



**NANYANG  
TECHNOLOGICAL  
UNIVERSITY**  
**SINGAPORE**

# **GROUP 3 SC2006 - SCSD**

## **LAB 2 DELIVERABLES**

Team Members	
1.	Vivek Shrey (U2440609F) - Team Leader
2.	Ching Xin Wei (U2421032K)
3.	Dharshini Saravanan (U2420758E)
4.	Li Sinuo (U2421736H)
5.	Sivasankar Sivagnana Sundari (U2420528F)

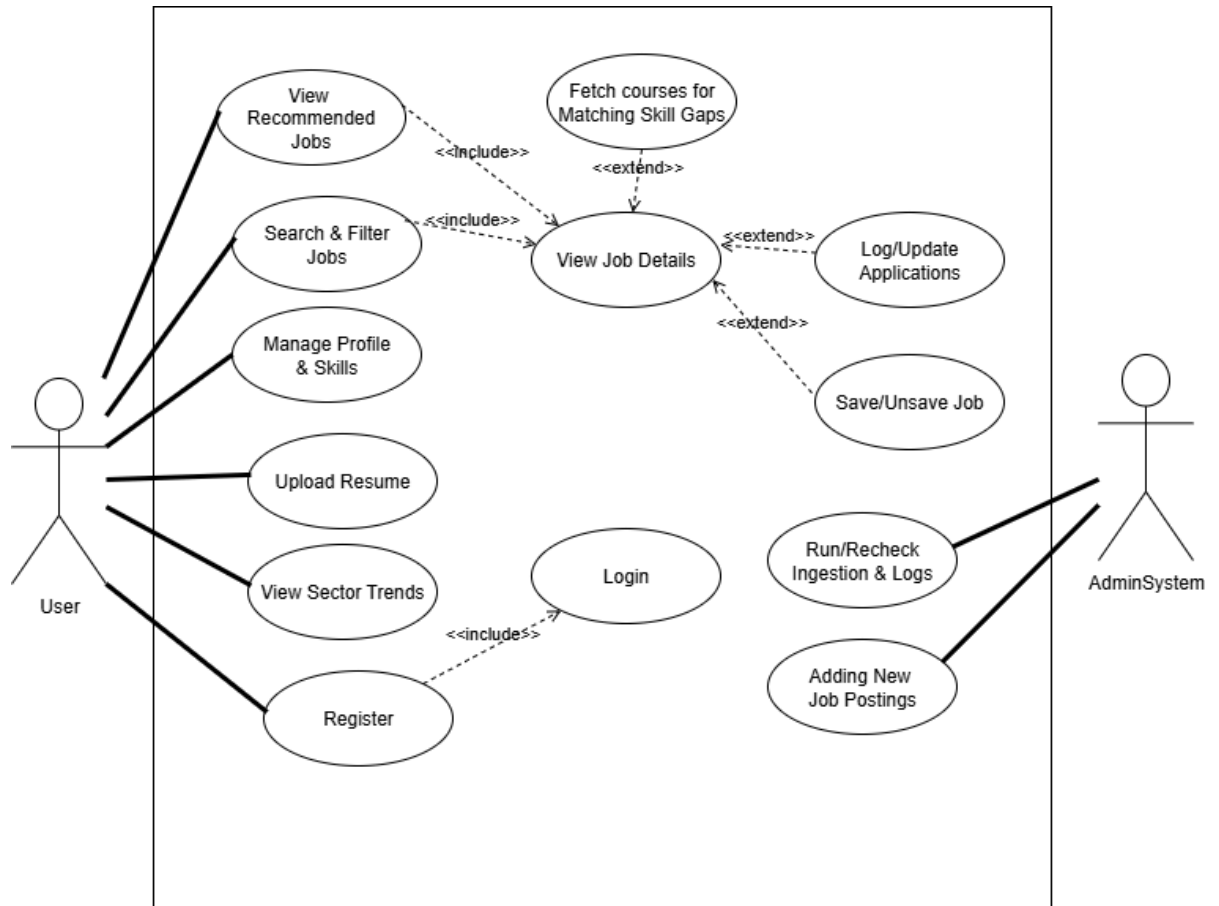
## **Use Cases**

1. UC-1 Register
2. UC-2 Login
3. UC-3 Manage Profile & Skills
4. UC-4 Upload Resume
5. UC-5 View Sector Trends
6. UC-6 View Recommended Jobs
7. UC-7 Search & Filter Jobs
8. UC-8 View Job Details
9. UC-9 Save/Unsave Job
10. UC-10 Log/Update Application
11. UC-11 Fetch Courses for Matching Skill Gaps
12. UC-12 AdminSystem: Run/Recheck Ingestion & Logs
13. UC-13 AdminSystem: Adding New Job Postings

### **Include/Extend:**

- **UC-1** is <<include>> in UC-3/4/5/6/7.
- **UC-8** is <<include>> in UC-6/7.
- **UC-2** is <<include>> in UC-1.
- **UC-9** <<extend>> UC-8; **UC-10** <<extend>> UC-8; **UC-11** <<extend>> UC-8.

## Use Case Diagram



(clearer image uploaded directly to GitHub)

## Use-Case Descriptions

Use Case ID:	<b>UC-1</b>		
Use Case Name:	<b>Register</b>		
Created By:	Sundari	Last Updated By:	Sundari
Date Created:	12 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	This is practically the first step for a new user to enter the application.
Preconditions:	The application is started and User is prompted to register.
Postconditions:	User has a newly created account and are logged in OR User is given a valid explanation for why their registration failed.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"><li>1. System displays fields for registration email and password.</li><li>2. User enters their details into the field.</li><li>3. User presses the Register button.</li><li>4. System verifies their email and password and creates a new account for them in the system database.</li><li>5. If the account is created, system displays a message that says 'Registration is successful!'</li><li>6. User is brought to the home page / landing screen.</li></ol>
Alternative Flows:	<p><b>AF-S1:</b> If User leaves the field blank</p> <ol style="list-style-type: none"><li>1. System highlights missing fields with error message "This field is required"</li></ol>

	<b>AF-S2:</b> If the email or password is entered incorrectly <ol style="list-style-type: none"> <li>1. The system displays a message saying 'Invalid email or password not matching criteria! Please Try Again!'</li> <li>2. Return to Step 1.</li> </ol>
Exceptions:	<b>EX-1:</b> System cannot reach database → Error: "Registration service temporarily unavailable".  <b>EX-2:</b> Network disconnected → Error: "Please check your internet connection".
Includes:	UC-2.
Special Requirements:	Passwords must be stored with secure hashing + salting
Assumptions:	User has access to their own email
Notes and Issues:	None

Use Case ID:	<b>UC-2</b>		
Use Case Name:	<b>Login</b>		
Created By:	Vivek Shrey	Last Updated By:	Vivek Shrey
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	This is practically the first step for an existing user to enter the application.
Preconditions:	The user already has an existing account, and has been authenticated previously.
Postconditions:	User has logged into their account OR User is given a valid explanation for why their login failed.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> <li>1. System displays fields for email and password</li> <li>2. User enters their details into the field.</li> <li>3. User presses the login button.</li> <li>4. System verifies their email and password with the account information in the system database.</li> <li>5. If the account is verified, system displays a message that says 'Login is successful!'</li> <li>6. User is brought to the home page / landing screen.</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> If the email or password is entered incorrectly</p> <ol style="list-style-type: none"> <li>1. The system displays a message saying 'Invalid email or password! Please Try Again!'</li> <li>2. Return to Step 3.</li> </ol>

	<b>AF-S2:</b> If User leaves the field blank  1. System highlights missing fields with error message "This field is required"
Exceptions:	<b>EX-1:</b> System cannot reach database → Error: "Login service temporarily unavailable".  <b>EX-2:</b> Network disconnected → Error: "Please check your internet connection".
Includes:	Included by UC-1.
Special Requirements:	Passwords must be stored with secure hashing + salting
Assumptions:	User has access to their own email
Notes and Issues:	None

Use Case ID:	<b>UC-3</b>		
Use Case Name:	<b>Manage Profile and Skills</b>		
Created By:	Vivek Shrey	Last Updated By:	Vivek Shrey
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	This is the section where users will update their profile and update their current skills that are relevant for their job hunt.
Preconditions:	User is Authenticated and logged in.
Postconditions:	System displays the updated profile and skills.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on profile.</li> <li>2. System displays any existing profile tied to the user, displaying the profile and skills</li> <li>3. User clicks the option to manage profile and skills.</li> <li>4. System displays the profile and skill fields that can be edited.</li> <li>5. User enter their changes for the fields and confirms their changes.</li> <li>6. System verifies these changes and confirms these changes to its database, displaying a message to the user "Your profile has been successfully changed!"</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> If any of User's profile fields is empty</p> <ol style="list-style-type: none"> <li>1. System displays an empty profile field, with a message that says "You have not added anything to this field! Add to it now!"</li> </ol>



	<p>2. Return to Step 3.</p> <p><b>AF-S2:</b> If User enters invalid skill proficiency (e.g. &lt;2)</p> <p>1. System rejects and highlights error</p> <p><b>AF-S3:</b> If User attempts to delete all skills</p> <p>1. System displays warning: “At least one skill must be present for recommendations.”</p>
Exceptions:	<b>EX-1:</b> Profile update fails due to database error → System displays “Update failed. Please try again later.”
Includes:	UC-1
Special Requirements:	Changes should sync across all modules within 5 seconds.
Assumptions:	None
Notes and Issues:	None

Use Case ID:	<b>UC-4</b>		
Use Case Name:	<b>Upload Resume</b>		
Created By:	Vivek Shrey	Last Updated By:	Vivek Shrey
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	The user is prompted to optionally upload their resume to keep their profile updated.
Preconditions:	User is authenticated; existing resume to be uploaded.
Postconditions:	Resume stored securely; system displays the updated resume.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on profile.</li> <li>2. System displays any existing profile tied to the user, displaying the profile and skills.</li> <li>3. User clicks the option to update their resume.</li> <li>4. System displays a message to "Upload your resume here!" and button to upload said resume.</li> <li>5. System displays a preview of the resume and User confirms their changes.</li> <li>6. System verifies these changes and confirms these changes to the resume to its database, displaying a message to the user "Your resume has been successfully updated!"</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> File too large/Invalid Format</p> <ol style="list-style-type: none"> <li>1. System rejects with the error "File exceeds 5 MB. Please upload a smaller file" or "File should be in PDF format."</li> </ol>

	<p>2. Return to Step 3.</p> <p><b>AF-S2:</b> If User's resume is empty</p> <ol style="list-style-type: none"> <li>1. System displays an empty resume page, with a message that says "You have not added your resume! Add to it now!"</li> <li>2. Return to Step 3.</li> </ol>
Exceptions:	<p><b>EX-1:</b> Upload interrupted by network failure.</p> <p><b>EX-2:</b> Virus scanner service unavailable → Upload blocked with error "Upload temporarily unavailable. Try later."</p>
Includes:	UC-1
Special Requirements:	Upload must complete in < 30 seconds.
Assumptions:	None
Notes and Issues:	None

Use Case ID:	<b>UC-5</b>		
Use Case Name:	<b>View Sector Trends</b>		
Created By:	Xin Wei	Last Updated By:	Xin Wei
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	The Trends section allows users to see the current trends in the market, specific to their interested job's domain.
Preconditions:	Authenticated; sector data exists in the database.
Postconditions:	System displays the data list of the selected sector.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> <li>1. User opens the "Trends" section.</li> <li>2. System lists sectors based on vacancy/growth and top skills.</li> <li>3. User selects a sector.</li> <li>4. System checks its database for the top roles and most suited skills for that sector.</li> <li>5. System displays the top roles and most suited skills for that sector.</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> If there is no data in the selected sector</p> <ol style="list-style-type: none"> <li>1. System displays a blank field with a message saying "No data found!"</li> <li>2. Return to Step 2.</li> </ol> <p><b>AF-S2:</b> Sector data partially missing (e.g. no top skills)</p>

	1. System displays available data with placeholder text for missing parts.
Exceptions:	<b>EX-1:</b> External jobs API unavailable → trends cannot load → System displays fallback message “Trends temporarily unavailable”
Includes:	UC-1
Special Requirements:	Trends data must update at least once daily.
Assumptions:	User has at least one valid sector available in Database
Notes and Issues:	Future version may support filtering trends by location/industry tags

Use Case ID:	<b>UC-6</b>		
Use Case Name:	<b>View Recommended Jobs</b>		
Created By:	Xin Wei	Last Updated By:	Xin Wei
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	Users can see suggested jobs that are ideal for them based on their requirements and skills.
Preconditions:	Authenticated; existing job postings, user has at least 1 skill.
Postconditions:	System displays ranked jobs filtered by the skills.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> <li>1. User selects "For You".</li> <li>2. System computes a match with matched/missing skills.</li> <li>3. System displays a list of jobs recommended for user.</li> <li>4. User scrolls through the list of jobs to select which one they like.</li> <li>5. User selects a job.</li> <li>6. See <b>UC-8</b>.</li> </ol>
Alternative Flows:	<b>AF-S1:</b> If no jobs are matched to the skills OR User has no skills <ol style="list-style-type: none"> <li>1. System displays a message "You have not added any skills!"</li> <li>2. System displays generic trending jobs below the message.</li> </ol>
Exceptions:	<b>EX-1:</b> Recommendation engine fails → fallback: display most popular jobs by posting frequency

	<b>EX-2:</b> Jobs API response timeout
Includes:	UC-1, UC-8
Special Requirements:	Match score must be displayed for each job
Assumptions:	Recommendation engine can run in <3 seconds for <40 skills
Notes and Issues:	In the future, may integrate ML-based personalisation beyond skill matching

Use Case ID:	<b>UC-7</b>		
Use Case Name:	<b>Search and Filter Jobs</b>		
Created By:	Xin Wei	Last Updated By:	Xin Wei
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	This section allows users to search for jobs based on keywords through filtering based on skills, salary, date posted, etc.
Preconditions:	Authenticated, existing job postings.
Postconditions:	System displays results reflecting query/filters.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> <li>1. System displays list of unfiltered jobs based on date updated.</li> <li>2. User opens the search / filter option box.</li> <li>3. User selects and enters filters or searches based on key words</li> <li>4. System searches database for jobs matching filters / key words</li> <li>5. System displays to User list of jobs based on filters / key words.</li> <li>6. User selects a job.</li> <li>7. See <b>UC-8</b>.</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> User enters invalid filter values (e.g. salary range min &gt; max)</p> <ol style="list-style-type: none"> <li>1. System prompts the user to correct input.</li> </ol>



	<b>AF-S2:</b> If there is no available jobs based on the key word / filter <ol style="list-style-type: none"> <li>1. System displays a message saying “No jobs matched! Use another keyword or select less filters”</li> <li>2. Return to Step 2.</li> </ol>
Exceptions:	<b>EX-1:</b> Search service unavailable.  <b>EX-2:</b> Slow API → system times out and shows “search unavailable, please retry”.
Includes:	UC-1, UC-8
Special Requirements:	Search results should appear within 2 seconds under normal load
Assumptions:	Job postings exist in database
Notes and Issues:	Might need to cache frequently searched queries for faster response

Use Case ID:	<b>UC-8</b>		
Use Case Name:	<b>View Job Details</b>		
Created By:	Dharshini	Last Updated By:	Dharshini
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	This section allows users to view the details of the job that has either interested them/appeals most to them based on their skills.
Preconditions:	User is authenticated; job is active.
Postconditions:	System displays the details of the job; user may save/log application.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on job.</li> <li>2. System checks the database for the job details.</li> <li>3. System displays the job details.</li> <li>4. User can apply for the job. See <b>UC-10</b>.</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> if user does not wish to apply but to view job later</p> <ol style="list-style-type: none"> <li>1. User can save jobs as a bookmark to the Saved section.</li> <li>2. See <b>UC-9</b>.</li> <li>3. User clicks on the “Saved” button at the bottom.</li> <li>4. System displays the saved jobs.</li> </ol> <p><b>AF-S2:</b> if job is removed</p> <ol style="list-style-type: none"> <li>1. System displays a message saying “Job has been removed!”</li> </ol>

	2. Returns back, depending on the previous page User was on.
Exceptions:	<b>EX-1:</b> Job details partially missing (e.g. no salary info) → show available details with “N/A” where missing.
Includes:	Extended by UC-9, UC-10, UC-11
Special Requirements:	Must show Apply button linking to external site
Assumptions:	Selected job is active in database
Notes and Issues:	details page must handle rich text job descriptions (bullet points, links)

Use Case ID:	<b>UC-9</b>		
Use Case Name:	<b>Save/Unsave Job</b>		
Created By:	Dharshini	Last Updated By:	Dharshini
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	The user has the option to save or unsave the job based on their interest.
Preconditions:	User is Authenticated; User is on an existing Job Detail page and the job is active.
Postconditions:	System stores / removes the selected job in the Saved section.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> <li>1. On the job's details page, User clicks on the bookmark button at the upper right corner.</li> <li>2. System stores the selected job in the "Saved" section..</li> <li>3. User goes to the saved jobs page.</li> <li>4. System displays saved jobs to user.</li> <li>5. User clicks on a saved job to go its detail page.</li> <li>6. User again clicks on the bookmark button.</li> <li>7. System removes the selected job in the "Saved" section.</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> if job has been removed</p> <ol style="list-style-type: none"> <li>1. System displays a message saying "Job cannot be saved, it has been removed!" when user tries to save it.</li> <li>2. Return to Step 1.</li> </ol> <p><b>AF-S2:</b> if job has already been saved</p>

	1. System changes button state to “Unsave”
Exceptions:	<b>EX-1:</b> Database save operation fails → system shows error “unable to save job. please retry.”
Includes:	Extends UC-8
Special Requirements:	Saved jobs must persist across user sessions
Assumptions:	User has at least one valid saved job entry
Notes and Issues:	Consider limiting max saved jobs (e.g. 100) to reduce DB overhead.

Use Case ID:	<b>UC-10</b>		
Use Case Name:	<b>Log/Update Application</b>		
Created By:	Sundari	Last Updated By:	Sundari
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	The user can log their application and record it in the applications section “Apps” or apply by clicking on the company’s link and they will be routed accordingly.
Preconditions:	User is Authenticated; User is on an existing Job Detail page and the job is active.
Postconditions:	Application submitted OR recorded in the “Apps” section
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on the “Apply on company site” button.</li> <li>2. System updates the job with an ‘Applied’ tag.</li> <li>3. External browser will be called to display the website.</li> <li>4. User complete the application on the website.</li> <li>5. User returns to the app and User can record the application.</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> User attempts to log duplicate job application</p> <ol style="list-style-type: none"> <li>1. System warns “Already logged. Update existing entry instead.”</li> </ol> <p><b>AF-S2:</b> User records the application</p> <ol style="list-style-type: none"> <li>1. User clicks on the “Log application” button.</li> <li>2. System stores the application in the “Apps” section.</li> </ol>

	<ol style="list-style-type: none"> <li>3. User clicks on the “Apps” button at the bottom.</li> <li>4. System displays all saved applications.</li> <li>5. User clicks on the “Edit notes” button to add the details / remarks of the application.</li> </ol>
Exceptions:	<p><b>EX-1:</b> Database write failure when logging application.</p> <p><b>EX-2:</b> External browser apply-link fails to load → system shows “external site unavailable”</p>
Includes:	Extends UC-8
Special Requirements:	Application log must timestamp entries
Assumptions:	User applies externally and only uses system for tracking
Notes and Issues:	Future versions may support syncing application status from external sites

Use Case ID:	<b>UC-11</b>		
Use Case Name:	<b>Fetch Courses for Matching Skill Gaps</b>		
Created By:	Sundari	Last Updated By:	Sundari
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	User
Description:	Users can see courses that will help them upskill and make them better suited for the job that they potentially want to apply for.
Preconditions:	User is Authenticated; User has existing skills in their profile, is on an existing Job Detail page and the job is active.
Postconditions:	Display the courses to supplement skill gaps.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on the option to view courses for skill gaps.</li> <li>2. System checks the database for matching courses to supplement skill gaps.</li> <li>3. System displays any results and corresponding links to the User.</li> <li>4. User can choose to click on the links to the courses.</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> if no courses available for skill gaps</p> <ol style="list-style-type: none"> <li>1. System displays a message saying that "There are no available courses!"</li> <li>2. Return to the Job Details page.</li> </ol> <p><b>AF-S2:</b> multiple courses overlap</p>



	1. System shows top 3 courses ranked by provider credibility/relevance.
Exceptions:	<b>EX-1:</b> Courses API unavailable → system shows “courses service temporarily unavailable”.
Includes:	Extends UC-8
Special Requirements:	Course links must open in external browser
Assumptions:	Courses API has at least some entries for common skill gaps
Notes and Issues:	Future integration with skillsfuture APIs directly for real-time funding info.

Use Case ID:	<b>UC-12</b>		
Use Case Name:	<b>Run/Recheck Ingestion and Logs</b>		
Created By:	Li Sinuo	Last Updated By:	Li Sinuo
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	AdminSystem
Description:	This is done by the admin system. This stage is predominantly to run/recheck ingestions and logs and debug errors.
Preconditions:	Admin Authenticated, Existing logs.
Postconditions:	Ingestion triggered or logs displayed.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> <li>1. Admin runs/rechecks ingestions and logs.</li> <li>2. System runs through all the logs and checks for error.</li> <li>3. If no error is found, the system displays a message saying "Everything is fine!"</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> if there is 1 or more error:</p> <ol style="list-style-type: none"> <li>1. The system displays a message, informing the admin of it and its details.</li> <li>2. Admin makes a correction.</li> <li>3. Repeat Step 1.</li> </ol>
Exceptions:	<p><b>EX-1:</b> Logs service corrupted → system shows "logs unavailable"</p> <p><b>EX-2:</b> Ingestion process interrupted mid-run</p>
Includes:	None

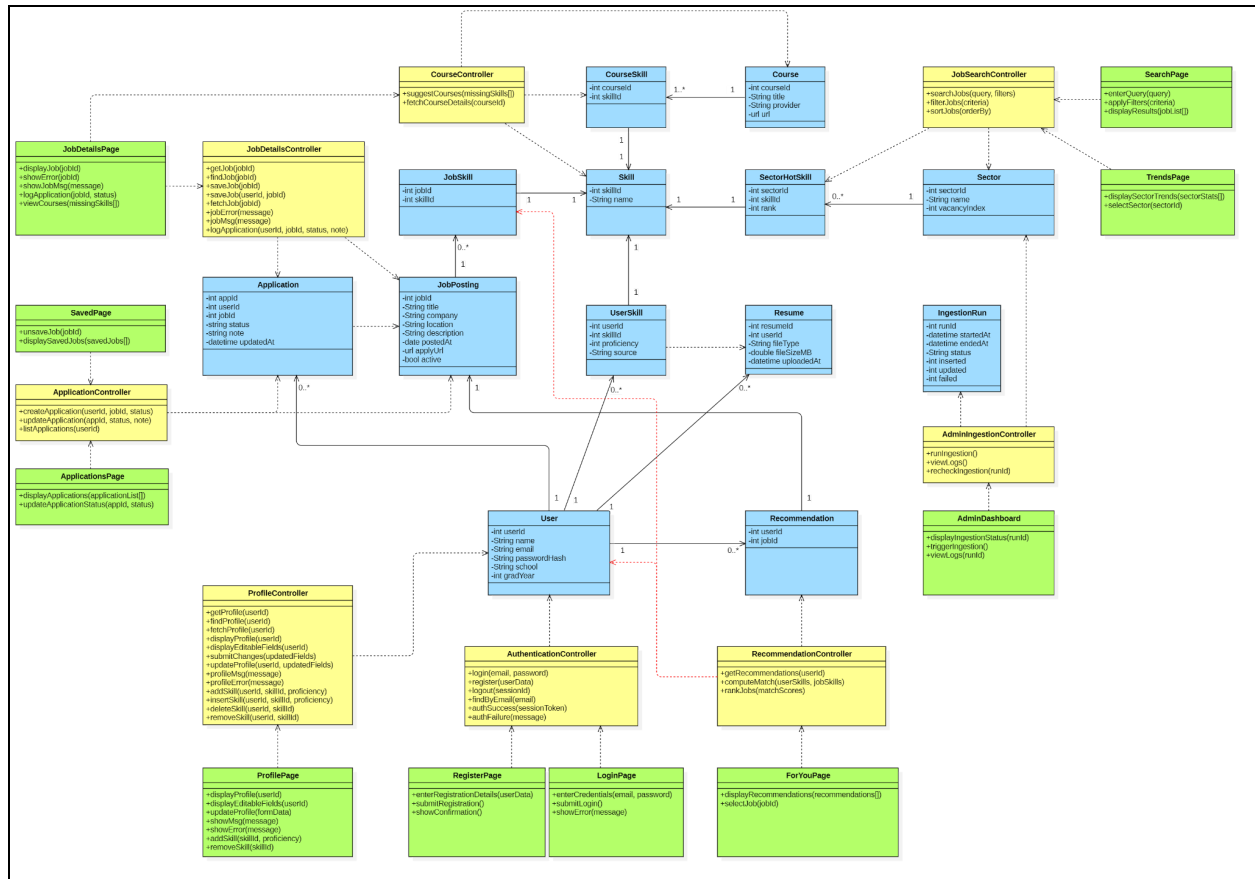
Special Requirements:	Logs must retain last 30 days of ingestion activity
Assumptions:	Admin authenticated
Notes and Issues:	Error logs may need severity categorisation (info, warning, critical)

Use Case ID:	<b>UC-13</b>		
Use Case Name:	<b>Adding New Job Postings</b>		
Created By:	Li Sinuo	Last Updated By:	Li Sinuo
Date Created:	01 - 09 - 2025	Date Last Updated:	12 - 09 - 2025

Actor:	AdminSystem
Description:	This is done by the admin system. This is to add any new job postings to the database.
Preconditions:	Admin Authenticated. Available job postings to be added.
Postconditions:	New job postings are added to the database.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> <li>1. Admin checks the job postings database.</li> <li>2. Admin adds the new job postings to the database.</li> <li>3. System confirms jobs posting details and adds to the database.</li> <li>4. System displays new job postings on all the lists to the user.</li> </ol>
Alternative Flows:	<p><b>AF-S1:</b> if new jobs postings have missing fields or error</p> <ol style="list-style-type: none"> <li>1. System displays a message indicating "Error! New job posting has missing or invalid details!"</li> <li>2. Admin fixes any mistakes on the job posting and adds it to the system again.</li> <li>3. Return to step 3.</li> </ol> <p><b>AF-S2:</b> duplicate posting detected</p> <ol style="list-style-type: none"> <li>1. System warns and blocks insert</li> </ol>
Exceptions:	<b>EX-1:</b> Database write error prevents saving posting.

Includes:	None
Special Requirements:	Job postings must include mandatory fields (title, company, location, posting date)
Assumptions:	Admin has authority to insert postings
Notes and Issues:	May require moderation for inappropriate/invalid postings.

## **Class Diagram**



(clearer image uploaded directly to GitHub)

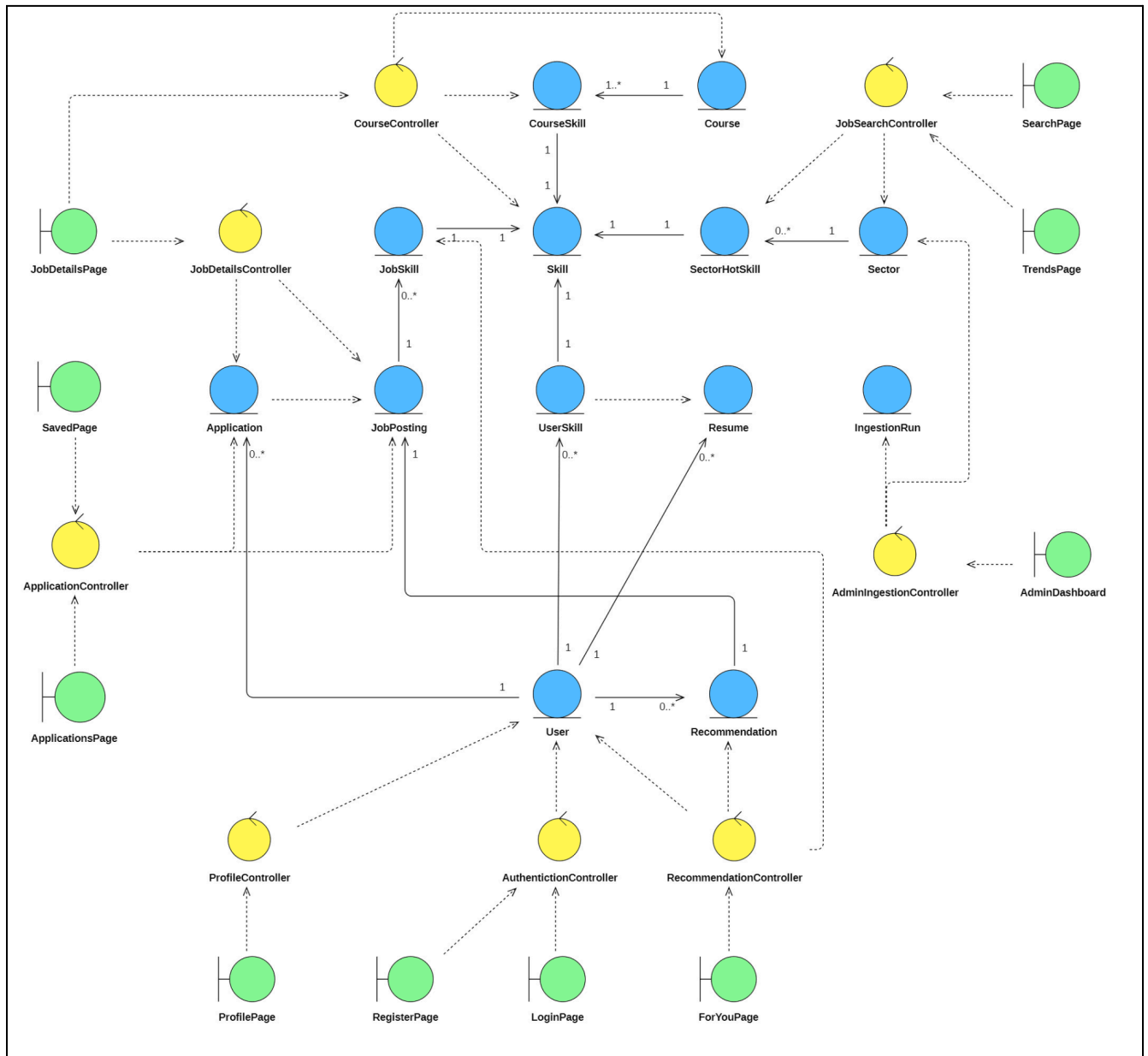
**Key:**

**Green - Boundary Classes**

**Yellow - Control Classes**

**Blue - Entity Classes**

## Relationship Between Boundary, Control and Entity Classes



(clearer image uploaded directly to GitHub)

### Key:

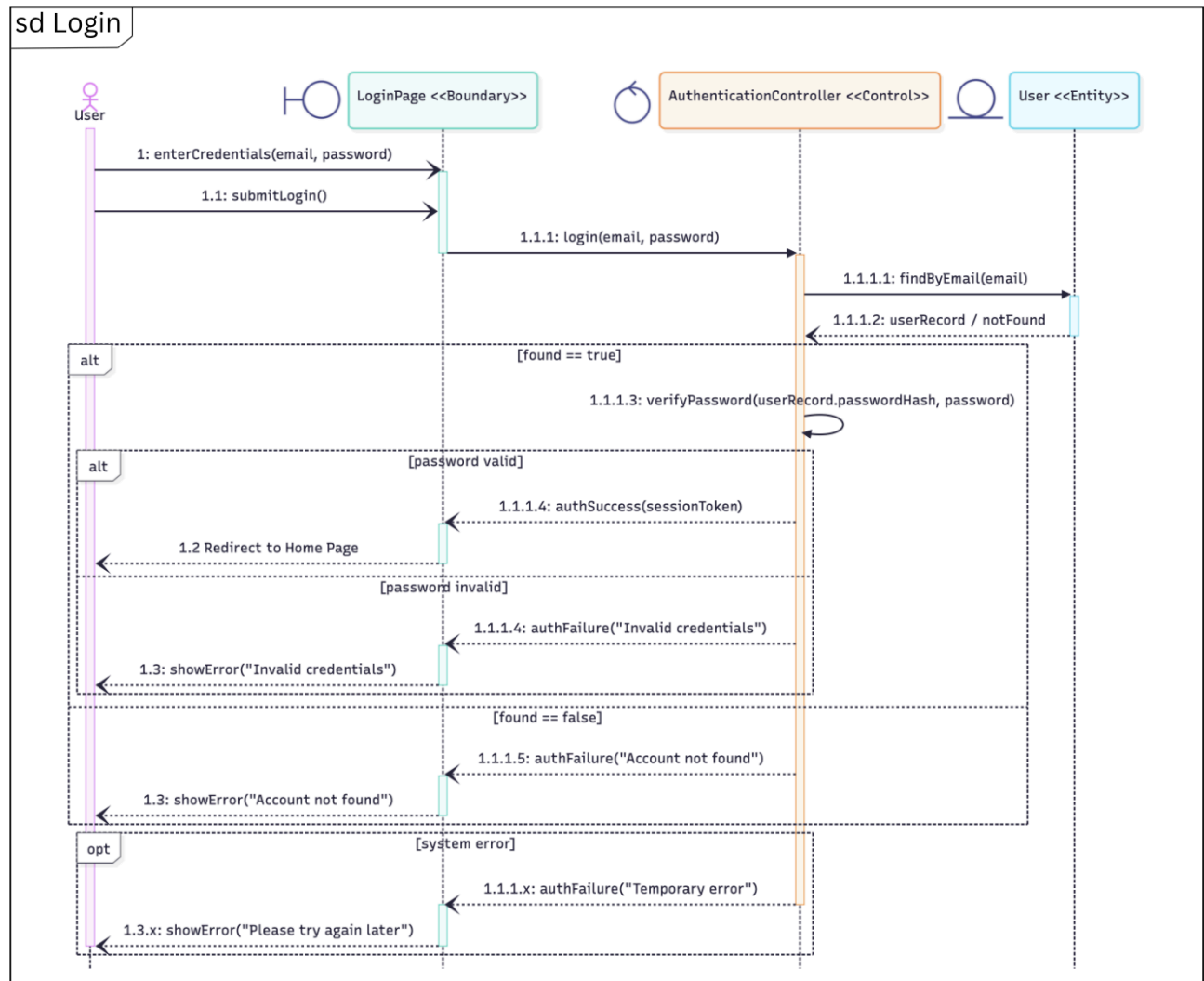
Green - Boundary Classes

Yellow - Control Classes

Blue - Entity Classes

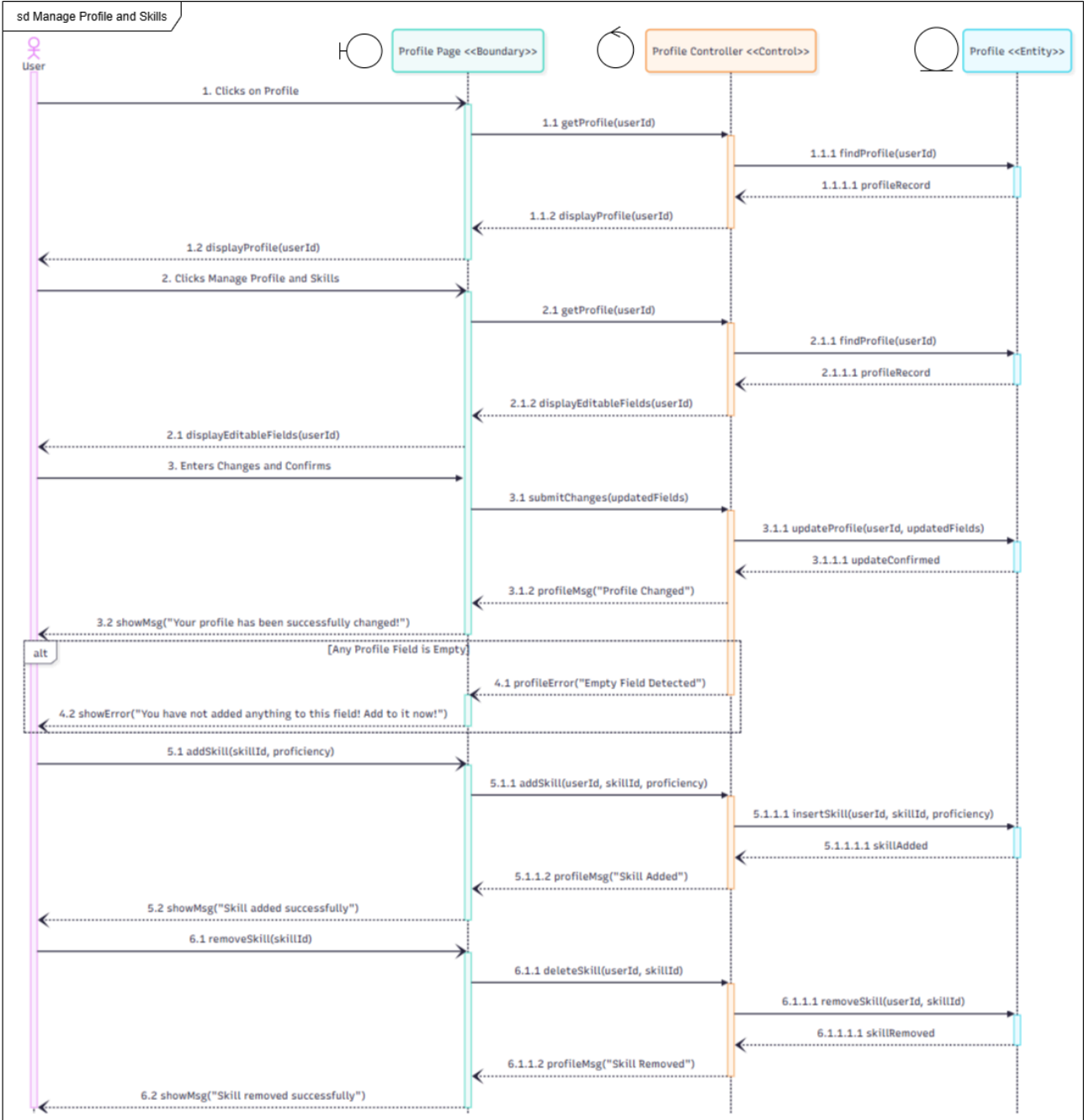
## Sequence Diagrams

## Use Case - UC-2 (Login)

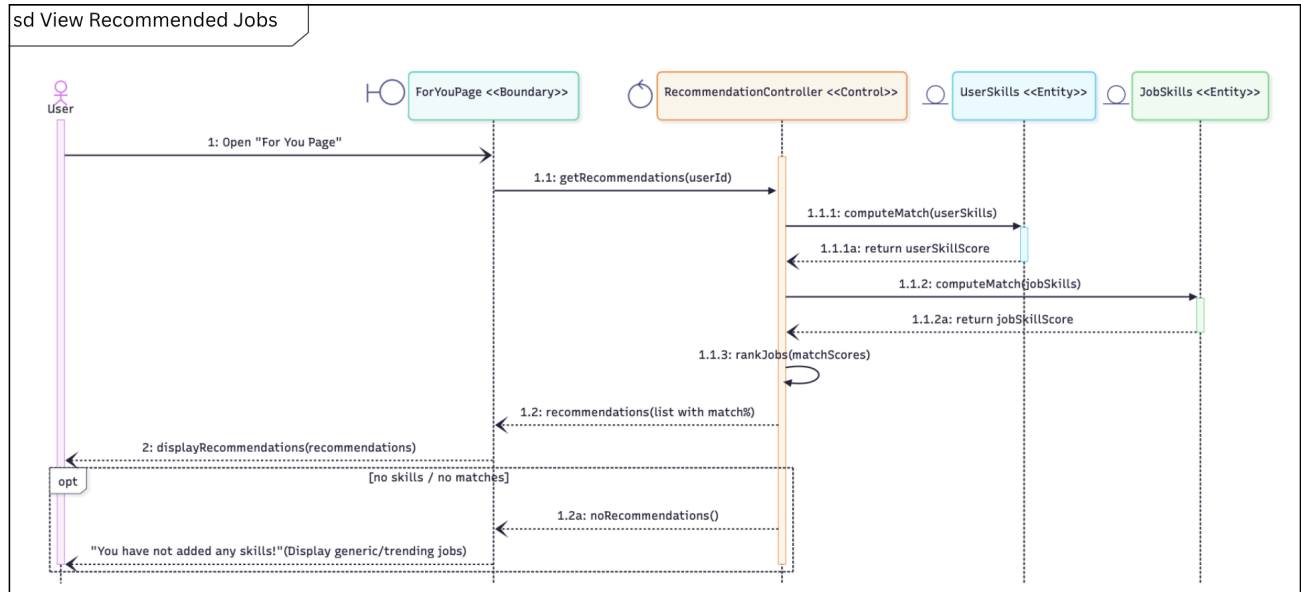


## Use Case - UC-3 (Manage Profile and Skills)

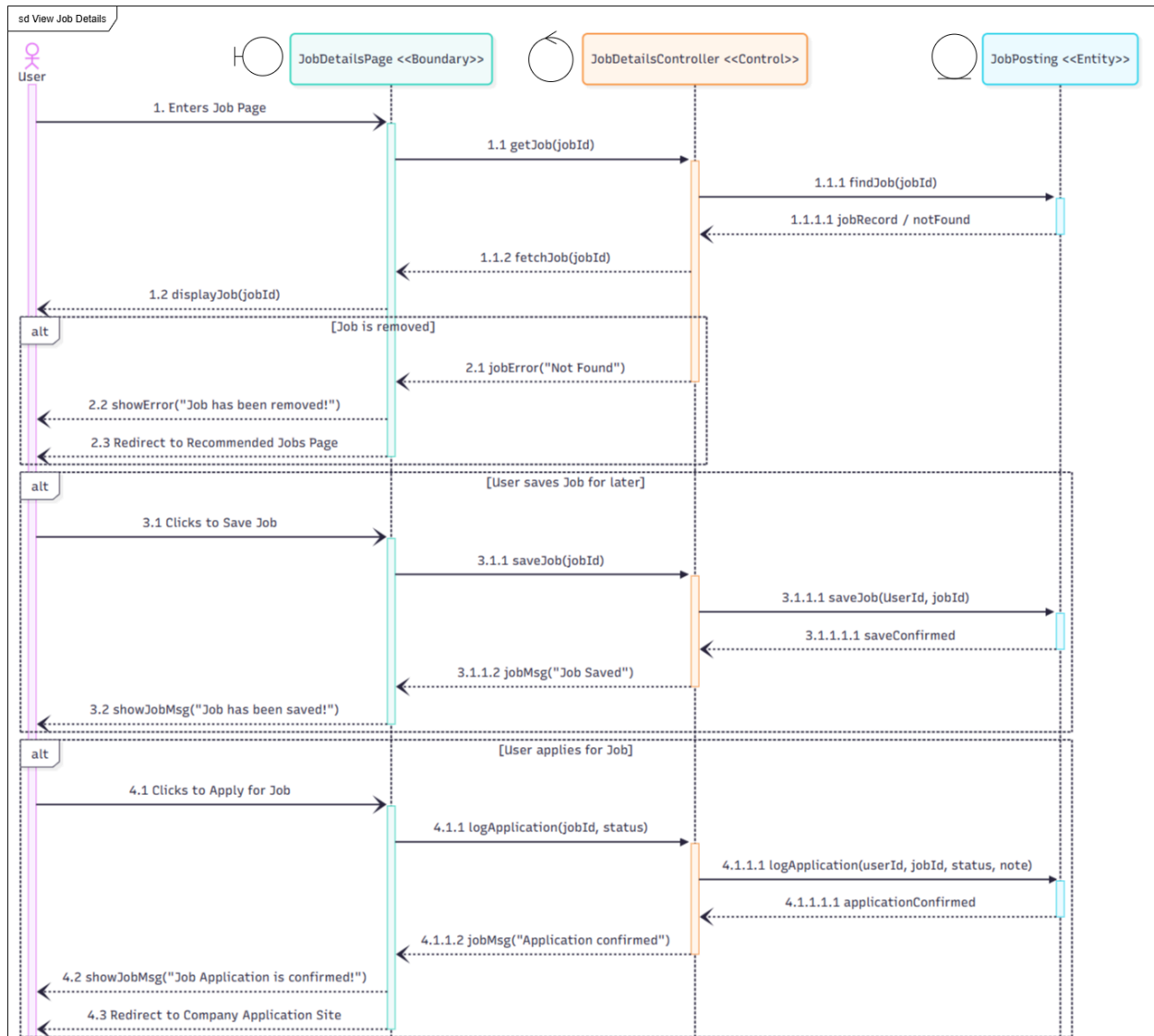




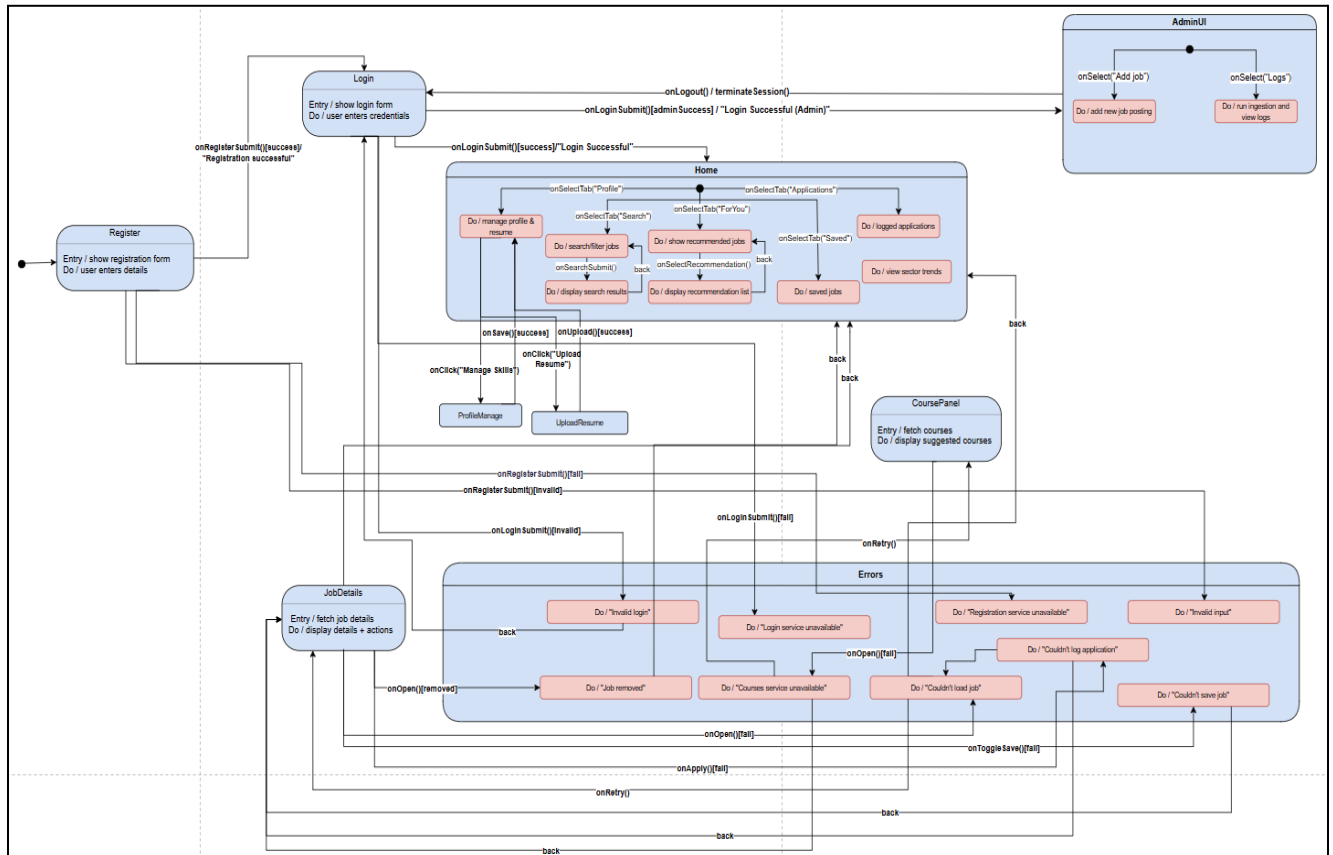
## Use Case - UC-6 (View Recommended Jobs)



## Use Case - UC-8 (View Job Details)



## Initial Dialog Map



(clearer image uploaded directly to GitHub)