Week 2
Version Control System & Kanban Board

Teaching Team of Basic Programming 2023

Learning Outcome

After finishing this topic, students:

- Must have a good understanding on version control, including how it works and why it is needed for developing program.
- Must be able to understand the fundamentals of kanban, including workflow visualization,work limitation, and continuous improvement.

Introduction

VS

Flashdisk

Repository

A repository allows developers team to **work together** in a structured manner, **track code changes**, and **manage versions** of various project components.

What is Repository?

A repository in a software development is a **digital storage** that is used to **manage source code**, **configuration files**, **documents**, and **other resources** related to a project.

The Main Component of the Repository

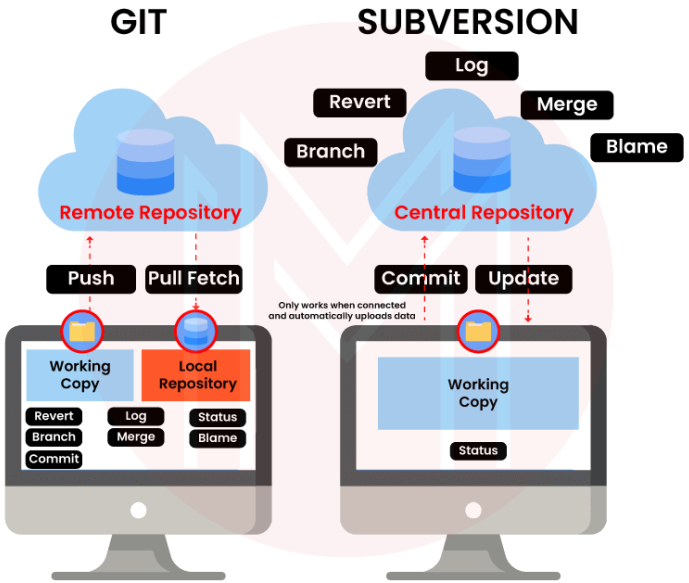
- **Source Code:** files that contain **instructions that will be executed by the computer** as a software.
- **Documents and Configuration Files:** it contains **documentation files**, **installation instructions**, and **configuration files** required to run the software.
- **Issues and Tasks (Issues):** it provides a feature to **report issues or problems found** in the software.
- **Branch and Merge:** it allows developers to work on specific features or fixes **without affecting the code in the master branch.** When finished, the branch can be merged back into the main branch.
- **Change History (Commit History):** it **records every change made** to the source code

Version Control System (VCS)

Version Control System (VCS) is a system that is used to **track changes** to source code and other files in a software project.

VCS Type

- Distributed Version Control System (**Distributed VCS**): every single member team has a full copy of the repository in their own machine. It means that the versions can be saved in the **remote repository** as well as in **local repositories** of the local machines. Example: **Git**.
- Centralized Version Control System (**Centralized VCS**): a **central repository stores all versions** of source code and other files. Example: **Subversion**.



The Benefit of VCS

- **Change Tracking:** it records every change made by developers to the code. It allows developers to view **change history** as well as to **roll back** to previous versions if needed.
- **Team Collaboration:** it allows developers team to **work on the same code simultaneously**, and to **safely merge the changes**.
- **Conflict Management:** in case there are two or more developers make changes to the same part of the code, VCS helps **detect and resolve conflicts** that may appear.
- **Recovery and Troubleshooting:** If there is a problem with the latest changes, it **allows to easily roll back** to a previous working version.
- **Revisions and Versions:** it assigns each version of **code a unique number**, allowing developers to refer to a specific version.
- **Software Testing and Testing:** it allows to develop in an isolated environment, VCS supports **testing** and **experimentation without affecting the main code**.

The Use of GitHub, GitLab, Other Platforms

GitHub and other platforms is VCS provider. Its role is to provide infrastructure to **manage Git repository** and **facilitate developers team** to collaborate in software development.

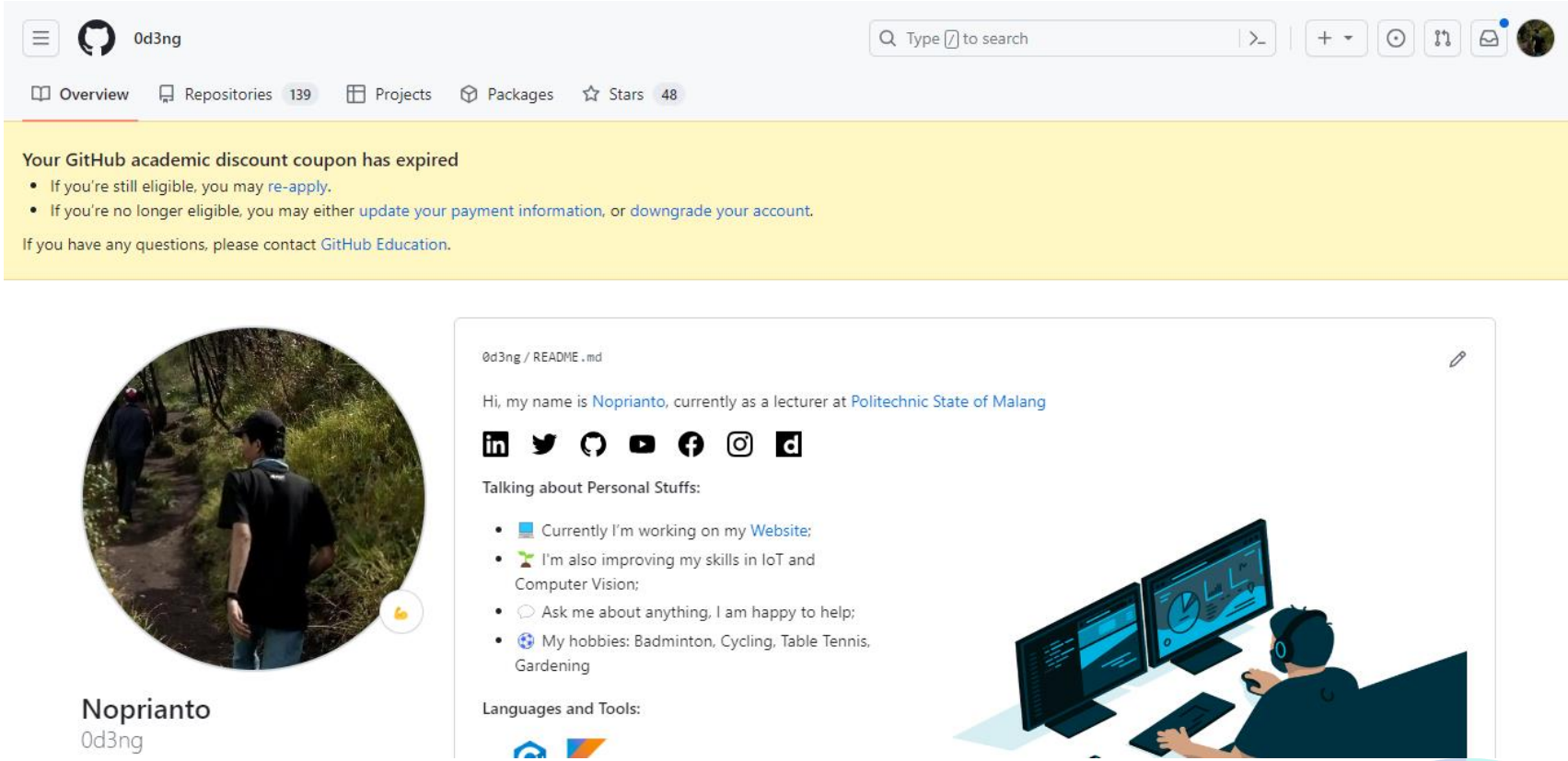
- **Hosting Repository**; a place to **store Git repositories** online.
- **Teams Collaboration**; it allows development teams to **work together** in a single repository.
- **Issues and Tasks Management**; it **tracks and manages issues**, **feature requests**, and **tasks** in projects.
- **Pull Request**; it **submits code changes** for a review by other developers before being merged into the main repository.
- **CI/CD (Continuous Integration/Continuous Delivery) Integration**; it supports **testing automation**, **integration automation** and **software delivery**
- **Access Settings**; it manages **permission and access** to ensure the right control to the repositories.

Peran GitHub, GitLab, dan Platform Serupa (2)

- **Project Documentation;** provides **project documentation** and **guidance** for developers and users.
- **Code Review;** support the **code review process with comments**, **suggestions**, and **other feedback**.
- **Third Party Integration;** has **integration with other development tools** such as project management tools, issue tracking systems, and planning tools.
- **Open Source Publications;** open source software development by making it **easy to share code with the community.**
- **Paid and Free Versions; paid with additional features**, while others have **free options for open source projects.**

GitHub

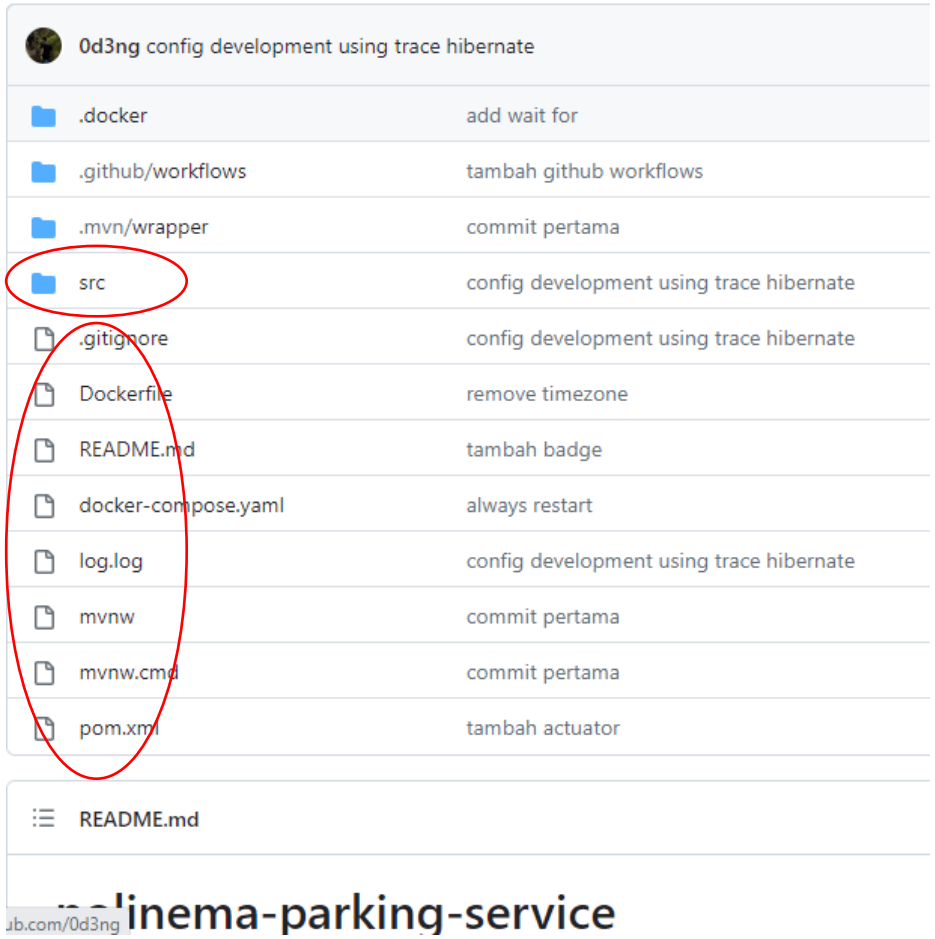
- GitHub is a web-based **repository management platform** based on the Git version control system.
- One of the most popular and widely used platforms by software developers, both in open-source projects and private projects.



Main Feature of GitHub

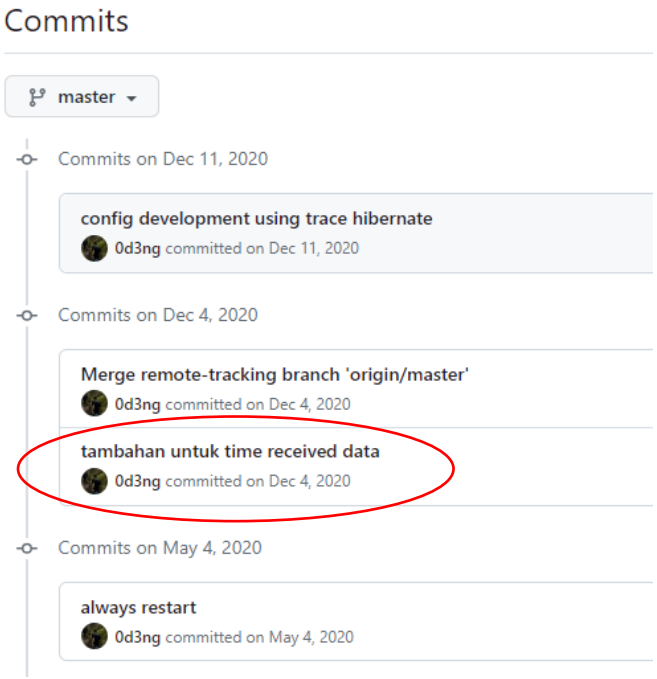
- **Repository;** create public or private **repositories** to **store source code, configuration files, documents**, and **other resources**
- **Git Integration;** perform standard **Git operations** such as **commit**, **branch**, **merge**, and **rebase** via the web interface or command line
- **Collaboration;** can **work together** in the same repository
- **CI/CD Integration;** GitHub Actions is a **Continuous Integration (CI) and Continuous Deployment (CD) integration tool** that integrates directly with the GitHub repository
- **GitHub Pages;** allows users to **host static websites** directly from GitHub repositories
- **GitHub Classroom;** an **educational platform** that facilitates **teaching** and **learning** Git and GitHub in a classroom environment

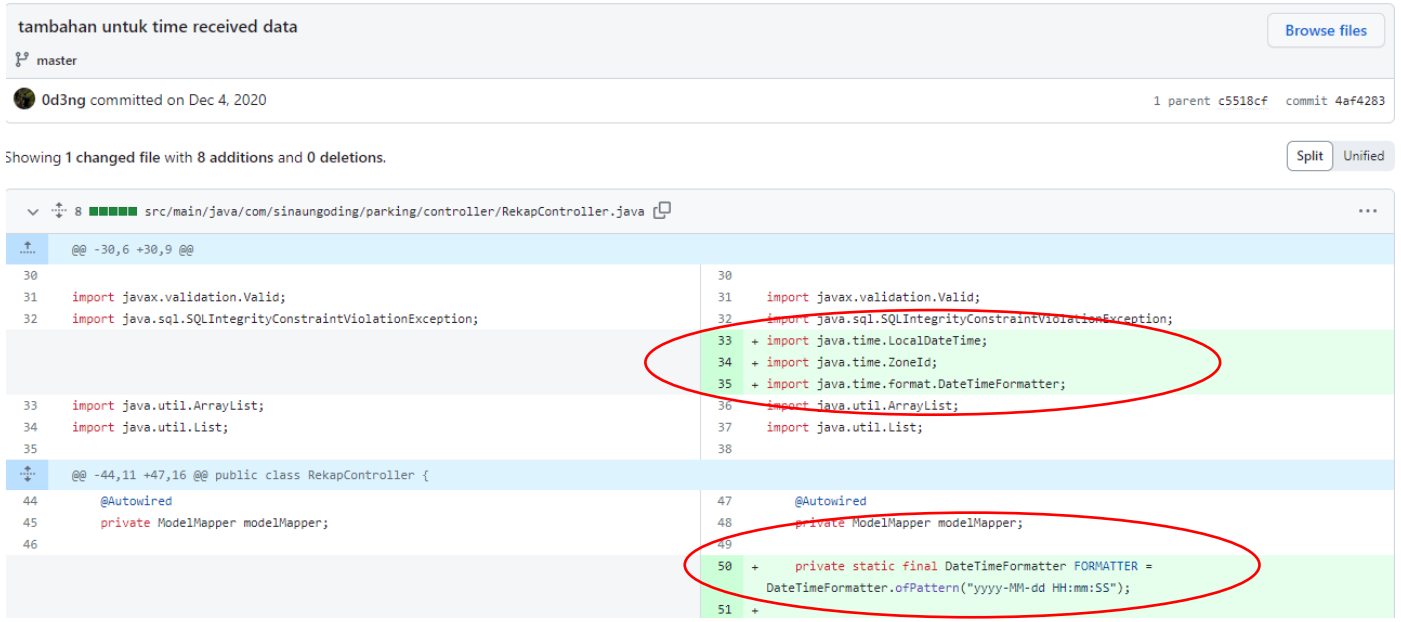
GitHub - Repository



- **src;** folder that is used to store program code.
- The other files contain the configuration of a project. For example the **pom.xml** file is a configuration file to store all dependencies/Libraries when using Maven (build tool).

GitHub - Git Integration





Commit is ensuring and recording every addition to program code. Every **commit** process will be recorded with information about when, who, which.

GitHub - Collaboration

JTI Dasar Pemrograman Menggunakan Binder

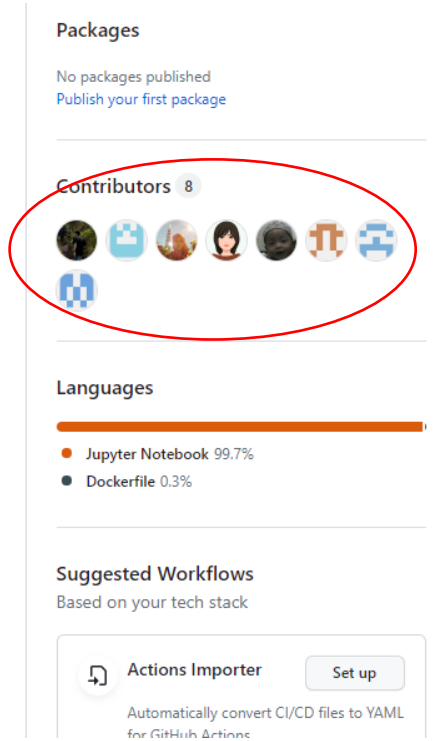
Tool online untuk membantu pembelajaran mata kuliah dasar pemrograman Java menggunakan jupyter notebook

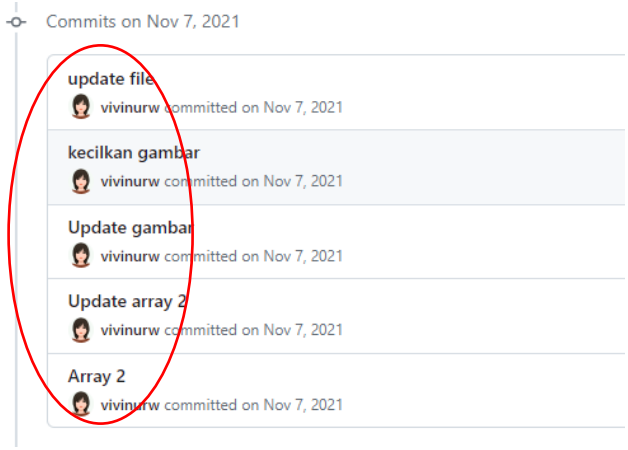
Klik link di bawah ini untuk menjalankan

- jti daspro - mode classic
- jti daspro lab - mode lab

Dosen pengampu matakuliah

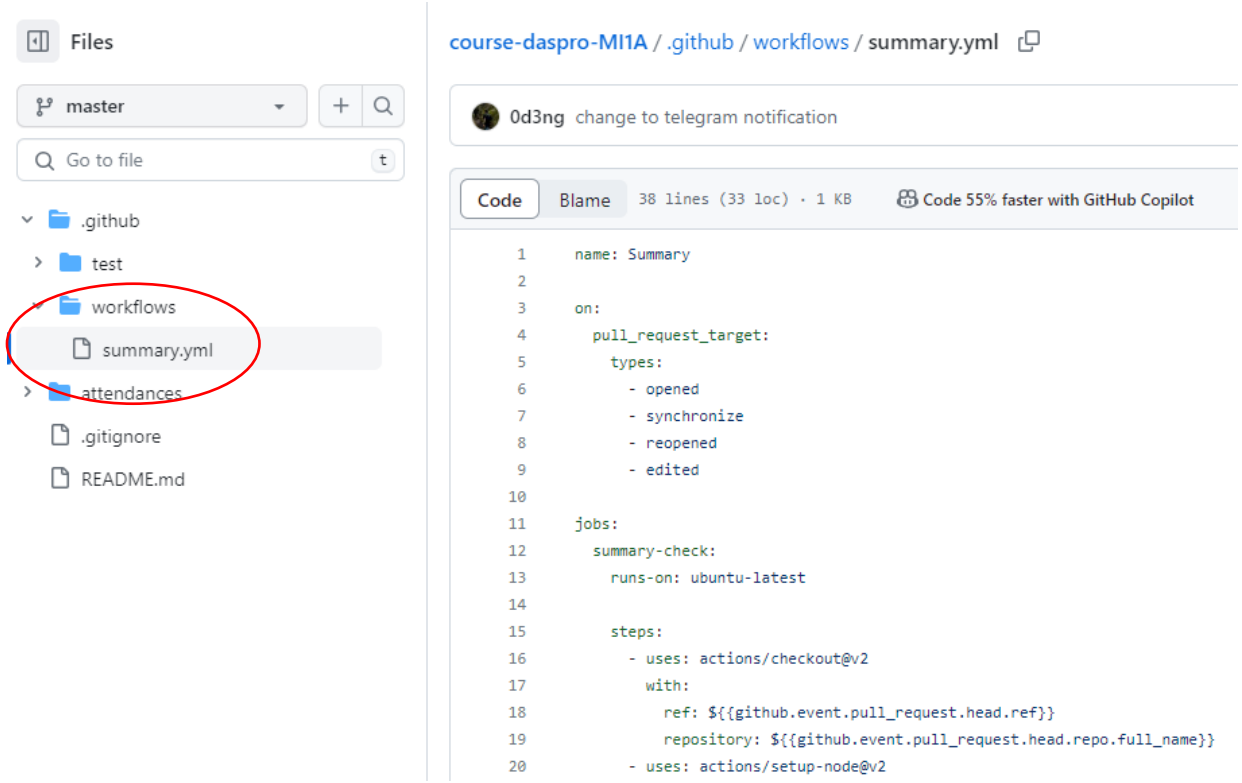
- Noprianto, S.Kom., M.Eng
- Adevian Fairuz Pratama, S.ST, M.Eng
- Ika Kusumaning Putri, S.Kom., M.T.
- Mamluatul Hani'ah, S.Kom., M.Kom.
- Mungki Astiningrum, ST., M.Kom.
- Mustika Mentari, S.Kom., M.Kom
- Pramana Yoga Saputra, S.Kom., MMT.
- Vivin Ayu Lestari, S.Pd., M.Kom
- Vivi Nur Wijayaningrum, S.Kom, M.Kom
- Moch. Zawaruddin Abdullah, S.ST., M.Kom
- Imam Fahrur Rozi, ST., MT.

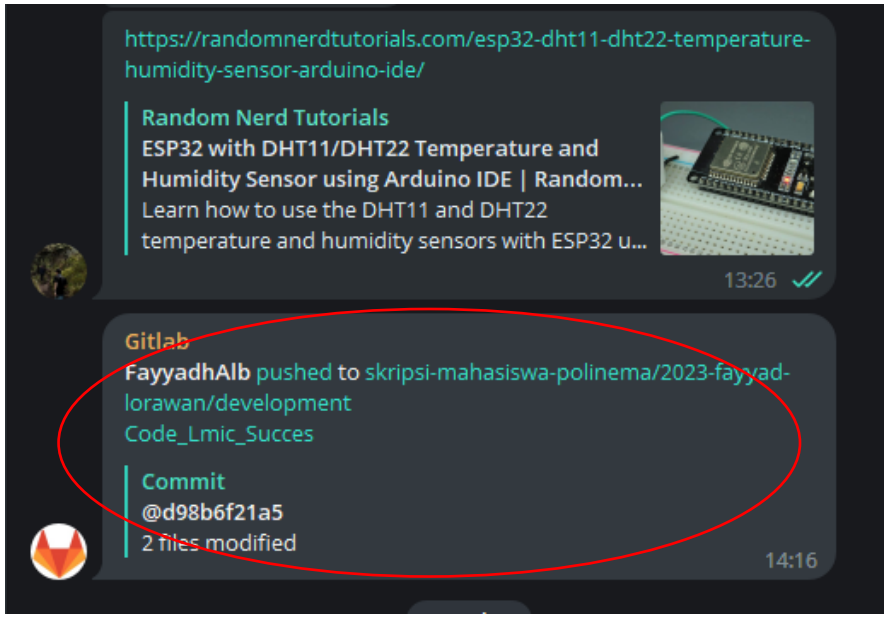




Multiple users can contribute to the same project.

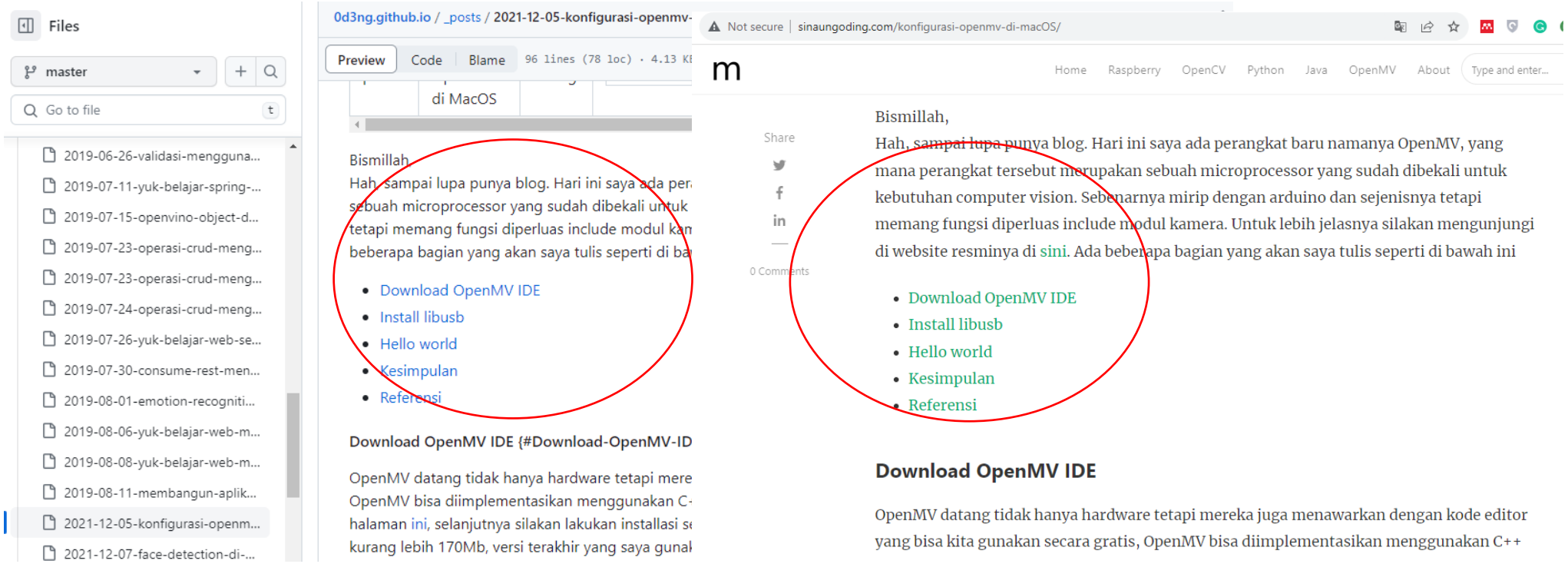
GitHub - CI/CD Integration





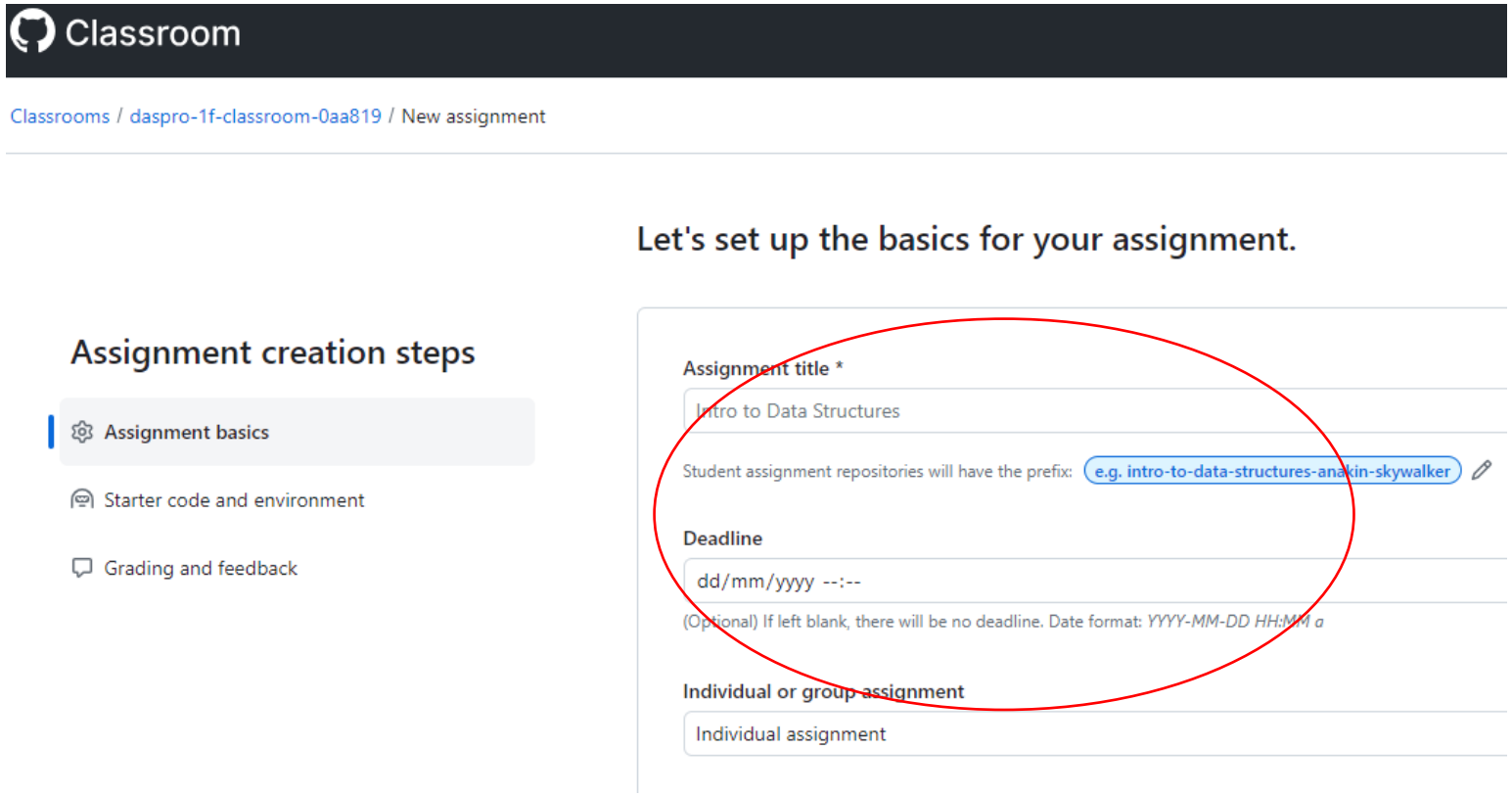
An example of using **continuous integration** is when there is a pull request or something else, it is adjusted to your needs and then merges it into the master, then provides a notification to Telegram.

GitHub - GitHub Pages



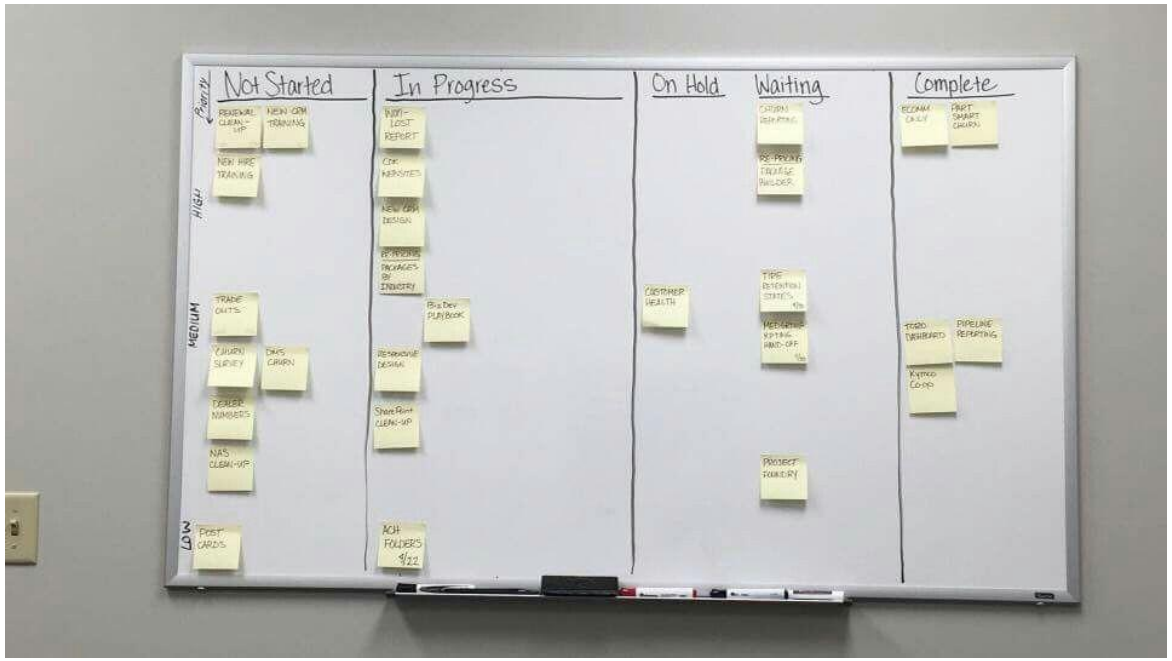
We can create a **static website** for **project documentation**, **portfolio**, or **personal pages** on GitHub using **GitHub Pages**.

GitHub - GitHub Classroom



GitHub Classroom has functions such as the **Learning Management System (LMS)** to **manage classes** such as giving assignments, providing materials, conducting assessments, and other functions for learning.

Kanban Board - Introduction



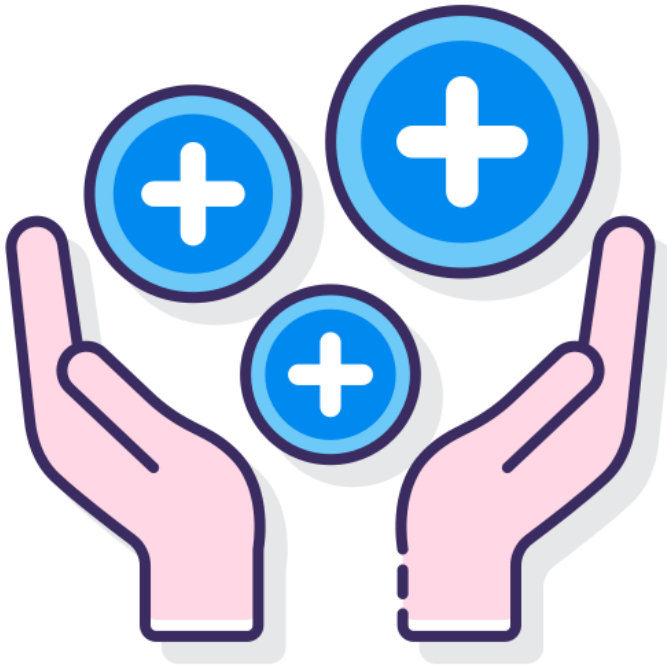
- Anyone popular with this?
- What is it for?

What is Kanban Board?

- **Kanban Board** as a **visual tool** for **tracking tasks** and **jobs**.
- **Kanban boards** provide a very effective way to **visualize task status, organize workflow, and increase productivity.**
- A Kanban board consists of **columns that represent different stages in the workflow**, as well as **cards or items that represent tasks or work** that must be done.
- Each card will be **visually moved** from one column to the next as the task progresses.
- Provides a clear overview of **workflow**, **task status**, and quickly **identifies bottlenecks** or **problems**.

Kanban Board Benefits

- **Workflow visualization;** provides a clear visual overview of workflow and task status**.**
- **Management of time and priorities;** allows to set up the priorities and to control the tasks flow**.**
- **Team collaboration;** helps the teams to communicate about work, take responsibility, and resolve problems.
- **Continuous improvement;** facilitates continuous evaluation and improvement in workflow**.**
- **Waste reduction;** reduces waiting times and excess work in the workflow**.**
- **Adaptation to change;** allows flexibility in dealing with changes and new priorities**.**
- **Transparency;** all team members can easily view task status and progress**.**



Kanban Board Structure

- **Backlog (To Do);** contains a list of tasks or **jobs that have not yet been started.**
- **In Progress;** contains tasks or **jobs that are being processed** by the team.
- **Reviews;** the stage where the tasks or works have been completed and **are being reviewed** or reviewed by other team members or related parties.
- **Testing/QA;** contains tasks or works that have passed the review stage and **are ready to be tested** or **verified**.
- **Done**; tasks or works that have been **completed**, **tested** and **approved** are moved to this stage.

The Kanban Board structure can be modified to the specific needs and workflow of a team or project.

Time Management and Priorities – Kanban Board

- Place the **highest priority tasks** at the top of the "To Do" or backlog column.
- Set a **limit on how many tasks** can be "In Progress" at the same time.
- Encourage the team to **complete one task before starting a new one**.
- **Reprioritize** tasks based on **needs/situation change**.
- When selecting the **next task** to work on, **consider its complexity, duration, and priority.**
- Use **color coding** or **labels** to mark tasks by priority or type.

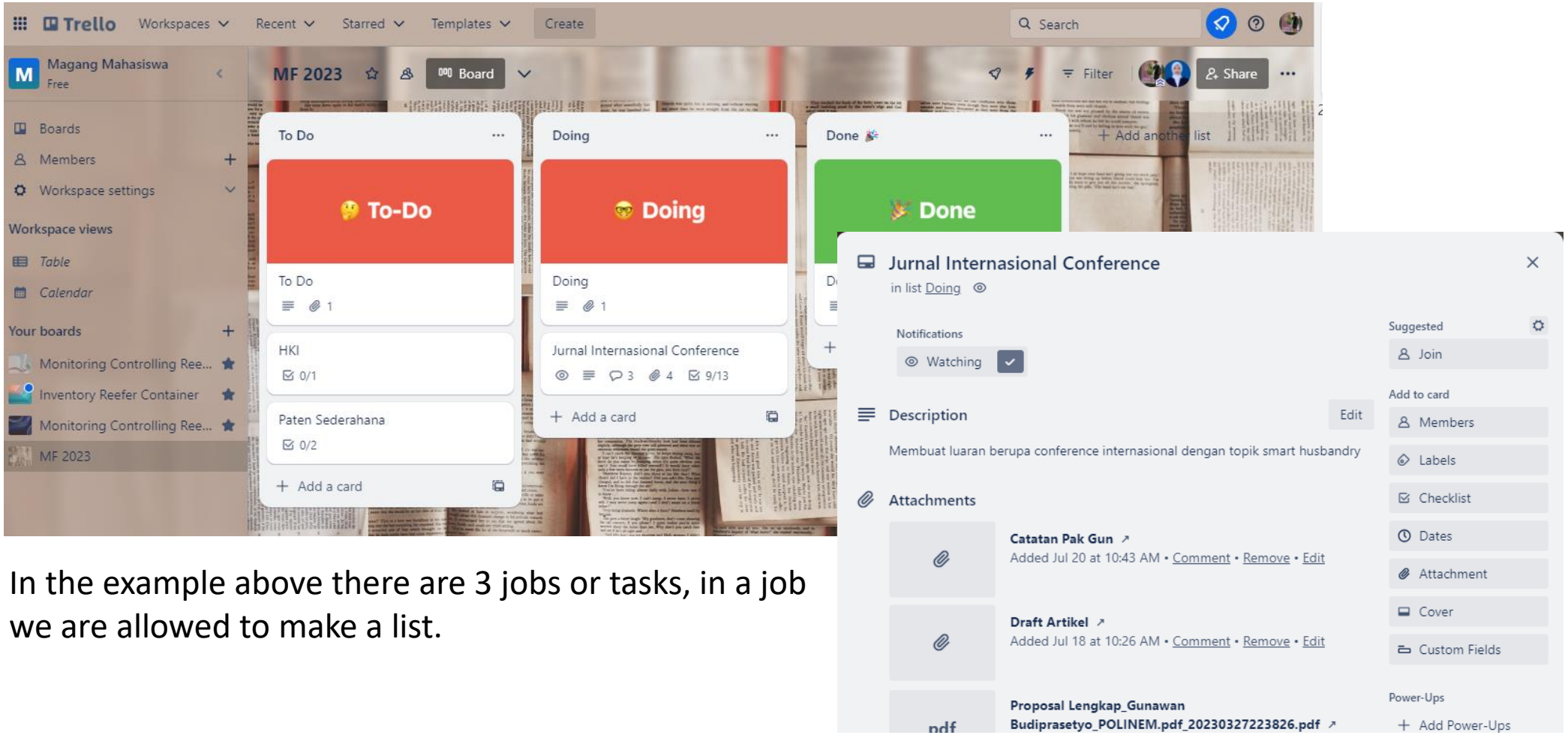
Tema Collaboration – Kanban Board

- Use the Kanban Board as a central point to **view the work status** of the entire team.
- When tasks are added to the Kanban board, explain **who will work on them**.
- Use the **comments feature** on Kanban cards or items to discuss work
- If your team is involved in software development, use the "**Review**" stage or a separate column for **code review**.
- If there are **obstacles** or **problems** affecting the work, tear members can **flag or add related issues to the task**.
- Perform a **regular team meetings** to be able to look at the Kanban board together and talk about work in progress.

Software – Kanban Board

- **Trello;** offers an intuitive interface with cards that can be moved between columns representing workflow stages.
- **Notion:** as a document creation tool, it's also possible to create Kanban boards that teams can access.
- **Jira software;** used for software development, has powerful Kanban Board features.

Kanban Board - Trello



In the example above there are 3 jobs or tasks, in a job we are allowed to make a list.

Kanban Board - Notion

Hide description

Kampus dan Uborampe

Task untuk daily activity urusan akademik bro

Incomplete | Complete | All +

Aa Task	Urgency	Importance	Compl...	Due date
Membuat RPP untuk MK Algoritma dan Pemrograman	Urgent	Important	☐	August 18, 2023
RPS MK Sistem Informasi Manajemen	Urgent	Important	☐	August 18, 2023
Cek BKD semester genap 2022/2023	Urgent	Important	☐	August 31, 2023

We can make priorities and deadlines of each task.

Assignment

- Please look for 5 **open-source** repositories with the **highest number** of contributors, stars and commits!
- Create a **daily activity** using a kanban board for 4 weeks (2 weeks back and 2 weeks ahead), at least the kanban structure consists of **To Do, In Progress,** and **Done**!