Software Engineering Project Workshop (SENG202)

Matthias Galster

Introduction to Phase 1

July 15, 2020

UC
UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

Focus: Requirements analysis, initial design, project planning and setup

Goals

- [Get your team and development environment up and running]
- Get an idea of what your project will be about
- Identify features for your product
- Decide how you will know they satisfy stakeholders
- Outline solution approach

Understanding what customers want is one of the most difficult things in software projects

Deliverables

- Project setup checklist
- Design document
- Reflections and logs
- Presentation

Deliverables

- Project setup checklist
  - See template on Learn
  - Team and technical setup
  - Mostly about material in labs and tutorials, **but not synced**
  - To be submitted as PDF with other deliverables
    - `seng202_2020_team<team>_checklist.pdf`

- Design document

- Reflections and logs

- Presentation

Deliverables

- Project setup checklist

- Design document
  - More details later
  - To be submitted as PDF with other deliverables
    - Diagrams must be PDF. XMI, Visio files, etc. are not accepted
    - `seng202_2018_team<team>_report<phase>.pdf`

- Reflections and logs

- Presentation

Deliverables

- Project setup checklist
- Design document
- Reflections and logs
  - See next slides
- Presentation

Reflections (individual)

- Why
  - Snapshot of project from the perspective of individual
  - Basis for feedback, intervention
  - Quantify engagement and achievement
  - Identify challenges
  - Understand process
  - Guide logging, etc.

- How
  - Brief weekly reflection via questionnaire on Qualtrics (link on Learn)
  - Note closing time – add to your calendar
  - **5% penalty** for related Phase for each late weekly reflection
    - E.g., three late reflections for Phase 1 = 15% penalty for Phase 1

Logging (individual)

- Individual logging of tasks and time spent

- Clockify (https://clockify.me)
  - Email invitation (check spam filter)
  - Project name: Team<team>
  - Log as you go, no “bursts” – must be current at all times
  - Teaching team will be members

- Project, description and tags
  - Each session or activity of work as separate task item in log
  - No tags in description
  - Logging will impact assessment

Tags for categorizing work (1)

- #requirements
  - Worked on requirements, use cases, etc.
- #design
  - Worked on design-related tasks, modelling, etc.
- #coding
  - Wrote code that implemented some functionality
- #document
  - Worked on documentation, e.g. user guide, code documentation, Javadoc

Tags for categorizing work (2)

- #test
  - Wrote and ran unit and/or integration tests
- #testmanual
  - Manually tested the software, etc.
  - Should be low (automated test where practical)
- #fix
  - Fixed a bug
- #refactor
  - Refactoring as a result of a code-test-refactor process

Tags for categorizing work (3)

- #infrastructure
  - Ancillary tasks, e.g., configuring environment, build system, merge conflicts

- #designdoc
  - Worked on design document (description needs to mention what part)

Tags for categorizing work (4)

- #meeting
  - Attending team meeting or meeting with lecturers/tutors
  - Does not include working meeting – use other tags
- #class
  - Attending lecture and lab
- #other
  - Should not be necessary, avoid

Logging of tasks

- Per logged activity: one tag + proper description
  - Detailed description of task carried out (2-3 sentences)
    - E.g., class/feature worked on/tested, what code documented
  - No tag, no credit
    - Additional tags will be ignored (talk to us if they seem necessary)
  - Later: consider including git commit IDs in description

- Examples
  - #coding: “Wrote parser (Parser.java) to parse activity data and to store each activity in a list of activities”
  - #test: “Wrote unit tests for Flight.java; cover all methods (5) except getters/setters”
  - #fix: “Fixed bug in Location.java that caused crash when loading empty flight data”

Additional comments in log

- May add additional comments separately in design document
  - About self
  - About team
  - About individual team members

Deliverables

- Project setup checklist
- Design document
- Logs
- Presentation
  - See next slide

Presentation (Deliverable 1)

- ~15 minutes
  - During the labs of the week of the due date
  - All team members present, 25% penalty for not presenting

- Content
  - What is the system context and business vision/rationale
  - Who are your stakeholders and concerns
  - What are main use cases (no detailed requirements), with GUI prototypes
  - What are the key drivers/quality requirements
  - Example (major) acceptance tests (optional)
  - Overview of UML class diagram (patterns, layers, packages, etc.)
  - What are main risks (what could go wrong; how is it mitigated)
  - Overview of project plan, how will goal be achieved (optional)

How to submit

- Through Learn
  - By 5:00pm on the submission date
  - No drop-dead date

- As ZIP containing all deliverables
  - `seng202_2020_team<team>_phase<phase>.zip`
  - Must contain all the previously mentioned files and resources