



Recitation 1 - 09/04/2020

Remote Work & Collaboration

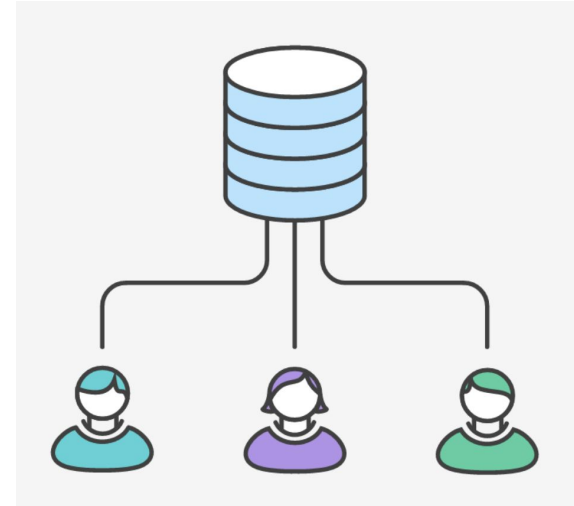


Technical & Non-technical Aspects

- Technical (code and non-code artifacts)
 - Code
 - Design
 - Documents
 - ...
- Non-technical
 - Communication
 - Availability
 - Common Understanding
 - ...

Introduction to Git

- Distributed version control system
- Has a central repository of the code
- Users can have local copies of the code
- Local updates get pushed to the central repository
- Commits to the central repository are authenticated
- Git is line-based
- Version control provides traceability benefits
- Desktop/GUI support (Source Tree, GitHub GUI, etc.)



Source: <https://info201-s17.github.io/book/git-collaboration.html>



Some Common Operations

- Clone : Make a local copy of a remote repository
- Pull : Download latest code from a remote repository to a local repository
- Commit / Check in : Update code / create a new revision in the local repository
- Push : Update the remote repository with local code changes
- Checkout : Switch to a branch / make a branch as your working branch
- Merge : Merge code from two branches



Commands

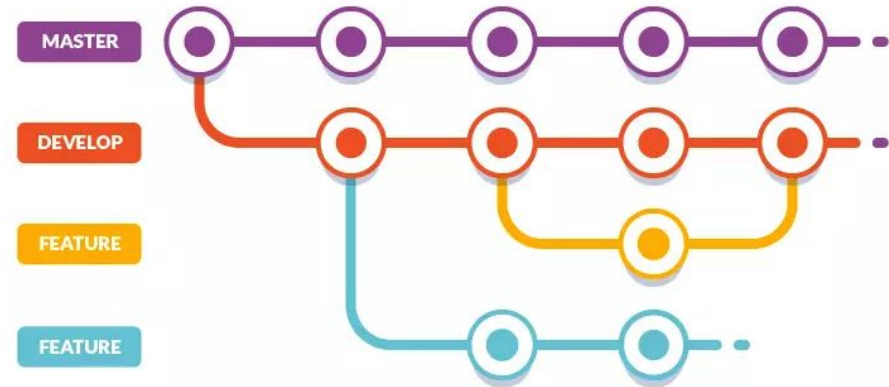
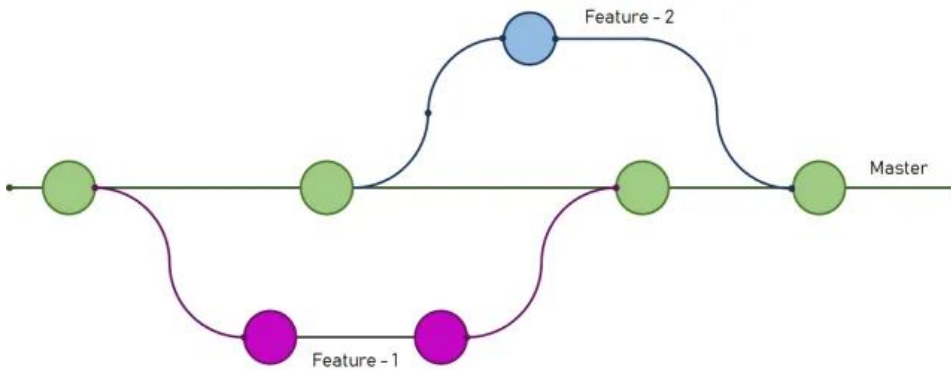
- git init
- git clone
- git fetch
- git checkout
- git pull
- git add
- git commit
- git push
- git stash
- git status
- git log
- git revert
- git reset
- git tag
- git merge
- git rebase
- git config
- git diff



Best Practices

- Use meaningful, descriptive commit messages
- Commit frequently (for completion of a small logical unit of code)
- Avoid committing generated files
- Use branches wisely
- Make use of pull requests (add description to help the reviewer)
- Have a defined process (or) Git workflow
- Aim to version control all code (IaaS, configurations, etc.)

Branching Strategy



Source: <https://digitalvarys.com/git-branch-and-its-operations/>,
https://www.sitepoint.com/use-git-branches-buddy/?utm_content=bufferd206b&utm_medium=social&utm_source=facebook.com&utm_campaign=buffer



Pull Requests

- We looked at branching strategies earlier
- Well, how would you merge code from a child branch to a parent branch?
 - Do you just merge directly?
 - Ideally, there should be a “process” in place
 - Pull request is a widely adopted way to do this
 - Similar to Git commit messages, pull requests also have certain good practices



Activity 1: Create a Pull Request

ckaestne / seai

Watch

9

Star

67

Fork

32

Code

Issues

Pull requests

Actions

Wiki

Security

Insights

F2020

6 branches

0 tags

Go to file

Add file

Code



ckaestne minor slide updates



9cd18a7 10 hours ago



314 commits



.assets

update template

13 days ago



.github/workflows

update to never version from S2020

13 days ago



assignments

Update l1_case_study.md

yesterday



lectures

minor slide updates

10 hours ago



other_lectures

added previous course material

15 months ago



other_material

add final exam and feedback form

8 months ago

About

CMU 17-445/645: Software Engineering for AI-Enabled Systems (SE4AI)

ckaestne.github.io/seai/

machine-learning

education

course-materials

software-engineering

carnegie-mellon-university

Readme

View license

ckaestne / seai

<> Code

! Issues

🔗 Pull requests

▶ Actions

🔗 F2020 ▾

🔗 6 branches

🔗 0 tags



ckaestne minor slide updates



.assets

update tem



.github/workflows

update to n



assignments

Update l1_c



lectures

minor slide



other_lectures

added prev



other_material

add final ex



.gitignore

migrate to custom gh-pages with circleci

14 months ago

👁 Watch ▾

9

☆ Star

67

🍴 Fork

32

Fork seai

×

Where should we fork seai?



vaithya94



17-654-MSE-Analysis



cmu-seai

Can't find what you're looking for?

You don't have permission to fork to these organizations:



PlanetaryRobotics

About

CMU 17-445/645: Software Engineering for AI-Enabled Systems (SE4AI)

🔗 ckaestne.github.io/seai/

machine-learning

education

course-materials

software-engineering


carnegie-mellon-university

📖 README

📄 View license

 **vaithya94** / **seai**

forked from [ckaestne/seai](#)

 Watch ▾

0


 Star

0

 Fork

33

 **Code**


 Pull requests

 Actions

 Projects

 Wiki

 Security

 Insights

 Settings

 **F2020** ▾

 **6** branches

 **0** tags

Go to file

Add file ▾

 **Code** ▾

This branch is even with ckaestne:F2020.

 Pull request

 Compare



ckaestne minor slide updates

9cd18a7 10 hours ago  **314** commits



.assets

update template

13 days ago



.github/workflows

update to never version from S2020

13 days ago

About



CMU 17-445/645: Software
Engineering for AI-Enabled Systems
(SE4AI)

 [ckaestne.github.io/seai/](https://github.io/seai/)

 Readme

 View license

vaithya94 / seai

forked from ckaestne/seai

Watch 0

Star 0

Fork 33

<> Code

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

F2020

6 branches

0 tags

Go to file

Add file

Code

This branch is even with ckaestne:F2020.



ckaestne minor slide updates

.assets

update template

.github/workflows

update to never version from S2020

assignments

Update I1_case_study.md

lectures

minor slide updates

10 hours ago

other_lectures

added previous course material

15 months ago

Clone with HTTPS

Use SSH

Use Git or checkout with SVN using the web URL.

https://github.com/vaithya94/seai.g



Open with GitHub Desktop

Download ZIP

About



CMU 17-445/645: Software Engineering for AI-Enabled Systems (SE4AI)

[ckaestne.github.io/seai/](https://github.com/ckaestne/seai)


Readme

View license


Releases

No releases published

[Create a new release](#)



```
VaithyathansMBP:se4ai-ta vaithya$ git clone https://github.com/vaithya94/seai.git
Cloning into 'seai'...
remote: Enumerating objects: 100, done.
remote: Counting objects: 100% (100/100), done.
remote: Compressing objects: 100% (68/68), done.
remote: Total 3347 (delta 64), reused 60 (delta 29), pack-reused 3247
Receiving objects: 100% (3347/3347), 336.77 MiB | 7.97 MiB/s, done.
Resolving deltas: 100% (1542/1542), done.
VaithyathansMBP:se4ai-ta vaithya$
```



```
VaithyathansMBP:se4ai-ta vaithya$ cd seai/
VaithyathansMBP:seai vaithya$ ls
LICENSE.md          learning_goals.md    lectures             overview.png        poster_small.jpg
README.md           lecture_dependencies.dot  other_lectures      poster.jpg          schedule.md
assignments         lecture_dependencies.svg  other_material      poster.pdf
VaithyathansMBP:seai vaithya$ mkdir practice
VaithyathansMBP:seai vaithya$ vi hello.txt
VaithyathansMBP:seai vaithya$ git add .
VaithyathansMBP:seai vaithya$ git commit -m 'Recitation practice: Added hello.txt with a hello message to practice pushing code to a remote repo'
[F2020 0c5791d] Recitation practice: Added hello.txt with a hello message to practice pushing code to a remote repo
 1 file changed, 1 insertion(+)
 create mode 100644 hello.txt
VaithyathansMBP:seai vaithya$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 334 bytes | 334.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/vaithya94/seai.git
   9cd18a7..0c5791d  F2020 -> F2020
VaithyathansMBP:seai vaithya$
```



 **vaithya94 / seai**

forked from [ckaestne/seai](#)

 Watch ▾

0


 Star

0

 Fork

33

< > Code


 **Pull requests**

 Actions

 Projects

 Wiki

 Security

 Insights

 Settings

Filters ▾

🔍 is:pr is:open

 Labels 9

 Milestones 0

New pull request

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).



base repository: ckaestne/seai ▾

base: F2020 ▾



head repository: vaithya94/seai ▾

compare: F2020 ▾

✓ **Able to merge.** These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

[Create pull request](#)

1 commit

1 file changed

0 comments

1 contributor



Commits on Aug 31, 2020



Recitation practice: Added hello.txt with a hello message to practice...

0c5791d

Showing **1 changed file** with **1 addition** and **0 deletions**.

Unified

Split

1 hello.txt

...

... @@ -0,0 +1 @@

1 + hello!

Recitation practice: Added hello.txt with a hello message to practice... #46

Edit

Open with ▾

🔗 Open vaithya94 wants to merge 1 commit into [ckaestne:F2020](#) from [vaithya94:F2020](#) 📄

💬 Conversation 0 🔗 Commits 1 📄 Checks 0 📄 Files changed 1

+1 -0 🟢🟡🔴🟠🟡🟢



vaithya94 commented now



No description provided.



Recitation practice: Added hello.txt with a hello message to practice... ⋮

0c5791d

Add more commits by pushing to the **F2020** branch on [vaithya94/seai](#).



This branch has no conflicts with the base branch

Only those with [write access](#) to this repository can merge pull requests.

Reviewers

No reviews

Still in progress? Convert to draft

Assignees

No one assigned

Labels

None yet

Projects

PudFish requested your review on this pull request.

Add your review

Feature/simplify #20

Edit

Open with ▾

🔗 Open PudFish wants to merge 38 commits into `develop` from `feature/simplify` 📄

💬 Conversation 0 🔗 Commits 38 📄 Checks 14 📄 Files changed 91

+2,820 -3,451 🟢🔴🔴🔴🔴



PudFish commented 4 days ago • edited ▾



I have reduced, simplified, and refactored the code with the intent of making it more maintainable, readable, consistent and therefore extendable. I spent some time over semester break learning best practices and testing out code patterns on MockRanger. I'm embracing the go way of keeping it simple and clean and trying not to follow habits from other programming languages. These changes impact how to deploy and get the code running, but the functionality should be left unaltered. Let me know in the comments if you have any concerns with the changes, need more rationale or I've broken something.

Docker

- Added a Dockerfile for the Postgres database
- Simplified the Dockerfile for the database access module
- Changed the database access module to execute the binary
- Simplified the docker-compose to build from Dockerfile
- Removed exporting both dev & non-dev ports from docker-compose

Reviewers



- mohanish
- lijiawu
- vaithya94

Still in progress? Convert to draft

Assignees



No one—assign yourself

Labels



None yet

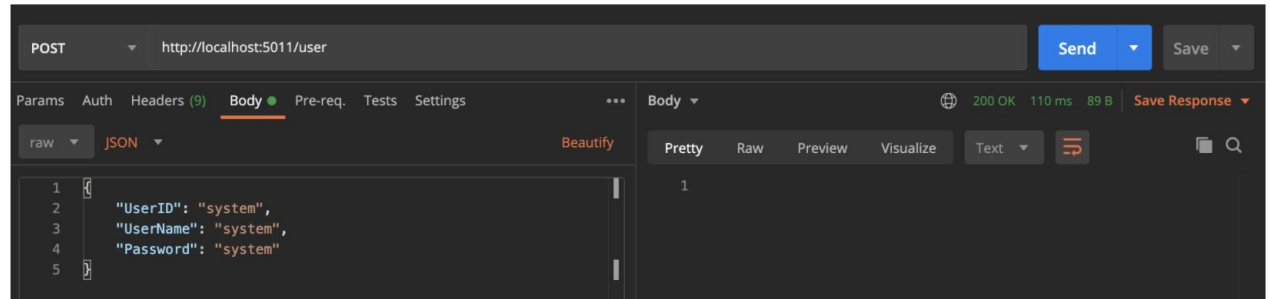
Story: MS-445

Design:

https://docs.google.com/document/d/1qOj9CtcrvzLkx3WEroD5m_MKYbyxw0mH2LGx8zcOKTc/edit#heading=h.1pag06vmzove

Test Plan:

- Sync the branch for frontend and database to your local.
- Run the frontend, backend and database services, and make sure that the ports are configured in such a way that they can talk to each other. Ensure that the database in the database config reflects your local database name.
- Run the SQL scripts `set_up_tables.sql` and `populate_seed_power_telemetry.sql` one by one using DataGrip. Your database should have mock rows in `backend.messages` table.
- Start all your three services.
- Create a user in the system using Postman.



- In the frontend, login using the username and password. Click on "Visualization" in the left pane.
- You should see the below screen:



Pull Requests: Best Practices

- Ensure that you've met the acceptance criteria for your work before requesting PR review
 - Part of your process
 - Should one round of manual testing be completed by the creator of the PR?
 - Should unit tests be available?
 - ...
- Meaningful and useful title, description, etc.
 - What are your changes? Why did you make those changes?
 - Information to assist the reviewer
 - Links to other resources/tickets that might provide better context of the PR to the reviewer
- Good commit messages



Pull Requests: Best Practices

- Nice to have
 - Test plans (if not already available as a resource)
 - Screenshots if applicable
- Add peer review comments on the pull request
- Merge when the PR passes the criteria that you set for approval/merging
 - Example: All review comments have been addressed, PR has approvals from 2 peers, etc.
- A pull request template could be beneficial for long-term projects



Activity 2: Remote Work - Barriers & Mitigation Tactics

- In your breakout rooms
 - Think about potential barriers to remote collaboration and teamwork
 - What are potential strategies that you can adopt to mitigate these problems?
- After 5 minutes, come back to the main room, and
 - Explain what problems you identified
 - What strategies did you come up with to resolve those problems



Things to Consider

- Communication [why? what? how? when?]
 - Consider availability of team members
 - Document discussions, key decisions, work to be done, commitments, etc.
- Areas of expertise
 - Get to know the strengths of each team member
- Representation of work
 - Break down your work into tasks / small chunks of work
 - Spend some time to identify dependencies
 - Is everyone aware of what they need to work on, and what is the status of the team as a whole?
- Remote pair programming
 - Personally, I've found it beneficial to do pair programming, especially for tasks that are new to the team



Things to Consider

- Plan for the future
- Help each other
- Don't rush to implementation
 - Spend adequate time in designing your system
 - Consider current and future requirements
- Clarify any assumptions you make
- Make choices considering the time available, expertise in the technology used, etc.



Resources

- <https://www.atlassian.com/git/tutorials/learn-git-with-bitbucket-cloud>
- <https://docs.github.com/en/github/authenticating-to-github/connecting-to-github-with-ssh>
- <https://info201-s17.github.io/book/git-collaboration.html>
- <https://git-scm.com/doc>
- <http://try.github.io/>
- <http://learngitbranching.js.org/>