internStudio

BACKEND DEVELOPMENT INTERNSHIP

TASK GUIDE

Build real-world backend systems.
Develop APIs. Solve real problems.

SIX ADVANCED BACKEND SYSTEMS — CHOOSE ANY TWO

01 **Multi-Tenant Project Management API**
02 **E-Commerce Backend System**
03 **Event Booking & Ticketing API**
04 **Finance & Expense Management API**
05 **Food Delivery Backend**
06 **Job & Application Management Platform**

INTERNSHIP	REQUIRED TASKS	FOCUS
Backend Development	Complete any 2	APIs · Databases · Authentication · Backend Architecture

IMPORTANT INSTRUCTIONS

Your Internship Tasks

Select and complete **any 2 projects** from the task list. Each project should be functional, properly structured, secure, documented and suitable for showcasing in your portfolio during internships, placements and interviews. **Build them in any backend technology or framework you prefer** — the stack is your choice; the architecture and quality are what count.

Choose	Plan	Develop	Test	Document	Deploy	Submit

BEFORE YOU SUBMIT

- [ ] **Complete any 2 projects** from the list
- [ ] Ensure all **major functionality** works
- [ ] Use proper **backend architecture**
- [ ] Implement **validation and error handling**
- [ ] Secure **sensitive routes and data**
- [ ] Upload the project to **GitHub**
- [ ] Add a **README** with setup steps and API docs
- [ ] Add a **live deployment** where applicable
- [ ] Submit details via the **submission form**
- [ ] Share your work on **LinkedIn** as required

Submit your completed work here

Once both projects are built, documented, deployed and pushed to GitHub, submit your project details through the official submission form.

https://forms.gle/vxL5JE8cvRXXfDBv8

> **A note on quality.** These are backend systems, not CRUD exercises. You should be able to explain your schema design, your authorization model and your error-handling strategy in an interview — that conversation matters more than the number of endpoints you ship.

Tech stack: your choice. Build these in any backend language, framework or database you prefer.

01 Multi-Tenant Project Management API

ADVANCED

Build a backend for a project management platform where multiple teams manage projects, members and tasks while keeping their data isolated.

CORE REQUIREMENTS

- User registration and authentication
- Team / workspace creation
- Role-based access
- Project management
- Task management
- Task status and priority
- Team member management
- Project-specific permissions
- Search and filtering
- Activity history
- Database relationships
- Input validation
- Error handling
- Secure protected APIs

Important: A user from one workspace must not be able to access another workspace's private data.

Deliverables: GitHub Repository · API Documentation · Database Schema · README · Live API deployment where applicable

02 E-Commerce Backend System

ADVANCED

Build the backend for a complete e-commerce platform handling products, users, carts, orders and inventory, with correct relationships between them.

CORE REQUIREMENTS

- User authentication
- Product management
- Categories
- Product variants
- Search and filtering
- Cart management
- Wishlist
- Validation and error handling
- Order creation
- Order status management
- Inventory tracking
- Address management
- Price calculations
- Order history
- Admin / product management APIs

Deliverables: GitHub Repository · API Documentation · Database Schema · README · Live API deployment where applicable

03 Event Booking & Ticketing API

ADVANCED

Build the backend for an event discovery and ticket booking platform.

CORE REQUIREMENTS

- User authentication
- Event creation and management
- Event categories
- Venue management
- Event search and filtering
- Ticket types
- User-specific booking access
- Ticket availability
- Booking system
- Booking history
- Booking confirmation
- Cancellation handling
- Capacity management
- Admin event management

Important: Prevent users from booking unavailable tickets or exceeding event capacity.

Deliverables: GitHub Repository · API Documentation · Database Schema · README · Live API deployment where applicable

Tech stack: your choice. Build these in any backend language, framework or database you prefer.

04 Finance & Expense Management API

ADVANCED

Build a backend that lets users securely manage personal income, expenses and financial categories.

CORE REQUIREMENTS

- User authentication
- Income management
- Expense management
- Categories
- Transactions
- Monthly summaries
- Budget management
- Spending statistics
- Date-based filtering
- Search
- Transaction history
- User-specific data isolation
- Validation
- Secure authorization

Important: A user must only be able to access their own financial information.

Deliverables: GitHub Repository · API Documentation · Database Schema · README · Live API deployment where applicable

05 Food Delivery Backend

ADVANCED

Build the backend for a food delivery platform connecting customers, restaurants and delivery operations.

CORE REQUIREMENTS

- User authentication
- Restaurant management
- Menu management
- Food categories
- Cart
- Order creation
- Order status
- Restaurant order management
- Customer order history
- Address management
- Delivery status
- Search and filtering
- Role-based access

Roles: Customer · Restaurant · Delivery Partner · Admin — each role should only reach the operations relevant to it.

Deliverables: GitHub Repository · API Documentation · Database Schema · README · Live API deployment where applicable

06 Job & Application Management Platform

ADVANCED

Build a backend for a job platform where companies publish roles and candidates discover and apply for opportunities.

CORE REQUIREMENTS

- User authentication
- Candidate profiles
- Company profiles
- Job creation and editing
- Job search
- Filters
- Job applications
- Application status
- Candidate application history
- Company applicant management
- Resume / document reference handling
- Role-based authorization
- Search and pagination
- Validation and secure APIs

Roles: Candidate · Recruiter/Company · Admin — ensure each role can only access authorized information.

Deliverables: GitHub Repository · API Documentation · Database Schema · README · Live API deployment where applicable

APPLIES TO EVERY PROJECT

API Requirements

Whichever two projects you choose, your submission is expected to demonstrate all of the following.

Authentication

Use secure authentication for all protected functionality.

Authorization

Users should only access resources they are permitted to access.

Validation

Validate incoming request data before it reaches your business logic.

Error Handling

Return clear, appropriate and consistent API errors.

Database

Provide a clear schema structure with meaningful relationships.

Deployment

Deploy the API live where applicable, and document the base URL.

API Documentation

Document each of the following for every endpoint you expose:

- Endpoints
- HTTP methods
- Request parameters
- Request body
- Authentication requirements
- Response format
- Error responses

Security

At minimum, consider each of these:

- Password hashing
- Protected routes
- Input validation
- Authorization
- Rate limiting where appropriate
- Secure handling of sensitive data

PROJECT SELECTION

Choose ANY 2 projects from the six available tasks.

You are encouraged to pick projects that expose you to different backend concepts — authentication, database design, API development, authorization, business logic, security and data relationships.

FINAL STEP

Ready to Submit?

Run through this checklist before you send your work in.

- [ ] Completed 2 projects
- [ ] All major API functionality works
- [ ] Authentication implemented where required
- [ ] Authorization implemented where required
- [ ] Input validation added
- [ ] Error handling implemented
- [ ] LinkedIn post completed where required
- [ ] Database properly structured
- [ ] GitHub repository created
- [ ] README added
- [ ] API documentation added
- [ ] Live API deployed where applicable
- [ ] Submission form completed

Submission Form

Submit your project details, repository links, API docs and live URLs here.

https://forms.gle/vxL5JE8cvRXXfDBv8

BUILD SOMETHING WORTH SHOWING.

Your internship project is more than a task — it's an opportunity to demonstrate how you design, build and secure real backend systems.

internStudio

LEARN · BUILD · GROW

www.internstudio.in

team.internstudio@gmail.com