A Collaborative Tool for Managing Small Projects

Bruno Miguel Domingos
Instituto Superior Técnico
bruno.domingos@taquus.ist.utl.pt

Abstract. Managing and leading project oriented organizational structures is different from the common model of most organizations. However, nowadays, this paradigm has become more practiced worldwide, mainly in IT services enterprises. In this paper, the main problem deals with the tiny success rate that IT projects showed in the last few years, specially the small projects. This kind of project presents specific features of time and cost that differentiates them from the large projects. Consequently, dissimilar requirements of planning and communication impact, inevitably, the success of the methodologies applied and the software (SW) tools. This paper’s goal is to help with the resolution of some of these problems. The proposal is based on two key points: methodologies and SW tools. It presents itself as an alternative way of managing small information systems projects. This proposal was materialized through a model designed according to the needs of the small projects and a SW tool. This tool was implemented to fulfill the requirements gathered by the analysis of the faults of most used project management SW tools and it was tested in a competitive environment with satisfactory results


1. Introduction

Nowadays, in many companies in the world, the discipline of project management is still a headache. This paper presents a study about this discipline with the purpose of understand which are the key factors that contributes for the low rate of success related with project management, especially with the small IT projects.

Small projects have certain characteristics that really make the difference and due to the lack of importance that people assign to this kind of projects, typically, they end in a catastrophic way.

From many time ago that the IT Management has assumed a critical role on business support in companies worldwide. This practice ensures that services offered by these companies can grow in quality and execution.

Most of these companies are “project oriented”. This is a common model nowadays and the planning and control of these projects is high important so they can be successfully at the end. For IT Projects the model stills the same and it is good practice follow other additionally methodologies that improve the quality of management and development of these projects.

As well, to help managers on their roles of planning and control, it’s very common nowadays the use of software tools. Once again, the existing tools were conceived to large projects and don’t match some requirements imposed by small projects like:
2 Bruno Miguel Domingos

- Multiple project environment
- Shared resources
- Support for communication
- Collaborative work
- Ease of use

Small Projects are projects of lower dimensions, with special features and special needs. Based on various references, a small project is a project that varies between 1 person / 1 week until 6 people / 6 months. But this definition is not rigorous so it can differ from organization to organization or even from country to country, depending on the dimension of the organization or its projects size and value [1][2][3][4].

Typically, a small Project is also small in budget and resources but this is not true to risks or its impacts, neither for the value of a small project that can be such big.

This is exactly the niche of project management that I want to focus and show how different it can be from the large projects.

The actual methodologies like Capability Maturity Model (CMM), Rational Unified Process (RUP), eXtreme Programming (XP), PRINCE2, Virtual Members or Inspection Method seem to produce satisfactory solutions when applied to Software Large Projects. However, in more specific cases, like Information Systems (IS) Small Projects, the results show otherwise.

As well, Software (SW) Tools, which are considered high useful to support project management, share the same problem. The existent applications like MS Project or Basecamp are adequate for large projects where the managers are already trained and specialized but they don't fulfill the needs of small projects that require tools with low complexity.

In this paper is proposed a methodology aligned with the characteristics of the small projects and a SW tool implemented according to its needs. After studying the problems of communication related to this kind of projects, it was decided to follow a web-based approach because of the ease of accessibility when the resources are geographically spread. The key points of a project management SW tool are described in the 4th Chapter of this document.

Next, is described the context of this paper, the problems and characteristics of small projects, the proposal as model and the prototype implementation of the described tool.

2. Project Management

A project is “a complex effort to reach a well-defined goal, with time and money boundaries, which has multi-functional behavior and is unique and non-repetitive in an organization”, Cleland and King (1983).

According to the previous definition, it must be distinguish between projects and operations so they might be managed differently from projects. Operations are the base of the company’s pyramid and, consequently, are already well structured to ensure that an enterprise can work at all. Other than, projects are unstable and risk susceptible.

The differences between Projects and Operations [5] are listed below in Table 1.
A Collaborative Tool for Managing Small Projects

Table 1. Difference between Projects and Operations (Source: [5])

<table>
<thead>
<tr>
<th>Projects</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Singularity</td>
<td>Repetitiveness</td>
</tr>
<tr>
<td>Finite</td>
<td>Forever</td>
</tr>
<tr>
<td>Revolutionary</td>
<td>Evolutionary Change</td>
</tr>
<tr>
<td>Unbalanced</td>
<td>Balanced</td>
</tr>
<tr>
<td>Transitory</td>
<td>Stable Resources</td>
</tr>
</tbody>
</table>

As said before, in IT Projects, additionally to the normal process of planning and control, we can apply some other methodologies that can improve the efficiency of the project team (XP, RUP), its maturity (CMM) and agile methods of information, data and resources sharing (Virtual Members). Some of these methodologies are well tested and proved in Large Projects by real cases implementations and we can find many of these testimonials in a large number of papers.

However, for Small Projects this scenario is not the same, most because this kind of procedures were not designed and conceived for them. There are some documented tries to approach a few methodologies to small projects (CMM, Inspection Method) but, every single method has diversified problems.

2.1. Existing Methodologies

The most popular methodologies applied to IT Projects are Capability Maturity Model (CMM) [6] [7] [8] [9] [10], Virtual Members / Teams [11] [12] [13] [14], Inspection Method [15] [16] [17] [18], eXtreme Programming (XP) [10] [19] [20] [21] [22], Rational Unified Process (RUP) [9] [23] [24] [25] [26], PRINCE2 [27] [28], Scrum [19] [29] [30] [31], Feature Driven Development (FDD) [32], Dynamic System Development Method (DSDM) [33] [34] [35] and Adaptive Software Development (ASD) [36]. Most of these methods were thought for Software Projects; therefore they are not simple to apply to Information Systems Small Projects because of:

- Documentation overload
- Lack of time to reviews
- Numerous teams
- Training cost
- Complexity
- Specific orientation to a kind of project
- Long term to implement

Figure 1 shows how far these methodologies are from the expected values of an ideal case. For example, CMM, DSDM, RUP, XP, FDD or Scrum were designed exclusively for software development projects which impose boundaries on the application to another kind of projects, like information systems. Furthermore, like PRINCE2, Virtual Members and Inspection Method, which are more adaptable methodologies, also CMM, DSDM and RUP are not easy to implement unless the users have been trained. This implies high costs and long terms which are impeditive factors in small IS projects.
2.2. Existing Tools

Additionally to the methodologies, most of project managers tend to use computer applications to support their managing activities. But, for years, this kind of tools has been developed exclusively for Large Projects.

Table 2 shows a summarized comparison of the most common tools used in Project Management relatively to Small Projects characteristics [1] [37] [38] [39] [40].

<table>
<thead>
<tr>
<th></th>
<th>MS Project</th>
<th>Quickbase</th>
<th>Basecamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ease of use</td>
<td>***</td>
<td>●</td>
<td>*****</td>
</tr>
<tr>
<td>Need of training</td>
<td>●</td>
<td>●</td>
<td>*****</td>
</tr>
<tr>
<td>Planning and Control</td>
<td>*****</td>
<td>*****</td>
<td>●</td>
</tr>
<tr>
<td>Resource Management</td>
<td>*****</td>
<td>*****</td>
<td>●</td>
</tr>
<tr>
<td>Risk Management</td>
<td>●</td>
<td>***</td>
<td>●</td>
</tr>
<tr>
<td>Requirements Management</td>
<td>●</td>
<td>●</td>
<td>●</td>
</tr>
<tr>
<td>Documental Management</td>
<td>▼</td>
<td>*****</td>
<td>●</td>
</tr>
<tr>
<td>Communication Support</td>
<td>●</td>
<td>●</td>
<td>*****</td>
</tr>
</tbody>
</table>

(● – Nonexistent; ●● – Weak; ●●● – Medium; ●●●● – Good; ●●●●● – Very Good)
Based on the problems presented by Small Projects identified on previous chapters, were identified some requirements that the existing tools of project management support don’t match.

In Figure 2 is shown the classification of some tools that have been tested according to 4 main variables; X axis – ease of use, Y axis – communication support, circles size – planning and control functionalities; circle color – resources management.

**Figure 2. Comparison between project management software tools (circles size – planning and control functionalities; circle color – resources management)**

MS Project is the most complete tool present in this universe of study. It presents a wide board of mechanisms to plan and control a project and a sophisticated tool to manage resources. Even though, the resource management can’t be extended to multiple concurrent projects. Furthermore, MS Project doesn’t congeries communication mechanisms to guaranty collaborative work – a very important feature of small projects.

These factors combined with the high complexity of the tool make it inappropriate to support small projects.

Closely to MS Project appears Quickbase. This tool is web-based and this factor could be a contribution to a good communication mechanism. In spite of, Quickbase presents itself as a complex and a “hard to use” tool, inferior in functionalities and resource management when compared with MS Project.

Opposite to Quickbase and MS Project, Basecamp is a very complete tool in communication mechanisms to use to advantage the fact of being web-based. At the same time this tool is immensely simple to use and find the information needed. However, two key factors make the things difficult to actually use the tool: poor planning and control functionalities and absence of resources management.

### 3. Small Projects Management

Small projects are much different from large projects, but why? The answer to this question is presented in the next three subchapters (3.1, 3.2 and 3.3) and it's supported by Table 3:
### Table 3. Differences between Small Projects and Large Projects (Source: [1])

<table>
<thead>
<tr>
<th></th>
<th>Small Projects</th>
<th>Large Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of projects</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>No. of estimations</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>No. of scheduling</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>Mean duration</td>
<td>1 Month</td>
<td>1 Year and 6 Months</td>
</tr>
<tr>
<td>Full time management</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

This values show that Small Projects Management requires shared attentions, in short terms, with multiple-project environment.

### 3.1. Project Manager

The first choice to consider when a new project begins is the selection of a project manager [41]. But makes sense to choose a manager for small projects? This question has two different answers based on two approaches:

First one says yes [1] [3] [4], should be a project manager (PM) to a small project. In this case, the manager has more than one small project to manage and he/she has to divide his/her time by all the projects he/she is managing.

Second approach defends the existence of a "juggler" instead of a PM [3] [4]. A juggler is simultaneously PM and developer so he/she must know how to manage his/her team and do their tasks (obviously less charging tasks), being conscientious in every moment about the project and it evolution and dedicate himself/herself exclusively to that project. Besides a juggler must have excellent communication channels to all the team so he/she can understand what is going on.

Both this approaches have advantages/disadvantages depending on the projects in cause and the number of team elements. Is more correct adopt the "juggler" approach in projects with scarce resources, because otherwise 1/n of the team won't be developing, which is a big waste in a scenario like this.

### 3.2. Design Stage

This stage is also much different in small projects [1]. The answer to the next two questions can, clearly, show the points of divergence.

1. Should we do the general design ahead?

2. Which is the right measure of effort to invest in this task?

The answer to the first question is yes, the general design of a project should be done ahead with all the documentation needed.

The answer to question 2 is not to invest too much effort on this task in the initial phase of the project, because small projects are very sensitive to requirements change. The right thing to do
would be the constant “polish” of the design as the project evolutes and requirements becomes sealed, and near the end of the project document the maintenance manuals.

3.3. Specific Characteristics

Most of the special characteristics of small projects become from short term of duration. This factor has a high strength in planning, delivering and timing which are very unstable variables in small projects.

Because of this and other conditions, Small Projects presents the following characteristics [1] [3] [4]:

- Time oriented and time sensitive;
- Opportunistic;
- End goal well defined;
- Not all requirements are well defined in the beginning;
- Can be extensions of large projects;
- Lower costs;
- Equipment and Resources distributed for more projects;
- Planning and Control are more flexible;
- Design Stage very characteristic;
- Strength Communication Network;
- Collaborative environment.

3.4. Difficulties

Some authors defend that managing a project is the combine of three groups of problems: client satisfaction, communication and, management and technical [42] [28]. In Small Projects this groups of problems are more evident and probable to happen, and sometimes with disastrous consequences.

Client Satisfaction

This factor e directly related with the change and validation of requirements. As it was said, any change in the requirements implies huge changes in all the project implementation. However, the satisfaction of the client is our first goal so, if the requirements modifications bring non-acceptable schedules or budget, the best way to deal with this problem is to present this problems to the client and make him responsible. This is the main reason for the fact of in small projects we shouldn't make any rigorous specifications but instead turned for change.

Communication

This is the more probable and important problem that we can find in any project. Is fundamental that all the project team have strong communication channels since the beginning of the project so this can be successfully and executed in collaboration:
In a project of a short term, there is no time for create communication channels. They must already exist when the team is chosen. Besides, the manager/juggler of the project can't take care of each team member because he/she simply doesn't have time and team members, sooner or later, will need to change and share project information and experiences somehow. This key factor is the pillar of a small project and just this way it has a solid base of sustainability.

Over-management: In some small projects this is a common situation due to the reduced number of the team elements. Usually the manager/juggler anticipates himself/herself to solve possible problems that don't happened yet instead of delegate work and wait for the team members to call him/her. A good communication network is a facility in this case.

Under-management: contrary to the previous problem, this one occur when the juggler "forgets" the management tasks and dedicates himself/herself exclusively, or most of the time, to technical tasks, or when a manager (not a juggler) is managing many projects simultaneously and focuses too much in management tasks, forgetting the support and the accompaniment to the team. In these situations, the work assigned to the juggler, in each project, must be reviewed so he/she can spend 45% of his/her time just to management tasks.

The introduction of new team members causes communication problems due to the 1-n new channels that the new member must create with the rest of the team. The more advanced the project is, the more laborious this process become.

Technical and Management Problems

Technical problems may occur during the execution of a small project. This kind of problems are normally documented from previous projects, so team members can easily access information gathered to resolve their problems and improve their own skills. It's usual that each project has a communication plan that helps the element of the project team to know what information should be sent to who, and from where comes an answer.

Management problems are more delicate and difficult to deal with:

- **Short project lifecycle:** implies less time to gather data, identify problems and correct them;

- **Shared responsibilities:** it is hard to guarantee that all commitments will be ready on schedule and it's either hard to assign faults;

- **Actual Data:** this problem can become from an incorrect accomplish of the communication plan or from its absence. Information that arrives late is, in certain situations, worst than if don't arrive.

- **Control of multiple small projects:** One small project itself is easy to control but, if we join more ten projects things go hard in the same ratio. Besides, the problem becomes a little hard if all these projects are in different phases of the management process. This situation is, in the most of multiple small projects environment, a high barrier to the manager's performance due to the impossibility of use to advantage the work from one to another project. Consequently to all these problems appears another one, the management of shared resources. This is an unfair job because it doesn't depend exclusively on us, but from all managers from all projects in the company. In a general way multiple small projects problems can be classified into five categories:
A Collaborative Tool for Managing Small Projects

- **Complexity**: in resource management of multiple small projects;

- **Capacity**: of the organization to support multiple project environment and ensures sufficient resources to match all needs;

- **Conflicts**: of people, systems and organization;

- **Compromises**: related to the assignments of each team member and the responsible entities to ensure the resources;

- **Context**: configurations of the project as procedures, culture, norms or behaviors.

**Common Problems**

Also, other authors assume that small projects causes of failure can be originated by the combination of some common problems that affect also large projects. These problems are [43] [44] [45] [46] [47]:

- Bad scope/goals definition
- Absent of skilled people to make decisions
- Past experiences (projects) ignored
- Weak risk management, planning and control
- First estimations fixation
- Overmuch trust in software
- Team problems
- Bad requirements definition
- Misunderstood of the client needs
- Unrealistic quality levels
- Business goals changes
- End users resistant to change
- Sponsor uninterested
- Inappropriate technology

Just a few of these causes can be directly solved with a methodology or a SW tool, because most of them are poor management practices. However, these factors can be fixed with formation and education in the companies.

**4. Proposal**

Following the evaluation made in chapter 2.1, this proposal approaches the two main contributions for IT Project Management: Methodologies and Software Tools.

As well, it is presented as a model mapped in a simple process and a tool implemented through a list of requirements identified by the faults of the existent tools presented on Table 2.

**4.1. Methodology**

A methodology that matches the needs of small projects should:

- Be simple
- Be aligned with the short-term of small projects
- Contemplate multi-project environment
Be adaptable to different types of projects
Promote a collaborative environment
Assure the basic processes of planning and control

The materialization of this set of requirements is presented in Figure 3 and described below.

**Figure 3. Proposed Process (Collaborative tasks are filled at pink and collaborative points of decision at green)**

The process begins with the request of the client. After, the project manager makes a review to understand the project scope and goals and while the project is not clarified, the cycle between the client and the PM is repeated. When the project specification reach the equilibrium both the PM and the team should estimate time, cost and effort. This is a very important task to be done in collaboration due to the difficulty of it. Usually, small projects problems begin in this stage because the PM does this task by himself/herself. Once more, the entire team should decide if all the conditions are stable to continue with the project and in affirmative case the team revalidates the initial information supplied by the client.

On contrary, if the team decides not to start the project, a revision is made by the client and the PM together to try to fix the problems encountered. In the end, in collaboration, they finally decide to GO / NO GO with the project. If the decision is NO GO than the project is kept for a later evaluation.

After the team revalidate the initial information the execution and the management begins. At the end of the project, its closure should be done by all the stakeholders (Client, PM and Team) so everyone could handle experience to the next time.

### 4.2. Software Tool

Therefore, it is proposed a tool for help and support Small Projects managers or jugglers doing their jobs. This tool features all the missing requirements identified on the actual applications.

This set of requirements is described below:

- Easy to use so the user can feel that is commanding the tool;
- No need of training or specialization, due to the short term of small projects;
- Quick planning design should be done directly on the tool, avoiding the need of pre-design;
- The application should manage resources in a multiple project environment;
- The number of projects per person should also be managed through the application;
- This kind of tools must provide basic functionalities of planning, like task management, resource management and team management;
A Collaborative Tool for Managing Small Projects

- Each user of the application could, at every moment, access his/her tasks calendar so he/she can clearly view the situation of all multiple projects;
- Provide, constantly, context information about any project or any task;
- Issues and Risks Management;
- Deliverables Management;
- Requirements Management;
- Provide centralized communication channels;
- Allow the storage of documentation and other information that can be helpful to the project.

In summary, a small project support tool should be able to offer the useful information, in the correct format, to the properly people and at the right time.

5. Prototype

In order to evaluate the proposal was built a prototype of a Small Projects support tool in order to evaluate the characteristics gathered from this analysis and investigation about Small Projects. This tool, named “Small Projects Manager (SPM)”, was tested during a period of 3 months, by 30 users, managing 20 small projects. It was published in the extranet web portal of OutSystems, the owner organization of the technological platform where this tool was implemented.

SPM is a Web-based application that helps a distributed team, belonging to one or many organizations, easily manage a large number of small projects. As shown on Table 2, MS Project is great for a few large projects managed by only one person, but fails with many small projects. The Small Projects Manager was specifically designed, built and tested with many small projects and many people interacting using the Internet, possibly in different time zones.

![Project screen](image)

**Figure 4. Project screen**

This tool presents a mechanism to distinguish between 3 types of users: project managers, rest of team elements and the administrator that can be a portfolio manager. After the login a board comes up with all the information about the projects that each user is assign, including a project Gantt chart (Figure 5).
Additionally, this board shows alerts about the ending and overdue tasks that the user should complete, indications of messages received and the possibility to propose new projects to the system administrator.

From the board screen, the user can access the calendar screen where are presented a set of views of all the tasks that the user is assigned. These views are manipulated by a group of options which filter the data by status, date, order and responsibility. It is also possible to view the tasks referent from a project that the user is managing so he/her can understand the project status and dependencies. In Figure 6 is represented the described screen.

As said before, from the initial board is possible to access to any project assigned to a user. In the project screen are introduced all the managing functions related to that project. Although all the functionalities are presented, just the manager can take all of it. For example, a common team member can’t create a task without the manager evaluation.

Through this screen, shown on Figure 4, are accessible the follow mechanisms:

- Task management – create, delete and edit tasks, assign resources, associate deliverables and requirements, create dependencies (predecessors) and change task status.
- Risks / Issues management – identify new risk/issue, set priority, status and impact, discuss resolution and contingency plan through the forums and set responsibilities.
- Files storage – upload and download files, attach files to deliverables and associate files to forums.
- Requirements management – upload requirements from files, include new requirements, associate to tasks, change requirements status and define priority.
- Deliverables control – Identify new deliverables, assign to task, manage status, notify (by message or e-mail) task responsible of ending and overdue deliverables and attach files.
A Collaborative Tool for Managing Small Projects

- Forums and notes – open new forum manually or automatically (on issue entry), create new note, comment existent threads and attach files.
- Import / Export tasks – import/export tasks from excel files (MS Project format).
- Gantt chart – graphical representation of tasks and projects.
- Visual Plan Grid - graphical view of tasks (duration, assignments and status) through a dimensional week window.
- Team management – Add/remove elements to/from the team, create groups of members to easily assign resources to tasks and send text messages to other teammates.

The main contribute of this prototype was the help provided in the activities involving transversal decisions of the team and the clients, especially in the tasks of time and effort estimative. Consequently, it was proved that communication difficulties can be minimized with easy collaborative tools.

6. Conclusion

In this paper is proposed a tool to support Small Projects. This tool differs from the other commercial tools because it matches exactly the failures of those other applications and guaranties:

- Help to manage a large number of small projects
- Uses the Internet to support a large number of users
- Shows tasks in a well-known Gantt chart
- Presents an easy to read Events Calendar per user
- Supports tasks, meetings and deadlines
- Centralizes user projects, user privileges, and user roles
- Provides email notifications
- Maintains information about old tasks and projects
- Keeps track of risks, deliverables, documents, resources and assignments
- Supports different configurations.

The main value added by this application is about the collaborative environment provided. In small projects is very important that everyone in the team could share information and participate actively in the key tasks and decision points mentioned in Figure 3. Besides, the proposed tool can also manage resources in a multiple-project environment which is a precious functionality to the managers. However, this is clearly a future work topic due to the lack of time to improve it.

At the same time, the basic functionalities of planning and control were implemented following the approach of “team work”, providing actual data to the team and the possibility of everyone participate on this task by suggesting tasks changes to the existent ones.

Also the risk and issue management was reflected by a mechanism of tracking. If a team member suggests a risk or an issue, it can be discussed by all the team recurring to forums and tracked by historical data, like impacts or status, kept by the system.

In the end, after the testing period, the application reached the pretended level of ease of use. This just was possible because of the growing interest of the testers on improve the tool and the constant feedback options supplied.

As future work stills some considerations that might bring new developments to the background work:
14 Bruno Miguel Domingos

- Test the tool with more people from various companies and different business areas to gather more knowledge and confer more credibility and applicability.
- Follow the methodology described in Figure 3 to prove the real scientific value as a benefit to small projects management.
- Improve the basic functionalities to ensure a more intuitive way of work.
- Review the graphical interface, for example, adding graphical representation of the tasks dependences in Gantt chart.

7. References

A Collaborative Tool for Managing Small Projects


16 Bruno Miguel Domingos


44. Using large vs. small group projects in capstone and software engineering courses. Stein, Michael V. Minneapolis : ACM, 2002.

