Politecnico di Torino

Dipartimento di Ingegneria Gestionale

Laurea Magistrale in Engineering and Management

Sprint to innovation: evaluation of benefits to use agile project management in software development

Supervisor: Alberto De Marco

Candidate: Sara Spezzigu

Academic year [2017-2018]

Ai miei nonni

"Volli,

e volli sempre,

e fortissimamente volli."

(V. Alfieri)

ABSTRACT

As the IT consultancy market is highly competitive, SMEs (Small Medium Enterprises) in this sector are strongly driven to innovation by their own nature. Most of them are constantly looking for new business models to gain a competitive advantage over their rivals. And what is the most revolutionary and innovative approach that an IT consulting firm can have today, in 2018, if not get closer to the world of the Internet of Things?
However, if the idea of developing the software of the Internet of Things could certainly create added value, on the other hand, it is such a new and complex subject that many times it could turn the company back rather than push it forward, with regard to the implementation of more complex applications at the software level but also to a wrong approach in project management.
Currently, the Agile Methodology has received wide recognition among project management specialists for its flexible and effective operating methods that can help to adapt to the evolving business environment. Through the analysis of a real case of application of this methodology (with a focus on the Scrum framework), we tried to evaluate its strengths and, above all, to understand the reasons for that effectiveness - and efficiency - that made it a reference point in the world of professional IT.
The dissertation is divided into three parts. The first part aims to provide - from a "theoretical" point of view - an overview of the main differences between waterfall and agile methodology in the software development field. The second deals, instead, with the real case study mentioned above, through a practical application of the methodology on an IoT project, supported by the use of a management tool properly chosen through a software selection with multi-criteria approach. Finally, the benefits obtained are evaluated, which can be summarized into lower risk projects and, ultimately, into an improvement in productivity and quality. The evaluation of benefits is not intended to be a synthetic measure that leads to a numerical result, but a multi-parameter evaluation that serves to demonstrate how much is efficient the work that has been carried out in order to create value for the company.

INDICE

1. INTRODUCTION

The following thesis is the result of an internship in PLM Systems, an IT consulting company based in Turin (To).

A detailed literature research was carried out to help clarify the main problems encountered in small/medium-sized IT consulting firms about project management and related adoptable solutions, which can be summarized with the use of agile methodologies and the digitization of project management.

The purpose of the thesis, therefore, is to examine the impact of Agile Methods, which should have capabilities that go beyond traditional methods as the ability to successfully deliver rapid and cost-effective results on complex projects with poorly defined requirements. The aim is to allow project managers to assess the feasibility using an agile method well supported by a management software tool carefully chosen. In fact, the dramatic increase in project management software development, combined with the lower cost of these solutions, has aroused considerable interest in these digital PM solutions, reducing the gap between project management theory and practice. However, the process of selecting a viable digital project management solution can be a very challenging task for any organization.

It was found that there are dozens, if not hundreds, of various digital PM solutions available on the market and that these solutions differ significantly in terms of adaptability, functionality, usability, visualization, security and cost affordability. Therefore, within the organization, it is important to try to understand which of these solutions available on the markets need to be evaluated.

The next step was the specific analysis of the company's issues, which demonstrated that the problems listed in the literature matched with the specific problems of the company under examination.

The empirical case study consists of an analysis of the current state and specification of the project management requirements in PLM Systems, the practical application of the Scrum framework on a real project considered the most suitable for this analysis, and finally a complete feasibility study of three cutting-edge digital project management solutions available on the markets: Jira, VisualStudio.com and FreedCamp.

These PM digital solutions can support project planning, execution, monitoring and control, but also make projects more predictable and manageable in terms of time, cost and quality, leveraging the key principles of the agile manifesto.

This three software were first selected from a shortlist of six candidates through a pre-selection. Then, since when the solutions are restricted to a few selected "finalists" we should start to demonstrate and test the usability of these solutions to determine more specifically which is the most suitable, they were used in practice in order to collect feedback from end users (project team) to have a more realistic view of the actual functionalities of each of them and be able to make a consistent multicriteria evaluation through a mathematical method, the Laplace's criterion for insufficient reasons.

From this study, several theoretical and management conclusions were drafted regarding the benefits of the use of agile methodologies as well as the digitization of project management.

2. LITERATURE REVIEW

This chapter lists the main principles of project management in general and then, more specifically, in the software development sector, with particular attention to the differences between a traditional waterfall methodology and the most innovative agile methodologies (Scrum in particular) in small and medium-sized IT consulting companies. The main problems they encounter and how the digitization of project management can bring considerable benefits are also discussed.

2.1. What is Project Management?

In order to efficiently define what "project management" means, there is no doubt that you need to be able to define what a project is in advance.

There are different definitions of project. One of the most appropriate, also because it creates a concise distinction with respect to other types of activities, is that projects have basically a well-defined beginning and end (Artto, Martinsuo, Kujala, 2011). A project, therefore, can be defined as a unique process, characterized by the performance of different coordinated and interconnected activities, aimed at achieving different goals.

For projects to be carried out according to the requirements that have been set, over the years, it has always been felt more necessary to have a professional figure who would ensure that the deadlines, costs and conditions of the project are respected, the Project Manager.

Risk and uncertainty are two components that are always present in the Project Manager's daily work, that’s why each project has its own unique and non-repetitive nature (Acebes et al., 2014).

the principal scope of the Project Manager will be exactly to try to minimize these risks and uncertainties that will inevitably arise.

Furthermore, what is even more complicated is that to become a good project manager, you do not just need to study certain techniques and methodologies, but you need to have mostly innate soft-skills.

The pm must have very strong communication skills, must inspire trust and be well-liked, but at the same time must be strong, with integrity, and above all credible. The main task of the pm is to efficiently coordinate the work of the various teams in the company, ensuring that the requirements of the customer are met.

A project team is a group of human resources who work together to carry out tasks assigned to them by the PM, within the set time frame. The project team is not only responsible for the proper performance of the required tasks, but also for the timely provision of reports, progress and issues related to the project itself. The PM will have to acquire the project team, understanding which the resource's availability is, what level of expertise they have for those tasks, how much they want to be part of that specific project and how much their labour costs are.

Then proceed with the development of the project team, trying to motivate the team as much as possible with incentives and rewards for the resources that achieve the most positive results, ensuring that the rule of the 3C (Cooperation, Coordination, Communication) is followed: a certain level of collaboration between the various components of the team must always be maintained, through proper coordination, all possible only thanks to excellent communication tools.

Finally, it will have to proceed with the actual management of the project team through scheduling, planning and allocation of resources, taking in consideration that in a company in continuous growth, each resource should have specific tasks related to their skills and competences.

So, what is really fundamental is that each resource is assigned a specific role according to its responsibilities (3R rule: Resources, Roles and Responsibilities) (De Marco, 2007).

In a nutshell, what has just been said can be effectively synthesized with the Project Management ferrous triangle (Figure 2.1). The first objective is the Scope, i.e. what to do, with what features and functions; the second is Schedule, i.e. within what time to carry out the various activities and in how many phases divide the project, and the third is related to Resources, focusing on the cost for each of these and the budget that can be used. (PMI, 2013)

Figure 2.1 - The Ferrous Triangle (source: Scott W. Ambler 2006-2012)

2.2. Project Management in the software industry

Everything that has been said in the previous paragraph, was related to the art of project management for any type of project, which can range from the project for the construction of a bridge, to that for the implementation of a service. Clearly what are the basic principles and the cornerstones remain the same for each type of company for which the PM works, but there are various facets that change depending on the type of project that needs to be implemented.

The software industry is characterised by a high level of turbulence, as IT companies must always adapt their services or must always develop information systems for complex organisations, where requirements are never easy to define.

In this regard it is interesting to analyse the concept of software life cycle that we can define as a description of the production industrial software, from its initial conception to its complete development, to his release and subsequent evolution (Pressman, 2008).

When making a product, it is important to take a number of predictable steps, a kind of guided path that helps to achieve high quality results in the time set (On Target, On Time, On Budget).

The software life cycle term indicates the way in which a methodology or a model decomposes the development of software products in sub-activities coordinated with each other and generally includes at least the following activity:

-**Definition of the objectives**, i.e. the aims of the project.

-**Analysis of needs and feasibility**, i.e. the formulation, gathering and formalisation of the needs of the applicant (the client) and of the set of constraints.

-**General conception**, which concerns the elaboration of the specifications of the architecture
of the software.

-**Detailed conception**, which intends to precisely define each subset of the software.

-**Unit coding and testing**, i.e. translation into a programming language of the functionalities defined in the design phase and the check that each subset of the software is implemented in accordance with the specifications.

-**Integration and Testing**, with the aim of ensuring the interface of the different elements (modules) of the software and verification of compliance of the software to the initial specifications.

-**Release**, i.e. delivery of the product to the customer.

-**Maintenance**, which includes all corrective actions (corrective maintenance) and evolutionary (evolutionary maintenance) to the software.

The presence and sequence of each of these activities in the software life cycle depends on the choice of a project management approach or methodology between the client and the development team.

The categories of methodologies currently present are as follows:

- Heavy methodologies (such as the cascade model).
- Agile methodologies.

All software development process models can include the structural generic activities that have been described in this paragraph, but each of the them apply a different emphasis to each activity and define a work stream that involves each structural activity (and the software engineering actions with their tasks) in a different way and at different times. (see figure 2.2 below).

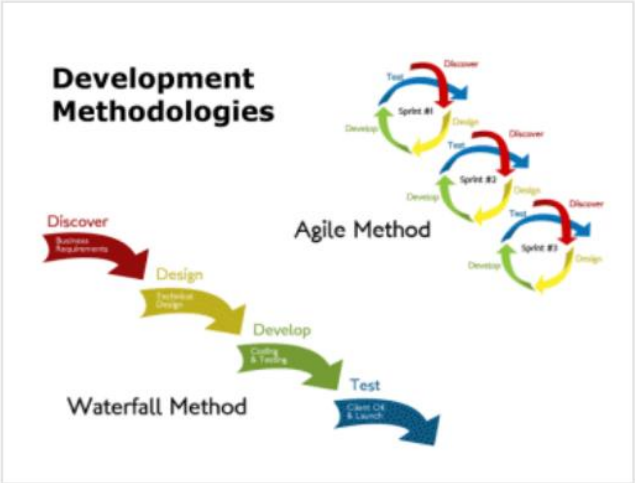


Figure 2.2 – The software Life-Cycle development in Agile Vs Waterfall methodologies (source: University of California, San Francisco)

2.3. Waterfall Model

The waterfall model is a prescriptive model according to which the realization of a software product consists of a well-defined sequence of phases.

They are called prescriptive because they prescribe a set of elements of the process: structural activities, software engineering actions, tasks, results, quality assessment, change control mechanisms for each project, also prescribing how they are related to each other.

The cascade model was adopted by computer scientists in the early 1970s as the first working method with the aim of reducing development costs, compared to previous practices, and of starting to use a certain work pattern. The waterfall model is probably the most widespread software development process in the world because it recalls the assembly line typical of industrial production, even if it does not have a pipeline structure (parallel development). It suggests a systematic and strictly sequential approach to software development in which each phase produces a precise output (deliverable) that is used as input for the next phase (from which the metaphor of the waterfall derives). The

big amount of documentation is needed to inform people who will be working in the next step and who are not familiar with the requirements of the project.

The main feature of this model is the rigid sequence of phases, which involves the complete absence of overlaps between them, and the absence of recycling, ie the inability to return to one of the previous stages to change something (Fig. 2.2). Another characteristic is the conviction that it was possible to design the application correctly right from the start, also thanks to the substantial stability of the requirements for the software of that time, a fact that, today, is highly unlikely.

Topics that are still widely considered are, for example, discipline (following a certain way of working according to precise rules), the planning (the need to dictate timescales and respect them), but also knowledge of the objectives before the implementation of the product and the division into phases.

The adoption of these principles may seem extremely productive but their application lead to the difficulties of coordination between the different activities (phases). Assuming that the requirements can be frozen at the end of the specification phase (and therefore cannot be modified), a large amount of documentation must be produced after each activity to make the work to be done in the next phase as clear as possible to colleagues. The difficulty of understanding what to do is one of the main causes of the increase in the number of errors made.

This problem is also highlighted by the absence of any form of feedback, both between the activities and the people who carry out the work, but also with the customer, with which there is contact only at the beginning of the project (specification phase) and at the end when you deliver the product.

A further difficulty of this model is the rigidity of its application: it is, In fact, necessary to complete a phase before you can start the next phase.

The inflexible division of the project into distinct phases makes it difficult to comply with the customer's requests for changes in requirements.

It is precisely for this reason that this model is only suitable if the requirements are clear from the outset and are unlikely to change during development.

Any difficulties or changes to the requirements lead to delays in the phases and to the entire project and this leads to higher development costs and a postponement of the “time to market”. The so-called time-to-market is another issue in this methodology. The time that can pass from project commission to project delivery to the customer can last even

years and this can be a problem because a product may already be old or obsolete after so long.
A further side effect is the inability to estimate resources and costs accurately until at least the first stage of analysis has been carried out.

2.4. Agile as Management mindset

For these reasons, software engineering has progressively tried to evolve from the more traditionally waterfall approach to the concept of agile software development.
Agile methodologies are a series of software development practices that have been emerging since the early 2000s. The basic idea of agile methodologies is not to be predictive, i.e. they do not try to predict how the system will evolve, but to be adaptive, i.e. they propose values and practices to better adapt to the evolution of user requirements. In software projects, customer needs are constantly changing: although having all user requirements in advance is a desirable aspect, it is often not achievable. (Bigatti, 2002).

> *"Everything changes in the software. Requirements change. Design Change. The business changes. The technology changes. The team changes. The team members change. The problem isn't change, per se, because change is going to happen; the problem, rather, is the inability to cope with change when it comes".*
>
> *Kent Beck*

When faced with turbulent situations where the market or technology changes rapidly and is difficult to predict, it is necessary to adopt a completely different approach to project management.

If it is not possible to be sure that no wrong choices are made, the alternative approach may be that, whatever happens, the reaction is immediately possible and inexpensive.
It is really important also the concept of iteration and continuous feedback that is well suited to modern dynamics. Each iteration has a limited duration of time, generally from one to four weeks. In jargon it is said that each iteration is "timeboxed" and can be imagined as a small project to be carried out, in which the usual phases of the software development process are present (see Figure 2.2), with the difference, compared to the old methodology, that there is an immediate response from the client, especially given by the tight collaboration between the development team and business experts (one of the major principles defined by Beck in the Agile Manifesto[1]).
Synthetizing the main techniques used by this approach are:

- Scheduling: The scheduling is not calculated on the basis of a detailed breakdown of all planned activities, from the beginning to the end of the entire project, but on the contrary, a detailed scheduling is developed for a near future -short term- (4 weeks). In addition, the scheduling is updated at regular intervals (1/2 weeks) called sprint.
- Flexible planning: activities are classified according to their uncertainty and creativity. Planning is therefore very flexible
- Time management: with the management of traditional waterfall projects the project manager had to ensure that the results were achieved by trying to manage the target time already set ex ante. With the agile method the opposite approach is followed, dividing the project into regular intervals (sprint) and then identifying the expected results at the end of each. No delays are allowed during execution and each project team must report any results achieved (sprint review).
- Specifications: Specifications evolve over time with a high number of iterations at the release of each new solution, each of which allows each team member to

[1] The Agile Alliance created a manifesto in which the principles and values of this methodology were highlighted in order to spread it around the world to encourage other developers to create better software. These are the 4 most important values on which the poster is based (without prejudice to the value of the items on the right, they considered the items on the left more important):
- Individuals and interactions rather than processes and tools.
- Software that works more than comprehensive documentation.
- Collaboration with the customer rather than negotiating contracts.
- Responding to change rather than following a plan.

have a deeper understanding of the problem and specifications. (Cantamessa, Montagna, 2016).

To have an even clearer and more explicit view, it is interesting to analyse the substantial differences between this innovative method and the more traditional waterfall one, taking into consideration the most significant tool of project management, the so-called ferrous triangle of which we have already spoken in the previous paragraph (Fig. 2.1). It is interesting to note how the importance attached to each of the key constraints' changes using these two different approaches.

As we can see from the figure 2.3 below in the traditional waterfall one the features or requirements are kept as fixed, clear and well defined, proceeding with the develop of the project only after the estimation of cost and time. On the other side, in the agile methodology the only thing that can change abruptly and iteratively is the purpose and the features, while the cost and the time slots to follow are kept stable.

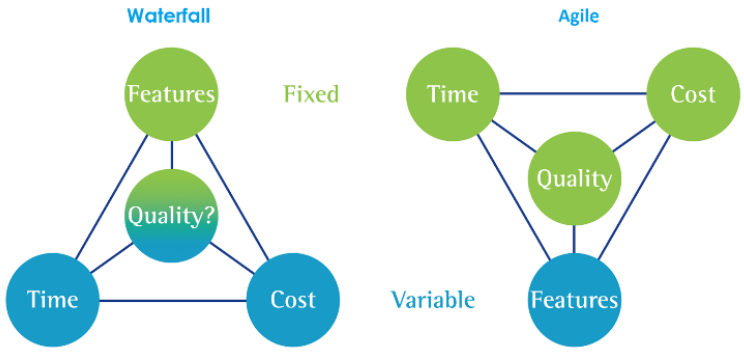


Figure 2.3 – How the ferrous triangle changes between agile and waterfall method (source: https://www.agilebusiness.org)

Therefore, if in the waterfall model it is the planning that guides the entire project, in the agile model the main scope it is to maintain a good quality trough fixed time-boxed iteration even changing the requirements. (Gerard, 2015).

2.4.1. Focus on one of the Agile frameworks: Scrum

In some way, we can see the Agile methodology as a collection of those techniques that transform traditional methods and share values and principles of the Agile Manifesto. Under this name, in fact, are grouped innovative methodologies (see figure 2.4) as Extreme Programming (or XP, conceived by Kent Beck), Crystal (designed by Alistair Cockburn), Scrum (designed by Ken Schwaber) and Lean Software Development (designed by Tom and Mary Poppendieck), which can be defined agile because they allow the specifications to be constantly reviewed and changed during development by actively exchanging information and opinions with the client.

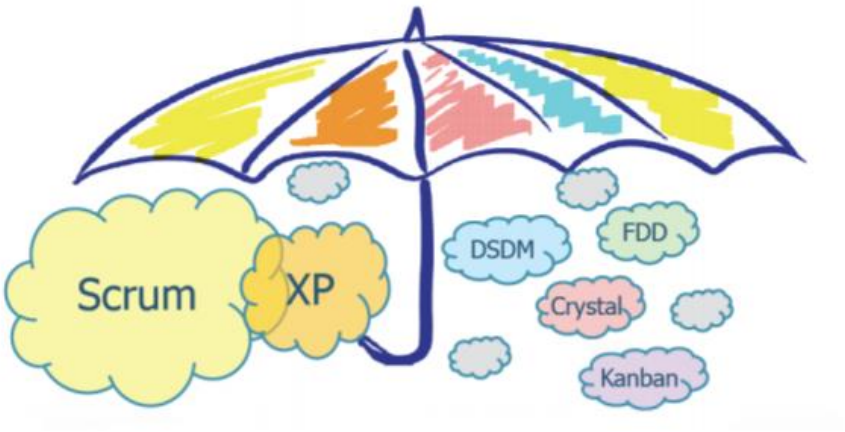


Figure 2.4 – Figure showing how the Agile method collects the others sub-methodologies or framework

Scrum is the most widespread Agile, particularly suitable for complex and innovative projects. It is a framework, a set of practices, that divides the process of managing a project in sprint to coordinate the process of product or software development. An iterative process in which the sprints last from 1 to 4 weeks. In this way, Scrum is not a rigid method, but rather represents a basis on which to build a process of development, which will be adapted from time to time to the reality in which it is placed. It is not by chance that Scrum's motto is "the art of the possible". (Casoni, 2010)

The theory behind this method is that of empirical process control (empiricism), according to which, on the one hand, knowledge comes from experience and, on the other hand, decisions are based on what is known. For this reason, an iterative process is

estimated with an incremental approach that optimizes, step by step (and sprint by sprint), predictability and risk control.
A method that is based on the principles of transparency, inspection and adaptation.
The main components of SCRUM are divided into: roles, artefacts and events.
There are 3 roles defined within the Scrum Team, and they work in close connection to ensure a continuous and fast flow of information.

- SCRUM MASTER: the process manager, who must ensure that the Scrum methodology is understood and implemented successfully. He must ensure that the team works in a manner consistent with the development of the project, remove any external obstacles to the Scrum Team that have an impact on productivity and organize and facilitate comparison meetings.
- PRODUCT OWNER: the person who knows all the requirements of the product and carries out the interests of all stakeholders.
  The interface between the business, customers and product requirements on the one hand and the team on the other. It must maximize the value of the product and the work done by the Development Team.
- DEVELOPMENT TEAM: the group of cross-functional and self-organized professionals, whose number is usually maintained from 5 to 9. He is in charge of product development and functionality testing and is responsible for organizing priorities and transforming them into tasks to be completed in order to complete that particular sprint.

The artefacts are 3, designed to maximize the transparency of key information (both for the Scrum Team and for all stakeholders) and the opportunity for inspection and adaptation.

- PRODUCT BACKLOG: the document that contains the list of all requirements necessary for the implementation of the project. The Product Owner is responsible for its content, availability and the ordering of its elements according to their respective priorities. In that case could be helpful the development of the product backlog iceberg with all the stories organized by priority. (Cohn, 2004)

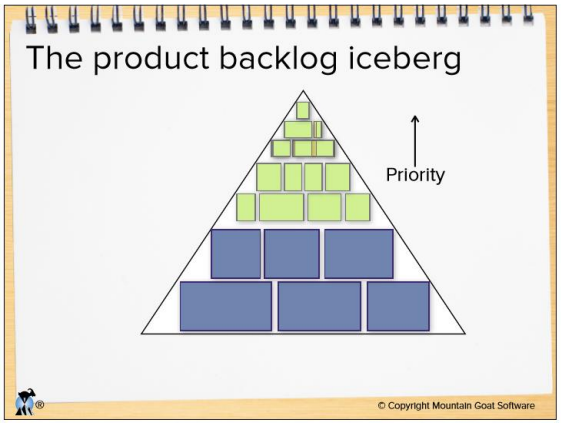


Figure 2.5 – A representation of the product backlog iceberg (source: Cohn, 2004)

- SPRINT BACKLOG: the document that defines all the tasks to be completed in the individual sprints. It is a forecast made by the Development Team in relation to the priorities indicated in the Product Backlog and the work necessary to achieve the objectives of the sprint.
- INCREASE: the sum of all Product Backlog elements completed during a sprint and during previous sprints. At the end of the sprint the increase will have to be realized according to what has been agreed by the Development Team to guarantee a usable product.

Four formal events are used in Scrum (with fixed duration) to create regularity, synchronize activities and minimize the need for undefined meetings. The objective of these events is to allow critical transparency and inspection of the progress of the project.

- SPRINT PLANNING: the meeting in which the Product Owner drew up the Product Backlog and, in the presence of the Development Team and the Scrum Master, describes the most important items and the objective to be achieved in the following sprint. At the end of the meeting the Scrum Master can fill in the Sprint Backlog.
- DAILY SCRUM: a short daily comparison (lasting 15 minutes) between Development Team and Scrum Master, which notes the work done the previous day and creates a plan for the next 24 hours (until the next Daily Scrum) to predict and synchronize activities.

- SPRINT REVIEW: a review at the end of each sprint to assess whether the goal has been achieved and with what results. The entire Scrum Team participates and, in addition, also the client of the product, which will be shown the work done up to the sprint.
- SPRINT RETROSPECTIVE: a further retrospective analysis carried out with the participation of the entire Scrum Team to assess what to do, what to stop doing and what to improve in the next sprint to achieve even more efficient performance.

So, as we have seen, the elements that characterize Scrum are many, but some certainly have a greater weight than others. In particular, a "new" concept is introduced: The User Story.
Describing a user story is very complicated and often its identification can be difficult. A good starting point to understand what we are talking about is the INVEST model. It is an acronym that is dissolved as follows:

- **I**ndependent
- **N**egotiable
- **V**aluable
- **E**stimable
- **S**mall
- **T**estable

According to this INVEST setting, a User Story will therefore have to meet several conditions.
It must not depend on any other story (Independent); it must be possible to clarify it in all its aspects (Negotiable) through the debate until all the actors agree on its content; it exists only if the end user also benefits (Valuable) from its realization and it must be possible to estimate it (Estimable); it must also be small (Small) and objectively verifiable (Testable).
In addition, during the "negotiation" phase, all obstacles that may prevent or slow down the realization of a story must be identified. In this way it is easier to highlight the risks and act in time to remove any impediments.

At this point, when a story is identified, it must also be recorded and catalogued in the Product Backlog iceberg (Fig. 2.5), where, as we have already said, each story is associated with a priority so that you can decide which ones to make before and which ones after.

In a nutshell, the use of Scrum offers many advantages. Here are some considerations:

- It allows you to work productively even in particularly chaotic and confusing situations.
- It is strongly oriented towards concrete results and change management, rather than precise planning.
- It leads to a strong involvement of the client and to an assumption of collective responsibility on the part of the team.
- Encourages the creation of cohesive teams.

At the same time, it has some intrinsic characteristics that increase the fears in adoption:

- It is a process that defines exclusively project management practices.
- It does not provide guidance on how to conduct other key disciplines (resources allocation management, analysis and design, testing, configuration management, etc.) and it is therefore appropriate to integrate it with other approaches to compensate for the missing parts.

2.5. Principal issues in SME IT Consultancy Companies

One of the biggest problems for SME[2] IT consultancy companies, is that the technological development and the ever-increasing drive of companies towards innovation, has led these companies to an almost uncontrollable exponential growth, in terms of new resources and new projects to be completed.

All this has meant that most of the leaders focused more on the proper development of projects rather than how to organize the job of the employees, often underestimating the problem. This, in the long term, leads to misalignment of the roles and responsibilities of the various resources, lack of communication and significant managerial problems that result in delays in the progress of the project and automatically loss of consumers.

This is clearly linked to another frequent issue, that of resource allocation management. In a software development company, any element used for product development could be intended as a resource for the project, from software libraries to production tools or human resources.

Taking into account the fact that SME IT consulting companies were usually born with two, maximum three resources, all developers or programmers, and have seen exponential growth over the last decade, most of these have never posed the problem of creating tools to obtain immediate visibility of the company in terms of resources or costs, having always been too focused on the goal that for them was the most important one, to write code correctly in order to maintain costumer's fidelity.

[2] Small and mid-size enterprises are businesses that maintain revenues, assets or a number of employees below a certain threshold. Every country or economic organization has its own definition of what is considered a small and medium-sized enterprise. In the United States, there is no distinct way to identify SMEs, but in the European Union, a small-sized enterprise is a company with fewer than 50 employees, while a medium-sized enterprise is one with fewer than 250 employees. In addition to small and mid-size companies, there are micro-companies, which employ up to 10 employees.

But it is very important to be able to follow a strategy for the management and allocation of resources, because a lack of resources can block and therefore delay the delivery of the product to the customer, while an excessive number of resources, on the other hand, can lead to an increase in the final cost of the project, and therefore to a loss in profit.
It should be born in mind that the more the company grows, the more complicated it will be to manage resources in terms of actual allocation.
Many companies still continue to use various tools, such as spreadsheets, which are totally inefficient to solve such complex problems for an increasing number of projects to follow.
Moreover, many companies are not informed about all the possible benefits that they could obtain using PMIS[3] (Braglia & Frosolini, 2014).
For the resolution of this type of problem there are many methodologies to use and digital tools that make us understand how much pushing project management towards digitization is perhaps the most appropriate solution to the whole list of problems that we have just listed.

2.5.1. Towards Project Management digitalization

Currently, many IT consulting companies find themselves with non-integrated, disconnected solutions that are no longer enough to manage an increasing number of projects. (Braglia and Frosolini, 2014).
Moreover, competition in this industrial sector is becoming increasingly pressing, so companies are doing everything to excel and gain a competitive advantage over their

[3] A Project Management Information System (PMIS) is a standardized set of automated project management tools available within the organization and integrated into a system.

rivals, increasing coordination, management and control with the help of tools, techniques and information systems (Hazir, 2015).

Since the early 1950s, software packages and solutions have been considered very important tools for programming, monitoring and control. Now these software solutions that help in all these core tasks for the execution and development of projects of various types of companies are called Project Management Information Systems (PMIS) (Archibald, 2003).

PMIS are documents that contain information about the project, but also procedures regarding preparation, maintenance and use. These procedures are then used for project management in all its parts, from monitoring to scheduling and planning.

For all these considerations, and because of the huge amount of work and the highly complex technological environment in which an IT company's project manager works daily, PMIS is considered as a crucial element in the field of project management today (Ahlemann, 2009). This is because PMIS offers not only support in planning, organising and controlling projects, but also in the whole decision-making process of the project (Caniels & Bakens, 2012).

Moreover, one of the main advantages of PMIS is that it allows teams to follow the progress of projects by providing immediate visibility and a consistent flow of information through the vision of the various levels of task completion, the overall status of the project itself plus a whole range of more detailed information such as scheduling, budget, resources, costs, time etc. (Braglia & Frosolini, 2014). All this through a wide and varied range of reports, graphs, tables and formats that can be produced in a very simple way through these electronic solutions (Archibald, 2003).

Nowadays, commercial offers on the market for software solutions of this type are increasingly in visual and interactive properties (Hazir, 2014): they are software-packages such as resource planning, risk management, scheduling and control packages, which are combined with different reports and other graphic output generators (Archibald, 2003).

Although PMIS were designed to have a strong link with the success of the project, since they are able to increase the budget and, in addition to all that we have already said, assist and help in scheduling meetings (Raymond & Bergeron, 2008).

There are several PMIS in the market and it is up to the organization to understand which one is the most suitable according to its needs. Furthermore, it must be taken into account that in the development of information systems, stakeholders need to be considered as

future users of the system, managers of the organization, internal or external professionals. This can lead to more problems in choosing the most suitable software package as people have different skills, know-how and backgrounds. (Sakka, Barki and Cote, 2016).

One of the most important trends in development today has certainly been the use of the Web and the Internet, which has offered unlimited accessible solutions for many services, as well as the ability to transfer information quickly, accurately and trackably (Alshawi & Ingirige, 2003).

Web-based solutions have allowed any authenticated user to access the information database from anywhere at any time without a specific software installation, reducing the common problems of geographically distributed organizations as well as costs, even significantly (Tarantilis et al.., 2008).

There are many other added values: solutions are much more comprehensible and easier to understand, data and solutions are transmitted in accessible and interoperable virtual platforms, and everything is available online and in real time (Shim et al.., 2002).

By now more and more important functionalities are brought to the user interface, also thanks to the use of previous project models and libraries of planning solutions that can be adapted to the needs of new projects: graphic interfaces, on-screen menus, online servers, all tools that allow you to store and edit information in real time between the various collaborators of the project (Archibald, 2003).

These are the prerogatives that characterize the digitalization of project management: the ability to have access to multiple computer systems without the need for separate software packages, allowing the PM to have all the necessary information in digital form on a web platform.

So it is clear that in recent years there have been important changes in the development of project management software, both graphically and purely functional: more projects are now manageable in a common database, Gannt diagrams always improving, histograms in time scale for people and money, general project plans with integrated milestones, etc. (Archibald, 2003).

The development of web solutions related to project management is in a key position to reduce the gap between project management theory and practice, but also to provide better tools for project managers. (Hazir, 2015.)

The digitized, cloud-based project management solution enables individuals to collaborate more naturally and more closely all the time. This development also supports the connectivity and usability of virtual project teams. (Shim et al., 2002.)

The process of selecting the most suitable solution to support the project management needs of a specific organization can be a challenging and highly complex task. The choice of these cutting-edge project management solutions is wide, but even these solutions differ in many respects. (Archibald, 2003.)

In order to compare the software packages available on the markets, it is necessary to consider different points of view in relation to different areas of project management.

There are several steps that can be identified in the process of selecting a digital project management solution. Defining user needs is the first step in starting to evaluate the solution of a specific organization or company.

Sufficient analysis is required to understand not only the current organizational needs, but also the future needs and desires of the organization. Selecting a feasible solution for a specific organization requires a systematic and comprehensive approach to ensure that the best possible decisions are made during the project management digitization process.

In conclusion, to carry out a good market research on the software solution best suited to the needs of your company, is certainly a small investment compared to the negative impacts and project costs that would be incurred if they were made wrong choices caused by the lack of a proper software tool.

3. EMPIRICAL CASE STUDY– PLM Systems

Figure 3.1 – PLM Systems' logo (source https://alteafederation.it/)

"On the one hand, the consultancy dedicated to the optimization of the process of development of industrial products, through new methodologies that allow to increase quality by shortening development times, on the other hand all those improvements that through IT technology guide and support the production world: these are the missions of PLM Systems."

3.1. Company background

PLM Systems was founded in 1996 as a high-level consulting company mainly concerning PLM (Product Life-Cycle Management[4]). In 2004 it was acquired by the Altea Federation group, above all to obtain a necessary and fundamental financial security since at that time there were not more than 20 employees, but despite everything it has practically continued to live on its own life.

Until 2012 it was divided into two Business Units only: one called MTS (Make to Stock) and the other ETO (Engineering to Order), which were basically two strands of companies on which employees carried out extensive consulting, regardless of the technology used or the activity done.

After that, a few years ago, the company was divided into four Business Units, three strictly dedicated to individual technologies, TeamCenter of Siemens, Windchill of PTC, Enovia of Dassault, and the fourth that deals with consulting and innovation in a transversal way, so not purely technical activities but governance.

Figure 3.2 – A representative view of the company organization (source: https://alteafederation.it/)

[4] PLM is not only an information technology, but rather an integrated approach, based on a set of technologies, methods of collaborative work organization and the definition of processes: the life cycle of products and services, from their conception, to their development, to their market launch, to their withdrawal, all things that require a strategic approach to the management of processes, resources and information.

The first three BUs carry out real project activities guaranteeing at least 40-45% of the total sales. We talk about software development activities such as porting (updating of the previous version of the software), or new implementations. They develop projects concerning lean product development, with the respective methodologies and product development process modelling and improvement, others of product life-cycle management with services such as product configuration and personalisation, parts management and physical testing, but also digital experience with data analysis and IoT, AR and ML solutions implementations.

Moreover, especially since the last few years, more than half of the turnover is given by AMS (Application Maintenance System), because a PLM system should not only be developed - writing requirements, detailed design, implementation, release into production - but, being a solution that effects on average from 200 to 600 users who usually work with integrated CAD that can create a lot of problems, is strictly necessary a remote help-desk service, or anyway the implementation of corrective activities as well as evolutionary activities.

The last BU instead deals with consulting in the truest sense of the term, both on PLM Systems projects activated on a customer that must be maintained, but also on third party projects managed by the customer on which support activities are made under the name of the customer.

In a nutshell, the consultants of this division can be considered as a sort of decoder between the two worlds, rather than tutors and controllers of the work of suppliers who perhaps use PLM different from their own. In addition, they are often called upon to implement a software selection for new customers who want to enter the PLM world without having the necessary knowledge (organization of workshops, theoretical explanations, analysis of software decontextualized of technical details).

PLM Systems therefore does not have an organizational structure divided by functions, but by technologies. In fact, it is true that everyone has their own skills and competences, but it is also true that everyone is able and capable to do everything. This could be considered a disadvantage, but in reality, being still a fairly small company, with a limited number of sites and customers reached (Fig. 3.3), it seems to be the most appropriate and effective organizational solution.

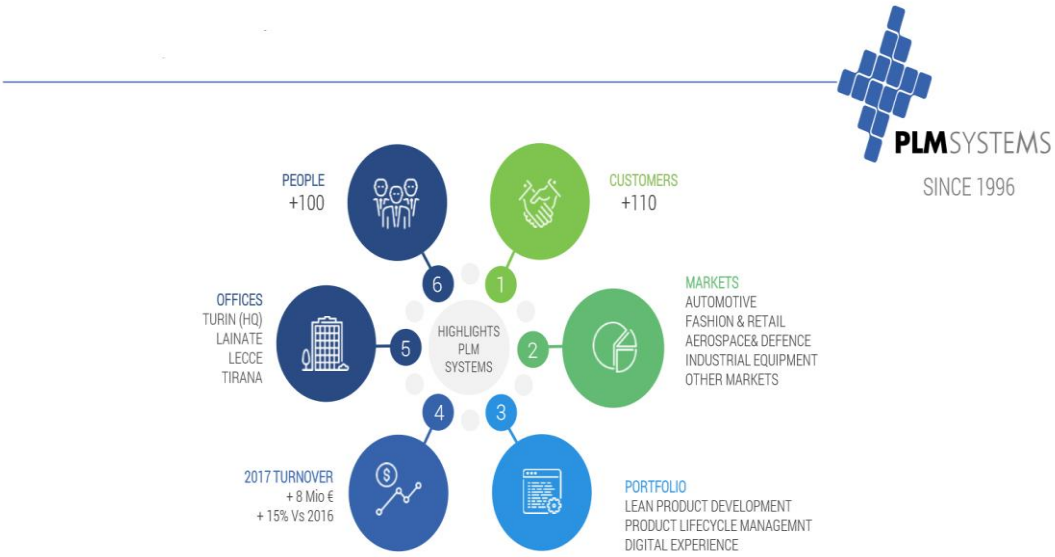


Figure 3.3 – A little bit of info about PLM Systems (# of employees, # of costumers, turnover, offices and markets reached) (source: https://alteafederation.it/)

3.2. Analysis AS-IS of the company's issues

Starting to analyse PLM Systems' situation, the main problem of the company came immediately to light: trying to push as much as possible towards innovation, with the intention of developing new types of projects, has created many complications at the organizational level, prompting leaders to invest in a new resource as a project manager. As the name of the company says itself, PLM Systems was born for the development of PLM solutions (Product Life-Cycle Management) but, in recent years, customers demand has increasingly focused on developing IoT solutions, and this has led to a substantial change in the company's organization and a new team of developers has been created for the implementation of some projects based on the IoT[5].

So the work to be done within the BU under my interest has been divided in two different teams: one dedicated to IoT implementations (ThingWorx software development), and the other concerning Windchill PLM implementations, plus some resources that are focusing on the development of a third tool, Navigate, which is nothing more than a connector between the two worlds: it allows to see on ThingWorx the data taken from Windchill.

The aim in that part of the case study was to understand how the projects in PLM Systems were implemented, monitored, planned and managed in all senses, with particular attention to the tools and techniques used by the various project managers in the company. Most of the research data were collected through interviews of 45-60 minutes on average, carried out with the main managers and developers of PLM Systems over a period of about two months: from the beginning of September to the end of October 2018.

---

[5] The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. The term "thing" or "object" can be interpreted more precisely as categories such as: devices, equipment, plants and systems, tangible materials and products, works and goods, machinery and equipment. These connected objects, which form the basis of the Internet of Things, are more appropriately defined as smart objects and are characterised by certain properties or functions. The most important are identification, connection, localisation, the ability to process data and the ability to interact with the external environment.

3.2.1. Concerns in the BU of PLM domain

The techniques used for the development of PLM projects are varied. Before the project manager can actually start with the project plan, it must be considered that there are some preventive phases: sales -procurement of costumers-, presale – following phase: when a customer has effectively been hooked, a more purely technical figure is committed to presenting the operation and offering of the proposed service-, and finally, if from the presale comes out a potential activity, the project manager can start with an estimation of the tasks and with the compilation of the high level WBS (Work Breakdown structure)[6]. Most of the time the WBS is presented and discussed with the customer. Starting from the client's own requirements, the main activities are divided into subtasks and, through an analysis of historical data, an attempt is made to make an estimate as reliable as possible. Clearly if the tasks have already been carried out in the past, the contingency[7] will be very low, almost tending to 0, since you know the dynamics, but if the project is new or very risky, you will come up with a higher contingency. Another thing to emphasize is that in the WBS are not written the names of the resources, but only the degree of seniority.

The next step is to transcribe the WBS as a Gannt Chart[8].

---

[6] A work breakdown structure (WBS) is a key project deliverable that organizes the team's work into manageable sections. The Project Management Body of Knowledge (PMBOK) defines the work breakdown structure as a "deliverable oriented hierarchical decomposition of the work to be executed by the project team."

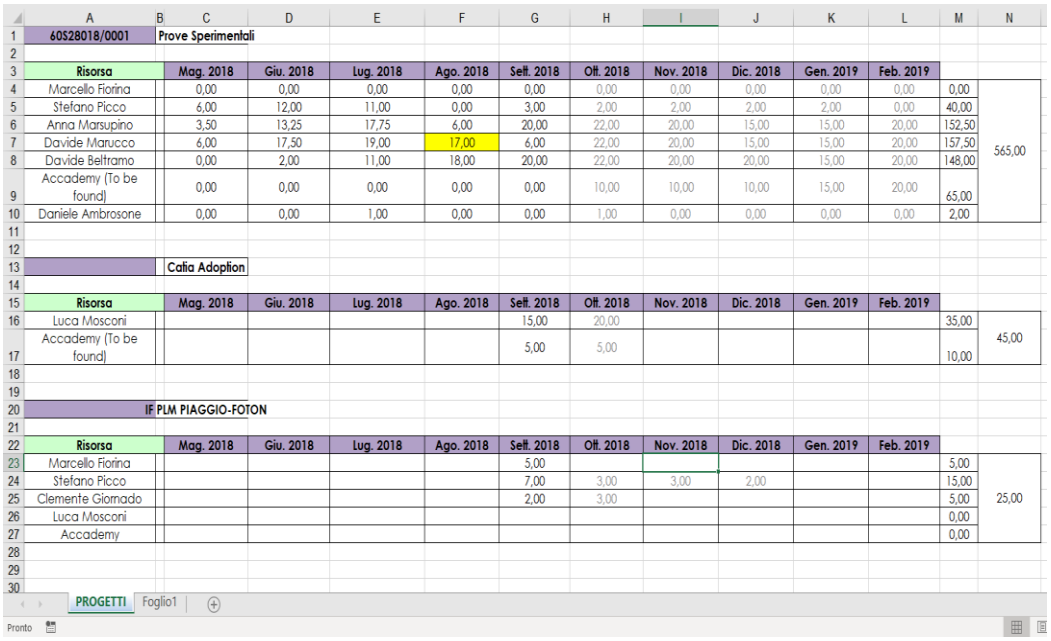
[7] In discussions of risk, the term “contingency” is often understood to be a number added to an estimate for project costs or durations to cover some element of risk or uncertainty. Owners establish contingency levels for each project based on acceptable risk, degree of uncertainty, and the desired confidence levels for meeting baseline requirements. When used to absorb the impacts of project uncertainty, the contingency is a form of risk mitigation, and so in evaluating potential project contingency funding, owners should apply risk assessment and probabilistic estimating techniques.

[8] A Gantt chart is a popular and useful graphical tool in project management. On the chart, tasks are shown on the vertical axis while the scheduled time-spend is laid out on the horizontal axis. Each task is represented by a bar that shows the time required for the project and dependencies, which simply means the interlinkages between various activities in the project.

If the projects are short enough, with a not too long-time horizon and with a limited number of activities, the Gannt diagram is made on excel, on a template offered to the company by a supplier. When, on the other hand, the projects are much more substantial and harder, even in terms of time, Microsoft Project (MP) is used.
So, it is true that here in PLM Systems, a project management software is actually used but is used as a human brain: of 100% of functionality are used less than 20%.
In fact, the allocation of resources on MP is not made (is also always used with infinite resources) but is simply made the transposition of the WBS to have a graphical view of the time frame over which the project will develop. This is because the client needs to have a visible proof of the progress of the project. And here comes out the second problem of MP: the suitability. Very often happens in fact that the customer does not have MP licenses and therefore it is not possible to export to his computer: in other words, it is almost useless.

3.2.1.1. Lack of Logistical Management tool for Resources Allocation

It is clear that one of the biggest problems of that company is how to manage the resources allocation. Indeed, my boss asked me, from the first days, to try to find a better solution than the one already used in the company.
As I have already said, no specific tool is used in PLM Systems to manage the allocation of resources, other than a simple excel sheet. So, he forwarded to me the usual excel sheet that Project Managers use during the meetings to administer the forecast allocation and I immediately noticed that it was absolutely inefficient.
As we can see from figure 3.4 below, the sheet was developed with three tables, one for each costumer, where for each resource there are specified the working days carried out in each month for that specific commission. The aim was that, as soon as a senior manager knew how many days that resource would be used for that job, he would update this sheet, which was then shared with all the other seniors.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	60S28018/0001		Prove Sperimentali											
2														
3	Risorsa		Mag. 2018	Giu. 2018	Lug. 2018	Ago. 2018	Sett. 2018	Ott. 2018	Nov. 2018	Dic. 2018	Gen. 2019	Feb. 2019		
4	Marcello Fiorina		0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
5	Stefano Picco		6,00	12,00	11,00	0,00	3,00	2,00	2,00	2,00	2,00	0,00	40,00	
6	Anna Marsupino		3,50	13,25	17,75	6,00	20,00	22,00	20,00	15,00	15,00	20,00	152,50	
7	Davide Marucco		6,00	17,50	19,00	17,00	6,00	22,00	20,00	15,00	15,00	20,00	157,50	
8	Davide Beltramo		0,00	2,00	11,00	18,00	20,00	22,00	20,00	20,00	15,00	20,00	148,00	565,00
9	Accademy (To be found)		0,00	0,00	0,00	0,00	0,00	10,00	10,00	10,00	15,00	20,00	65,00	
10	Daniele Ambrosone		0,00	0,00	1,00	0,00	0,00	1,00	0,00	0,00	0,00	0,00	2,00	
11														
12														
13			Catia Adoption											
14														
15	Risorsa		Mag. 2018	Giu. 2018	Lug. 2018	Ago. 2018	Sett. 2018	Ott. 2018	Nov. 2018	Dic. 2018	Gen. 2019	Feb. 2019		
16	Luca Mosconi						15,00	20,00					35,00	
17	Accademy (To be found)						5,00	5,00					10,00	45,00
18														
19														
20			IF PLM PIAGGIO-FOTON											
21														
22	Risorsa		Mag. 2018	Giu. 2018	Lug. 2018	Ago. 2018	Sett. 2018	Ott. 2018	Nov. 2018	Dic. 2018	Gen. 2019	Feb. 2019		
23	Marcello Fiorina						5,00						5,00	
24	Stefano Picco						7,00	3,00	3,00	2,00			15,00	
25	Clemente Giornado						2,00	3,00					5,00	25,00
26	Luca Mosconi												0,00	
27	Accademy												0,00	
28														
29														
30														

PROGETTI | Foglio1 | ⊕

Pronto

Figure 3.4 – A rough excel sheet used by the company to manage the allocation of resources (source: PLM Systems documents)

However, this tool was definitely ineffective and not productive. For example, the fact that there were three different tables for different commissions, with the same resources, was not very efficient because, if it was necessary to add some resources or commission, it would have to proceed with the creation of a new table. Moreover, the formatting did not allow to make filters by resource or even less by customer, while in my opinion this could be useful to have a clearer view of the situation.

3.2.1.2. Introducing of a new tool for Resources allocation management

Following one of the meetings held by seniors for the forecast of resource allocation in fact, I realized that the most common problems were almost two:

1. To understand what the percentage of monthly allocation for each resource was (both in the forecast and especially in the past months in order to have a feedback at the time of the final balance of hours on SAP).
2. Know what resources could be allocated in each month. In fact, from the excel sheet that was used previously, there was no immediate visibility of who was available and who was not.

So, I decided to introduce a new tool that could lead some benefits. First, I proceed by changing the main sheet for entering monthly days per resource, creating a single table with customer, commission, resource name and months as columns (Table 3.1) so that when a new resource is assigned to a new job, you simply have to add a new row to the table.

Table 3.1 – Excel Table to enter input data on working days of resources for each month (source: own elaboration)

	A	B	C	D	E	F	G	H	I	J	K	L	M
1						Consuntivo					Preventivo		
2	CLIENTE	COMMESSA	RISORSA	mag-	Giu. 2(	Lug. 2(	Ago. 2(	Sett. 2(	Ott. 2(	Nov. 2(	Dic. 2(	Gen. 2(	Feb. 2(
19	60528018/0010	Altea - Leonardo APM Evo	Bruno Stabile						7				
20	i0528018/0015-01 - Olivet	Sviluppo Applicazioni Thingworx (SmartFarm 1.0)	Antonio Di Stasi					2,5	6				
21	i0528018/0015-01 - Olivet	Sviluppo Applicazioni Thingworx (SmartFarm 1.0)	Sara Saccente					7	6				
22	i0528018/0015-01 - Olivet	Sviluppo Applicazioni Thingworx (SmartFarm 1.0)	Gianpiero Di Blasi					9	2				
23	i0528018/0015-01 - Olivet	Sviluppo Applicazioni Thingworx (SmartFarm 1.0)	Nicolò Mari					1	5				
24	i0528018/0015-01 - Olivet	Sviluppo Applicazioni Thingworx (SmartFarm 1.0)	Luigi Dell'Aversano						2				
25	i0528018/0015-01 - Olivet	Sviluppo Applicazioni Thingworx (SmartFarm 1.0)	Michele Santoro					2,75					
26	60528018/0019	Portale Fornitore THX	Daniele Ambrosone						3	1,5			
27	60528018/0022	Bottero - AMS Agosto Ottobre	Daniele Ambrosone					13	12				
28	60528018/0025-01 - Celli	Implementazioni Thingworx	Antonio Di Stasi										
29	60528018/0025-01 - Celli	Implementazioni Thingworx	Sara Saccente										
30	60528018/0025-01 - Celli	Implementazioni Thingworx	Gianpiero Di Blasi					11					
31	60528018/0025-01 - Celli	Implementazioni Thingworx	Nicolò Mari					10					
32	60528018/0025-01 - Celli	Implementazioni Thingworx	Luigi Dell'Aversano					11					
33	60528018/0025-01 - Celli	Implementazioni Thingworx	Michele Santoro					1					
34	60528018/0026-01	AMS - Sviluppo DNA MAP	Antonio Di Stasi					17	6				
35	60528018/0026-01	AMS - Sviluppo DNA MAP(Configurazione Macchine)	Nicolò Mari					2					
36	60528018/0026-02	AMS - Sviluppo DNA MAP	Sara Saccente					6	6				
37	60528018/0026-03	AMS - Sviluppo DNA MAP	Gianpiero Di Blasi					1	5				
38	60528018/0026-04	AMS - Sviluppo DNA MAP	Nicolò Mari					5	5				
39	60528018/0026-05	AMS - Sviluppo DNA MAP	Luigi Dell'Aversano					5	6				
40	60528018/0026-06	AMS - Sviluppo DNA MAP	Michele Santoro					5	5				
41	Avio	AMS	Marco Cresto	0	0	0	0	21	22	22	22	22	22
42	Avio	AMS	Andrea Pellegrino	0	0	0	0	1	1	1	1	1	1
43	Avio	AMS	Nadia Bertelli	0	0	0	0	3	5	2	2	3	3
44	Avio	Evolutive	Raissa Secci	0	0	0	0	0	0	10	0	0	0
45	Avio	Evolutive	Nadia Bertelli	0	0	0	0	6	0	5	0	0	0
46	Formazione PTC	Formazione	Michele Santoro					3	3				
47	Magneti-Marelli	AMS	Nadia Bertelli	0	0	0	0	6	6	5	6	6	6
48	MASTER HOMI 4.0	Formazione	Raissa Secci					1	4	4	4		
49	MASTER HOMI 4.0	Formazione	Simone Condelli					1	4	4	4		
50	MASTER HOMI 4.0	Formazione	Davide Beltramo					1	4	4	4		
51	Master Polito	Formazione	Sara Saccente					5	5	5	4		

Foglio Gestione dati | Allocazione % risorse (MENSILE) | PROGETTI | Allocazione Risorse Totali

Moreover, being a table, you can filter by customer or commission or, even better, by resource in order to have a quicker and more efficient view, and, above all, it can be used as a database for the pivot tables created to solve the two main problems listed in the previous chapter.

For the first of them, I thought to create a pivot table that had as line labels the customers and for each customer the specific commission related to that customer, with the total number of days carried out, each month, by the resource chosen through the filter for that job (top left corner). I have also added a bar chart in order to have immediate visibility of the percentage of allocation and availability of that single resource for each month. (Fig. 3.5).

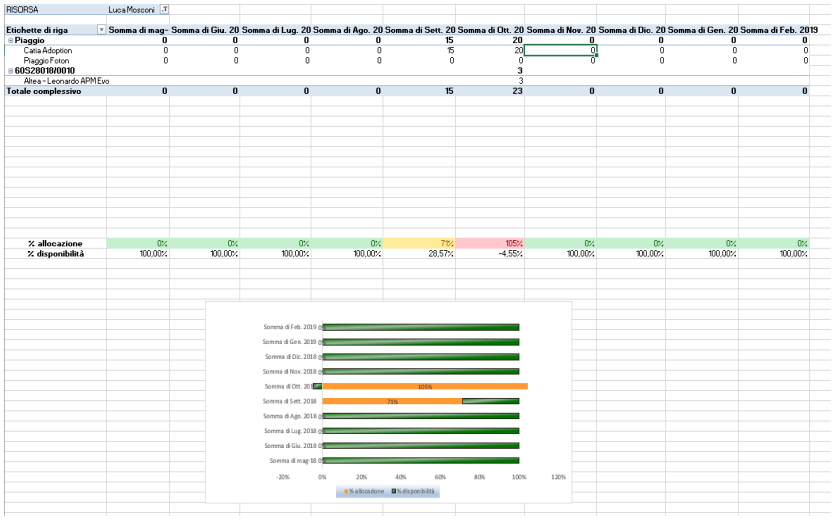
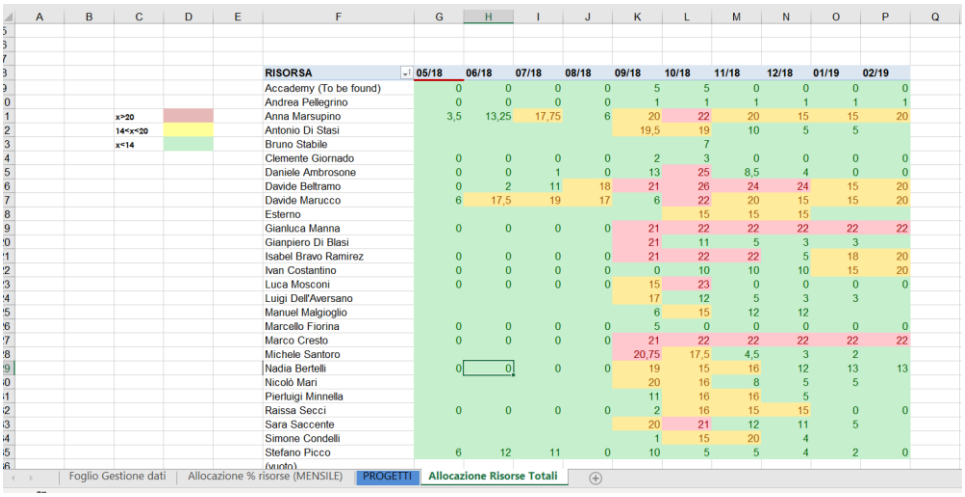


Figure 3.5 – An excel tool to have immediate visibility about % of allocation and % of availability of each resource (source: own elaboration)

But the most common problem is certainly the second one. It often happens in fact to need more days than those already provided for a given job and you need to know in a timely manner what are the resources that can actually be allocated in that month: a general view of all resources by month, without filtering resource by resource.

So I created another pivot table made in this way:



RISORSA	05/18	06/18	07/18	08/18	09/18	10/18	11/18	12/18	01/19	02/19
Accademy (To be found)	0	0	0	0	5	5	0	0	0	0
Andrea Pellegrino	0	0	0	0	1	1	1	1	1	1
Anna Marsupino	3,5	13,25	17,75	6	20	22	20	15	15	20
Antonio Di Stasi					19,5	19	10	5	5	
Bruno Stabile						7				
Clemente Giornado	0	0	0	0	2	3	0	0	0	0
Daniele Ambrosone	0	0	1	0	13	25	8,5	4	0	0
Davide Beltramo	0	2	11	18	21	26	24	24	15	20
Davide Marucco	6	17,5	19	17	6	22	20	15	15	20
Esterno						15	15	15		
Gianluca Manna	0	0	0	0	21	22	22	22	22	22
Gianpiero Di Blasi					21	11	5	3	3	
Isabel Bravo Ramirez	0	0	0	0	21	22	22	5	18	20
Ivan Costantino	0	0	0	0	0	10	10	10	15	20
Luca Mosconi	0	0	0	0	15	23	0	0	0	0
Luigi Dell'Aversano					17	12	5	3	3	
Manuel Malgioglio					6	15	12	12		
Marcello Fiorina	0	0	0	0	5	0	0	0	0	0
Marco Cresto	0	0	0	0	21	22	22	22	22	22
Michele Santoro					20,75	17,5	4,5	3	2	
Nadia Bertelli	0	0	0	0	19	15	16	12	13	13
Nicolò Mari					20	16	8	5	5	
Pierluigi Minnella					11	16	16	5		
Raissa Secci	0	0	0	0	2	16	15	15	0	0
Sara Saccente					20	21	12	11	5	
Simone Condelli					1	15	20	4		
Stefano Picco	6	12	11	0	10	5	5	4	2	0
(vuoto)										

x>20
14<x<=20
x<=14

Foglio Gestione dati | Allocazione % risorse (MENSILE) | PROGETTI | Allocazione Risorse Totali

Figure 3.6 – Pivot table that counts and shows the availability of all the resources for each month (source: own elaboration)

The legend on the left specifies that if the value of days of work of a resource in a month is greater than or equal to 20, the resource is almost no longer allocable -given that on average in a month there are 22 working days- and then the colour is red. In this way, just look for the resource in green, or at most in yellow, to understand who can definitely be allocate in that month. Clearly, both pivot tables are updated when a change is made in the main sheet (Table 3.1).

In addition, this excel file has been uploaded to Microsoft Teams, a sharing tool used in the company, which ensures that changes made to the shared file are updated for all team members and warrant a better coordination and communication between employees.

4. PROBLEM SOLVING IN THE BU OF IOT DOMAIN

Moving our analysis to the team that is developing IoT implementations, the problems I found were so consistent that I had to dedicate an entire chapter of this dissertation to try to figure out it.

The IoT's team is a team of 7 people, 5 developers based in Turin and the so-called "project managers" in Milan. So, first, there is a real logistics problem: every morning the PM was supposed to send an email to the guys in Turin with the tasks to perform, using a day-to-day approach, and if someone needed explanations about problems that arise daily, the only solution was to make a call to the office in Milan.

Another of the most significative issue is that the roles of each employee are unclear: the person who should be the "project manager" of the group is an architect of IoT solutions, so he has strong technical skills and competences, but he is also the only one who approaches the customer and performs purely commercial activities. He therefore finds himself wearing more hats, when instead he should focus only on the actual development of the project, using his strong technical skills and leave this task to a new professional figure that can efficiently manage this type of project which, as we will see below, due to its intrinsic variable nature, needs a different management approach than that used for PLM projects.

As Gallup says in "The Real Future of Work: Agility Issues", the agility in a company could be measured by asking employees two questions:

- In my company, have we the right mindset to respond quickly to business needs?
- In my company, have we the right tools and processes to respond quickly to business needs?

If these questions had been asked in PLM Systems at the beginning of September, they would have both had negative answers. In fact, the two most relevant problems can be synthetized as 1) the lack of knowledge, and so automatically the non-implementation of the Agile methodology, 2) the absence of a support tool.

That's why the goal of my analysis was to introduce the team to the new organizational framework, Scrum, with the support of a management software properly chosen.

A first pre-selection among the most competitive software on the market was carried out and only then the Scrum methodology was applied to a real project conducted by the team with the support of the three selected software, in order to understand which was the optimal for the company.

4.1. Software Pre-Selection

In recent years, Project Management software products have undergone considerable development in terms of application and use in wide and diversified contexts.

The software selection consists in evaluating the adoption of a project management program within a range of options, proposed by operators on the market, more or less suitable for the specific activity. The most important factor in the implementation of a market research of a project management software is the analysis: it is important to have clear objectives to understand which characteristics should have the most proper solution. Therefore, the need to define a rational set of criteria arises with the need of select the most appropriate software to meet the goals of an organization.

But, if we follow a multi-criteria approach well done, before the final selection must always be carried out a pre-selection, necessary to reduce the set of alternatives to be analysed, since the market of tools for project management is increasingly thriving.

There are several methods for analysis and selection that can range from the simplest basic intuition to counting the number of requirements or something intermediate, the important thing is that the pre-selection is done in a consistent and quantifiable way to be effective.

After having carried out a market research of the most used and competitive project management tools and software, I focused the study on 6 of them - chosen also thanks to the interviews with company managers as well as web research: Monday, Jira, Microsoft Project, VisualStudio.com, Project Libre and Freedcamp.

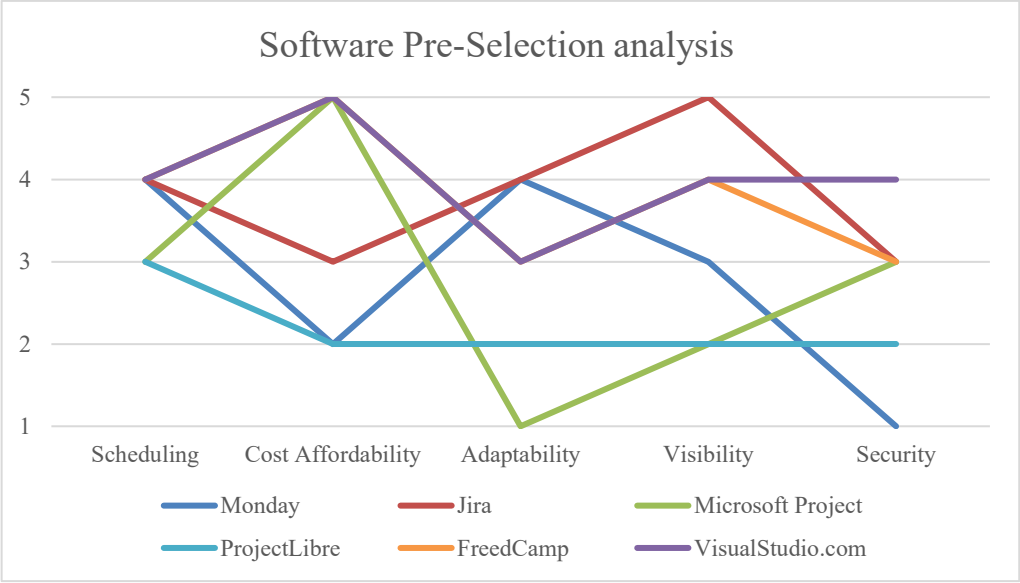
After analysing all the literature concerning the project management techniques in a software development company, as well as especially the most frequent problems that come up, a group of functional requirements has been drawn up, requirements that the software to be selected should have in relation to the size, the issues and the activity of the company.

No software ever has all the desired features but having the most important ones in mind is the basis for a good analysis that you will then complete after an initial pre- selection.

Considering that the overall market offer seems to have reached a stage of maturity with small upgrades that do not change the architecture but introduce from time to time only marginal improvements, particularly in terms of usability of products, I decided to focus the pre-selection considering 5 major areas of interest that could contain the most relevant requirements for this type of software research, and go into more detail only in the final selection (after having actually tested the alternatives):

-**Scheduling**, i.e. all the functions related to tasks management and the allocation of resources
-**Cost Affordability**, the annual cost
-**Adaptability**, in term of suitability and compatibility with other systems
-**Visibility**, the possibility of having a total vision of the project's scope and everything related to communication between users
-**Security**, data backup and security

Another very important thing in implementing a right decision analysis, is the evaluation of the score. It is clearly necessary not to use a single subjective judgment but, for example, to create a defined set of discrete values, a scale, which corresponds to a certain explanation for each value.
Having carried out this pre-selection at the beginning of the internship and therefore not having yet worked empirically with the team, I could not have a clear idea of how much each criterion was relevant and it would not have been totally correct to assign a weight to each of them.
Estimating therefore that each big requirement has equal relative importance to the others, a scale of 5 values, from 1 to 5, has been used: 5 is the max and means that alternative fully satisfies the decision criterion, 1 is the min and means that alternative fully dissatisfies the decision criterion.



Graph 4.1 - Line graph that shows the values associated to each criterion for each software

Analysing the functionality of the chosen software we created this graph that is an excellent visual tool to understand which of the alternatives can actually be taken into account for a more specific study.

- **Monday**: we have heard about this tool thanks to an advertisement and we decided to know more about it to understand if it was worth testing it. Looking at the features proposed in the online presentation page it has strong scheduling (4/5) and visibility (3/5) and a very powerful integrability or suitability (4/5) with other systems (Dropbox, Import/Export from Excel, Trello). What is disconcerting, however, is the cost (2/5) that increases significantly with the number of registered users and that is closely related to the addon of important features such as security (1/5). In a nutshell, only if you paid for the Pro upgrade would you have all the best features like 24/7 costumer service or total data security with single sign on.
- **Jira**: is Atlassian's best known software for the use of the Agile methodology, one of the most requested in the market and for this reason we could not consider it among our alternatives. According to the managers who have already used, it has very good performance in scheduling (4/5), adaptability (4/5) and visibility (5/5) ensuring the ability to manage tasks by assigning priorities, estimated execution times and a global view of the progress of the work among all users connected, as well as compatibility with external tools. Small point to disadvantage in relation

to the cost (3/5) that, despite not being too high, increases when you want to add features that at first impact do not seem important but that in the long run can also be essential, such as data security (3/5).

- **Microsoft Project**: is the most widely used project management software. Also in PLM Systems it is used, but only for projects developed following the waterfall methodology. We decided to take it into account in this analysis because in the company they already know how to use it and because of the zero cost, having already licenses (5/5). It has good scheduling performance (3/5) and security (3/5) being one of the most powerful products of Microsoft. A low performance visibility (2/5) due perhaps to the complexity of use and bad evaluation in suitability (1/5) as it is a system almost completely closed: if you, as business consultant, present an advance of the project carried out on Microsoft Project to a customer who does not have Microsoft Project, a flood of problems such as exporting data to other formats etc. will emerge.
- **VisualStudio.com**: is another Microsoft tool that we got to know by chance by talking with the colleagues. VisualStudio.com includes a whole circle of connected software that makes the final package very good. Thanks to Azure DevOps you can use Agile tools and collaborate more effectively, thanks to Visual Studio Code you get a very powerful, free and open source editor that can run anywhere. It is therefore an IDE (Integrated Development Environment) that offers excellent performance [Scheduling (4/5), Adaptability (3/5), Visibility (4/5), Security (4/5)]. The only more regrettable point would be the cost but, having a partnership with Microsoft, PLM Systems has the licenses for this alternative too, so Cost (5/5).
- **Project Libre:** This tool was created with the idea of providing an open source replacement for Microsoft Project, but expectations do not seem to have been met. Watching some demo videos on how this tool works, we could see that it has quite good requirements in scheduling (3/5) but poor functionality in all other 4 criteria (2/5). This is mainly because not enough information has been found about it. For example, even in the official site that gives you the opportunity to do a free trial for 14 days, there is no price transparency, i.e. does not explain what the cost will be after the free trial. This just makes me think that the price is too high to be shown: Cost (2/5).

- **FreedCamp**: the knowledge of this tool has been acquired during project management's course and, being relatively a good tool, we considered to analyse it in this pre-selection. The most positive feature is certainly the zero cost (5/5). But a professor of Polytechnic of Turin one day said, "when the goods are for free, you are the good" and it is normal to think that the functions of this tool may be inferior to those of the others. However, having already tried it previously, it is possible to ensure that it was all in all a good tool in all the fields taken into account [Scheduling (4/5), Adaptability (3/5), Visibility (4/5), Security (3/5)].

In conclusion, from the Graph 4.1 it is easy to see how the red, the orange and the purple lines, representing Jira, VisualStudio.com and FreedCamp respectively, are the best alternatives regarding Scheduling, Visibility and Security. Jira loses a bit in cost, VisualStudio in adaptability, and FreedCamp in security, but still remain better than the other alternatives considered.
In order to proceed with the final decision for the proper digitalization of the project management environment in the company, these three tools will be analysed and tested in depth, using them during the development of a new project assigned to the IoT's team: the DEMO project.

4.2. Presentation of the IoT DEMO project

The most significant project that IoT's team was working on, was a project for the development of an application for the management of Smart Farm for a very important customer at national and European level: an application where, thanks to IoT-based sensors, is possible to control and monitor all agricultural machinery, from the setting of the timer for sprinklers, to the check of the actual agricultural machinery, to monitor humidity, temperature and light or automate irrigation systems with a single click. The project started in May 2018 with the development of the application from scratch.

But how is Smart Farm structured?

The application has a landing page with a menu on the left containing all the sections that can be visited: Dashboard that shows the list of the companies, Areas of company, Devices that belongs to each company, Users where you can manage your profile, Search measures, Advices, Alarms and Commands where there are all the data received from the devices.

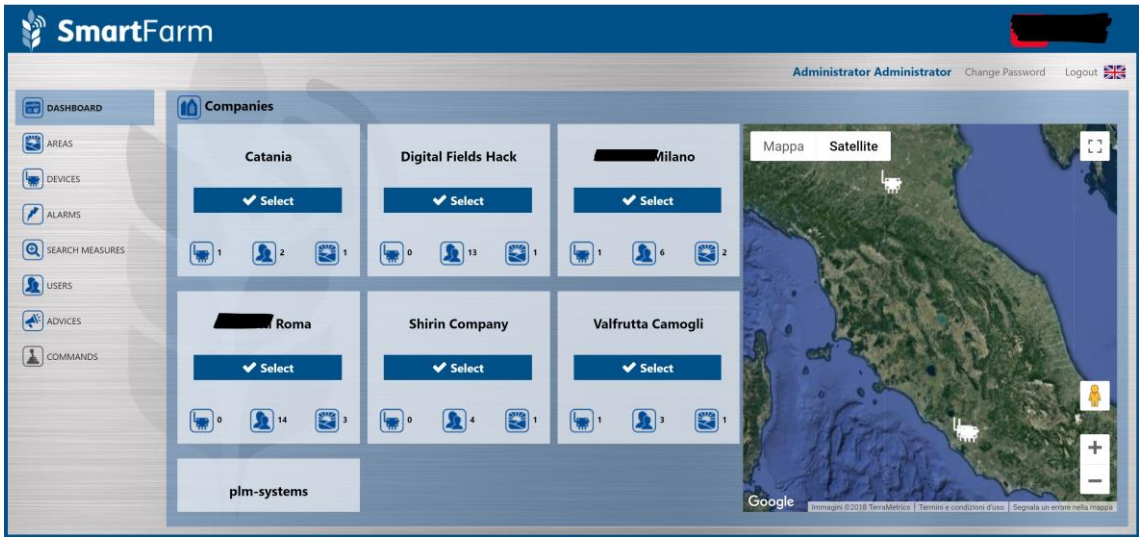


Figure 4.1 – Home Page of SmartFarm application (source: PLM Systems' documents)

At the end of September 2018, a new request was made by the client, to create a sort of archetype called Smart Object instead of Smart Farm, which had two basic requirements:

1. That it was an application with the basic features very similar to its "parent" but easily and quickly customizable according to market and customer's needs.
2. That it could also be used as a kind of DEMO: the customer can test the operation of the product without making substantial changes in the original application.

So, I started to follow the project team starting from this request, considering it suitable to implement the agile methodology and especially the Scrum technique using the three software - Atlassian's Jira, Microsoft's VisualStudio.com and FreedCamp - chosen after the pre-selection analysis (Graph 4.1).
In this way it has been possible to make a comparison between the three selected software to understand which the optimal is, considering several key criteria for the team in question and be able to convince the company to use the tool to get better results in the long term.
Furthermore, thanks to the fact that the new request from the client could be seen as a sort of generalization of what had already been done over a period of almost 3 months (May/September), I could practically compare the project management methodology used in the past by the company with the Scrum framework, identified in the literature as one of the solutions to most of the problems in a software development company.

4.2.1. Practical Application of Scrum Framework

Several meetings or more precisely defined as Sprint Planning, have been organized, where me and my team have outlined the sub-activities to be implemented, first defining the priority for each of them, and only then, discussing with the team members who would have preferred to carry out certain tasks, we proceeded with the assignment of them, also specifying an estimated time of development and the so-called velocity.
The velocity of a team is the effort that can be made during a sprint. Theoretically it is therefore very easy to calculate this information:

$$Vt = nT * nD \tag{4.2.1}$$

Where nT is the number of developers and nD is the number of days the sprint was run. The units of measure used in Scrum can be different: story points, days, hours, depending on the preferences of the various development groups. We have decided to adopt story points as units of measure. Empirically, however, it can be established that the average speed (Vm) of a good team should be between 60% and 70% of the theoretical speed.

$$Vm = nT * nD * 0{,}65 \tag{4.2.2}$$

Clearly, these are only indications: you need to use common sense to understand the actual ability of your team and work to ensure that the velocity is as much as possible.

Beyond common sense, many exploiters of the Scrum framework, use the Planning Poker Fibonacci[9] method to assign a difficulty level to each story, and then calculating the sum of the story points at the end of each sprint, compare this number to the average velocity found with (4.2.1). This concept is very important because it helps the team and the PO to understand if the assignment of tasks of the sprint has been done properly or if it is necessary to decrease/increase the velocity in the next sprint. In the development of this project, we used the same methodology.
Additional help to keep track of development progress and respect of the deadlines is given by the Burndown chart which can be of two types:

- Release Burndown: measures the amount of remaining releases of a Product Backlog with respect to the total release plan.
- Sprint Burndown: measures the number of remaining elements of a Sprint Backlog during a Sprint.

In this project, the second type of chart was used: for each sprint performed, the actual remaining working hours were tracked in relation to the estimated remaining hours.
Moreover, every development of the scrum has been recorded on the three software selected in the preselection, VisualStudio.com, Jira and FreedCamp, through a 30-days free trial, giving the team the possibility to use these support tool.

[9] Planning poker is the most widely used method for agile estimation of requirements and is based on the relative size of a requirement in terms of expected effort to implement it. The unit of measure used for the estimate are the story points, abbreviated as SP.
The team plays planning poker; the estimation of each activity is achieved by exploiting the principle of "wisdom of the crowd" and unanimously (consensus).
Each member of the development team holds a deck of cards with a sequence of numbers, the Fibonacci Succession (1,1,2,3,5,8,13,21,34,55,89,144...), and must choose the one that is the most appropriate value for the story under consideration. each shows the card chosen at the same time: in case of discrepancies, the group discusses the requirement again trying to understand the different points of view and the reasons that led to discordant assessments and repeats the process.
This is done until the team reaches consensus, i.e. unanimity, and all - or almost - show the same card.

I. SPRINT N°1: New App DEMO

Following what I discovered writing the literature about the Scrum framework, with the role of product owner I called a first Sprint Planning on Monday, October 1, 2018. During this meeting, starting from what were the customer's requirements, and following the cornerstones of the agile methodology, all the team members collaborated to plan the first sprint in the most efficient way.

Since the first step was to configure and install the tools for the new development environment, was created an Epic, a grouper of stories that could be imaged as a sort of tag, called "Technical Stories" under which to enclose all the stories regarding this type of tasks.

Analysing in more detail the requirements to be met, a story called "Preparation of the new DEMO environment" was created and grouped under "Technical Stories". It is important to note that this story does not slavishly follow the typical structure of a User Story, just because it is a Technical Story, but it was still included in the product backlog of our project in order to be traceable.

The table below lists the various activities with their respective resources, priorities and estimated time.

Table 4.1 – List of tasks of "Preparation of the new DEMO application" story (source: own elaboration)

TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Configuration and installation of version 1.5 di Docker	Giampiero Di Blasi	Medium	5h
2	Installation and configuration of version 8.0 of server apache TomCat with port: 8081	Giampiero Di Blasi	Medium	2h
3	Configuration of version 8 of postgres database inside docker [Username: postgres - Password: postgres]	Shirin Mohammadyari	Medium	1,5h
4	Configuration of version 8.02 of thingworx [Username: Administrator - Password: Administrator]	Shirin Mohammadyari	High	4h
5	Installation of thingwork extensions	Antonio Di Stasi	Low	2h

Proceeding with the discussion on the development of the project, another group of tasks was identified relating to the analysis of the functions already existing in the parent application: be able to import them into the DEMO application and proceed with the

necessary changes and customizations, as well as be able to add the functions requested by the customer, that the original application did not have.
As for the first case, all these tasks have been grouped in a new story, called "Create/Import object" that has been inserted in the product backlog.
Unlike the previous one which was a Technical Story, this one can be catalogued as a User Story, as it follows what is the general structure of them:

As a { user },

I want { goal } so that

I { receive the benefit }.

“AS a developer I WANT to Create/Import objects SO THAT I have the new DEMO application”.

Specifically, these are the tasks, the resources to which they have been allocated and the respective priorities and estimated time:

Table 4.2 – List of tasks of “Create/Import objects” Story (source: own elaboration)

TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Import/Create Dashboard features	Antonio Di Stasi	Medium	2h
2	Import/Create Company features	Antonio Di Stasi	Medium	2h
3	Import/Create Areas features	Nicolò Mari	Medium	2h
4	Import/Create Devices features	Nicolò Mari	Medium	2h
5	Import/Create Sensors features	Nicolò Mari	Medium	2h
6	Import/Create Users features	Sara Saccente	Medium	2h
7	Import/Create Alarms features	Sara Saccente	Medium	2h
8	Import/Create Advices features	Sara Saccente	Medium	2h

Finally, it was decided to assign hours of work to carry out a fast-test of all the features imported and / or added, with the aim of finding and reporting on the product backlog bugs encountered.

So, another story has been created for this Sprint, called "Fast check of the new DEMO application" with its tasks, priorities and estimated times:

Table 4.3 – List of tasks of “Fast check of the new DEMO application” Story (source: own elaboration)

TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Fast check / test of the DEMO application	Sara Saccente	Medium	1,5h
2	Open SW for the bugs have been found	Nicolò Mari	Medium	1h

It was then necessary to assign to each of these 3 stories, a priority and above all an estimated value of difficulty to compute the velocity of that sprint.
It was therefore decided to give to the Technical Story (Table 4.1) a very high priority (since without those tasks you could not go ahead to complete the others) with a difficulty of 8, to the second story (Table 4.2) a high priority with an even higher difficulty, 13 points, and to the last story instead, consisting of only two tasks, a medium priority with difficulty 3.
The story points have been chosen through the planning poker of Fibonacci, already explained at the beginning of the paragraph.
We thought that the velocity for this sprint [8+13+3=24] was too low for a team of 5 developers and that we could add another User Story to finish within the first sprint.
Another of the customer's requests was to completely overturn the implementation of the Login Page, so it seemed more appropriate to implement a totally new one, creating a User Story called "Login Page" and giving to this last story a low priority compared to the other three, because the others were related, while the latter remains in its own right:

“As a USER I want to LOGIN to the Web Application SO THAT I can use my credentials”

The following table lists the sub-areas with their respective allocations of resources, priorities and estimated time:

Table 4.4 – List of tasks of “Login Page” User Story (source: own elaboration)

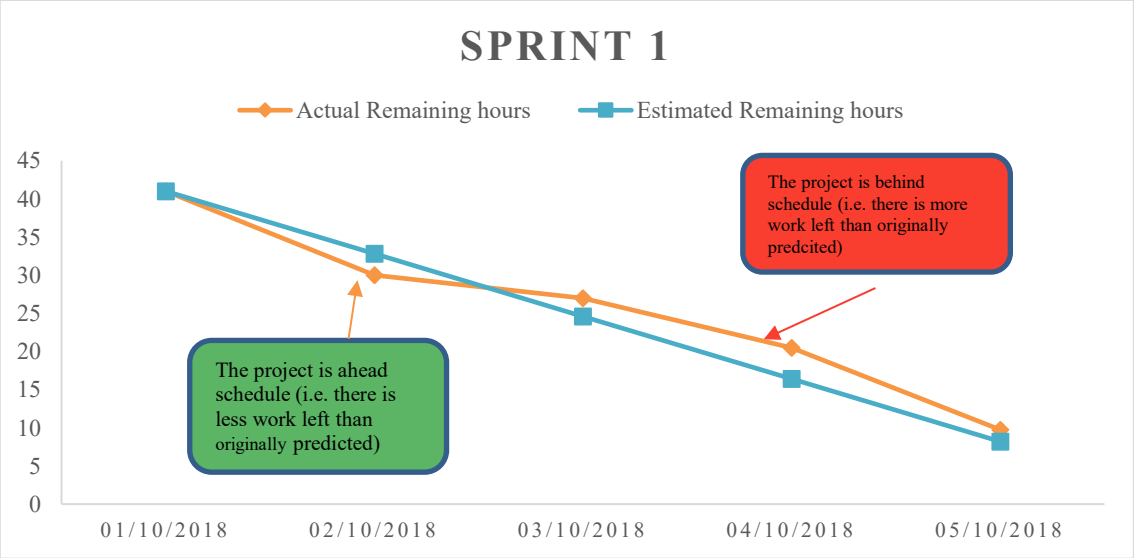
TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Button and Text Fields	Antonio Di Stasi	Medium	1h
2	Create the service to check the user existence	Giampiero Di Blasi	Medium	4h
3	Associate the link of Log in Page to the Log in button	Giampiero Di Blasi	Medium	1h
4	Test the funcionality of the new feature for successfull & not successfull login	Shirin Mohammadyari	Medium	2h

Following again the planning poker Fibonacci method we assigned a difficulty level for this history of 5 points, and we arrived at a velocity for the Sprint 1 of 29.
Considering the inverse equation of (4.2.1) we found that the right number of days should have been 5,8 so we decide to associate to this first sprint a duration of 1 week.

SPRINT N°1: REVIEW AND RETROASPECTIVE

At the end of the first sprint, a few hours have been dedicated to the sprint review and sprint retrospective, analysing and studying the burndown graph created by the software used.
As already mentioned at the beginning of this paragraph, it is a graph that helps to visualize the remaining effort for a certain period of time and is therefore an excellent tool for project management to help understand if you are behind or ahead schedule.

Graph 4.2 – Burn down chart of Sprint 1



Analysing Graph 4.2, which contains the working days of the sprint in the x-axis and the actual working hours in the y-axis, it is easy to observe that there have been quite considerable delays in the last three days of the sprint, up to 5/10 with a delay of about an hour and a half with respect to what was expected. However, to be the first sprint for the implementation of a new methodology, could be considered satisfactory.

II. SPRINT N°2: Tests and Bugs

On Monday 8th October the second Sprint Planning was called by me, to plan the new week of work of the team.
Thanks to the activities carried out in the "Fast Check" history of the first sprint, 5 bugs were detected and added to the backlog of our project.
A new story called "Fix Bugs" was then created consisting of these five subtasks, which have been assigned priorities and estimated development time:

Table 4.5 – List of tasks of “Fix Bugs” story (source: own elaboration)

TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Substitute email subjects and template from Smartfarm to Smartobject	Nicolò Mari	Medium	1,5h
2	Clicking on Breadcrumb does not work	Shirin Mohammadyar	High	1,5h
3	Wrong Labels and Data for Device table	Antonio Di Stasi	Low	2h
4	Wrong text in Pop-up when creating new company	Sara Saccente	High	2h
5	Change Title Bar to smart object (instead of smart farm)	Nicolò Mari	Medium	1h

To this story has been assigned a difficulty level of 13, since in discussions with the team it was decided to assign these tasks to the Junior resources in order to allow them to make practice, and a high priority.
The second step after planning the bug-solving is to do a full and complete test of every feature introduced in the DEMO environment. The second story has therefore been called "Full Test/Check":

"As a developer I WANT to make a Deep Test of the DEMO Application SO THAT I can go ahead with the requirements of the client".

For this particular story, a careful analysis was made with the team members, as it was decided to assign the complete test to those who had not imported /created the features, so as to have a more objective approach.

In the following table the detail of the assignments of the tasks with their estimated time and priorities:

Table 4.6 – List of tasks of "Full test/check" story (source: own elaboration)

TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Test Dashboard section	Giampiero Di Blasi	Medium	4h
2	Test Company section	Giampiero Di Blasi	Medium	4h
3	Test Areas section	Antonio Di Stasi	Medium	4h
4	Test Devices section	Antonio Di Stasi	Medium	4h
5	Test Sensors section	Antonio Di Stasi	Medium	4h
6	Test Users section	Shirin Mohammadyar	Medium	4h
7	Test Alarms section	Shirin Mohammadyar	Medium	4h
8	Test Advices section	Shirin Mohammadyar	Medium	4h

Considering the importance of this list of tasks, as it is essential to thoroughly test every feature introduced into the system also to find other possible bugs, and following the planning poker Fibonacci rule it was decided to give this story a score of 13 and a high priority.

So already with these two stories alone we came to a velocity for this second sprint of [13+13=26].

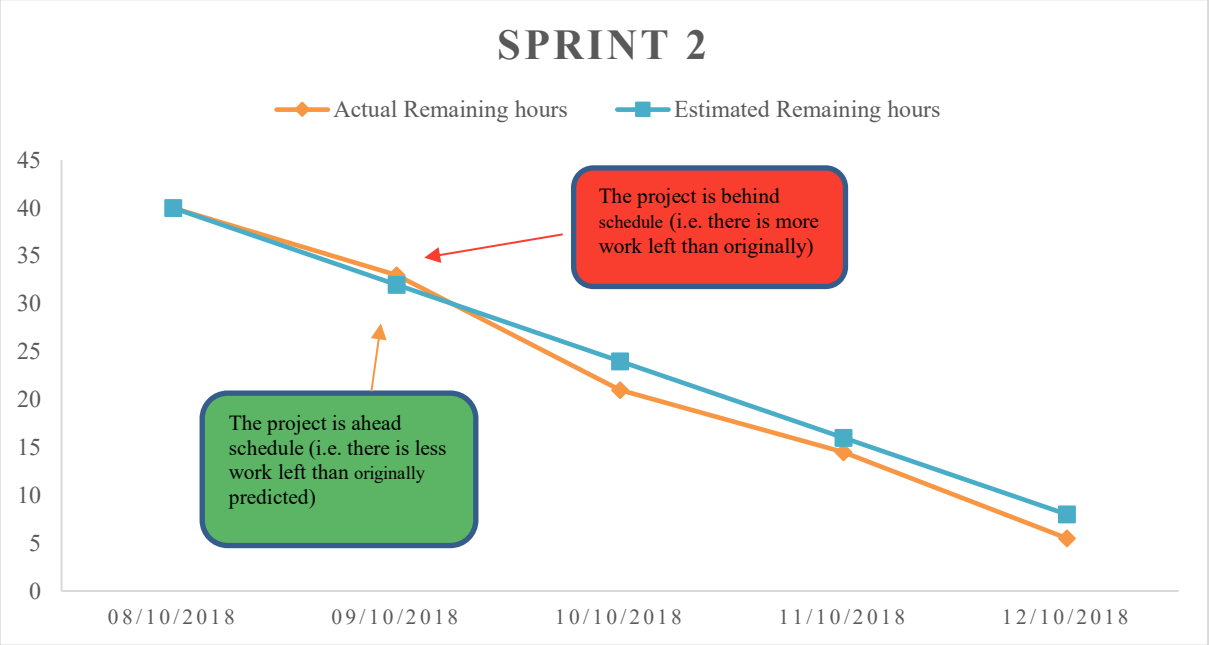
Since in the first sprint the velocity was 29 and we achieved the goal set on time, it was decided not to add more tasks in this second sprint being 26 very close to 29 and to the theoretical velocity [Vt =5*5=25].

SPRINT N°2: REVIEW AND RETROASPECTIVE

Also after the second sprint, a couple of hours have been spent to review project's progress, analysing the burn down graph (Graph 4.3).

The second week started with a slight delay, brought back from the first sprint, and stretched until 09/10. After that, following the development through the graphical burn down, we have been able to catch up, succeeding in completing some tasks before the estimated time, finishing the second sprint with project ahead schedule (less work left than originally predicted).

Graph 4.3 – Burn down chart of Sprint 2



III. SPRINT N°3: User Management

The third Sprint Planning has been called for Monday, October 15. After carrying out the Sprint Retrospective of the previous sprint, defining which tasks were carried out properly and which were not, and if any new bugs were discovered, we proceeded with the analysis of the new customer requirements that focused on user management.

The requests concerned the development of methods for the management of a new user, the forgotten password and for the issue of sending e-mails to confirm registration. Therefore, three different User Stories have been created, grouped under an Epic called "User Management".

The first, called "New User" has been developed as follows:

"AS a new User I WANT to sign in/register SO THAT I can create my new credentials",

Table 4.7 – List of tasks of “New User” story (source: own elaboration)

TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Creation of the landing page for the registration	Giampiero Di Blasi	High	2h
2	Creation of the form with the boxes in order to registrate the new user (Name, Surname, Email, Retype Email, Password, Confirm Password)	Shirin Mohammadyari	High	2h
3	Control check for the email format	Nicolò Mari	Medium	2h
4	Creation of pop-up for password requirements	Antonio Di Stasi	Medium	2h

Includes all the tasks related to the creation of a new web page, different from the Login page, which allows the new user to register to the system.
Also in this case, after defining the priorities for each task, we proceeded with the discussion with the entire team regarding the estimated development time and the assignment of each task to a specific resource.
In addition to the assignments for each task, a high priority was then given to this story - because all the other stories in that sprint are related to this one - and a difficulty level of 13.
The next story we built is the one about managing the user who forgets his password, called "Forget Password":

"AS a User I WANT to Handle Forget Password SO THAT I can enter in the System"

Table 4.8 – List of tasks of “Forget Password” story (source: own elaboration)

TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Create a message error when the password insert is wrong	Nicolò Mari	High	2h
2	Create an address link to another page to reset the forgetten password	Giampiero Di Blasi	High	4h
3	Create the form with the box to enter the email and a button with an associated link that send a mail to the user	Shirin Mohammadyari	Medium	4h
4	Create the standard email template and the generation of a new random password	Sara Saccente	High	2h
5	Test all the new features	Antonio Di Stasi	Low	4h

To this User Story has also been given a high priority, with a difficulty score of 13.

In addition, since we were specified that we should also manage the sending of e-mail confirmation of registration, and since the associated tasks were considered quite

consistent, we created a third and final story, called "Confirmation Email" always under the Epic User Management:

"AS a new User I WANT to receive a mail SO THAT I can have the Confirmation"

Table 4.9 – List of tasks of “Confirmation e-mail” story (source: own elaboration)

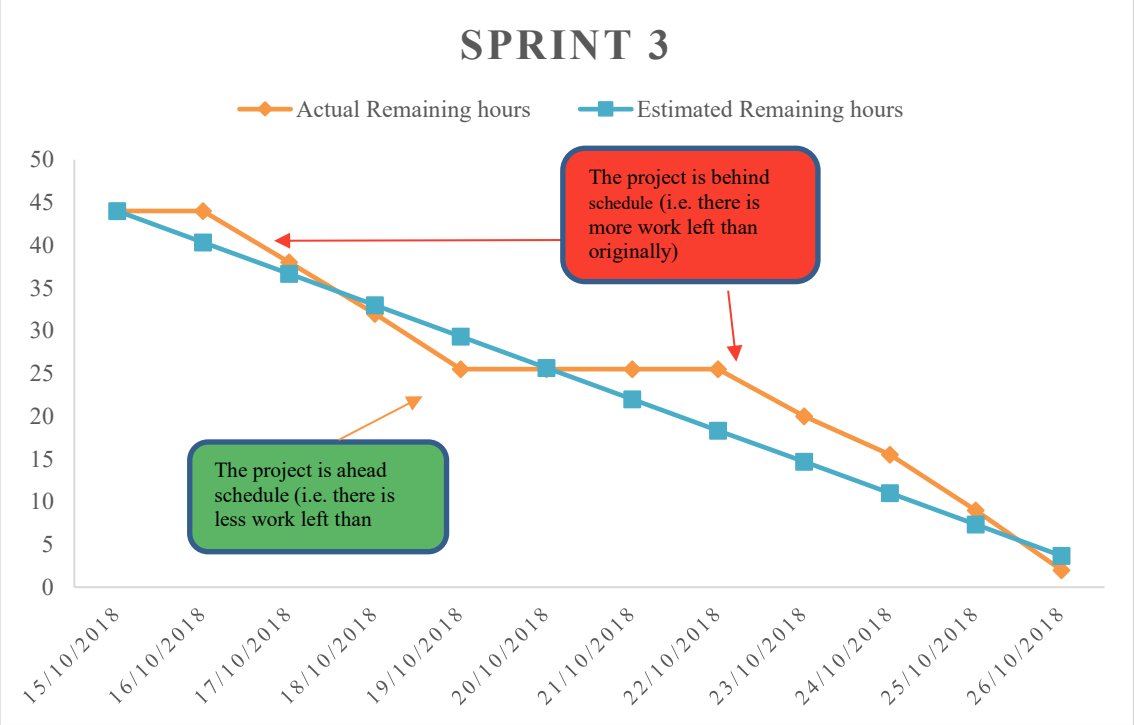
TASK	TASK NAME	ASSIGNED TO	PRIORITY	ESTIMATED TIME
1	Redirect the user to a new page that confirm the registration	Giampiero Di Blasi	High	4h
2	Send an email to the new user	Antonio Di Stasi	Medium	2h
3	Create the standard email template	Sara Saccente	Medium	2h
4	Create a method that generate a random password	Giampiero Di Blasi	High	4h
5	Create an address link that redirect the new user from the email to the login page	Shirin Mohammadyari	High	4h
6	Test all the new features	Nicolò Mari	Low	4h

In the table above we see the individual tasks with their respective resources, priorities and estimated time, always decided in collaboration with the team.
Even to this story, considered quite challenging because it was related to the management of the database, we decided to assign a difficulty score of 13 through the planning poker Fibonacci method.
So, because the velocity in this sprint would be [13+13+13=39], following the inverse equation (4.2.1) we found nD =39/5= 7,8 optimal days, so it was decided to estimate the total duration of this sprint at 2 weeks instead of 1 week like the other two.

SPRINT N°3: REVIEW AND RETROASPECTIVE

Even after the third and final sprint, which lasted two weeks, a meeting was organized to take stock of the situation.
As you can see from the burn down graph of this sprint (Graph 4.4), the progress of the project is always variable, but it never deviates too much from the straight line of the estimated remaining hours. We can see a considerable deviation only in the last week of the sprint, where the project was behind schedule, but still, at the end of the week we were able to conclude in time the scheduled activities, succeeding in ending with the project ahead schedule on 26/10.

Graph 4.4 – Burn down chart of Sprint 3



In conclusion, we have accomplished the DEMO project in 4 weeks. This month of work was very instructive both because we were able to expand our knowledge, competences and skills regarding the framework scrum, and especially because, by testing the two software by hand, we could proceed with a thorough analysis and selection of the best of them.

4.3. Laplace's insufficient reasons criterion as decision-making model for SW selection

After testing for 30 days the functionalities of the three selected software through the multicriteria approach preselection (paragraph 3.2) I had to concretely choose which is the best option for the company, considering more specific requirements and criteria.
To do this, it is very important to follow a structure and a decision-making model that is appropriate to the case under consideration, in order to ensure a high degree of scientific validity to the solution chosen.
In the literature there are different methods of decision making and they are based on different conditions: 1) in conditions of certainty, 2) in conditions of risk and 3) in conditions of uncertainty.
Uncertainty is always present in managerial contexts and even more in a selection for a management software, also because the whole decision-making refers to the future development of results.
Reading the literature and coming across a dissertation called "Selecting the optimal software solution under conditions of uncertainty" (Aldea, Olariu, 2014), I have been able to deepen the study of the Laplace's insufficient reasons criterion, which basically states that if there are not enough data available on the probability of different results, it is reasonable to assume that these are equivalent. So, if there are n results, the probability of each of them is 1/n. The approach then suggests calculating the expected payoff value of each alternative and that chose the alternative with the maximum value.
The main difficulty encountered in trying to structure a decision-making model for this software selection was to be able to give a consistent and well-motivated weight to each criterion, since the criteria chosen, which reflect the desired requirements of the end user, had the following characteristics:

1. They were all, or almost all, binary criteria and were not quantitative criteria
2. We did not have enough time to enter in details of all the functionalities that they actually have

3. The same criterion could be more relevant for a manager and less important for a developer (and vice versa).

However, by applying the Laplace's criterion to this case, it has been possible to solve this issue, assuming that the criteria priority is equal to the probability of each state of nature. This leads to the following formulation:

- To each Cj in C, for j=1,2,…,n where Cj is each state of nature, in our case each criterion

$$pj = P(Cj) = 1/n \tag{4.3}$$

- To each Ai (payoff of each alternative, e.g. VisualStudio.com, Jira, FreedCamp), calculate its expected value

$$\mathrm{E(Ai)} = \Sigma \mathrm{j\ pj\ (Rij)} \tag{4.4}$$

  where Rij = 1 if the software Si has the criterion Cj, 0 otherwise.

- Because pj is a constant in Laplace

$$\mathrm{E\ (Ai)} = \Sigma \mathrm{j\ pj\ (Rij)} = \mathrm{pj}\ \Sigma \mathrm{j\ (Rij)} \tag{4.5}$$

- Choose the higher value of E(Ai) that is the optimal decision under uncertainty

So in this case we have S1 - Jira , S2 - VisualStudio.com and S3 – FreedCamp and as criteria:

- <u>Usability</u>
  C1: Intuitiveness to use
- <u>Team Collaboration</u>
  C2: Virtual workplace where to share data and project information between team members

C3: Instant communication tool in order to replace email

- Task Management

  C4: Virtual board where to see which tasks are To Do, In Progress or Done.

  C5: Easy possibility to assign resources, priorities and estimated time for each task.

- Document Management

  C6: Provide a repository for upload documents and attachments with the possibility to share files among all the users

- Monitoring & Visibility

  C7: View project progress and team performance with charts and graphs

- Cost Affordability

  C8: Annual cost

- Reliability

  C9: Security that protect data integrity and availability

- Suitability

  C10: Data integration to other system

The list of these criteria was possible only by make a practical trial of the 3 software with the colleagues of the team that will be the real end-users of the final software selected.
The problems found working with them have been divided into 8 areas, 5 of them had already been considered in the pre-selection software, but in the final selection it was necessary to make a more specific analysis focusing on more features -Usability, Team Collaboration, Task Management, Document Management, Monitoring, Cost Affordability, Reliability and Suitability- and for each of these a list of more detailed criteria has been drawn up, trying to summarise and group together the criteria that were most relevant, so as not to make the analysis too complex.

Table 4.10 - List of criteria with a briefly description and list of SWs

Criteria	Briefly Description	Software	
C1	Intuitiveness to use	S1	Jiira
C2	Virtual workplace where to share data and project information between team members		
C3	Instant communication tool in order to replace email		
C4	Virtual board where to see which tasks are To Do, In Progress or Done.		
C5	Easy possibility to assign resources, priorities and estimated time for each task.	S2	VisualStudio.com
C6	Provide a repository for upload documents and attachments with the possibility to share files among all the users		
C7	View project progress and team performance with charts and graphs		
C8	Annual cost		
C9	Security that protect data integrity and availability		
C10	Data integration to other system	S3	FreedCamp

As has already been said, almost all 10 criteria taken into account are binary criteria. It was therefore decided to assign a value equal to 1 if the software has that functionality, 0 if it does not.
It is necessary to make another clarification regarding criterion 8: the annual cost. In this case, in fact, the criterion can be defined as qualitative binary, as its expression is performed using qualifications (free/not free). In this case we assign a value equal to 1 if the software is free, 0 if you need licenses.

Table 4.11 – The decision matrix for choosing the right SW tool

Criteria / Software	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
S1	1	1	1	1	1	1	1	0	1	1
S2	1	1	1	1	1	1	1	1	1	1
S3	1	1	1	0	0	1	1	1	1	1

As we can see from table 4.11, the three software have almost all the requirements listed, being these alternatives the best also in the pre-selection.

Criteria C4 and C5 deserve a special consideration regarding S3 (FreedCamp). It has been assigned to these criteria a value equal to 0 which in theory would mean not having those features. There is a reason behind this valuation: clearly also FreedCamp has tools to create a virtual board and assign tasks to resources, but, compared to the other two alternatives, the graphics and usability of this function is strictly limited. Considering instead C8, we can notice that it has been assigned a value equal to 0 for the S1 (Jira) as it is the only one, among the three alternatives, to have an annual cost.

Following the Laplace methodology, all criteria (c1,c2,...c10) are accepted as equal (1/10=0.10), none of them have a priority.

The Laplace value for each software tool was then found by multiplying all 10 criteria by 0.10 and sum them up (the calculation was done on Excel). The result is visible in the following table:

Table 4.12 – Optimal Solution Result

Software	Sum of Cn	Laplace's Sum
S1	9	0,9
S2	10	1
S3	8	0,8

The software tool VisualStudio.com from Microsoft is therefore the best choice to implement the agile methodology in our company, even with one tiny delta compared to the other alternatives.

5. EMPIRICAL FINDINGS: VALUE CREATION THROUGH AGILE METHODOLOGY

The purpose of this final chapter is to highlight the empirical results found through the analysis of this work.

First, it is necessary to make a consideration regarding the final decision of the support tool. The result obtained by applying the Laplace criterion shows how the cost had in the end a fundamental relevance for the decision: the two best management software were practically identical in every aspect and it would have been quite difficult to make a decision if the company did not already have the license to own VisualStudio.com.

This seems to be the most appropriate choice considering also the fact that the company, having a partnership with Microsoft, is Microsoft-oriented: every tool used by employees is Microsoft, from the most common Office package, to Outlook, to Microsoft Team, thus ensuring a strong suitability between the selected software and other business tools as well as a greater intuitiveness of use by employees who are already close to the Microsoft world. The research of the most suitable digital tool carried out for PLM Systems provides an overview of the steps that should be considered when starting the digitization path of project management. Through the concrete use of the selected software, it has been demonstrated that digital project management solutions can provide substantial support in managing the areas and processes of knowledge of project management, but also in increasing organizational efficiency and proficiency in project management.

At this point it is necessary to make an evaluation of the benefits that have been obtained by passing from the use of a "traditional" methodology to an agile one supported both by the scrum framework and by the information tool appropriately chosen.

However, as we have seen in chapter 3, the case study analysed in this dissertation is limited to a small part of the entire project "Demo" developed in the company, so it is not possible to express final evaluations but it is still possible to give an overview of some significant results that the agile choice has produced, making a comparison between the original project carried out in the period prior to my arrival in the company and that of the case study.

The evaluation of benefits is not intended to be a synthetic measure that leads to a numerical result, but a multi-parameter evaluation that serves to demonstrate how efficient the work done has been for the creation of value in the company.

5.1. Evaluation of benefits

Taking into consideration the three Burn Down graphs (Graph 4.1, 4.2 and 4.3) which show the time spent by the team to complete the tasks included in the product backlog for each sprint, it is possible to note that the goal of completing the activities in 4 weeks (3 sprints, 2 of 1 week and the last one of 2 weeks) has been successfully completed.
The results obtained can be considered highly satisfactory, especially after having made some important considerations in this regard.
First of all, we have to take into account the fact that only a small part of the generalization (customization of the existing project application) has been analysed and put into practice. Making a generalization is always a more complex implementation activity, as it is necessary to stop and think about how to make the software adaptable to any case, and it is therefore an operation that would take longer than normal. Moreover, it is better considering the fact that the actual development time has been limited in comparison to the time spent on learning and consolidating the new agile methodology and the time spent on testing the three different software chosen with the preselection.
Despite all these considerations, comparing the phases developed during my training period with the same phases developed with the traditional methodology in the original project, a significant saving of time was obtained.
It is interesting to list several benefits that this saving of time has brought to the company in terms of added value.

Communication and Visibility between customer and team: increasing customer satisfaction

Communication between the development team and the client is of paramount importance, both in positive and negative cases: clients forgive mistakes, but if they are

not informed in time, much more serious problems can arise. As we have demonstrated with the case study, the Agile methodology and in particular the Scrum framework allows you to manage any new needs and priorities very quickly. It also provides better visibility (in the sense of monitoring) of the progress of the project. Working in close contact with the client, we were able to better and more quickly understand his needs, avoiding taking wrong paths and wasting time unnecessarily, as happened in the previous work implemented following a traditional methodology (Figure 5.1).

In fact, in the traditional model the visibility is high during the phase of gathering requirements, then you lose communication with the customer and you do not have the slightest idea to build the right thing, and then suddenly you send the product regaining maximum visibility. Unfortunately, however, it is often too late to correct something if things have gone in the wrong direction.

The effectiveness of the communication with the customer during all the iterations clearly leads to complete customer satisfaction, which in turn leads to a high degree of customer loyalty, a fundamental requirement nowadays for an SME IT consultancy to remain competitive in the market.

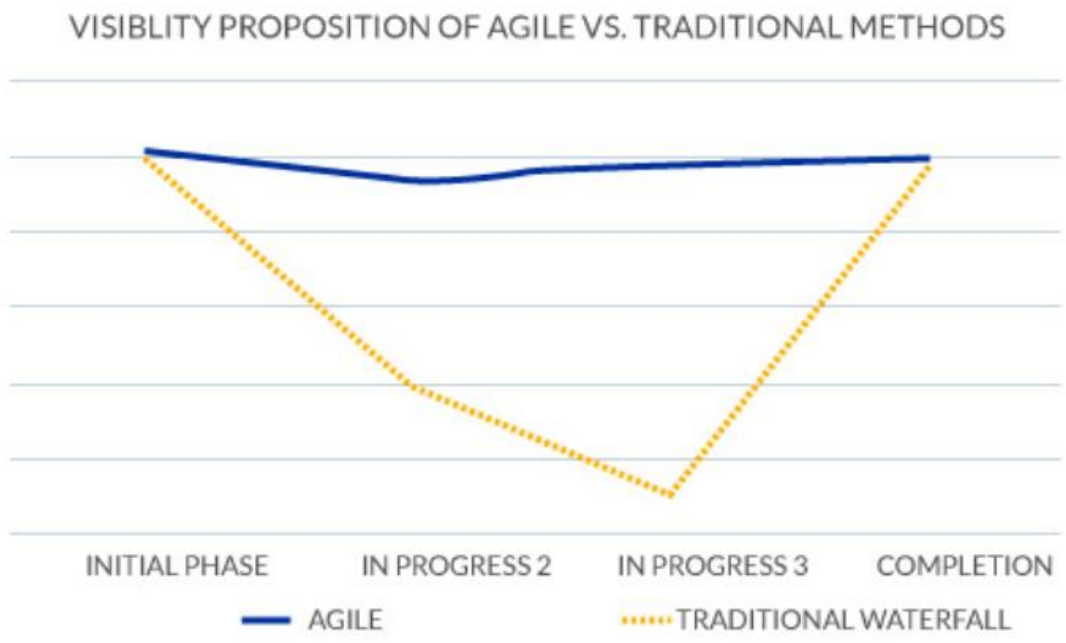


Figure 5.1 – Visibility proposition of Agile VS. Traditional Methods (source: www.knowledgehut.com/blog/agile/value-proposition-of-agile-development)

Adaptability and lower cost of change

The concept of adaptability is closely linked to what has just been said. It refers to the ability to manage the high degree of change and uncertainty in today's software development. At the beginning of each project the adaptability is maximum, since the project is started and the requirements are collected with the idea that any request or change can be easily adapted.

In the traditional cascade model, however, adaptability starts to decrease considerably from the design phase and becomes lower and lower during project delivery, because once certain tasks have been completed, there is no way to change them without starting all over again or without facing a considerable cost of change.

While with the Agile methodology, although adaptability is also maximum during the initial phase, it continues to fluctuate around the same value for the duration of the project, as each sprint is an opportunity to take something new or change direction depending on the business needs or the customer.

Also from Figure 5.2 below, you can see that the difference between traditional methodology and Agile is substantial, and has been tested practically during the development of our study project. In fact, while in the development of the " parent " project with the traditional model it took a long time, at the end of the development, to find and correct the defects of the code, since the tests were carried out only at the end of the implementation, in the development of our project, however, were discovered and corrected many more defects of the code during the development because the tests are an integral part of the implementation (they are present at the end of each story).

This also leads to lower costs.

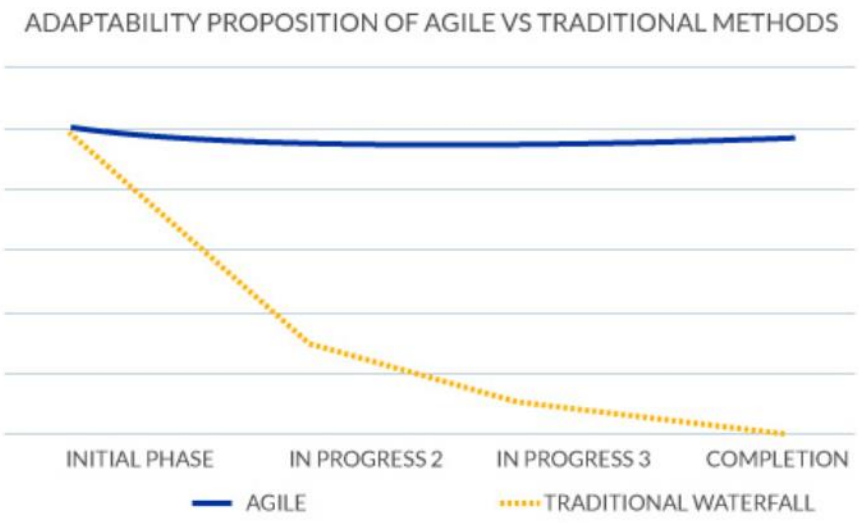


Figure 5.2 – Adaptability proposition of Agile VS. Traditional Methods (source: www.knowledgehut.com/blog/agile/value-proposition-of-agile-development)

Maximisation of business value and minimisation of risk

The concept of business value is closely related to how the project or product will help the company or satisfy the project owner. It is also very important when the product is delivered, because the company does not get any value until it provides something that can be used by end consumers.

In this sense, both Agile and Waterfall sound similar to us, but in reality, they are not because they vary in the way they provide business value.

Since the traditional method believes in delivery at the end or when the product is completely ready, we can see from Figure 5.3 that the business value starts from 0 and suddenly gets higher towards the end.

With Agile instead, we begin to provide incremental versions of the product from the first sprint; so the value of the business starts to be generated by the Sprint 1 and continues to rise until the last sprint or the moment when the owner of the product asks us to stop.

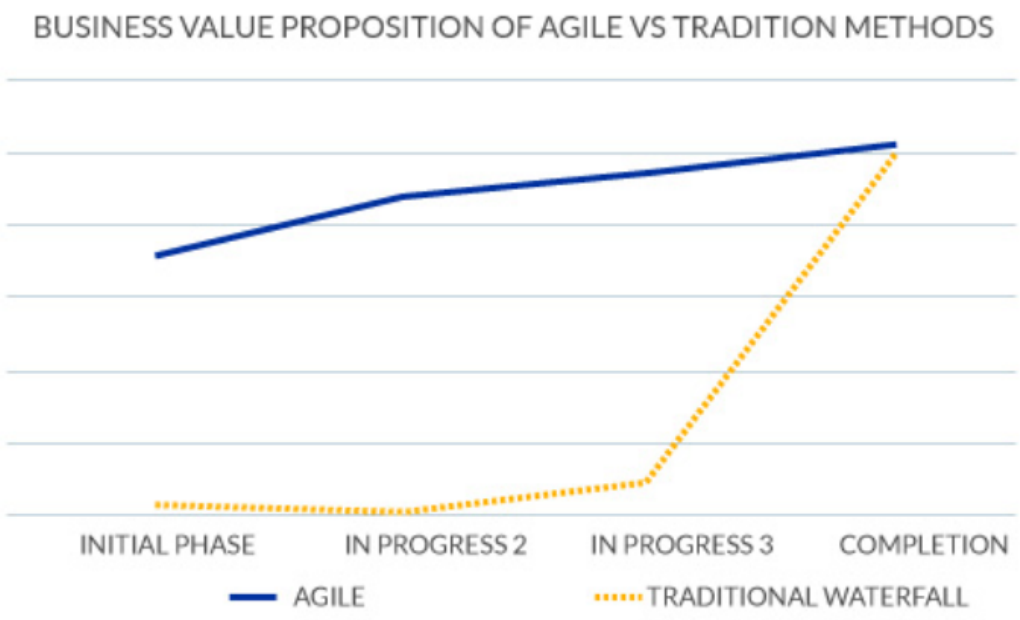


Figure 5.3 – Business Value proposition of Agile VS. Traditional Methods (source: www.knowledgehut.com/blog/agile/value-proposition-of-agile-development)

If we stop and analyse everything that has just been highlighted, we understand that high visibility and communication, high adaptability and frequent iterations on which the Agile methodology is based allow us to achieve a significant minimization of risks.

In fact, as we can see from Figure 5.4, if it is true that at the beginning both types of project start with a very high risk, because uncertainty in software development is an intrinsic feature, over time it is clear that, through the continue releases and controls by the customer in the Agile methodology, sprint after sprint, allow us to minimize the risk because the problems are detected in time and in time can being corrected.

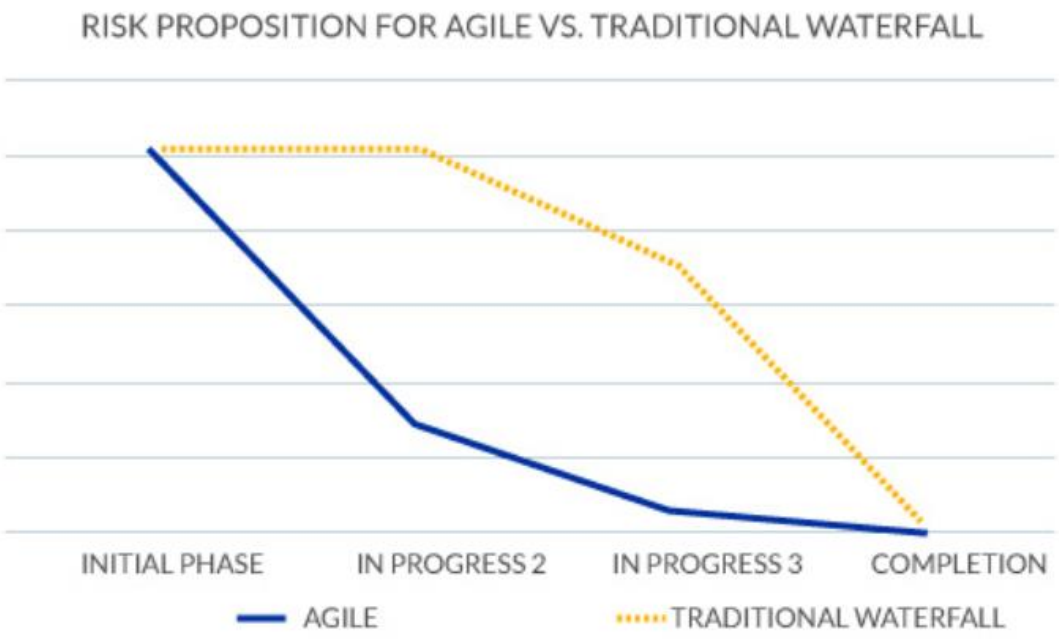


Figure 5.4 – Risk proposition of Agile VS. Traditional Methods (source: www.knowledgehut.com/blog/agile/value-proposition-of-agile-development)

In conclusion, we have demonstrated that, with an Agile methodology, thanks to its speed of delivery and short iterations, it is possible to anticipate the competition and get out on the market first acquiring business value, while reducing the risk of project failure and giving the customer the opportunity to benefit from the new software much faster than other methodologies.

6. CONCLUSION

Based on this study, can be drawn several theoretical and management conclusions that support the change in project management methodologies and the transition in the digitization of project management of small and medium enterprises in the software development sector. In fact, the analysis as-is of the company's issues has shown that the main problems described in the literature, such as the lack of knowledge of agile methodologies as well as the lack of use of IT support tools, have also been found in PLM Systems.

The study was then based on the possible way to solve these problems, adopting the framework scrum on a real project and performing a multi-criteria analysis on three main software previously selected with a pre-selection. It is important to specify that the case study analysed in this dissertation is limited to a small part of the entire Demo project developed in the company and it is better considering the fact that the actual development time has been limited in comparison to the time spent on learning and consolidating the new agile methodology and on testing the three different software chosen through the preselection.

That’s also why it was not possible to make an effective comparison between the phases of the project developed during my internship and those developed previously, or to express final numerical evaluation as a detailed quantitative analysis of the financial savings obtained, as I did not have access to the company's data on time and cost. But, despite all these considerations, it was proven in practice that, with an Agile methodology, thanks to its speed of delivery and short iterations, and with a right management tool, several objectives has been achieved.

From the multi-criteria approach, supported by the Laplace’s criterion for insufficient reasons, it was deduced that the best software for the company taken into account is VisualStudio.com, and, from the application of the methodology Scrum the results obtained show great savings in management time.

Management time is the most significant indirect cost in many organizations. A lot of time is spent guiding, planning and monitoring the work to be done. Much time is also spent trying to integrate new systems into work practices, as was done in this study, where there was not only the indoctrination of a new methodology, but also the use of three new

tools never used before in the company. Despite it was supposed to waste a lot of time in this management activities, the results show instead that there was still a saving of time, and therefore of cost, considerable.

For that reason, further research on this topic should be done to provide support to organizations, operating in various sectors, which are considering starting their path of change of project management. More theoretical approaches should be provided on the steps to be considered in this process, but also further research on digital project management solutions available on the market, and their practical use in project management, supported by the use of the most appropriate management methodology for the company under consideration.

This work could therefore be taken into consideration by other companies of similar size and with the same problems, in order to make an effective analysis of the cost of the investment to take the digitalizing project management and to move ahead with agile project management. Throughout the analysis, it can be determined the time each different project management functions takes. Comparison can be done with the current and future time estimates. These productivity increase estimates can be then compared to the license, maintenance, implementation and training costs of the digital solution. In this way the profitability of the investment that is required can be determined, with the computation of other key figures (e.g. return on investment, investment payback time etc.) for the senior-management of a company.

In conclusion, this work could be taken into account by other similar companies in the software development sector, so that they understand how profitable it is to make a change, and as soon as possible:

> *"The world is changing very fast. Big will not beat small anymore. It will be the fast beating the slow."*
>
> (Rupert Murdoch)

As Mr. Murdoch, CEO of Fox Industry said, speed and change are now essential key point in that century. The speed in changing the way of thinking and acting both at the management and business level. If you are not fast enough to react, to change, you risk being outclassed by those companies and organizations that can make the change more efficiently and quickly, inevitably losing competitive advantage in the market.

LIST OF FIGURES

LIST OF GRAPHS

LIST OF TABLES

BIBLIOGRAPHY

Abhinav Gupta, *"Value Proposition of Agile Development"*, January 2018 https://www.knowledgehut.com/blog/agile/value-proposition-of-agile-development

Acebes, Fernando, et al. "*A new approach for project control under uncertainty. Going back to the basics."* International Journal of Project Management 32.3 :423-434, (2014)

Ahlemann, Frederik. *"Towards a conceptual reference model for project management information systems."* International Journal of Project Management 27.1 :19-30, (2009).

Ahmad, Norita, and Phillip A. Laplante. *"Software project management tools: making a practical decision using AHP."* Software Engineering Workshop, 2006. SEW'06. 30th Annual IEEE/NASA. IEEE, (2006).

Aldea, Cosmina Carmen, and Cristian Olariu. *"Selecting the optimal software solution under conditions of uncertainty."* Procedia-Social and behavioral sciences 109 333-337, (2014).

Alshawi, Mustafa, and Bingunath Ingirige. *"Web-enabled project management: an emerging paradigm in construction."* Automation in construction 12.4: 349-364, (2003).

Archibald, Russell D. *"Managing high-technology programs and projects"*. John Wiley & Sons, (2003).

Artto K., Martinsuo M., Kujala J., *Project business*, Helsinki, (2011).

B. Boehm. *Get ready for agile methods, with care. Computer*, 35(1):64–69, (2002).

Bandor, Michael S., *"Quantitative methods for software selection and evaluation."* No. CMU/SEI-2006-TN-026. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, (2006).

Beck, Kent, et al. *"Manifesto for agile software development."* (2001): 2006.

Beck, Kent, and Erich Gamma. *"Extreme programming explained: embrace change."* addison-wesley professional, (2000).

Bigatti Massimiliano, *"Extreme Programming e Metodologie Agili di Sviluppo Software: concetti, prodotti e risorse"*. (2002)

Braglia, Marcello, and Marco Frosolini. "*An integrated approach to implement project management information systems within the extended enterprise."* International Journal of Project Management 32.1: 18-29, (2014).

Caniëls, Marjolein CJ, and Ralph JJM Bakens. *"The effects of Project Management Information Systems on decision making in a multi project environment."* International Journal of Project Management 30.2: 162-175, (2012).

Cantamessa, Marco, and Francesca Montagna. *"Management of innovation and product development"*. London: Springer, (2016).

Casoni Mirco, "*Scrum: un processo agile*", September 2010, http://www.mokabyte.it/2010/09/scrum-1/.

Cohn, Mike. "*User stories applied: For agile software development."* Addison-Wesley Professional, (2004).

Coram, Michael, and Shawn Bohner. *"The impact of agile methods on software project management."* Engineering of Computer-Based Systems, 2005. ECBS'05. 12th IEEE International Conference and Workshops on the. IEEE, (2005)

De Marco, "*Project Management Theory Book"* URL: http://hdl.handle.net/11583/1659508, (2007).

E. Arisholm, H. Gallis, T. Dybå, and D.I.K. Sjøberg. *"Evaluating pair programming with respect to system complexity and programmer expertise. IEEE* Transactions on Software Engineering", 33(2):65–86, (2007).

Eldrandaly, Khalid. *"GIS software selection: a multicriteria decision making approach."* Applied GIS 3.5: 1-17, (2007).

Górecka, Dorota. *"Multi-Criteria Decision Aiding in Project Management-Outranking Approach and Verbal Decision Analysis."* Studia Ekonomiczne 137: 11-38, (2013).

Guide, A. "*Project Management Body of Knowledge (PMBOK® GUIDE)*." Project Management Institute. (2001).

Harter Jim, "*Why managers are central to an agile culture."* Gallup, October 2018, https://www.gallup.com/workplace/243455/why-managers-central-agile-culture.aspx

Hazır, Öncü*. "A review of analytical models, approaches and decision support tools in project monitoring and control."* International Journal of Project Management 33.4: 808-815, (2015).

Hu, Zhi-gen, Quan Yuan, and Xi Zhang. *"Research on agile project management with scrum method."* Services Science, Management and Engineering, 2009. SSME'09. IITA International Conference on. IEEE, (2009).

J. McAvoy and T. Butler. "*The role of project management in ineffective decision making within agile software development projects*." European Journal of Information Systems, 18(4):372–383, (2009).

Kaygısız-Ertuğ, Z., & Girginer, N. *"A multi criteria approach for statistical software selection in education."* Hacettepe Üniversitesi Eğitim Fakültesi Dergisi [Hacettepe University Journal of Education], 29(2), 129-143, (2014).

Kasra Madadipouya "*An Examination and Evaluation of Agile Methodologies for Systems Development*." Australasian Journal of Computer Science, 2: 1-17, , (2015).

Kerzner, Harold, and Harold R. Kerzner. *"Project management: a systems approach to planning, scheduling, and controlling"*. John Wiley & Sons, (2017).

Kittlaus, Hans-Bernd, and Peter N. Clough. *"Software product management and pricing: Key success factors for software organizations."* Springer Science & Business Media, (2008).

Love, Peter ED, et al. *"The enigma of evaluation: benefits, costs and risks of IT in Australian small–medium-sized enterprises."* Information & Management 42.7: 947-964 (2005).

Mark, A. "*PMI's Pulse of the Profession: The High Cost of Low Performance. A Core Competency for Project and Program Success."* Project Management Institute, (2014).

Pressman, Roger S., and Elena Addomine. *"Principi di ingegneria del software."* McGraw-Hill, (1991).

Raymond, Louis, and François Bergeron. *"Project management information systems: An empirical study of their impact on project managers and project success."* International Journal of Project Management 26.2: 213-220, (2008).

S. Ambler. "*Agile software development at scale. Balancing Agility and Formalism in Software Engineering",* pages 1–12, (2008).

Saaty, T.L. "*Fundamentals of the Analytic Hierarchy Process*," RWS Publications, Pittsburgh, PA, (2000).

Sakka, Ouafa, Henri Barki, and Louise Côté. *"Relationship between the interactive use of control systems and the project performance: The moderating effect of uncertainty and equivocality."* International Journal of Project Management 34.3: 508-522, (2016).

Serrador, Pedro, and Jeffrey K. Pinto. *"Does Agile work? A quantitative analysis of agile project success."* International Journal of Project Management 33.5 :1040-1051, (2015).

Shim, Jung P., et al. *"Past, present, and future of decision support technology."* Decision support systems 33.2: 111-126, (2002).

Sutherland, Jeff, et al. *"The scrum papers: Nuts, bolts, and origins of an agile process."* (2007).

Tarantilis, Christos D., Chris T. Kiranoudis, and N. D. Theodorakopoulos. *"A Web-based ERP system for business services and supply chain management: Application to real-world process scheduling."* European Journal of Operational Research 187.3: 1310-1326, (2008).

ACKNOWLEDGEMENTS

Everyone says that the moment of writing the Master Thesis is one of the most important challenges of life, because it puts you in front of the brink of desperation. The same happened to me, and I'm sure that without the encouragement and support of some people now I would not be here to write these acknowledgements.

First, with a little sadness, I would like to thank the IoTeam, with whom I worked to draw up this paperwork and who has allowed me to live these last few months with more lightness and serenity. A special thank you to Shirin, who so many times stayed with me in the office until 8 pm to help me define the thesis, and to Gian Piero, who read and corrected my entire work, helping me to achieve this result. To all of you, however, I say a big thank you, it was a pleasure to meet you.

A very special thank goes to my colleague and now a consolidated friend, Maria Concetta. It is an honour to be able to celebrate this day with you by my side, our university adventure began together, and it could only end in the same way. The help you have given me in these years is not quantifiable, and you know very well that without you I would not have been here now. You are a special person, and I am very proud to have this strong link with you, which I am sure will be forever. Thank you very much, for everything.

It is also necessary to thank you, Betta. You are my second mother in the true sense of the word, in every moment of difficulty I wouldn't know who to call but you, and your advice is always the right one to follow. Thank you because with your wisdom, you also make me a little wiser.

A big thank you to all my friends who have lightened the heavy days that seemed endless and have made sure that these university years remain forever indelible in my mind.

Thanks to you, mom, because with your enthusiasm you can always make my worries lighter.

Finally, a big thanks to my family, without whom I could not have been here. It is only thanks to you that I am who I am, and I will forever be grateful.
And then, to keep a bit of the egocentricity that has always distinguished me, I wanted to make a small thank you to myself.
For the kilometres I have travelled in pursuit of a future that was uncertain but full of expectations. For the strength I had to cross the sea and build a new life, alone. For the composure and rigidity of my soul, for the teachings and thoughts defended with pride and fierceness.

Because I thought I couldn't do it, and instead...