

1. What is exactly a work item.

Gabriel: It's "everything" that can be created in a work process inside a work cycle (an iteration): a story, a requirement, a defect, a task. The system should not force any work-process (be it agile or not), so we made the decision to keep it simple enough so the final user can adapt it to his/her own workflow (by using a certain name pattern to separate defects and requirements, for example).

2. Why a work item could be linked with another one.

Gabriel: the decision to make the system as simple as possible drove us to also make it flexible. We wanted to be able to break a requirement (work item) into smaller tasks (other work items), or make a requirement have a dependency with other requirement (work items also). This is why the link-to-work-item feature shall exist.

3. The user can choose the new iteration name, if he/she doesn't put anything we assigned for example the name "Iteration 06" where "06" is the number of how many iterations we have (with the new one) for a given project.

Gabriel: In order to keep the system simplicity, we decided to not allow the user to define the Iteration name and just make the system create "iteration N". If you folks think it would be useful to allow that, we can talk further and create a couple more UCs with that in mind.

4. What is exact the role of each user: his/her profession or the role in the project?

Gabriel: In order to keep the system simplicity, we don't have many roles: we have the owner of the project (who receives some notifications, can change the project and can promote members to owners) and the members (who don't receive notifications and can't modify the project name). Project role is not a concept that we've used: all are members, Devs, QAs, BAs, Ops, etc.

5. What is the means of "Backlog iteration": the history of the project or the first iteration for a certain project?

Gabriel: The client asked us to have the Scrum's concept of a Product Backlog, a place where "someone" could create "something" and not necessarily attach to any work cycle (Iteration). So our Backlog Iteration was born, a single Iteration that is created with the Project and can't be deleted. Lauriane will remember the exact phrase that led us to that decision, for sure :)

6. what are the point of each work item? The requirement document said that when the user creates a new work item he/she inserts these points. We didn't understand after that how we should manage it.

Gabriel: Story/Task points is another Scrum concept that we had to have. By Scrum perspective, "points" stands for "complexity" and thus all our Work Items should be ordered by points in descendent order (higher to lower), putting at evidence the most complex story first. However, since we didn't put any restriction to that particular field, a creative user can put there "hours" or "days" or anything to effectively rank all his/her work items (the semantics of the field is not restricted by the system). Also, the user can drop this field altogether (by putting "0" everywhere). You folks can limit

it on positive integers range only (points would vary from 0 to N, be it N the max_integer), but we didn't write it down anywhere

Lauriane - adding...

These points should be displayed on the dashboard too, because this information give us the projects status.

"System loads all open work items owned by this account (State = Not started/Ongoing), ordered by project and have a "+" / "-" to expand/collapse. These listed projects should have a status field where there are projects statuses (Done per Not done/Not deleted).

Iterations are in an expanded Project, ordered by most recent iteration created and have a "+" / "-" to expand/collapse. The open work items should be in links to redirect to their work item page;"

7. Regarding the modification of a work_item, when someone change the iteration, he could choose only among the iteration of the original project or among all iterations of all projects?

1st option. Our Design does not allow to move a WI between projects, just between Iterations