

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



Flashdisk

VS



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)

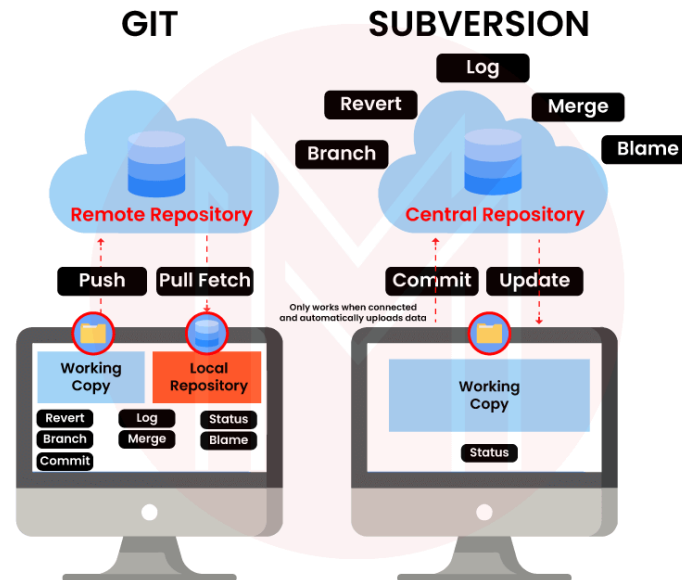
Version Control System (VCS) is a system that is used to **track changes** to source code and other files in a software project.





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.

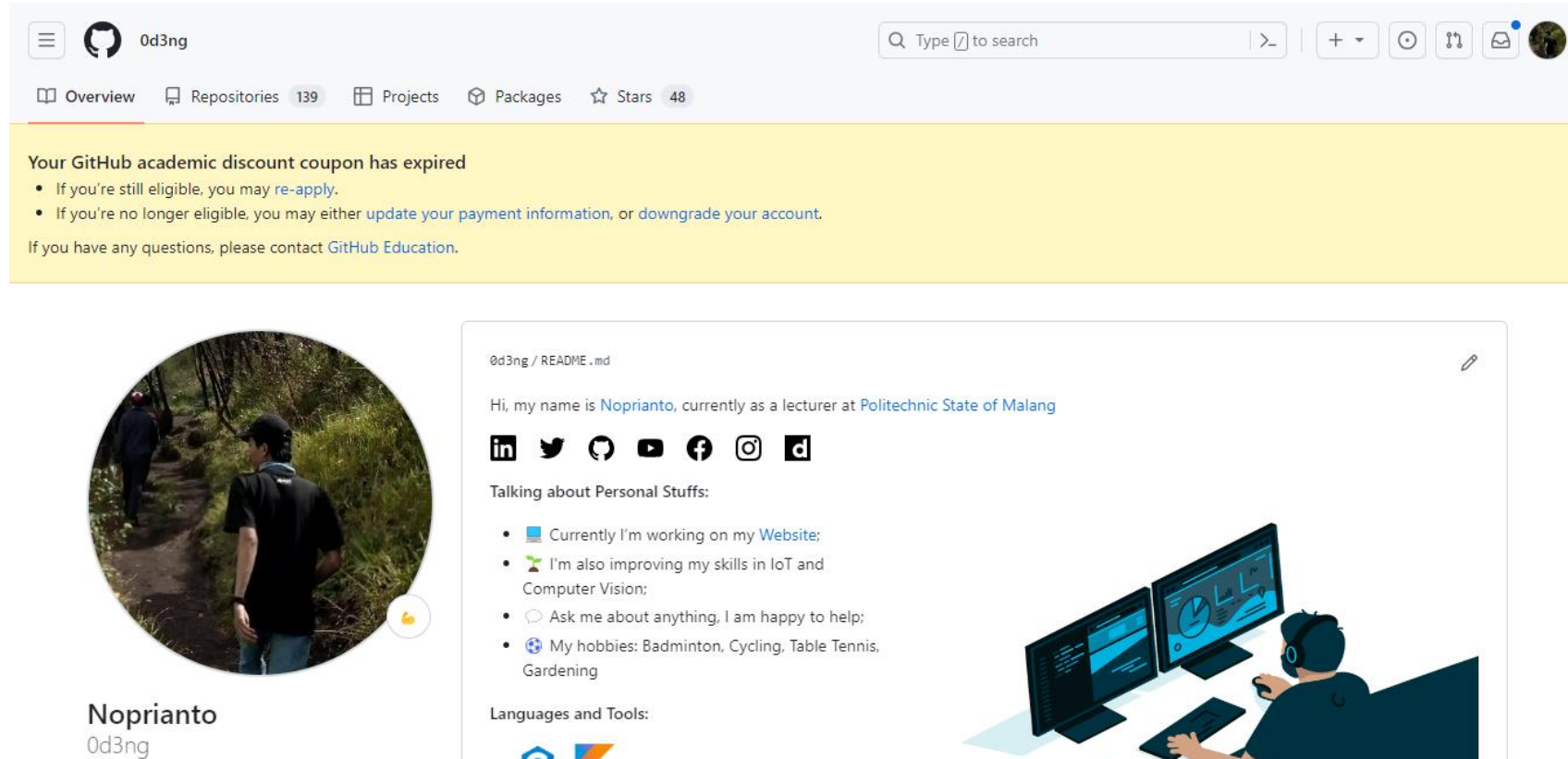
- **Hosting Repository**; a place to **store Git repositories** online.
- **Teams Collaboration**; it allows development teams to **work together** in a single repository.
- **Issues and Tasks Management**; it **tracks and manages issues, feature requests, and tasks** in projects.
- **Pull Request**; 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.

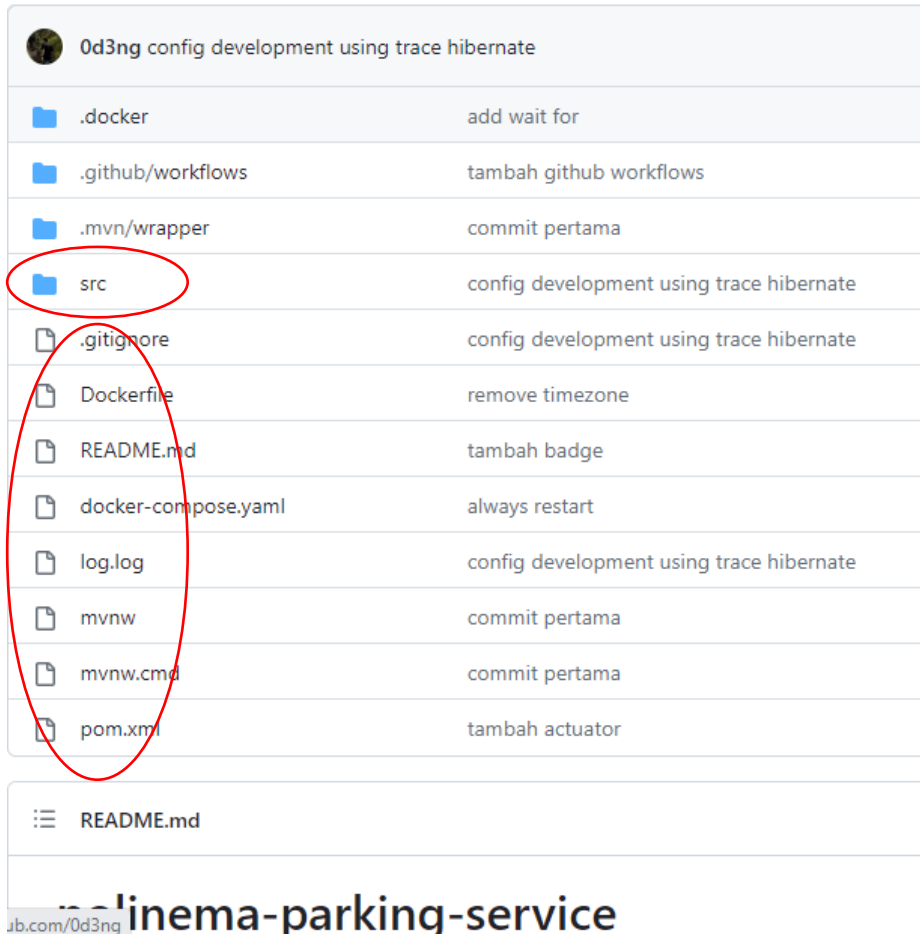


The screenshot shows a GitHub profile for user '0d3ng'. At the top, there's a navigation bar with a search bar and icons for repository management. Below this, a yellow banner states: 'Your GitHub academic discount coupon has expired'. It lists two options: 're-apply' if still eligible, or 'update your payment information, or downgrade your account' if not. A link to 'GitHub Education' is provided for questions. The profile section features a circular profile picture of a person on a path, the name 'Noprianto', and the username '0d3ng'. To the right, the 'README.md' file is displayed, containing a bio: 'Hi, my name is Noprianto, currently as a lecturer at Politechnic State of Malang'. Below the bio are social media icons for LinkedIn, Twitter, GitHub, YouTube, Facebook, Instagram, and Discord. A section titled 'Talking about Personal Stuff:' lists: 'Currently I'm working on my Website;', 'I'm also improving my skills in IoT and Computer Vision;', 'Ask me about anything, I am happy to help;', and 'My hobbies: Badminton, Cycling, Table Tennis, Gardening'. A 'Languages and Tools:' section shows icons for HTML and JavaScript. An illustration of a person at a computer with multiple monitors is in the bottom right corner.

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



0d3ng	config development using trace hibernate
.docker	add wait for
.github/workflows	tambah github workflows
.mvn/wrapper	commit pertama
src	config development using trace hibernate
.gitignore	config development using trace hibernate
Dockerfile	remove timezone
README.md	tambah badge
docker-compose.yaml	always restart
log.log	config development using trace hibernate
mvnw	commit pertama
mvnw.cmd	commit pertama
pom.xml	tambah actuator

README.md

cinema-parking-service

0d3ng

- **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

Commits

master

Commits on Dec 11, 2020

config development using trace hibernate
0d3ng committed on Dec 11, 2020

Commits on Dec 4, 2020

Merge remote-tracking branch 'origin/master'
0d3ng committed on Dec 4, 2020

tambahan untuk time received data
0d3ng committed on Dec 4, 2020

Commits on May 4, 2020

always restart
0d3ng committed on May 4, 2020

tambahan untuk time received data

master

0d3ng committed on Dec 4, 2020

1 parent c5518cf commit 4af4283

Showing 1 changed file with 8 additions and 0 deletions.

Split Unified

src/main/java/com/sinaungoding/parking/controller/RekapController.java

```
30
31 import javax.validation.Valid;
32 import java.sql.SQLIntegrityConstraintViolationException;
33 + import java.time.LocalDateTime;
34 + import java.time.ZoneId;
35 + import java.time.format.DateTimeFormatter;
36 import java.util.ArrayList;
37 import java.util.List;
38

33 import java.util.ArrayList;
34 import java.util.List;
35

44 @Autowired
45 private ModelMapper modelMapper;
46

47 @Autowired
48 private ModelMapper modelMapper;
49
50 + private static final DateTimeFormatter FORMATTER =
    DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
51 +
```

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

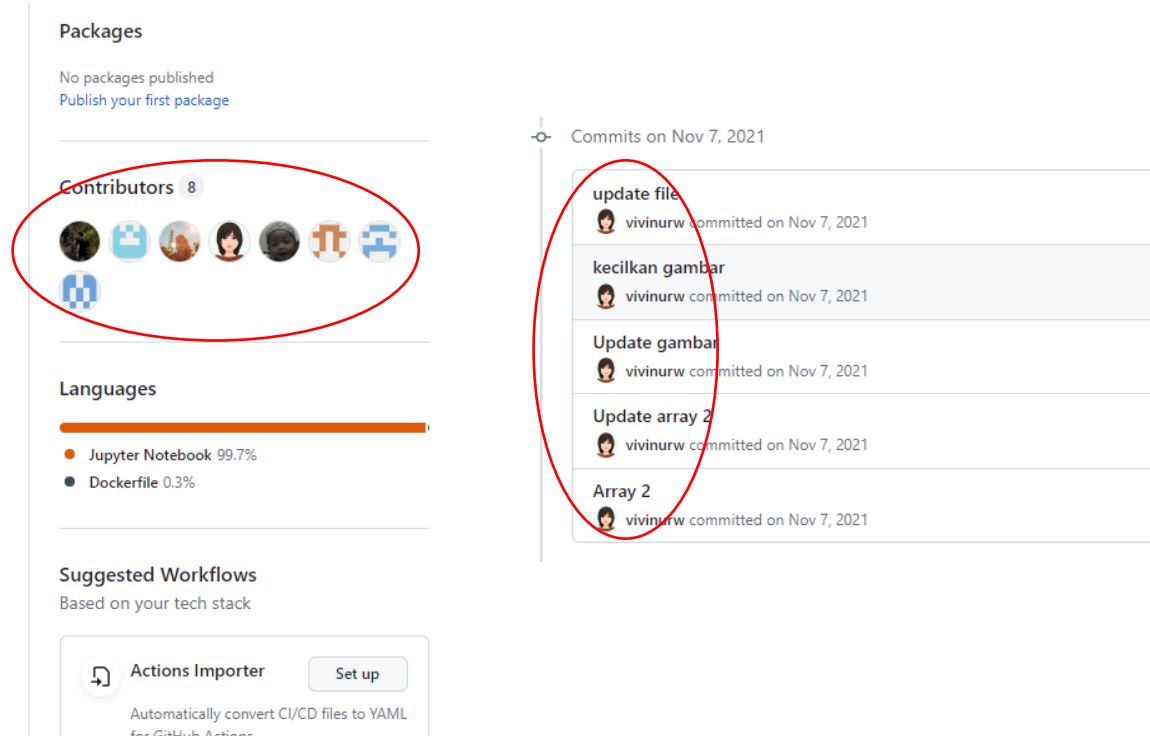
Tool online untuk membantu pembelajaran mata kuliah dasar pemrograman Java menggunakan jupyter notebook

Klik link di bawah ini untuk menjalankan

-  - mode classic
-  - 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 Wijyaningrum, S.Kom, M.Kom
- Moch. Zawaruddin Abdullah, S.ST., M.Kom
- Imam Fahrur Rozi, ST., MT.



Packages

No packages published
[Publish your first package](#)

Contributors 8

Languages

- Jupyter Notebook 99.7%
- Dockerfile 0.3%

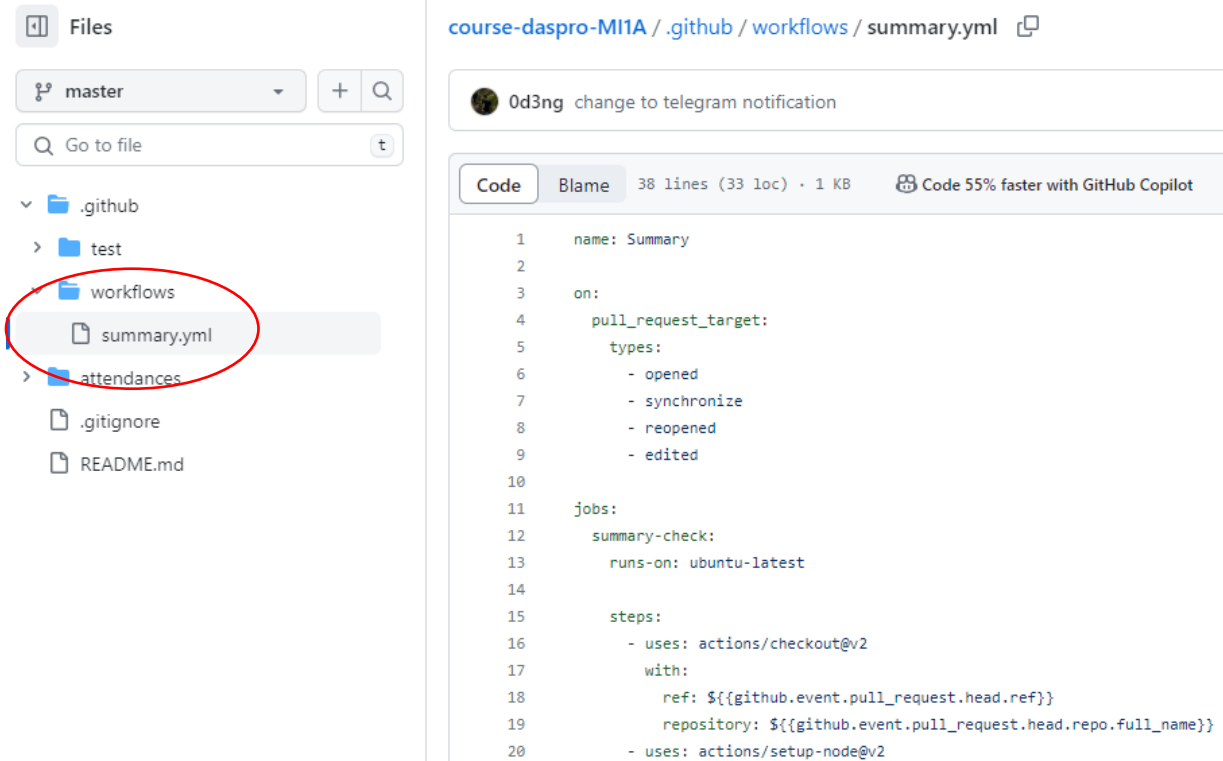
Suggested Workflows
Based on your tech stack

Commits on Nov 7, 2021

- update file
vivinurw committed on Nov 7, 2021
- kecilkan gambar
vivinurw committed on Nov 7, 2021
- Update gambar
vivinurw committed on Nov 7, 2021
- Update array 2
vivinurw committed on Nov 7, 2021
- Array 2
vivinurw committed on Nov 7, 2021

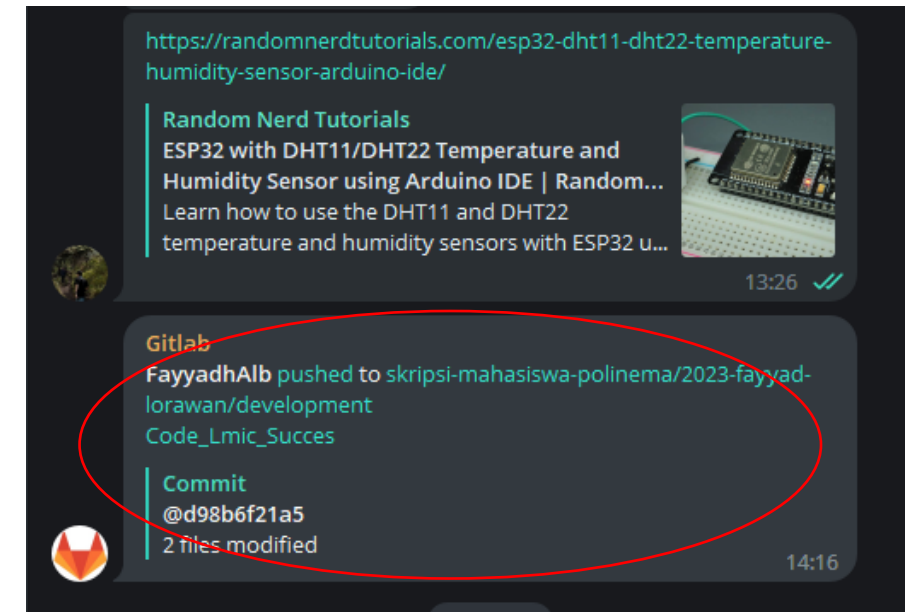
Multiple users can contribute to the same project.

GitHub - CI/CD Integration



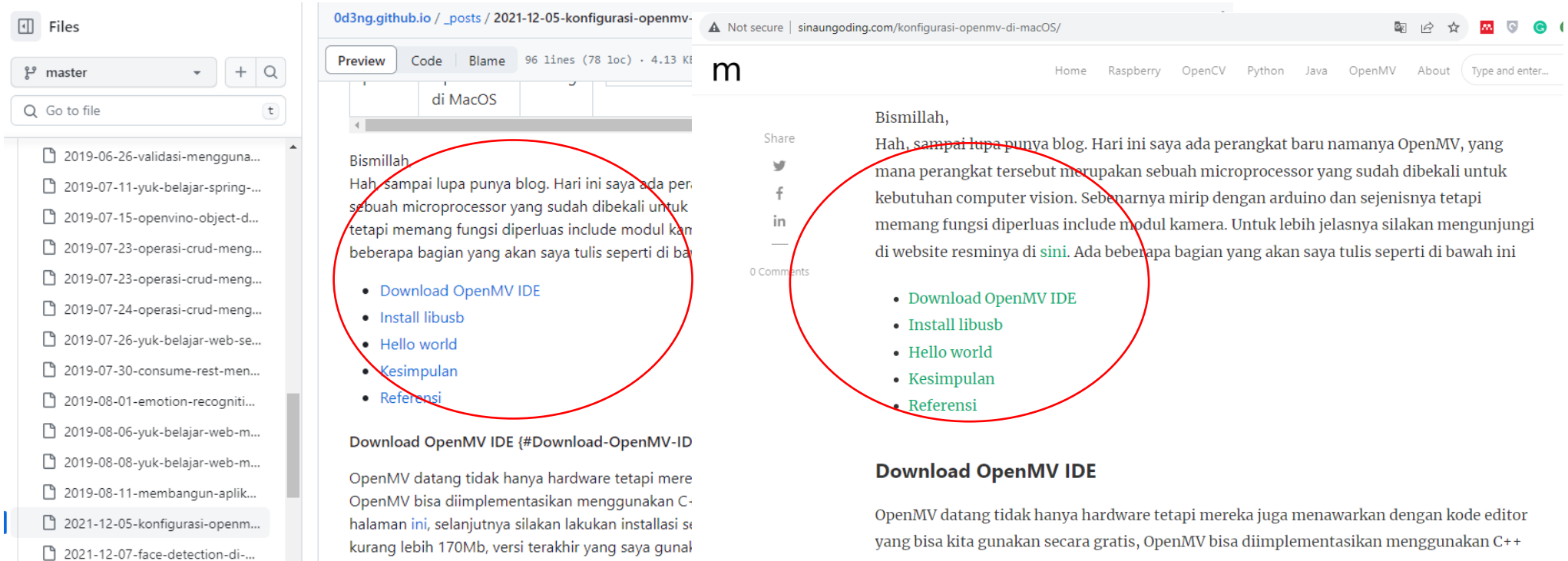
The screenshot shows a GitHub repository interface. On the left, the file explorer shows a directory structure with `.github`, `test`, `workflows` (highlighted with a red circle), `attendances`, `.gitignore`, and `README.md`. The `summary.yml` file is selected within the `workflows` directory. The main area displays the content of `summary.yml`, which is a GitHub Actions workflow. The commit message "0d3ng change to telegram notification" is visible at the top.

```
1 name: Summary
2
3 on:
4   pull_request_target:
5     types:
6       - opened
7       - synchronize
8       - reopened
9       - edited
10
11 jobs:
12   summary-check:
13     runs-on: ubuntu-latest
14
15   steps:
16     - uses: actions/checkout@v2
17     with:
18       ref: ${github.event.pull_request.head.ref}
19       repository: ${github.event.pull_request.head.repo.full_name}
20     - uses: actions/setup-node@v2
```



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.

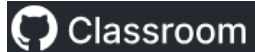
GitHub - GitHub Pages



The image shows a side-by-side comparison of a GitHub repository and its corresponding static website. On the left, the GitHub interface displays a file list on the left sidebar and a file viewer for '2021-12-05-konfigurasi-openmv-di-MacOS'. The file content is in Indonesian, starting with 'Bismillah,' and 'Hah, sampai lupa punya blog...'. It includes a bulleted list of links: 'Download OpenMV IDE', 'Install libusb', 'Hello world', 'Kesimpulan', and 'Referensi'. On the right, the rendered website is shown in a browser. It has a navigation bar with links like 'Home', 'Raspberry', 'OpenCV', 'Python', 'Java', 'OpenMV', and 'About'. The main content area mirrors the GitHub file content, including the same bulleted list of links. Red circles highlight the bulleted lists in both the GitHub file viewer and the rendered website, showing they are identical. The browser address bar shows the URL 'sinaungoding.com/konfigurasi-openmv-di-macos/'.

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

GitHub - GitHub Classroom



[Classrooms](#) / [daspro-1f-classroom-0aa819](#) / New assignment

Let's set up the basics for your assignment.

Assignment creation steps

⚙️ Assignment basics

💻 Starter code and environment

💬 Grading and feedback

Assignment title *

Intro to Data Structures

Student assignment repositories will have the prefix: `e.g. intro-to-data-structures-ana` `kin-skywalker` ✎

Deadline

dd/mm/yyyy --:--

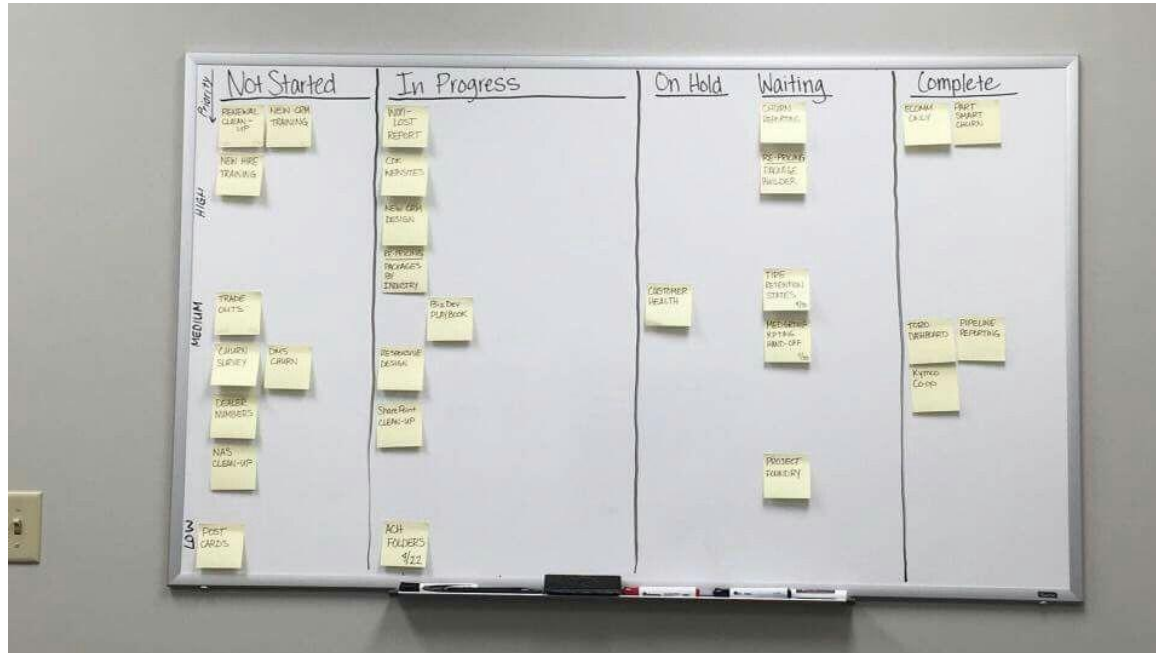
(Optional) If left blank, there will be no deadline. Date format: YYYY-MM-DD HH:MM a

Individual or group assignment

Individual assignment

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, team 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.

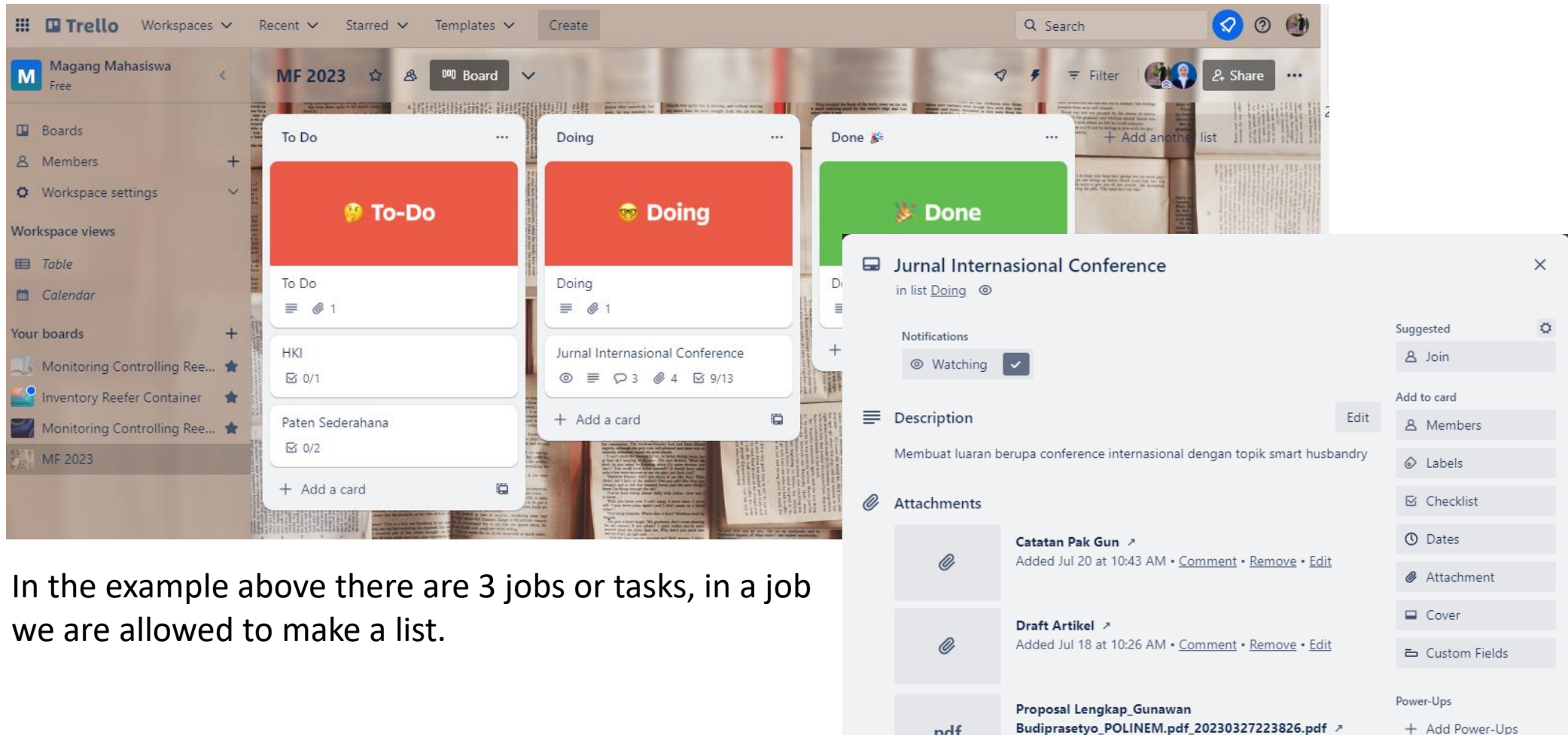


Software – Kanban Board

- **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	<input type="checkbox"/>	August 18, 2023
👤 RPS MK Sistem Informasi Manajemen	Urgent	Important	<input type="checkbox"/>	August 18, 2023
👤 Cek BKD semester genap 2022/2023	Urgent	Important	<input type="checkbox"/>	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!**

