

Career Path App

Authors:	Bobe Andreea Radu Denisa Pakcan Merve Stoicescu Stefan
Status	Draft 1

Summary

<i>Introduction</i>	3
Product vision	3
User classes	4
Client avatar	4
Business workflow	7
<i>System requirements</i>	7
<i>Use-cases</i>	8
<i>System overview: use-case diagram</i>	11

Introduction

Product vision

For students, recent graduates, and professionals **who** struggle to understand which career paths best align with their strengths, interests, and skills, and who feel overwhelmed by the number of options available, **The Career Path App** is an intelligent career guidance and learning recommendation platform **that** analyzes user profiles, identifies personal strengths, and provides tailored career path suggestions, learning roadmaps, and real-time job market insights to support confident and informed career decisions.

Unlike traditional career counseling, generic online quizzes, or guess-based decision-making, **our product** provides data-driven insights, personalized recommendations, real-time job-market alignment, and an interactive skill-assessment experience empowering users to confidently choose a career that fits who they are and who they want to become. So, who's ready to make career exploration simpler, smarter, and more empowering?

User classes

The users that will benefit from and use the Career Path App are the following: Students, Professionals and System Administrators.

The **Students** are the end users who are looking for a job that matches what they are currently studying for and learning about, therefore they should have access in the application to the following:

- Creating a personalised profile, writing down education, skills, experience, interests and personal details
- Assessing skills, interests, personality
- Viewing the personalised career recommendations on their profile
- Accessing and following learning paths and development plans
- Tracking progress in terms of skill acquisition and career readiness

The **Professionals** are users who want to change their career path and find a new job. In the application, they can do the following:

- Build a career profile including work experience, qualifications and skills
- Analyze transferable skills to identify compatible or emerging career options
- Update career path and skills as they evolve
- Access training resources and certification exams and courses
- View personalised career recommendations on their profile according to the jobs market

The **System Administrators** are the users with the administrative access who ensure maintenance and provide technical support in terms of user management, database security and analytics tools. They have the following options in the application:

- Manage user accounts, i.e. create, update, suspend, or delete users
- Monitor and maintain system performance and data integrity, security and user permissions
- Update or configure assessment tests, career databases and content sources
- Perform troubleshooting and software updates and patches

Client avatar

Student User

Name: Anna Lester

Age: 22

Education Level: Undergraduate student (Second-year, Software Engineering)

Location: Bucharest, Romania

Goals:

- Discover a career path and job opportunities which best match her strengths and interests
- Understand her skills and learn more about the certifications which are most valuable in the job market
- Build confidence in her career decisions
- Create a clear plan to achieve her chosen career goals

Problems and Pain Points:

- Feeling overwhelmed by the numerous number of career options
- Feeling unsure and insecure about how her current skills can apply in a real job
- Lacks access to personalized career guidance or mentoring
- Needs motivation and structure to be able to study efficiently

Background:

- Comfortable using mobile apps and online learning platforms
- Spends several hours daily on her smartphone for academic and personal purpose
- Prefers apps with visual dashboards, progress tracking, and goal-setting tools

Sources of information:

- University career centers and job fairs
- Websites LinkedIn Learning, Coursera

Professional User

Name: Matt Smith

Age: 31

Education Level: Master's degree in Business Administration

Current Occupation: Marketing Executive Specialist

Location: San Francisco, California

Goals:

- Discover new career paths that are compatible with his experience
- Get career recommendations based on the real demand of the job market
- Identify new skills and possible certifications for career conversion

Problems and Pain Points:

- Unclear career conversion process
- Lack of personalized guidance and up-to-date market insights

- Full schedule due to full-time employment, making focused planning essential

Background

- 10 years hands-on experience in the current job domain
- Prefers logical, straight-up interfaces with clear information
- Spends little time on his phone

Sources of Information:

- LinkedIn articles and professional networking groups
- Online business publications (Harvard Business Review, Forbes, Business Insider)

System Administrator User

Name: Alex Graham

Age: 29

Education Level: Bachelor's degree in Computer Science

Current Occupation: IT Support and Systems Administrator

Location: Bucharest, Romania

Goals:

- Maintain smooth operation and performance of the Career Path App
- Ensure the system's security, data protection and user privacy compliance
- Troubleshoot issues and perform updates or patches successfully
- Manage user roles and permissions

Problems and Pain Points:

- Faces challenges when coordinating between technical maintenance and user support
- Needs reliable tools in order to assure monitoring and reporting system status
- Concerned about maintaining strong data protection policies and avoiding breaches
- Fears that troubleshooting may be difficult due to inconsistent logs or lack of proper documentation
- Must balance daily maintenance tasks with long-term improvement projects

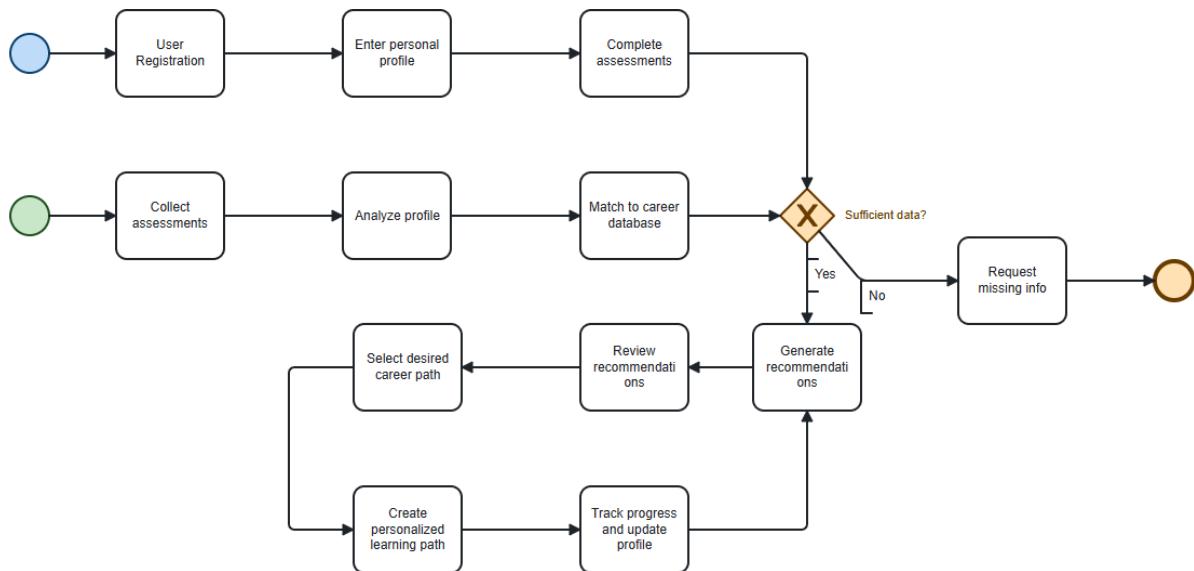
Background:

- Developed strong technical skills in system maintenance, troubleshooting and database management
- Detail-oriented and she values structure, automation and efficiency in her daily work
- Focused on ensuring the best technical performance of the application

Sources of Information

- Technical documentation and communities (Stack Overflow, GitHub, ServerFault)
- Tech webinars and online certifications (CompTIA, Cisco, AWS, Google Cloud)

Business workflow



System requirements

The **Career Path App** is a web and mobile platform that provides personalized career recommendations, learning paths, and job-market insights.

The system consists of a central backend server that manages data processing and storage, and client applications for web and mobile users that provide access to the system's functionalities.

HARDWARE REQUIRED FOR THE SERVER

For the proper operation of the Career Path App, the backend system must ensure stable and efficient performance for concurrent users performing profile analysis, recommendation generation, and data retrieval.

The following are the **minimum recommended server requirements**:

- **Operating System:** Linux (Ubuntu 22.04 LTS or equivalent)
- **Processor:** Quad-core (4 cores) 2.4 GHz or higher (Intel Xeon / AMD EPYC)
- **RAM:** Minimum 16 GB (recommended 32 GB for production environment)
- **Storage Capacity:** 512 GB SSD (expandable, depending on number of users)
- **Database:** PostgreSQL or MySQL
- **Bandwidth:** Minimum 100 Mbps for network traffic (1 Gbps recommended for scalability)
- **Backup & Security:** Daily database backup, encrypted communication using HTTPS (TLS 1.3), firewall and API access control for protection of user data.

These specifications ensure fast response times for recommendation generation and secure management of user information and analytics.

HARDWARE REQUIRED FOR END USERS

The Career Path App is designed to be accessible and lightweight, available through both web and mobile platforms. Therefore, no specialized hardware is required, any standard modern device with an internet connection is sufficient.

For Web Users:

- **Device:** Desktop or Laptop
- **Operating System:** Windows 10+, macOS 11+, or Linux
- **Browser:** Google Chrome, Mozilla Firefox, Safari, or Edge (latest versions)
- **Processor:** Dual-core 2.0 GHz or higher
- **RAM:** Minimum 4 GB (recommended 8 GB for optimal performance)
- **Internet:** Stable broadband connection (minimum 5 Mbps)

For Mobile Users:

- **Operating System:** Android 9.0+ or iOS 13+
- **Processor:** Quad-core or higher
- **RAM:** Minimum 3 GB
- **Storage:** At least 200 MB free space for installation and data caching
- **Connectivity:** Wi-Fi or 4G/5G mobile data connection
- **Optional:** Camera for CV or profile photo upload

OPTIONAL / ADMINISTRATIVE HARDWARE

For Administrators: A standard laptop or desktop with stable internet connection and access to the system's admin dashboard.

For Development & Testing: Standard laptop or desktop with internet access and code editor (e.g., Visual Studio Code).

Non-Functional Usability Requirement

The system supports features such as adjustable text size, high-contrast mode, and responsive layout for improved readability and accessibility across devices.

SOFTWARE REQUIREMENTS

The Career Path App is developed using modern web and mobile technologies.

The following software components are required for development and deployment:

- **Server:** Node.js (or Python Flask/FastAPI) backend
- **Database:** PostgreSQL or MySQL
- **Frontend:** React (web) and Flutter or React Native (mobile)
- **Hosting Environment:** AWS, Google Cloud, or Azure
- **Version Control:** GitHub for code management
- **Development Tools:** Visual Studio Code or similar IDE

Use-cases

For each use case, describe the following details:

- Name
- Actors
- Summary
- Pre-condition
- Main scenario
- Alternative scenarios
- Post-conditions
- Non-functional requirements
 - Priority: high, medium, low
 - User interface mockups

Examples:

Color coded: red – high priority, blue medium, gray low

Authentication	
Actors	Guest
Summary	This use case describes the process of authenticating a visitor to their subsequently created account using their email address

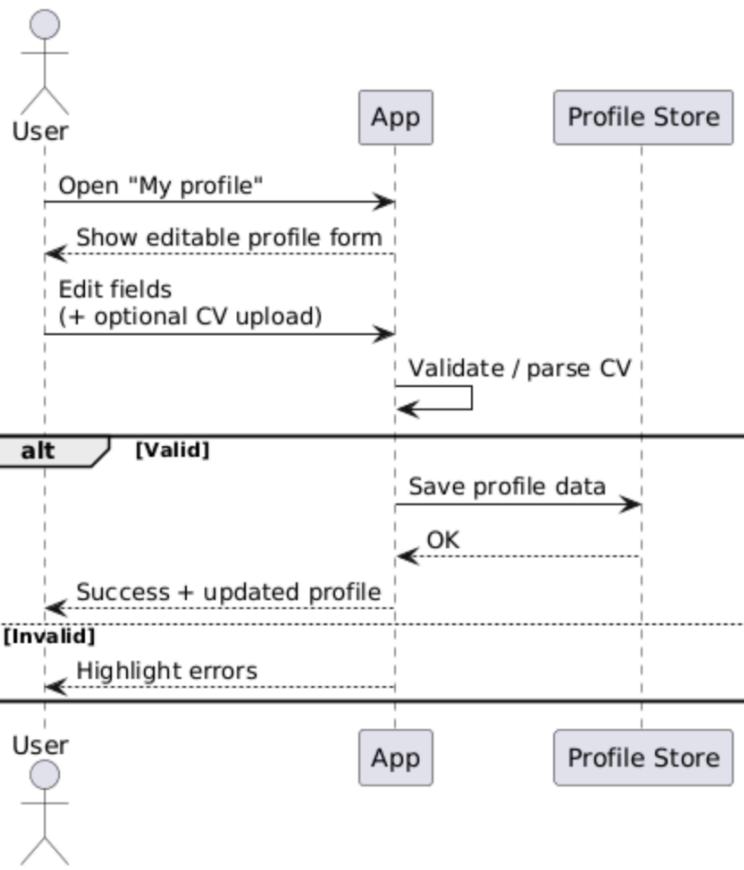
Precondition	The user must have already created an account in the application
Postcondition	The user account is displayed, with all available information
Main scenario	<ol style="list-style-type: none"> 1. Select the "Authentication" option 2. The system displays the corresponding window with the information specified in Requirement.Authentication.Authentication 3. The user fills in the email address 4. The user submits the form for validation 5. The system checks the validity of the entered data 6. The system displays the user account on success

Alternative scenarios	<ul style="list-style-type: none"> - In step 5, if the data entered is not valid, the user is notified and returns to step 3 5. a. If there is an empty field 5.B If the email address and / or password do not match those of the account 	
Sequence Diagram	<p style="text-align: center;">UC-1 – Authentication</p> <pre> sequenceDiagram participant Guest participant App Guest->>App: Open "Authentication" activate App App->>Guest: Show login form Guest->>App: Enter email + password Guest->>App: Submit activate App App->>Guest: Validate credentials alt [Valid] Guest-->>App: Show user account else [Invalid] App-->>Guest: Error message ("Return to form") end deactivate App </pre>	
Non-functional req.	<p>Priority</p> <p>UI</p>	<p>High</p> <p>The interface should be simple, minimalist, easy to use. It will contain 2 input fields (email and password), a submit button, a password reset button, as well as a button with the logo of each social authentication system.</p>
	Security	The password entered by the user will have hidden characters

Create and Update Profile

ID	UC-2
Actors	Student / Professional
Summary	This use case describes how the user can create or update their personal profile by entering details such as education, skills, experiences, and interests, and optionally upload a CV for analysis
Precondition	The user must have already created an account in the application The user must be authenticated
Postcondition	Profile information is displayed, with all available information
Main scenario	<ol style="list-style-type: none"> 1. The user selects "My profile" option from the main menu 2. The system displays the current profile form with editable fields 3. The user enters or updates information such as education, skills, experience, interests, and personal details 4. Optionally, the user can upload a CV file. 5. The user reviews and submits the form 6. The system validates the input and saves the updated profile. 7. A success message is displayed confirming profile update
Alternative scenarios	<p>In step 6, if the data entered is not valid, the user is notified and returns to step 5</p> <p>61: Missing required fields → system displays error message and highlights empty fields.</p> <p>62: Invalid CV format or parsing error → system notifies user and allows manual editing.</p> <p>63: Connection error → data temporarily stored locally until sync resumes.</p>

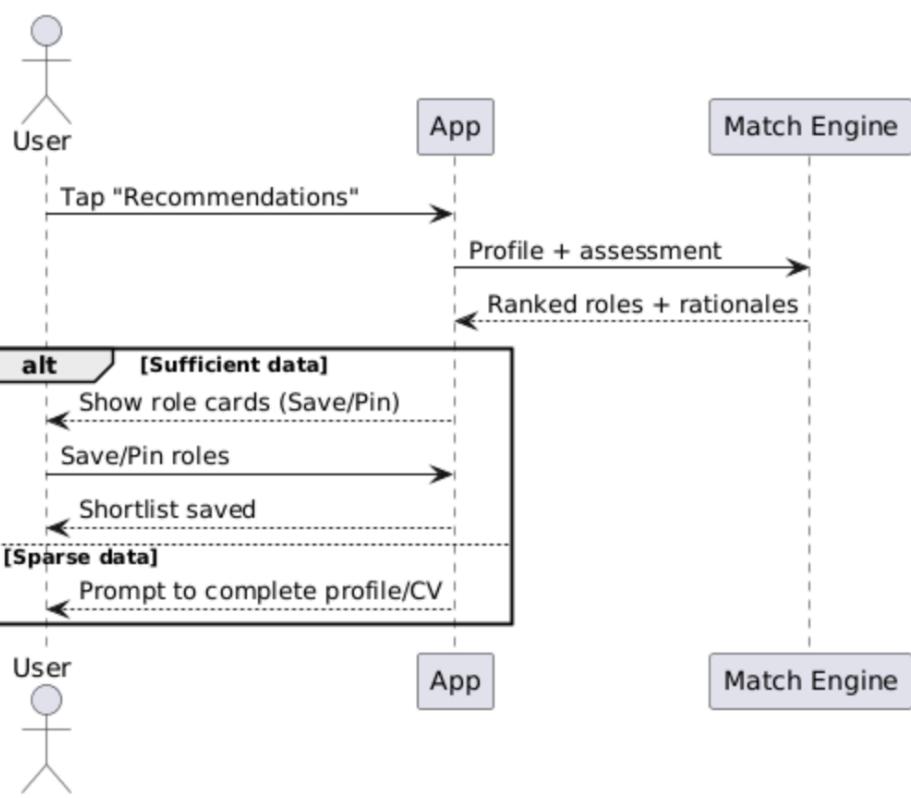
Sequence Diagram

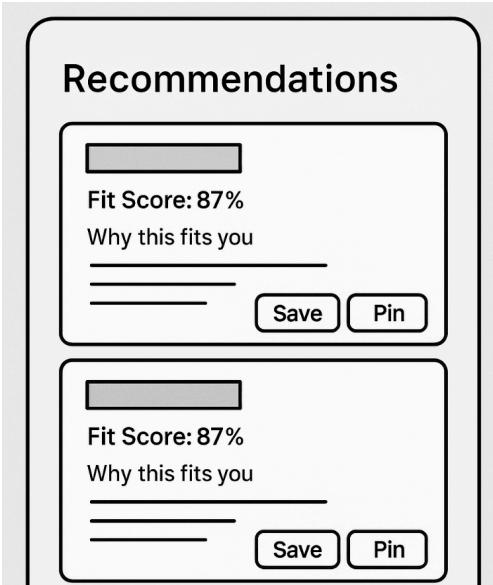
UC-2 — Create & Update Profile

Non-functional req.	Priority	High (critical for personalized recommendations))
	UI	<p>Minimalist form layout. Fields grouped by category (Education, Experience, Skills, Interests). Upload button for CV file.</p> 
	Security	<p>Personal data encrypted in storage. CV files stored securely and deleted on user request</p>

Get Career Recomendations	
ID	UC-3
Actors	Student / Professional
Summary	This use case generates a ranked list of careers aligned to the user's strengths and interests, backed by market data
Precondition	UC2 (Profile + CV)
Postcondition	Saved shortlist of roles for insights/roadmap
Main scenario	<ol style="list-style-type: none"> 1. User taps Recommendations 2. System matches profile + assessment signals to a role catalog 3. System augments ranking using job-market alignment signals. 4. System shows ranked roles with short rationales ("why this fits you") 5. User saves or pins preferred roles
Alternative scenarios	A1: Sparse data → system prompts to finish UC1/UC2 first A2: No good fit → show adjacent roles and ask preference tuning

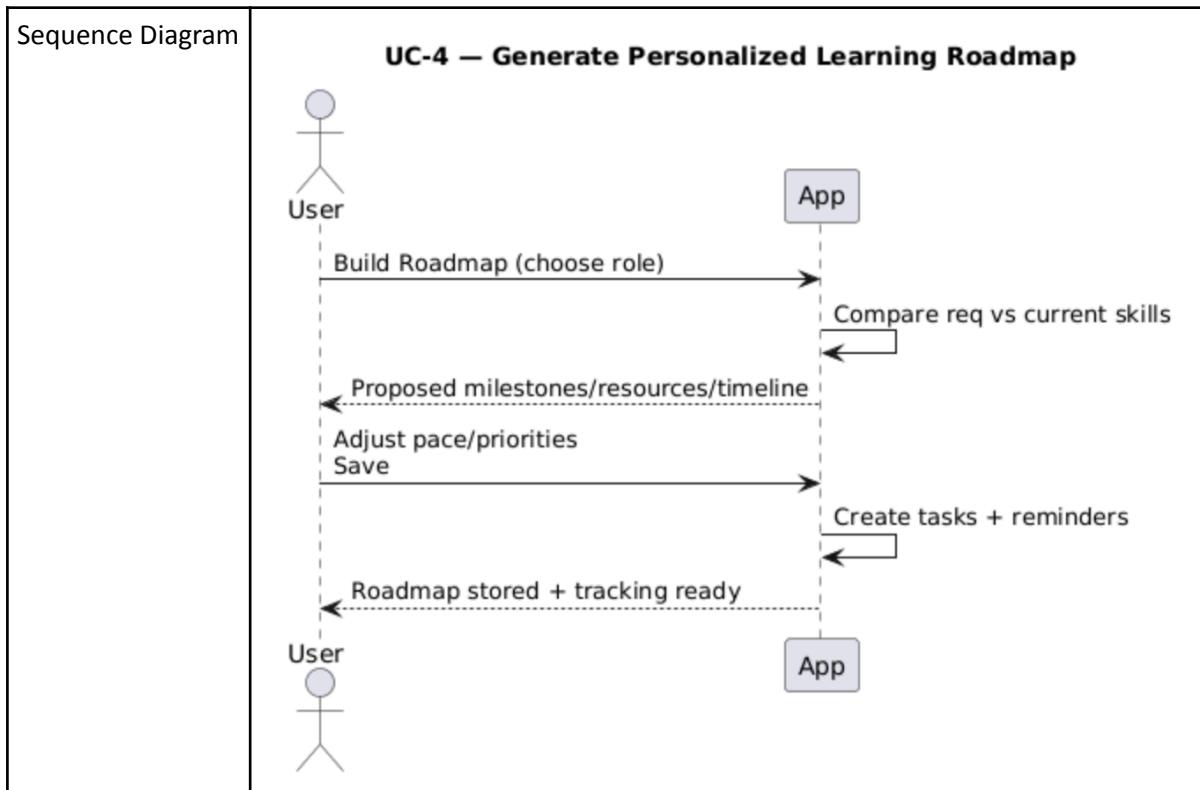
Sequence Diagram

UC-3 — Get Career Recommendations

Non-functional req.	Priority	High (critical for insights and plans)
	UI	Cards with title, fit score, rationale, quick actions (Save, View Insights) 
	Security	No external sharing by default; explicit consent for exports.

Generate Personalized Learning Roadmap

ID	UC-4
Actors	Student / Professional
Summary	This use case creates a step-by-step upskilling plan that closes skill gaps for a chosen role
Precondition	UC2 and UC3 completed
Postcondition	Roadmap stored; tasks ready for tracking
Main scenario	<ol style="list-style-type: none"> User selects Build Roadmap for a target role. System compares required vs. current skills to identify gaps. System proposes milestones, learning resources, and a timeline. PROJECT User adjusts pace/priorities and saves roadmap. System schedules tasks and reminders.
Alternative scenarios	A1: Multiple targets → allow parallel tracks with priority tags A2: User rejects resources → suggest alternatives



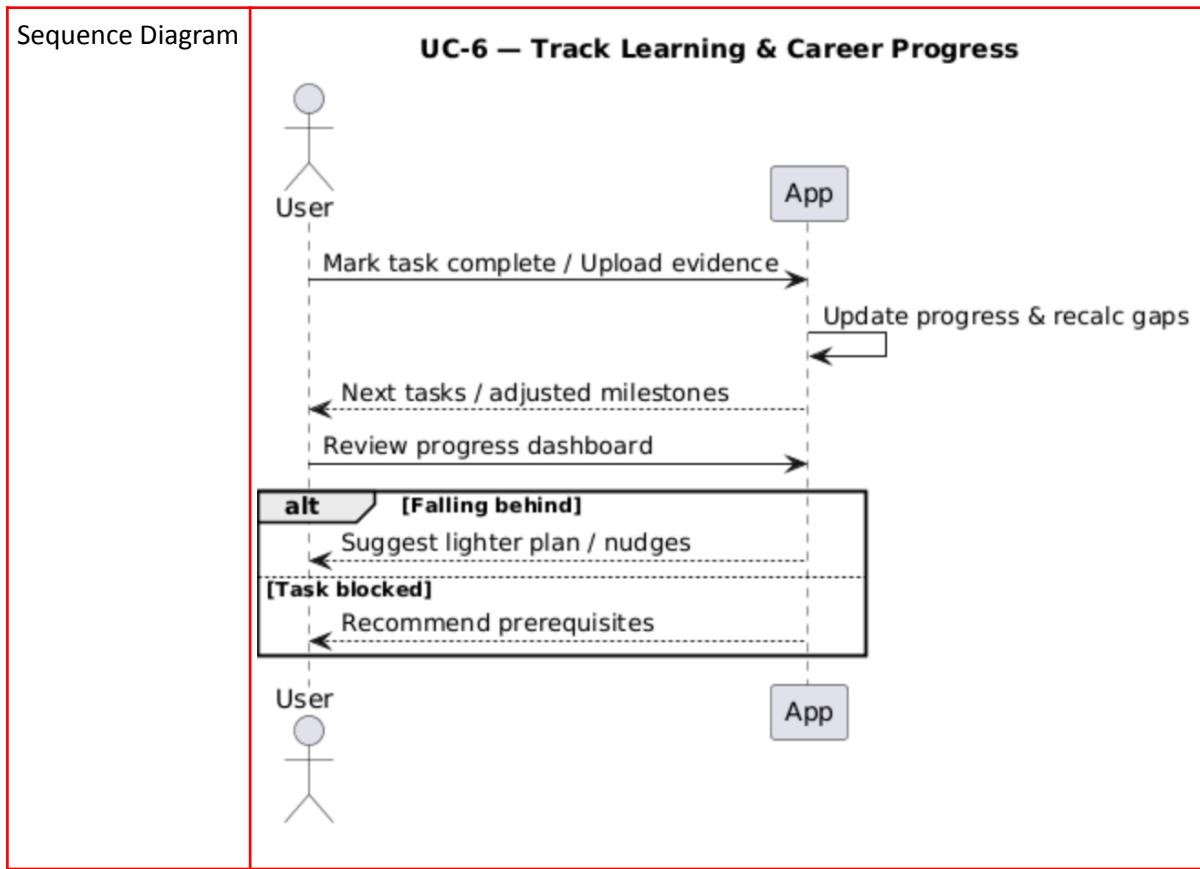
	Priority	High (core value: efficient, personalized upskilling)												
Non-functional req.	UI	<p>Milestone timeline, task list, progress ring, “Reschedule” quick actions)</p> <p>Personalized Learning Roadmap</p> <p>The screenshot displays the 'Personalized Learning Roadmap' interface. At the top, there is a horizontal timeline with four circular milestones labeled 'Milestone 1', 'Milestone 2', 'Milestone 3', and 'Milestone 4'. Below the timeline, there are two main sections: a 'Task List' and a 'Progress Ring'. The 'Task List' section contains a table with four rows, each representing a task due on a specific date in May. The 'Progress Ring' section features a large donut-style progress indicator with two 'Reschedule' buttons below it.</p> <table border="1"> <thead> <tr> <th></th> <th>Task 1</th> <th>Due May 1</th> </tr> </thead> <tbody> <tr> <td>Task 2</td> <td></td> <td>Due May 8</td> </tr> <tr> <td>Task 3</td> <td></td> <td>Due May 15</td> </tr> <tr> <td>Task 4</td> <td></td> <td>Due May 22</td> </tr> </tbody> </table>		Task 1	Due May 1	Task 2		Due May 8	Task 3		Due May 15	Task 4		Due May 22
	Task 1	Due May 1												
Task 2		Due May 8												
Task 3		Due May 15												
Task 4		Due May 22												
	Security	Resource links examined; opt-in calendar sync.												

Request and Schedule mentorship / Career Planning

ID	UC-5
Actors	Student / Professional / Mentor
Summary	This use case connects a user with a suitable mentor for career guidance
Precondition	UC3 completed (target role chosen)
Postcondition	Session scheduled and logged; action items added to roadmap.
Main scenario	<ol style="list-style-type: none"> 1. User opens Mentorship. 2. System recommends mentors based on role/skills focus. 3. User requests a session; proposes times. 4. Mentor accepts; system books and sends confirmations. 5. System creates calendar invites and sends reminders and reschedule links. 6. After session, mentor adds notes
Alternative scenarios	<p>A1: No mentor available → waitlist or async Q&A.</p> <p>A2: Timezone conflict → show alternatives.</p> <p>A3: Meeting type changed (online/in-person) → update invites.</p>
Sequence Diagram	<p style="text-align: center;">UC-5 – Request & Schedule Mentorship</p> <pre> sequenceDiagram actor User actor Mentor actor App actor CalendarService User->>App: Open "Mentorship" activate App App-->>User: Recommended mentor list User->>App: Request session + propose times activate App App-->>User: Session request alt [Mentor accepts] User->>App: Confirm time activate App App-->>User: Confirmation User->>App: Add session notes activate App App-->>User: Notes available + action items end User->>CalendarService: Create invites + reminders deactivate App CalendarService-->>User: Confirmation User->>Mentor: Notes available + action items </pre> <p>The sequence diagram illustrates the interaction between the User, Mentor, App, and Calendar Service. It begins with the User opening the 'Mentorship' feature in the App, which returns a 'Recommended mentor list'. The User then proposes a session and times. An alternative scenario for 'Mentor accepts' involves the User confirming the proposed time, receiving a confirmation, adding session notes, and finally getting notes available and action items. The App then creates invites and reminders, which are confirmed by the Calendar Service, and the User is notified of the availability and action items.</p>

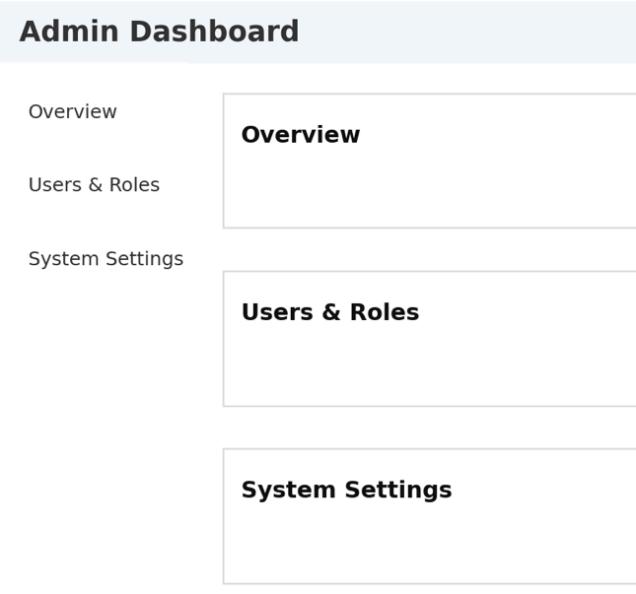
	Priority	High																																				
Non-functional req.	UI	<p>Mentor cards (bio, expertise), notes section, calendar picker, timezone chip, notification settings.</p> <div style="text-align: center; margin-top: 10px;"> <input type="button" value="Request Session"/> </div> <p>Recommended Mentors</p> <div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="border: 1px solid #ccc; padding: 5px; width: 150px; height: 100px;">  </div> <div> Mentor Name Expertise </div> <div style="border: 1px solid #ccc; padding: 5px; width: 150px; height: 100px;">  </div> <div> Mentor Name Expertise </div> </div> <p>Select a Date & Time</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <table border="1" style="margin-right: 20px; border-collapse: collapse; text-align: center;"> <tr><td>S</td><td>M</td><td>T</td><td>W</td><td>T</td><td>F</td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td> </td><td> </td></tr> <tr><td>5</td><td>6</td><td>8</td><td>9</td><td>10</td><td>11</td></tr> <tr><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>18</td></tr> <tr><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>25</td></tr> </table> <div style="display: grid; grid-template-columns: 1fr 1fr; gap: 10px;"> <div style="border: 1px solid #ccc; padding: 5px; text-align: center;">9:00 AM</div> <div style="border: 1px solid #ccc; padding: 5px; text-align: center;">10:00 AM</div> <div style="border: 1px solid #ccc; padding: 5px; text-align: center;">1:00 PM</div> <div style="border: 1px solid #ccc; padding: 5px; text-align: center;">3:00 PM</div> <div style="border: 1px solid #ccc; padding: 5px; text-align: center;">3:00 PM</div> </div> </div> <p>Notes</p> <div style="border: 1px solid #ccc; padding: 5px; height: 40px;"></div>	S	M	T	W	T	F							1	2	3	4			5	6	8	9	10	11	12	13	14	15	16	18	19	20	21	22	23	25
S	M	T	W	T	F																																	
1	2	3	4																																			
5	6	8	9	10	11																																	
12	13	14	15	16	18																																	
19	20	21	22	23	25																																	
	Security	Mentor access limited to session context; audit trail of access.																																				

Track Learning & Career Progress	
ID	UC-6
Actors	Student / Professional
Summary	This use case tracks completion of roadmap tasks and updates recommendations accordingly.
Precondition	UC4 exists
Postcondition	Progress stored; refreshed recommendations for UC3.
Main scenario	<ol style="list-style-type: none"> 1. User marks tasks complete or uploads artifacts (e.g., project links). 2. System updates progress and recalculates gaps. 3. System proposes next tasks or adjusts milestones automatically. 4. User reviews a progress dashboard.
Alternative scenarios	A1: Falling behind → system suggests a lighter plan and nudges. A2: Task blocked → recommend prerequisite micro-tasks.



Non-functional req.	Priority	High (drives outcomes; efficient & empowering) Progress bars, streaks, “Add evidence” button.
	UI	<h3>Progress</h3> <p>20-day streak</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Complete online course Algorithmic Thinking <input type="checkbox"/> Submit project link exansample.com
	Security	User controls visibility of artifacts; private by default.

Manage Application, Mentors	
ID	UC-7
Actors	Admin
Summary	This use case manages core application configuration (users, roles, permissions, content), maintains mentor accounts, configures data integrations, and monitors system health and usage analytics.
Precondition	Admin authenticated with elevated permissions.
Postcondition	Application configuration is updated and saved. Mentors are correctly onboarded and active.
Main scenario	<ol style="list-style-type: none"> 1. Admin creates, edits, or disables users and assigns permissions 2. Admin edits career role templates, skill taxonomies, and learning roadmap templates. 3. Admin onboards mentors, verifies identity/expertise, and manages availability. 4. Admin reviews access logs and handles user data requests (export/delete). 5. Admin saves and publishes the configuration changes.
Alternative scenarios	<p>A1: Invitation email fails → System prompts Admin to resend or generate manual activation link.</p> <p>A2: Admin attempts an action outside their role scope → System displays “Permission Denied”.</p> <p>A3: Mentor profile verification fails → System flags profile as pending review.</p>
Sequence Diagram	<p style="text-align: center;">UC-7 — Manage Application & Mentors</p> <pre> sequenceDiagram participant Admin participant AdminConsole as Admin Console participant App Admin->>AdminConsole: Open Admin Dashboard AdminConsole->>Admin: Users/Roles/Mentors/Config Admin->>AdminConsole: Create/Edit users, roles, mentors Edit templates & taxonomies AdminConsole->>App: Apply configuration changes activate App alt [Authorized] App-->>Admin: OK Admin->>App: Preview changes App->>Admin: Publish Admin->>App: Published successfully else [Permission denied] App-->>Admin: Error: Permission Denied end deactivate App Admin->>AdminConsole: </pre> <p>The sequence diagram illustrates the main scenario and alternative scenarios for UC-7. It shows the flow from Admin interacting with Admin Console to making changes and applying them to the App. It includes a decision point for authorization, leading to either a successful workflow or a permission denied error. The diagram uses standard UML notation with participants, lifelines, messages, and conditions.</p>

	Priority	Medium (operational).
Non-functional req.	UI	<p>Modular Admin Dashboard with searchable tables, bulk actions, “Preview Changes”, “Publish”, and version history.</p>  <p>The screenshot shows a modular Admin Dashboard. At the top is a header bar with the title "Admin Dashboard". Below it are three main sections: "Overview", "Users & Roles", and "System Settings". Each section has a sub-header and some placeholder text.</p>
	Security	Role-based access control

System overview: use-case diagram

