

# TEAM CONTRACT

## 1. Team Roles and Responsibilities:

Roles	Responsibility	Primary task
Project Manager	Hoàng Minh Giang	Ensures the project is delivered on time and meets quality standards by coordinating people, processes, and tools.
UX/ UI design	Võ Trung Hiếu	Design user interfaces (UI) and user experiences (UX), ensuring the application is user-friendly, intuitive, and visually consistent.
Frontend developer	Huỳnh Lê Duy Khánh	Build the mobile user interface, handle client-side logic, and ensure a smooth user experience
Backend developer	Lưu Vĩnh Phát	Business logic processing, authentication, API development
Backend developer	Đoàn Lê Gia Bảo	Error handling, database management

## 2. Communication Plan:

- Communication: Discord
- Task management: Jira
- Source code: Github
- Meeting frequency: At least once a week

All team members must frequently check the group communication channel, a message from the project manager should be responded within 10 hours.

## 3. Work Schedule and Deadlines:

### 3.1. Project Milestones

Milestone	Date	Goal
1	07/06/2025	Finish the design for UX/UI
2	19/07/2025	Complete setup and foundational development
3	02/08/2025	Complete primary features and prepare for integration/testing
4	09/08/2025	Complete all features and final testing
5	16/08/2025	Prepare for release

### 3.2. Deadlines and contingency plan

Deadlines: Will be discussed when a task is given.

In case a member's task is missed or delayed, all team members will join forces to finish that task before moving on to their next task.

## 4. Code and Documentation Standards:

### 4.1. Coding standards

Coding conventions:

Use lowerCamelCase for variable, function, parameter names.

Use UpperCamelCase for class names.

Try to separate and reuse modules.

File naming:

Use snake\_case when naming files, folders.

Testing:

Use a test folder to test.

Github workflow:

Each member is required to clone the source code repository from the Github link above.

For each feature being worked on, create a feature/(task names) for it.

To submit the code, members must create a pull request for other members to review and test. If it does not match the requirement above, the pull request will be rejected.

### 4.2. Documentation standards

Code documentation should be concise, clear, and focused on essential information.

## 5. Accountability and Performance:

### 5.1. Criteria for measuring quality of works

Team members can evaluate their works according to the table below

Percentages	Correctness	Performance	Scalability	Documentation
100	Meet all requirements	Code is fast and optimal	Modular, stateless, horizontally scalable, tested for large loads, optimal DB use.	Complete, well-structured, clear, up-to-date, and developer-friendly.
80	Slightly off requirements	Code is acceptable	Good architecture and performance, but minor bottlenecks exist under heavy load.	Mostly complete and understandable, but may lack depth or full consistency.
60	Off requirements	Code is slow	Basic structure works for small-scale, but lacks optimization and breaks under load.	Minimal or outdated; may miss important details and be hard to follow.

## 5.2. Underperformance handling

Underperformance and lack of participation will be identified through regular monitoring using project tools and team meetings. Concerns will be addressed early through private, respectful conversations focused on specific behaviors and their impact. If the issues continue, a structured improvement plan will be developed collaboratively, outlining clear goals, deadlines, and necessary support. Progress will be monitored consistently, with recognition for improvement or escalation if needed.

Consequences for not adhering to the team contract: A team member will have 3 times to make a mistake before being having their performance grading lower by the Product Manager.

## 6. Decision-Making Process:

All members are encouraged to elaborate on their ideas. Each idea will undergo a majority vote to determine whether it should move forward. The Product Manager's vote will carry the weight of three votes during this initial round.

If disagreement persists during implementation, a proposal should be submitted to the Product Manager to initiate further discussion and resolution.

## 7. Conflict Resolution:

If two approved ideas later come into conflict, the members responsible for each will present the pros and cons of their respective proposals. A second majority vote will then be held to resolve the conflict, with the Product Manager's vote now counting as one vote.

## 8. Review and Update Process:

This contract will only be updated after being presented during a meeting in which all members participated.

Any member may propose an update, provided it includes specific content and a clear rationale.

The proposal will be subject to majority voting. The Product Manager's vote will carry the weight of three votes. All proposals will be documented for future usage.