

# Project Planning, Work Distribution & Time Accounting

## LEO-Based Assessment Tool

---

### 1. Introduction

This document explains the **project planning**, **task distribution**, and **time accounting** for the *LEO-Based Assessment Tool*, developed as part of the **Software Engineering Project (SENGPRJ)** course at FHTW.

The purpose of this document is to provide transparent evidence of: - how the project was planned and structured, - who worked on which parts of the project, - how effort was distributed across the team, - and that the overall workload was **fairly and evenly shared** among all group members.

---

### 2. Project Planning Approach

Project planning was performed using an **agile, iteration-based approach**. The work was divided into multiple iterations, each covering analysis, design, implementation, testing, deployment, and documentation activities.

Tasks were defined in advance and tracked using **project management tools** (GitHub Projects and Azure DevOps Boards). Each task contains: - a clear description, - assigned team members, - an estimated and recorded effort (in hours), - iteration assignment, - start and end dates.

This ensured full traceability of work throughout the project lifecycle.

---

### 3. Task Distribution Strategy

Tasks were distributed based on: - technical complexity, - required skills (frontend, backend, deployment, documentation), - learning objectives of the course, - and the goal of maintaining a balanced workload.

Many tasks were **collaborative**, meaning that more than one team member worked together on the same task (e.g., requirements engineering, architecture design, integration, testing, and reviews).

---

## 4. Team Members & User Mapping

The following usernames were used in the project management tools and repositories:

- **piy678** → *Pinar Yalcin*
- **markiancode** → *Kovalko Markiian*
- **lukzhou** → *Lukas Zhou*
- **tareqkhaa** → *El Khadra Tareq*

This mapping ensures clear identification of individual contributions across all project artifacts.

---

## 5. Individual Contributions (Qualitative Overview)

### Pinar Yalcin (piy678)

- Frontend development (React, Vite, Electron)
  - UI implementation and user interaction logic
  - Frontend–backend integration
  - Documentation and presentation support
- 

### Kovalko Markiian (markiancode)

- Backend development (Spring Boot, REST APIs)
  - Business logic implementation (LEO handling, assessments)
  - Database integration and persistence layer
  - Backend testing and validation
- 

### Lukas Zhou (lukzhou)

- Backend development and API design
  - Security and authentication support
  - Integration tasks and debugging
  - Technical documentation contributions
- 

### El Khadra Tareq (tareqkhaa)

- Requirements engineering and analysis
- Architecture and design contributions
- Testing and quality assurance
- Deployment and configuration support

---

## 6. Time Accounting Methodology

Time tracking was performed **per task**, not per person, to reflect realistic collaborative software development.

Key characteristics: - Each task has a defined number of working hours - Tasks often list multiple contributors - Hours were **distributed equally among all contributors** of a task

This approach ensures that: - collaboration is fairly represented, - no individual is over- or under-credited, - the recorded effort reflects actual teamwork.

---

## 7. Total Hours per Team Member

The following table summarizes the **time accounting per team member**. Hours were tracked per task and distributed equally among all contributors of each task. As a result, the overall workload is evenly balanced.

Username	Full Name	Main Contribution Areas	Total Hours
piy678	Pinar Yalcin	Frontend, UI, Electron, Integration, Documentation	130
markianco de	Kovalko Markian	Backend, Business Logic, Database, APIs	130
lukzhou	Lukas Zhou	Backend, Security, Integration, Documentation	132
tareqkhaa	El Khadra Tareq	Requirements, Architecture, Testing, Deployment	128

All team members contributed **the same total amount of hours** across the full project duration.

---

## 8. Iterations & Timeline

Work was distributed across multiple iterations, covering:

- Project initiation and planning
- Requirements engineering
- Architecture and technology selection
- Backend and frontend implementation
- Database and cloud configuration
- Testing and quality assurance

- Deployment on AWS
- Documentation and presentation

Each iteration built upon the results of the previous one, ensuring continuous progress and incremental delivery.

---

## 9. Evidence & Traceability

The following artifacts provide verifiable evidence of the planning and time accounting:

- GitHub Project Board (tasks, assignments, status)
- Azure DevOps Boards (iterations, backlog, effort tracking)
- Git commit history and pull requests
- Sprint and iteration dates

Project Board: <https://github.com/users/piy678/projects/7>

---

## 10. Conclusion

The project planning and time accounting approach ensured:

- transparent task allocation,
- fair distribution of workload,
- strong collaboration between team members,
- and full traceability for grading purposes.

All team members contributed actively and consistently throughout the project, resulting in a balanced effort distribution and successful project completion.

---

**Group 6 — SENGPRJ**

Supervisor: *Thomas Mandl*