CS189A
Software Engineering: Concepts and Practices

Jianwen Su
Department of Computer Science
UC Santa Barbara

https://capstone.cs.ucsb.edu/cs189a/cs189a_sched.html
Today’s Lecture

- Progress checking and upcoming deadlines

- Team meetings
  - MAN^2 (NAVAIR)
  - So Far So Good (Xenon)
  - 404: Team Name Not Found (Teladoc)
  - #Koki's Kookies (Invoca)
  - Panda (Alcon)
  - Sea++ (NAVSEA)
  - Inspector Royale (Bill)
Overall Plan @Week 8

- Four 2-week sprints (dates adjusted)
  - Oct 11-25 (PRD v1 – tools, technologies, design, terminology)
  - Oct 25-Nov 8 (use cases/user studies, prototyping, PRD v1, PRD v2)
  - Nov 8-22 (design, prototyping, testing, PRD v2)
  - Nov 22-Dec 3 (prototype demo/pres prep, prototyping and testing)

- Fall presentations and demos: Dec 3, 3:30-5:30 (Girvetz 1004)

- Specify what the product will do
  - Vision statement
  - Product Requirements Document (PRD) (due Oct 29 and Nov 26*/29)
  - Design tools, brainstorming, coding (tests and implementation)

- Build and test an initial prototype
  - Typically teams iterate on these activities until they converge to a working prototype!
This Week’s Plan

- Team activities
  - Scrum: Sprint 3, PRDv2
- Section: TA meetings

Upcoming deadlines:
- Nov 22: Sprint 3 ends (Sprint 4 starts)
  - Product Requirements Document v2
- Nov 29: PRDv2 due
- Dec 3: Fall presentations and demos
Sprints 3 and 4

- Sprint 3 starts today
- Break down stories into tasks & components associated with design
  - Prioritize stories
  - Assign timings to stories/use cases AND store/use-case tasks
  - Specify acceptance/test that can be used to verify a story is finished
- Sprint: Prototype tasks (primary implementation before demo)
  - Prioritize tasks
  - Assign timings to tasks
  - Specify what test(s) are to be used as evidence of task completion/acceptance (use case post condition OR user story acc test)
  - Each member/developer chooses task, implements, and tests task
  - Another member does code review/test and accepts the pull request
    - Test is the one specified above (Acceptance)
  - When store/case is complete, some member performs story test/acceptance
PRDv2: Your Living Requirements Document: A Shared Google Doc

- Authors, Team, Project Title
- Intro: problem, innovation, science, core technical advance (3+ pages)
  - Define project specifics, team goals/objectives, background, and assumptions
- System architecture overview
  - High level diagram (1 page)
  - User interaction and design (1+ pages) – ie detailed design
- Requirements (functional and non-functional)
  - User stories or use cases (links) → 20+ for PRDv2 prioritized w/acc. tests
  - Prototyping code, tests, metrics (10+ user stories): github commits/issues
- System models (1+ pages)
  - Contexts, interactions, structural, behavioral (UML)
  - Use cases, sequencing, event response, system state, classes/objects
- Appendices - Technologies employed
Your Project Design: PRDv2

- **Architecture (hardware/software)**
  - Evolve your overview picture from PRDv1 to provide significantly more detail and any updates or changes

- **Detailed design**
  - UML diagrams of primary data structures that comprise the system architecture connected via their associations (if any)
    - Ensure that each "class" is balanced in terms of cohesion & coupling
    - Annotate with pre/post conditions when appropriate
  - Sequence diagrams
    - Synchronous and asynchronous for key interactions between classes
      - At least 3 different interactions
    - User interactions with the system
      - At least 3 different interactions
      - Can be a human user or a machine user (API) interaction
        - Event response, updated application state
      - If you have a user interface: **Provide mockups for primary UIs**
PRDv2 User Stories / Use Cases

- Revise spec to add detail to the functional specification to match your design
- Add user stories and break up the stories you have into finer grained stories
  - Provide UML, sequence diagrams, dataflow diagrams
  - Goal: a CS senior should be able to take your doc and implement the project
- For each fine-grained story, provide a description and acceptance test
  - Provide time estimates (1 person-hours) for each story implementation
    - Ensure you can finish the implementation in the time you have (this/next quarter)
  - Prioritize tasks to have a complete prototype by the end of this quarter
    - Focus on the externally facing interfaces, mock out what you cannot get to
  - Write unit tests to implement tasks for mandatory tasks
    - Document these tasks (autogen the documentation/usage)
  - Add trello/pivotal task links (titles must match) to PRDv2 for each story
- Prototype designed mandatory tasks; add github commit ID/link to PRDv2
  - Github must have unit tests, documentation (for anything without unit tests), and prototyping implementations for each story in Sprint
- If you have a user interface
  - Provide mockups that are tied to the functionality described in 1+ components
Completing the Fall Quarter

- Nov 15: Team meetings
  - Meetings with 7 teams

- Nov 22: Potpurri
  Presentation/demo details
  looking ahead (break & Winter)
  - Meetings with 3-5 teams (to be invited)

- Nov 29: short class meeting

- Dec 3, 3:30-5:30, Project presentations with demo
  - Will be recorded: Check with your mentor if concerns
  - All mentors are invited:
    Team leads—please invite your mentors
Today’s Lecture

- Progress checking and upcoming deadlines

- Team meetings
  - MAN^2 (NAVAIR)
  - So Far So Good (Xenon)
  - 404: Team Name Not Found (Teladoc)
  - #Koki's Kookies (Invoca)
  - Panda (Alcon)
  - Sea++ (NAVSEA)
  - Inspector Royale (Bill)