

IT 2001 - Web Programming

Project Requirements, Milestones and Deadlines

PROJECT

Introduction

Students will **individually** develop a **fully functional, responsive**, and **deployed single-page web application** with a backend implemented using the **FlightPHP framework** and **MySQL**. The project will consist of **5 milestones**, each with detailed requirements and deadlines. The application must include **5 database entities**, **support CRUD operations**, and be documented within a **README file**.

Project Requirements

1. The project has to be a web application implemented by using PHP, JS and MySQL technology stack. Libraries like bootstrap, highcharts, datatable and similar are very desirable in the project.
2. Mixing backend and frontend code is not allowed. REST backend should be pure PHP code while frontend should be just HTML, CSS and JS.
3. Usage of FlightPHP framework for REST API is a must.
4. Usage of PHP PDO is required for accessing the database layer.
5. Project API must be documented using Open API documentation.
6. Application must have AJAX requests implemented.
7. Every student has to create a github repository where they should implement their project. The GitHub repository will be used to monitor students' activity and progress of work.
8. Application has to be mobile friendly.
9. Application must be a single page application.
10. The project should have at least 5 different entities (ex. Users, Products, Orders, Categories, Reviews) with basic CRUD operations implemented.
11. JWT token is a must (JSON Web Tokens for authentication and authorization).
12. The project should be hosted and it should be publicly available through domain. Github education pack can be used for this purpose. <https://education.github.com/pack>
Any other hosting providers allowed and desirable.

13. Application must support at least 2 different roles.
14. Usage of HTML templates is recommended. Bootstrap based templates are also recommended.
15. All additional components and libraries are desirable.

Milestones

Milestone 1: Initial Setup and Frontend Development

Deadline: End of Week 3 (October 19, 2025) - latest by 8:00 am (morning) on Monday (October 20)

Requirements:

Submit your topic and github repository [here](#).

1. (1pts) Project Structure:

- Set up the project folder structure with frontend and backend directories.
- Prepare basic folders for backend:
 - routes
 - services
 - dao
- Prepare basic folders for frontend:
 - css files,
 - js files,
 - HTML templates - views,
 - static assets

2. (3pts) Static Frontend:

- Prepare all the pages and components needed for the frontend part of the application.
- These pages and components should be static and function as a single-page application (SPA), meaning all navigation should occur without full page reloads.
 - There is no need to connect them with a backend at this stage.

- Each view/page must be in its own file. Combining all HTML code into one file is not allowed.
- Common JS and CSS files must be included only once, when the first page loads.
- You may use HTML/Bootstrap templates to streamline development; there is no need to code everything from scratch.
 - However, ensure that the templates are customized to fit the specific needs of your project.
- The following pages must be included, along with any others relevant to the project:
 - Register/Login pages: For user authentication interfaces.
 - Other necessary pages: Based on the requirements of the project (e.g., Dashboard, Entities, Profile, etc.).
- Use a consistent theme and responsive design techniques.

3. (1pts) Database Schema (Planning Only):

- Identify at least **5 entities** for the application (e.g., Users, Products, Orders, Categories, Reviews).
- Submit a draft Entity-Relationship Diagram (ERD) showing relationships among these entities.

Deliverables:

- Git repository with the initialized project structure.
 - Single page application with all frontend components of the application
 - Draft ERD for at least 5 entities.
-

Milestone 2: Backend Setup and CRUD Operations for Initial Entities

Deadline: End of Week 5 (November 2, 2025) - latest by 8:00 am (morning) on Monday (November 3)

Requirements:

1. (1pts) Database Creation:

- Design and document a relational database schema in MySQL with at least 5 entities.
- Create a database according to the prepared design.

2. (4pts) DAO Layer:

- Implement the **DAO Layer** to interact with the database for at least five entities.
- Provide CRUD functionality for these entities:
 - Create (POST).
 - Read (GET).
 - Update (PUT/PATCH).
 - Delete (DELETE).

Deliverables:

- MySQL database .sql file with at least five entities created.
 - DAO classes for all entities.
-

Milestone 3: Full CRUD Implementation for All Entities & Open API

Deadline: End of Week 7 (November 16, 2025) - latest by 8:00 am (morning) on Monday (November 17)

Requirements:

- 1. (2pts) Business Logic Implementation:**
 - Create services for managing each database entity.
 - Implement application-specific logic and constraints (e.g., validation rules).
 - Ensure services encapsulate reusable logic and are modular.
- 2. (1pts) Presentation Layer:**
 - Implement the presentation layer to render dynamic content using FlightPHP.
- 3. (2pts) OpenAPI Documentation:**
 - Document available API endpoints using the OpenAPI standard.
 - Generate a visual API documentation page (Swagger).

Deliverables:

- CRUD API endpoints for all entities.
 - Backend documentation using OpenAPI standard.
-

Milestone 4: Middleware, Authentication, and Authorization

Deadline: End of Week 11 (December 14, 2025) - latest by 8:00 am (morning) on Monday (December 15)

In this milestone you can start to connect the frontend and backend. Application should be fully connected and functional by the end of milestone 5.

Requirements:

1. (1pts) Authentication and Middleware:

- Add middleware to handle:
 - Request validation.
 - Error handling (optional).
 - Logging (basic - optional).
- Implement user authentication using FlightPHP and middleware.
- Allow users to register and log in.
- Hash passwords for security.

2. (1pts) Authorization:

- Implement role-based access control:
 - Admins can perform CRUD operations for all entities.
 - Regular users have restricted access (e.g., view-only for certain entities).

3. (3pts) Frontend Updates:

- Add dynamic features for authenticated users (e.g., personalized dashboard, admin panel).
- Ensure UI components reflect role-based access (e.g., buttons visible only to admins).
- Frontend fully connected with backend

Deliverables:

- Middleware for authentication and role-based access control.
- CRUD operations with restricted access based on roles.
- Authenticated user actions and role-based access on the frontend side.
- Fully connected frontend and backend.

Milestone 5: Final Deployment, API Documentation, and CI/CD

Deadline: End of Week 13 (December 28, 2025) latest by 8:00 am (morning) on Monday (December 29)

Requirements:

1. **(2pts) Frontend MVC Implementation:**
 - Services files on frontend
2. **(1pts) Front End Validations and Security Enhancements:**
 - Add client-side and server-side form validations (e.g., required fields, email validation).
3. **(2pts) Deployment:**
 - Set up and deploy the web application on a hosting platform (e.g., Heroku, DigitalOcean or AWS).
 - Ensure the database is hosted and connected to the deployed backend.

Deliverables:

- Deployed web application with a live link.
- Link of deployed web application should be added to README file.
- Git repository with final code.

How to submit a milestone?

1. Each milestone should be implemented on a separate branch (milestone1, milestone2, milestone3, milestone4, milestone5).
2. All milestone related code should be added to the milestone branch
3. Before the deadline PR should be created and user **ibu-web-programming2001** should be added as reviewer.
4. Late submissions will not be accepted and a given milestone will be graded with 0.
5. Once PR is approved, the reviewer will merge the code to the main branch and you can create a branch for the next milestone and continue with your work.

- Do not merge the milestone branch into the main branch. Doing so will prevent the reviewer from verifying your submission.
- 6. The first four milestones will be defended after the deadline of the fourth milestone (weeks 12 and 13). Only milestones submitted on time will be eligible for defense.
- 7. The fifth milestone will be defended after the deadline of milestone five. Only a milestone submitted on time will be eligible for defense.