Addressing Team Awareness By Means Of A Requirement Prioritization Tool

Paolo Busetta

Delta Informatica, Trento, Italy
paolo.busetta@deltainformatica.eu

Abstract. The progress of PRESTO, a R&D project by Delta Informatica that ran between 2013 and 2016, was often hampered by misalignments of perceived project's goals between management and development team as well as disagreements on the importance of features, in turn often caused by different opinions on stakeholders. To improve this situation, Delta Informatica decided to experiment the introduction of an agile but structured development process, which included the adoption of a software tool for requirement prioritization created by the SUPERSEDE EU project, on a specific sub-project. In this short report we focus on the social and collaborative experience, leaving scientific conclusions out. We remark that the mere utilization of a tool, by forcing the complete and clear formulation of available options and an appreciation of how they would affect different users, improved awareness and reduced the level of disagreement, so its effects on the group went beyond a purely mathematical exercise. By contrast, it is harder to provide a comparably unequivocal feedback on the tool's impact on the engineering process and on the quality of its results.

Keywords: user centred design, team awareness, conflict negotiation

1 Introduction and Context

PRESTO (Plausible Representation of Emergency Scenarios for Training Operations) [1][2][3] was a R&D project run by Delta Informatica in collaboration with a few research institutions between 2013 and 2016. PRESTO's objectives were the development of highly innovative tools in serious games and virtual reality derived from previous research experience.

PRESTO was often hampered by misalignments of perceived project's goals between management and development team as well as disagreements on the importance of features, in turn often caused by different opinions on stakeholders. The reasons for this situation were multiple but they essentially boiled down to two. The first was an insufficient participation of the end-users involved in the project's pilot study, whose initial interest was mild and thus their attitude was a typical show-me-what-you-have-to-offer rather than positive and pro-active; this attitude radically changed only during the final

phases when software artefacts were finally available and user value became evident. The second reason was a lack of a person with a strong, long-term vision driving the daily engineering activities.

To compensate this situation, Delta's management decided to adopt a hands-off, developer-driven approach to development based on rapid prototyping, in the assumption that having artefacts to experiment with would compensate the impossibility of writing proper requirements. It turned out that, in several cases, this approach led to the issues mentioned at the beginning, since lack of user participation made practical validation often impossible and lack of vision made external judgement on progress highly arguable.

During the development of a user interface, which was becoming a sort of battleground between opposite visions, management decided to switch to a different approach. A user-centred, agile, collaborative but better structured process was adopted, in collaboration with the University of Trento and FBK, the latter within the scope of the SUPERSEDE EU project [4]. This process, described in [5], included a series of workshops, led by a University representative, to which both management and development team participated. During the workshops, the team went thru a series of steps: (1) examination of existing prototypes (Figure 1 gives a feeling of their diversity, justified by very different ideas on their context of use); (2) identification of *personas* to represent the potential users (differentiated e.g. by technical preparation and own objectives); (3) creation of further mock-ups to complete the original set; (4) pinpointing of requirements and how they would better address each persona; (5) prioritization of the requirements according to the overall's group preferences by means of an experimental software tool created by SUPERSEDE; (6) agreement on the design of the final deliverable (see Figure 2 below) according to the preferences. The entire process started in the second half of January 2016 and completed by the end of March, with approximately one workshop per week; the prioritization activity was performed in early March.

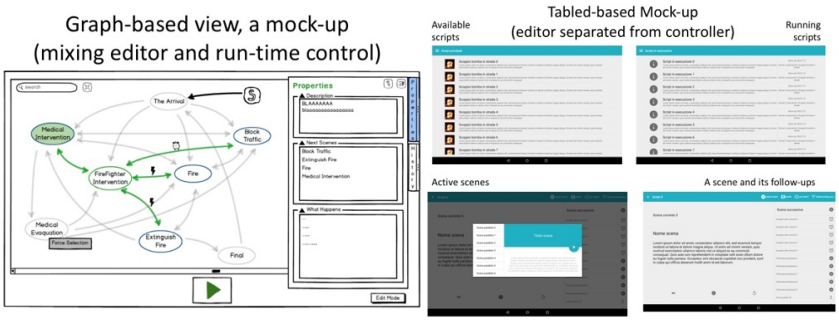

Figure 1: Two examples of alternative prototypes

2 Tool-mediated requirement prioritization

Requirement prioritization was key to the process outlined above, because it forced participants to clearly express what they thought the requirements were and to discuss

them publically before prioritizing them. This was done during the workshops prior to prioritization. The SUPERSEDE prioritization tool adopted pairwise comparisons of requirements, requested to each participant, to compute group preferences according to the AHP methodology. In summary, 8 developers with varying levels of experience were called to express their opinion. Further, inspired by a gamified approach, the prioritization tool involved a so-called negotiator, i.e. somebody that, rather than expressing a direct opinion, was called to solve major disagreements of opinions by either reaching consensus or deciding for all; the negotiator role was played by a manager. A set of 16 independent requirements were eventually chosen. In turn, they were grouped into 4 clusters, concerning general system requirements, game-time control, decisions on game progress, historical visualization and rolling back to previous states. Finally, requirements were prioritised according to two criteria: perceived impact on the user and development effort. The prioritization ran fairly smooth and fast; participants gave their own independent contribution via a Web interface invoked by their own desktop machine, with no need to coordinate with each other. The intervention of a negotiator wasn't necessary, since disagreements were relatively limited in degree. The results of the prioritization were quickly obtained when everybody eventually finished and they were circulated by the process' supervisor (the University consultant leading the process) to all participants.

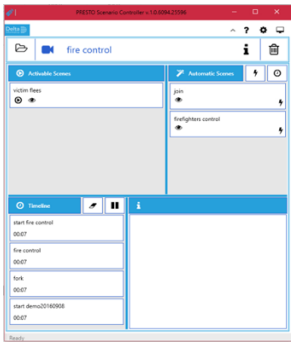

Figure 2: A snapshot of the delivered software

3 Observations and Considerations

Scientific observations on this experience of prioritization, including comparisons with other cases, are the objectives of work by the SUPERSEDE project. We focus here on the social and organizational impact of its introduction as perceived within the company. From this perspective, the most important observation is the final result was overall well accepted by participants; indeed, there was no major surprise in the relative order, and eventually most requirements were implemented anyway, leaving only the last ones out because the necessary effort was clearly not affordable within the remaining of the project. This observation may sound surprising, given the conflictual situation described above, and may create doubts about the sense of using a tool at all.

The critical point here is that the entire process described above was set up to confront and clarify opinions. Being in a collaborative setting, this process required time (some developers and management were strongly opinionated) but discussions were not considered a cause of conflicts by themselves because they were an opportunity to express perspectives that had never been clarified before. This process lead to a consensus to emerge spontaneously about the expected characteristics of the end-user, ratified by the application of the prioritization activity to the functional requirements for the system to be developed.

It is worth to highlight that most of the process could have been handled informally; reaching an agreement on who was the end-user and its working environment made team members aware of who they were working for, thus it would have been a clear improvement on the previous situation. However, using the SUPERSEDE tool forced the complete and clear formulation of the available options, something that was not part of the daily practice of the development team. This exercise helped in furtherly improving awareness; what's more, having an external, off-line tool acting as a sort of oracle removed psychological pressure due to negotiations and helped making the process fast in everybody's opinion. In other words, the effects on the group of applying a prioritization tool went beyond a purely mathematical exercise, and as such its introduction must be judged very positively.

Providing a comparably unequivocal feedback on the tool's impact on the overall engineering process (would a different tool or a manual computation have led to the same results? and in more or less time?) is very difficult. Similarly, it is hard to comment on its impact on the quality of the results of the engineering process. The author's opinion is that the delivered software (which is a user interface, as shown in the figures above) gives an unsatisfactory "designed-by-a-committee" feeling, which may be more the result of the final implementation decisions rather than of the prioritization in itself; still, the author strongly believes that the delivered software is no worse, probably much better, than if wrong or insufficient requirements were addressed.

References

1. M. Robol, P. Busetta, Applying BDI To Serious Games: The PRESTO Experience, University of Trento, Technical Report DISI-16- 011, November 2016, http://hdl.handle.net/11572/154697
2. Marco Robol, Paolo Giorgini, Paolo Busetta, Applying social norms to implicit negotiation among Non-Player Characters in serious games, Proceedings of WOA 2016, Sep. 2016, http://ceur- ws.org/Vol-1664/w5.pdf
3. Paolo Busetta, Mauro Fruet, Piero Consolati, Mauro Dragoni, and Chiara Ghidini, Developing an ontology for autonomous entities in a virtual reality: the PRESTO experience, Proceedings of MESAS 2015 workshop, Prague (CZ), April 2015, Springer LNCS 9055
4. SUPERSEDE project, https://www.supersede.eu (last checked: 12 Jan 2017)
5. Bordin, S., De Angeli, A.: Supporting cooperative work by integrating user-centred design and Agile development. Submitted at the European conference on Computer-Supported Cooperative Work (ECSCW 2016).