Striving for successful team projects

Professor Hossein Saiedian

EECS 348: Software Engineering

Fall 2024

Organization

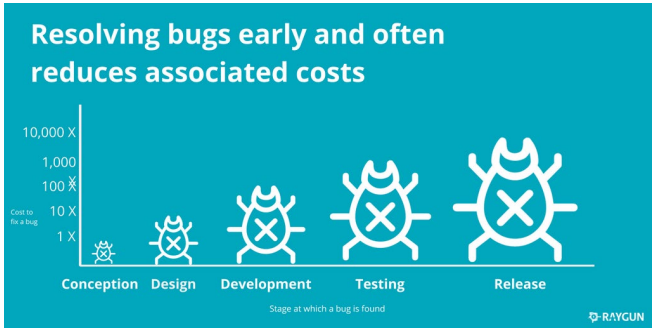
- Technical, managerial, team-work, and communication competencies
- Choosing a process
- Artifacts and deliverables

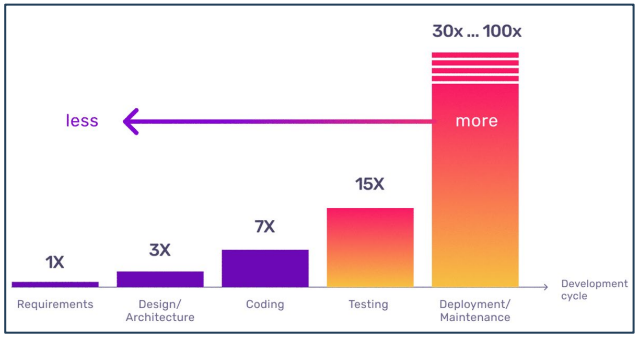
Fall 2024 project

- Project objective: arithmetic expression parser
  - **Develop an expression parser**: Create a program to parse and evaluate arithmetic expressions with operators (+, -, *, /, %, **) and numeric constants
  - **Handle parentheses:** Ensure the parser correctly interprets expressions with parentheses to define precedence and grouping
  - **Focus on process**: Deliverables include a project plan, requirements document, design document, and rigorous test cases
  - **Integration into larger product**: Your parser component will be integrated into a larger compiler, ensuring coherence with overall design and requirements

Fall 2024 project

- The emphasis is on the process
  - **Understand the development lifecycle**: Focus on the entire process from planning through delivery, not just the coding phase.
  - **Engage with each step**: Fully grasp the tasks and artifacts required at every stage: requirements gathering, design, implementation, testing, and documentation
  - **Prioritize quality and process**: Ensure each phase is executed thoughtfully to produce high-quality results and effective project management.





Technical requirements

- **Real world characteristics**. A popular domain, compilers design
- **Programming language/Platform**: C++ on iOS, Android, Linux, Windows, cloud-computing platform (e.g., AWS)
- **Scope**: a semester project scope but with defined deliverables throughout the semester
- **Innovative**. For example, a cloud-based, a mobile or Web application with a modern GUI that allows localization, a database component, and provisions for security and privacy (e.g., authentication)

Managerial requirements

- **Organize the team**: Define roles and responsibilities clearly to ensure everyone knows their tasks and contributions
- **Schedule weekly meetings**: Set regular meeting days and times to discuss progress, address issues, and plan next steps
- **Divide the work**: Break down the project into manageable tasks and assign them based on team members' strengths and interests
- **Maintain meeting logs**: Keep a record of meeting discussions, decisions, and action items to track progress and follow up effectively

Why team projects [1/4]

- Practically all real-world software products in industry are designed and implemented by team of individuals
- Need to develop coordination and communication skills required in real-world
  - Team meetings (with well-defined agenda)
  - In lab and outside lab
  - Verbal, F2F discussions
  - Presentations
  - Emails

Why team projects [2/4]

- Team members will have to do their part of the project using another member's specs
  - Generates substantive discussion among team members
  - Exposes team members to the challenges of integrating someone else's work
  - Further emphasizes communication and clear documentation

Why team projects [3/4]

- Because of the nature of team development and team environment, other issues emerge that will require
  - Egoless programming/development trait
  - Assessing, evaluating, respecting and appreciating one's own and others' strengths and weaknesses
  - Adjusting accordingly

Why team projects [4/4]

- Team members realize that they "can't do it all" and have to depend on each other
  - All teams are equally responsible to contribute to development and maintenance of both technical and managerial efforts
- Team-oriented grade
  - Some may be better programmers, some may be better administrator, …, but all must contribute equally
  - All are responsible for contributing to technical and organizational tasks
  - All are responsible for the overall project quality
  - Team member assessments will impact the project grade

Team members' assessment

- Team assessment
  - Real world: Once a term project is completed, a project manager would have to evaluate the performance and contributions of their developers
  - At the end of the semester, team members will play the role of a manager to evaluate other team members contributions to the project
  - Team member evaluations will be confidential
- **Final grade project: determined by team assessment**
  - 25% of the final project grade: completing the team members peer evaluations
  - If not completing the team assessment, will lose 25% of the project grade automatically; team evaluation may further reduce the grade

Professionalism is important

- Present all project artifacts in high-quality artifacts
  - Common cloud repository (GitHub)
  - Submit URL to Canvas by the team administrator
- A great asset for job interviews

PDF or HTML

Green Sustainability Software Project Artifacts

This page will include software project management plan artifacts for our project entitled Green Sustainability Software (GSS). The GSS project management plan is organized using the xxxx template and includes the following artifacts (or chapters or sections or parts or documents).

Artifact: Vision statement and team profile
Date created: February 15, 2021

Artifact: Stakeholder management
Date created: February 22, 2021

Artifact: Project charter and requirements
Date created: March 1, 2021

...

What to include in a GitHub portfolio

- Information about the team members
  - Name, photo, and contact
  - Expertise (programming languages, operating systems, computing platforms, …)
  - Available times
  - Contact info
- All project artifact (including test cases, and user's manual)
- The final C++ code

Use a standard template

- While working on an IT (software) project, many intermediate project artifacts may have to be developed
  - Project management plan
  - Requirements definition document
  - Software architecture document
  - Test cases
  - Coding
  - More …

Why a standard templates

- A template defines the overall look, layout, and structure of an artifact
- A template indicates what to be included, thus facilitates completeness
- Examples
  - IEEE/ISO standards for software engineering
  - DoD standards for software engineering
  - Organizational standards for software engineering
  - IBM's Unified Process
    * A simplified version of UP for education called UPEDU

Use Unified Process (or UP for EDU)

- Project management plan
- Project requirements
- Project design
- Project test cases
- …
- A greatly simplified version of IBM's Unified Process
- All available on Canvas

Keep a log of *all* activities

- You are required to keep a log of all project activities
  - Team meetings and the main discussion points
  - A record of division of responsibilities
  - A record what everyone did for the project
- Many professional industries require this
- Good for individual team members: how their time is spent

A fun and rewarding process

- Must be dedicated and enthusiastic
- Make it a fun and rewarding process
- Learn and develop real-world experiences and skills

A brief intro to Git and GitHub

KU THE UNIVERSITY OF KANSAS

- Git is a version control system (VCS)
- Allows to maintain multiple versions of a code base
  - Keeps a history of previous changes
  - Let's you see the changes you make to your code and easily revert them
  - Sometimes across multiple developers
    * Collaborate with other developers
    * Push and pull code from repositories such as GitHub
- Available on EECS Linux Cycle servers
- Available for installation on Windows and Mac machines

What is GitHub

- GitHub.com is a website server that hosts git repositories
- Hosting repositories facilitates the sharing of codebases among teams by providing a GUI to easily clone repos to a local machine
- When you push your code repositories on GitHub, you will be creating your own developer's portfolio
- Lots of resources online
  - https://github.s3.amazonaws.com/media/progit.en.pdf

Summary

- Understanding a process
  - Phases, milestones
- Teamwork, communication, collaboration
- Log of activities
  - Consistent with what appears on the project artifacts
- Use of a standard or template
  - Part of the learning process