

Scenario - Freelancer Marketplace Platform

You're building a freelancer marketplace platform where companies can hire freelancers for projects. The platform needs to handle complex organizational structures, flexible billing models, and intelligent matching between clients and freelancers.

Please choose one of the below three features and build that marketplace prototype. As part of the exercise, please tell us why you chose that feature.

What to Build

1. Permission & Group Management Solution

Create a flexible permission system that supports organizational complexity.

Requirements to consider

- Companies can create groups/teams with multiple members
- Group admins can define custom roles and permissions within their organization
- Support hierarchical permissions (company admin, team lead, team member, etc.)
- Permissions control access to features like: posting projects, managing budgets, viewing team analytics, hiring freelancers
- System must be easily extensible for new permissions and roles

2. Multi-Format Billing Solution

Implement a billing system that handles diverse payment models for different organizational needs.

Requirements to consider

- Support multiple billing formats:
 - **Pay-as-you-go:** Charge per project posted or freelancer hired
 - **Prepaid credits:** Companies buy credit packages for platform usage
 - **Seat-based:** Monthly/annual subscription per team member
- Credit limits can be set per group member by group admins
- Integration with permission system (billing status affects feature access)
- Handle overage scenarios and billing notifications

3. Marketplace Search & Matching Engine

Build an intelligent search system for finding and filtering freelancers.

Requirements to consider

- Multi-criteria search with dynamic filtering:
 - Skills and expertise categories

- Freelancer ratings and reviews
- Response time and availability
- Location and timezone
- Hourly rate ranges
- Portfolio quality and past work
- Search results should respect group permissions (some team members see different freelancer pools)

Technical Stack

- NestJS (TypeScript)/Next.js (React)/PostgreSQL
- Authentication
 - Your choice of implementation
- Additional tools
 - Use what makes sense for your solution

What to Submit

- GitHub repo with your code
- Deployed link (Vercel, Railway, etc.)
- A README with:
 - Setup instructions
 - Architecture overview
 - Notes on trade-offs or areas for improvement
- Explanation as to why you chose the feature selected

What to Expect Next

Our team will review the work and if the team requires an explanation, you'll walk us through your work.