Rasale Shreya

Darshan RK

Objectives:

- Seamless Authentication – Users can sign in effortlessly with Google.
- AI-Powered Course Creation – Automatically generates structured courses with descriptions, chapters, and code examples.
- Enhanced Learning with Videos – Fetches relevant video content for each course chapter.
- Full Course Customization – Users can edit course details, descriptions, and images before publishing.
- Scalable & Efficient – Designed for smooth performance and future expansion.

Technologies/Tools Used:

1. Empathize – Understanding User Needs

- Clerk → Seamless authentication for users with Google Sign-In and session management
- Google API → Enables intuitive third-party integrations for a better user experience
- YouTube API → Allows embedding relevant video content to enhance learning

2. Define – Identifying Challenges

- Firebase → Secure and scalable file storage for course images and multimedia
- PostgreSQL → Ensures structured data storage and efficient query performance for large-scale course databases

3. Ideate – Exploring Solutions

- Drizzle ORM → Simplifies database interactions, improving development efficiency
- Google Gemini API → AI-driven content generation for personalized and structured course creation

4. Prototype – Implementing & Testing

- Real-time authentication using Clerk for a smooth user login experience
- Course content customization powered by AI
- Efficient database queries optimized through Drizzle ORM and PostgreSQL

5. Test – Continuous Improvement

- User feedback collection to refine AI-generated content and authentication flow

Agile Methodology:

1.Planning & Requirement Analysis (Sprint 1-2)

- Define core functionalities: AI-powered course creation, user authentication, course customization, and content management
- Prioritize features in the product backlog
- Select technology stack: Next.js, TailwindCSS, Drizzle ORM, Firebase, Clerk, Gemini API
- Define Agile roadmap and sprint goals

2. Feasibility Study & UX Research (Sprint 3-4)

- Conduct technical feasibility analysis for AI content generation and database performance
- Apply Design Thinking principles:
    - Empathize: Understand user pain points through surveys and interviews
    - Define: Identify course creation challenges
    - Ideate: Brainstorm UI/UX solutions
    - Prototype: Develop wireframes and mockups
    - Test: Gather feedback and refine designs
- Plan database architecture and API structure

3. Iterative Development & Sprint Execution (Sprint 5-10)

Sprint 5-6: Authentication & User Management

- Implement Google Sign-In using Clerk
- Develop middleware for session management and security
- Set up role-based access control (RBAC)

Sprint 7-8: AI-Powered Course Generation

- Integrate Gemini API for AI-driven content generation
- Design AI prompts for structured and high-quality course content
- Implement API routes for real-time course generation

Sprint 9-10: UI/UX & Course Management

- Develop a responsive and modular UI using TailwindCSS and ShadCN UI
- Implement real-time content preview functionality
- Enable course customization and integrate Firebase storage for images

Sprint 11: Performance Optimization & Security

- Optimize API response times and database queries
- Implement advanced error handling and logging

4. Testing & Continuous Integration (Ongoing in Each Sprint)

- Perform unit testing for individual components and API endpoints
- Conduct integration testing to validate AI-authentication-database interactions
- Optimize database queries and AI response times through performance testing
- Ensure data security through authentication and SQL injection testing
- Gather user feedback for continuous UI/UX improvements

5. Deployment & Continuous Improvement (Sprint 12 & beyond)

- Deploy on Vercel for high availability and scalability
- Implement CI/CD pipelines with GitHub Actions for automated updates
- Improve AI-generated content accuracy based on user feedback
- Introduce monetization features

Technical Implementation of Gemini for Content Generation

In this project, the Gemini API was used for content generation by integrating Google's gemini model. The process began with authentication using genai.configure(api_key="YOUR_GEMINI_API_KEY"), ensuring secure access. When the generate_content(prompt) function was called, the given prompt was tokenized, analyzed, and processed by the model, leveraging deep learning techniques based on Google's proprietary transformer architecture. The model generated a response in natural language, which was returned in a structured format, typically as JSON. The generated content was extracted via response.text, making it accessible for further processing or display.

The prompt itself was provided as a direct argument within generate_content(), defining the type and scope of content required. The model interpreted the input contextually, leveraging extensive pre-training data to generate coherent and relevant outputs. If tracking of past prompts was needed, logs or API request records could be referenced. This structured approach ensured a seamless and efficient content generation workflow.

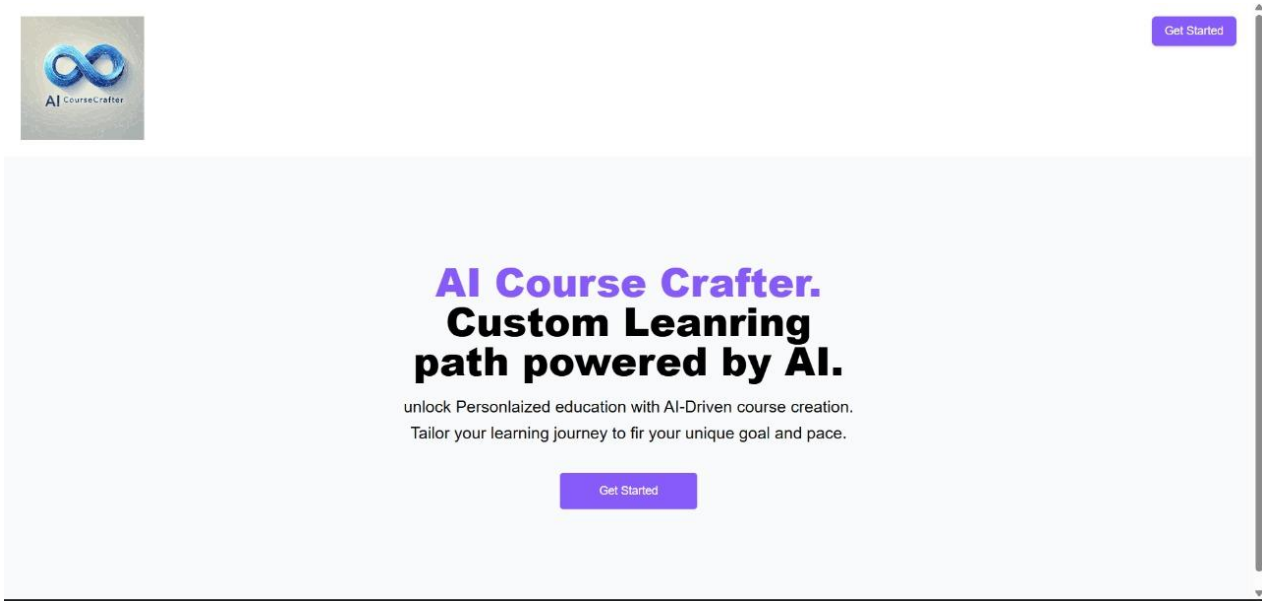
Prototype Analysis:

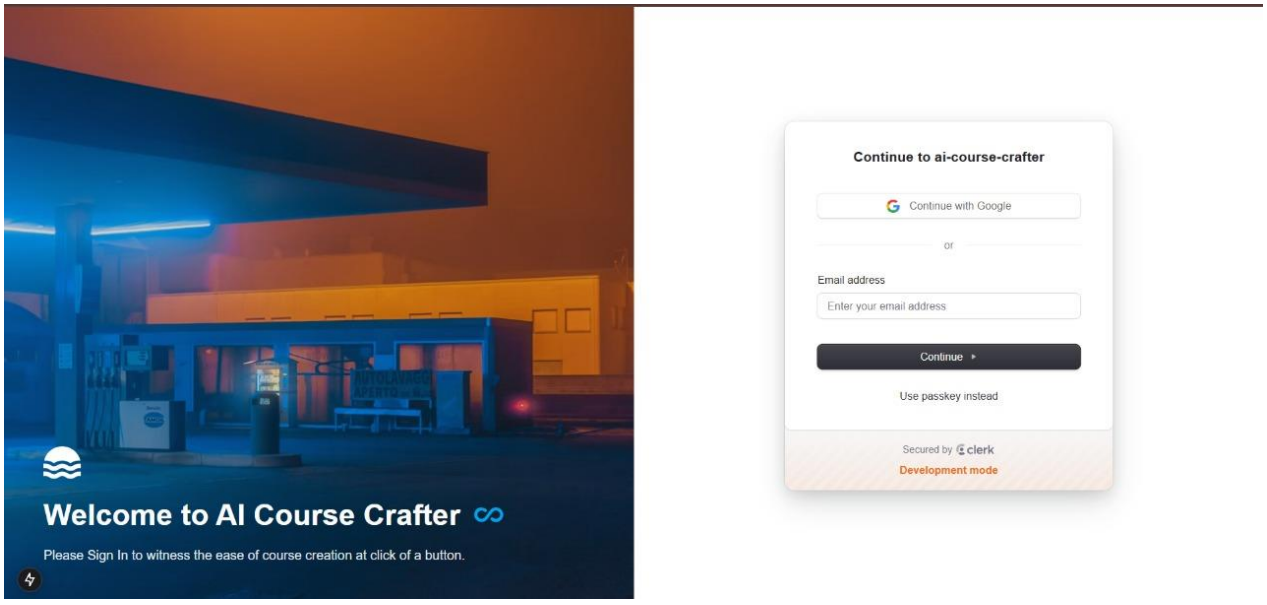
Initial vs. Final Prototype

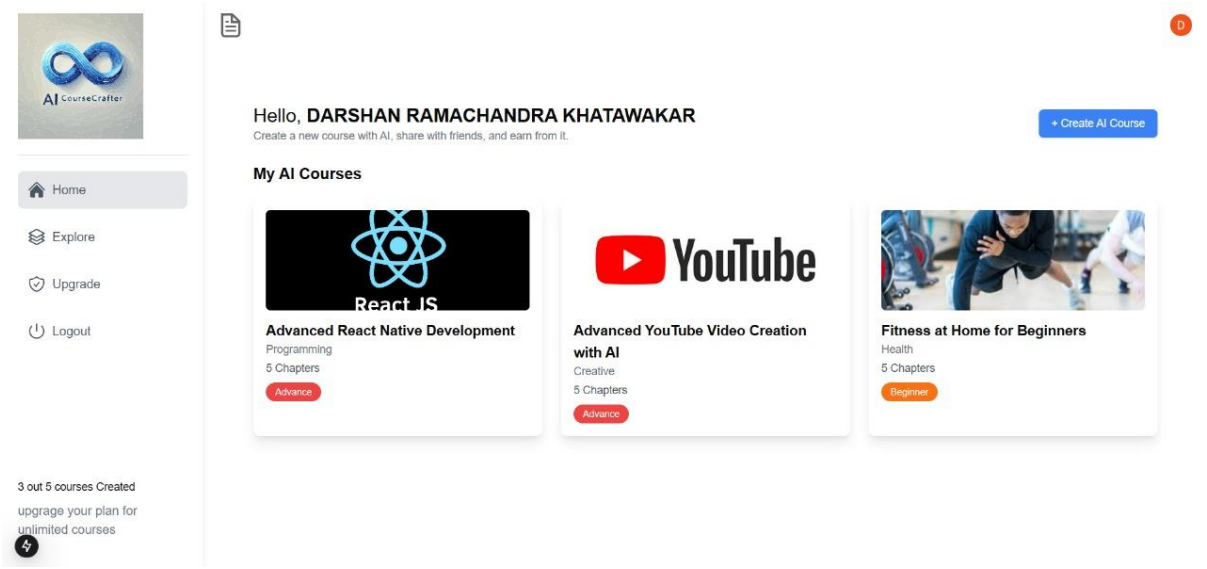
Feature	Initial Prototype	Final Prototype
UI Design	Basic layout with minimal styling	Refined UI with enhanced visuals and consistent design
Navigation	Simple navigation, fewer steps	Improved step-by-step navigation with better UX flow
Course Creation	Basic category selection	Detailed course customization with topics and description
Course Details	Limited customization (only category selection)	More options including difficulty, duration, video integration, and chapters
Course Generation	Basic structure, limited topics	AI-powered structured course generation with predefined modules

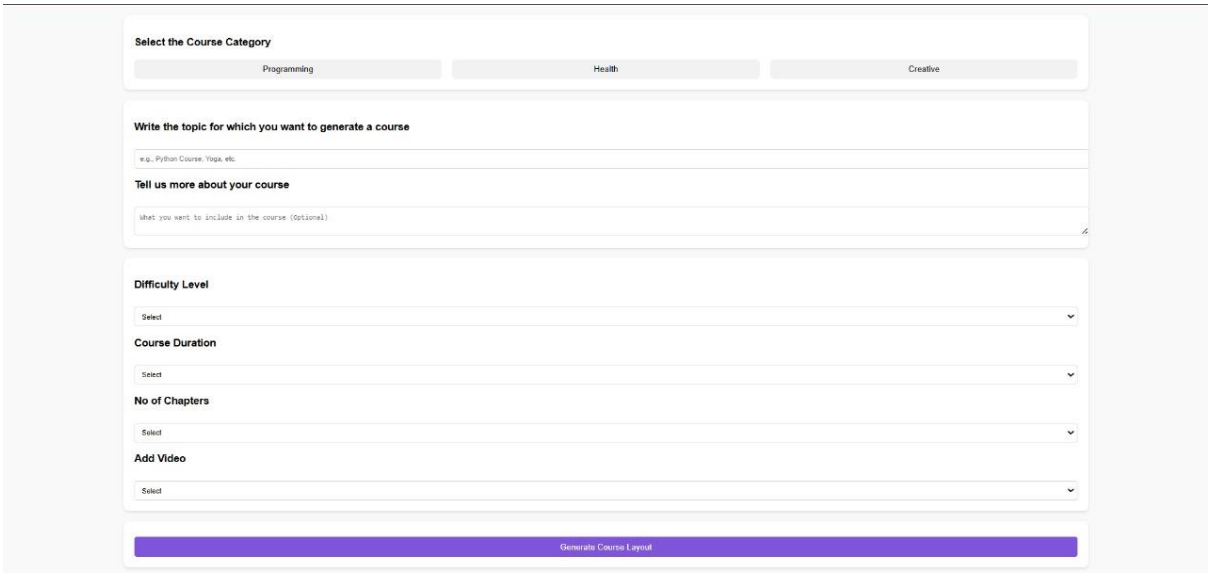
Feature	Initial Prototype	Final Prototype
Final Course View	Plain text output	Fully formatted course layout with edit options
Subscription Model	Not present	Introduced Pro and Starter plans with different features
User Experience	Functional but basic UI/UX	More intuitive with enhanced interaction and design

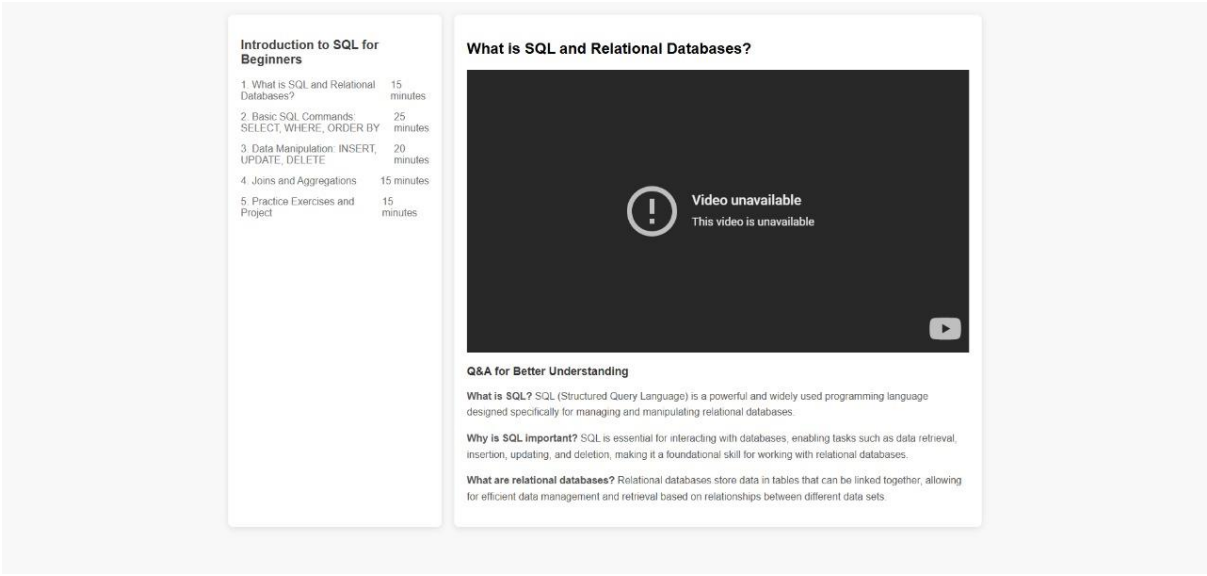
Initial Prototype

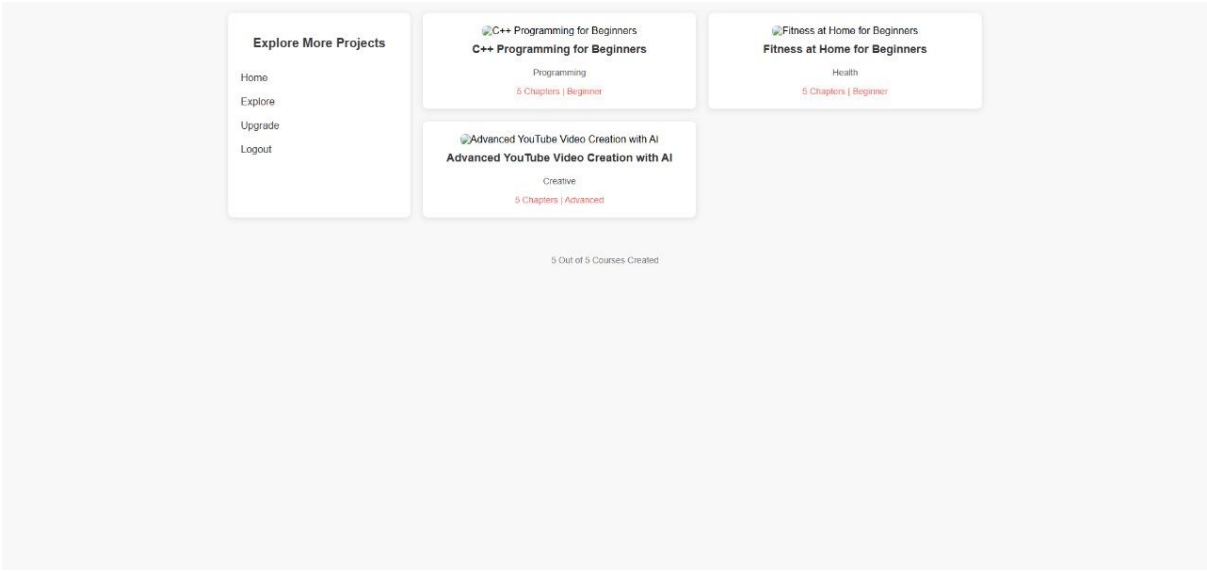


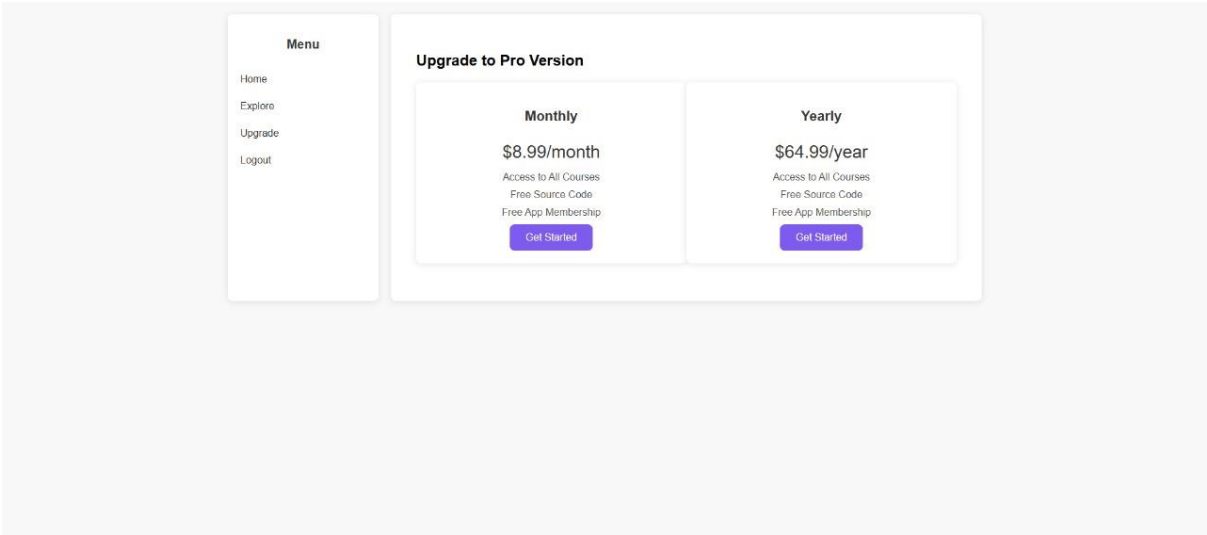



AI CourseCrafter
Home
Explore
Upgrade
Logout
3 out 5 courses Created
upgrage your plan for unlimited courses
Hello, DARSHAN RAMACHANDRA KHATAWAKAR
Create a new course with AI, share with friends, and earn from it.
+ Create AI Course
My AI Courses
React JS
Advanced React Native Development
Programming
5 Chapters
Advance
YouTube
Advanced YouTube Video Creation with AI
Creative
5 Chapters
Advance
Fitness at Home for Beginners
Health
5 Chapters
Beginner

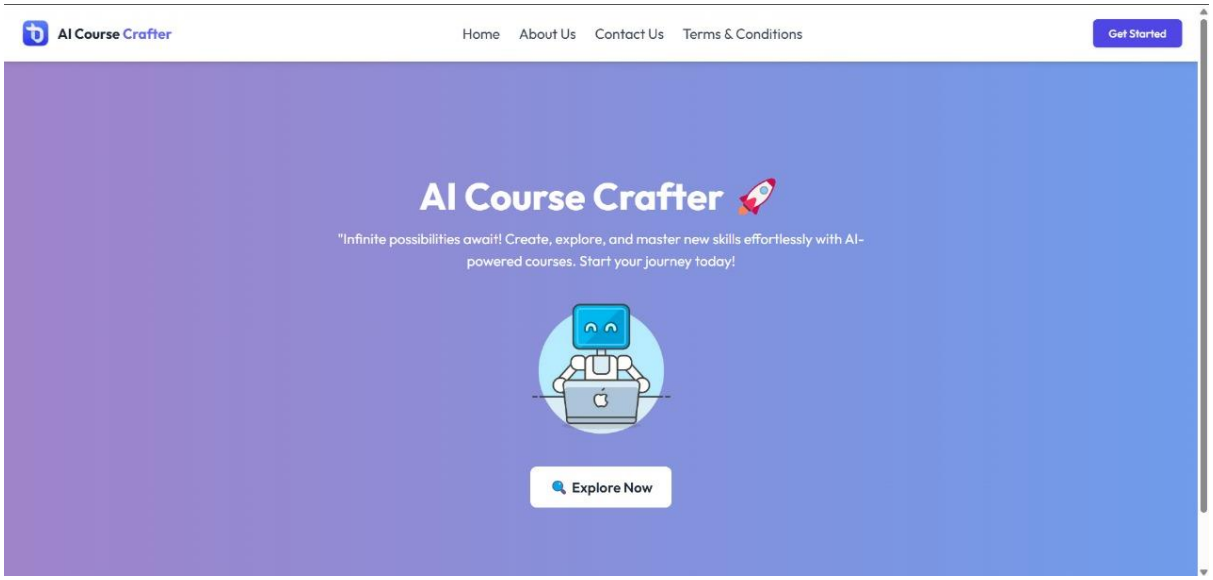

Select the Course Category
Programming
Health
Creative
Write the topic for which you want to generate a course
e.g., Python Course, Yoga, etc.
Tell us more about your course
What you want to include in the course (Optional)
Difficulty Level
Select
Course Duration
Select
No of Chapters
Select
Add Video
Select
Generate Course Layout

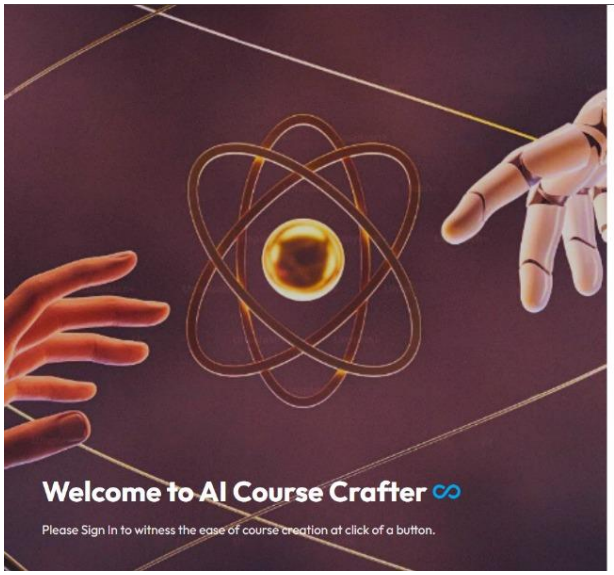

Introduction to SQL for Beginners
1. What is SQL and Relational Databases? 15 minutes
2. Basic SQL Commands: SELECT, WHERE, ORDER BY 25 minutes
3. Data Manipulation: INSERT, UPDATE, DELETE 20 minutes
4. Joins and Aggregations 15 minutes
5. Practice Exercises and Project 15 minutes
What is SQL and Relational Databases?
Video unavailable
This video is unavailable
Q&A for Better Understanding
What is SQL? SQL (Structured Query Language) is a powerful and widely used programming language designed specifically for managing and manipulating relational databases.
Why is SQL important? SQL is essential for interacting with databases, enabling tasks such as data retrieval, insertion, updating, and deletion, making it a foundational skill for working with relational databases.
What are relational databases? Relational databases store data in tables that can be linked together, allowing for efficient data management and retrieval based on relationships between different data sets.

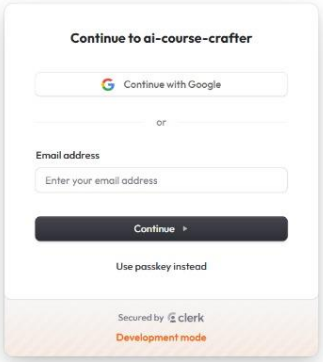

Explore More Projects
Home
Explore
Upgrade
Logout
C++ Programming for Beginners
C++ Programming for Beginners
Programming
5 Chapters | Beginner
Fitness at Home for Beginners
Fitness at Home for Beginners
Health
5 Chapters | Beginner
Advanced YouTube Video Creation with AI
Advanced YouTube Video Creation with AI
Creative
5 Chapters | Advanced
5 Out of 5 Courses Created

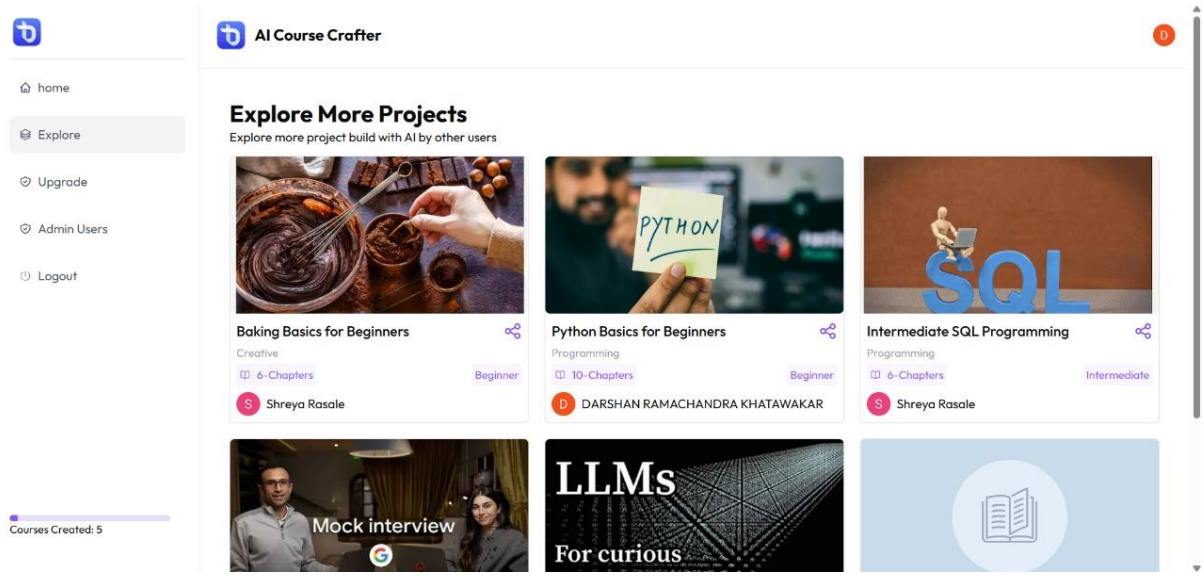

Menu
Home
Explore
Upgrade
Logout
Upgrade to Pro Version
Monthly
$8.99/month
Access to All Courses
Free Source Code
Free App Membership
Get Started
Yearly
$64.99/year
Access to All Courses
Free Source Code
Free App Membership
Get Started

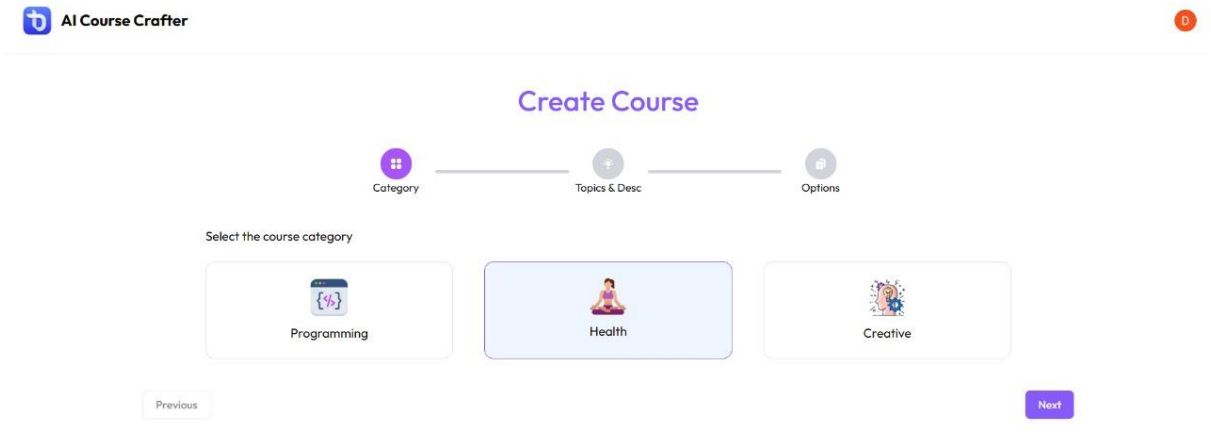
Final Prototype

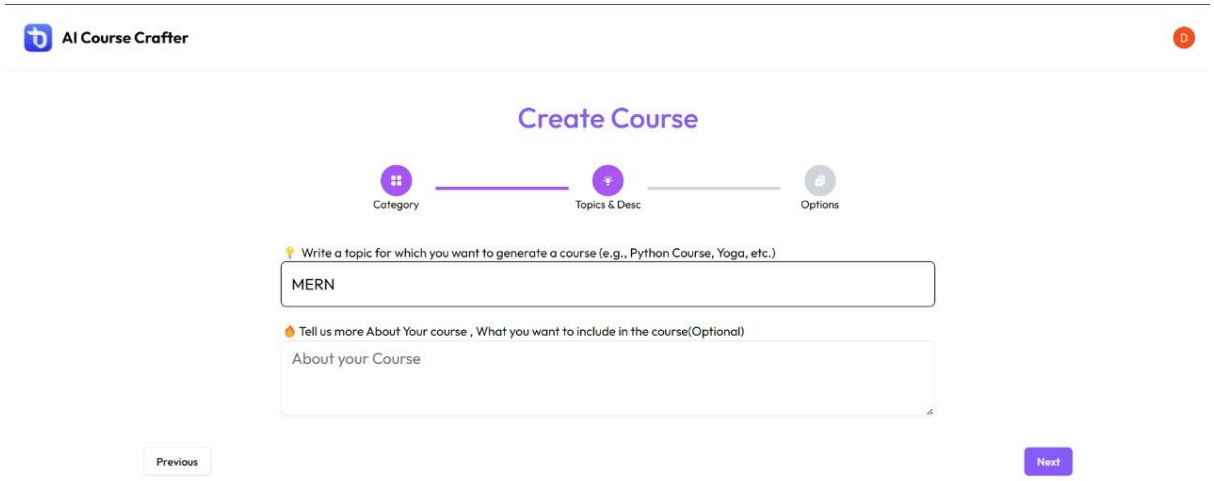


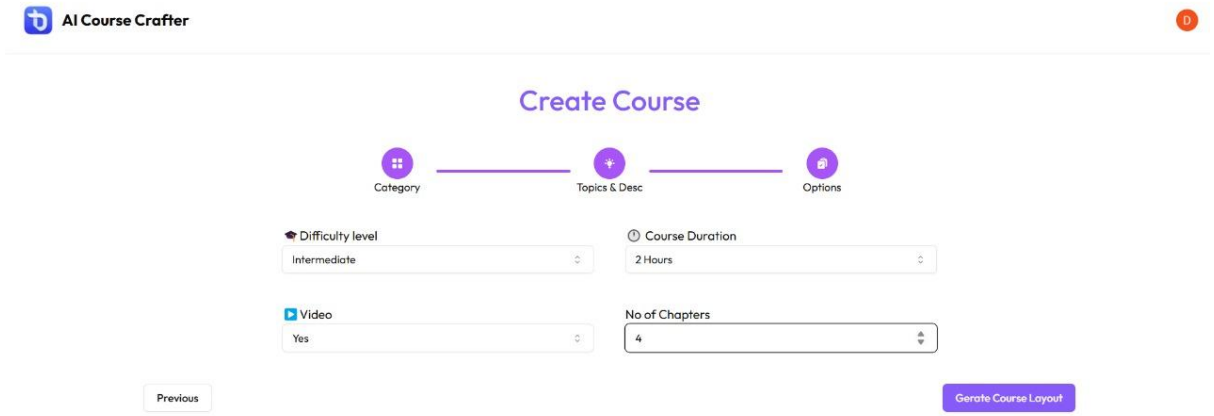


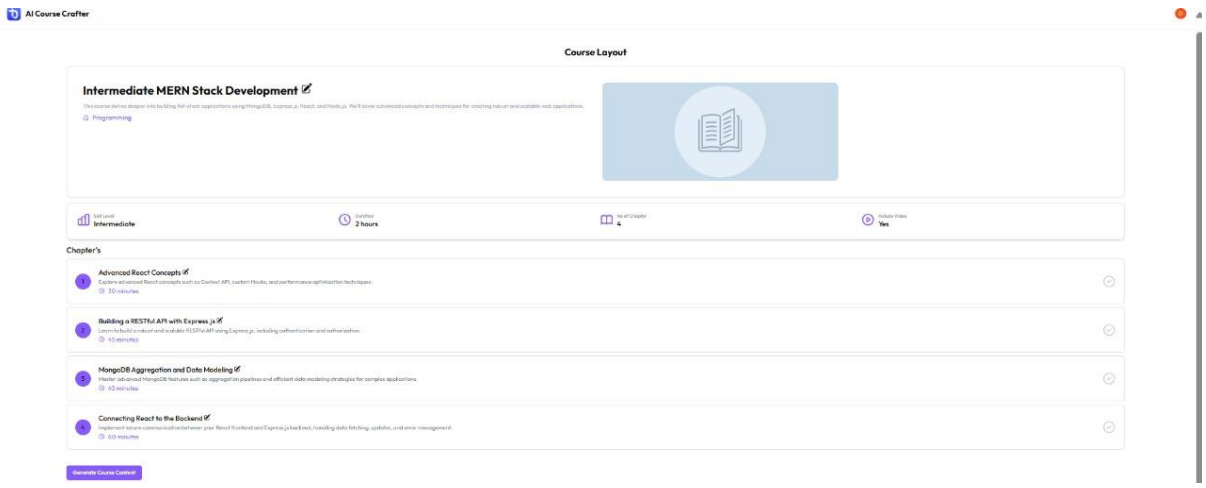


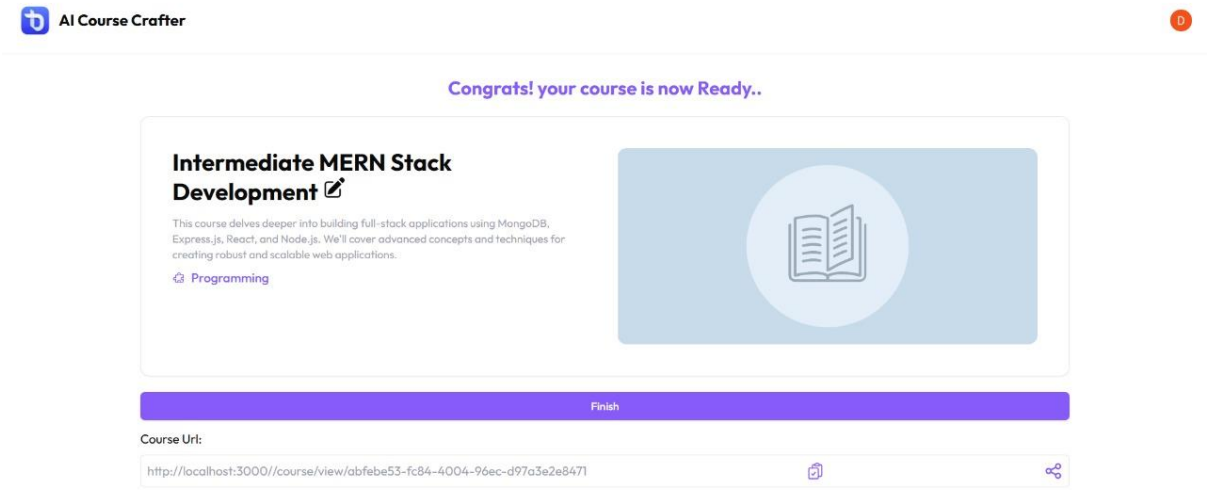

AI Course Crafter
home
Explore
Upgrade
Admin Users
Logout
Courses Created: 5
Explore More Projects
Explore more project build with AI by other users
Baking Basics for Beginners
Creative
6-Chapters
Beginner
Shreya Rasale
Python Basics for Beginners
Programming
10-Chapters
Beginner
DARSHAN RAMACHANDRA KHATAWAKAR
Intermediate SQL Programming
Programming
6-Chapters
Intermediate
Shreya Rasale
Mock interview
LLMs
For curious

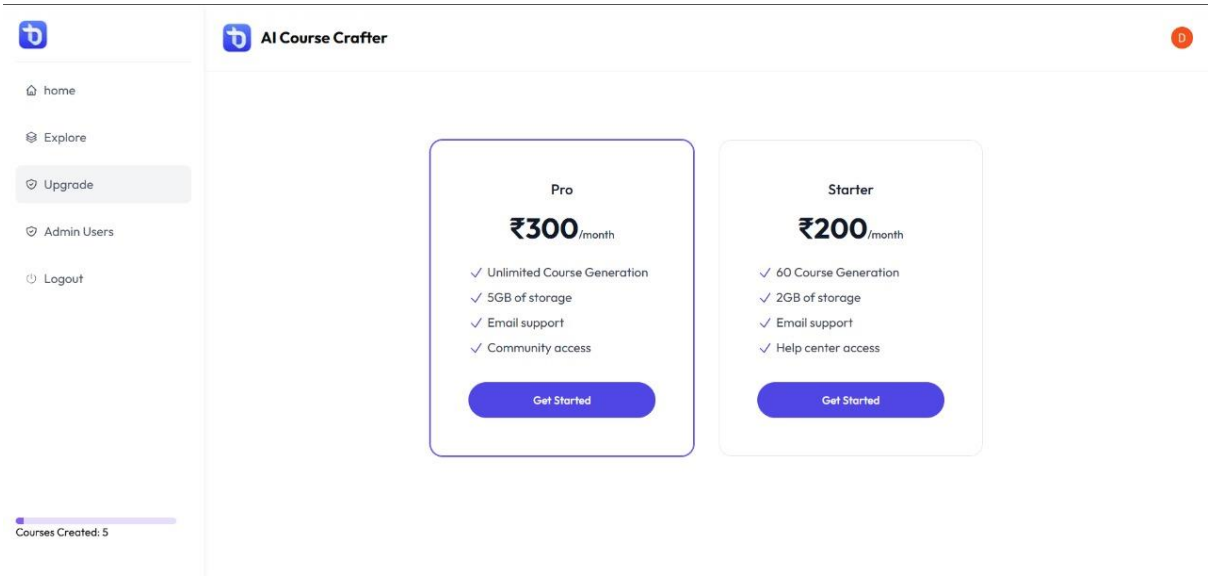

AI Course Crafter
Create Course
Category
Topics & Desc
Options
Select the course category
Programming
Health
Creative
Previous
Next


AI Course Crafter
Create Course
Category
Topics & Desc
Options
Write a topic for which you want to generate a course (e.g., Python Course, Yoga, etc.)
MERN
Tell us more About Your course , What you want to include in the course(Optional)
About your Course
Previous
Next


AI Course Crafter
Create Course
Category
Topics & Desc
Options
Difficulty level
Intermediate
Course Duration
2 Hours
Video
Yes
No of Chapters
4
Previous
Gerate Course Layout


AI Course Crafter
Course Layout
Intermediate MERN Stack Development
Programming
Intermediate
2 hours
4
Yes
Chapter's
Advanced React Concepts
10 minutes
Building a RESTful API with Express.js
45 minutes
MongoDB Aggregation and Data Modeling
45 minutes
Connecting React to the Backend
60 minutes


AI Course Crafter
D
Congrats! your course is now Ready..
Intermediate MERN Stack Development
This course delves deeper into building full-stack applications using MongoDB, Express.js, React, and Node.js. We'll cover advanced concepts and techniques for creating robust and scalable web applications.
Programming
Finish
Course Url:
http://localhost:3000//course/view/abfebe53-fc84-4004-96ec-d97a3e2e8471


AI Course Crafter
D
home
Explore
Upgrade
Admin Users
Logout
Courses Created: 5
Pro
₹300/month
Unlimited Course Generation
5GB of storage
Email support
Community access
Get Started
Starter
₹200/month
60 Course Generation
2GB of storage
Email support
Help center access
Get Started

Results and Inference:

Results

- Authentication & UI: Secure Google sign-in via Clerk with a responsive UI built using Next.js, TailwindCSS, and ShadCN.
- AI-Powered Course Creation: Step-based UI with Gemini API for auto-generating course outlines, descriptions, and chapters.
- Content & Database Management: Drizzle ORM handles structured data, while Firebase enables image uploads and real-time updates.
- Course Customization: Users can edit details, upload images, and preview AI-generated structured content with embedded media.

Inference

- Secure Authentication: Ensures safe access with smooth onboarding and dashboard navigation.
- Modern UI: Modular, responsive, and scalable, built with TailwindCSS and ShadCN UI.
- AI-Powered Courses: Uses Gemini API for structured outlines and YouTube API for video integration

- Efficient Storage: Drizzle ORM for structured data, Firebase for real-time updates.
- Flexible Editing: Users can modify course details, images, and content before finalization.

Future Scope

1. AI-Powered Course Customization – Enhance course personalization with advanced AI models.
2. Multimedia & Interactive Content – Integrate videos, quizzes, and AI tutors for better engagement.
3. Multi-Language & Global Access – Expand support for various languages and regions.
4. Marketplace & Monetization – Allow users to publish and sell courses.
5. Mobile & Cloud Integration – Enable seamless access across devices with cloud storage.

Conclusion

The AI Course Crafter project has successfully automated course creation across various domains, integrating AI for content generation, personalization, and structuring. This enhances accessibility, efficiency, and scalability in e-learning. The project demonstrates the potential of AI in revolutionizing digital education, making high-quality learning more widely available. Future improvements, such as interactive content, multilingual support, and adaptive learning paths, can further enhance its impact and usability.