STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LVIII**, Number 3, 2013

COMPARATIVE STUDY OF TASK DELEGATION MODELS IN SOFTWARE AS A SERVICE PROJECT MANAGEMENT APPLICATIONS

BOGDAN POP[(1)] AND FLORIAN BOIAN[(1)]

Abstract. Task delegation and resource allocation are two of the most important aspects of project management. Bad judgement and errors with regard to task delegation can result in loss of time, resources and a lack of successful project outcomes. Most of the currently available project management applications, no matter what their platform or distribution model is, offer a wide range of tools to ease task delegation. However, none of them have successful, automated task delegation mechanisms, although an automated process would help by reducing losses caused by human error or poor decisions, thus improving overall project results. This paper presents a comparative study between commercial, publicly available project management applications and a proposed application that automates task delegation and showcases the benefits found by using an automated task delegation process.

1. Introduction

Project management is a complex activity that requires proper application of skills, knowledge, different tools and techniques in order to reach or surpass project requirements. Project management consists of five main process groups: 1) the initiation of the project, 2) it's planning, 3) the execution of the project, 4) the proactive management of the project and 5) it's closing. Successful projects can be defined in many ways, mainly because project success can be measured based on a number of factors. Such a complex activity, usually managed by humans, is easily subjected to errors and losses. A study

Received by the editors: April 5, 2013.

2010 *Mathematics Subject Classification.* 90B50, 90B90.

1998 *CR Categories and Descriptors.* K.6.1 [**Management of Computing and Information Systems**]: Project and People Management – *Life Cycle*.

Key words and phrases. project management, tasks, task delegation.

This paper has been presented at the International Conference KEPT2013: Knowledge Engineering Principles and Techniques, organized by Babeş-Bolyai University, Cluj-Napoca, July 5-7 2013.

in 2005 [7] shows that 27% of a manager's time and more than USD 100 billion is spent each year to counteract the effects caused by improper project management, especially problems created by workers that are not suited for their tasks.

The paper is structured as follows: section 2 briefly describes project management and the different methods and factors one can use to measure a project's success, section 3 presents the proposed real-life project that is used within the proposed applications. The 4^{th} section shows the results of the study performed and whether the proposed model is successful or not, while the 5^{th} section presents possible future studies or tests that may be performed and future developments.

2. Background: Project Management and Project's Success

Before presenting the real-life study and its results, a simple classification of project types, how their success or failure is measured and how one can tackle the management of the project via different techniques is required. This allows for a better understanding of the actual study and the performance of each of the tested applications. Most project management applications are tailored towards a specific domain or field and few of them are geared towards project management in a general sense.

A project, disregarding its scope, can be viewed as a series of tasks that have start and end dates, sometimes even times, budget limits, and a specific objective. Usually, this objective must be met while keeping the work within certain specifications and constraints. The tasks that compose a project require resources, both human and non-human [3].

Projects can be in-house and be developed within the company. Others can be contracted projects where a business to business relationship is formed between the project owner and the developer. Projects can be subcontracted where the whole project or a part of it is sold for development outside the company. In this case, the seller may be a contractor as well, not necessarily the project owner. Larger projects are consortium-based, as multiple companies or organisations are forming a joint venture to have the tasks completed with well or ill defined responsibilities.

Project management is the complex set of activities performed by an individual or a group of individuals that requires proper application of skills, knowledge, tools and techniques in order to reach or surpass project requirements.

Project management usually includes the following [1]:

- Identifying the requirements and objectives of the project

- Proactively addressing the concerns and expectations of the project owners / stakeholders since the project is started and even after it is completed
- Balancing of the project constraints:
  - Scope
  - Budget
  - Schedule
  - Resources
  - Quality
  - Risks

Project management is a proactive activity since many factors can change at any given point in time and countermeasures must be performed to preserve the scope and the end goal of the project.

Sometimes it is even the case that a simple change in one of the important factors or constraints determines a chain reaction modifying a lot of other variables in the project.

Project owners or stakeholders can each have a different grasp on the most important factors creating added pressure on the project manager(s) and the workers. Changing the terms and environment of the project can also add additional risks and the development team must be able to asses the situation quickly and make the proper adjustments in order to deliver the project successfully.

Proactive management involves continuously improving and detailing a plan of action as more detailed and specific information and accurate estimations become available during the project's lifecycle. This allows a project management team to manage to a greater level of detail as the project evolves [1].

There are many ways to measure a project's success. Such an assessment is usually made based on the most important factors of the project and its desired outcome. One of the generally accepted set of measurable objectives that are taken into consideration are showcased in Table 1. Some potential benefits of proper project management are shown in Table 2 [3].

2.1. **Project Management Applications.** This subsection briefly describes the most popular project management applications available on the market as of March, 2013. The popularity is measured by the number of users each application has and the user-base growth over the past 12 months. Data was collected from the official website of each service provider, or by manual inquiries, if not otherwise specified. Based on data collected and shown in Table 3, one can infer that proprietary web based project management applications are more popular than the self hosted ones.

TABLE 1. Measurable objectives taken into consideration while assesing a project's success

Objective	Importance
Customer acceptance	High
Time constraints	High
Budget constraints	High
Effective and efficient usage of resources	Medium
Desired outcome, quality and performance	Medium

TABLE 2. Potential benefits of project management

Benefit	Importance
Identification of functional responsibilities to ensure that all activities are accounted for, regardless of personnel turnover	High
Minimizing the need for continuous reporting	High
Identification of time limits for scheduling	High
Identification of a methodology for trade-off analysis	Medium
Measurement of accomplishment against plans	Medium
Early identification of problems so that corrective action may follow	High
Improved estimating capability for future planning	Medium
Knowing when objectives cannot be met or will be exceeded	High

Most project management applications are geared to a specific domain a group of domains. Others are designed for a wider range and are very general in what they can perform. One domain-oriented application is Trac [18], which is a simple application that makes issue tracking easy for software projects and uses a minimalistic approach in its design, focusing on actual development rather than imposing techniques and policies on the people using it.

Launchpad [13] is also tailored towards software development. Its main features include code hosting and reviewing, bug, issue and specification tracking, and more. It was launched as a proprietary application in 2005, but since 2009 the license has been modified and it is now open source under AGPL.

Redmine [16] isn't designed specifically for software development and can be used to manage projects in a more general sense, being bundled with a lot of features and tools. It is cross-platform and cross-database, supports multiple projects, has flexible role based access control and issue tracking systems,

TABLE 3. User base and user growth for SaaS project management applications

Application	License	User Base	User Base Growth
Trac	BSD	16.200	Low
Launchpad	AGPL	30.759	Low
Redmine	GPL	N/A	Low
MantisBT	GPL	1.500.000	Very low (-40%)
Jira	Proprietary	14.500	Medium
Basecamp	Proprietary	200.000	High
Mavenlink	Proprietary	125.000	High
Assembla	Proprietary	500.000	Very High
TeamworkPM	Proprietary	N/A	Medium

has Gantt charts and calendars, per project wikis and forums, advanced time tracking tools and more.

MantisBT [14] is a self-hosted issue tracking system that can support multiple projects per instance, sub-projects and categories. Users in MantisBT can have a different access level per project, with no limit on the number of users, issues, or projects. MantisBT was launched over a decade ago and is quite popular, with more than 1.500.000 downloads. However, statistics show a decrease of those numbers of almost 35% from month to month, with unclear information on whether what percentage of the downloads are actually used in production. Figure 1 shows download statistics within the past 12 months (March 2012 - February 2013).

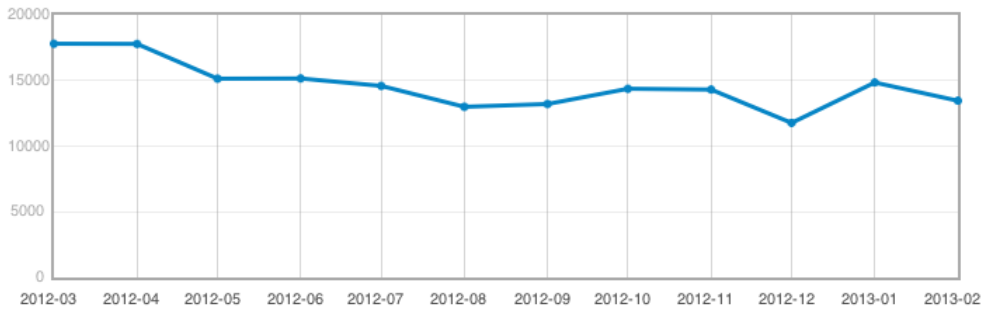

FIGURE 1. Monthly downloads of MantisBT in the past year [8]

Jira [11] is a project management tool used to track teams, planning, building and launching great products. It is one of the most complex applications with more than 150 tools such as capturing any types of issues, from bugs and features to stories and requirements to simple tasks and action items.

Jira can be extended by using one or multiple add-ons out of more than 400 that are available. These add-ons ca be used for agile project planning or to simplify planning and reporting when developers use Scrum [6] or Kanban [4]. Jira also has time tracking add-ons as well as Gantt chart plugins, which are all very important throughout the development and management of any project.

Basecamp is [12] one of the most popular project management applications on the market and is developed by 37signals which have also pioneered the RubyOnRails framework. With Basecamp, projects can be stored safely in the same system. Basecamp has advanced reporting tools that allows project managers to easily grasp the status of their projects, no matter how many there are. Basecamp makes easy for all team members, clients, contractors, and vendors to interact using the same platform. Project managers have full control over advanced users permissions, from project access to user interactions. Basecamp feature-set is very granular and allows users be organized in groups.

One can also see everyone's schedule on a visual advanced calendar, provided permissions are granted. Tasks can easily be assigned and delegated to team members. Additionally, Basecamp allows the creation of to-do lists and items, and all the features are integrated within a central email based notification system. Basecamp can be integrated with third party applications that allow easy time tracking, budget monitoring and more.

Mavenlink [15] is a business management application that extends basic project management techniques and activities to a wider whole business coordination. Mavenlink can easily be used for collaboration purposes, tracking tasks and time, budget management and even accounting with invoice generation.

The project management applications chosen for the task delegation model study are the following: Basecamp, Teamwork PM and Automated.PM [10], the later being the application developed based on the model described in [5].

The choice was made based on popularity and feature set of the applications. The proposed study should reveal and point the advantages of the described model over the ones in use by current applications. The complete scenario is described in the following section while the result of the study and whether the described model is successful or not are presented in section 4 and 6.

3. Approach: Proposed Real-Life Study Scenario

In order to study the performance of the selected project management applications, a real-life development project was chosen. Specifically, the project that is used as a test subject is building a treehouse. This choice was made because the size of the project isn't large to pose any serious issues and it is easier to measure its outcome. One can also measure the performance of the applications and their task delegation models that are being studied and tested.

A secondary reason why a treehouse project was selected for the study is the fact that every treehouse project is different than the next one. Most treehouses' bases are different from each other because usually a treehouse base is composed out of at least 2 or 3 large tree branches which always have irregular shape, size and growth angles. Apart from the base structure, most treehouses have different surrounding environments which also affect their own structure, planning, required materials and completion times [2].

The branches used for the support of any treehouse should always be in close proximity to each other. It is also recommended that the wood used to build the treehouses is pressure-treated and coated in order for it to support the structure and the floor. The roofs can be made out of various materials, from wood to thick film, depending on overall design, weather conditions and climate etc. One could also use recycled wood instead of fresh treated timber. No matter what type of wood is used, the structure of the treehouse shouldn't be rigid and fixed. Movement and tree growth should be allowed for long term durability. If the treehouse is fixed in its structure and binding points with the tree, it is likely that during time the structure will suffer greatly due to integrity changes from the base up [2].

Although only one solid tree with multiple thick branches can be used, it is recommended that more are used, resulting in bigger treehouses and a healthier environment for each of them as the actual treehouse would be supported by multiple plants instead of one. The average height of the base structure of a treehouse is usually at around 3 meters high, but this varies due to field conditions and age-group destination. Another important aspect in treehouse building is choosing the species of trees used for the support structure. Strong, slowly growing trees should be chosen against others that may grow faster or may not be sturdy [9].

If treehouses are built high off the ground movement caused by weather such as wind and gusts have to be taken into consideration and the structure be built even more elastic to allow for movement. If the base trees are strong, sturdy and the treehouse isn't built high off the ground, a fixed platform can also be used [2].

The height of the treehouse should be decided based on manpower available for the construction. If the project is built by only one individual, it's recommended the house isn't very high to reduce the complexity of the tasks, since most of the wood bolting and nailing is done overhead. The higher the distance from the ground, the harder and riskier the construction tasks are, especially when working with structure joists which may be even 50 kg in weight. A common-sense rule in treehouse constructions is making them lightweight.

Treehouse building can be simplified by using additional manpower or by using mechanical tools, such as chain hosts, most often used by car mechanics, with or without ratcheted brakes, power drills and screwdrivers, preferably electric, cordless ones, sabersaws, circular saws, hand saws, hammer drills, different sized bits, nails, screws, preferably galvanized, tape measures, metal yardsticks, carpenter's squares and levels and a tall ladder [9].

Other parts or tools that may be required, depending on design, layout and terrain details are: plastic sheathed galvanized steel cables that usually support up to 200 kg, heavy duty cable tension adjusters, flat washers and lock washers, bevels to protect against sharp edges, rivets and more.

3.1. **The treehouse project.** This subsection describes the general steps and development process of a treehouse while tables 4 and 5 present the tasks required to complete the project, their description, estimated duration and minimum recommended manpower. The initial phase in building a treehouse is selecting the trees it will rest upon. After the location is chosen, the area must be cleared, by tree surgeons or at their recommendations. The following task is to buy the necessary materials and equipment, if none is available. This includes lumber, bolts, cables and required machinery.

Once the necessary materials and tools are on the construction site, the main beams are raised, leveled and bolted in the trees. The base platform of the treehouse is composed out of 2 individual pieces, one being built on the ground and then raised into the tree on the structure, while the other should be built directly on the main beams. The next step required to complete the entire base structure of the treehouse is to add the plywood to the structure and hold it in place with galvanized screws.

The construction continues by cutting an opening into the base structure to form an access door and to build the ladder needed to climb into the treehouse. Once the base platform is completed and easy access is obtained, the external walls are built by using joists for the structure and plywood to coat the wall. Cuts are made into the wall's plywood in order to obtain windows. Joists are then placed in position to form the roof structure at an angle with splinters applied on top to protect the structure against rain, snow or debris.

TABLE 4. Treehouse project design & base structure tasks

Task name	Task description	Estimated duration (hours)	Recommended manpower
Treehouse design	Selecting the location of the treehouse, structure design and layout	20	1
Clearing the construction site	Clearing the area around the trees, by or at the recommendations of a tree surgeon	24	2
Lumber acquisitions	Buying lumber, ranging from large joists to splinters and plywood	4	2
Hardware acquisitions	Buying bolts, cables etc	2	1
Tool acquisitions	Buying all tools and machinery needed for the construction	8	1
Leveling base beams	Level base structure's main beams between trees using carpenters level	4	2
Assemble central base platform	Cut, trim, sand and bolt together the central area of the platform	4	1
Build the perimeter base platform	Raise joists in the tree and bolt them to the base beams	4	2
Raise central base platform	Raise central base platform and bolt it to the main beams	3	3
Add plywood to the entire base structure	Measure, cut and raise plywood and screw it tightly to the base structure using galvanized screws	4	1
Create access door	Cut a whole in the base structure and the applied plywood, add hinges and lock	2	1
Build ladder	Measure, cut and bolt together joists and poles to form a ladder; put ladder in position below access door	3.5	1

TABLE 5. Treehouse project walls and roof tasks

Task name	Task description	Estimated duration (hours)	Recommended manpower
Build exterior walls' structure	Measure, cut, position, screw exterior joists for treehouse walls	10	2
Apply plywood to exterior walls	Measure, cut and bolt plywood to exterior walls	6	2
Cut windows	Position, measure and cut windows within the walls	3	1
Build roof structure	Measure, cut and screw into position joists for the roof	6	2
Apply splinters to roof	Measure, cut and bolt splinters on top of the roof structure	10	2

3.2. **Testing and studying the proposed project management applications.** Based on the data presented within tables 4 and 5 content was generated and introduced into the project management applications in order to thoroughly manage the tasks presented. There were 3 workers assigned to develop the project, a carpenter, a taxi driver and an unqualified individual. The crucial design and structure tasks were assigned to the carpenter while the rest of the tasks to any worker that was available. Section 4 presents results gathered from the project management applications after 3 consecutive treehouse development projects. This was needed in order to train the proposed model (Automated.PM [10]) and have it assign tasks automatically.

4. Study Results and Successfulness of the Studied Model

Although Basecamp has many features as described in section 2, it has many drawbacks. For instance, using Basecamp with Projectite, its Gantt diagram add-on, had its issues, mainly because a task cannot be assigned to more than one person. Another drawback was that one cannot set the start and end times as well, only dates are available. This is a major issue since some tasks may be solved within minutes or hours instead of entire days. Further more, the Gantt charting add-on didn't allow to set multiple predecessors for a single task. This was needed for instance in the case of leveling the base beams which required both lumber, tools and hardware, and the presence of the carpenter and an additional assistant.

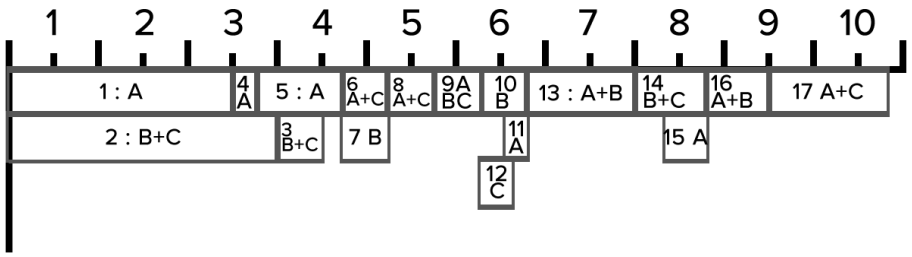

FIGURE 2. Chart showing ideal task delegation within the project. Numbers 1 to 17 are task numbers as denoted in tables 4 and 5. Capital letter A, B and C represent carpenter, taxi and unqualified workers

This resulted in a project estimated time to complete of 8 days for the design and base structure alone. The ideal task delegation scenario only needs roughly 5 days and a half for the first part of the construction, while the whole project estimated time to complete is just under 10 days, as shown in figure 2. The overall estimated time according to Basecamp and Projectite is 16 days, almost double than the ideal scenario.

Teamwork PM [17] has the same issue as Basecamp, not allowing time input, only dates. However, Teamwork PM allows to assign multiple people to the same task. A drawback for Teamwork PM is that it allows project managers to assign multiple tasks to the same individuals, although it is clear that they are already working on another task within the same time slot. Teamwork PM also has time loggers which can be used by workers to effectively track worked hours on each task they are assigned to. It also shows improved features but extra time is required on project manager's part when assigning tasks to workers with their start and end dates. Teamwork PM project plan estimated the time to complete at around 14 days, still 4 days behind the ideal schedule.

While testing and studying Automated.PM, better results and potential for improvement was detected. As the proposed model uses past task delegation to automatically assign tasks, the first phase of the project was initially introduced and delegated manually. Automated.PM allows time input instead of date constraints the other applications had, resulting in more accurate estimations. When creating the tasks for the second part of the construction, the software automatically assigned them as they were added to available workers, based on their past success or failures. Since the carpenter is more experienced with structural engineering and design of buildings, the software automatically assigned this individual to the two most important tasks of the second phase,

namely building the walls' and roof's structure. On the initial run of the project, Automated.PM yielded an estimation of just under 12 days for the completion of the project, while the third and last run yielded an estimation of 11 days and 5 hours.

5. Future Studies, Tests and Work

As the study has shown, with each similar project managed using the proposed model, the estimated time to complete slightly improved. However, the study has been performed on a small scale project. Future studies and tests should be designed with larger, more complex projects in mind, within larger companies and with multiple teams working on the project.

Future developments may include enhancements of the current model, such as assigning tasks based on users skill set not just past performance, time tracking, taking user preference into consideration when automatically assigning the tasks, the ability to assign tasks to multiple individuals or to select task predecessors for newly created ones in order to create proper dependencies.

The proposed model performed better than the ones currently on market with overall completion times being 26.77% faster than Basecamp and 15.45% faster than TeamworkPM. However, the estimation of the proposed model is 19.23% longer than the ideal project plan leaving room for improvements.

References

[1] *A guide to the project management body of knowledge*, 4^{th} Ed, P.M.I., 2008
[2] A. Aikman, *treehouses*, CreateSpace Independent Publishing Platform, 2010
[3] H. Kerzner, *Project Management a Systems Approach to Planning, Scheduling, and Controlling*, Tenth Edition, John Wiley & Sons, 2009
[4] P. Klipp, *Getting Started with Kanban*, Amazon Digital Services, 2011
[5] B. Pop, *Building an Automated Task Delegation Algorithm for Project Management and Deploying It As Saas*, Studia Univ. Babes-Bolyai, Informatica, Vol LVIII, No 1, 2013
[6] K. H. Pries, *Scrum Project Management*, CRC Press, 2012
[7] J. Skabelun, *Are non-performers killing your bottom line?*, Credit Union Executive, Vol 31, No 13, 2005
[8] Sourceforge, *Statistics for MantisBT 01.03.12 - 28.02.13*, http://goo.gl/TXUkX
[9] L. Tai et all, *Designing Outdoor Environments for Children: Landscaping School Yards, Gardens and Playgrounds*, McGraw-Hill Professional, 2006
[10] http://automated.pm
[11] http://www.atlassian.com/software/jira/overview
[12] http://www.basecamp.com
[13] https://www.launchpad.net
[14] http://www.mantisbt.org
[15] https://www.mavenlink.com/
[16] http://www.redmine.org
[17] http://www.teamworkpm.net/
[18] http://trac.edgewall.org

(1) Babeş-Bolyai University, Department of Computer Science, Cluj-Napoca, Romania

E-mail address: popb@cs.ubbcluj.ro, bogdan.pop@webraptor.eu

E-mail address: florin@cs.ubbcluj.ro