Unregistered email	"User not found"	"User not found"	Pass
TC06	Signup with valid details (ECP)	Valid name, email, password (>=8 chars), age=22, gender=Male, experience=2	"Signup successful! Redirecting to login..."	"Signup successful! Redirecting to login..."	Pass
TC07	Signup with invalid email (BVA)	Valid name, invalid email (no domain), valid password, age=22, gender=Male, experience=2	"Please enter a valid email address"	"Please enter a valid email address"	Pass
TC08	Signup with short password (BVA)	Valid name, valid email, password (5 chars), age=22, gender=Male, experience=2	"Password must be at least 8 characters long"	"Password must be at least 8 characters long"	Pass

TC09	Signup with invalid age (BVA)	Valid name, valid email, valid password, age=17, gender=Male, experience=2	"Age must be between 18 and 100"	"Age must be between 18 and 100"	Pass
TC10	Signup with missing gender (BVA)	Valid name, valid email, valid password, age=22, gender empty, experience=2	"Please select a gender"	"Please select a gender"	Pass
TC11	Signup with invalid experience (BVA)	Valid name, valid email, valid password, age=22, gender=Male, experience=51	"Experience must be between 0-50 years"	"Experience must be between 0-50 years"	Pass
T12	Create contract with valid data (ECP)	All fields filled with valid data (job selected, salary=50000, probation=30, start=2024-01-01, end=2024-12-31, benefits="Health insurance")	"Contract Created Successfully! " with contract details displayed	"Contract Created Successfully! " with contract details displayed	Pass
T13	Missing required fields (BVA)	Empty required fields (job, salary, probation, dates, benefits)	Alert "Please fill in all required fields: job_id, salary, probation_period, start_date, end_date, benefits"	Alert with correct missing fields	Pass

T14	Invalid salary (BVA)	Salary=0,-10	Alert "Salary must be a positive number."	Alert "Salary must be a positive number."	Pass
T15	Invalid probation period (BVA)	Probation=0	Alert "Probation period must be a positive number."	Alert "Probation period must be a positive number."	Pass
T16	Invalid date range (BVA)	start=2024-12-31, end=2024-01-01	Alert "Contract start date must be before the end date."	Alert "Contract start date must be before the end date."	Pass
T17	Same start and end date (BVA)	start=2024-01-01, end=2024-01-01	"Start date must be before end date"	"Start date must be before end date"	Pass
T18	Form reset after submission	After successful submission	Form fields cleared except HR_ID	Form fields cleared except HR_ID	Pass
T19	Submit valid recommendation (ECP)	Select HR, select their shortlisted applicant, choose recommendation	"Recommendation submitted successfully!"	"Recommendation submitted successfully!"	Pass
T20	No shortlisted applicants (BVA)	HR with no shortlisted	"No shortlisted	"No shortlisted	Pass

		applicants selected	applicants for this HR"	applicants for this HR"	
T21	Send valid message (ECP)	Select applicant, enter message (>5 chars)	"Message sent successfully!"	"Message sent successfully!"	Pass
T22	Missing message (BVA)	No message entered	"Please type a message"	"Please type a message"	Pass
T23	Message reset after send	After successful send	Message field clears	Message field clears	Pass
T24	Empty dropdowns	No data available	Shows "No applicants/HR s available"	Shows empty state	Pass
T25	Back button functionality	Click back button	Navigates to previous page	Navigates to previous page	Pass

ADMIN:

TestCase ID	Description	Input	Expected Output	Actual Output	Pass/Fail

TC01	Login with valid admin credentials (ECP)	Valid admin email, correct password	"Login successful redirect to admin menu"	"Login successful redirect to admin menu"	Pass
TC02	Invalid email format (BVA)	"admin@example" (missing domain)	"Please enter a valid email address."	"Please enter a valid email address."	Pass
TC03	Empty email (BVA)	Empty email, valid password	"Please enter a valid email address."	"Please enter a valid email address."	Pass
TC04	Empty password field (BVA)	Valid email, empty password	"Password cannot be empty."	"Password cannot be empty"	Pass
TC05	No applicants available (BVA)	Empty applicant list	"No applicants found"	"No applicants found" message	Pass
TC06	Select valid applicant(ECP)	Choose applicant	Shows detailed applicant information	Shows correct details	Pass
TC07	No HRs available	Empty hr list	Dropdown shows only default option	Shows only "-- Choose --"	Pass

TC08	Select valid Hr	Choose HR from dropdown	Shows detailed HR information	Shows correct details	Pass
TC09	No recruiters available	Empty recruiter list	"No recruiters found" message	"No recruiter found"	Pass
TC10	Select valid recruiter	Choose recruiter from dropdown	Shows detailed recruiter information	Shows correct details	Pass

Recruiter:

TestCase ID	Description	Input	Expected Output	Actual Output	Pass/Fail
TC01	Signup with valid details (ECP)	Valid username, email, password, age=22, gender=Male	Account created successfully	Account created successfully	Pass
TC02	Signup with existing email (ECP)	Existing email, new username and password	"Email already exists" error	"Email already exists" error	Pass

TC03	Signup with age>=18(ECP)	Valid username, email, password, age=17, gender=Female, experience=1	Error: Age must be greater than 18	Error: Age must be greater than 18	Pass
TC04	Signup with experience ≥ age (ECP)	Valid username, email, password, age=22, gender=Male, experience=22	Error: Experience must be less than age	Error: Experience must be less than age	Pass
TC05	Login with non existing email (ECP)	Unregistered email and any password	Error: No account exists	Error: No account exists	Pass
TC06	Add Qualification with all valid fields(ECP)	qualType="Bachelors", university="MIT", yearCompleted="2022-05-15", cgpa="3.8", field="Computer Science"	Qualification added and linked to recruiter	Qualification added and linked to recruiter	Pass
TC07	Submit with invalid CGPA (>4)(BVA)	qualType="Bachelors", university="MIT", yearCompleted="2022-05-15", cgpa="4.5", field="Cybersecurity"	Failed to add qualification	Failed to add qualification	Pass
TC08	Invalid yearCompleted	yearCompleted="abcd"	Failed to add qualification	Failed to add qualification	Pass

	format (non-date)(ECP)				
TC09	Create test with valid fields(BVA)	jobId="101",noofQuestions="20,min score="70.5"	Test created successfully	Test created successfully	Pass
TC10	Submit with 0 questions(ECP)	jobId="102", noOfQuestions="0", minScore="60"	Failed to create test	Failed to create test	Pass
TC11	Invalid minScore format (string instead of float)(BVA)	minScore="abc"	Failed to create test	Failed to create test	Pass
TC12	Submit with negative minimum score(ECP)	jobId="101",noofQuestions="20,min score="-70.5"	Failed to create test	Failed to create test	Pass
TC13	Submit with missing recruiterId(BVA)	localStorage recruiterId=null, all fields filled properly	Failed to create test	Failed to create test	Pass
TC14	Valid job post(ECP)	All fields filled correctly	Success message: "Job posted successfully!"	Success message: "Job posted successfully!"	Pass
TC15	Post job with empty Job Title(ECP)	Leave Job Title empty and click Post Job	Error message displayed	Error message displayed	Pass

			(Job Title required)	(Job Title required)	
TC16	Post job without selecting job type (ECP)	Leave Job Type unselected and click Post Job	Error message displayed (Job Type required)	Error message displayed (Job Type required)	Pass
TC17	Post job with negative experience(BVA)	Enter experience as -2 and click Post Job	Error message displayed (Invalid experience)	Error message displayed (Invalid experience)	Pass
TC18	Post job with status 'Closed' (ECP)	Set Job Status to 'Closed' and click Post Job	Job posted successfully with closed status	Job posted successfully with closed status	Pass
TC19	Fetch and display jobs posted by recruiter(ECP)	Set a valid recruiterId in localStorage, load page	List of jobs posted by recruiter displayed	List of jobs posted by recruiter displayed	Pass
TC20	Fetch applications when job is clicked(ECP)	Click on a job title in the list	Applications for selected job are fetched and displayed	Applications displayed correctly in table	Pass
TC21	Fetch and display recruiter profile (ECP)	Set a valid recruiter id in local storage	Recruiter profile details (username, email, age, etc.) are displayed	Recruiter profile displayed correctly	Pass

TC22	Handle no qualifications available for recruiter(ECP)	Load page with recruiterId having no qualifications	Message "No qualifications found." displayed	Message displayed correctly	Pass
------	---	---	--	-----------------------------	------

Applicant:

TestCase ID	Description	Input	Expected Output	Actual Output	Pass/Fail
TC01	Signup with valid details (ECP)	Valid username, email, password, age=22, gender=Male	User created successfully	User created successfully	Pass
TC02	Signup with existing email (ECP)	Existing email, new username and password	"Email already exists" error	"Email already exists" error	Pass
TC03	Signup with invalid email format (ECP)	username, "invalidemail", password	Validation error: Invalid email	Validation error: Invalid email	Pass
TC04	Signup with minimum boundary age (BVA)	age = 18	User created successfully	User created successfully	Pass
TC05	Signup with below minimum age (BVA)	age = 17	Validation error: Age must be ≥ 18	Validation error: Age must be ≥ 18	Pass

TC06	Login with correct email and password (ECP)	Correct email and password	Login successful with user ID	Login successful with user ID	Pass
TC07	Login with wrong password (ECP)	Correct email, wrong password	"Invalid password" error	"Invalid password" error	Pass
TC08	View applicant profile after login (Functionality)	Logged-in applicant ID	Applicant profile displayed	Applicant profile displayed	Pass
TC09	Apply for a job with eligible CGPA (ECP)	Applicant ID, Job ID (CGPA met)	Application submitted successfully	Application submitted successfully	Pass
TC10	Apply for a job with below-required CGPA (ECP)	Applicant ID, Job ID (CGPA not met)	"Not eligible to apply" error	"Not eligible to apply" error	Pass

12. White box Testing(Jest)


```

    at server/config/db.js:30:25

console.log
    Applying with: 1 1

    at server/server.js:47:13
[PASS] server/applicant.test.js
Applicant API Tests
  ✓ POST /api/qualification - Add a qualification (140 ms)
  ✓ POST /api/jobs/apply - Apply for a job (13 ms)
  ✓ POST /api/messages - Send message to HR (8 ms)
  ✓ POST /api/messages/recruiter - Send message to recruiter (10 ms)
  ✓ GET /api/messages/fromhr - Fetch latest HR message (7 ms)
  ✓ GET /api/messages/fromrecruiter - Fetch latest recruiter message (8 ms)
  ✓ GET /api/appointments - Fetch appointments with recruiter (6 ms)

Test Suites: 1 passed, 1 total
Tests:       7 passed, 7 total
Snapshots:   0 total
Time:        0.892 s, estimated 4 s
Ran all test suites matching /applicant.test.js/i.
Jest did not exit one second after the test run has completed.

This usually means that there are asynchronous operations that weren't stopped in your tests. Consider running Jest with '--detectOpenHandles' to troubleshoot this issue.

```

```

  ✓ should create a new contract (16 ms)
GET /api/auth/hrs/:id
  ✓ should fetch a single HR by ID (8 ms)
GET /api/auth/applicants
  ✓ should fetch all applicants (8 ms)
GET /api/auth/applicants/:id
  ✓ should fetch a single applicant by ID (8 ms)
GET /api/auth/recruiters
  ✓ should fetch all recruiters (8 ms)
GET /api/auth/recruiters/:id
  ✓ should fetch a single recruiter by ID (6 ms)
POST /api/auth/recommendations
  ✓ should create a new recommendation (28 ms)
GET /api/auth/recommendations
  ✓ should fetch all recommendations (8 ms)
GET /api/auth/recommendations/applicant/:id
  ✓ should fetch recommendations for an applicant (5 ms)
GET /api/auth/recommendations/hr/:id
  ✓ should fetch recommendations by HR ID (4 ms)

Test Suites: 1 passed, 1 total
Tests:       14 passed, 14 total
Snapshots:   0 total
Time:        2.19 s
Ran all test suites matching /Hr.test.js/i.
Jest did not exit one second after the test run has completed.

This usually means that there are asynchronous operations that weren't stopped in your tests. Consider running Jest with '--detectOpenHandles' to troubleshoot this issue.

```

```

    ✓ 201 adds qualification (7 ms)
GET /api/auth/recruiter/:recruiterId
    ✓ 200 returns recruiter & quals (15 ms)
GET /api/auth/recruiter/:id/jobs
    ✓ 200 returns jobs (11 ms)
GET /api/auth/recruiter/:id/job-ids
    ✓ 200 returns ids (9 ms)
GET /api/auth/jobs/:jobId/applications
    ✓ 200 returns applications (3 ms)
POST /api/auth/tests
    ✓ 201 creates test (4 ms)
GET /api/auth/job/:jobId/applicants
    ✓ 200 returns applicants (5 ms)
POST /api/auth/testEvaluation/submit-score
    ✓ 201 & shortlist (5 ms)
    ✓ 400 if below minimum (3 ms)
POST /api/auth/appointments
    ✓ 201 books appointment (4 ms)
POST /api/auth/messages
    ✓ 200 sends message (6 ms)
    ✓ 400 bad query params (5 ms)
    ✓ 200 fetches merged (3 ms)
POST /api/auth/shortlist/:applicationId
    ✓ 201 shortlists (4 ms)
    ✓ 404 if not found (5 ms)

Test Suites: 1 passed, 1 total
Tests:       22 passed, 22 total
Snapshots:   0 total
Time:        2.217 s
Ran all test suites matching /recruiter.test.js/i.

D:\github_se\se_project_recruiter\SErestored\se_project_recruiter\my-app\server\routes>

```

13. Work Division between group members

Member	Responsibilities
Eman (HR,Admin)	Contracts, Recommendations, HR Communication,Reports Generation
Wajiha (Recruiter)	Job postings, Shortlisting, Tests, Applicant communication
Iffah (Applicant)	Apply to jobs, View contracts, Track application, Manage portfolio

14. Lesson learnt by group

Lessons Learned from ERMS Project (Software Engineering Perspective)

- **Requirement Engineering:** Clear, detailed requirements (functional & non-functional) are crucial for defining system goals and reducing ambiguity.
- **User-Centered Design:** Prioritizing end-user needs (HR, recruiters, applicants) ensures a more intuitive and effective system.
- **Scalability & Performance:** Early performance testing for handling large user bases prevents future bottlenecks.
- **Collaboration & Communication:** Effective team collaboration with tools (Trello, sprint planning) enhances progress tracking and issue resolution.
- **Testing & Quality Assurance:** Comprehensive testing (unit, integration, system testing) ensures system meets functional and non-functional requirements.
- **Dependency Management:** Properly managing system dependencies ensures seamless interaction between subsystems (HR, recruiter, applicant).
- **Project Management:** Structured planning (WBS, Gantt charts) and sprint planning ensure project scope and deadlines are met efficiently.