

Project Evaluation

Project Evaluation Template

Intern Name: Mahmoud Miqdad

Evaluators: Waleed Jubeh and Nelson Ortega

Evaluation Date: 20 Aug 2025

Scoring Guide

- **0-1:** Needs significant improvement
- **2-3:** Satisfactory, meets basic expectations
- **4:** Good, above-average performance
- **5:** Excellent, outstanding performance

1. Proficiency

- **Readme File:** [0-5]: 5

Comments: Architecture image has to be updated to reflect CA.

- **Git Usage:** [0-5]: 4

Comments: Very good commit messages, commits size are good, didn't use branches because no reviewer.

- **Documentation:** [0-5]: 4

Comments: The README file provides good documentation, and the controllers are documented as well, but the functions lack proper documentation.

2. Clean Coding: [0-5]:

- **Code Clarity and Simplicity:** [0-5]: 4

Comments: He applied CQRS in a excellent way, the function names are very good. Used logging when needed, Applied SOLID as much as he can, specially L and D. Used generic repository pattern.

- **Adherence to Coding Standards:** [0-5]: 4

3. Communication Skills [English]

- **Clarity:** [0-5]: 3

Comments: Switched to Arabic from time to time, but he was able to handle the conversation in the most time. His thoughts were clear. Sometimes he don't understand the questions.

- **Conciseness:** [0-5]: 3

Comments: He should be more concise.

4. Understanding of the Used Technologies

- **Depth of Knowledge:** [0-5]: 3

Comments: He understands the concepts, and is able to explain them, but he doesn't have really clear on why and when it is good to apply them, so he needs to work more on identifying their use cases or he can over engineer too much

He has to work more on understanding the JWT properties like audience and issuer.

- **Application** [0-5]: 4

Comments: He has good understanding of the application functionalities and libraries.

5. Unit Testing

- **Coverage:** [0-5]: 4

Comments: He has unit and integration tests for most of the files.

- **Quality of Tests:** [0-5]: 4

Comments: The tests are very good and written in a correct way, there are some code duplication that we point him to.

6. Database Design: [0-5]: 4

Comments: We didn't dive deeply into the database, but from what we reviewed, the relationships look good and he applied the necessary constraints. He also used indexes and prepared migration files.

7. API Design

- **RESTful Conformity/Best Practices:** [0-5]: 4

Comments: Applies **RESTful** in an excellent way. He has a class that hides the status code usages and adds extra level in the payload.

8. Concurrency Handling: [0-5]: 3

Comments: He handled the Reserve case using transaction, used transaction in many places. He didn't use `Task.WhenAll` where possible

9. Pagination: [0-5]: 5

Comments: Excellent implementation

10. Error Handling and Logging

- **Error Handling:** [0-5]: 4

Comments: Very good error handling, he should invest more in the error exceptions.

- **Logging Practices:** [0-5]: 5

Comments: Used logging where needed.

11. Bonuses (Docker, Deployment, Third-party Payment Integration)

- **Dockerization:** [0-5]: 3

Comments: He has dockerized the project, but we were not able to spin the docker because we need env variables.

- **Deployment Practices:** [0-5]: 0

Comments: Its not deployed.

- **Third-party Payment Integration:** [0-5]: 3

Comments: He used Stripe, but we didn't test it or discuss it.

12. Project Management

- **Use of Tools (e.g., Jira):** [0-5]: 0

Comments: he didn't use any tool

Overall Comments (Feel free to point any point that the evaluation doesn't cover):

Overall, this is a solid project with good engineering practices. He should continue improving his understanding of architecture and its appropriate use cases. The RESTful design is excellent. His English skills are generally good, but there is room for improvement.

Overall score: 4