

PRODUCT DOCUMENTATION

CareerMap India – Smart Career Decision Platform for Engineers

1 Product Overview

Product Name

CareerMap India (Working Title)

Tagline

“Your Smart Career Navigator After B.Tech”

Product Type

Web-based career intelligence and decision platform for Indian engineering students.

2 Problem Statement

Engineering students in India face:

- Confusion after B.Tech
- Lack of structured guidance
- Too much scattered information
- No personalized recommendations
- No comparison between career paths
- No clear preparation roadmap

Existing blogs provide static content, not decision systems.

3 Vision

To become India's most structured and personalized career decision platform for engineering students.

4 Mission

Help students:

- Make informed decisions
 - Reduce career confusion
 - Plan structured preparation
 - Compare options intelligently
 - Understand ROI of their choices
-

5 Target Audience

Primary:

- 3rd & 4th year B.Tech students
- Fresh graduates (0–2 years)

Secondary:

- Diploma students
- Parents
- Career counselors

Geography:

India (Phase 1)

6 Core Features (MVP)

We will build:

1 Career Categories

- Private Jobs
 - Higher Studies
 - Government Jobs
 - Entrepreneurship
-

2 Branch-Specific Pages

Branches supported in MVP:

- CSE
- ECE
- Mechanical
- Civil

- EEE

Each branch page includes:

- Core roles
 - IT switch options
 - Govt options
 - Higher studies options
 - Demand distribution chart
-

3 Career Detail Pages

Each career page includes:

- Overview
 - Salary (Fresher / 3 Years / 5 Years)
 - Demand Level
 - Required Skills
 - Risk Level
 - Study Requirement
 - Who Should Choose This
 - 12-Month Roadmap
 - Compare Button
-

4 Personalized Career Quiz

Input parameters:

- Branch
- Interests
- Risk tolerance
- Salary expectation
- Study preference
- Stability preference

Output:

Top 3 recommended careers with:

- Match %
- Explanation
- Comparison option

5 Career Comparison Engine

User selects 2–3 careers.

System compares:

- Salary
 - Preparation time
 - Competition level
 - Risk
 - Stability
 - Growth potential
 - ROI
-

7 Detailed Roadmap Structure

Each roadmap contains:

- Timeline (Month-wise)
- Skills to learn
- Tools required
- Projects to build
- Certifications (Optional)
- Job preparation plan
- Interview focus areas

Example:

Data Analyst – 12 Month Plan

PSU Preparation – 18 Month Plan

MBA via CAT – 12 Month Plan

8 India-Focused Data Integration

Data sections:

- Salary trends (Indian market)
- Exam details (GATE, CAT, SSC JE, UPSC)
- Competition levels
- Typical ROI
- College tier reality check

9 Exam Planner (Phase 2)

- Exam countdown
 - Study calendar generator
 - Syllabus breakdown
 - Monthly targets
-

10 User Flow

User Journey:

Home
→ Take Quiz
→ View Recommended Careers
→ Explore Career Details
→ Compare Options
→ View Roadmap
→ Create Study Plan

Retention loop:

Save careers → Return to track progress

11 Unique Differentiation

Blogs provide:

- Information

CareerMap provides:

- Personalization
- Comparison system
- Data-driven insights
- Structured roadmaps
- India-focused execution plan

This is a decision engine, not a blog.

12 MVP Scope (Keep It Simple)

Phase 1 (Build First):

- Homepage
- Career Categories
- 10 Detailed Career Pages
- 5 Branch Pages
- Basic Quiz Logic
- Comparison Table

No login required initially.

1B Database Structure (High-Level)

Tables / Collections:

Branches
Careers
Career Categories
Roadmaps
Quiz Questions
Quiz Logic Mapping
Salary Data
Exam Data

1H Tech Stack Recommendation

Frontend:

React + Tailwind CSS

Backend:

Node.js + Express

Database:

MongoDB

Alternative (Beginner Friendly):

HTML + CSS + JS + Firebase

15 Monetization (Future)

- Premium roadmaps

- Personalized AI consultation
 - Mentorship booking
 - Affiliate course links
 - Ads
 - College partnerships
-

16 Future Enhancements

- AI Chat Career Advisor
 - Resume Builder
 - Internship Matching
 - College Predictor
 - Skill Gap Analyzer
 - Dashboard with progress tracking
 - Mobile App
-

17 Development Phases

Phase 1 – MVP (2–3 months)

Core structure + quiz + 10 careers

Phase 2 – Data Expansion (3–5 months)

More careers + salary dashboard

Phase 3 – AI Integration (6+ months)

Smart recommendation engine

18 Risk Analysis

Challenges:

- Maintaining updated data
- Ensuring accuracy
- Competition from blogs
- User trust building

Mitigation:

- Quarterly updates
- Data references
- Clean UI

- Student testimonials
-

19 Success Metrics

- Monthly active users
 - Quiz completion rate
 - Comparison usage rate
 - Roadmap downloads
 - User retention rate
-

20 Final Product Positioning

This is not:

- ✗ Just a blog
- ✗ Just exam information
- ✗ Just salary listing

This is:

- ✓ A structured career decision system
 - ✓ India-focused engineering guide
 - ✓ Data-backed roadmap planner
-

🔥 Now we are officially designing a startup-level product.

Next step — choose:

- 1 Create a professional PDF documentation file
- 2 Convert this into a final year project report format
- 3 Break this into development tasks (step-by-step build plan)
- 4 Create technical system architecture diagram

Tell me your goal.

We move to next stage. 🚀

ok u have gave me the documentation. can u also give me for the specific career path which include these fields (Career categories, Branch-specific pages, detailed roadmaps, Basic quiz and also like these quizzes like this Branch (CSE, ECE, Mech, Civil, etc.) Interest (Coding, Management, Govt, Research) Risk tolerance (Low/Medium/High) Salary expectation Study

preference (Want to study more? Yes/No) Work-life balance importance Financial condition
(Can afford MS/MBA?)

Excellent 🌟

Now we are moving into **deep structured career documentation**.

I will create a **complete example for ONE specific career path** so you understand the full system structure.

Let's choose a strong example:

🎯 Career Path Example: Data Analyst (India Focused)

This will include:

- Career Category
 - Branch-Specific Mapping
 - Detailed Roadmap
 - Career Quiz Logic
 - Decision Engine Mapping
 - Comparison Readiness
-

1 Career Category Placement

Category:

🟢 Private Sector Jobs → IT / Software Careers

Sub-Category:

Data & Analytics

2 Branch-Specific Mapping

🎓 Eligible Branches

Direct Fit (High Compatibility)

- CSE
- IT
- ECE
- EEE

Moderate Fit (Needs Skill Transition)

- Mechanical
 - Civil
 - Chemical
 - Production
-

Branch-Specific Guidance Example

If Branch = CSE

- Advantage: Programming foundation
- Preparation Time: 6–8 months
- Focus: SQL + Power BI + Projects

If Branch = Mechanical

- Need to learn basics of coding
 - Preparation Time: 8–12 months
 - Focus: Excel → SQL → Python → Projects
-

3 Career Detail Documentation

Role Overview

A Data Analyst collects, cleans, analyzes, and visualizes data to help companies make decisions.

Salary (India)

- Fresher: ₹4–8 LPA
 - 3 Years: ₹8–15 LPA
 - 5 Years: ₹15–25 LPA
-

Demand Level

High (Growing due to AI & digital transformation)

Risk Level

Medium

(Competitive but skill-based entry)

Who Should Choose This?

- ✓ Enjoy working with numbers
 - ✓ Analytical thinking
 - ✓ Comfortable with computers
 - ✓ Prefer corporate work
-

Detailed Roadmap (12 Month Plan)

Month 1–2:

- Excel (Advanced)
- Data cleaning
- Pivot tables

Month 3–4:

- SQL (SELECT, JOIN, GROUP BY)
- Practice on real datasets

Month 5–6:

- Power BI / Tableau
- Build 2 dashboards

Month 7–8:

- Python (Pandas, NumPy)
- Data visualization

Month 9–10:

- 3 Real Projects:
 - Sales dashboard
 - HR analytics
 - E-commerce analysis

Month 11:

- Resume building
- LinkedIn optimization

Month 12:

- Apply to 200+ companies
 - Mock interviews
-

5 Basic Quiz Integration Logic

Now we integrate your requested quiz fields.

🎓 Branch Input

If:

Branch = CSE → +15 score
Branch = Mechanical → +10 score
Branch = Civil → +8 score

💡 Interest

If:

Interest = Coding → +20
Interest = Research → +10
Interest = Govt → -10

⚖️ Risk Tolerance

If:

High → +10
Medium → +15
Low → +5

(Data analytics is medium risk)

💰 Salary Expectation

If:

₹8 LPA → +15
₹5–8 LPA → +10
< ₹5 LPA → +5

Study Preference

If:

No (don't want more study) → +15
Yes (want M.Tech) → +5

Work-Life Balance Importance

If:

Medium → +15
High → +10
Low → +5

Financial Condition

If:

Cannot afford MS/MBA → +15
Can afford → +5

(Data Analyst is cheaper compared to MS abroad)

Final Recommendation Logic

If total score > 70

→ Recommend Data Analyst (Strong Fit)

If 50–70

→ Moderate Fit

If < 50

→ Suggest alternative like:

- MBA

- Govt Jobs
 - Core Engineering
-

6 Comparison Readiness

When user compares:

Factor	Data Analyst	PSU	MBA
Study Time	6–12 months	1–2 years	2 years
Cost	Low	Low	High
Salary Growth	High	Stable	Very High
Risk	Medium	Low	High

7 System View (How It Works in App)

User fills quiz →
System calculates score for each career →
Top 3 displayed with match % →
User views roadmap →
User compares →
User selects career →
System suggests preparation timeline

8 This Same Framework Can Be Repeated For:

- PSU Engineer
- MBA via CAT
- M.Tech via GATE
- MS Abroad
- Software Developer

- Government SSC JE
- Entrepreneurship

Each career gets:

- Category mapping
 - Branch compatibility
 - Detailed roadmap
 - Quiz scoring logic
 - Risk profile
 - ROI model
-

🔥 Now important question:

Do you want me to next create:

- 1 PSU Engineer full documentation
- 2 MBA via CAT full documentation
- 3 MS Abroad full documentation
- 4 Software Developer full documentation

Choose one. We build it completely structured like this.

can u provide me BRS document and also the end to end screen flow ?

Excellent. Now we move to **professional-level documentation** 🌟

I'll give you:

- 1 **BRS (Business Requirement Specification) Document**
- 2 **End-to-End Screen Flow (Complete User Journey)**

This will look like something you can submit as:

- Startup planning document
 - Final year project documentation
 - Investor pitch support document
-

BUSINESS REQUIREMENT SPECIFICATION (BRS)

Project Name:

CareerMap India – Career Decision Platform for Engineers

1 Business Objective

To build a structured, personalized, India-focused career guidance web platform that helps B.Tech students:

- Explore career paths
 - Compare options
 - Receive personalized recommendations
 - Access structured roadmaps
 - Plan preparation timelines
-

2 Problem Statement

Engineering students in India face:

- Career confusion after graduation
- Unstructured guidance
- No personalization
- No career comparison tools
- No actionable preparation roadmap

Existing solutions are:

- Static blogs
 - YouTube advice
 - Coaching-driven platforms
 - Non-personalized portals
-

3 Business Goals

- Reduce student career confusion
 - Increase decision clarity
 - Provide structured planning
 - Build scalable digital career platform
 - Enable monetization in future
-

4 Scope of the Project

Included in MVP:

- Career Categories
- Branch-Specific Pages
- Career Detail Pages
- Personalized Career Quiz
- Comparison Engine
- Basic Roadmaps

Not Included in MVP:

- AI chatbot
 - Paid mentorship
 - Resume builder
 - Internship matching
 - Mobile app
-

5 Stakeholders

Primary Users:

- Engineering students
- Fresh graduates

Secondary Users:

- Parents
- Career counselors

Internal Stakeholders:

- Product Owner
 - Developer
 - Content Manager
-

6 Functional Requirements

6.1 Home Page

System shall:

- Display career categories
- Provide access to quiz
- Show trending careers

6.2 Career Category Page

System shall:

- Display careers by category
 - Allow filtering by branch
 - Show salary range
 - Show demand level
-

6.3 Career Detail Page

System shall display:

- Career overview
 - Salary growth data
 - Required skills
 - Risk level
 - Study requirements
 - 12-month roadmap
 - Comparison option
-

6.4 Branch-Specific Page

System shall:

- Show career distribution chart
 - List branch-compatible careers
 - Display recommended paths
 - Provide branch-specific roadmap suggestions
-

6.5 Career Quiz Module

System shall:

- Collect user inputs:
 - Branch
 - Interest
 - Risk tolerance
 - Salary expectation

- Study preference
 - Work-life balance importance
 - Financial condition
 - Calculate score for each career
 - Display top 3 recommendations
 - Show match percentage
-

6.6 Career Comparison Module

System shall:

- Allow selecting up to 3 careers
 - Compare on:
 - Salary
 - Study time
 - Competition
 - Risk
 - Growth
 - ROI
 - Highlight strongest attributes
-

7 Non-Functional Requirements

- Responsive design (Mobile + Desktop)
 - Fast loading time (<3 seconds)
 - Secure data handling
 - Scalable database structure
 - Clean UI/UX
-

8 Assumptions

- Salary data based on Indian market trends
 - Users provide honest quiz answers
 - Career mapping logic is pre-defined rule-based
-

9 Constraints

- Limited initial data sources

- Manual content updates required
 - Competitive market (career blogs)
-

10 Success Metrics

- Quiz completion rate
 - Comparison usage rate
 - Average time on site
 - Repeat visits
 - Roadmap engagement
-

END-TO-END SCREEN FLOW

Now I'll give you complete user journey flow.

1 Landing Flow

User enters website

→ Home Page
→ Clicks "Take Quiz" OR "Explore Careers"

2 Quiz Flow

Screen 1:
Select Branch

Screen 2:
Select Interest

Screen 3:
Select Risk Tolerance

Screen 4:
Select Salary Expectation

Screen 5:
Study Preference

Screen 6:
Work-Life Balance

Screen 7:
Financial Condition

→ Processing Screen

→ Quiz Result Screen

3 Quiz Result Flow

Displays:

Top 3 Recommended Careers
Match Percentage
Short explanation

Options:

- View Details
 - Compare
 - Explore Roadmap
-

4 Career Detail Flow

User clicks a career

→ Career Overview
→ Salary Graph
→ Skills Required
→ Roadmap Timeline
→ Compare Button

5 Comparison Flow

User selects 2–3 careers

→ Comparison Screen

Displays table:

Salary | Study Time | Risk | Growth | Stability

User decides

→ Click “View Roadmap”

6 Branch Exploration Flow

User clicks Branch Guide

→ Select Branch
→ Branch Overview
→ Career Distribution
→ Branch-specific recommendations

17 7 Roadmap Flow

User selects roadmap

→ Timeline View
→ Monthly Breakdown
→ Project Recommendations
→ Preparation Plan

Retention Loop

User can:

- Bookmark careers
 - Retake quiz
 - Compare again
 - Check exam dates (future phase)
-

Overall System Flow Diagram (Text Representation)

Home



Quiz / Explore

↓
Career Recommendation

↓
Career Details

↓
Compare

↓
Roadmap

↓
Decision

Product Structure Summary

This platform consists of:

- Information Layer
- Decision Layer (Quiz + Logic)
- Comparison Layer
- Execution Layer (Roadmap)

That makes it a complete career decision ecosystem.

Now I need to know something important:

Are you building this as:

- 1 Final Year Major Project
- 2 Startup Idea
- 3 Portfolio Project

Because if it's a final year project, I can now convert this into IEEE-style documentation format.

Tell me your direction.

3

Perfect. 🌟

Since this is a **Portfolio Project**, we will position it differently.

For portfolio:

- It must look **product-level**
- Show **problem solving**
- Show **system thinking**

- Show **technical architecture**
- Show **decision logic**
- Show **scalability thinking**
- Not just UI screens

Now I'll give you a **Portfolio-Optimized Version** of your project documentation.

Portfolio Project Documentation

CareerMap India – Intelligent Career Decision Platform

1 Project Summary (For Resume / LinkedIn)

CareerMap India is a structured, rule-based career decision web application designed for Indian engineering students. The platform provides personalized career recommendations, comparison tools, and step-by-step preparation roadmaps based on user inputs such as branch, interests, financial condition, and risk tolerance.

Core Modules:

- Career Recommendation Engine
 - Branch-Specific Career Mapping
 - Career Comparison System
 - Structured Preparation Roadmaps
-

2 Problem Statement (Portfolio Version)

Engineering students in India lack a structured, personalized, and comparative decision-making system for post-B.Tech career planning. Existing solutions provide static information but lack interactive intelligence and execution planning.

This project solves:

- Career confusion
 - Lack of personalization
 - No structured comparison
 - No actionable timeline planning
-

3 System Architecture (High-Level)

Frontend Layer

- React (UI rendering)
- Tailwind CSS (Styling)
- Chart library (salary visualization)

Backend Layer

- Node.js + Express
- REST APIs

Database Layer

- MongoDB

Collections:

- branches
 - careers
 - roadmaps
 - quizQuestions
 - quizMappingLogic
 - salaryData
-

4 Core Modules Breakdown

Module 1: Career Intelligence Engine

Inputs:

- Branch
- Interest
- Risk tolerance
- Salary expectation
- Study preference
- Work-life balance importance
- Financial condition

Logic:

Weighted scoring system for each career.

Example:

Data Analyst:

Branch (CSE) → +15

Interest (Coding) → +20

Risk (Medium) → +15

Financial (Low budget) → +15

Total Score → 85

Match % → 85%

Output:

Top 3 careers sorted by score.

This demonstrates algorithmic thinking in your portfolio.

Module 2: Branch Mapping Engine

Each branch has:

- Direct-fit careers
- Transition careers
- Govt-specific options
- Higher studies

Example:

Mechanical:

- Core Mechanical
- Data Analyst (transition)
- PSU
- MBA

This shows system design capability.

Module 3: Comparison Engine

User selects up to 3 careers.

System fetches attributes:

- Salary
- Risk
- Preparation time
- Growth

- Stability

Renders dynamic comparison table.

Shows decision-support modeling.

Module 4: Roadmap Engine

Each career roadmap includes:

- Monthly breakdown
- Skill dependency order
- Project milestones
- Interview preparation phase

This demonstrates structured execution planning logic.

5 End-to-End Screen Flow (Portfolio-Ready Format)

Entry Point

Landing Page



CTA: Take Career Quiz / Explore Careers

Quiz Flow

Branch Selection



Interest



Risk



Salary Expectation



Study Preference



Work-Life Balance



Financial Condition
↓
Processing (Score Calculation)
↓
Results Page

Results Page

Top 3 Recommendations

Match %

Explanation

Buttons:

- View Details
 - Compare
 - Explore Roadmap
-

Career Detail Page

Overview

Salary Chart

Skills

Risk

Roadmap

Compare Button

Comparison Page

Dynamic Table View

Highlight Differences

Decision Support Summary

Branch Guide Flow

Select Branch

↓

Branch Overview

↓

Career Distribution



Recommended Careers

Roadmap Flow

Timeline View

Monthly Milestones

Resources

Interview Phase

6 Unique Technical Highlights (Important for Portfolio)

You should mention:

- Rule-based recommendation engine
- Weighted scoring algorithm
- Modular architecture
- Scalable database design
- Data modeling for career mapping
- Decision-support system design

Recruiters love this.

7 Future Scalability (Shows Vision)

- AI-powered recommendation using ML
 - User account system
 - Progress tracking dashboard
 - Internship integration
 - Resume analyzer
 - Mobile app version
-

8 How To Present This in Portfolio

On GitHub:

README should include:

- Problem
- Solution
- Architecture
- Screenshots
- Demo link
- Features
- Tech stack

On Resume:

"Built a rule-based career recommendation web application using React, Node.js, and MongoDB that provides personalized career suggestions, comparison tools, and structured roadmaps for engineering students."

9 Why This Is Strong For You

Since you are interested in:

- Data analytics
- Web development

This project shows:

- Logical decision modeling
- Data structuring
- Frontend development
- Backend API creation
- Product thinking

This is much stronger than a simple CRUD app.

Now let's go deeper.

Next, I can give you:

- 1 Database schema design (actual MongoDB structure)
- 2 API endpoint structure
- 3 Scoring algorithm pseudo-code
- 4 Complete development roadmap (what to build week-by-week)
- 5 How to turn this into a strong GitHub project

Choose your next step. 

after analyzing everything can u provide me a end to end screen flow that if this documentation is given to any designer or any user must understand it detailed and develop me the BRS document

Excellent. Now we will create a **professional, designer-ready End-to-End Screen Flow + Refined BRS Document.**

This version will be:

- Clear enough for UI/UX designers
- Structured enough for developers
- Professional enough for portfolio
- Detailed enough for product understanding

No confusion. No gaps.

BUSINESS REQUIREMENT SPECIFICATION (BRS)

Project: CareerMap India

Type: Portfolio Product (Decision Support Web Application)

1 Executive Summary

CareerMap India is a web-based decision-support platform designed to help Indian B.Tech students choose structured career paths through:

- Personalized career recommendations
- Branch-specific guidance
- Career comparison tools
- Step-by-step preparation roadmaps

The platform uses a rule-based scoring engine to provide intelligent recommendations.

2 Business Objective

Build a scalable, structured career intelligence system that:

- Reduces career confusion
- Enables informed decision-making
- Provides execution-ready preparation roadmaps

- Demonstrates system design & algorithmic thinking (Portfolio Purpose)
-

③ Scope of the System

Included (MVP Scope)

- Home Page
- Career Categories
- Branch-Specific Pages
- Career Detail Pages
- Personalized Quiz Engine
- Comparison Module
- Roadmap Module

Excluded (Future Scope)

- Login/Signup
 - Payment integration
 - AI chatbot
 - Resume builder
 - Internship integration
-

④ User Roles

Primary User

Engineering Student

No authentication required in MVP.

⑤ Functional Requirements

FR-1: Home Page

System shall:

- Display hero section
- Provide CTA buttons:
 - Take Career Quiz

- Explore Careers
 - Display Career Categories
 - Display Trending Careers
-

FR-2: Career Categories Page

System shall:

- Display 4 main categories:
 - Private Sector
 - Higher Studies
 - Government Jobs
 - Entrepreneurship
 - Allow filtering by:
 - Branch
 - Salary range
 - Study requirement
 - Risk level
-

FR-3: Branch-Specific Page

System shall:

- Display selected branch overview
 - Show career distribution chart
 - Display:
 - Core careers
 - IT switch careers
 - Govt careers
 - Higher studies
 - Recommend top career paths for that branch
-

FR-4: Career Detail Page

System shall display:

- Career overview
- Salary progression (Fresher / 3Y / 5Y)
- Required skills
- Risk level

- Study requirement
 - Competition level
 - Preparation time
 - Step-by-step roadmap
 - Compare button
-

FR-5: Career Quiz Module

System shall:

Collect inputs:

- Branch
- Interest
- Risk tolerance
- Salary expectation
- Study preference
- Work-life balance importance
- Financial condition

System shall:

- Apply weighted scoring logic
 - Calculate career match percentage
 - Display top 3 recommended careers
 - Show explanation
-

FR-6: Career Comparison Module

System shall:

- Allow selection of 2–3 careers
- Display comparison table
- Compare:
 - Salary
 - Study duration
 - Cost
 - Risk
 - Competition
 - Growth potential
 - Stability

FR-7: Roadmap Module

System shall:

- Display timeline view
 - Show monthly breakdown
 - List required tools
 - Suggest project milestones
 - Include interview preparation stage
-

6 Non-Functional Requirements

- Responsive (Mobile + Desktop)
 - Page load < 3 seconds
 - Clean UI
 - Modular code structure
 - Scalable database model
-

7 End-to-End Screen Flow (Designer-Ready)

Now this is very detailed.

This can directly be given to a UI/UX designer.

GLOBAL NAVIGATION STRUCTURE

Navbar:

Logo | Home | Explore Careers | Branch Guide | Quiz | Compare | About

SCREEN FLOW 1: Entry Journey

Screen 1: Home Page

Sections (Top to Bottom):

1. Navbar
2. Hero Section

- Headline
 - Subtext
 - CTA Buttons:
 - Take Career Quiz
 - Explore Careers
3. Career Categories Grid (4 Cards)
 4. Trending Careers Horizontal Scroll
 5. Footer

User Actions:

- Click Quiz
 - Click Category
 - Click Branch Guide
-

SCREEN FLOW 2: Career Quiz Journey

Screen 2: Quiz – Step 1

Question: Select Your Branch

Options: CSE, ECE, Mechanical, Civil, EEE

Next →

Screen 3: Quiz – Step 2

Question: Primary Interest

Options:

- Coding
- Management
- Government
- Research
- Entrepreneurship

Next →

Screen 4: Quiz – Step 3

Risk Tolerance:
Low / Medium / High

Next →

Screen 5: Quiz – Step 4

Salary Expectation:
<5 LPA / 5–8 LPA / >8 LPA

Next →

Screen 6: Quiz – Step 5

Study Preference:
Yes / No

Next →

Screen 7: Quiz – Step 6

Work-Life Balance Importance:
Low / Medium / High

Next →

Screen 8: Quiz – Step 7

Financial Condition:
Can afford higher studies / Limited budget

Submit →

Screen 9: Processing Screen

Loading animation
"Analyzing your responses..."

Screen 10: Quiz Result Screen

Displays:

Top 3 Careers

Each card shows:

- Career Name
 - Match %
 - Why recommended
 - Buttons:
 - View Details
 - Compare
-

SCREEN FLOW 3: Career Exploration Journey

Screen 11: Career Category Page

Filters on left

Career cards grid on right

User selects career →

Screen 12: Career Detail Page

Sections:

1. Title + Category Tag
 2. Overview
 3. Salary Graph
 4. Skills Required
 5. Risk Level
 6. Study Requirement
 7. Roadmap Timeline
 8. Compare Button
-

SCREEN FLOW 4: Comparison Journey

Screen 13: Compare Selection Page

User selects 2–3 careers

Click Compare →

Screen 14: Comparison Result Page

Table Layout:

Rows:

- Salary
- Study Time
- Cost
- Competition
- Risk
- Stability
- Growth

Highlight best option visually.

SCREEN FLOW 5: Branch Journey

Screen 15: Branch Selection Page

Cards:

CSE | ECE | Mechanical | Civil | EEE

User selects branch →

Screen 16: Branch Detail Page

Sections:

1. Branch Overview

-
- 2. Career Distribution Chart
 - 3. Core Careers
 - 4. IT Switch Careers
 - 5. Govt Careers
 - 6. Higher Studies
 - 7. Recommended Roadmaps
-

 **17 SCREEN FLOW 6: Roadmap Journey**

Screen 17: Roadmap Page

Timeline View:

Month 1–2

Month 3–4

Month 5–6

...

Each expandable:

- Skills
 - Tools
 - Projects
 - Interview Prep
-

 **Complete User Journey Summary**

Home

→ Quiz

→ Result

→ Career Details

→ Compare

→ Roadmap

→ Decision

OR

Home

→ Explore

→ Career Details

→ Compare
→ Roadmap

OR

Home
→ Branch
→ Branch Details
→ Career
→ Roadmap

System Layer Model

Information Layer
Decision Layer (Quiz Engine)
Comparison Layer
Execution Layer (Roadmap)

Final Result

Now this documentation is:

- Clear for Designer
 - Clear for Developer
 - Clear for Evaluator
 - Strong for Portfolio
-

If you want next level:

I can now give you:

- UML Diagrams (Use Case, Sequence, Activity)
- Database ER Diagram
- API Documentation
- UI Component Hierarchy
- Development sprint plan

Tell me which direction you want to go. 

UML Diagrams (Use Case, Sequence, Activity)

Excellent. 🔥

Now we move into **proper system design level documentation**.

I will give you:

- ① Use Case Diagram (Actors + System Interaction)
- ② Sequence Diagram (Quiz & Recommendation Flow)
- ③ Activity Diagram (Overall User Flow Logic)

I'll describe them clearly so:

- You can draw them in Draw.io / Lucidchart / StarUML
 - You can submit them in portfolio
 - Any designer/developer can understand system behavior
-

① USE CASE DIAGRAM

Actor

Primary Actor:

 Engineering Student

System:

 CareerMap India System

Main Use Cases

Student can:

- View Home Page
 - Explore Career Categories
 - View Career Details
 - Select Branch Guide
 - Take Career Quiz
 - View Quiz Results
 - Compare Careers
 - View Roadmap
-

Use Case Relationships

Core Use Cases

Student

- (Take Career Quiz)
 - (View Recommendations)
 - (View Career Details)
 - (Compare Careers)
 - (View Roadmap)
-

Text Representation (For Drawing)

Actor: Student

System Boundary: CareerMap India

Inside System:

- UC1: Browse Career Categories
- UC2: View Career Details
- UC3: Select Branch Guide
- UC4: Take Career Quiz
- UC5: Generate Career Recommendation
- UC6: Compare Careers
- UC7: View Career Roadmap

Relationships:

UC4 (Take Quiz) → includes → UC5 (Generate Recommendation)

UC2 (View Career Details) → extends → UC6 (Compare Careers)

SEQUENCE DIAGRAM

Scenario: User Takes Career Quiz

Participants

- User
- Frontend (UI)
- Quiz Controller (Backend)
- Recommendation Engine
- Database

Flow

1. User selects "Take Quiz"
 2. Frontend displays quiz questions
 3. User submits answers
 4. Frontend sends answers to Backend
 5. Backend calls Recommendation Engine
 6. Recommendation Engine:
 - o Fetches career data from Database
 - o Applies scoring logic
 - o Calculates match %
 7. Backend sends result to Frontend
 8. Frontend displays Top 3 Careers
-

Text Sequence Representation

User → UI: Start Quiz
UI → User: Display Questions
User → UI: Submit Answers
UI → Backend: Send Quiz Data
Backend → Recommendation Engine: Process Data
Recommendation Engine → Database: Fetch Career Data
Database → Recommendation Engine: Return Career Data
Recommendation Engine → Backend: Return Scored Results
Backend → UI: Send Recommendations
UI → User: Display Top 3 Careers

ACTIVITY DIAGRAM

Overall Career Decision Flow

Start
↓
User opens website
↓
Decision Node:
Take Quiz?

YES → Go to Quiz Flow
NO → Explore Careers

Quiz Flow

Fill Answers



Submit



System Calculates Score



Display Top 3 Careers



Decision:

Compare Careers?

YES → Go to Compare Page

NO → View Career Detail

Career Detail Flow

View Overview



View Salary



View Roadmap



End

Explore Flow (Without Quiz)

Browse Categories



Select Career



View Detail



Compare (Optional)



[View Roadmap](#)

↓

End

⌚ Simplified Activity Representation

Start

↓

Home

↓

[Quiz OR Explore]

↓

Career Detail

↓

Compare (Optional)

↓

Roadmap

↓

Decision Complete

↓

End

🔥 Bonus: Recommendation Engine Logic (For UML Detail)

Decision Engine Activity:

Start

↓

Initialize Score = 0

↓

Evaluate Branch Weight

↓

Evaluate Interest Weight

↓

Evaluate Risk Weight

↓

Evaluate Salary Expectation

↓

Evaluate Study Preference

↓

Evaluate Financial Condition

↓
Compute Final Score

↓
Sort Careers by Score
↓
Return Top 3

End

How This Strengthens Your Portfolio

Now your project includes:

- Use Case Modeling
- System Interaction Modeling
- Behavioral Flow Modeling
- Algorithmic Thinking
- Backend Architecture Thinking

This is now above average portfolio level.

If you want next level documentation, I can give:

- Class Diagram
- ER Diagram (Database Design)
- Component Diagram
- Deployment Diagram
- Recommendation Algorithm Pseudocode
- Microservice-ready architecture (advanced level)

Tell me how deep you want to go. 

UI Component Hierarchy

Excellent. 

Now we move into **Frontend System Design Level** — this is very powerful for portfolio.

I'll give you a **clear UI Component Hierarchy** structured like a real React application.

This will help:

- UI/UX Designer understand layout
- Frontend Developer build clean structure

- Recruiter see component architecture thinking
 - You show modular design capability
-

UI COMPONENT HIERARCHY

CareerMap India (React-Based Structure)

① Root Structure

App

```
  └── Router
      |   ├── HomePage
      |   ├── QuizPage
      |   ├── QuizResultPage
      |   ├── CareerCategoryPage
      |   ├── CareerDetailPage
      |   ├── ComparePage
      |   ├── BranchSelectionPage
      |   ├── BranchDetailPage
      |   └── RoadmapPage
    └── Navbar
    └── Footer
```

② Global Components (Reusable)

These are used across multiple pages.

Common Components

```
|—— Button  
|—— Card  
|—— Tag  
|—— Badge  
|—— ProgressBar  
|—— Modal  
|—— Loader  
|—— ChartComponent  
|—— ComparisonTable  
└—— Timeline
```

These make your UI scalable and modular.

③ Home Page Component Hierarchy

HomePage

```
|—— HeroSection  
|   |—— Heading  
|   |—— SubHeading  
|   |—— CTAButtons  
|—— CareerCategoryGrid  
|   |—— CategoryCard (x4)  
|—— TrendingCareersSection  
|   |—— CareerCard (horizontal scroll)  
└—— Footer
```

4 Career Category Page Structure

CareerCategoryPage

```
|—— FilterPanel  
|   |—— BranchFilter  
|   |—— SalaryFilter  
|   |—— RiskFilter  
|   |—— StudyPreferenceFilter  
|—— CareerGrid  
|   |—— CareerCard (dynamic list)  
└—— Pagination (optional)
```

5 Career Detail Page Structure

CareerDetailPage

```
|—— CareerHeader  
|   |—— CareerTitle  
|   |—— CategoryTag  
|   |—— CompareButton  
|—— OverviewSection  
|—— SalarySection  
|   |—— SalaryChart  
|—— SkillsSection  
|   |—— SkillTagList  
|—— RiskStudySection
```

└── RoadmapPreviewSection

 └── TimelineComponent

└── RelatedCareersSection

6 Quiz Page Component Hierarchy

QuizPage

 └── QuizProgressBar

 └── QuizQuestionCard

 └── QuestionText

 └── OptionCard (multiple)

 └── NavigationButtons

 └── BackButton

 └── NextButton

 └── SubmitButton (final step)

7 Quiz Result Page

QuizResultPage

 └── ResultHeader

 └── RecommendedCareerList

 └── MatchCard

 └── CareerName

 └── MatchPercentage

 └── ExplanationText

```
|      └── ViewDetailsButton  
|      └── CompareButton  
└── RetakeQuizButton
```

8 Comparison Page

ComparePage

```
├── SelectedCareerSummary  
├── ComparisonTable  
|   ├── SalaryRow  
|   ├── StudyTimeRow  
|   ├── RiskRow  
|   ├── GrowthRow  
|   ├── StabilityRow  
└── HighlightBestOption
```

9 Branch Selection Page

BranchSelectionPage

```
├── BranchGrid  
|   ├── BranchCard (CSE, ECE, Mech, Civil, EEE)
```

10 Branch Detail Page

BranchDetailPage

```
├── BranchHeader  
└── CareerDistributionChart
```

```
└── CoreCareersSection
    |   └── CareerCard
    └── ITSwitchCareersSection
        |   └── CareerCard
    └── GovtCareersSection
        |   └── CareerCard
    └── HigherStudiesSection
        └── CareerCard
```

11 Roadmap Page

RoadmapPage

```
└── RoadmapHeader
    └── TimelineView
        |   └── MonthBlock
        |   └── SkillsList
        |   └── ToolsList
        |   └── ProjectList
        |   └── InterviewPrepSection
    └── ResourceLinks
```

🔁 State Management (Important for Portfolio)

Global States:

- SelectedCareers (for comparison)
- QuizAnswers

- QuizResults
- SelectedBranch
- FiltersState

Can be handled using:

- React Context API (MVP)
 - OR
 - Redux (Advanced)
-

Component Layer Architecture

Presentation Layer (UI Components)

Business Logic Layer (Quiz Scoring Engine)

Data Layer (API Calls / MongoDB)

This shows layered architecture thinking.

Why This Is Strong for Portfolio

This demonstrates:

- Component modularity
- Reusability
- Clean separation of concerns
- Scalable frontend design
- Real-world project architecture

This is much stronger than basic CRUD apps.

If you want next-level frontend documentation, I can now give:

- Folder structure (Production-level React setup)
- State management architecture diagram
- API integration flow
- Design system guidelines (Typography, spacing, color scale)
- Complete frontend development roadmap

Tell me how advanced you want to go. 

UI/UX Designer understand layout Frontend Developer build clean structure

Perfect 

Now I'll explain this in a **professional documentation style** so:

-  **UI/UX Designer clearly understands layout & behavior**
-  **Frontend Developer understands structure & implementation logic**

This will look like something you submit in a real product company.

UI/UX + Frontend Structural Understanding Document

Project: CareerMap India

1 Design Philosophy

Goal

Help B.Tech students choose the right career path using:

- Structured career categories
 - Branch-specific guidance
 - Personalized quiz
 - Roadmaps
 - Comparison tools
-

2 UI/UX Layout Understanding (For Designers)

1. Home Page Layout

Layout Structure (Top to Bottom)

| Navbar |

| Hero Section |

| Headline |
| Subtext |
| [Take Quiz] [Explore Careers] |

| Career Categories (Grid – 4 Cards) |

| Trending Careers (Horizontal Scroll) |

| Footer |

UX Intent

- Clear CTA above the fold
 - Grid for quick scanning
 - Horizontal scroll for mobile-friendly design
 - Minimal cognitive load
-

2. Quiz Page Layout

| Progress Bar |

| Question Card |

| Question Text |

| Option Cards (Clickable Blocks) |

| Back Next |

UX Rules

- One question per screen (reduce overwhelm)
 - Large tappable option cards
 - Visual feedback on selection
 - Animated transitions between questions
-

3. Quiz Result Page Layout

| Your Career Matches |

| Career Match Card (x3) |

| Career Name |

| Match % |

| Short Explanation |

| [View Details] [Compare] |

| Retake Quiz |

UX Intent

- Show Top 3 results only
 - Highlight best match visually
 - Allow comparison immediately
 - Provide explanation (trust building)
-

4. Career Detail Page Layout

| Career Title + Category Tag |

| Overview Section |

| Salary Section (Graph) |

| Skills Required (Tag Style) |

| Risk & Study Info |

| Roadmap Timeline |

| Related Careers |

UX Principles

- Sectioned layout for scannability
 - Graph for salary (visual data)
 - Timeline for roadmap (step clarity)
 - Related careers for exploration loop
-

5. Branch-Specific Page Layout

| Branch Header (CSE / ECE etc.) |

| Career Distribution Chart |

| Core Careers |

IT Switch Careers	
Govt Careers	
Higher Studies	

UX Logic

Students think in branch-first mindset
So structure matches their mental model.

③ Frontend Developer Structure Understanding

Now let's convert UI thinking → Code thinking.

- ◆ **Folder Structure (Production Ready)**

```
src/
  └── components/
      ├── common/
          ├── Button.jsx
          ├── Card.jsx
          ├── Modal.jsx
          ├── ProgressBar.jsx
      └── quiz/
          ├── QuizQuestion.jsx
          ├── QuizOptions.jsx
          └── QuizResultCard.jsx
```

```
|      |
|      +-- career/
|          |      +-- CareerCard.jsx
|          |      +-- CareerDetail.jsx
|          |      +-- SalaryChart.jsx
|          |      +-- RoadmapTimeline.jsx
|
|      |
|      +-- branch/
|          |      +-- BranchCard.jsx
|          |      +-- BranchDetail.jsx
|
|      |
|      +-- pages/
|          |      +-- HomePage.jsx
|          |      +-- QuizPage.jsx
|          |      +-- ResultPage.jsx
|          |      +-- CareerPage.jsx
|          |      +-- ComparePage.jsx
|
|      |
|      +-- context/
|          |      +-- QuizContext.jsx
|          |      +-- CompareContext.jsx
|
|      |
|      +-- data/
```

```
|      └── careers.json  
|      └── quizQuestions.json  
|  
└── utils/  
    ├── quizEngine.js  
    └── scoringLogic.js  
|  
└── App.jsx  
└── main.jsx
```

- ◆ **Component Responsibility (Clear Ownership)**

UI Components

Only responsible for:

- Rendering
- Styling
- Accepting props

Logic Components

Responsible for:

- Quiz scoring
- Filtering
- Recommendation logic

Data Layer

- Static JSON (MVP)
 - Later → Backend API
-

- ◆ **Data Flow Example (Quiz)**

1. User selects option
2. State updates in QuizContext
3. On submit:
 - o quizEngine.js calculates score
 - o Returns ranked careers
4. ResultPage renders sorted careers

Clear separation of concerns.

♦ **Compare Feature Logic**

Global State:

selectedCareers: []

Flow:

- User clicks Compare
 - Career added to state
 - Navigate to ComparePage
 - ComparisonTable renders dynamically
-

4 UX + Dev Alignment Rules

UX Decision

Dev Implementation

Single question per screen

Conditional render

Animated transitions

CSS + Framer Motion

Highlight best match

Conditional styling

Filter panel

Controlled components

Roadmap timeline

Dynamic map loop

5 Scalability Considerations

Future upgrades:

- Backend (Node + MongoDB)
 - Admin panel to add careers
 - AI recommendation engine
 - User login & save results
 - Analytics dashboard
-

💡 Why This Makes You Strong

This shows:

- Product thinking
- UX clarity
- Frontend architecture
- Scalable component design
- Real startup-level documentation

This is NOT just a student project anymore.

If you want next level, I can now give:

- Complete Design System (colors, typography, spacing)
- Responsive grid system (mobile-first structure)
- Advanced state architecture diagram
- How to pitch this project in interviews
- Convert this into a startup-level PRD

Tell me what direction you want to go next.

Convert this into a PRD with clear and detailed info

Excellent

Now we convert everything into a **Professional Product Requirements Document (PRD)**.

This version is structured like something used in:

- Product companies
 - Startups
 - Tech teams
 - Portfolio case studies
-

PRODUCT REQUIREMENTS DOCUMENT (PRD)

Product Name: CareerMap India

Version: 1.0

Document Owner: Product Team

Target Audience: B.Tech Students (India)

1 Product Overview

1.1 Vision

To help B.Tech students in India make informed, structured, and personalized career decisions using guided exploration, intelligent quizzes, and detailed roadmaps.

1.2 Problem Statement

Many B.Tech students:

- Are confused about career options after graduation
- Lack structured comparison between career paths
- Follow trends blindly (e.g., everyone moving to IT)
- Do not understand risk, salary reality, or study requirements
- Have financial constraints affecting higher studies

Existing blogs provide information but lack:

- Personalization
 - Structured comparison
 - Branch-based clarity
 - Decision-making support tools
-

1.3 Solution

CareerMap India provides:

- Career category exploration
 - Branch-specific career pages
 - Personalized career recommendation quiz
 - Detailed structured roadmaps
 - Career comparison tool
-

2 Goals & Objectives

Primary Goals

- Reduce career confusion
- Provide structured decision support
- Improve clarity within 10 minutes of usage
- Enable informed decision-making

Success Metrics (KPIs)

- Quiz completion rate $\geq 70\%$
 - Average session time ≥ 5 minutes
 - 40% users view at least 2 career detail pages
 - 25% use comparison feature
-

3 Target Users

3.1 Primary User Persona

Final Year B.Tech Student

- Age: 20–24
 - Branch: CSE / ECE / Mech / Civil / EEE
 - Confused about career direction
 - Concerned about salary and stability
 - May or may not afford higher studies
-

3.2 Secondary Users

- 2nd/3rd Year Students
- Parents
- College counselors

4 Feature Requirements

4.1 Career Categories

Description

Categorized career groups for easy exploration.

Categories

1. IT & Software
 2. Core Engineering
 3. Government Jobs
 4. Management & Business
 5. Higher Studies & Research
-

Functional Requirements

- Display category cards on homepage
 - Clicking category navigates to filtered career list
 - Each category contains 8–20 careers
 - Filter support within category
-

4.2 Branch-Specific Pages

Description

Dedicated pages for each engineering branch.

Supported Branches (MVP)

- CSE
 - ECE
 - Mechanical
 - Civil
 - EEE
-

Functional Requirements

- Show:
 - Core careers
 - IT switch careers
 - Govt jobs
 - Higher studies
 - Career distribution visualization (chart)
 - Branch-specific recommendations
-

4.3 Career Detail Page

Each career must include:

Mandatory Fields

- Overview
 - Required Skills
 - Average Salary (India)
 - Risk Level (Low/Medium/High)
 - Study Requirement (Yes/No)
 - Work-life balance rating
 - Growth potential
 - Roadmap preview
 - Related careers
-

4.4 Career Roadmap

Structured timeline:

Example (Software Developer):

- 0–3 Months: Basics (DSA, Language)
 - 3–6 Months: Projects
 - 6–9 Months: Advanced concepts
 - 9–12 Months: Interview prep
-

Functional Requirements

- Timeline visualization
- Expandable milestone blocks

- Resource references (optional future)
-

4.5 Career Recommendation Quiz

Purpose

Provide personalized career suggestions.

Quiz Parameters

- Branch (CSE, ECE, Mech, Civil, etc.)
 - Interest (Coding, Management, Govt, Research)
 - Risk tolerance (Low/Medium/High)
 - Salary expectation
 - Study preference (Yes/No)
 - Work-life balance importance
 - Financial condition (Can afford MS/MBA?)
-

Functional Requirements

- One question per screen
 - Progress bar
 - Save answers in state
 - Submit triggers scoring engine
 - Return top 3 recommended careers
 - Show match percentage
-

4.6 Career Comparison Tool

Purpose

Help users compare 2–3 careers side by side.

Comparison Fields

- Salary
- Study duration

- Risk level
 - Stability
 - Growth
 - Work-life balance
-

Functional Requirements

- Add career to compare
 - Max 3 careers
 - Tabular comparison
 - Highlight best option per category
-

5 Non-Functional Requirements

5.1 Performance

- Page load under 3 seconds
 - Smooth transitions
 - Responsive on mobile
-

5.2 Security (Future Phase)

- Secure API endpoints
 - Basic authentication (optional phase 2)
-

5.3 Scalability

- Modular component structure
 - API-ready architecture
 - Expandable career database
-

6 Technical Requirements

Frontend

- React.js
 - Responsive CSS (Tailwind or custom CSS)
 - Chart library (Recharts/Chart.js)
-

Backend (Future)

- Node.js + Express
 - MongoDB
-

State Management

- Context API (MVP)
 - Redux (Phase 2)
-

7 User Flow (High-Level)

1. User lands on homepage
 2. Either:
 - Takes quiz
 - OR
 - Explores categories
 3. Views career details
 4. Compares careers
 5. Reads roadmap
 6. Makes informed decision
-

8 Assumptions

- Users are first-time visitors
 - Users have basic internet literacy
 - Data shown is indicative (not guaranteed)
-

9 Constraints

- Salary data varies by region

- Career outcomes not guaranteed
 - Initial MVP static data-based
-

10 Future Enhancements

- AI-based recommendation engine
 - Save user profile
 - College recommendations
 - Internship tracker
 - Placement statistics integration
 - Admin dashboard
-

11 Risks & Mitigation

Risk	Mitigation
Incorrect salary data	Add disclaimer
Over-simplified quiz	Improve scoring logic
Users skip quiz	Promote CTA strongly
Data outdated	Quarterly update plan

12 MVP Scope (Phase 1)

Included:

- Homepage
- Categories
- 15–20 careers
- Quiz (basic scoring)
- Comparison tool
- Roadmaps
- 5 branches

Excluded:

- Login system
 - AI personalization
 - Backend database
 - Payment features
-

⌚ Product Positioning

CareerMap India is not:

- Just another blog
- Random career article site

It is:

A structured decision-support platform for engineering students.

💼 How You Can Present This in Interview

You can say:

"I designed a structured career decision-support platform for B.Tech students with quiz-based personalization, branch-specific mapping, and comparison tools. I created the full PRD, UI structure, user flow, and modular frontend architecture."

That sounds like product-level thinking.

If you want next level, I can now give:

- Product Roadmap (6-month execution plan)
- Monetization strategy
- Go-to-market strategy for India
- Detailed scoring algorithm logic
- Agile sprint breakdown

Tell me what you want next.

Product Roadmap (6-month execution plan) Monetization strategy Go-to-market strategy for India

Perfect 🔥

Now we move from **product design** → to **startup execution level thinking**.

I'll structure this professionally:

- 1 6-Month Product Roadmap
 - 2 Monetization Strategy
 - 3 Go-To-Market Strategy (India-focused)
-

1 PRODUCT ROADMAP (6-MONTH EXECUTION PLAN)

Product: CareerMap India

Goal: Launch MVP + Validate Market + Prepare Monetization

Month 1 – Research & Foundation

Objectives:

- Validate problem
- Finalize scope
- Prepare design system

Tasks:

- Conduct 30–50 student interviews
- Validate quiz parameters
- Define 20 initial careers
- Finalize PRD
- Create low-fidelity wireframes
- Finalize UI component structure

Deliverables:

- Final PRD
 - Wireframes
 - Confirmed MVP scope
 - Career data spreadsheet
-

Month 2 – UI/UX & Frontend Setup

Objectives:

- Build design system
- Setup project structure

Tasks:

- Design high-fidelity UI (Figma)
- Create React project structure
- Build reusable components:
 - Button
 - Card
 - ProgressBar
 - Tag
- Setup routing

Deliverables:

- Functional homepage
 - Static category pages
 - Base UI framework
-

Month 3 – Core Feature Development

Objectives:

- Implement main product features

Tasks:

- Build quiz engine
- Implement scoring logic
- Build career detail pages
- Create roadmap timeline component
- Add branch-specific pages

Deliverables:

- Quiz working end-to-end
 - 20 career detail pages
 - 5 branch pages
-

Month 4 – Advanced Features + Testing

Objectives:

- Add comparison tool
- Improve UX
- Optimize performance

Tasks:

- Develop comparison feature
- Add filters in category pages
- Optimize mobile responsiveness
- User testing with 50 students
- Fix usability issues

Deliverables:

- Complete MVP
 - Mobile responsive
 - Bug-free beta version
-

 **Month 5 – Beta Launch (Soft Launch)****Objectives:**

- Validate product-market fit

Tasks:

- Launch to:
 - 2–3 engineering colleges
 - Student communities
- Collect feedback
- Track analytics:
 - Quiz completion rate
 - Most viewed careers
 - Drop-off points

Deliverables:

- Usage analytics report
 - Feedback analysis
 - Feature improvement list
-

Month 6 – Growth & Monetization Preparation

Objectives:

- Prepare revenue streams
- Improve engagement

Tasks:

- Add premium roadmap preview (locked content)
- Implement email collection
- Add basic login (optional)
- Improve scoring logic
- Prepare partnerships

Deliverables:

- Monetization-ready version
 - Investor pitch-ready metrics
-

2 MONETIZATION STRATEGY

We must design monetization that aligns with students' financial condition.

Revenue Model 1: Freemium Model

Free:

- Career exploration
- Basic quiz
- Basic roadmap

Premium (₹199–₹499 one-time):

- Detailed roadmap
 - Curated learning resources
 - Interview preparation guide
 - Resume templates (career-specific)
-

Revenue Model 2: Affiliate Revenue

Partner with:

- EdTech platforms (Coursera, Scaler, Udemy)
- MBA/MS consultants
- GATE coaching
- Government exam coaching

Earn commission per signup.

⌚ Revenue Model 3: Sponsored Listings

Example:

- "Top MBA Colleges"
- "Recommended MS Consultants"
- "Best GATE Coaching"

Clearly mark as sponsored.

⌚ Revenue Model 4: College Partnerships (B2B Model)

Offer:

- Career analytics dashboard
- Student interest reports
- Career counseling tool

Charge colleges subscription (₹50,000–₹2L/year).

⌚ Revenue Model 5 (Future)

- AI personalized career coaching
 - Paid 1:1 mentoring sessions
 - Resume review marketplace
-

📈 3 GO-TO-MARKET STRATEGY (INDIA FOCUSED)

India-specific approach is critical.

Phase 1 – Student Community Penetration

Target:

- Tier 2 & Tier 3 engineering colleges
 - Final-year students
 - Placement-focused communities
-

Channels

WhatsApp & Telegram Groups

Engineering groups have 200–1000 students.

Strategy:

- Share free quiz link
 - Position as career clarity tool
-

LinkedIn Content Strategy

Post:

- “Top 5 Career Paths After B.Tech”
- “IT vs Govt Jobs Comparison”
- “How to choose between MBA & MS?”

Drive traffic to site.

Campus Ambassadors

Select 1 student per college.

Offer:

- Certificate
- LinkedIn recommendation
- Small incentive

They promote platform internally.

Phase 2 – Influencer Strategy

Collaborate with:

- Small YouTube channels (10k–50k subs)
- Instagram career pages
- Coding influencers

Offer affiliate share.

Phase 3 – SEO Strategy

Target Keywords:

- “Career options after B.Tech”
- “Best career after mechanical engineering”
- “MBA or MS after engineering”
- “High salary jobs for engineers in India”

Build blog content around these.

Long-term organic traffic strategy.

Phase 4 – College Tie-Ups

Approach:

- Training & Placement Officers
- Offer free career workshop
- Provide demo

Convert into B2B subscription.

Metrics to Track

- Daily Active Users (DAU)
- Quiz completion %
- Average session duration
- Conversion to premium %
- Cost per acquisition (CPA)

Competitive Positioning

Existing Platforms:

- Shiksha
- Careers360
- Collegedunia

Your Differentiator:

- Structured quiz + branch logic
 - Side-by-side comparison
 - Clear roadmap timeline
 - India-focused salary & risk data
-

Strategic Vision

Year 1:

- 50,000+ students
- ₹5–10L revenue

Year 2:

- College partnerships
- AI recommendation engine

Year 3:

- Become India's engineering career guidance platform
-

Why This Is Powerful

You now have:

- PRD
- BRS
- UML
- UI Architecture
- Product Roadmap
- Monetization Model

- GTM Strategy

This is startup-founder-level thinking.

If you want next level, I can now build:

-  Investor Pitch Deck structure
-  Detailed scoring algorithm
-  Financial projection model
-  Execution team structure
-  SWOT analysis

Tell me your next move. 