Total 77/100

Personas: _16__ / 20

Format: _3_ / 4

must contain: gender, age, personality, skills, environment, attitude towards technology in general, attitude towards computer software, goals when using the system

Too many irrelevant details are not useful!

-1

Professor X: No personality, goals
Student Y: No personality, no goals

Information content: _5_ / 8

-1 Need to describe Prof X's ability to use git / github

-2 Not clear how student Y's relevant, grad student but not really related to Prof X… Can the Prof trust this guy?

Coverage: _8__ / 8

personas must include at least two of the following type of users:

-- "Hanno type"

-- Hanno's grad student: can approve modifications to the data, but cannot add other such users (only Hanno can)

-- A less tech-savvy astronomer: Sure, Hanno is the client who makes the order, but the software will be released as open-source, so other types of users are 100% realistic.

Product Backlog: _36_ / 50

_10_ / 10 : User stories follow the format "As __name__ (__role__), I want __something__." ("so that __benefit__" is optional) AND each __name__ corresponds to one of the personas.

_5_ / 10 : User stories, together, reflect all the user requirements (from all prospective users).

-5, Good start, covered most requirements, need to clarify on 5 maybe should think about things like admin access? And what happens if the data isn’t really coherent/ what you expected and draft those out as user stories. Also could a user edit existing data? How would you integrate pull requests?

_9_ / 10 : User stories correspond to actual requirements (no "invented" features

-1, 7 seems invented

_5_ / 10 : User stories contain enough information for devs to estimate how long it would take to implement it.

-5, generally has enough details, focus more on 5, the merging feature.

1. Good, im assuming you’re using git commits
2. What am i running exactly, a script? A command? A set of commands? A webserver?
3. Okay
4. So do you send one email per new change? One big summary email? What about changes that weren’t approved but was brought up again?
5. Have to define ‘accurately compare’, what kind of typos are we expecting, where are we expecting typos? In which fields? What happens if a planet’s named something else but the data is the same as yours? Is that a new planet then? Have to elaborate more on this.
6. Ok
7. Define ‘guide me’, how much assistance is required? Are we talking about a video tutorial, a tutorial walkthrough? Documentation?
8. What do you mean by ‘give’ an application? Send them a link? Send them the binaries? Do i have to assign them write permissions first? Is it done through the app?
9. Ok
10. Are we highlighting some details / differences? Or are we just dumping two versions of a planet from 2 different sources into 1 table?
11. Ok, see above
12. What constitutes a reference? Just the catalogue source?
13. What are these standard units? What units do the other databases use? So the user gets to choose what units before adding/updating?

_7_ / 10 : Each user story addresses one specific requirement (no "world peace" stories).

-3

2, 5, 8, were as ‘world peace’ as it gets

Presentation: __17_ / 20

_4__ / 4 : well-presented, is easy to read and to navigate

__3_ / 4 : spelling and grammar

__8_ / 10 : looks professional.

__2_ / 2 : quality of the README file

Interview: _8__ / 10

1. Come up with idea of what’s expected
    a. Each person thinks of personas & user stories on their own
    b. And then collaborated together on gdoc
    c. Kent & Keerush copied content over
2. Process for coming up with personas
    a. Focused primarily on the prof, and then added the grad/ta position
    b. They prepared questions in the tutorial and they were answered in the tutorial.
3. Process of coming up with the user stories
    a. Heavily relied on presentation
4. Details of User Stories -2
    a. Wasn’t sure why they needed a login system, just said they stated.
    b. Needed help to remember other user stories (3 answers were assisted)
    c. Keerush was able to answer the rest of the questions
5. Sprint backlog
    a. Pulling data is done first
    b. Order by technical orders,
    c. They would ask the client as to which one they would want