Term project

CS3141
Fall 2010

Project #1

- A Calendar Software
  - Core Functions
    - multiple user access
    - control other user to read/modify your calendar
    - event management
    - meeting management
      - Schedule meeting
      - Invite people to the meeting
      - Check people's availability automatically
      - Allow individual to Deny/access meeting
    - Display and print as day/month view
  - Example:
    - Google calendar
      - www.google.com/calendar
    - Zimbra calendar
      - https://huskymail.mtu.edu/zimbra
    - Microsoft outlook
Project #2

• Galaxy Sleuth
  – Core functions
    • multiply players
    • Internet based
    • Game logic – similar to board game Clue

Project #3,4 …

• Any other project that you’ve been wanted to do?
  – Other games
    • Monopoly
    • Texas Hold’em Poker
  – Course Scheduler
  – Exam Master
Choose a project

- Project size is important
  - Not too big, not too small
    - Doable for 4-6 team member to finish it within
      - 8 hours/wk * 10 wk * # persons
  - Motivated is critical
    - switch to another team if necessary
- Comprising is needed
  - do you best to fit into the team

Class Structure

- Limited Lectures
  - Team work guidelines
  - Large software development guidelines
    - Waterfall
    - Agile
- Group work time
  - Monday is the group meeting time
  - W,F : the first 5 minutes, quick update
Team Leader

• Responsible
  – Track the team’s progress
  – Meeting organizer
  – Bridge between the team and the instructor
• One for each group
  – Instructor assigned
    • based on personal interest, previous experience
  – Can rotate between iterations.

Waterfall

• Software life cycle
  – Requirement analysis
  – Design
  – Implementation
  – Release
Agile development

• Many iterations
  – Each iteration is short (1, 2, 3 weeks, or a month)
  – Small piece of the product is DONE after each iteration
    • Tested
    • Documented
    • Can be released

• Some agile practices
  – Common work room
  – Daily Standup meeting
    • Test driven
    • Pair programming
  – More oral communication, less written documentation upfront.

Which process for your project?

• With stable requirements
  – Do extensive requirement analysis and design upfront is a very good choice
    • But it is not easy for inexperienced software engineers
    • And the design will have to be changed
    • But again, it is much better than no design, no planning
  – You are strongly encouraged to spend significant time to focus on designing before coding