GitHub

CMSE 890-402

Issues

- Records text and files about specific topics
  - Bugs
  - Improvements
  - Anything you want!
- Can be:
  - Assigned to developers
  - Labeled
  - Attached to “projects” and “milestones”
- Automatically closed if a commit or pull request references the issue as “fixed” or with similar language

Pull Requests

- Control branch merging
- Multiple merge options
  - Squash
  - Merge
  - Auto-merge
- Can be reviewed, labeled
- Connect to projects, milestones, releases…

Projects

- Collect issues and pull requests
- Can contain notes
- Great for organizing development areas

Actions

- Workflows *for your code*
- Written in YAML
- Typically runs bash commands
- Works using a “runner”
  - Virtual machine that runs a container
  - Executes your commands following the workflow

Other features

- Milestones: track versions and group issues/PRs
- Discussions: threads for…discussion
- Wiki: markdown pages on the repository
- Statistics

Not-GitHub

- GitLab
- BitBucket
- Both have most of the same features as GitHub

In-class assignment

- Create a repository
- Commit your unit test files and functions from **In-class assignment 3**
- Create a project board with the rest of the tasks as **issues**:
  - Commit documentation markdown files
  - Add a workflow to run the unit tests (https://docs.github.com/en/actions/automating-builds-and-tests/building-and-testing-python#testing-your-code)
  - Add a workflow to build and deploy the documentation (https://parkererickson.github.io/portfolio/blog/MkDocsCD/)
- **Clone** the repository you made
- Create a branch for each workflow file
- Add the workflows, commit, and run them

Discuss your semester project with your table!

Homework: get started on your semester project DFD

- Draft in whatever software you feel comfortable with
- Follow the symbol guides from class 1
- Think about how to break your project into small parts
- Think about reasonable scope!
- Submit to me for feedback by Oct 9th midnight
- Since you all mostly did this for class 1, I expect this to be a quite polished DFD taking into account feedback from class 1

Pre-class 7

Take a look at some of these links to learn a little about code review:

https://about.gitlab.com/topics/version-control/what-is-code-review/

https://google.github.io/eng-practices/review/

https://google.github.io/eng-practices/review/reviewer/

https://en.wikipedia.org/wiki/Code_review

https://blogs.oracle.com/javamagazine/post/five-code-review-antipatterns

In the spirit of code review, please put your semester project ideas onto the D2L discussion as your pre-class assignment.