

Phase 1 - Project Planning, Requirements Capture, and Tool Selection

Estimated time: 32-36 hours per person

Objectives

- Start working together as a group and begin to become familiar with each other's backgrounds and strengths
- Become familiar with the basic components of a project plan and their purpose
- Become familiar with several techniques for eliciting requirements
- Become familiar with at least one technique for documenting requirements without prescriptive design decisions
- Gain experience in selecting and setting up a tool stack

Overview

This is the first of four phases to the course project. During this phase, you will focus on establishing your team organization and procedures; gathering and documenting requirements; and setting up your tool stack.

Description

Each phase will be a little like a Sprint in an Agile method and a little like an iteration in a Spiral process. Specifically, each phase will be like a Sprint, in that work to be done will be organized into small tasks, placed into a sprint backlog, and prioritized. Then, using on time-box scheduling, the team will decide which tasks the phase (Sprint) will address. The team will use a GitHub Repository to keep track of tasks (issues) in the product backlog. Those tasks that will be part of the current Sprint will be kept in the GitHub Project, those in progress, and those that are done.

Each phase will also be a little like an iteration in a Spiral process, in that each phase will include some risk analysis and that any development activity (requirements capture, analysis, design, implementation, etc.) can be done during any phase. Early phases will focus on understanding (requirements capture and analysis) and subsequent phases will focus on design and implementation. Each phase will include a retrospective.

In this first phase, you will work together to do at least the following:

- Identify at least two people outside of your group who are willing to act as potential end users for your app.
- Select a tool stack and build some simple things (prototypes) with that tool stack for the purpose of making sure everyone in the group is familiar with each tool.
- Decide on your team's communication policies, procedures, and tools.
- Decide on your team's configuration management policies, procedures, and tools. Note that you must use Git and provide a GitHub repository for version control.
- Write your project plan using the lightweight template provided by the instructor
- Document
 - Tool stack
 - Setup instructions
 - Configuration management policies and procedures in your README.md file.
- Meet with end users to gather ideas about potential user goals and features that would fulfill those goals
- Document users (actors) and their goals in UML Use Case diagrams
- Document functional and non-functional requirements in a Requirements Definition Document

Deliverables

- A Git repository containing the following artifacts: Project plan, Use Case Diagrams, Requirements Definition, and README.md
- The initial project plan will include:
 - A summary of the project being built
 - A description of team organization,
 - A description of the overall software development process (provided by the instructor)
 - Policies, procedures, or tools for communication, including plans for team meetings, online-coordination, reporting, etc.
 - Risk analysis
 - reference to the README.md for the configuration management plan
- The README.md will include
 - An explanation of the organization and name scheme for the workspace
 - Version-control procedures
 - Tool stack description and setup procedure
 - Build instructions
 - Unit testing instructions
 - System testing instructions
 - Other development notes, as needed
- A set of Use Case Diagrams that document the actors and their goals
- The Requirements Definition will contain both functional and non-functional requirements for the envisioned project

Other Software Engineering Activities Only Lightly Touched On or Skipped

- Feasibility study
- Review of existing solutions
- User acceptance test plan
- Formal cost estimation
- A more thorough project plan
- Risk Analysis
- Quality assurance plan, including testing plan
- Documentation plan
- Rollout or deployment plan
- Relevant privacy, confidentiality and security policies
- Other configuration management tools and techniques

Submission and Grading

To complete this assignment, make sure all documents listed in the deliverables section are committed to your git repository under /docs/ (some may be MS Word documents) with the exception of the README.md which will be in the project root folder. This should be committed and merged to the MASTER branch and add a git tag with a tag of "Milestone_1". Submit the assignment in Canvas to let me know that the branch has been merged to MASTER with the appropriate tag.