
Project 1: PHP (Raw, No Framework) Project

Title: Mini Support Ticketing System (Backend Only)

Objective

Build a **RESTful PHP backend (without a framework)** for a minimal ticketing system, including users, departments, and support tickets.

Tech Requirements

- **No frameworks** (vanilla PHP, PDO for DB access)
- **OOP** and **MVC-like** folder structure
- Use **SQLite** or **MySQL**
- Route requests manually (e.g., via .htaccess or index.php front controller)
- Responses should be in **JSON**

Functional Requirements

1. Users

- Fields: id, name, email, password_hash, role (admin / agent)
- Implement:
 - Registration
 - Login (return token – store in memory or file-based store for simplicity)
 - Logout (delete token)
 - Token-based auth middleware

2. Departments

- Fields: id, name
- Only admins can create/update/delete departments

3. Tickets

- Fields: id, title, description, status (open, in_progress, closed), user_id, department_id, created_at
- Functionality:
 - Submit ticket (authenticated users only)
 - Agents can assign themselves to tickets
 - Change status
 - Add notes (separate ticket_notes table with user_id, note, ticket_id, created_at)

Bonus Features

1. Add rate limiting for ticket submission
2. Add file upload support for ticket attachments
3. Write minimal unit tests (in a custom test file)

Deliverables

- PHP files (organized in folders: controllers/, models/, routes/, helpers/)
- SQL schema



- Postman Collection or API doc
- Sample data seed script
- Send the GitHub link to sadman@skifftech.com before the deadline.
- **Do not use AI-generated code.** A manual review will be conducted.

Evaluation Criteria

Area	Weight
OOP/MVC Structure	25%
Security (auth, SQLi)	20%
API Design	20%
Functionality	25%
Bonus Features	10%

Project 2: Laravel Project

Title: Freelance Time Tracker API

Objective

Build an API in **Laravel** that allows freelancers to log and manage their work time across clients and projects.

Tech Requirements

- Laravel 10+
- Sanctum for authentication
- Eloquent ORM
- Factories, Seeders
- Tests (optional but preferred)

Functional Requirements

1. Users (Freelancers)

- Register/Login/Logout via Sanctum
- Basic profile info (name, email, password)

2. Clients

- Freelancers can manage their own clients
- Fields: name, email, contact_person

3. Projects

- Belongs to a client
- Fields: title, description, client_id, status (active, completed), deadline

4. Time Logs

- Fields: project_id, start_time, end_time, description, hours (calculated)



- Freelancers can:
 - Start/end a time log (auto-calculate hours)
 - Add/edit manual entries
 - View logs per day/week

Filtering & Reports

- Get total hours logged:
 - Per project
 - Per day
 - Per client

API example:

GET /api/report?client_id=1&from=2024-01-01&to=2024-01-07

Bonus Features

1. PDF export of time logs
2. Tags for logs (billable, non-billable)
3. Notification (email) when 8+ hours logged in a day

Deliverables

- Laravel app in Git repo
- Postman collection
- README with setup steps and DB structure
- Seeders for at least:
 - 1 user
 - 2 clients
 - 2 projects
 - 5+ time logs
- Send the GitHub link to sadman@skifftech.com before the deadline.
- **Do not use AI-generated code.** A manual review will be conducted.

Evaluation Criteria

Area	Weight
Laravel Best Practices	25%
Data Modeling (Eloquent)	20%
API Design and Validation	25%
Filtering & Reporting	20%
Bonus Features	10%



Notes and condition:

Ownership and Copyright

- By submitting your solution to this task, you agree that **SKIFF Technologies** retains full copyright and ownership of the submitted work.
- You acknowledge that your submission may be used internally for evaluation, research, or process improvement purposes.

Compensation

- No payment will be provided for completing the task.
- This task is solely for the purpose of candidate evaluation and is a voluntary part of the recruitment process.
- Submission of the task does not constitute any form of employment, contract, or service agreement with Skiff Technologies.

Confidentiality

- Candidates must not share, distribute, or reproduce the task content or their solutions publicly or with third parties.
- The task should be treated as confidential material provided for evaluation purposes only.

Original Work Requirement

- Candidates must ensure that all submitted work is entirely their own.
- Any use of third-party code must be properly cited, and plagiarism or submission of someone else's work will lead to disqualification from the hiring process.

