

# Project Proposal

## Project Title

(To be finalized)

---

### 1. Problem Statement

Self-learners and students increasingly rely on AI tools (e.g., ChatGPT) to learn skills. However, current solutions suffer from three major issues:

1. **Lack of structure** – AI gives answers but not a disciplined learning path.
2. **No progression control** – Users jump topics without mastering prerequisites.
3. **Generic learning experience** – Same roadmap for everyone, regardless of background, pace, or goals.

Traditional platforms (e.g., Udemy, Coursera) provide structured courses but fail to adapt dynamically to individual learners. Conversely, AI tools adapt well but lack academic discipline and measurable progression.

This creates a gap between *personalization* and *structure*.

---

### 2. Proposed Solution

We propose an **AI-guided learning platform** where:

- The **system controls structure, rules, and progression**
- AI acts as an **intelligent assistant**, not a free chat bot
- Learning is **personalized per user**, but **disciplined per system**

The platform combines: - Predefined learning frameworks - User-specific AI-generated content - Controlled feedback and progression mechanisms

---

### 3. Target Users

- University students
  - Self-learners
  - Early-career professionals
  - Learners frustrated with unstructured AI learning
-

## 4. Core System Design

### 4.1 Canonical Skills

A *canonical skill* refers to a standardized, system-recognized skill category (e.g., “Frontend Web Development”, “JavaScript Fundamentals”).

User-entered skills are: 1. Analyzed by AI 2. Mapped to the nearest canonical skill 3. Passed through feasibility and scope checks

This ensures: - Academic evaluation consistency - Controlled roadmap generation

---

### 4.2 Roadmap Skeleton (System-Controlled)

Each skill follows a predefined roadmap structure stored as JSON:

- Levels
- Modules
- Checkpoints
- Progression rules
- Task types

AI is **not allowed** to change this structure.

---

## 5. Role of AI (Bounded Intelligence)

AI is used only within **strict boundaries**:

### AI Can:

- Generate explanations inside predefined lesson slots
- Adapt examples to user level
- Provide feedback on task outcomes
- Reframe explanations when user is stuck

### AI Cannot:

- Add or remove lessons
- Skip checkpoints
- Advance the roadmap
- Act as an unrestricted chat tutor

AI behaves as a **mentor**, while the system remains the **authority**.

---

## 6. User Flow

1. User signs up and creates a profile
  2. User provides learning preferences and background
  3. User defines a skill (guided input)
  4. System maps skill to a canonical category
  5. Roadmap is generated using system skeleton + AI content
  6. User completes tasks step-by-step
  7. Progress is tracked and gated
  8. AI provides feedback based on performance
- 

## 7. Task & Feedback Model

### Task Types:

- Practice tasks (Done / Not Done)
- Concept checks (MCQs / short answers)
- Creation tasks (text-based description)
- Code tasks (only for programming skills)

### Submission Logic:

- User submits task result, not free-form explanations
  - If task fails:
    - Step remains locked
    - AI re-explains the same concept
  - Partial success creates backlog reinforcement
- 

## 8. Data Handling

- User-specific roadmaps stored in database
  - AI-generated content tied to user account
  - Progress persists across sessions
- 

## 9. Limitations

- Not a full AI chat assistant
  - Limited skills in demo version
  - API dependency for AI features
-

## **10. Distinction from Existing Solutions**

- Unlike Udemy: personalized per learner
  - Unlike ChatGPT: structured and gated
  - Unlike institutions: adaptive and self-paced
- 

## **11. Future Scope**

- Additional skills
  - Richer task submissions (files, links)
  - Enhanced analytics
  - Mobile application
- 

## **12. Academic & Industry Value**

- Demonstrates applied AI integration
- Shows system design discipline
- Scalable to real-world SaaS
- Strong portfolio and job-market relevance