

# GitHub

# What is GitHub?

- A cloud platform for collaborative software development
- Key features of GitHub:
  - Remote repositories using Git
  - Pull requests (PR)
  - Issues
  - Wikis, Pages, etc.

# Pull Request (PR)

- A pull request is a request to merge changes from a feature branch to the main branch on GitHub
  - Note: PR is not a Git command, but a GitHub feature
- PR makes the merge process easier and safer than the `git merge` locally
- Advantages of PRs
  - Code review
  - Discussion on the changes
  - Conflict detection
  - Integration with issues (tracability of changes)

# GitHub Flow

- A standard workflow for software development with Git/GitHub
- Branch rules
  - The **main** branch should always be **deployable**.
  - Code changes are made in a **feature branch**, created from the main branch.
    - For enhancements:  `feature/xxx`
    - For bug fixes:  `fix/xxx`
- The feature branch is pushed to GitHub and a **pull request (PR)** is created.
- The PR is **reviewed**, **approved** and **merged** into the main branch.

# GitHub Action

- GitHub Actions is a feature to automate workflows for software development.
- It can run tests, build and deploy the software, and more.
- Workflows are defined by a YAML file in the `.github/workflows` directory of the repository
- Triggers can include: push events, pull request, issue changes, etc.

## Example of GitHub Action YAML file

```
name: Run Tests and Build
on:
  push:
    branches:
      - main # Trigger on push to main branch
  pull_request:
    branches:
      - main # Trigger on pull request to main branch

jobs:
  build:
    runs-on: ubuntu-latest # Use the latest Ubuntu runner (virtual environment)
    steps:
      - name: Checkout code
        uses: actions/checkout@v2 # Checkout the code from the repository..
      - name: Set up Go
        uses: actions/setup-go@v2 # Set up Go environment
    ...
```

## GitHub Issues

- GitHub Issues are used to track bugs, enhancements, and tasks.
- Issues can be assigned, labeled, commented, and organized with milestones.
- Issues can be linked to PRs to ensure traceability.

# CD/CI (Continuous Deployment/Integration)

- A software development practice to automate the software development process
- CD: Continuous Deployment
  - Automatically build and test the code when changes are pushed.
- CI: Continuous Integration
  - Automatically deploy the application after the code is merged into the main branch.
- **GitHub Actions** can be used to implement CD/CI

Note: CD/CI can also be implemented with other platforms such as Jenkins, Travis CI, CircleCI, etc.

# Summary

- GitHub is a cloud platform for collaborative software development
- Pull requests are used to review and merge code into the main branch
- GitHub Flow is a standard workflow for team-based development
- GitHub Actions automate testing and deployment workflows
- GitHub Issues help manage tasks and changes

CD/CI cannot be implemented without a platform such as GitHub and other tools.