CptS 487
Software Design and Architecture

Lesson 1

Overview, project & teams,

EECS gitlab, git

WASHINGTON STATE UNIVERSITY
World Class. Face to Face.

Instructor: Bolong Zeng

- **Instructor:**
- Name: Bolong Zeng, PhD. (Bob)
- E-mail: bzeng@wsu.edu
- Office Hours: MW 1pm – 2:30pm via zoom
  - See zoom link on Canvas
- Or by appointment.

Assumptions for this Class

- Prerequisites:
  - "CptS 321 Object Oriented Principles"
  - "CptS 322 Software Engineering Principles I"

  Both with a C or better.

- Assumptions:
  - Familiar with git
  - Familiar with UML, especially Class diagram

Times and Locations

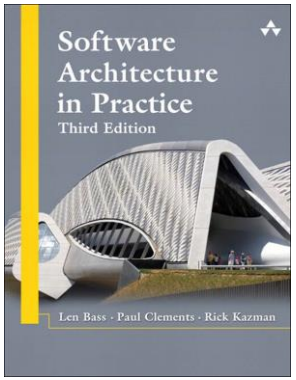
- TuTh 12:05 – 1:20pm
- Everett: Room 452; Pullman: CUE 114

- We'll use Canvas for all course materials
- Host the code on gitlab.eecs.wsu.edu
  - Requires your EECS account; please have it ready asap

Text Book and Reading Materials

- Required Textbooks:
  - L. Bass, P. Clements, R. Kazman

  Software Architecture in Practice, 3rd ed., Addison-Wesley, 2012.



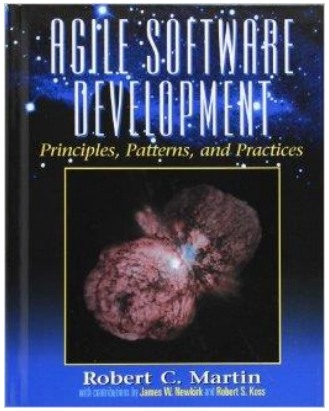
  - Erich Gamma, Richard Helm, Ralph Johnson, and John M. Vlissides,

  Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional. 1994.

Text Book and Reading Materials

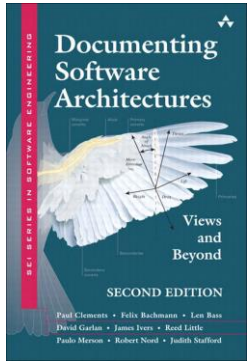
- Recommended Textbooks:
  - R. C. Martin

    Agile Software Development: Principles, Patterns and Practices, 1st ed., Pearson, 2002



  - P. Clements et al.

    Documenting Software Architectures: Views and Beyond, 2nd ed., Pearson Education, 2010



Text Book and Reading Materials

- Recommended Reading:
  - UML 2 Tutorial.
    http://www.sparxsystems.com/resources/uml2_tutorial/
  - git manual.
    https://git-scm.com/documentation
  - Design patterns basic.
    http://www.oodesign.com/

Project

- See project specifications in the course space.

- Project teams:
  - Try to find a team by yourself first. 4-5 people is preferred unless exceptions need to be made by me.
  - One member in your team would serve as a liaison for the team, and will be responsible for the communications of your team with me.
  - Have the team liaison submit the team and liaison information in their submission of Milestone 0.
  - For those who haven't found a team after the due date, I'll assign you into a team.

EECS gitlab

- We will use EECS' gitlab server to host all your project codes.
  - Address: gitlab.eecs.wsu.edu
  - Use your EECS account to log in to this server at least once before Milestone 0 due date.
  - If you don’t have, or don’t know your EECS account, see the help desk links on Canvas
    - Make sure you are able to log in to the gitlab server address at least once after it’s set up.
    - It’s not the same as your mywsu login.

FYI slides on Git usage

Coding with a Team

- What kind of problems did you encounter with your coding assignment/project(s) before?

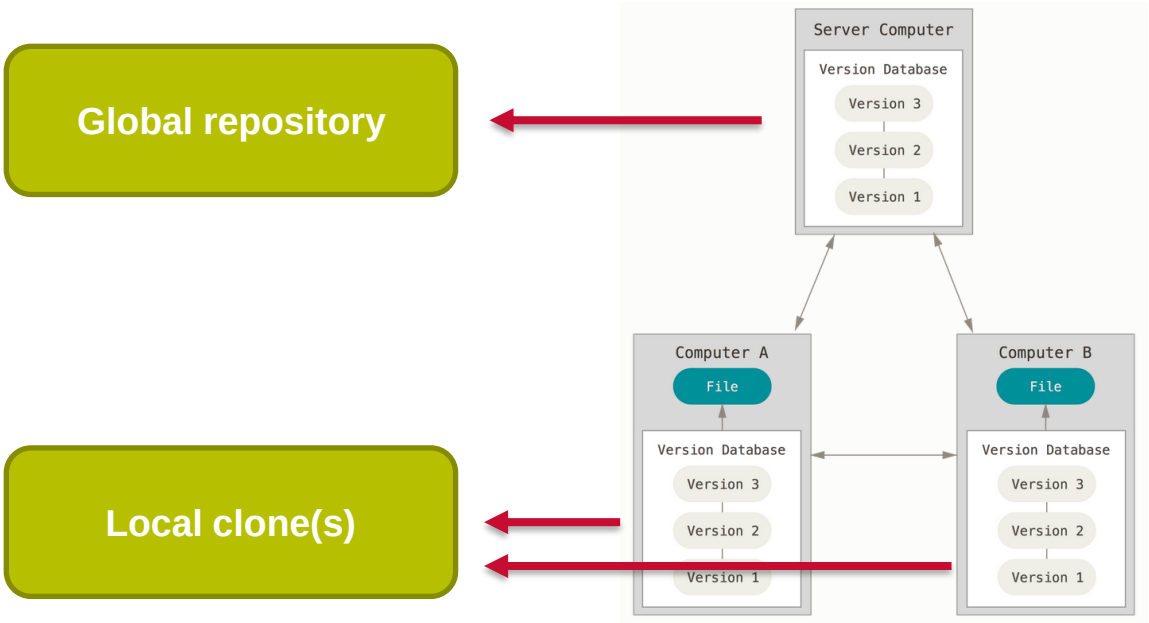
- Imagine what kind of problems will you encounter with your team when coding the projects?

Version control system

- Quote from git manual:
  - Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.
- Applicable for any kind of files.
  - Source code.
  - Keeps full history of all work done.
- Various systems available
  - Subversion
  - CVS
  - Mercurial
  - Git
    - New standard. A must-learn.

Introducing Git

- Git
  - Resources
    - Documentation: https://git-scm.com/doc
      - Rule book for everything git-related.
    - Google and Stackoverflow
  - Support local and distributed version control



Create a repository

- Init
  - Takes an existing project or directory and imports into Git.
  - Refer to the manual.
- Create one via github/gitlab*
  - Github/gitlab are services that host your repositories for you.
  - github.com is a public server
    - Have an account
    - Move your past/future projects/assignments onto github.com
    - Include the link in your resume!
  - gitlab.eecs.wsu.edu is the EECS server.
    - For our course's projects.
- .gitignore

Clone a repository

- git clone [your repository location]
  - SSH (requires public/private key)
  - HTTPS
- Clones the repository into your local machine.
  - Prompt for username and password
  - You can find plugin to store them

Work with Git

- Basic usage:
  - Commit your work in 3 steps
    - git add [filename]
      - //add the files you want to include for commit to the repository. Use ‘git status’ to see what files you have changed/added.
    - git commit –m ‘/*messages*/’
      - //have the commit ready, attached with messages you want to include with the commit.
    - git push
      - //push the commit to the global repository
  - Get other people’s work into your local clone.
    - git pull

Work with Git and conflicts

- Branch!
- Default branch: master
- Create a new branch when working on a new feature or a fix
  - git checkout –b [branch name]
    - Or: git branch [branch name]
      git checkout [branch name]
- Concept of “working branch”
- Work on the new branch as usual.
- Then merge
  - Switch to the branch A that you want to merge branch B into.
  - git checkout A
    git merge B
- Read more on branch and merge*
  - https://git-scm.com/book/en/v2/Git-Branching-Basic-Branching-and-Merging

More with Git

- Refer to the Git manual.
  - Remove a file(s) from the repository.
  - Revoke a commit/merge, etc.
  - Move the branch back to a previous commit etc.

Other issues

- Gitlab (and other similar repo hosting services)
  - Issue tracking
  - Milestone management
  - Assigning to responsible party
  - Code review
    - comment on commit changes
  - Etc…

Summary on git/github

- What you absolutely need to learn for this course:
  - Init, Add, Commit, Status, Diff, Rm, Log and Installation
  - GitHub, Remote Repositories, Push, Pull, Remote, Fetch, and Clone
  - Branching, HEAD
  - ".gitignore"
    - https://git-scm.com/docs/gitignore
    - https://github.com/github/gitignore

- Therefore, watch the series of git tutorial here: finish and understand the first two asap, and the third one for branching
  - 1: http://www.newthinktank.com/2014/04/git-video-tutorial/
  - 2: http://www.newthinktank.com/2014/04/git-video-tutorial-2/
  - 3: http://www.newthinktank.com/2014/04/git-video-tutorial-3/
  - 4: http://www.newthinktank.com/2014/05/git-video-tutorial-4/

Additional Resource

- MonoGame tutorials
  - https://www.gamefromscratch.com/page/MonoGame-Tutorial-Series.aspx
  - There might be an issue with your MonoGame installation to Visual Studio if your path contains a space. If your MonoGame is not working, please let me know as I've fixed this before.
- Libgdx tutorials
  - http://www.gamefromscratch.com/page/LibGDX-Tutorial-series.aspx
- First project deliverable is in 5 weeks.
  - It's not that far away! ☺