SIM 2017 / 14th International Symposium in Management

A new vision over Agile Project Management in the Internet of Things era

Andrei Gal[a], Ioan Filip[a], Florin Dragan[a*]

[a]Department of Automation and Applied Informatics, University Politehnica of Timisoara
Bvd. V. Parvan, No.2, Timisoara 300223, Romania

Abstract

Living in the modern society, where the *Internet of Things* (*IoT*) network is constantly evolving and *Things* belonging to the network are expanding new functions in a fast pace, there is inevitably a need of having a new vision over the software project management that should keep up the pace with these changes. The impact of the new smart devices is major and can be seen at all levels of software project management. By creating a project management solution across the *IoT* network, we aim to improve the efficiency and to optimize parameters such as: teamwork, analysis of a large amount of data, project managers and stakeholders input, resource planning, project development and feedback speed. A great contribution in achieving optimization and efficiency improvement for the software project management solution have all the smart devices using high computing power, high data transmission speed and efficient data processing. Therefore, this paper proposes an *Agile* project management solution that adapts to the *IoT* network, embedding Things with their features correspondingly into the project management process. Putting the project management process across the *IoT* network, the *Agile* project management solution slightly changes the steps of software project management as to increase different project parameters as like: team collaboration, team efficiency, project technique level, project technological level, project automation level.

1. Introduction

It can be easily noted that nowadays any company or individual that develops a product or provides services to an end user usually follows a pre-defined plan with exact steps to follow as for the product/service to reach out to the end user as planned from the beginning. There is a project plan created for almost every product/service we can buy from the market, and behind the final product/service there are groups of tens or even thousands of people that have worked all together in realizing the product/service as we see it.

As the process of making a product or providing a service is very complex, there is a need in creating a project management plan that begins from the idea of providing that product/service to the end user and can never end as it usually includes the maintenance of the product/service for as long as the user uses it.

In particular, the software industry follows the same rules when referring to the process of creating a software that is to be delivered to a certain end user, individual or company. So, we can define the term called software project management (SPM) as being the process that starts from the idea of creating a software continues with developing the software itself and maintains the software for the end user during the whole lifecycle of the software. In detail a SPM defines a plan with steps needed for the software to be delivered in terms of defining the software specifications, creating tasks for developing the software, plan the actual tasks to the allocated resources, maintaining the software by getting constantly feedback and improving the software as for what the end user expects to have as a final product.

A software project management plan can be defined as being as a process composed of six steps:

1. *Brainstorming for an idea*: This is the first step where the idea of the software is elaborated and where the investigations take place for finding out if the idea can be implemented and what are the pro arguments in implementing the idea. The most important role in deciding if the project can go on has the project manager at this phase. The main goal of this step is to determine if the software is feasible for the team to implement (Baars, 2006).

2. *Creating specifications*: The second step represents the phase where all the requirements must be completed. This involves the identification of all the expectations, analysis of all possible solutions for implementing the idea from the first step and defining the limitations if any found during the process of selecting the proper solution for implementing the idea (Baars, 2006).

3. *Defining the software architecture*: This third step requires a run through all the requirements noted at the second step and use them to create schemas, UML diagrams, architecture prototypes, flow charts that will represent the basis for the final software architecture that will be defined. On this step, there can be multiple schemas, diagrams or flow charts created so for the team to have different point of views when designing the final architecture. The final decision over the architecture is taken by the project manager after consulting the architects and the team leader responsible for designing the architecture.

4. *Developing an implementation strategy based on specifications and architecture*: At this point in the SPM there can be asked for the help of contractors or even developers in order to supply the project with every resource needed for being able to start the implementation. All 3ʳᵈ party tools needed for the implementation are bought/downloaded (for open source ones) installed and set to be ready for use in the implementation phase (Baars, 2006).

5. *Implementing the actual software*: This phase of the SPM is dedicated mostly to the work of the developers. Even so there is no pause in the SPM process for the other project categories involved. The designers must monitor that the implementation respects the design created in step 3, the contractors must sustain a live feedback between them and the developers to fulfil all the necessary changes if needed during this phase, and nonetheless the project manager who is responsible to coordinate the project and all the resources allocated to it to obtain the desired software for the end user. At this stage, the project manager can reorganise the teams, the resources, the plans, the diagrams accordingly to the implementation phase in case the implementation implies changes to these project entities. Also at this point in the SPM there shouldn't be made changes to the list of requirements only if the final customers require them and if it has this defined in the initial agreements with the company developing the software (Baars, 2006).

6. *Maintaining the final product*: This last step of the SPM might be looked over by an external party as a being a small and "can be excluded" step from the SPM. Though by looking to the newest researches over the software project management domain this is most cases the highest important step as it effectively translates the software from a proof of concept to a reliable, feasible software ready to compete on the software market. This is due to the fact that the main purpose for almost all the software products nowadays is to get out live as soon as possible, using in fact the feedback from the company as a real live testing system for their software. Of course, this is highly not recommended but it's an accepted approach as end users are getting more and more used to updating and accommodating to the newly features of a software. At this step, there are also accepted small changes to the requirements if