

Managing a Small Software Team

or

“How to build a 6.170 final project with your friends and still be friends at the end of the term”

Today's Agenda

- Meetings that matter
- Ideas that influence
- Visions of versioning

Meetings that matter

At every team meeting, you should assign members to one of four roles:

1. Benign dictator

- *Four people are not enough for a democracy*
- Someone needs to be in charge and willing to make tough decisions when teammates disagree. Flipping a coin is usually a bad way to break a 2-2 tie. Always be sensitive to others' conflicting ideas, but in the end there can be only one. The other teammates must recognize the dictator's authority for the duration of the meeting.

2. Grand inquisitor

- *Speak now or forever choke on your dog food*
- Few ideas are perfect when they are first born. Someone needs to be consciously thinking about tough questions to challenge new ideas before they are accepted. Teams dominated by one-person newborn ideas will have lackluster designs at best. The other teammates must not be offended; the GI interrogates ideas, not the people who spawned them.

Meetings that matter, cont.

3. Timekeeper

- *Stick to the agenda*
- Always have an agenda with a list of goals, discussion points, and actionable items. The timekeeper is responsible for keeping an eye on the time and informing the group when they need to make a decision and move onto the next item. Don't make your meetings interminable. There's no light at the end of the tunnel unless there's a track to drive on and a conductor tossing more coal into the boiler.

4. Note taker

- *"A meeting without minutes never happened"*
- Every idea, design decision, justification, and task should be meticulously recorded and distributed to the group. Without notes, your discussions will run in circles and your progress will move slower than molasses.

Ideas that influence

1. Do not attach your ego to your ideas.

You should not be afraid to share ideas with your team, and you should not try to maintain ownership of an idea once you have shared it.

2. People attach their egos to their ideas, so be sensitive when you critique them.

Do not be afraid to critique an idea, especially if you are the Grand Inquisitor. However, people forget Rule 1, so be sensitive.

3. Treat newborn ideas like newborn babies.

Newborn ideas are like newborn babies. They are neither good nor bad, just inexperienced. It takes a village to raise a child, and it takes a team to cultivate an idea.

Visions of versioning

- **Update, edit, commit, repeat.**

Some of you may have gotten away with waiting until you were finished with the problem set to commit your code. Now that you are working with a team, you should use CVS to protect your code base and keep everyone up to date.

- **Don't break the build.**

Always update to get the latest changes whenever you begin working, and always commit whenever you have completed a small chunk of work, provided that it does not introduce REGRESSIONS.

- **Write meaningful commit messages so you can easily track your changes.**

- **Consider e-mail notifications on commit.**

Consider configuring your repository to send your teammates e-mail notifications with your commit record.

Last minute tips

- **Apply your design to your code repository.**

Just as you should not hack your code, you should not hack your code repository either. Appoint someone to be in charge of organizing the source, doc, and resources branches within your repository tree.

- **Store meeting and other notes in CVS.**

CVS should include all of your documentation, notes, and tests as well.

- **Use your MDD to define tasks and give yourselves assignments and deadlines.**

The MDD should be used to define package hierarchies as well as class modules. Choose package and class names carefully; renaming files in CVS and software systems is undesirable and difficult.

- **Document as you go.**

Don't save your documentation until the end. Document as you go, and not only will it not be a looming chore on the due date, but it will clarify your development process as well.

Last minute tips

- **Schedule meetings and group work time.**

- **Think about usability from the beginning.**

The UI is not just a skin; it requires a clean underlying model. Don't assign it to just one person, and don't put it off to the last minute.

- **Try to get it right the first time instead of wasting time “adding quality in” later.**
- **Consult with your TA when you get into trouble.**