CSE 430
Autumn 2013

Team Organization Plan Instructions
(Refer to slides in Lecture #5, "Teams")

Warm-up Thoughts for this Exercise:

Your team will be given a summary of the survey results for technical and other skills. This information is not to say that anyone is any "better" or "worse" than any other person at any particular skill, but just to give the team an idea of what level of resources are **perceived** to be **currently** available.

- **Perception:**
    - Many times, we can be our own harshest critics. Just because you are not perfect at something doesn't mean you are terrible at it.
    - Likewise, others may have different ideas than you about "levels" for quality of work (what is "ok" vs. "good" vs. "excellent"?) or quantity of work (what is "less than adequate" vs. "expected" vs. "above and beyond the call of duty"?).
- **Current Level of Skill:** With effort, you are adaptable. You can always work at improving a skill or learning something new. The amount of work necessary may vary between individuals and between skills or topics.
    - Contrast Carol Dweck's theories of learning: "incremental" vs. "entity" (http://www.learning-theories.com/self-theories-dweck.html). Which one do you think is more likely to take you farther in life?
    - If the team does not have many "Strongly Agree", "Agree", or "Neutral" answers for a skill that you think is critical for the success of the project, how as a team will you address that current deficit?
- **Your Contributions**: Everyone has some way(s) in which to contribute to the success of the team and the project. There are many **roles** (slide 5) and **functions or tasks** (slide 12) involved in a software development project. Based on your skills (or your desire to learn new ones),
    - where do you see yourself best contributing to this project?
    - where will you take the lead?
    - where will you have more of a supporting role?
    - note that team members most likely will have to take on different roles throughout the project

Specific to the Plan Your Team Will Turn in Friday October 10, 2013:

Key Issues of Team Organization to Address (slide 2)

Management/Leadership: There are 2 major tasks that happen simultaneously during project development:

One main task is that people need to be organized and project status needs to be monitored. The **Project Manager** role usually has these responsibilities.

- Who is supposed to be doing what by when?
- What progress, and how much progress, has been made to date?
- What is still left to do, and what is the immediate next thing to be done?
- What is everyone's current workload in case adjustments need to be made?
- How will everyone be informed of meetings, decisions, work assignments, status updates, and changes?

The other main task is that the architecture of the software needs to be decided, and the software system decomposed into small enough pieces that can be assigned to individuals or small sub-sets of the team. Persons in the **Technical Lead** role understand the overall architecture and decomposition of the system, and the dependencies between parts, so that the Project Manager can schedule work to be performed on the most critical pieces first.

We will not force your team to have a single person be a Project Manager, but it is a good idea to have some person or persons formally take on these responsibilities. Decide amongst your team how you will handle this task (one person, split amongst several people, rotate through people, etc.) and who will do it.

Likewise, it is a good idea to have some person or persons formally take on Technical Lead responsibilities. You may find it easier to have different people take the technical lead on subsystems, especially if there is a difference in technical skill level among the team. Decide amongst your team how you will handle this task and who will do it.

Decision-Making Process and Team Structure (slides 6, 7, 8, 9, and 10)

Related to both management and leadership. Decide amongst your team how decisions will be made, and who makes them. Decide amongst your team if you will have a more hierarchical structure or a more "democratic" or "flat" one, or something in-between. Google "software engineering team structure" for ideas and images; also consider Mike Cohn's thoughts on team formation: (http://www.mountaingoatsoftware.com/blog/nine-questions-to-assess-team-structure)

Communication

Good communication is **vital** to the success of your team and your project. We expect that every team will maintain a wiki to archive documentation and communicate important decisions and changes. How else will your team communicate (email, phone, chat, etc.)?

In particular:

- When and where will face-to-face team meetings be held?
    - Who will take notes?
    - When will meeting minutes be added to the wiki? By whom?
- What are the consequences if someone misses a meeting?
- Who will finalize documents such as requirements or design? Who will add them to the wiki?

Division of Labor

Can be related to team structure, roles, and architecture.

- Will each person work independently?
- In pairs?
- In smaller sub-teams?
- Will partners change throughout the project depending on what is being worked on?
- Who is responsible for doing what, with whom?