

Team 37 - UniFreelancer Academy

CS Capstone - Requirements Update

Date: 02/01/26

Team Members: Aidan Caughey, Aiden McCoy, Baron Baker, Daniel Molina, Nafizur Rahman

Project Partner: Alina Padilla-Miller - UniFreelancer

0. Change Log

2/1/26 - v1.1 - Requirements & Risks - Nafizur

Changes:

- Added tracking to requirements
- Reworked the risks section
- Added new risks

Driver:

- Learnings from implementing
- New requirements format presented to us

1. Problem Statement and Scope

UniFreelancer Academy is a learning and development platform designed for freelancers to upskill through Courses, Seminars, Tutorials, and Podcasts. The platform will support both free and paid tiers, reward completion with badges, and integrate achievements into the UniFreelancer marketplace to enhance professional visibility.

Primary Users: Freelance learners seeking skill development.

Secondary Users: Instructors, guest speakers, and platform partners who create content on the platform.

Context of Use:

- Web-based platform accessible globally.
- Desktop and mobile web support.
- Integration with Sharetribe marketplace for profile visibility.

System Boundaries:

- Include learning modules, user progress tracking, analytics, subscription system, live streaming and video hosting capabilities, and basic AI grading components.
- Excludes any marketplace payment logic (handled by Sharetribe).

Success Criteria:

- Over 70% course completion rate for paid users within 90 days.

- Over 50% return rate for free users.

Explicit Non-Goal: Replicating a full Canvas LMS. We will implement a lightweight, custom learning system in React.

2. Requirements

ID	Derived From	Requirement	AC	Priority	Completed?
REQ-001	A1, D2, B2, E1	The system shall provide a unified Academy homepage linking to Courses, Tutorials, Seminars, and Podcasts.	<ul style="list-style-type: none"> • Homepage contains four primary links. • Each link routes correctly to its respective section. • Navigation verified on desktop and mobile. 	MUST	YES
REQ-003	A3, D3, B8	The platform shall be fully responsive and usable on both desktop and mobile web devices.	<ul style="list-style-type: none"> • Layout adapts to ≥ 3 breakpoints (mobile, tablet, desktop). • All buttons and text remain legible and functional. 	MUST	
REQ-004	A4, E3	Within each course, a sidebar shall display module navigation and indicate progress.	<ul style="list-style-type: none"> • Sidebar lists all modules. • Clicking a module navigates without full reload. 	MUST	

			<ul style="list-style-type: none"> • Current module is visually highlighted. 		
REQ-005	D10, C8	The system shall implement a unified search and filter function across all content types (Courses, Tutorials, Seminars, Podcasts).	<ul style="list-style-type: none"> • Search returns relevant results within 2 s. • Filters (type, tier, difficulty) work independently and in combination. 	MUST	YES
REQ-006	D7, E4, C1, C9	The system shall track user progress, manage tiered access (free/paid), and issue badges or certificates upon completion.	<ul style="list-style-type: none"> • Progress saved server-side. • Paid lessons blocked for free users. • Badge/certificate auto-issued and visible on profile after 100% completion. 	MUST	
REQ-007	D6, E5	Each course or tutorial shall include downloadable learning materials.	<ul style="list-style-type: none"> • Each module displays “Download Materials.” • Downloads succeed and match 	MUST	

			correct file type.		
REQ-008	C2, B3, B4, D8	The platform shall support live streaming and video hosting for Courses, Seminars, and Podcasts.	<ul style="list-style-type: none"> • Live sessions hosted via integrated provider (e.g., Zoom/Agora). • Recordings stored and playable within 24 h. 	MUST	
REQ-009	C1, B10, E10	The system shall implement secure authentication and user accounts with persistent data storage.	<ul style="list-style-type: none"> • Users can sign up/in/out and reset passwords. • Auth uses encrypted tokens. <ul style="list-style-type: none"> • Data persists across sessions. 	MUST	YES
REQ-010	D1, B9	The system shall provide a secure subscription and payment integration via Sharetribe and Stripe (no internal payment logic).	<ul style="list-style-type: none"> • Payment handled by Stripe Checkout. • Successful payment updates Sharetribe entitlement. • No card data stored locally. 	MUST	YES

REQ-014	A6, C6	Each course shall display a progress bar and support subtitles/transcripts for accessibility.	<ul style="list-style-type: none"> • Progress bar updates in real time. • Closed captions auto-generated and editable. 	MUST	
REQ-015	A7	The system shall maintain consistent typography and spacing across all pages.	<ul style="list-style-type: none"> • Style guide defines font, size, and spacing tokens. • All pages pass visual QA against style guide. 	MUST	YES
REQ-019	B7	The system shall use a centralized database (cloud or managed service) to store users, progress, and achievements.	<ul style="list-style-type: none"> • DB schema includes user, course, progress, badge tables. • CRUD ops validated via API tests. 	MUST	
REQ-020	C10, C6, B8	The platform shall meet basic NFRs: accessibility (WCAG 2.2 AA), performance (< 2 s load), security (OWASP Top 10), and	<ul style="list-style-type: none"> • Lighthouse accessibility $\geq 90\%$. • Median TTI < 2 s. • Weekly uptime report $\geq 99.9\%$. 	MUST	

		uptime \geq 99.9%.			
REQ-002	A2, B5	The system shall include a consistent global navigation bar accessible on all pages.	—	SHOULD	
REQ-011	E2	The system shall include Course Catalog and Course Detail pages listing essential metadata (title, description, instructor, skills, price).	—	SHOULD	YES
REQ-012	C3, B6	The system shall provide analytics dashboards for learners and instructors showing progress, completion, and engagement.	—	SHOULD	
REQ-013	C5, D4, E8	The system shall recommend related or next-step	—	SHOULD	

		courses/tutorials based on user activity and skills.			
REQ-016	A8	The system shall auto-save learner progress every 10 seconds of activity.	—	SHOULD	
REQ-017	C7	The system shall support per-lesson discussion threads with instructor moderation.	—	COULD	
REQ-018	C10	The system shall implement adaptive onboarding checklists and periodic reminder notifications	—	COULD	

3. Product and Stakeholder Risks

Risk / Constraint	Affected Requirement (s)	Impact	Likelihood	Mitigation / Policy
Decoupled from the main Sharetribe code	REQ-009, REQ-010, REQ-020, REQ-006	Low	High	Start with an independent build based on the structure of

				the current sharetribe build. Maintain strong contact with the Sharetribe team.
Need for a constantly updating database since new courses/seminars are added often	REQ-001, REQ-005, REQ-019, REQ-012, REQ-013	High	High	Use an automated database accompanied by an online service like MongoDB. Create automated checks to ensure data isn't being misplaced. Confirm token refresh and error handling early.
Potential separate databases for payment and user information	REQ-006, REQ-009, REQ-010, REQ-019, REQ-020	Low	Medium	Strict boundaries between databases. Do not store payment data locally.
AWS and the process of hosting videos on it	REQ-008, REQ-009, REQ-010, REQ-019, REQ-012, REQ-013	Medium	Medium	Use proven compatible tools like AWS S3 and CloudFront. Compress media before upload.
Payment confusion	REQ-010, REQ-006, REQ-009,	High	Medium	Keep payment logic external and simple

	REQ-019			
Analytics perceived as surveillance	REQ-012, REQ-020, REQ-009	Medium	Low	Keep analytics scoped to learning value (progress/completion/engagement), avoid sensitive inference, and provide transparency on what is tracked.