Workshop
con Intelligenze Artificiali

Indice

1 GitHub

1.1 Version Control System

First of all, Git is a powerful tool called a **version control system**. It helps developers work together on the same project without overwriting each other's work.

Imagine two people changing the same piece of code in different ways — this can create a conflict. Git allows us to safely combine these changes through a process called **merging**.

To make teamwork smoother, we should always work on separate **branches**. Each new task gets its own branch. Once the task is done, we open a **pull request** (or merge directly) to bring the changes into the main code.

This method is very effective because:
- We can save our progress online anytime.
- It reduces the risk of conflicts.
- It keeps the main code stable.

When conflicts do happen, here is a simple rule: 👉 **The person who created the branch resolves the conflict.** If they are unsure, we solve it together. This keeps responsibility clear and teamwork strong.

1.2 Issue Tracking System

GitHub includes a useful **Issue Tracking System**. An **issue** is like a small task or problem that needs attention. We can use issues to:
- Assign work to each other.
- Keep track of what needs to be done.

- Share ideas clearly.

Anyone can create issues at any time. During our weekly meeting (with Jakob or just between us), we will review all new issues and ask:

- Does everyone understand the issue?
- Should we solve it next week?
- Who will take care of it?
- How long will it take?

For an issue to be clear, it must include:

- The project it belongs to.
- A clear description.
- An estimated time to complete.
- If a task takes more than 3 hours, we split it into smaller pieces. This makes planning easier and progress clearer.

1.3 Discussions

Every GitHub repository can have **Discussions**. We can use them to:

- Share useful resources.
- Talk about future ideas.
- Post solutions to common problems.

Think of Discussions as a team blog. We can start new topics, write updates, and keep everything organized. Compared to emails, Discussions are:

- More open and easy to find.
- Grouped by topic.
- Helpful for new team members — they can read past conversations and quickly get up to speed.

1.4 Google Drive Strategy

In a past project, we used GitHub like a **shared drive** — and it worked very well.

We can create a repository called `Notes`. In it:

- Each person has their own branch (named after themselves).
- We share general materials that don't belong to a specific project.
- The `README.md` file explains how we organize the content.

Everyone can manage their own branch freely. This gives personal space while keeping everything in one place.

2 Meetings

We will meet with Jakob once a week, and we may have short team meetings during the week. To make the most of this time, meetings should be focused, short (less than 15 minutes), and follow a clear structure:

1) **Brainstorming**: Share what each of us has worked on this week.
2) **Problems**: Talk about difficulties and how to solve them.
3) **Task Definition**: Review new issues, clarify them, and set priorities.
4) **Task Assignment**: Decide who will work on which task next week.

Additionally, once a week we can have a short **team reflection** meeting to discuss:

- Communication issues.
- Workflow problems (e.g., tasks taking too long).
- Ways to improve how we work together.

3 Reports

Should we write a report after every meeting with Jakob? How often do we need formal reports?

Actually, we might not need to write full reports every time. Instead, we can:

- Use **Discussions** to record important decisions.
- Use **Issues** to track who is doing what.

This way, progress and decisions are always visible, up to date, and easy to find — without extra paperwork.