Presentation #2:

Presentation Brainstorming

- The process of choosing an issue (fairly recent, third-party?, etc).
- Post-Mortem of the ‘set minus’ issue. (mathjax,
- Determining who are your stakeholders.
- User Stories.
- Redmine for SCRUM Agile Development/Debugging

Ask ourselves, what do we have to offer the other groups? What do we have to teach them? What valuable things have we learned that we should be telling them?

Scoping

Devs

- Tasks: well defined and separate
- Individual teams
- Logging time
- Issue investigation

Nugget: Prioritizing work lets us work more efficiently

Tool Usage

- Gathering tools
    - Platform tools: vagrant, github
    - Nugget: Working with a singular environment allows for uniform process
    - Design/Communication tools: Slack, redmine
    - Nugget:
- Separating communication tool from project management tool

Nugget: A well-managed workbench is the foundation of every project.

SCRUM Master

- Project management
- Stand ups
- Sprint organization

Nugget: Organization is key to progress.
Nugget: Teams need to be flexible and willing to change roles as the circumstances demand it.

Documentation

- User stories
- Ticket notes
- Wiki

Nugget: Comprehensive documentation allows for transparent process.

More on devs:

The individual teams that we split into helped us prioritize our work and work more efficiently. Each of our divided teams made a lot of progress and clearly communicated with the rest of the group during our meetings.

We saved a lot of time with investigating issues when we were split into teams. When working with only one other person we had a constant stream of communication with our progress and ideas we had while working on an issue. We were also able to take on more issues at once.

More on documentation:

The notes in the tickets offer experiences we can learn from. This reduces the amount of work that needs to be done when we approach issues that are somewhat similar in the future. They also make our process transparent so we know how we are approaching an issue and how we can change our approach in the future.

The user stories clearly outline our intentions for this project. With these requirements set we have a clear direction for the course of our process.

The same applies to the wiki. The information we provide there displays our vision and intentions.

You have to have a Scrum meeting follow the Scrum methodology. (Luke)

Title: Never just be a person who does stuff. (not as humorous as we thought…)
Overall topic: Team and process maintenance. (and what we've learned from it)
(CJ)

Nugget: Doing daily standups allowed our team to remain focused. (Luke)
Nugget: A process management tool is not a bulletin board.(Manoj)
Nugget: Clearly defining roles helped us to clearly define tasks.(Stephen)
Nugget: Slack fosters bad process communication habits. (Holly)

Conclusion: (CJ)

Speech Outline:

Presentation #3:

Ideas

- Submitting issue process? Gerrit and code review?
    - Too redundant?
    - What unique perspective can we add on this?
- Industry stuff?
    - Things we learned from working that ties into this project. Sharing our own experiences with working and sharing with one another to gain value from our eachother.
    - How industry stuff connects to the open source community.
    - Sprint review
    - Learned how to enforcing process model
- Using documentation.
- Experience: This is what I taught my team:
- Relying on each others experience: Teams have to rely on each others experience to be effective.
    - Enforcing a structure: CJ

  CJ: I learned about enforcing a given structure from the manager at my company. CJ: Creating and adhering to a structure made our team more professional and more understood by the entire company.

  I learned how to hold myself to a structure from CJ.
    - Sprint review: Manoj
    - Conflict resolution: Stephen

  Steve: I learned how to help resolve conflicts by bringing up unasked questions by trying different strategies in previous semesters.
  CJ? Asking the questions that are usually assumed and needed.
    - Prototyping/Testing/Environments: Luke

    - Sprint Review: Manoj
    - Getting stuff together, morale, deadlines: Holly

LUKE:
I learned about being thorough and quick in communication with stakeholders so we can avoid confusion in how we convey a message.

Luke spent the time and was quick in communicating with the developers active on the forum for the issue we were working on. He was able to constantly look for updates and present our thoughts clearly so the developers understand our findings and progress. He helped me present our progress with our current issue while clearly identifying both our progress and roadblocks. He knows how to effectively tell the client what he knows and how to get more information from the client.

Luke learned this thorough and prompt communication through his work as a small business web developer in a group of 5.

Transition from Manoj to CJ: Now CJ will talk about what he learned from Stephen.

HOLLY:
I learned about keeping your priorities straight. It’s so easy to get absorbed in an issue and lose sight of what needs to be done. The importance of adhering to a process and caring about the project. Testing before you submit your work.-

Holly learned this from her experience working with a space systems company (Lockheed Martin) and how easy for projects to get behind for not reaching deadlines and/or sloppy by not adhering to conventions. She was also caught on the receiving end to redo people’s work since some didn’t test their work prior to submission.

Nugget type summations
CJ from Steve - adhere to a structure and hold others accountable - learned from his manager at company that started to enforce structure and made team more professional.

Manoj from Luke - through and quick in communication with stakeholders - learned from job as small business web developer - small environment so more responsibility.

Holly from Luke - Keeping priorities straight - learned from Lockheed that projects are easy to get behind

Steve from CJ - Open with concerns and questions - learned from working in school groups bc people don’t want to argue but you have to ask the questions that no one wants to ask.

Conclusion: Teams naturally have a wide range of experience and we believe that it is important to take advantage of these experiences. You have to utilize the experience of your teammates.

Presentation 3 slides:
https://docs.google.com/presentation/d/1YVl4nfA-BvMqj_MfI_LvK4BOaVJPNTqYdJJWqpeLyns/edit?usp=sharing

PRESENTATION 4

Team building exercises
Staying motivated
Tasking and accountability, team structure.

Accountability:
Enforcing, how to do?
Responsibility, structure
Metrics - how to check: kind of do it in standups, acceptance criteria, deadlines, reflections including sprint review.

Examples of how we enforce accountable =
mathoid, notifying stakeholders instead of just walking away
Creating tasks with deadlines and acceptance criteria to know when something is done.
Standups made it so we have to say what we were supposed to do and we actually did so have to tell teammates to their faces if they didn’t get things done. Sprint reflections to talk about progress.
Adhering to testing - regression testing.

Additional presentation thoughts:
-Talking about the difficulty with using RedMine.
-The importance of using a good tool.