Master's degree course in Management and Industrial Engineering

How to ensure a quality uniformity to a set of web platforms

Supervisors

Prof. Fabio Salassa

Prof. Maria Franca Norese

Prof. Michel Tollenaere

Candidate

Samuele Vannoli

Co-supervisor

Mathieu Coquelet - Engineering Project Manager

Index

1. Introduction

This work is mainly the result of the experience gained during the curricular internship that I carried out within Agorize. The internship took place in Paris at the Agorize headquarters from September 2020 to March 2021 and the activities that I carried out were in support of the Engineering Project Manager (EPM).

Agorize is a scale-up specialized in the organization of innovative challenges. A challenge is a competition that encourages participants to propose their most relevant and innovative solutions for a business, technical or a social problem.

The subject of the Case Study is a study of the current situation for the processes of creation of Landing Pages and challenge display and how it is possible to improve the current situation. The aim of the project is therefore to improve the internal process of creating the challenge displays and the Landing Pages. This goal was achieved through the creation of a new web platform.

It has been decided to create a tool that would reduce and simplify the work of EPMs, trying to centralize HTML, CSS, and JavaScript knowledge to the whole organization. This tool has been called *library* and it is a web platform, for now only for internal purposes, and consists of a platform with all the HTML, CSS, and JavaScript structures used in the past by Agorize.

For the development of the project, it has been decided to adopt an Agile logic, breaking down the process into short and iterative cycles, using the Kanban framework within the Scrum methodology. The agile approach offers to maximize flexibility during the project, trying to "crystallize" the new product at the last possible moment, to react promptly to a change in customer requests or, more generally, the market. What we are experiencing is an era characterized by continuous changes, both from the point of view of the market and of technological innovations and the skills necessary to integrate and manage them better. Being

able to integrate new agile methods for project management is essential to manage continuous change.

In Chapter 1 I tried to give an overview of the company where I have been working for 6 months, going into the details of the structure and organization.

Chapter 2 is dedicated to the study of the literature on Agile Project Management, its evolution over the years, and all the methodologies that are part of it.

The improvement actions induced a careful analysis of the problems that characterize the company. They have been detailed in Chapter 3, where it is shown how the methodology has been applied in the specific case of Agorize.

2. Agorize

Agorize is a scale-up specialized in the organization of innovative O2O challenges (online to offline). The company was founded in 2011 in Paris and now has offices in Paris, in Stuttgart (Germany), Montreal (Canada), Hong Kong and, Singapore, with more than 200 multinational companies as clients. From its foundation to today, Agorize has received many awards and recognitions, among which the one of being included among the 500 most technologically dynamic companies in the world by Deloitte.

The team includes more than 80 people who all share the same purpose: to build bridges between large companies and innovators (students, startups and, IT developers) by organizing a competition (challenge).

A challenge is a competition that encourages participants to propose their most relevant and innovative solutions for a business or a social problem. Since 2011, approximately 1000 challenges have been organized, including a community of 5 million innovators.

Usually, the challenge is organized by companies or universities addressed to students or startups to find innovative solutions for a given theme. Nowadays the challenges have become mostly digital and in fact, they have gone from being 100% physical events to hybrid competitions that start online and end with a big final event in attendance. This process is known as O2O (online to offline).

The theme the challenge is based on is a crucial aspect because the more interesting and attractive it is for the target audience you want to involve, the higher the participation rate will be. Figure 2.1 shows which are the main topics covered in the challenges, analyzing 318 organized by Agorize from 2011 to 2016.

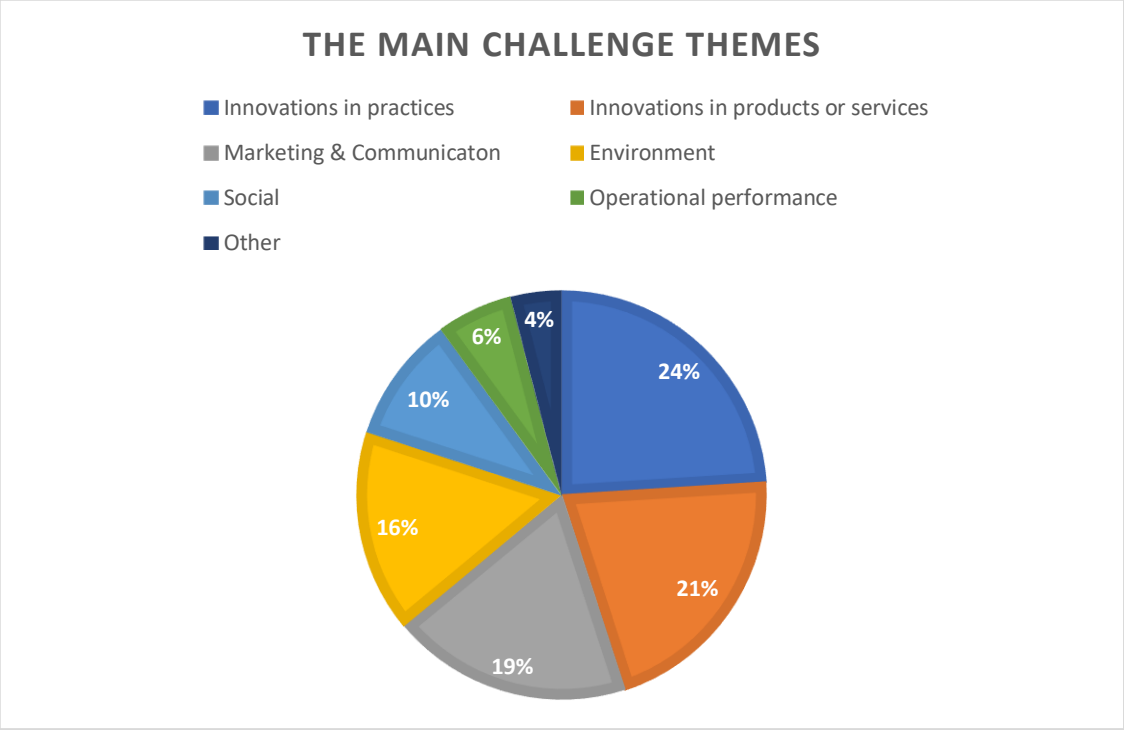

Figure 2.1: The main themes of the challenges organized by Agorize.

The most common issues that are addressed in the challenges concern innovation in business practices (24%). An example of a competition that falls into this category of the theme is given by the challenge organized by Galeries Lafayette to redesign the customer's experiences inside and outside the store. The second category of most common themes is that of product and service innovation (21%). For example, the SEB company organized a competition to imagine the kitchen of the future. The category with topics related to Marketing and communication follows (19%) and an example is given by the challenge organized by Philips, whose goal was to organize a marketing campaign to launch a new product called Philips room speaker system. Then there are issues related to the environment (16%) and an example was a competition whose purpose was to find innovative and eco-friendly solutions to remove vegetation from train railways organized by SNCF. Finally, there are issues related to Social Media (10%) and those to the improvement of operational performance (6%). An example of a social theme is given by the competition organized by Fondation Cognacq-Jay to find a way to

promote solidarity through social media, while an example of the improvement of operational performance was the challenge whose goal was to imagine an operator of the future to improve the performance of corporate groups organized by Total.
There are several reasons why a company decides to launch a challenge: the main advantage for the organizing company is to innovate faster and easier, gathering the best ideas that will be proposed by the many participants, leaving the company free to do not hire an additional large number of employees and not burdening the R&D department with too much work. This type of mechanism is the basis of the concept of open innovation. Another advantage is that of being able to easily identify and hire the greatest talents on the market by testing the technical and interpersonal skills of the candidates during the stages of the challenge.
Participants also have good reasons to sign up for these challenges as students and startups have the opportunity to test their creativity in solving a real problem and go beyond passive and traditional training methods thanks to a learn-by-doing approach. Furthermore, participants can get noticed by large companies more easily and students can have a chance to be hired while startups, on the other hand, raise some useful funds for their growth.

2.1 The origin of open innovation

Modern companies need agility and innovation and even if innovation is very often the domain of the Research and Development department of a company, changes are starting to occur. Over the years the most important companies (led by Google, Microsoft, and Bouygues) have understood the importance of adopting an approach that includes more people and different backgrounds. In addition to their employees, efforts are increasingly being made to include people outside the company such as students and startups to help the company innovate. This is open innovation and although some are still skeptical thinking that it is only a

momentary trend, in reality, it has existed for a long time. By appealing to collective intelligence and creating meeting places, our ancestors laid the foundations for open innovation. Without knowing it, they opened the doors to a new model of innovation, currently booming thanks to the Internet and new technologies. The 18th century was a time of great discoveries. Christopher Columbus had already discovered America, but there were still many problems. The methods for calculating longitude at sea were very rudimentary, despite the frantic search for the greatest minds in the world. The problem has resulted in many fatal accidents, such as the Scilly naval disaster. Admiral Cloudesley Shovell, convinced that he was sailing on the high seas, was indeed dangerously close to the coast and his fleet eventually crashed into the rocks, killing over 2,000 men. Determined to put an end to these tragedies and faced with a lack of concrete solutions, the UK launched what was a revolutionary initiative for the time: the Longitude Act. The law promised a reward of £ 20,000 to anyone who could develop a reliable method for calculating longitude with an accuracy of 0.5 degrees. No one has managed to achieve such a degree of accuracy, but John Harrison, a watchmaker, has achieved the most conclusive results. He spent 31 years researching in collaboration with the British government, eventually achieving the necessary accuracy and creating the first marine chronometer, which is still used today. This is how the concept of open innovation was born. The initiative led to the experimentation of totally innovative new approaches and the discovery of talents. This was probably the first real open innovation challenge ever organized, demonstrating the effectiveness of collective intelligence in the search for concrete solutions (Chesbrough, 2003).

World exhibitions have also played a role in spreading the concept of open innovation. These international events brought together the greatest innovators of their time. They have encouraged technological emergencies and stimulated business development. The very first world exhibition was organized in London in 1851 to bring together interested parties from around the world to show the

public the best of innovation. The event was held as a competition and the best inventions won medals which gave the winning companies credibility and visibility. As perfect meeting spaces, these events facilitated collaboration between exhibitors and promoted technical progress. World trade fairs have acted as incredible accelerators for the company's development. The ice machine exhibited by Raoul Pictet during the 1878 World Exhibition in Paris is the perfect illustration. Following this event, the Compagnie des Glacières de la Seine ice company asked to use the machine's revolutionary process. Today the company is known as Picard and is the leading frozen food supermarket in France. From then on, it became easier to create collaborative partnerships and innovate quickly and in a less conventional way.

Faced with this phenomenon, which is far from new, it is important that the mentality evolves and that companies attract even more communities of innovators (students, developers, startups, and employees) to work together on creative processes. Collective intelligence is a powerful engine of innovation and is a surefire way to form partnerships that will contribute to a company's success and future growth.

2.2 The phases of an Agorize challenge

When launching a competition, it is very important to be as clear as possible, making the rules, prizes, and timing of the challenge transparent.

A challenge, in general, is characterized by 3 phases as illustrated in figure 2.2. The first is preceded by the preparation of the competition in which the topic, objectives, rules, and timing are defined.

The first phase consists of registering the participants in teams or individually and in a certain period of time, decided in the preparation of the competition, a short document must be handed in which briefly presents the idea to solve the problem

launched by the company. After the presentation time, all the candidate's ideas are evaluated and voted by a jury defined in the preparation of the competition. Both internal juries are generally involved, such as the employees of the organizing company, and external juries such as Agorize's business partners or external consultants. The aspects that are evaluated are originality, feasibility, efficiency, budget, sustainability, and quality of the report. Usually, the best 30 proposals pass into the next phase, which consists of developing the idea in more detail and delivering a specific business plan. Also, in this case, at the expiry of the time, there is a vote by the jury that selects from 3 to 10 finalists of the challenge. In the third phase, a mentor review is carried out, in which the few remaining participants are helped by the final proposal. In the final round, there will be a final presentation, usually in person, and the announcement of the winners.

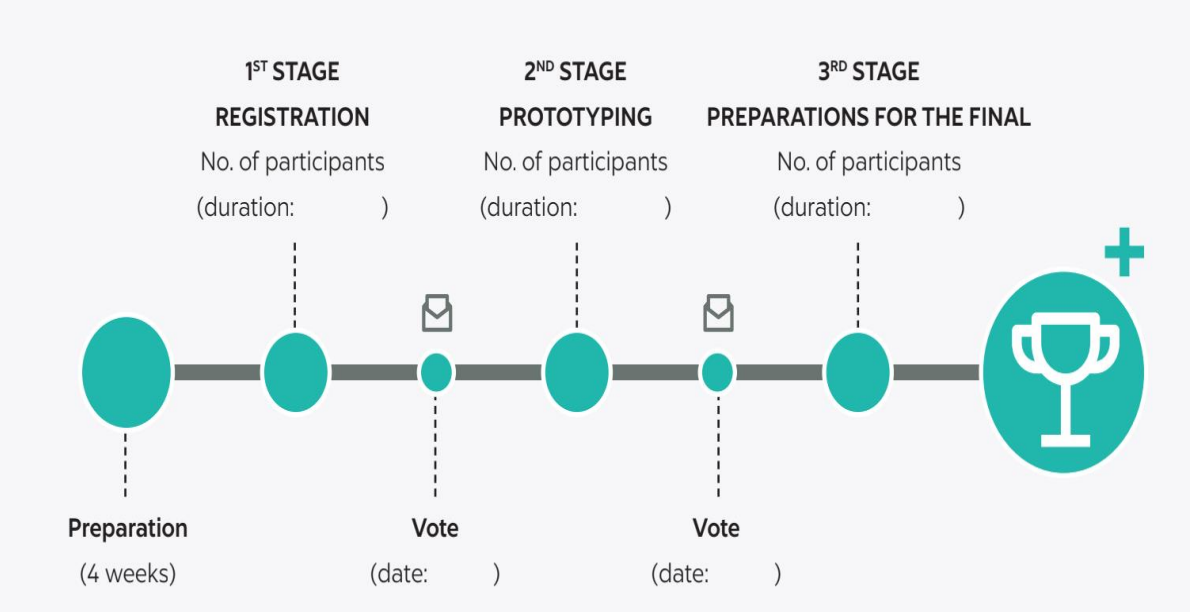

Figure 2.2: The various phases of a challenge.

It is also very important to define the prizes and benefits for the winners as the competition requires a high investment of time. Among the most attractive benefits for participants is the fact that they can add a very important experience to the CV,

get more visibility, and receive mentoring from professionals. Besides, material goods such as TVs or laptops are often added as well.

2.3 Agorize team

The company is made up of 6 departments (teams): R&D (Research and Development), Project Management, Community, Design, HR & Admin, and Sales & Marketing.

The R&D department is composed of three main figures: IT developers, Technical Project Managers or also called Engineering Project Managers (EPMs) and Product Owner/Manager. Developers have mainly been involved in programming the back end of the web platform on which the challenges are launched to add new features. The EPMs are instead involved in various activities concerning the development of the challenge, such as the management of the URL in which the challenge will be published, but also in the general improvement of business processes and the technical characteristics of the platforms by staying in contact with IT. Furthermore, during the production phase, the EPMs assist the Project Managers in operational activities and maintenance, such as the creation of the display (what the participant will see) of the challenge. EPMs are responsible of the quality of the platforms. They configure them, monitor them and maintain them. Their key role is also mostly a role of coordination: they make sure that Product Owners and Software Developers understand each other (internal coordination) and act as a bridge between the R&D and other departments, between R&D and clients. They are also responsible of implementing, documenting R&D internal processes and make sure they are followed. Finally, they are responsible of collecting OKRs (Objectives and key results) of each team in order to take appropriate measures if needed.

The role of the Project Managers (PMs) team consists not only in managing relations with the customer but also in understanding their requests and assistance in the various stages of the challenge, also in the construction of the various pages of the challenge, adding the related graphic elements and text to specify the challenge rules, theme or prizes for the winners.
The Community team has the task of establishing a long-term relationship with customers to increase loyalty, but also to keep the interest in the ongoing competitions high.
The Design Team takes care of the creation of banners, images, icons, or videos to eventually add to the challenge.
The Human Resources department, on the other hand, takes care of hiring new talents for the company and the Sales & Marketing Team defines strategies for acquiring new customers and decides the prices to propose, based on the service that will be chosen.

2.4 The services offered by Agorize

Agorize offers 3 different solutions for creating a challenge. The first solution is the more classic one, called white label, and that is to create a challenge for a customer, who will publish it on their website. This type of solution is required by large companies established on the market that do not need sponsorships as they manage to have great visibility even without the help of Agorize. In this case, the EPMs provide the publication of the challenge in the URL chosen by the customer.
The second solution consists of organizing a competition and publishing it on the Agorize website. This type of service allows the rebranding of the challenge making it appear as if it was organized by Agorize. Generally, the white label solution is requested by small and medium-sized companies or universities that do not have much visibility on the web and ask to insert the challenge on the Agorize website to acquire more participants and make the competition more attractive and

stimulating. Furthermore, the challenge published on the Agorize website benefits from the company's Community team which, as described above, is responsible for recruiting an increasing number of students or startups and always keeping the interest in the challenge high.

The third and final solution is Software as a Service (acronym SaaS) which is a set of means, services, and skills that allows companies to outsource certain aspects of their information system, as shown in figure 2.3, and replace them with an operating cost. rather than with an actual investment. Specifically, Agorize grants the web platform for the development of a challenge to a customer, who receives a short training on how to use it, and subsequently, the realization, sponsorship, and management are completely entrusted to him.

Figure 2.3: Representation of the Saas service.

Saas is a cloud solution that offers many benefits for both users and companies such as:

- A quick start-up and immediate provision of services to customers: instead of finding an agreement between Agorize's PMs and the customer on how

to organize the different aspects of the challenge, such as the display, the platform for the realization of the challenge is completely released to the customer who, after a short training, manages every single aspect.

- Cloud architecture offers all users the same security standards: by storing all data remotely on a cloud server, data loss due to local hardware and software problems is minimized. Smaller companies enjoy the same safety standards as larger companies.
- New functions and updates are implemented and integrated much faster: since the software is executed on the Agorize side, the latter also takes responsibility for its management. This way the user can be sure that he is always using the latest version of the services.
- This is a much cheaper solution for the customer who only has to pay a cost of use since the Agorize team is not involved in the creation of the challenge. This allows even small businesses, which cannot pay out large amounts of money, to organize challenges and therefore to innovate.

3. Agile Project Management

A project is defined as a temporary effort undertaken to create a product or service. UNI ISO 21500 states that the project is a unique set of processes, consisting of coordinated and controlled activities with start and end dates, which are carried out to achieve the objectives of the process.

Project Management refers to a series of activities carried out to analyze, design, and implementing a project, managing it in all its characteristics during each of its evolutionary phases. Project Management is defined as the application of knowledge, attitudes, tools, and techniques to the activities of a project to achieve its objectives (Tereso et al., 2018).

The achievement of the project objectives and the allocation of resources are the main challenges to be faced, always respecting the three constraints of time, cost, and purpose.

The management of a project is usually entrusted to a Project Manager who mainly focuses on the coordination and control of the various components and the various actors involved to minimize the probability of failure. Its role is to manage and control the resources available so that a certain activity is carried out within the constraints of the project, that is, in compliance with times, costs, and performance (Kerzner, 2014). Upon completion of the project, usually, the product or service created is taken over by a different operating figure, typically the product manager or service manager.

In large-scale projects, the project management activity can be delegated to several people: a project management group is therefore created. Commonly in the group, there is a leader who is in any case called project manager, alongside this other person who deals with the management activities of parts of the project according to a view by system components or specific areas. These are sometimes referred to as task managers.

The culture of Project Management has remote origins, in fact, the first examples of projects in which there is a hint of this culture are those relating to the construction of the Egyptian Pyramids, the Colosseum, and the Roman aqueducts. In modern times, the first applications of Project Management occur in the construction sector, only later in the industrial and software development fields.

One of the most important contributions was made by Henry Gantt, an American engineer, who at the beginning of the twentieth century introduced a planning technique, called by his name the Gantt chart and still fundamental in planning activities. Every single activity is identified by a bar of variable length according to the duration and is related to a time scale that I define as the beginning, end, and duration.

In the 1950s, methodologies were introduced that helped improve project management, such as the CPM (Critical Path Method) and PERT (Program Evaluation and Review Technique) techniques. The CPM technique makes it possible to identify which are the priority activities and which must be carried out as soon as possible and those in which a shift is possible, without making the project longer (Gaio, 2010).

While PERT is a project evaluation and review technique and a network-based aid for planning and scheduling the many related activities in a large and complex project. It was developed during the design and construction of the Polaris submarine in the United States in the 1950s, one of the most complex tasks ever attempted at the time. Nowadays PERT techniques are routinely used in any large project such as software development or building construction. The goal is to reduce times and costs with the Montecarlo probabilistic method which can anticipate the degree of criticality that operations can assume (Chinneck J., 2016).

Subsequently, in 1969 the Project Management Institute was founded which had as its objectives to define and disseminate the principles of Project Management through seminars, publications, periodicals, and a guide called PMBOK (Project Management Body of Knowledge) The purpose of the PMBOK is to standardize

and document commonly accepted project management practices applicable to different types of projects: construction, software development, automated processes, and industrial processes. It is a guide containing all the standards and dictates to follow.

However, in the last 20 years, with the evolution of the IT sector and its areas of application, the need arose for a Project Management methodology that would allow us to respond quickly to the rapid and frequent changes that characterize this sector.

Thus, was born in 2001 the Agile approach, which offered an alternative to the traditional model. Agile Project Management arises from the need to evolve and adapt to sudden changes given by the dynamism of the current market context. In recent decades, traditional Project Management methodologies have often been ineffective in guaranteeing the realization of a project in terms of cost, time, and quality. In an increasingly dynamic market, the new methodologies based on an Agile approach have as their common goal to increase the productivity and efficiency of the workgroup, through a change in the organizational structure and especially in the corporate culture.

Between 1970 and 1980, the manufacturing sector encountered numerous difficulties in managing production operations through the use of traditional management systems; more or less satisfactory levels of success were achieved through the use of formal planning of production and materials, of programming and control of the production line, and of systems such as the "Planning of material requirements" (Manufacturing Resource Planning - MRP) and "Enterprise Resource Planning - ERP).

The trend to move from linear and more conventional development approaches to non-linear and iterative ones in software development and testing is well established nowadays and is called the Agile approach. Agile project management is rather defined as an extension of traditional project management models (Leybourne, 2009).

3.1 The Agile Manifesto

In 2001, a group of software designers got together to create a guide that would allow people to approach the Agile approach and understand its main features, without however going into detail on the type of Agile methodology to use. The result of this work was the Agile Manifesto, in which the 4 values and 12 principles on which the Agile approach is based are proposed (Hazzan and Dubinsky, 2014). The 4 values are:

1. Individuals and interaction rather than processes and tools: It is specified how the team must adapt the processes for each specific project and must not follow a standard plan. Since the projects are unique and extremely variable between them, the team is obliged to modify the processes and when necessary also to deviate from the plans.
2. Software operating primarily over comprehensive documentation: the second principle refers to the software industry, but the concept can be extended to any type of project. With this principle, we want to specify how important it is to divide the project into short cycles rather than setting a single goal to achieve the final result.
3. Collaboration with the client rather than the negotiation of the contract: the Agile methodology also places the client at the center of development, becoming an integral part of the project. At the end of each process cycle, the customer must provide essential feedback to decide what should be changed and what can continue to be developed.
4. Respond to change rather than following a pre-established plan: the Agile approach requires that each initial plan is subject to continuous changes during the course of the project and therefore change must be expected during the process.

Figure 3.1 summarizes the 12 principles on which the Agile method is based: attention to satisfy all customer requirements, readiness for change, delivery at regular intervals to the customer, collaboration with the customer and among team members, continually improve, and have constant attention to achieving excellence.

Figure 3.1: The 12 principles of Agile Project Management.

3.1.1 Comparison between Agile and traditional methodology

Traditional project management is based on the waterfall model for software development, a methodology that involves the following phases: specification of requirements, design, implementation, verification, and maintenance. The traditional cascade model in software engineering was formally presented as an idea, through a paper published by Winston Royce in 1970. This methodology has been successfully implemented by many software companies, despite having numerous limitations. On the other hand, the agile model of software development evolved in the 1980s and 1990s when developers decided to break away from the traditional structured, segmented, bureaucratic approach to software development

and shifted towards more flexible development styles. The "Agile" or "Light" methods as they were called, were formally defined in a research paper by Edmonds in 1974, but it was in 1988 that the real revolution in this sector took place with the transition to the spiral model for development. software. The spiral model of the software process has evolved over several years, based on the experience with various refinements of the waterfall model applied to large government software projects (Boehm, 1988). In 2001, a group of pioneers in agile software development came together and declared the "Agile Manifesto", which is a canonical set of rules for agile software development methods.

The waterfall model, as its name indicates, is a sequential process of developing a product or service. Just like in a waterfall, the water progressively descends from one altitude to the lower one, in a similar way, and in this way, the production cycle proceeds sequentially, from one phase to another, as shown in figure 3.2. The figure also shows the development phases that are adopted in this model: specification and analysis of requirements (Requirements), design (System Design), implementation and installation (System Implementation), and, finally, testing and maintenance (System Operation and Support). In this sequentially structured approach, the development team moves to the next stage of development, only after the previous stage has been fully completed. The belief that drives this type of software development model is the considerable time spent in the initial design effort, fixing all bugs in advance. Once the design phase is finished, it is implemented without changes thereafter. Often the analysis, design, and programming teams are separated and work on small parts throughout the entire development process. Special emphasis is placed on documentation at each stage of software development.

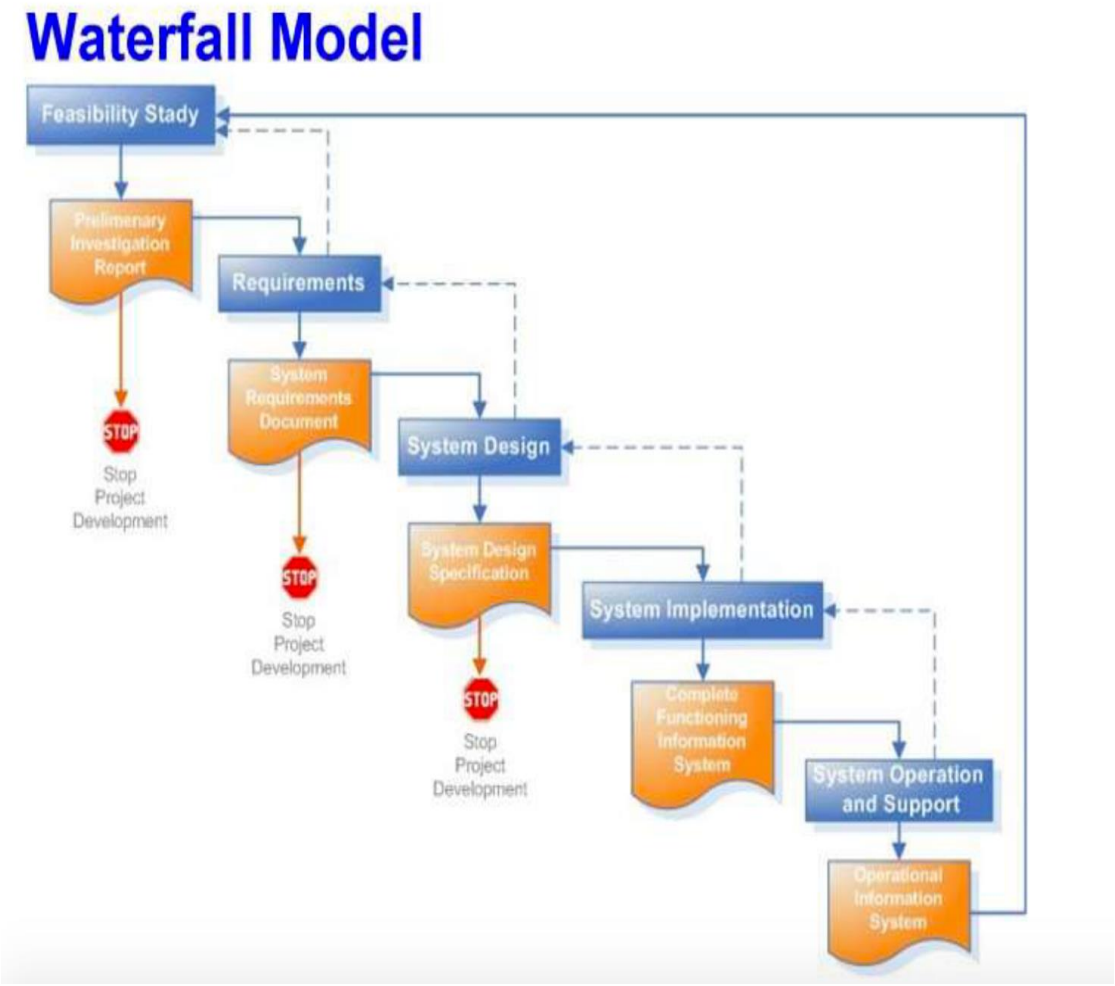

Figure 3.2: Representation of the waterfall model in the traditional methodology.

In the Agile methodology compared to the cascade development approach, we focus on agility and "adaptability" in development. Instead of a long and rigid development schedule, agile models involve multiple iterative development programs that try to improve output with each iteration. As shown in figure 3.3, each iteration passes through all the phases of design, programming, and testing. The design is not decided a priori and is kept open to last-minute changes due to the iterative method of implementation. Less importance is given to documentation, which risks being continually modified, and more to speed in providing a developed work program to be tested. Customers can be provided with demonstrations at the end of each iteration, and their feedback can determine the

next course of changes in the next iteration. The iterative cycle continues until the customer is delivered a documented product that exactly matches their expectations.

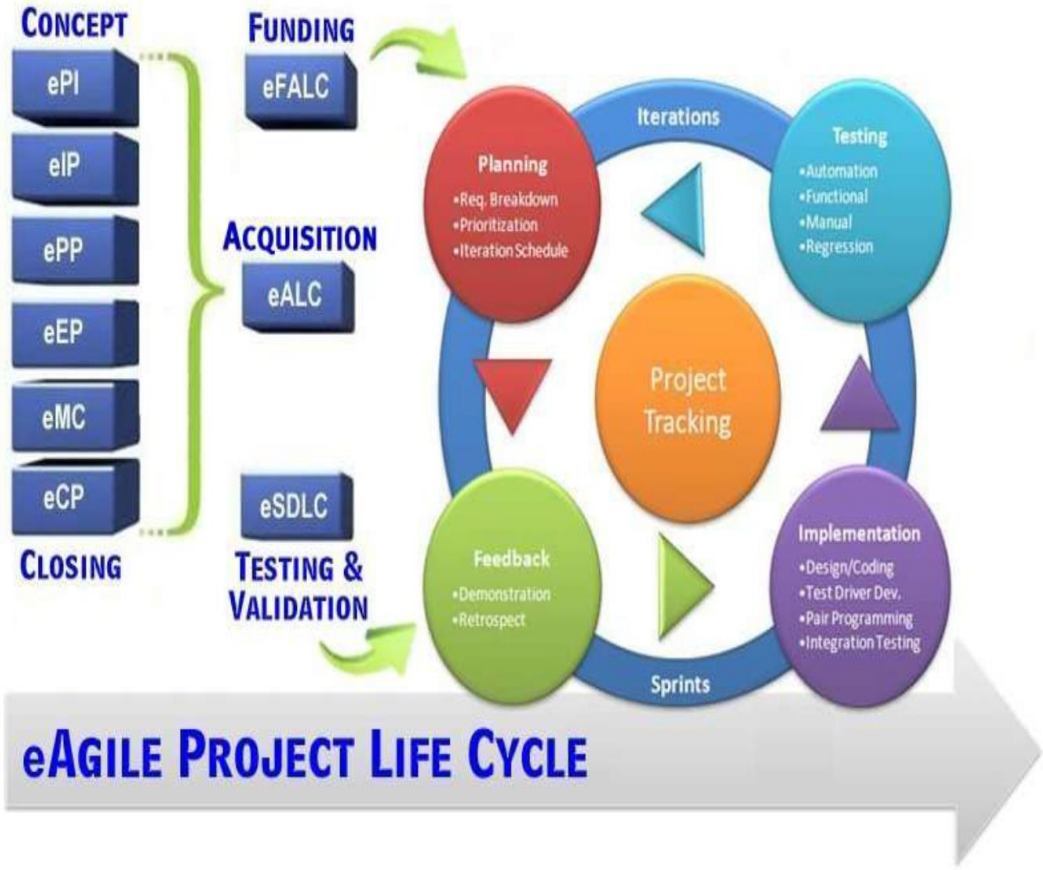

Figure 3.3: Representation of the iterative cycle in the Agile methodology.

The most important advantages of this model are the ability to respond to changing project requirements. This ensures that the development team's efforts are not wasted, as is often the case with other methodologies. The changes are integrated immediately, which allows you to have no problems later. There is no misunderstanding between the development team and the customer, as there is continuous communication with the customer.

The choice of one method rather than another depends on the type of product or service but also other factors. For example, if you want to ensure high development efficiency, the Agile methodology guarantees more satisfactory results than the waterfall method. Efficiency is decided by the quality of the final product, the number of bugs, and the development time used. "One Phase" and "Rigid" development makes it virtually impossible to make any last-minute changes to requirements or design. While Agile methods, thanks to their iterative and adaptable nature, can incorporate changes and release the product in less time.

Besides, the Agile methodology allows for more fluid communication within team members and with the customer, a factor that contributes to improving efficiency. The waterfall model, on the other hand, is suitable for the development of already stable programs, that is, when their design does not need major changes and in situations where designers can accurately predict the problems that may arise. It is a natural choice when the customer has provided a clear list of requirements, which are likely not to change. On the other hand, when the customer does not have clear requirements or expectations from the final product, the Agile model may be the best solution.

Another point to consider is the time frame within which the project should be finished. When the time frame is long enough, the Waterfall path can be chosen, while fast delivery projects are best handled in the "Agile" way. (Abrahamsson et al., 2009). The Agile model is applicable more immediately and less difficultly with small-medium sized projects. In addition to great difficulty, large projects involve very high costs: small projects involve small reprogramming (use of low costs and time), large projects involve large reprogramming (use of very high costs and time).

Until now, the Agile methodology has been described as a foolproof tool for managing processes of any kind. In reality, just like any procedure in the working world, the Agile methodology involves risks that can then lead to real problems that affect the efficiency of the process itself. (Leffingwel, 2010) One of the

biggest difficulties that can be encountered for a good implementation of the approach, lies in the fact that it is very difficult to build a good, truly collaborative team. An uncooperative environment causes the inevitable failure of the methodology, as the lack of synergistic work leads to non-compliance with the established times for carrying out a process operation. Another problem lies in the impossibility of producing a Business Case, due to the iterative and adaptive nature of the Agile methodology. This is because it is not possible to have a forecast of how the final goal will be achieved, consequently, it is therefore not possible to have a reliable estimate of costs. Another criticality of the procedure lies in the impossibility of having "Virtual Teams", or teams whose members are geographically distributed and are not present in the same location. The methodology requires a precise physical place where all members can meet daily for comparison and updating activities.

3.2 The characteristics of Agile Project Management

Like all Project Management methodologies, the Agile method is a set of practices. Below are the common practices of a project managed in an Agile way.

Iterative and Incremental Development: This is a development technique used in the timebox. Each iteration includes the activities of analysis, design, implementation, and testing and during which it is possible to add parts to the product (Iterative Development) or refine it (Incremental Development). It is based on the feedback obtained in the previous iteration to improve subsequent iterations.

Test: an important element in the Agile methodology is testing, i.e. the verification of the correct functioning of the system. Without testing to determine the status of a project, many of the Agile project management practices become impossible or ineffective. Testing is a practice that brings benefits in terms of quality, design, reliability, and the metrics it provides.

Feature-Based Development: In Agile project management, the customer is asked to prioritize features for each iteration and invites customers to provide feedback that can be incorporated into iteration planning and feature prioritization. In iterations, as the project grows, low-value features can be cut from the program (saving time and money) while those that offer the most value are identified and implemented. This technique can fundamentally change the initial set of requirements.

Cross-functional and self-organized teams: The team working on a project must be composed of cross-functional figures, that is, who can carry out all the activities necessary to complete the project and who have all the necessary skills. The team is usually small, preferably composed of a minimum of 5 people to a maximum of 9 since through this configuration greater efficiency is achieved and there are no internal communication problems.

Documentation: Agile project management favors limited, flexible, and just-in-time documentation. Initial and final documentation should be a simple, high-level overview of important features. As the tests are updated, this documentation is also updated. Development teams can use flexible and collaborative documentation systems such as Wiki (a website that allows each user to add new content or modify existing ones) to easily acquire and update essential procedures or product specifications. Such systems are effective in communicating between the team itself and the larger organization. All parties involved in a project are brought together for a one-day documentation meeting to be edited and refined later by a single editor. This concentrates the documentation effort, maximizes communication between the team, and ensures that documents are as current as possible. Figure 3.4 shows the typical documentation strategies for the traditional and Agile methodology and the curves represent the total effort invested to write the entire documentation of a project. It can be seen that in the traditional methodology a lot of energy is spent at the beginning of the project to detail the work plan and specifications, but also at the end to create a support document for

the Users of the final product or service. In the Agile methodology, on the other hand, since we are aware that the initial plan can be upset, no energy is used in the initial phase to detail the plan, but only at the end of the project will we proceed to the creation of a dynamic Wiki.

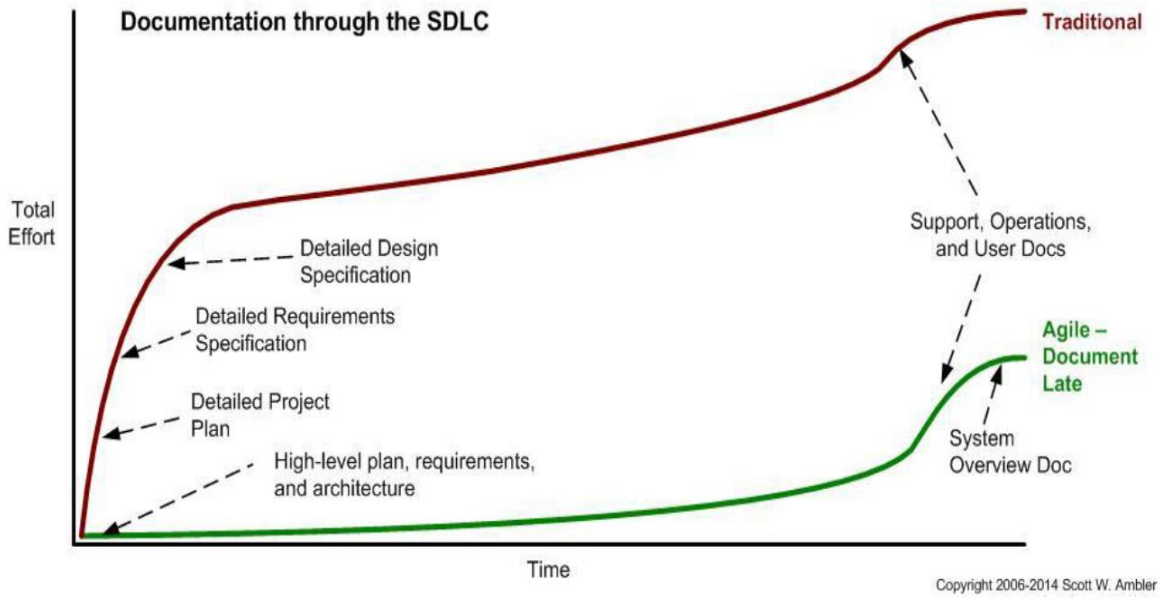

Figure 3.4: Comparison between the effort required for documentation in traditional and Agile methodology.

Communication: compared to a type of vertical communication, characteristic of the traditional approach to Project Management, communication develops horizontally, and this means that there is no real hierarchy, but an equal comparison is encouraged. In this way, communication tends to be simpler and more immediate, ensuring greater transparency, continuous improvement, and customer proximity. Generally, tools are used to improve communication such as the Kanban Board and the Burndown Chart, which will be explored later.

Agile project management, therefore, is well equipped to help project managers and development teams manage risk, budgets, and programs to create successful and valuable products (Karlesky and Vander Voord, 2008).

3.3 Scrum Methodology

Numerous methodologies have been introduced to implement Agile Project Management. Figure 3.5 shows the interest rate of the different Agile methodologies.

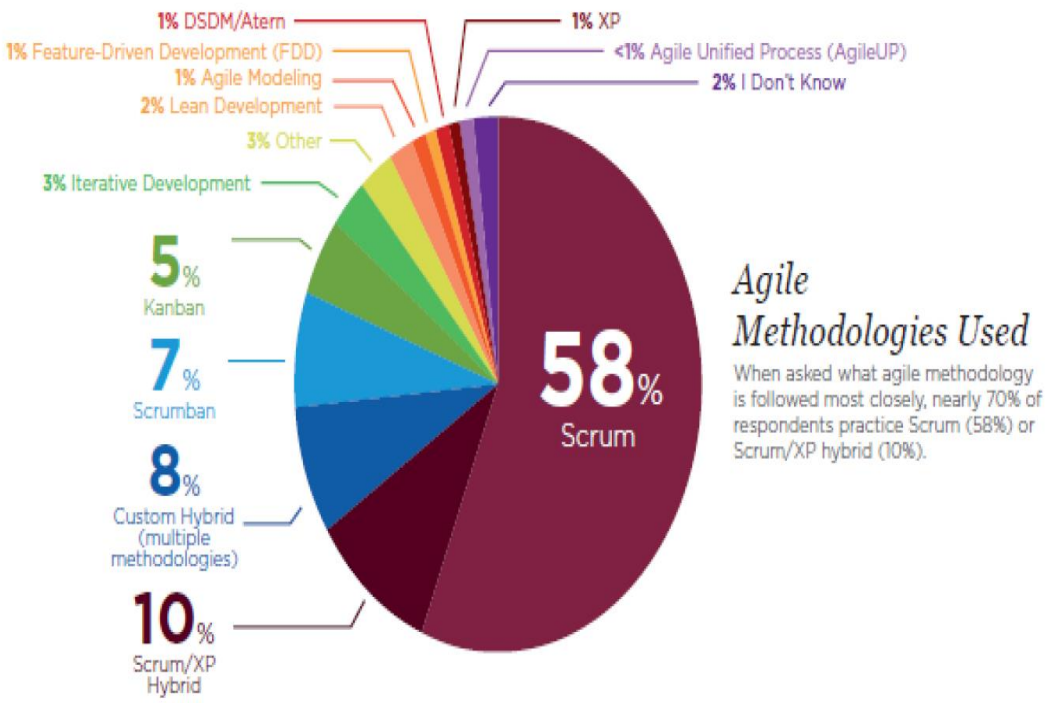

Figure 3.5: Use of the different Agile methodologies (VersioOne. 11th Annual State of Agile Report, 2010).

In this paragraph, we will analyze in more detail the Scrum methodology, which appears to be the most widespread in the market, with 58% of projects managed with this method, as shown in Figure 3.5.

The term Scrum comes from rugby and is a method of restarting the game after a minor infringement. In a management context, the idea behind Scrum emerged in 1986 when a new approach was introduced to increase speed and flexibility in product development (Schwaber, 1997). The idea focused on teams working together towards common goals. They likened this approach to a rugby team where, even if individual players throw the rugby ball at each other, the whole team is aiming for a goal.

The life cycle of projects managed with the Scrum methodology is divided into 3 main phases, as shown in figure 3.6: the Pre-Game phase, the Development phase, and finally the Post-Game phase. The basic roles in Scrum are the Product Owner (PO), team, and Scrum Master (SM).

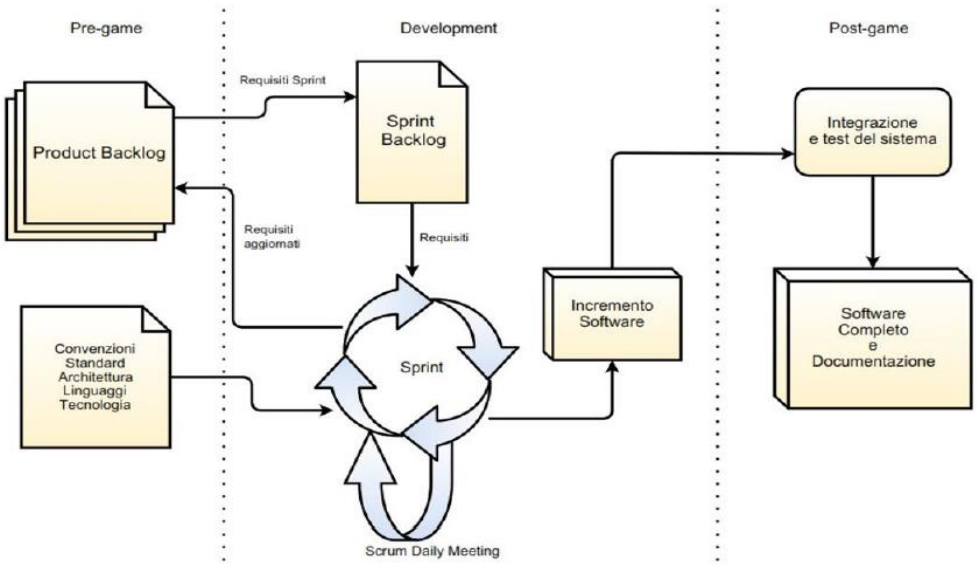

Figure 3.6: Scrum life cycle.

In the Pre-Game phase, the OP, which can be a person or a small team, is responsible for gathering all the known requirements of the project in a document called the Product Backlog and for associating a priority and development estimate of the activities to be carried out. In the Pre-Game phase, standards, conventions,

architecture, and technology are also defined on which the product created will be based.

The development phase represents the main phase of the entire project. The SM is the representative of the project team and is responsible for making sure that the team complies with Scrum methods. The MS organizes the meeting planning sprint where the team organizes the next job progression - a sprint that usually lasts two to four weeks. During the sprint planning meeting, which is the meeting fixed at the beginning of each cycle, the activities with the highest priority are reviewed and divided into smaller units that can be performed within the sprint. The SM and the team together with the OP prepare the sprint backlog, which is the list of activities the team will perform during the next sprint. The SM observes the progress of the sprint through the daily sprint meetings, called the Daily Scrum Meeting, where the member of each team discusses what they did the day before, what they will do today and if there are any obstacles. Such meetings are useful for the team as they ensure transparency on progress. This makes it easier to react to any obstacles. The sprint ends with a presentation of the product by the team, which the OP can then accept or make corrections in case of misunderstanding. At the end of the sprint, before the next sprint begins, the Sprint Review Meeting is held which is a retrospective of the sprint, where the main purpose is to learn from the last sprint. The team looks back and discusses what went well and what could have been done better. The progress of each sprint is represented through the use of a Burndown chart, which shows how much work is still to be completed and which is updated every day after the sprint meeting. The graph consists of a Cartesian diagram in which the sprint days are indicated on the X-axis and the effort days on the Y-axis. Generally, two curves are represented: one that highlights the estimate of the work still to be done and the other that highlights the actual missing work. This shows if the team is active, earlier or later than expected. In the Post-game phase, i.e. the final phase of the process, the last tests are carried out after completing all the customer's requests and, once passed, the application

is released to the customer together with the related documentation (Schwaber, 2004).

Figure 3.7 graphically summarizes the entire Scrum process, from the creation of the product backlog to the delivery of the project.

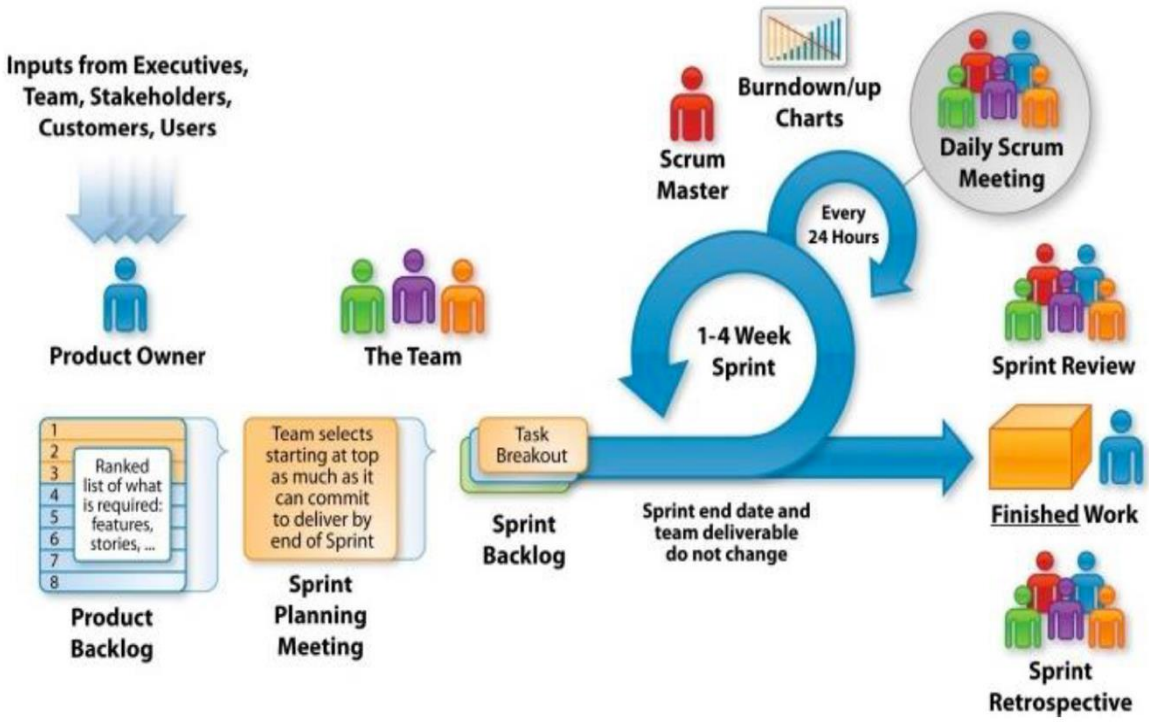

Figure 3.7: Scrum cycle: meetings, documents and actors involved in the process.

3.4 Kanban Methodology

Just in Time (JIT) is a manufacturing philosophy used in many manufacturing companies to produce what is needed with the right quality in the right place and at the right time. The Kanban methodology, which in Japanese means paper or label, when coupled with a pull system is a means to implement JIT. The pull-type organizational logic is based on the principle according to which it is necessary to

produce only what has already been sold or which is expected to be sold in a short time. This type of logic contrasts with the push production method, typical of the Taylorism assembly line, which aims to produce finished products for the warehouse waiting to be sold. Kanban, as a management method, uses cards as visual symbols to activate and control the flow of a production process. The Kanban methodology has recently been used in many manufacturing systems, assembly systems, and supply chain systems very effectively. It is a work scheduling system that maximizes a team's productivity by reducing downtime. Downtime can occur within any process, workflow, or procedure and can usually be traced to opportunities within the process itself. The Kanban system focuses on reducing waste in all information overproduction, unnecessary movement, defects, excessive processing, and waiting. The goal is to have a visual workflow, showing all the steps in the process. The visibility of the flow makes it easier to identify any defects, such as bottlenecks (Liker, 2004).

Although kanban was introduced in the manufacturing industry, in 2004 it had its first application in the IT and software development world. The first to adopt the Kanban methodology in the world of software development was David Anderson at the Microsoft company based in India. From then on, the Kanban method has been well received in software engineering and its use is becoming more and more widespread (Ahmand et al, 2017), in fact, the annual "State of Agile" reports show that the use of Kanban increased from 31% to 39% in 2016 and from 39% to 50% in 2017 (10th Annual State of Agile Report, 2016 and 11th Annual State of Agile Report, 2017). Software engineering has been plagued by numerous problems such as lack of reliability, poor response to change, limited agility, and cost. The Kanban approach is seen as a way to overcome these challenges, enabling teams to respond to dynamic market changes, increase quality, reduce waste, and improve predictability.

The Kanban methodology is also used as an approach to project management and is a method that aims at the complete elimination of all waste. Actions and

activities (tasks) can add value to the project (value-added tasks), not add value but be inevitable and not add value and also be avoidable. Anything that adds no value to the project and is avoidable is known as waste.

The Kanban methodology applied to Project Management includes 5 practices that give indications for the implementation of this new working method in a team (Mircea, 2019):

- View work: The team must constantly see the steps that need to be taken during the process. This is possible using some boards (electronic or physical), called Kanban boards, on which progress can be highlighted. The Kanban board is a blackboard in which columns are inserted. Each column represents a stage in the development process.
- Limit of activities in progress (WIP): a limit is defined and this means that the maximum number of activities that can be performed in parallel cannot be greater than the limit.
- Flow Management: Process flow needs to be improved, so lead time or time-to-market needs to be reduced. This represents the point from which the organization must be changed along with the way of working. In Kanban, a term has been defined to also measure the actual time spent by a development team on a task, called cycle time.
- Make process policies explicit: Everyone needs to know the processes, policies, and practices to suggest improvements.
- Implementation of feedback loops: Fast delivery is not everything, as product quality is very important. Feedback should come from other people within the team, but also from the customer, to make sure that the final product or service ultimately matches their expectations.

3.4.1 Main features of the Kanban methodology

The main idea of Kanban systems is to attach physical tags called kanbans to each component produced within a factory and limit the number of kanbans available according to production capacity. This method allows limiting the quantity of work in progress pieces in the system, obtaining more control of the pieces put into production, and avoiding producing more finished products than needed. They give the possibility to have a general picture of the production by collecting all the kanban cards and placing them on a blackboard that represents the production process. This method is replicated and adapted to Project Management, where each kanban is no longer assigned to a piece or component, but an activity.

The first step in creating a Kanban board consists of collecting all the kanbans on a panel and then detailing the process, dividing the column corresponding to the processing, to specify the various stages of the process. In Figure 3.8, for example, the columns corresponding to the typical phases of a software development process have been created: analysis, development, test, and deployment (in this case identified by the run label).

Sometimes there is the introduction of the buffer columns (or queues) necessary to "park" the activities for which the processing relative to the current column has been completed, waiting for them to be processed in the following phase, which involves the move to the next column.

Finally, on the various phases of the processing cycle, limits can be introduced to the number of activities (WIP limit) that can be performed simultaneously. In the absence of a limit WIP or the presence of an unsuitable value, given the natural propensity that many people have to start new things before having finished those in progress, you will notice an accumulation of tags in certain positions on the board, that is real traffic jams of uncompleted activities.

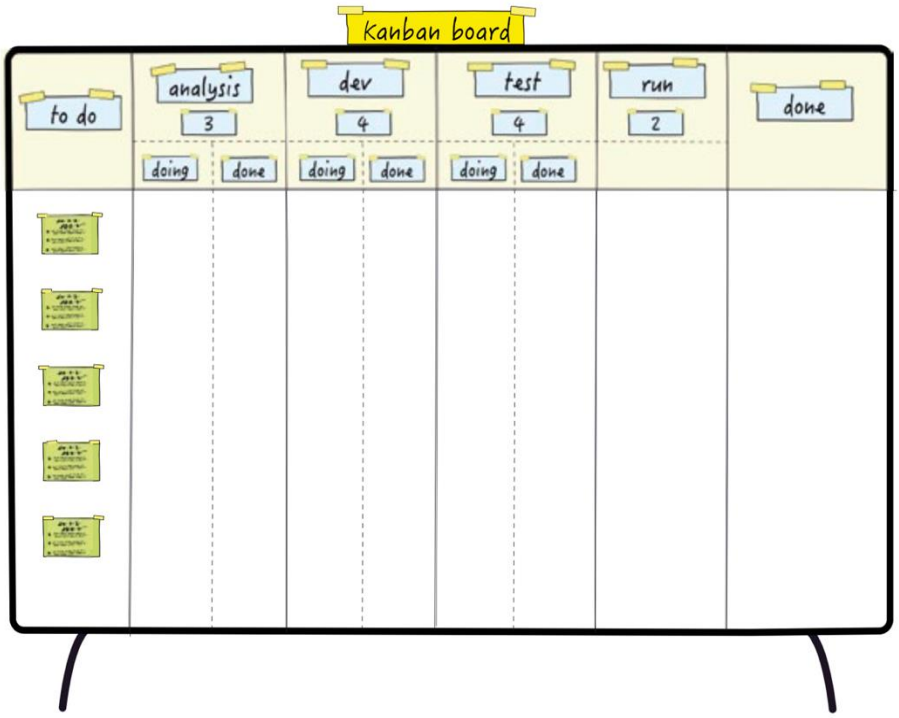

Figure 3.8: Kanban board in initial situation.

In the initial phase of the project, all the activities are placed in the “to do” column since they have not yet been taken over.

The first step is to start the activities in the first phase which is the analysis in the case of the Kanban board in figure 3.9, paying attention to the maximum number of WIPs set for that particular column. As soon as the processing of one of the analysis tasks is completed, it will be parked in the column of activities done, ready to be processed in the development phase. Continuing the simulation, it is observed that the various tasks move to the right giving rise to a real workflow as shown in figure 2.9 below.

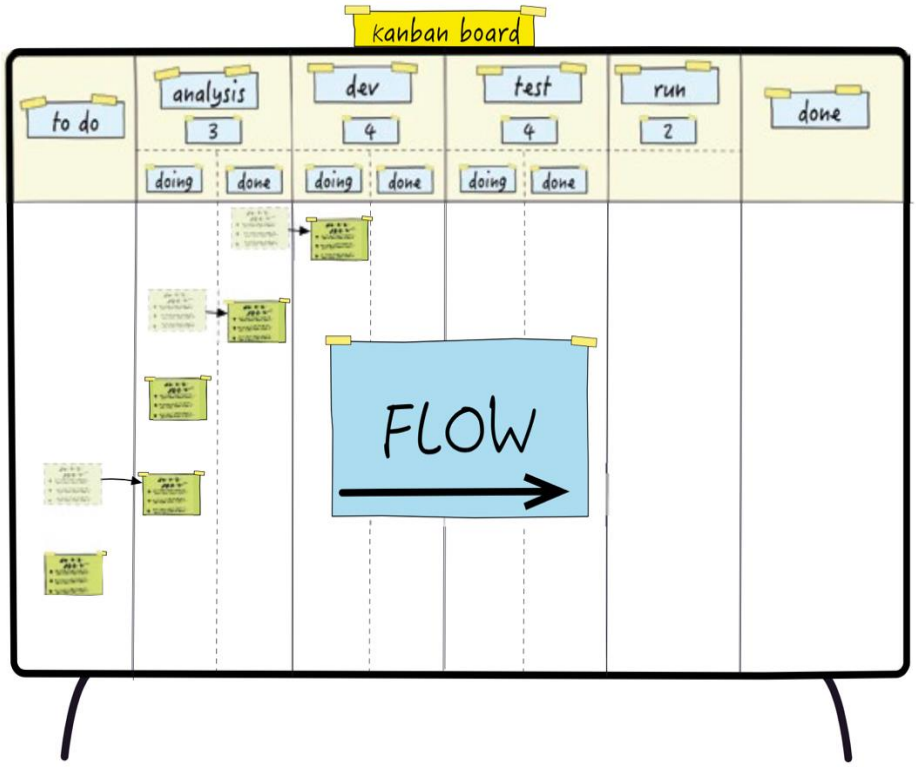

Figure 3.9: The flow of tags in the Kanban board

An effective way to improve the control of the workflow but also to proportionally balance the work between the team members, icons representing the various team members are used to be placed on the tags of the different activities. The icons can be photos or symbols that represent the role one plays. In this way, it is possible to verify the use of team members and coordinate the movement when critical activities arise that must be solved instantly.

In figure 3.10 you can see that when a task is assigned to a member, he takes his icon and places it on the activity and when he finishes it, the icon is repositioned in the appropriate space on the blackboard. During the assignment, it must also be taken into account whether the person on the team has the knowledge and skills to carry out that activity. The optimal condition would be that of having cross-functional teams, that is, that of people capable of carrying out all the activities required by the process. Having cross-functional teams doesn't have to be seen as a mandatory goal, but it certainly offers many benefits; for example, it improves

efficiency, reduces dependencies on individuals because if a team member is absent, the activities in charge of him can still be completed and lower the pressure on the individual by distributing the activities to colleagues.

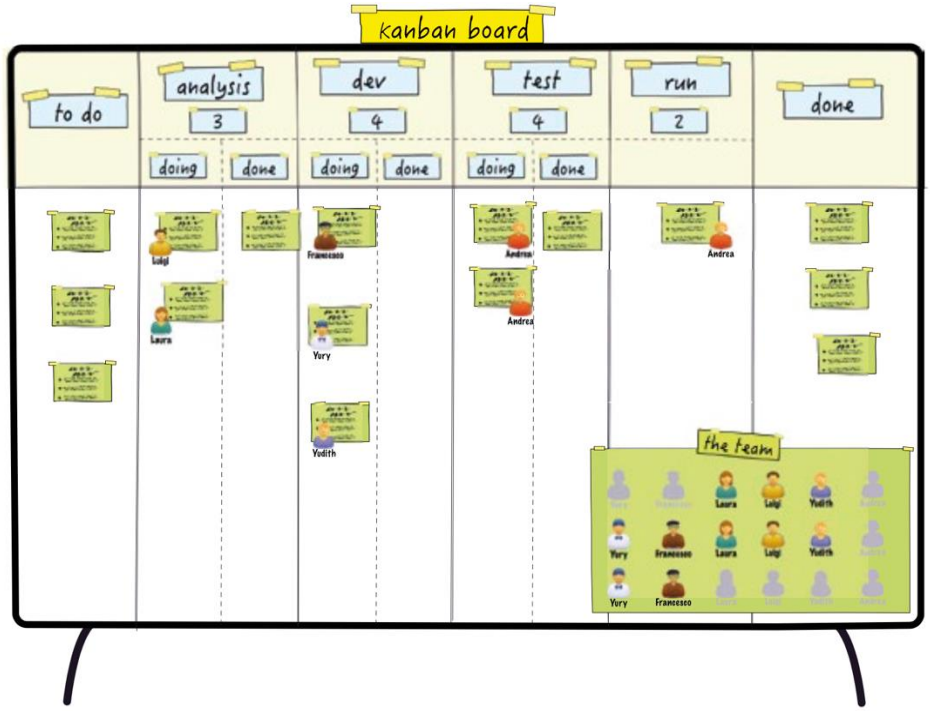

Figure 3.10: Kanbard board with team member icons.

Normally the sequence in which the various activities are put into operation follows the ordering of the tags in the initial column of the To-Do: the priority could be established based on the importance or urgency of the request or simply on the time of arrival. Sometimes it happens that the team has to interrupt normal planning to carry out some high-priority activities, often real emergencies. The solution could be to make a small change to the board and that is to insert an emergency lane or Expedite Line, often identified with a particular color as can be seen from figure 3.11, which allows you to manage those activities that must be fulfilled in very fast times. As can be seen from figure 3.11, when an emergency occurs, the kanban is placed in the emergency lanes and two or more people

interrupt their normal work and dedicate themselves full time to resolving the emergency. When the resolution is complete, the kanban goes into the “done” column and the team members go back to doing the work they stopped. Another way to manage this situation could be to reposition the tag corresponding to the emergency in the column of the processing necessary to solve the problem (for example, again in analysis or development). This solution is the simplest to implement and also the least impactful on an organizational level: nothing remains pending at the point where the error was generated and therefore a WIP is freed for other activities that can be carried out.

A variant, more inconvenient to implement but more in line with agile principles, could be to leave the tag in error in its position and to insert a new tag in the process that will “close” the problem that has emerged. In this case, the tag with the bug awaits the solution of the problem by occupying a WIP, which could lead to the system being blocked: if there are many "abandoned" tags waiting for a solution, nothing else can be performed until the problems were not resolved.

This strategy certainly has a cost: according to the Lean philosophy, this cost must be an incentive not only to solve the problem quickly but, also and above all, to make the necessary changes to avoid that problem from occurring again.

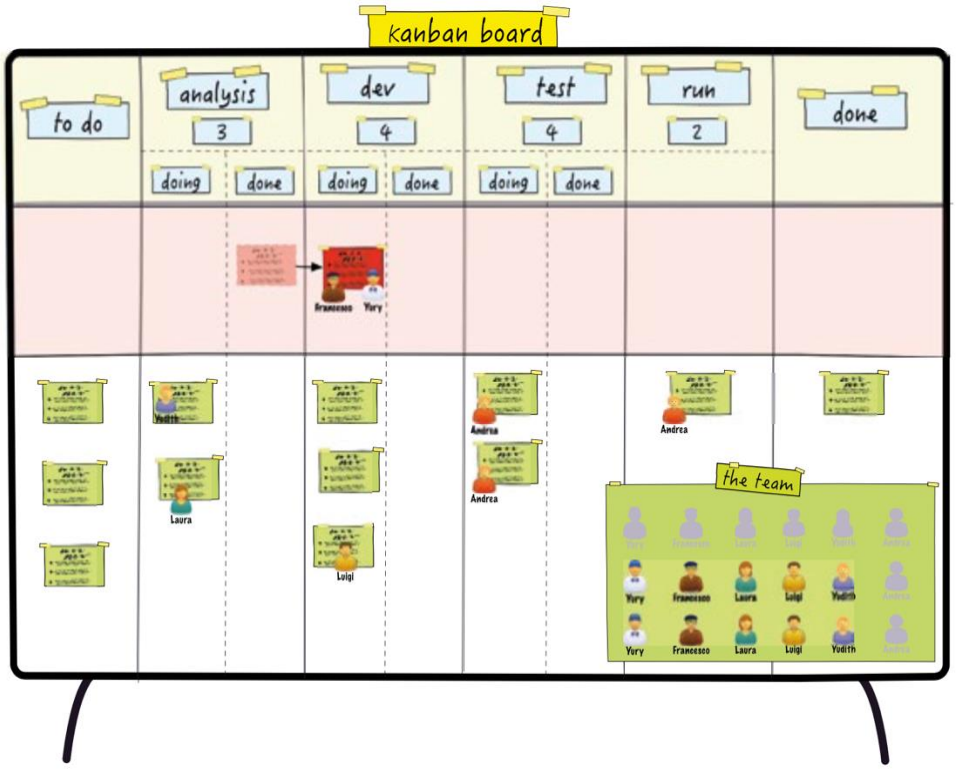

Figure 3.11: Kanban Board with the emergency lane.

In the cases seen up to now, the process is broken down sequentially and that is that each activity must pass through all the columns of the blackboard. However, sometimes it is not possible to break down the process in this way because there may be activities that require to be carried out in parallel.

Each column is divided into sub-columns, one for each sub-activity, and then horizontal lanes are introduced below in the relative column. In figure 3.12 on the left, you can see how the card coming from the previous column will be broken down into sub-folders, and on the right when all the sub-activities have been carried out, they can be combined into a single Kanban which then proceeds normally on the blackboard.

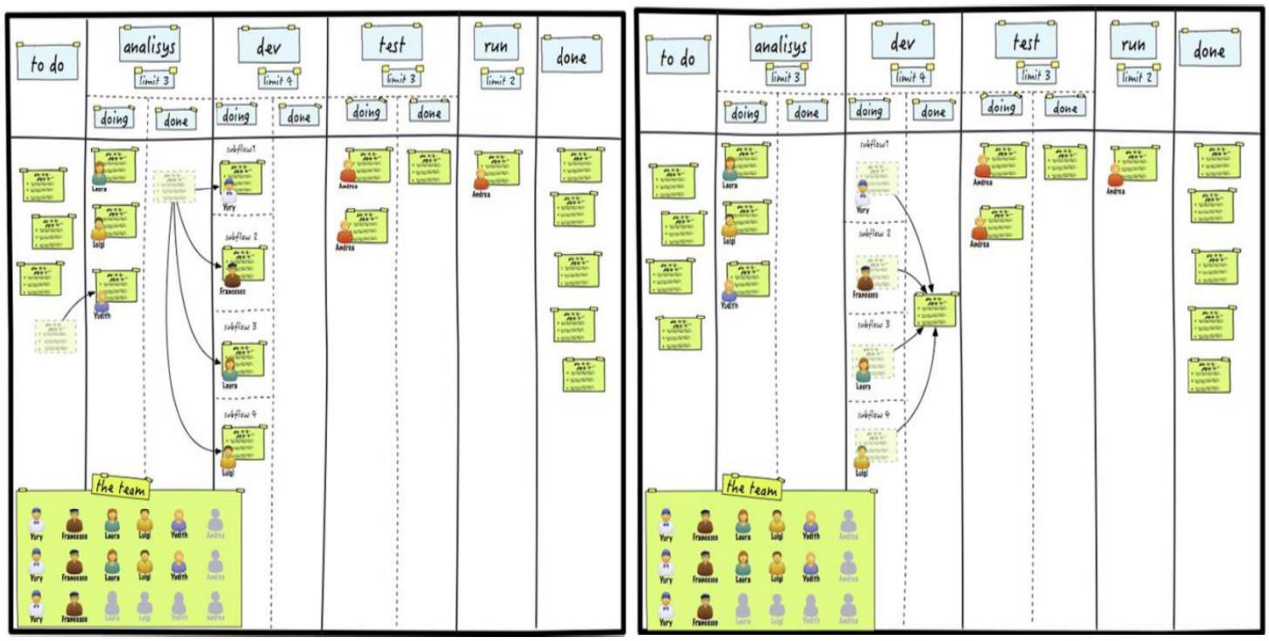

Figure 3.12: Breakdown of the workflow in the case of activities that must be carried out in parallel.

Among the advantages of the Kanban board is that it is characterized by high flexibility and an overview given by the fact that all collaborators know the current progress of the project. Furthermore, the method allows you to finish every single task efficiently, discouraging multitasking and consequently increasing productivity, in fact, team members can focus on a limited number of tasks and do them better. Finally, there are well-defined Time-frames that allow the reduction of wasted time.

3.4.2 Kanban VS Scrum

Even though Scrum and Kanban are both considered Agile methodologies, the approach is different for each of them. Table 3.1 details the main differences. In general, while Scrum has very strict rules and processes, Kanban offers freedom in choosing the way to work and organize the team.

Table 3.1: Main differences between Scrum and Kanban methodology

SCRUM	KANBAN
The roles must be changed according to the new principles and must be well defined (Product Owner, Scrum Master, Development Team)	Current roles and responsibilities are maintained.
Cross-functional teams so that each member can perform and help in any business.	Specialized members who can help with many of the activities if they know, but it is not mandatory to have cross-functional teams.
Total change of old development processes.	Adaptation of current processes.
Periodic meetings and already decided at the start of the project (daily meeting and Sprint planning).	There are no predefined and mandatory meetings, but they are organized when the team wants.
Continuous iterations (sprints).	Continuous work.
The tools used are the Scrum Boards and the Burndown chart.	The most used tool is the Kanban board.

At the same time, however, the two methods share the Agile principles and in fact, even if the working procedures differ, there are some important similarities. In fact, in both methodologies, there is the goal of eliminating any kind of waste and obstacles, and a team leader is never appointed for the assignment of tasks, but there is a team able to organize itself independently. Furthermore, the two methods have whiteboards (Scrum / Kanban boards) to make processes and activities clearer and more transparent and to visualize which activity is in progress, which has yet to start and which is completed. In prior, it is difficult to define which

method guarantees greater performance, but it is possible to define one that is more suitable and effective depending on the situation in which you find yourself.
When you already have well-defined processes, in which the frequency of meetings is chosen directly by the team, the Kanban methodology is the most suitable as it leads to improvements in organizational processes in both management and development and also leads to optimizing processes without going to modify the whole system. If, on the other hand, you do not yet have processes well-rooted in the corporate mentality and are ready for a total reorganization, the Scrum methodology guarantees very rigid rules and processes that lead to results even with small teams.

3.5 Scrumban

The term Scrumban was first mentioned in 2009 and is defined as a transition method for software development teams from Scrum to a more advanced development framework (Ladas, 2009). From 2009 onwards, organizations have thought using the Kanban method in conjunction with Scrum, not only for software development but also to achieve other types of results. The main purpose for which the Scrumban method is used is that it helps teams and organizations to accelerate their transitions to Scrum from other development methodologies, to enable new skills within teams and organizations to help them overcome challenges ahead and to help organizations develop new Scrum-like processes and practices that work. Switching to a new method is never easy, and these changes are best done if they are not drastically implemented but introduced gradually. In this context, Scrumban is proposed as a valid middle ground between the two methods in order to abandon the tools and principles of Scrum and adopt those of Kanban (Cohn, 2009).

To better understand Scrumban, it is important to identify what the limitations of the Kanban and Scrum methodologies are. Scrum tends to completely change the organizational aspect of the company by introducing the logic of Sprints and well-defined roles and tends to isolate the team from external influences as much as possible, representing a clear and radical break with the methodologies previously used by the company. The problem is that the methodology does not give indications on how to deal with any management and organization problems that could arise after a total reorganization of the processes. Also, there has been criticism of the Scrum methodology because it is argued that the team is not focused on finding innovative solutions to problems, but the only goal is to deliver the activities that you have as you try to execute the dictated rules in detail . What is criticized is the mentality that has brought the Scrum methodology among professionals, a mentality that favors work completion over innovation (Cohn, 2009).

As for Kanban, this methodology cannot be properly defined as a Project Management framework since it does not introduce new techniques but tends to improve the previous ones through visualization tools. Kanban is a method that allows individuals, teams, and organizations to better understand how they are performing.

The Scrumban methodology is not only the union of some elements of Scrum and some of the Kanban, but it tends to apply the Kanban system within the Scrum methodology to improve the use of the Scrum methodology and improve those defects that can create the radical change of processes with the past (Reddy, 2016). Scrumban is a method that amplifies the capabilities already present in Scrum by adding new features that can aid in an evolutionary change. Scrumban helps improve agile project management practices by pushing the boundaries of Scrum with the help of Kanban (Bhavsar et al, 2020).

There are essentially three potential starting conditions for Scrumban launch:

- A project involves a team or organization for which both Scrum and Scrumban represent new ways of working.
- A project involves an existing Scrum team or organization that will use Scrumban to improve their command of the responsibilities and meetings associated with the Scrum framework.
- A project involves an existing Scrum team or organization that will use Scrumban to monitor performance, diagnose problems, and adapt existing practices to the form that best suits their context.

The building blocks of Scrum that are used in the blend are planning and review sprints and prioritizing tasks in the backlog. The Kanban Fundamentals that are used are the explicit limit which indicates how many work items must be processed at a time and the clear view of activities as they transition through each stage. Scrumban differs from Scrum in that it emphasizes certain principles and practices that are very different from the traditional Scrum foundations.

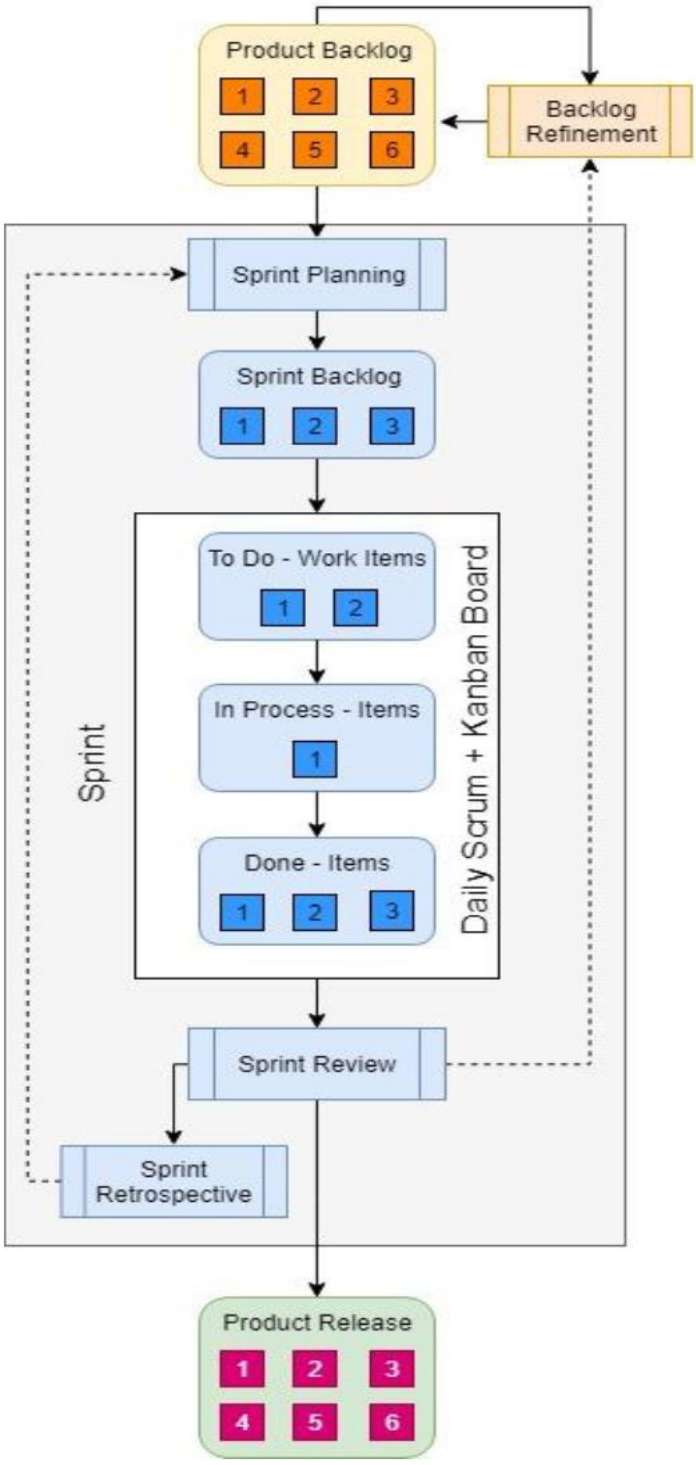

Figure 3.13: Representation of the process in the Scrumban methodology.

Figure 3.13 represents the system flow for the Scrumban methodology by adopting fundamental elements and processes of Scrum and Kanban. Activities are collected

in the Product Backlog pool, where they are refined each time changes are requested by the customer or stakeholders. Therefore, it can be seen that also in the Scrumban methodology the role of stakeholders is central. The refinement process includes prioritizing activities which are then entered into the Scrum Backlog during the Sprint Planning event.

The role of Kanban starts from this stage onwards. Activities are converted to Kanban cards in the Sprint Backlog by members of the development team. Each contains one or more specific sub-activities with accurate estimates for each activity that passes through each state ("to Do", "in Process" and "Done"). Completed activities are then moved to Scrum Review events where they are inspected by the Scrum team together with stakeholders (Ladas, 2019).

Listed below are all the main differences of Scrumban versus Scrum and Kanban that are not possible to see by looking solely at the process.

Management: in the Scrumban methodology, unlike the Scrum one, the role of management is recognized. It has a leader who helps all team members gain the ability to see and understand concepts such as blockages and bottlenecks and who also ensures that the team does not get stuck in a comfort zone, encouraging team members to always look for ways to improve.

While teams can get the job done simply by employing Scrumban independently, an active management level is essential to realize large-scale benefits (Yuval, 2012). Improvements or innovations are less likely to occur without a manager to maintain contact, help solve problems outside the team domain, and oversee the extent to which teams spend time and effort researching discovered improvement opportunities. In traditional Scrum, the idea is that teams are self-organized and will work and coordinate, however in practice there are always gaps between how a team has to organize and how things are going.

Kanban board: is a Kanban concept that aims to reduce delivery times for all activities and improve productivity. Scrumban allows for internal workflow

management by assigning a status (such as "To Do", "Doing" and "Done") to each work item.

Work in Process: Scrum does not provide any specifics on viewing and limiting work items that are in progress. Scrumban, thanks to the Kanban Board, limits the overload of the work item by implementing the WIP limit in the Kanban board on each work item.

Iterative Approach: Scrumban also differs from the Kanban method because it recognizes the organization in teams and also the value of Sprint iterations when appropriate (Paul and Rahman, 2018). Kanban does not support the continuous integration of development activities with iterative cycles and the continuous delivery of releasable functionality as in Scrum. Scrumban allows to apply sprint logic to release new features at each cycle.

Scrum's time-boxed Sprint remains the primary vehicle for coordinating the delivery of completed work and replenishing the team with new work. The Scrum-compliant team is used to daily or weekly check-ins. Employing Scrumban, however, offers the opportunity to shift the focus of the meeting away from status and engagement to more proactive planning. The kanban card already provides status information and should detail who is working on what. Obstructions should also be displayed. As a result, the focus of the team discussion can shift to a collaborative effort aimed at identifying potential impediments, handling required policy exceptions and otherwise discovering how the work is done.

Scrumban has iterations, which shouldn't last more than 2 weeks, but unlike Scrum, it allows for long-running tasks that can span multiple iterations. This can lead to an incomplete product at the end of the iteration, which is why Scrumban introduces Feature Freeze (FF). In Scrumban, work is entered as and when needed in a queue unlike the approach used in a traditional Scrum where all work to be completed within a sprint is assigned at the start of the sprint to the sprint backlog.

Duration of the project: the Scrumban provides a project of average duration for the organization, ideally around 3-4 months. This is plenty of time for teams to

start mastering the framework and see the benefits. It is usually also sufficient to demonstrate that the new approach will lead to similar success in other contexts.

Team Size: Scrum limits the number of team members to between 3 and 9 for a single Scrum team, which is the ideal number for small or medium-sized projects. But it limits progress for large-scale projects, and Kanban has no specifics on team size. So in the Scrumban, the number of team members remains between 3 and 9.

External Stakeholders: Kanban has no specifics on the involvement of external stakeholders. Review, retrospectives, and daily meetings - these are the very important ceremonies that Scrumban preserves from Scrum. The review provides the team with direct feedback from product owners and key team stakeholders, such as product managers and customers. Scrumban offers multiple opportunities for stakeholder and team engagement - before, during, and after each Sprint

Role: Scrum limits the role of team members between Product Owner, Scrum Master, and Development Team Member. Activity-specific roles are essential within development team members and define their involvement in the team. According to Scrum, the development team is an inter-functional team and they have all the skills to achieve the required goal, during the Sprint. Kanban has no specifics on the roles of team members. But in the real-world situation, development team members most often depend on skilled experts to complete their tasks (Al-Ratrout, 2019).

Regarding the roles of team members, it is also possible to make a transition from Scrum to Kanban. Starting from the tasks already present in Scrum, a transition to Kanban can be made by reassigning the tasks, trying as much as possible to maintain linearity of responsibilities in the transition.

As far as the Development Team is concerned, there are no changes to be made, even if Kanban allows for the integration of more members involved in the implementation. The Product Owner can become the Service Request Manager in the new team configuration, in fact, he must be able to communicate with the

customer and to understand what priority to assign to each activity, and to understand which one needs to be completed in the short term.
The Scrum Master is probably the best person to take on the role of Service Delivery Manager. In fact, he must have the quality of managing a Scrum Team to direct the team's activities by assigning them to the members he believes are best suited to carry them out.
The role of the Kanban Coach may require the introduction of a new person within the organization if no member of the team has already had experience in the Lean or Kanban world. Over time, as the team has become familiar with the new method, this role can be assigned to the Service Delivery Manager.

3.6 Agile tools

Nowadays, in projects involving project development, it is necessary to maintain control over all activities, because a lot of data and requirements are involved, and it requires levels of productivity and quality. The Kanban Board, for example, is supported by some online project management software, such as Trello and Jira.

3.6.1 Jira and Confluence

Jira is a planning-capable tool that allows the Project Manager to plan activities that need to be carried out flexibly, involving the whole team is working together to finish the task (Garcia et al., 2019).
Growing day by day in the software projects industry, JIRA, created by Atlassian in 2002, is work software widely used to track software bugs and schedules and is also commonly used for Agile projects. Thanks to the ability to document which activities are pending, it helps in many projects based on an agile methodology. It is software designed for each member of the application development team to plan

and monitor. Jira Software is an agile management tool that supports any agile methodology be it Scrum or Kanban.

You can create activities, plan sprints, and distribute activities for the entire team, with the ability to prioritize and discuss the work done in that context with full visibility. From there, the team involved can use the Kanban board to report in real-time what is happening in the project.

Jira allows you to create a section called "backlog", which contains the priority activities that must be carried out in the immediate future, as described in the paragraph on the Scrum methodology, and which everyone can access to view their activities and those of others. As shown in figure 3.14 below, the resources can also see the estimated times, the priority and in general the details useful for the development of the project for each activity. Jira Software makes the backlog the center of job planning, so you can estimate times, adjust scope, check speed, and re-prioritize problems in real-time.

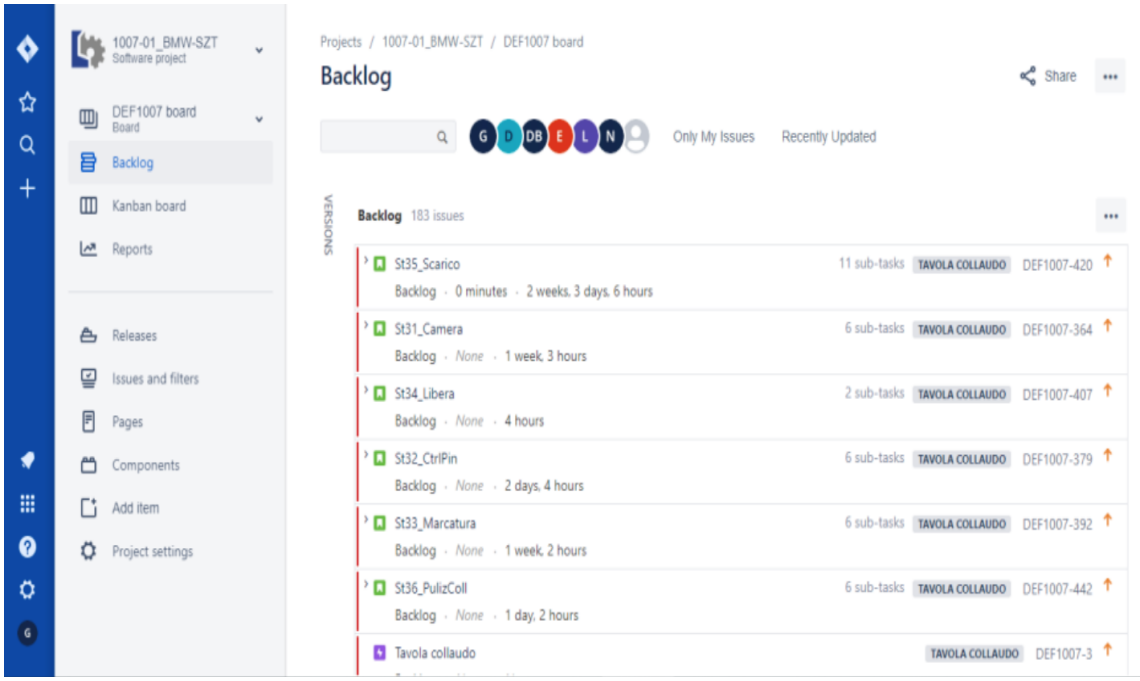

Figure 3.14: Backlog in Jira Software.

Besides, Jira also allows you to create a Kanban board or Kanban board, fully described in the paragraph on Kanban methodology. The software allows you to create different columns such as "in progress" or "done", in which cards that represent the activities are positioned inside and both the Project Manager and team members can move them from one column to another as if they were real post-its. Also, for each activity, the resource can mark the time they spent working on it and any possible comments or notes that the rest of the team could use.
Another fundamental element of Jira is that of “issues” which are problems raised by a user. When the user, who can be a member of the team or the customer, finds bugs or malfunctions, ticket issues are created. Once an issue has been opened, the Product Owner decides whether the problem is important or not and is eventually entered in the backlog to be taken over by an operator and can go through the following phases that form a workflow:

- Open: the opening of the issue;
- In progress: once a new workflow has been opened the problem goes through this second phase in which the operators take charge of the problem, but it is not yet solved;
- Resolved: the problem has been resolved, but the workflow has not yet been closed;
- Closed: the stream is closed. Note that the problem does not necessarily have to be solved to be closed; due to various factors, the hypothesis in which the problem cannot be solved or has been solved by the user may occur and, therefore, does not require any intervention by the operator.
- Reopened: a previously closed problem, identified by a ticket, is reopened by the customer.

When a ticket is taken over, the status cannot be changed until the workflow is closed. Once a single report has been entered, or a problem raised by a user, the

possibility is given to follow its progress also by e-mail, through a system that keeps track of the single ticket.

Another useful software for developing Agile projects is Confluence as it allows you to spread transversal knowledge and skills to all team members. In detail, Confluence is a tool used to help teams collaborate and share knowledge efficiently, organizing content through the creation of spaces, pages, and blogs that can be commented on and edited by all team members easily thanks to the functionalities sharing. Confluence and Jira have been designed to complement each other, for example by giving Confluence users the ability to view, interact, and reference Jira issues from a wiki page. Confluence promotes collaboration, the collection of functional specifications, during the discussion of the project, and Jira for the visualization of problems or the creation of new ones within the same environment by changing only and immediately the application that is needed. It is used to share knowledge and transmit specific skills to all team members to have a cross-functional team required in the Scrum methodology and desired in the Kanban one.

3.6.2 Trello

Another software that is based on a kanban board-style vision for project management is Trello, which was launched in 2011. The software does not allow the creation of a private workspace dedicated exclusively to the company. Only individual boards can be private, visible only to a workgroup. The boards can be customized by adding or removing columns from the bulletin board, where the various boards are added inside them, i.e. the tasks to be performed. Team members can be of two types: an administrator or a regular user. Both can view, create, and edit group boards, while only the administrator can also change the group's settings and permissions.

The Trello home of each user shows all the boards, or kanban boards, to which they have access, divided by work team. Once created, the board is empty and you need to create a column, where you can enter the various activities to be carried out. Generally, the columns "to do", "doing" and "done" are created, as shown in figure 3.15. Also, in the figure, it’s possible to see on the right the settings menu where you can, for example, add a new member to the board, change the permissions of a user, filter the view of the cards. At the bottom, there is a simple flow of the various activities updated in real-time. The options for card filters are different, such as by label, who they are assigned to, and when they expire. On the left, however, you can create a new activity in any list of the board. Furthermore, by dragging the activities with the mouse, they can be moved from one column to another.

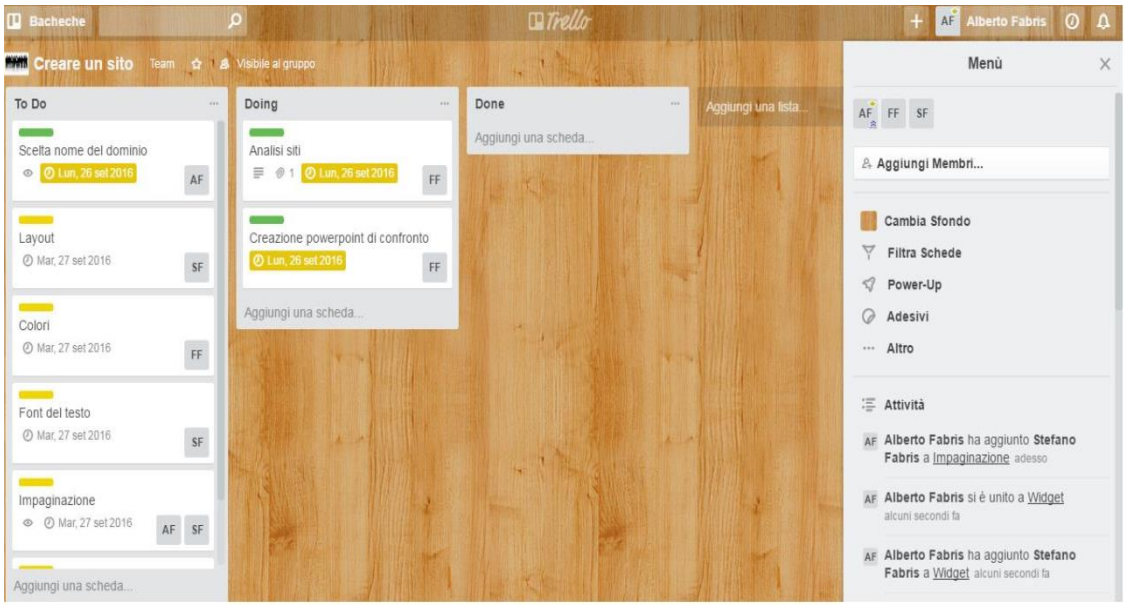

Figure 3.15: A dashboard of a Trello Kanban Board.

4. The Agorize case

The internship took place in Paris at the Agorize headquarters from September 2020 to March 2021 and the activities that I carried out were in support of the Engineering Project Manager. While carrying out the internship activities, it was possible to observe and analyze the current processes and take part in the decision-making processes regarding how the future situation will be and regarding the management methods of a platform development project in an innovative context, characterized by uncertainty and turbulence. Moreover, I have been part of the development team that will introduce later.

The subject of the Case Study is a study of the current situation for the processes of creation of Landing Pages and challenge display and how it is possible to improve this current situation. The aim of the project is therefore to improve the internal process of creating the challenge displays and the Landing Pages. This goal was achieved through the creation of a new web platform.

The chapter starts with an overview of where and how the challenges are built. Subsequently, an as-is analysis of the current situation is carried out and a to-be analysis with a risk analysis that involves a change in business processes. Finally, an analysis is made on the project management methodology used to create a necessary tool for realizing the to-be situation.

4.1 Platform structures

A web platform is a basic software system on which web programs and applications are developed and/or executed. A web application generically indicates all distributed applications or applications accessible and usable through a network, for example, an Intranet within a computer system or through the

internet, i.e. in a typical client-server architecture, offering certain services to the client user.

In the case of Agorize, web applications are the challenges that can be used through the Internet and which are developed on web platforms created by Agorize.

Each Agorize platform is created for a specific client who wants to organize a challenge. A web platform is a tool that will allow you to reach different categories of participants since it can include several challenges within it (for example if a company wants to create a challenge for startups and another just for students with two different themes or organize two challenges for two different areas). Figure 4.1 shows that the platform includes a general back office (to define and manage the entire platform) and a back office for each challenge to create and manage each challenge individually. Just as there is a front part of the entire platform that allows you to view and access all the organized challenges and a front part for each competition in which it is presented. Sometimes customers ask for the creation of a Landing page, which is a presentation page of all the challenges organized by the company and which must be as comprehensive as possible since it must explain the theme of the competition and the objectives, but at the same time the most engaging as possible because it must convince possible participants to sign up for the Challenge.

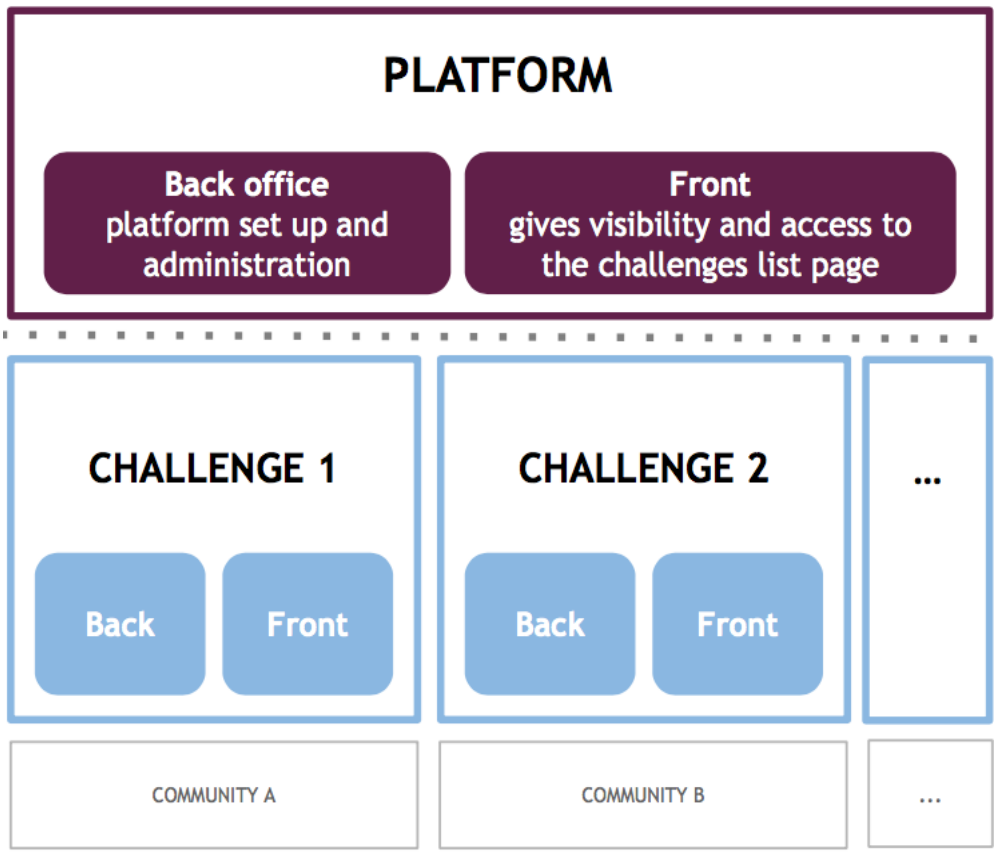

Figure 4.1: A platform structure.

Users who register in the challenge can have various roles:

- Super Admin: is a user responsible for the entire platform and all competitions. He can change the settings of the entire platform, create, modify and manage all challenges. In general, EPMs have the role of Super Admin;
- Admin: is a user responsible for one or more challenges and can edit and manage the entire challenge, can contact participants and add administrators and mentors. Generally, the Project Manager is the Admin;
- Mentor: is the user in contact with one or more participants to guide them during the final phase of the challenge. He can contact them, he can see the participation and he can also be a member of the jury. Generally, the Mentor is represented by employees of the organizing company and therefore by people outside Agorize;

- Voter: the user who has access to the platform exclusively to vote for the best project and define the winner. The Voter is one of the members of the jury and can be a person both internal to Agorize and external;
- Participant: is a user who takes part in a challenge. He can create and join a team, he can cancel his participation and contact other users.

Figure 4.2 summarizes the difference in permissions between Super Admin and Admin; the Super Admin is the person in charge of the platform in general, while the admin is responsible for the individual challenge.

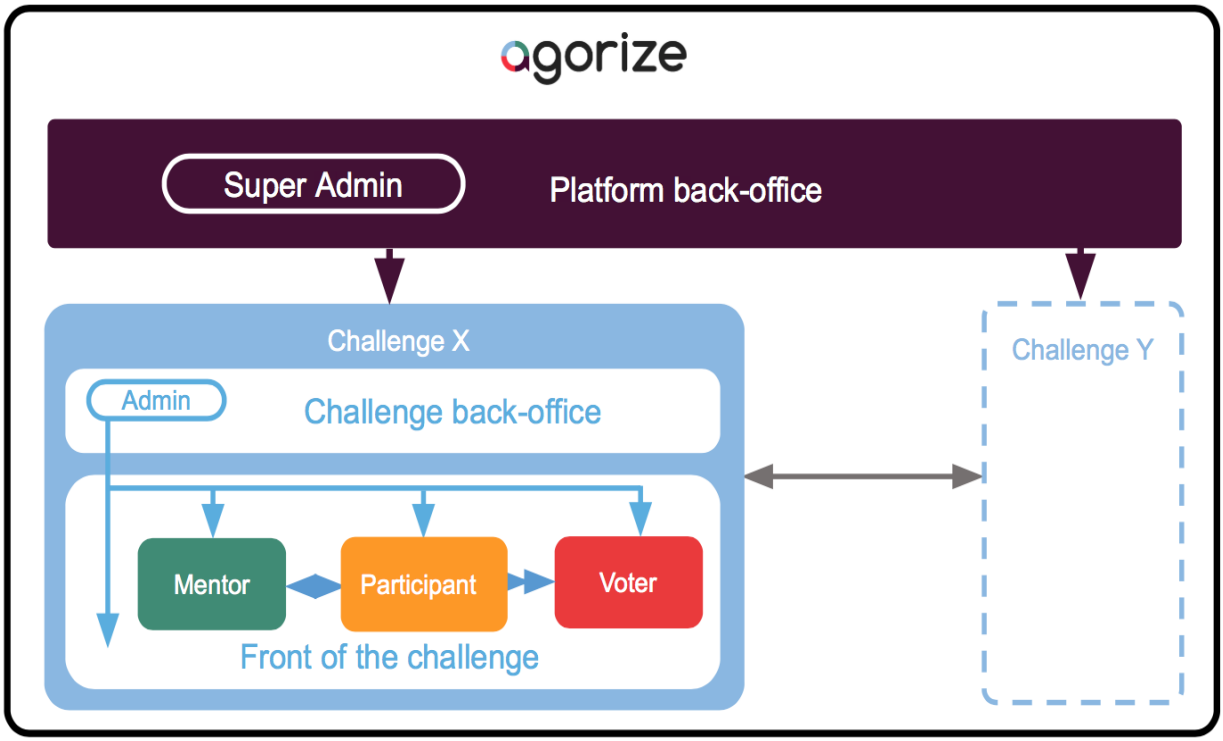

Figure 4.2: Permission scheme in a platform.

The Super Admin can configure the platform settings. Figure 4.3 shows a screen of the platform settings. There are many sections and the most important are:

- Challenges: it is possible to consult and add all the competitions desired by the customer;
- Users: allows you to control all the participants in the challenges, in which challenge and in which team they are;

- Admin & Mentors: allows you to control all administrators and mentors of the challenges;
- User exports: allows you to download an excel with all the data of the participants (name, surname, place of origin, etc. ..);
- Landing Pages: Allows you to create Landing Pages with a programming environment for HTML, CSS and JavaScript.

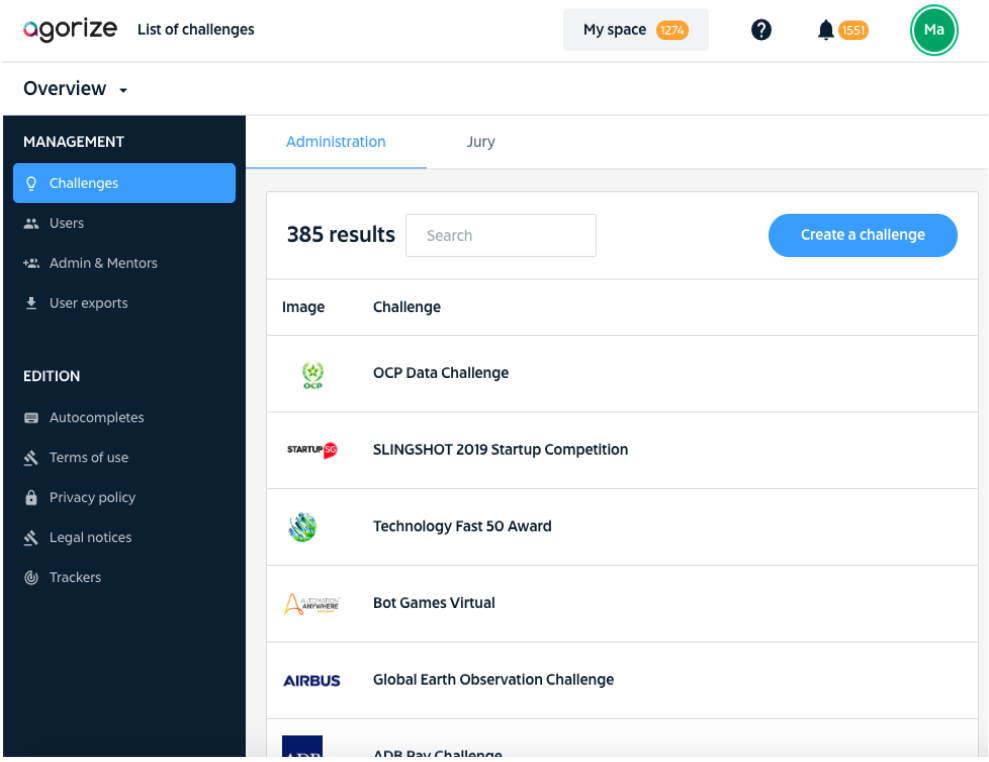

Figure 4.3: Screen with the configuration options of a platform.

To customize and configure each challenge, a page similar to the previous one allows you to modify or check the settings of a single competition, as shown in figure 4.4.

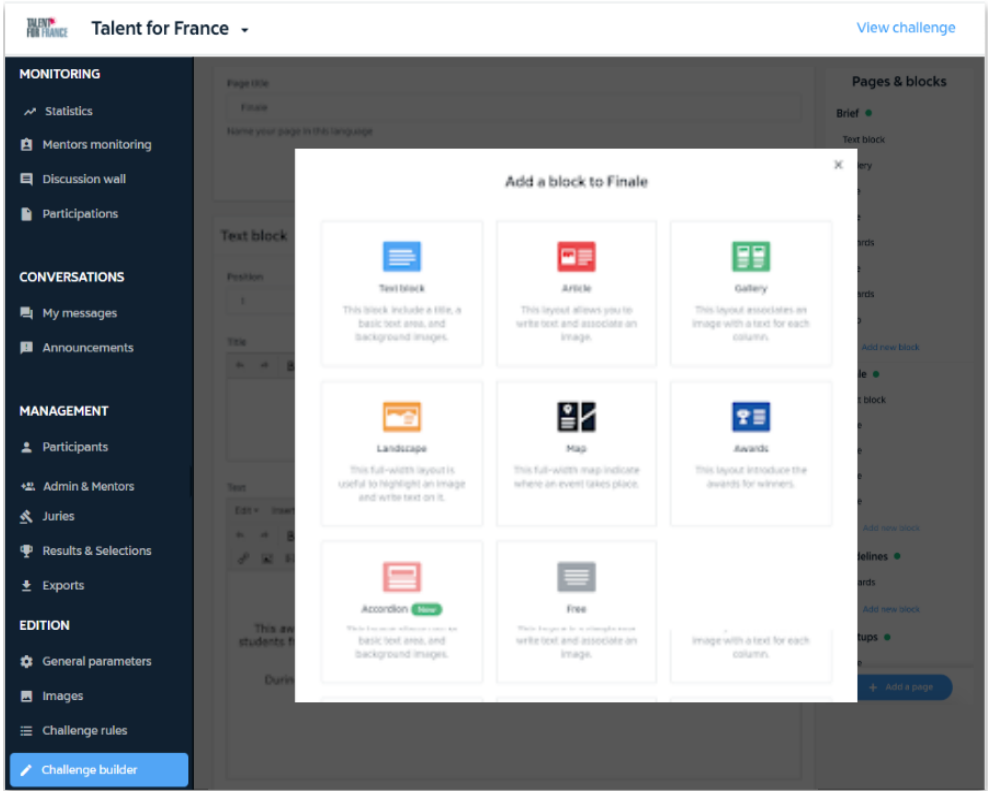

Figure 4.4: Screen with the configuration options of a challenge.

Here too there are all the elements described above but applied to a single challenge. Furthermore, in the Challenge builder section (selected in figure 3.4) there is a programming environment where PMs can graphically create the challenge (the challenge display) and add different pages to illustrate the prizes, rules and objectives of the competition.

4.2 Overview of challenge display

In this paragraph, the challenge overview and the various pages and sections that are made available to the user will be presented.

At the top of the main page of the competition, the user finds the platform header, shown in figure 4.5, which is visible to everyone and from any page of the challenge. It allows users to navigate between the various challenges to which they

are registered, register on the platform and participants to access their participant space. The question mark on the header allows you to ask the administrator a question.

If the challenge is hosted on the Agorize.com platform, this section cannot be customized, as it is shared with all the other challenges. If, on the other hand, the competition is hosted on the client's website, the logo can be customized.

Figure 4.5: Header of a challenge.

Immediately below is the banner and an example can be seen in Figure 4.6. This banner gives a visual identity to the challenge and will be visible from all pages. To make the platform fully responsive, the banner will need to be designed in three different formats so that it can fit mobile, tablet and desktop.

Figure 4.6: Banner of a challenge.

Immediately after, you will find the header of the challenge and as shown in figure 4.7 it is divided into three different sections:

- Presentation of the challenge with the logo, the name of the challenge, the subtitle or the final line of the challenge, information on whether it is an individual or team challenge and a countdown.

- Participation button (only during registration) and social sharing button to encourage users to participate and share the challenge on social networks
- Navigation: a menu to navigate between the different pages of the challenge.

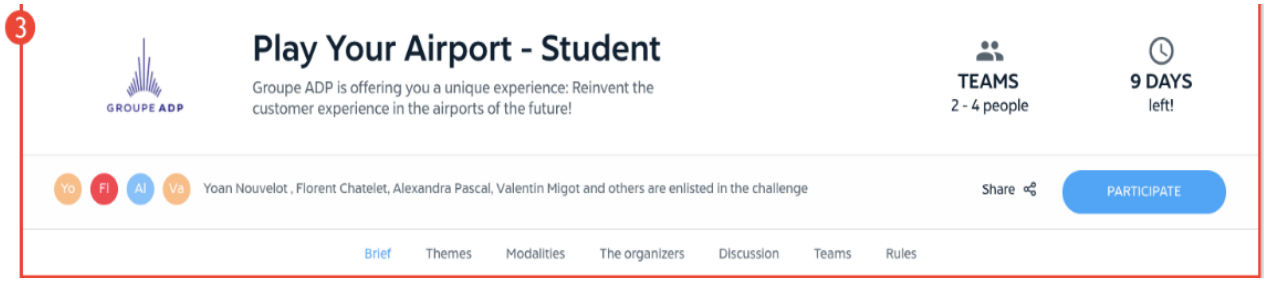

Figure 4.7: Header of a challenge

Then it's possible to find the actual pages of the challenge. There are four different types of pages on Agorize: the content pages (such as "Brief", "Themes", "Modalities" or "The organizers"), the discussion page ("Discussion"), the page with the Teams ("Teams") and the Rules page. In the content pages, such as the challenge brief, the opening paragraph, themes, rewards, information about the organizer or the final event are inserted, as can be seen in figure 4.8.

Among the content pages usually used there is one to present the challenge as a whole, one to present the themes if there were more than one, one with guidelines and timeline and one to provide more information about the organizer and the potential partners.

PMs can customize the content pages thanks to the standard blocks that have been introduced previously and which will be explored in the next paragraph.

As for the discussion page, it was created to encourage participants to share information. Administrators can also use this channel to interact with attendees.

On the Team page, potential participants will find all the information on the teams already registered for the challenge. Each team briefly describes their project and indicates which type of profile is missing to complete it in case you want to hire

other members. If a user is interested in a team, he can apply directly on this page using the "apply" button.

Finally, on the rules page, admins will upload the rules so that users can accept your terms and conditions when they join the challenge.

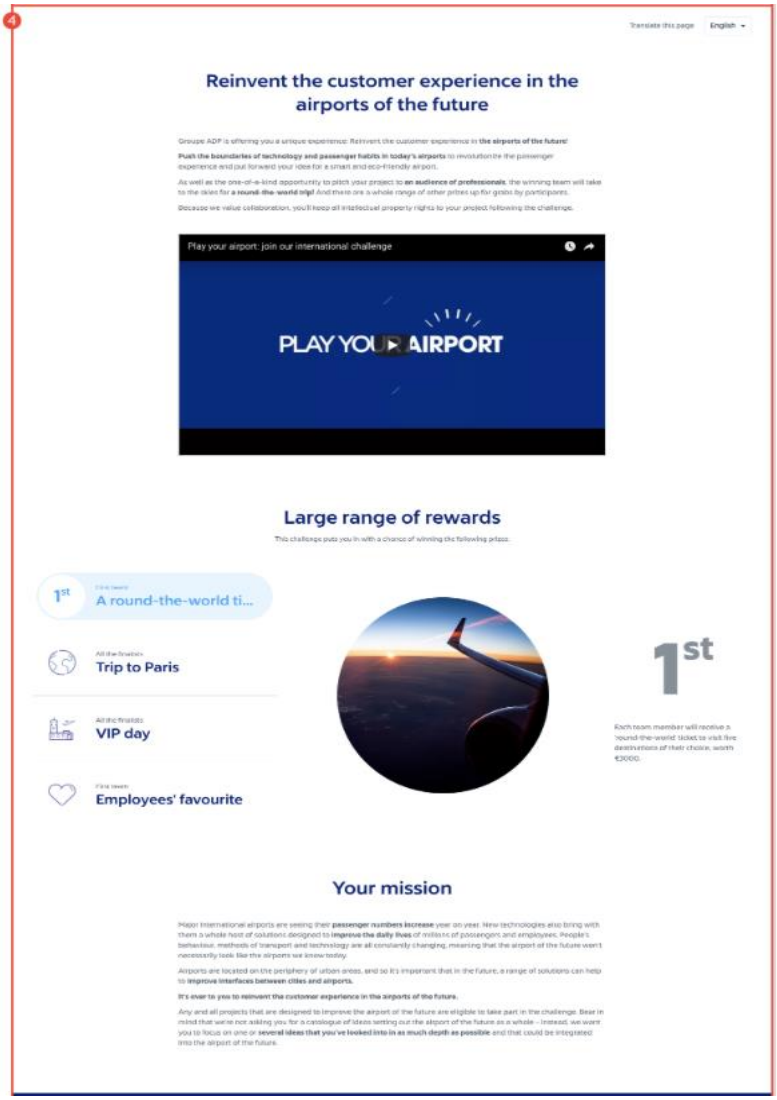

Figure 4.8: Presentation of a typical brief page of a challenge

In the final part of the display of a challenge is the footer of the platform, as shown in figure 4.9, where the terms of service and the privacy policy of the platform are found and cannot be customized if the competition is published on Agorize.com. This is also where you can change the language of the platform. If, on the other

hand, the challenge is published on the customer's website, it is possible to customize the terms of service and the privacy policy according to needs.

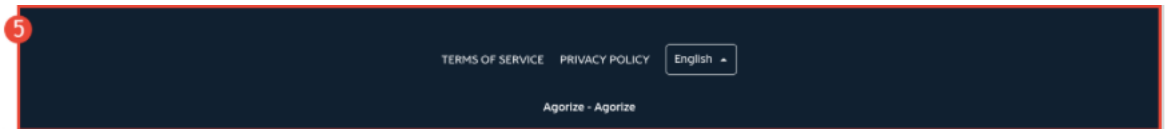

Figure 4.9: Footer of a challenge.

4.3 Creation of a challenge

Each project begins with the kick-off meeting between the client and the PMs, during which the understanding of the client's requests and the evaluation of the feasibility of the project takes place, where PMs try to understand the challenges and problems of the clients, advise them and offer them a tailor-made operation that best meets their needs.

After the evaluation and acceptance of the requests, the PMs contact the EPMs to start the configuration of the back-end parameters of the platform (URL, Challenge name, creation of the Challenge platform, the configuration of the platform, etc.).

If customers do not choose the Saas service, the PMs together with the customer decide in more detail what the theme of the challenge will be, to whom it is addressed, the timing of the various phases, if you want to create a landing page to make more attractive a challenge and how the PMs plan to structure the layout and the display, showing examples of previous competitions already made.

Once the details have been agreed upon, the PMs begin to create the layout, the display, and the contents of the challenge (Content, Rules, etc.) and the final result is given by the elements presented in the paragraph 4.2. At the same time, the customer is trained to use the platform.

Given that PMs are not experts in HTML and CSS, the computer programmers have created a challenge builder within the platform that automatically creates, without programming, some HTML, CSS, and Javascript blocks, called standard

blocks, often used in the creation of competition. The standard blocks are presented in figure 4.10.

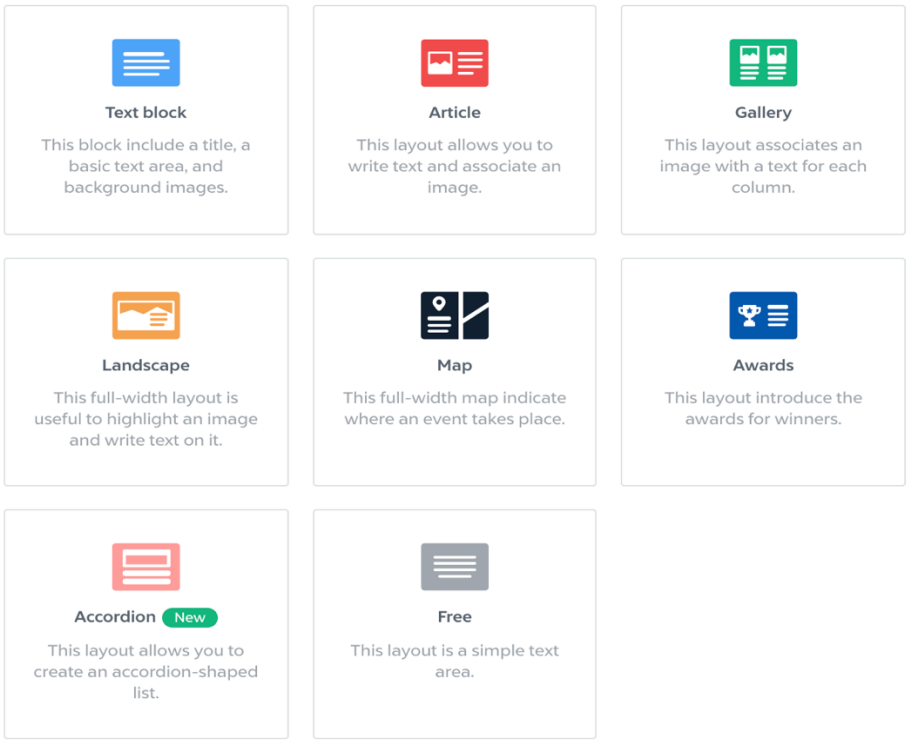

Figure 4.10: Screen of the standard blocks in the Agorize platform

As it's possible to see, there are 8 standard blocks. The first is the Text block which allows you to insert simple text within pages. Then there is the Article block that allows PMs to insert text alongside an image as shown in Figure 4.11. Usually, this block is used to present the company, placing the logo next to the text.

Figure 4.11: Example of a standard Article block.

The Awards block instead allows to create a dynamic menu to view the rewards for each position reached as shown in figure 4.12.

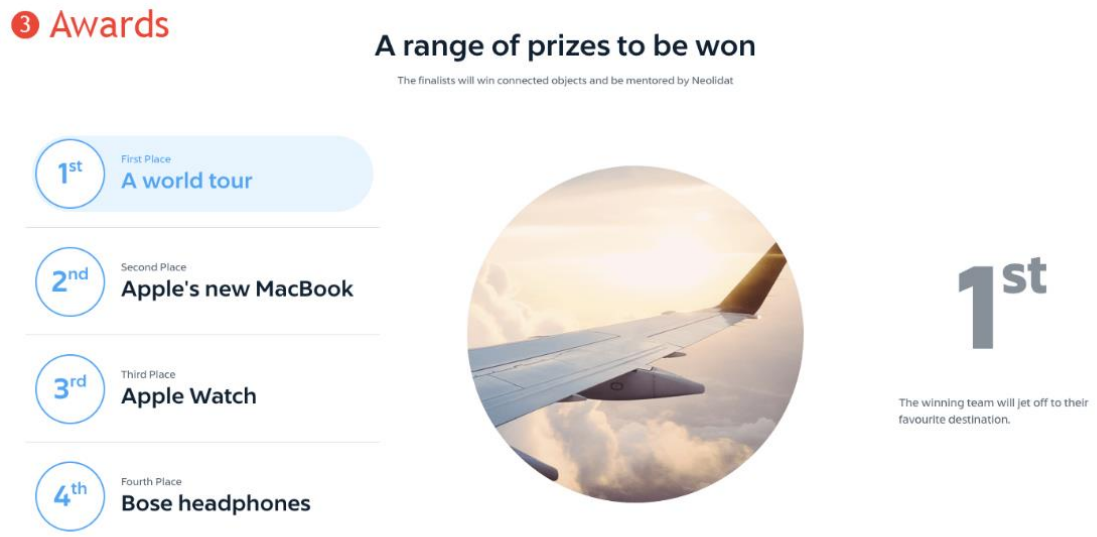

Figure 4.12: Example of a standard Awards block.

The Gallery block allows to insert cards with relative images and text as shown in figure 4.13. It is usually used to show all the advantages of a participant in participating in a challenge or to detail the theme of the competition.

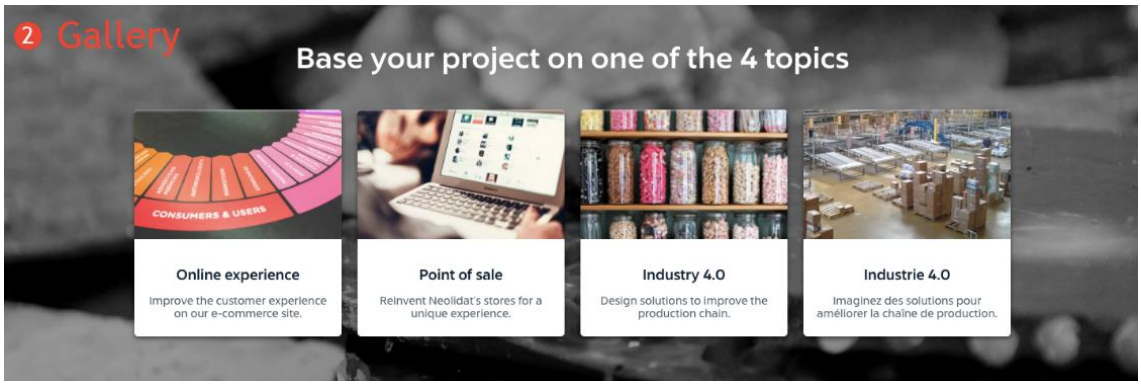

Figure 4.13: Example of a standard Gallery block.

The Accordion block is always a dynamic menu in which it's possible to click on one of the Accordions to view the information within it and is shown in more detail

in figure 4.14. It is usually used when several partner companies have organized the challenge and you want to enter information about each of them, but at the same time, it saves space and allows the user not to scroll too much the page.

Figure 4.14: Example of a standard Accordion block.

Then you have the Landscape block that simply allows to insert a background image to the page and the Free Block that opens a free HTML, CSS, and Javascript programming environment in case you don't want to use any of the other standard blocks and then customize the challenge. Of course, the standard blocks create responsive HTML, CSS, and JavaScript structures that are adapted to any screen size.

Once the challenge has been created, the final step is to send the prototype for customer review and any ordinary changes to be made will be resolved within 1-2 working days. As regards any special unscheduled requests, the PMs evaluate the commitment and time to implement them and reserve the right to accept or decline. Sometimes this type of request is interesting for a long-term improvement of the platforms and then they are sent to the EPMs, which together with the IT developers evaluate whether to include it in the improvement and update plan of the platform.

Meanwhile, the Community team prepares to promote the challenge to the target communities by defining a communication strategy.
After acceptance of the challenge prototype, the project moves on to the production area that can publish the challenge on the web (the challenge can be published on the Agorize platform or a specific customer website as needed).
Once the challenge has been launched, the PMs are responsible for ensuring the smooth running of the challenge, ensuring the reporting of communications to the customer, and following the different selection phases. At the end of the challenge, an awards ceremony is organized. The PMs lead the organization of the closing event together with the client, defining the modality of the event and ensuring its proper organization, in compliance with the expectations of customers and participants.
If, on the other hand, customers choose the SaaS service, the work of the PMs consists only of training the customer to use the platform. The PMs illustrate the whole platform, the various user permissions, and also the standard blocks to create the challenge display. From here on, the customer manages the various stages of the challenge independently. After having trained the client, the PMs and EPMs are no longer involved in the competition implementation process, but the PMs remain the first point of contact for customers for any questions and problems they may encounter in organizing the challenge.

4.4 Definition of the problem

The purpose of this paragraph is to define in detail the problem that has been faced, presenting the initial situation also called as-is analysis, and the goal to be achieved also called to-be analysis, i.e. the new condition in which the company will have to find. Besides, the project risks will also be defined and how they are tried to control.

4.4.1 Website design

Successful web designs are the key components to attract users as websites have become one of the most important parts for running a business, maintaining customer relationships, spreading business, shifting costs, moving of space, etc. illusory. (Robbins, 2012)

To begin with, a website's user interface is what visitors use to interact with the website. Visual design is a component of the user interface and deals with the aesthetics of a website. It includes parts like layout, color, typography, spacing, and other aspects of a website's appearance. The visual design consists of choosing the color palette, font style, the shape of a button, and file spacing between each element, etc. which make the website aesthetically pleasing.

Another determining factor in user interface design is information architecture. Information architecture is how website content is rendered in a logical way that will not force users to spend a lot of time or effort finding what they are looking for. The layout frame needs to be carefully planned to be able to achieve this. Information architecture is a crucial factor for a website to be considered a high-quality product because it makes navigation easier and more intuitive. Additionally, information architecture has a huge impact on the user experience as the way the layout skeleton is built will affect how users use the website.

Obviously, the way the content is presented visually affects how people perceive it. In most cases, an User Interface (UI) is the first impression web users have of the website, therefore, a bad UI design immediately pulls people away from the website.

4.4.2 As-is analysis

Landing Pages are the pages that introduce the challenges, and which are designed to convert web visitors into challenge participants. They are different from other

pages because they must be attractive and at the same time explanatory and easy to use, to convince web visitors to participate in the challenges.

Agorize does not have specific figures hired only for the creation of Landing pages, as front-end developers, because not all customers require them. Typically, these pages are mainly requested by important and large customers such as L'Oréal or Schneider Electric because they are willing to spend more. So, on the one hand, it could be very important to have a specialized figure to create an attractive landing page and establish a long-term relationship with the most important customers but on the other hand, it could be a waste because there could be a period without needing them.

Furthermore, as explained in paragraph 4.3, the displays of the challenges is mainly done by PMs who do not have a deep knowledge of HTML, CSS, and JavaScript and are helped by the standard blocks inserted in the platform. The problem arises when they need to build a more specific structure that cannot be built from standard blocks.

The company did not have yet a clear strategy and the process of creating the Landing Pages and challenge displays varies according to the various Agorize offices located around the world (Asia or Europe).

As of the creation of Landing Pages, there was not a standard process in Europe, and it varied greatly depending on the situation. After the kick-off meeting between PMs and client, it is defined whether the Landing Page will be created following a draft decision by the client or if the realization is left to the client. In the first case, if the customer's needs are fairly easy and basic and without complex elements to create, the Landing Pages were created directly by the PMs, even if in the programming environment of the Landing Pages it is not possible to use the standard blocks. Most of the time, when the client's requests are too complex for the PMs, the implementation is entrusted to the EPMs who have a good knowledge of the front-end languages, but the risk is to have too high a workload on them. In fact, the numerous tasks to be carried out to improve the efficiency of the platforms

and solve any bugs carried out by the EPMs risked greatly delaying the creation of the Landing Pages which, to respect the tight and rigid deadlines, could be carried out quickly and not guarantee the expected quality. Sometimes, when the requirements were too difficult for the PMs and the EPMs realize that they could never meet the deadline set by the client due to too much work on them, the company decided to outsource the work to a company, but the costs increased a lot. Figure 4.15 summarizes the typical process of creating Landing Pages in offices in Paris.

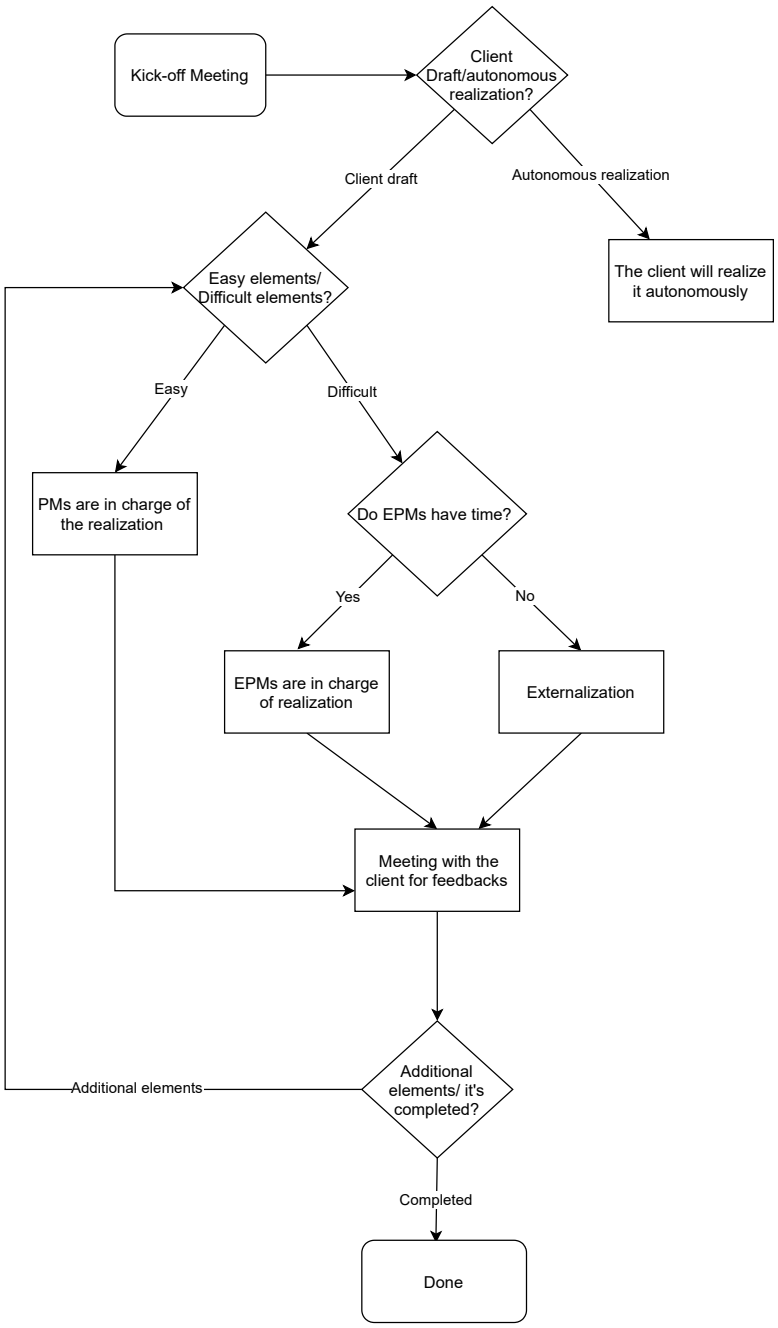

Figure 4.15 Workflow for the creation of Landing Page in the Agorize offices in Paris.

In the Paris offices, the communication between PMs and EPMs takes place mainly by voice and in an informal way, as the EPMs have to understand if the requests are simple enough to be carried out exclusively by the PMs, or if they could create the page or entrust it to an external company.

In Asia, the process of creating the Landing Pages is completely entrusted to the EPMs both in the case of a client's draft and in the case of autonomous creation since they have less workload and can take advantage of their in-depth knowledge of front languages.

For the creation of the display of the various internal pages of a challenge, here too there is a division of the work between PMs and EPMs. In this case, after the kick-off meeting between the client and PMs in which they establish rules, prizes, and themes, it is the PMs who propose a display prototype, using above all standard blocks. Subsequently, the prototype is shown to the customer who can accept it, propose to modify it, or reject it, and if the proposed modifications include elements that require in-depth knowledge of HTML, CSS, and JavaScript, the implementation is entrusted to the EPMs. Figure 4.16 below summarizes the process of creating the display of the pages of a challenge.

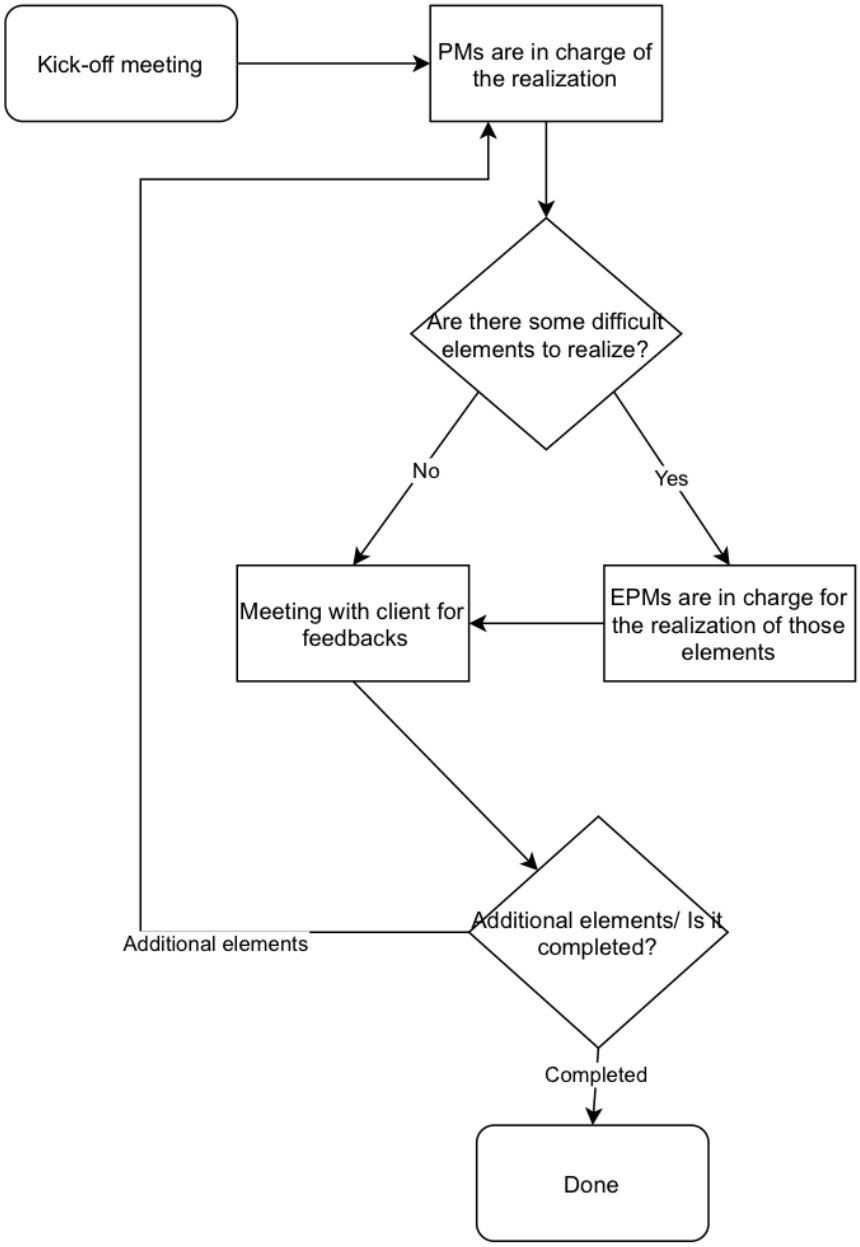

Figure 4.16: Workflow for the realization of the display of a challenge in the Agorize offices in Paris and Asia.

The communication between PMs and EPMs takes place via Jira tickets, described previously. In case there are HTML, CSS, and JavaScript structures that are too difficult to implement, a ticket is created for the EPMs in which the problem is usually described, a PowerPoint draft it is attached and the URL where the graphic

elements must be created. EPMs receive the ticket instantly but will only be processed when priority tasks are completed. When the element is created, the EPMs can insert the solved option on the ticket and instantly a notification arrives at the PMs that it has been solved. In this case, the process is the same both in France and in Asia.

The lack of standardization has led to different qualities for the Landing Pages and the challenge display depending on how and where they were created. The problem of different quality can have important consequences because there may be customers who are extremely satisfied with the work while others may be seriously disappointed and thus become future lost customers. Among the most important consequences of a lack of standardization are:

- High time for launching competitions or for making Landing Pages: in addition to the extra time required for communication between PMs and EPMs, PMs are not well prepared and take more time than an expert. Furthermore, when the implementation is entrusted to the EPMs, it is necessary to wait for the EPMs to be available of their work, and therefore usually this does not happen instantly;
- High response time to change or customer feedback: when a prototype is presented to the customer there may be feedback or change propositions and often there are high response times due to the same reasons described above. PMs feel comfortable with standard blocks but for more customized elements they can have many difficulties and EPMs could be busy with other activities;
- Difficulty in attracting new customers: sometimes potential customers want to evaluate and test the company's skills and require the creation of the Landing Page before deciding to purchase Agorize services. Any page that does not meet their expectations and that disappoints them could cause the

loss of the potential customer who may decide to contact Agorize competitors;

- High cost of user acquisition: even if potential customers are satisfied and convinced to work with Agorize, the cost is high because, as explained above, the time taken is excessive and the costs for any outsourcing are high;
- Low customer loyalty: even if customers decide to contact Agorize, they may be disappointed by how the challenge was presented or by the fact that there may have been few participants and may decide to organize subsequent challenges with Agorize competitors.
- Different quality: as regards the implementation of Landing Pages, in Asia, unlike in Europe, the process is completely entrusted to EPMs who have a thorough knowledge of front-end languages. So the quality of Landing Pages created in Asia is better than those created in Europe.

4.4.3 To Be analysis

If there are discrepancies between the As-Is analysis and the strategic objectives, a new state or a new condition must be defined which must be formalized and defined in detail. This will be the To-Be analysis, which is the new condition in which the company will have to find itself to comply with its objectives (Dennis et al., 2012).

Agorize's objectives are to standardize Landing Page creation processes and challenge displays, to guarantee a high quality of competitions and high reactivity to change customer requests, through the transmission of knowledge to all company levels, a centralization of the most common practices and continuous attention to graphic excellence. As for the creation of the challenge display, the transmission of knowledge and the centralization of practices have the aim of making the PMs autonomous for each type of request from the customer and using

EPMs as little as possible. All this can be possible through the creation of a platform called a library, in which all the HTML, CSS, and JavaScript elements created in the past, even the most complex, are inserted. In this way, the process would be completely revolutionized in France and Asia.

The new process begins with a meeting between the client and the PMs to define the pages that will constitute the challenge. After a first realization by the PMs in which most elements are added thanks to the standard blocks, the customer can accept the final product or make proposals for changes. In this case, the implementation is again completely entrusted to the PMs who, if complex elements are requested, can consult the library made available without contacting the EPMs. In this way, there will be an important filter in requests to EPMs, which will be called only if the graphic element to be added is not present on the platform.

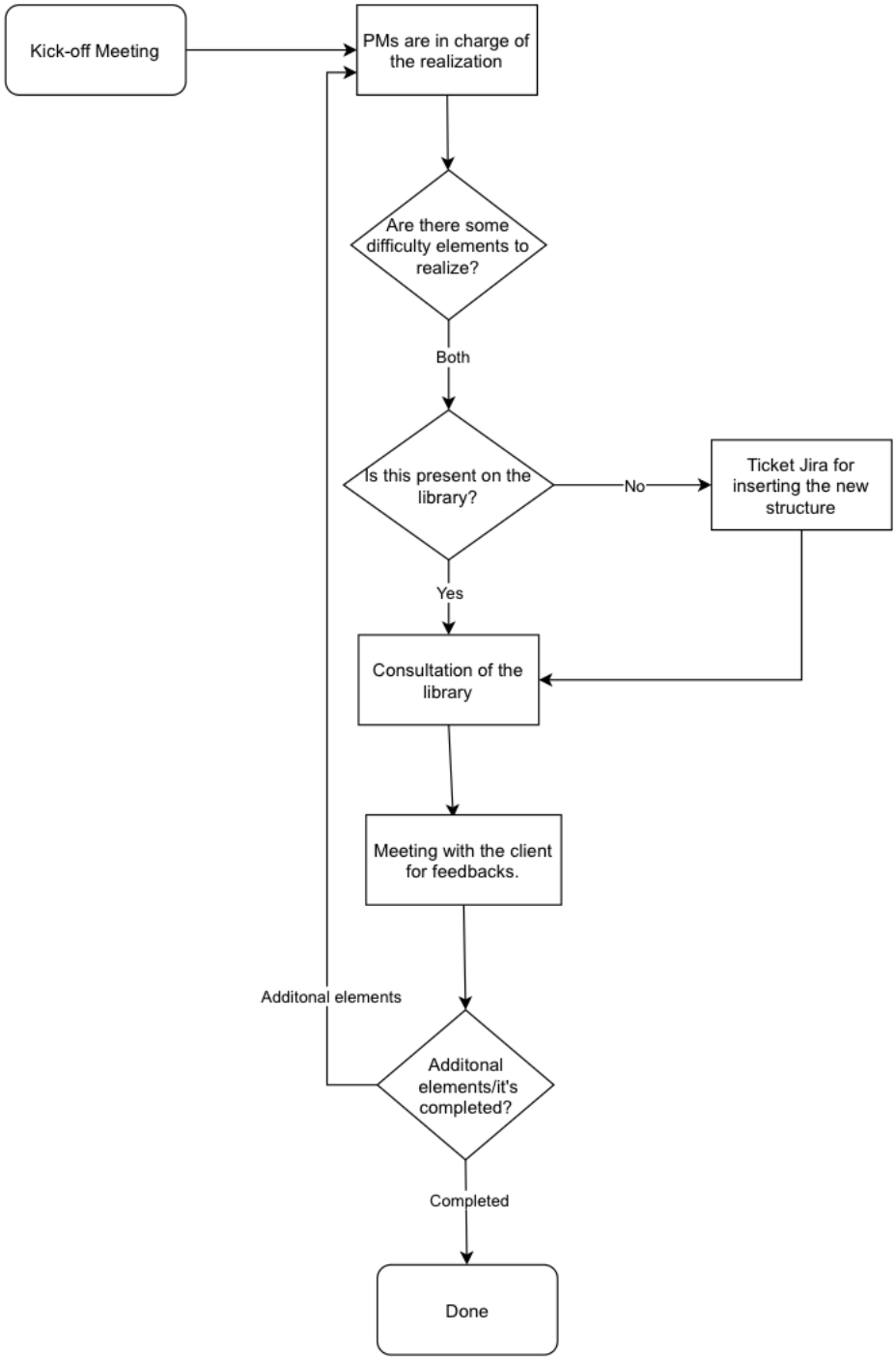

Figure 4.17: Workflow for the creation of the display of a challenge to be carried out.

As for the Landing Pages, it has been decided to exclusively address the EPMs, in particular those in Asia that have less workload due to the fewer requests for

challenges and the fact that it is mainly the EPMs in the Paris offices that deal with the improvement of the efficiency of the platform, since their realization requires a deeper knowledge of HTML, CSS and JavaScript languages since it is not possible to use standard blocks and the customer can require a high degree of customization. Even in this case, however, the EPMs are helped by the library, greatly reducing the creation time since many of the structures created to date and also all the Landing Pages created to date are inserted into it.

After the kick-off meeting in which PMs and client talk about the challenge and the eventual realization of the Landing Page, two different situations are possible: the customer creates a prototype of the Landing Page with a PowerPoint to fully customize the launch page, a draft, or the customer leaves the total realization to Agorize. In both cases, the Asian EPMs have the task of satisfying the customer's requests. To simplify the work of the EPMs as much as possible and reduce the implementation time, EPMs can consult the library. In the event that there is the presentation of the draft of the Landing Lage by the customer, elements can be taken from the library while in the event the implementation is completely entrusted to Agorize, the EPMs can consult, through the library, all the Landing Pages made previously, take inspiration and use all the more complex structures according to the customer's expectations.

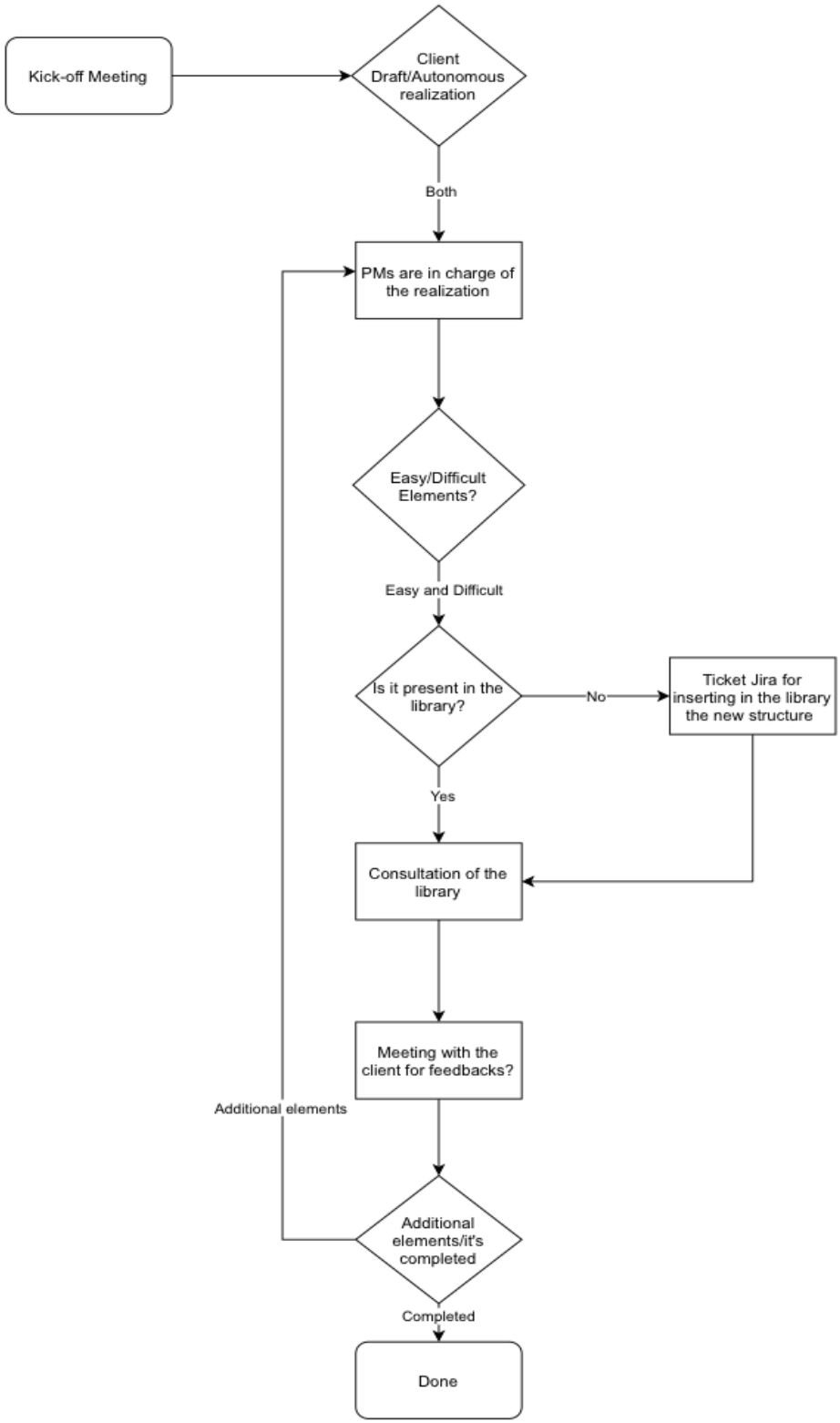

Figure 4.18: Workflow for the creation of the Landing Pages to be obtained.

4.4.4 5W+1H

The "5Ws and How" model is usually used in journalism and research to gather any information needed to carry out a comprehensive analysis of a given problem or challenge (Hart, 2002). Within the project, its main purpose is to explain, in addition to the stakeholders involved, how it has done and how Agorize wants to standardize and make more efficient the process of creating a challenge display and landing page in all its offices around the world.

What - What is the goal of the project? The goal is to redefine and standardize the process for the creation of Landing Pages and the visualization of challenges to guarantee an equivalent and satisfactory level of quality.

Who - Who is involved in the project? One of the activities that include the role of EPMs is to improve business processes to ensure the efficiency and quality of web platforms. Therefore, the EPMs are the main players in the organization of the project. Furthermore, the EPMs will also benefit from the effects of the project since it tends to overload them less with work, trying to involve them less in the implementation projects of the Landing Page and display of the challenges. The PMs will be the major beneficiaries of the project as it will tend to make them autonomous in the implementation of the challenge and will have to follow a standardized process.

Why - Why this project? The confusing and different processes for each Agorize site have led to the creation of challenges and Landing Pages with completely different qualities since it varies depending on where the implementation takes place and by whom. Poor quality could have major consequences for Agorize, as it can cause the loss of many both new and loyal customers (poor loyalty). The cost of implementation is high since the implementation time is greater due to the lack of clarity about who has to create the Landing Page. In addition to this, the lack of clarity and standardization leads to the outsourcing of some tasks that greatly increase the cost of implementation.

Where - Where is this project carried out? The project is carried out in the Agorize headquarters in Paris but involves all the offices in the world as it tends to standardize a process throughout the company.
When - When is the project realized? The project is implemented from the beginning of October 2020 until the first months of 2021.
How - How do you plan to reach the goal? The transmission of knowledge and the centralization of practices are the key to the standardization of business processes. The aim is to create a web platform, called a *library*, where all the HTML, CSS, and JavaScript knowledge of the EPMs can be contained and made available to the PMs. Furthermore, the platform will contain all the Landing Pages of the past to take some structures and create other equally captivating ones.

4.4.5 Risks of the Project

Process innovation is a real and difficult challenge in any organization. Several elements of complexity and uncertainty make a "rational" approach not so complete to tackle a problem involving people, technology, and organization. The integration of different tools, techniques, and technologies is particularly useful when innovation implies a good knowledge of the whole system and its decision-making and operational contexts.
Agorize has been organizing challenges for more than 10 years and most PMs have their own routines for the implementation of challenges. However, the tacit condition of the knowledge incorporated in the routines implies not a few problems, such as invisibility to management. In other words, although the role of management also consists of the responsibility to control and supervise the work of workers, it is often difficult to carry out these activities as tacitness does not allow to identify and concretely define the routines performed. Tacitness is understood as the inability to verbalize the rules followed for action.

This leads to an unrealistic view of the business situation (Nelson and Winter, 1982). The invisibility of the routines, caused by the fact that often the person who has to supervise the process is foreign to certain levels of such meticulous analysis, makes some decisions useless and often wrong because they are oriented elsewhere, on other activities. In light of what has been said, therefore, most of the choices made for the execution of a process are taken by those who perform the tasks, often without having an overall view of the situation and with the unawareness of having made a decision, because by operating without actually knowing the possible alternatives and the consequences that (implicit) decision could have on the whole process or on the organization itself. In the same way, management, not directly involved in carrying out the tasks, is not aware of the decisions implicitly made without its knowledge. So, the new library may not enter the PMs routine, making its implementation useless.

A possible method to control this risk is given by the ticketing system that manages the communication between PMs and EPMs. If the number of tickets created by the PMs to implement those complex structures does not decrease, the EPMs could respond by attaching the URL of the library platform to the ticket, advising to consult it. In this way, it would be possible to get the consultation of the library into the daily processes of the PMs.

As will be explained later, the library will be implemented following an agile approach. So anothere risk is to introduce an Agile methodology among team members who have never used it. There is a risk that team members, with IT training, do not grasp the importance of management activities, which could be perceived as time stolen from programming, and then these are not applied at full capacity. Instead, they must understand how a few minutes of planning can avoid many hours of programming, the importance of intra and inter-team communication for the quality of the product, the importance of unit and integration review and retrospective activities lead to concrete actions that improve everyday work.

A final risk is to leave the library incomplete. In fact, by not introducing many of the templates that are used by PMs or EPMs, the platform risks not being useful for the implementation of the challenge and not entering the mechanisms of the new process.

4.5 Library

The project aims to make it easier to transmit knowledge to multiple team members and therefore to centralize best practices to ensure homogenization of the quality of the challenges created. The tool that was thought to guarantee this centralization is a *library* that contains all the HTML, CSS, and JavaScript structures most used during the construction of competitions or Landing Pages.
In paragraph 4.1 the platforms in which Agorize implements the challenges were introduced. The library was created on a platform on which one or more challenges were usually created for a client. In this case, instead of the challenges, all the blocks, also called templates, are inserted with related HTML, CSS, and Javascript codes that can be copied and useful both for the creation of the Landing Pages and for the implementation of the displays. In addition to the various templates, a block has also been included with all the Landing Pages created to date, to be able to take inspiration if the realization is left autonomous to the EPMs or to show the customer during the kick-off what Agorize can do, convincing them to create a Landing Page similar to one of those already created or even to urge the customer to create one if he is not convinced. Besides, a page has also been included in which the result of the standard blocks is shown to facilitate their use by the PMs. Each graphical structure, even called template, will contain various elements divided according to the difficulty of the code used to implement them ("Low", "Intermediate" and "Advanced"). For example, the timer element that signals the expiration of a phase of the challenge can be implemented in different ways, according the complexity of the code. Besides, a section will also be inserted in

which all the Landing Pages made in the past will be collected, and also in this case, there will be a division of them based on their quality.

The library will be used every time the PMs have to build a challenge or the EPMs a Landing page. PMs will have the possibility to choose between the various graphic elements present within it, decide which is the most appropriate for a challenge, copy and adapt the HTML, CSS, and Javascript code in the challenge that is being implemented. As for the construction of Landing Pages, the EPMs as well as having the ability to replicate the graphic elements present in the library will be able to consult all the previous Landing Pages delivered to customers to take inspiration.

In figure 4.19 it's possible to see some of the various blocks to which the user has access and from which he can extrapolate the code of interest. Besides, within each block or page, several sections divide the graphic elements according to the difficulty for implementation. In the next paragraph, the topic will be further explored.

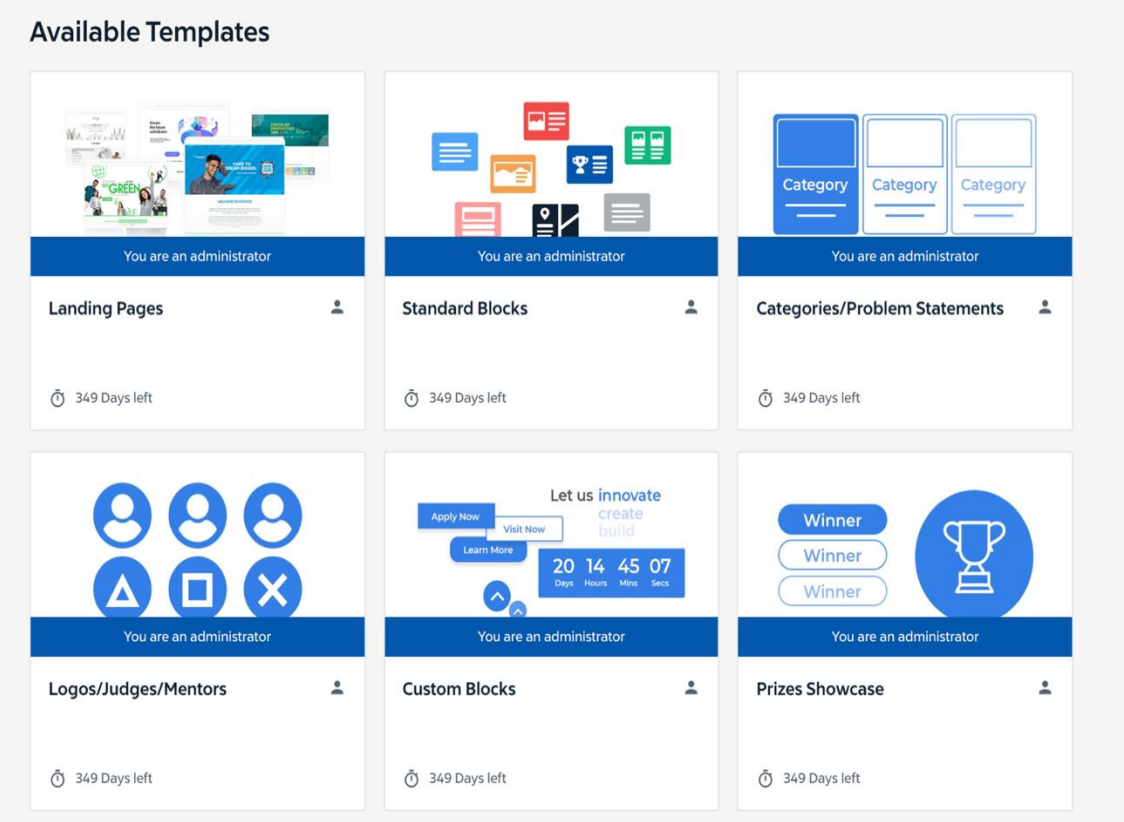

Figure 4.19: Schematic with some of the templates available in the library.

4.5.1 Landing Page block

All the Landing Pages created in the past by Agorize have been included in the Landing Pages block. The pages have also been divided according to the difficulty of implementation. The difficulty ranges are 3: Basic, Intermediate, Advanced. To evaluate the quality range in which each Landing Page is inserted, 9 most common graphic elements (such as Videos, Images, or Timeline) have been defined that are present within the Landing pages. Among these elements, some are easy to implement at the code level because they require a basic knowledge of HTML and CSS, while others are more complex because, in addition to requiring a basic knowledge of HTML and CSS, they also require knowing how to program in JavaScript. If in a Landing Page there are only basic graphic elements, that is, they require HTML and CSS then the Landing Page is classified as Basic. If on the

other hand, in addition to all the simpler elements, the launch pages include a complex element (with JavaScript), they are classified as Intermediate. If in addition to all the simple elements it contains two or more complex elements, then it is classified as Advanced. As you can see from figure 4.20, in the left Landing Page card, the basic one, there are only three elements, Images, Videos, and Buttons to redirect the page, in the one in the center, which is *intermediate*, the Timeline is also added and in the *advanced* one there are more elements complex. Also, as you can see, two buttons redirect the user to the Landing Page view or to the page where you can see and possibly copy the code.

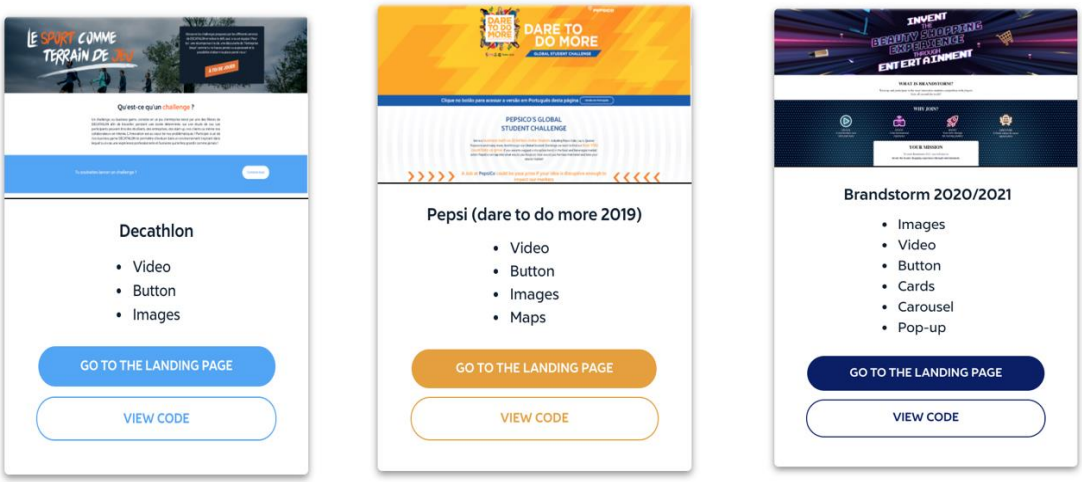

Figure 4.20: Screen inside the Landing Page block.

4.5.2 Templates

Inside the library, all the blocks have been inserted that contain the HTML, CSS, and Javascript structures or templates most used in the past to create the challenge displays or the Landing Pages such as the Timeline to clearly display the time left at the end of the current phase and at the beginning of the next one, or the Prizes to show what the competition prizes are. Each block with the graphic elements inside it is divided into *Basic, Intermediate, and Advanced,* as shown in figure 4.21, depending on the complexity of the code. For example, in figure 4.21, the

example of the Timeline is shown, in which the basic structure is a simple image that does not require knowledge of programming languages, while the Intermediate part is made entirely with HTML and CSS and the Advanced one as well as with HTML and CSS it is also realized with the use of JavaScript code. In this way, in the moment of the realization of the challenges, the PMs can evaluate which element to insert according to the customer's needs.

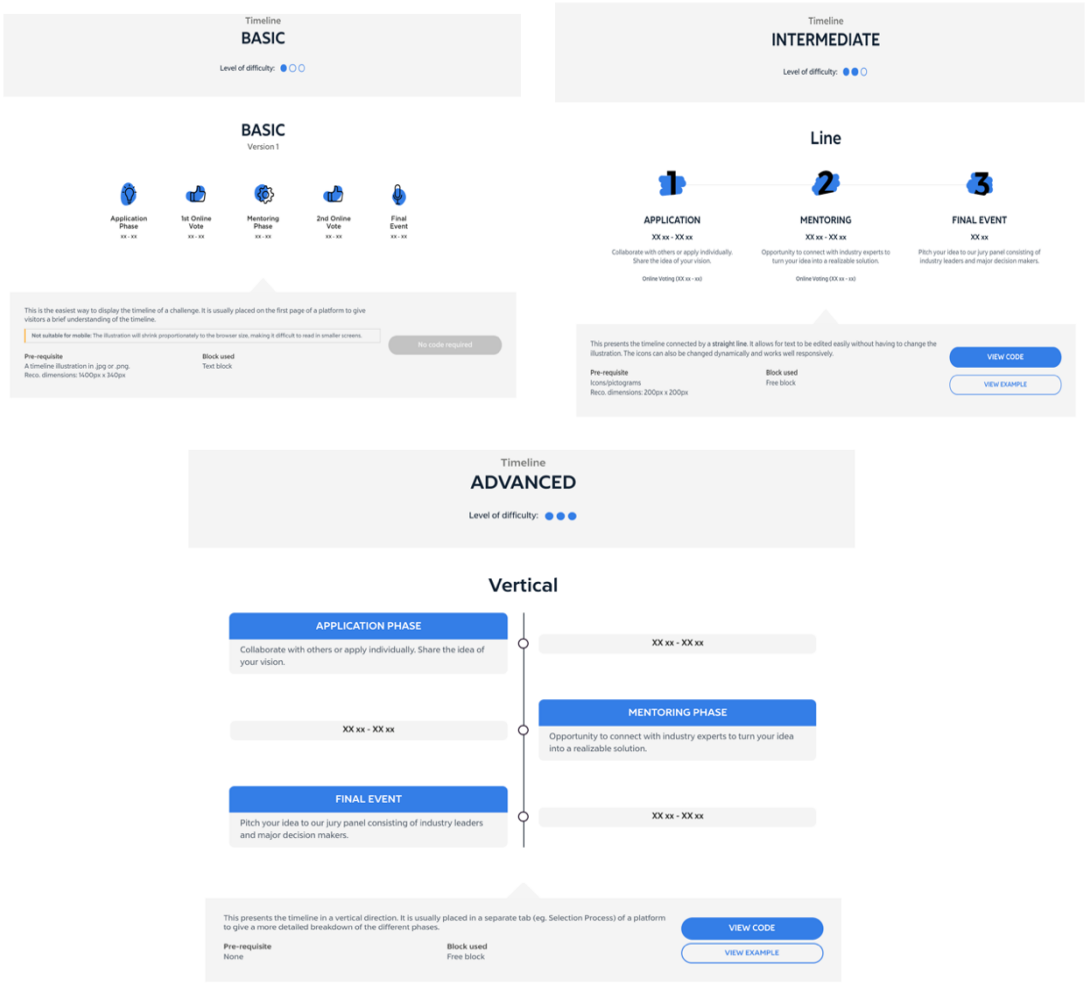

Figure 4.21: Examples of Timeline with different levels of difficulty.

4.5.3 Library Landing Page

To facilitate the use of the library and make it easy for future Users to use, it has been decided to create a Landing Page that allows users to better understand the functioning and structure of the library and from which they can access it. In fact, as for all challenges, to access the contents it is necessary to create an account or if you already have the credentials a login. Unlike the other challenges, however, it has been decided for the moment to keep the library platform for internal use by Agorize and not for all. Although it is not excluded that in a possible future this platform may also be used for other purposes, such as providing help in the realization for SaaS customers (customers who purchase the right to use a platform but who do not receive any service in the realization and distribution) at an additional price or to be used by the Sales & Marketing department to propose different prices depending on the difficulty of the graphic elements used.

As shown in Figure 4.22, the Landing Page introduces the User to enter the platform and explains the functioning and organization of the platform. There is the "Find Your Block" section in which the graphic elements that have been inserted are explained. Then there is the "Based on Difficulty" section where the subdivision according to the difficulty is explained and then the "Use the Code" section which shows how to copy the code. Finally, there is the section that introduces the page with all the Landing Pages which can be accessed directly.

Figure 4.22: Landing Page of the Library platform.

4.6 Library implementation

The agile approach offers to maximize flexibility during the project, trying to "crystallize" the new product at the last possible moment, to react promptly to a change in customer requests or, more generally, the market.
The choice of the Project Management methodology to be used is dictated mainly by one factor: the need to guarantee a short time to market. Short times to market allow the library to enter briefly into the processes and the corporate mentality because short time to market ensures new features in the shortest possible time.
There are therefore two main constraints of the project:

- the need for a short time to market to stay up to date with customer requests
- the lack of maturity of the context which leads to uncertainty of the requirements.

The uncertainty of the context and the short time to market are better suited to an agile Project Management methodology. For the development of the project, it has been decided to adopt an Agile logic, breaking down the process into short and iterative cycles, using the Kanban framework within the Scrum methodology, to exploit and also to enhance the specific, but also transversal, skills of the resources involved in the project team, to involve them from day one and maximize their productivity.
The theoretical basis on which the scrum ceremonies are based is the PDCA approach, known as the "Deming wheel": that is the concept that 4 types of activities must be carried out cyclically, the "Plan", the "Do" that is actually carry out the planned activity, the “Check” that is the control of the activity carried out, and the “Act” implementation of both the corrections of what was detected in the check and of the process improvements.

4.6.1 Agile team

4 members of the Agorize team are involved in the design and implementation of the library project: 3 EPMs, of which 2 working in the Paris offices and one in the Honk Kong offices, and a PM who usually deals with projects for Agorize process improvement.

In the case of the library, the team figures are:

- Scrum Master: represented by the PM included in the project. He is in charge of organizing the weekly meetings, checking the status of the work and he is the one who is in contact with the project stakeholders to communicate each new version of the platform.
- Product Owner: role held by the most senior EPMs of the offices in Paris. He is the reference point of the work team and he can have decision-making power in the execution of the work. In fact, it is he who decides the graphic elements to be inserted, collecting feedback from external stakeholders (PMs) and based on the knowledge acquired over the years. The PO in this case plays the dual role of programmer since he also has a programmer background but he has also organizational and motivational skills.
- Development team: represented by all the EPMs involved in the project and also the oldest EPMs of the offices in Paris. Their role is to insert all the HTML, CSS, and JavaScript structures that are decided every week within the library. The main peculiarity of the Agile group is the self-organization of the team; members are not specifically instructed on how to proceed with product development but are given responsibility for making decisions.

Among the areas on which Agile focuses is that of communication and collaboration between the members of the project team. In fact, an interdisciplinary horizontal communication between the different areas working on the project is privileged, composed of members who know each other personally and who have the opportunity to interact constantly online or physically.

4.6.2 Sprint

Among the main features of Agile programming is the breakdown of processes into short iterative cycles with a duration ranging from 1 to 4 weeks, depending on the type of project. In the case of the project for the creation of the library, a week was decided to guarantee more frequent communication and increase the awareness of the individual members of the group on what the others are following, making it easier to identify the different responsibilities and also facilitating the control of any risks. During these weekly meetings, it is established what was accomplished during the week and what will be achieved in the following week. Sprint is the name given to the iteration, which takes place over a week, on which the team works. This is a fixed period of time in which the team creates and releases a working library increment. During this period the team autonomously carries out all planning, development, and testing activities.

Every week there is a one-hour meeting in which the following week is planned and the fundamental characteristics of the methodology are highlighted such as: prioritizing activities based on value, assigning tasks to the team in the correct way, not being distracted by activities outside of the backlog and review the work done. At the weekly meeting, to review the problems encountered in the previous sprint, all team members are always present, some of whom interacted directly with stakeholders to immediately give evidence of compliance with the established criteria of what had been implemented.

At the beginning of each weekly meeting, a series of tasks are identified by the Product Owner and are added by the Scrum Master in the To-Do column. Subsequently, during the meeting, the activities that pass in Work In Progress are established depending on the availability and time of the Team Members. At the end of each meeting, an updated version of the library is released with new graphic

elements inside that can be used and exploited by PMs and EPMs in the construction of Landing Pages and Challenge displays.

Good communication must be organic, clear, and concise. However, communication is not to be understood only as a means of exchanging information on the progress of the project, but also the management of documents, the collection of lessons learned, communication and data management techniques, as well as the identification and management of stakeholders from a communicative point of view.

The internal developments of the functionality lead to additional complexity: the members of the development team are located in two different cities and coming from different corporate cultures must jointly develop some functionalities. The complexity, in this case, is that relating to the organization of work and the implementation of adequate communication channels. To ensure the effectiveness of communication flows, in addition to the weekly meeting with all project members, weekly meetings were also organized but only with the development team lasting about 45-60 minutes. The purpose of the meetings is to update on the progress of activities compared to the previous meeting, to report any critical issues, and to address their resolution.

The initial plan was to release each new version weekly but graph 4.23 shows the reality. In the graph on the x-axis, there are the dates when a new version is released and the version number, and on the y-axis, there are the sprint needed to move to the next version. It can be seen that only in versions 2,3,5 and 7 the constraint is respected. The adoption of a Scrum methodology mixed with Kanban allows not to deliver a new version every week and the fact that being an internal project that did not require a particular urgency since there were no strict deadlines, led to release times much higher than expected.

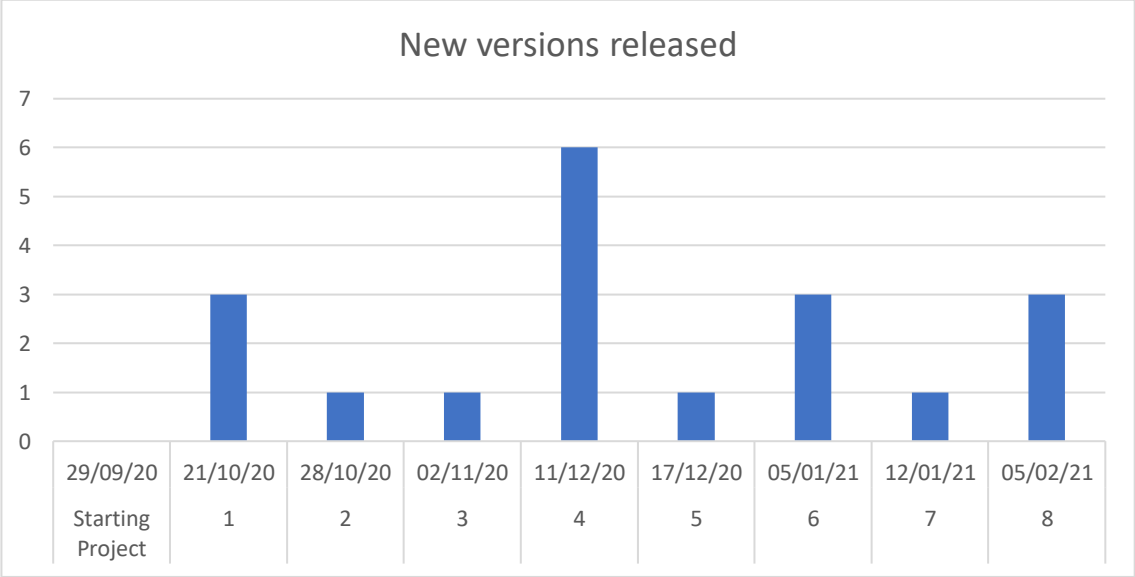

Figure 4.23: Date of new version of the library realesed.

Also concerning the meetings, there has been a change from the initial planning. In fact, the project manager who had the role of scrum master in the project, after 17/11/2020, as it is shown in figure 4.24 where meetings with all the teams and meetings within the development team are compared, stopped organizing meetings for several reasons, one of these being that of having completed his role as a teacher of the methodology. In figure 4.24 on the x-axis there are all the meeting days and on y-axis if it is equal to one, it means that there was a meeting and if it is equal to zero there wasn't.

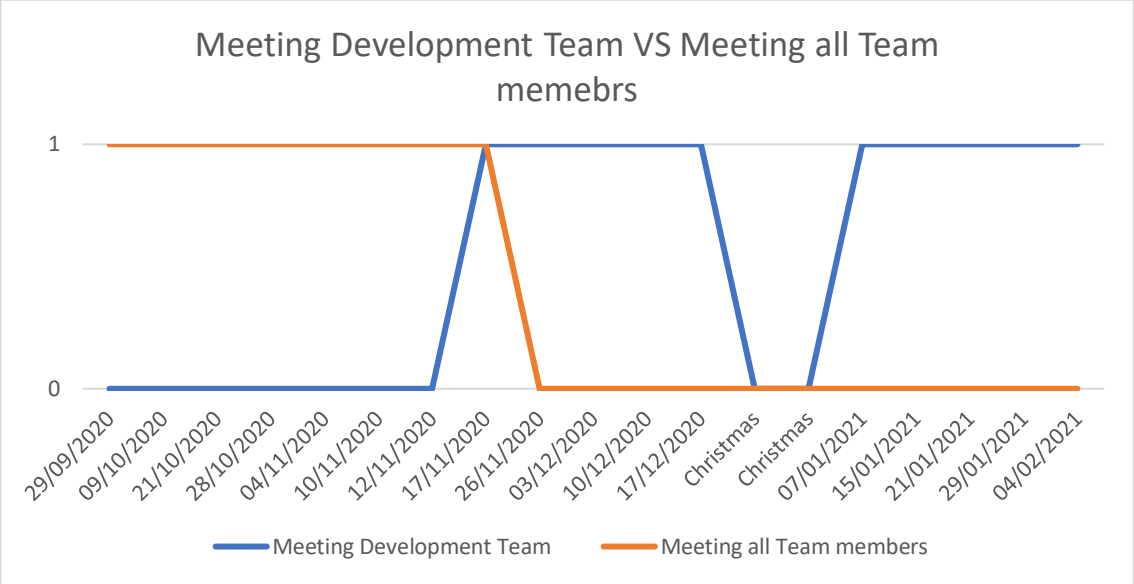

Figure 4.24: meeting between the development team VS meeting between all members of the group.

4.6.3 Kanban Board

Aa a practical tool to monitor the status of work during the processes, it was decided to implement a Kanban board with Tello, introduced in paragraph 3.6.2, on which to place all the cards, or the Kanban, corresponding to the scheduled activities. To better understand the current status of the work within the organization, it is useful to have three distinct columns that indicate the status of the activity to be carried out: "To Do" which contains the tasks that must be started, "In Progress ”where the tasks in progress are inserted and “Done” which collects the tasks performed or completed, as shown in figure 4.23. On the one hand, this tool allows to increase the sense of responsibility of the members of the work team who have the opportunity to see how their eventual delay could affect the work of others, on the other hand, it allows to have a clear vision of the work process, i.e. to know what are the activities that must be undertaken, those in progress and those already completed.

To choose the tasks to be carried out, the team members refer to the PO which, after having collected the feedback from the stakeholders and according to its experience, decides which tasks to be included in the To-Do where they are present. The developer takes over one of these and moves it from To-do to In Progress and finally, when completed, to Done, attaching his avatar on the post-it to allow traceability of who developed what. In figure 4.23 it is possible to see how each activity in progress is associated with an avatar with the name of the person on the team who is doing it.

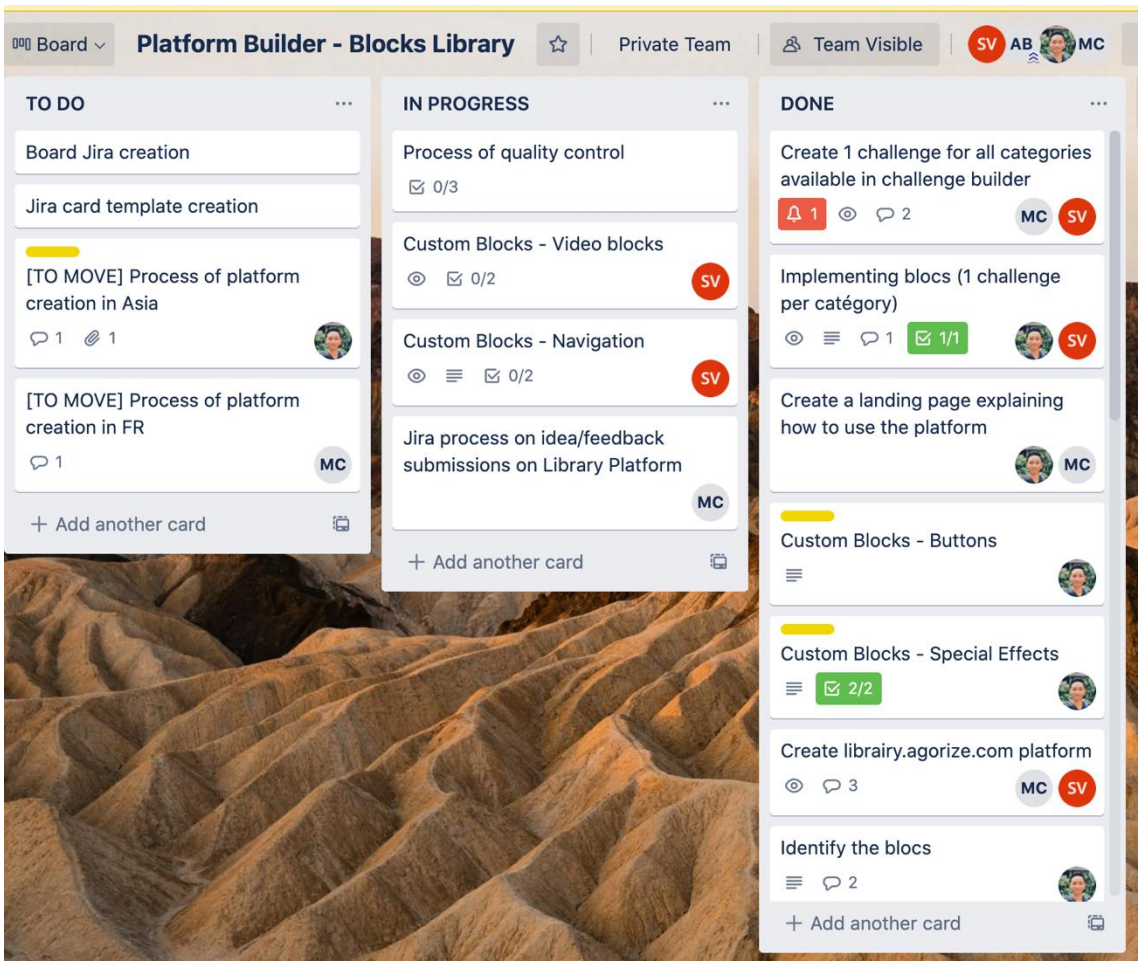

Figure 4.23: Kanban Board screen for the library.

The integration of the Kanban methodology with the Scrum method allows not to have to complete the tasks in WIP by the end of each week. Furthermore, during the meeting, the team members are asked what activities have been completed and

before being placed in the Done column, a validation activity is carried out by the PO and the SM. In the case of the library, validation consists of checking if the HTML, CSS, and JavaScript elements inserted in the platform work correctly and if they are responsive to the change of the page size. In fact, these elements must be adapted to the size of the page since the challenge must be visible both on laptops, tablets, and smartphones.

In this case, therefore, the responsibility for the activities necessary for the realization of the final deliverable has passed from the Project Manager to the Product Owner, who identifies and plans tasks together with the development team.

4.6.4 Communication with stakeholders

The assumption of Agile Project Management in the process implies the involvement of all the stakeholders of the project and therefore in addition to the EPMs also the PMs, who will then use the library platform. This type of approach, in which the customer is continuously involved and whose opinion is constantly requested, transforms the figure of the stakeholders, leading them, on the one hand, to be involved in the project and on the other to take on greater responsibility.

Stakeholders are certainly among the most important interlocutors with whom to establish effective communication as they are the bearers of interest in the activity in progress, more generally they are the individuals who can influence or be influenced by the organizational activity in terms of products and work processes. However, stakeholders are first of all people actively involved in the project and who are called to make specific decisions. The PO and SM must try to understand the needs of stakeholders and find the most effective ways to work with them to solve any problems or satisfy specific requests.

Through Jira, anticipated in paragraph 3.6.1, a communication channel has been created between team members and stakeholders (PMs and EPMs). In fact, after the weekly release of a new version of the library, PMs can open Jira tickets with feedback and any improvement proposals, such as adding other graphics or reporting some malfunction. This system guarantees concise and immediate communication, but at the same time clear and allows their points of view to be taken into account. Furthermore, it allows the PO to have a clear view of which areas to improve or implement and to mark as done those updates that have been made. At the beginning of each cycle, when the Product Owner decides the backlog, that is the activities to be prioritized during the week, he takes into consideration the feedbacks of the project stakeholders.

As for the communication of the release of a working version, the SM is responsible for communicating them through a Slack channel shared with all the stakeholders, keeping them always updated on the salient aspects of the project to avoid blocks or delays in communication.

4.6.5 Testing

It was initially decided to carry out iterative testing. Testing in the iterative model is performed at each iteration. Within the single iteration, a set of features is developed and immediately tested. The features to be tested are relatively few and the testing activities are performed quickly. The iterative method has the advantage of allowing the correction, changes, or additions inferred from the test results to be made immediately in the next iteration.

Thanks to the iteration of processes, testers can improve the learning process and the speed of execution of activities.

The testing, in this case, consisted of verifying if the graphic structures actually worked and if they were responsive, that is, they adapted not only to the laptop screen but also to the screens of tablets and mobile phones.

Figure 4.26 shows when the iterative testing phase actually took place. On the x-axis, there are the various versions released over time and in the y axis, 1 means that the test has been performed and 0 that it has not been done. So, the figure shows that in reality, the testing phase occurred only for the first two versions and then was left.

Figure 4.25: Kanban Board screen for the library.

The interruption of the testing phase also had a considerable impact on the release of new versions. In fact, after each release, the PMs often reported errors that were not caught by the development team, due to the lack of testing of the new version, which had to be solved. This meant a further extension of the release of the next version.

4.6.6 Data analysis

The team was able to work in an agile environment and welcomed this innovation proactively and with a lot of enthusiasm recognizing the relevance of Atlassian Jira as a tool for project management.

Due to the great differences between the various projects and the absence of objective evaluations of past orders, find an adequate benchmark was not possible.

The Kanban system has provided qualitative data on the completion of the activities in fact the team members are more effective in terms of time by focusing on their work, defined and visible to all members.

Problems arose during the application of the agile methodology, especially since the scrum master decided to stop following the project. Figure 4.26 shows an interesting analysis of Kanban cards, where on the x-axis there are some of the most important tasks inserted in the Kanban table in chronological order (from the one that was finished first to the one finished after) and on the axis of y the difference in days between the actual completion date of the task and when it went to Done on Trello (on the Kanban Board). It can be noted that for some tasks such as for example the addition of the Standard blocks or the Prizes block has been transferred to done more than 40 days after the actual completion date.

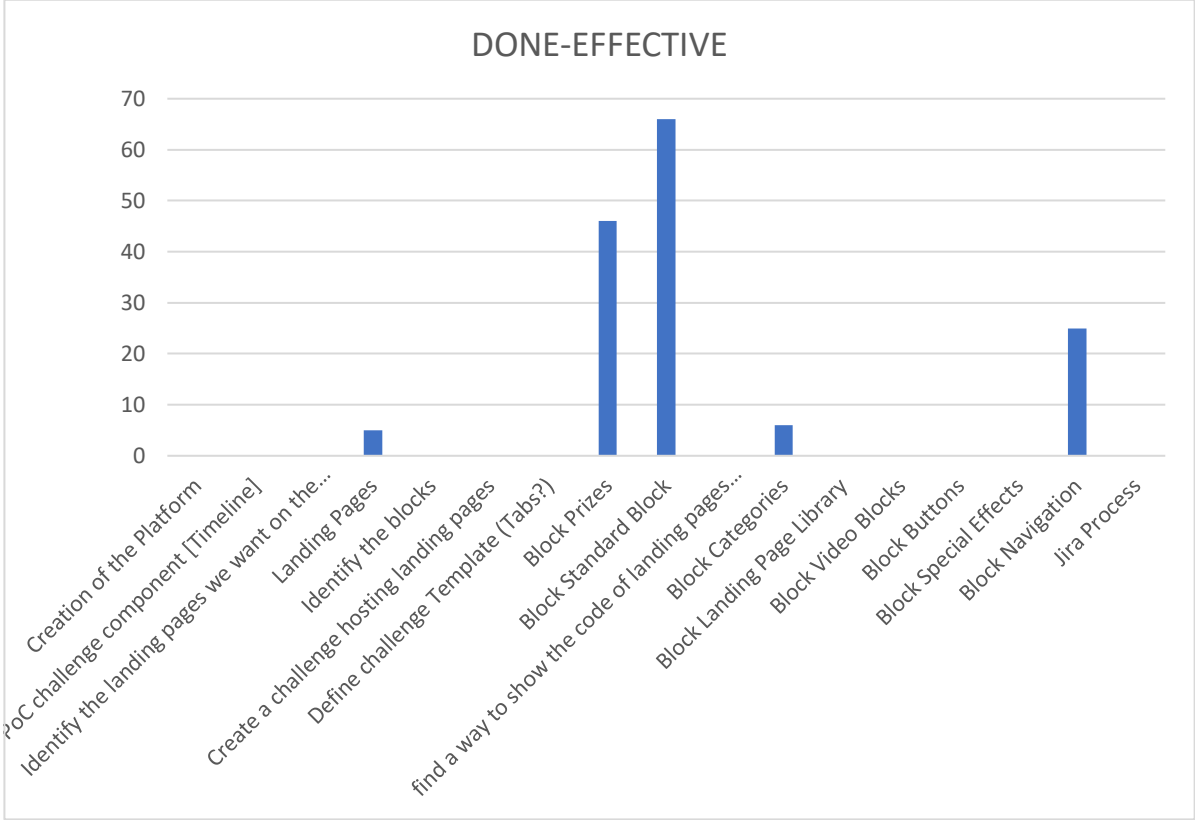

Figure 4.26: The difference between the effective day of completion of a task and the day when the Kanban shifted to Done.

Precisely this could be one of the problems of the fact that new versions were released much later than one every week. In fact, having seen those activities still in progress and therefore assigned when in reality they were finished, may have prompted the PO to assign fewer tasks than the development team could do and therefore may have slowed down the process. The cause of the failure to update the Kanban cards was most likely caused by the abandonment of the project SM as shown in figure 4.24. Furthermore, the lack of real management in the Agile methodology meant that there were no real deadlines to respect, and therefore the development team had more time for implementation.

The same goes for the testing phase which after leaving the Scrum Master was no longer performed.

What emerges from this analysis is that the Agile methodology actually allows a shorter Time to Market and releases advanced versions and allows you to use the library immediately as shown in figure 4.27. The figure shows a graph extracted from google analytics regarding the user of the platform over time and you can see how since November the library has started to enter the Agorize processes.

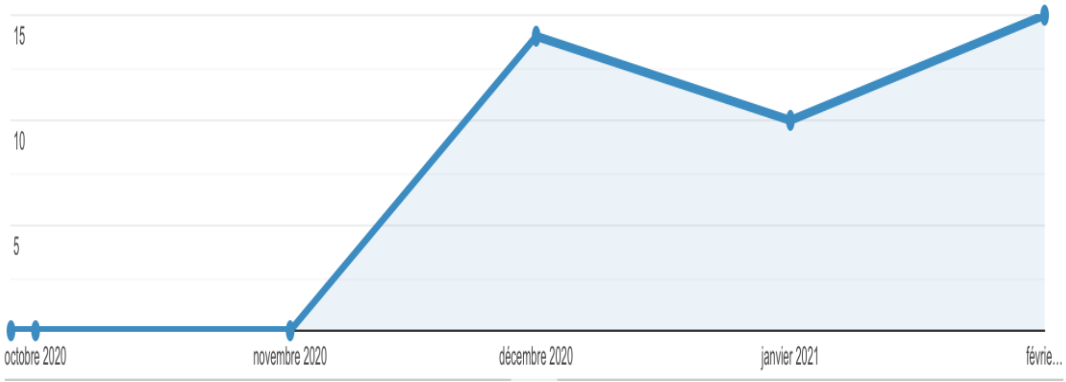

Figure 4.27: Users of the library overtime.

A priori it is difficult to define how long the sprint duration can be, which are then hardly respected. Furthermore, the figure of the Scrum Master becomes necessary to manage all the events that involve the Scrum and Kanban methodology. In fact, even for a small project like that of the library, although there have not been all the meetings involving the Scrum methodology, without the help of a figure specialized in enforcing this type of method it is difficult to follow it.

Conclusion

The quality and uniformity of web design are the key success factors for a company like Agorize. The increase in demand over the last few years has meant that the work of the EPMs has increased and that it will not be possible to guarantee an adequate level of quality for all organized competitions. So, it was decided to create a tool that would reduce and simplify the work of EPMs, trying to centralize HTML, CSS, and JavaScript knowledge to the whole organization. This tool has been called *library* and is a web platform, for now only for internal purposes, and consists of a platform with all the HTML, CSS, and JavaScript structures used in the past by Agorize, with code and example, in order to make them accessible to all those who were to create challenges or Landing Pages.

The procedures most adopted in the management of projects of this type are mostly cyclical and roughly follow the agile philosophy. It is not correct to think of managing everything through steps that follow one another linearly due to the difficulty of the cases to be treated and the many situations that gradually emerge creating the need to reformulate the problem to be faced. Adopting a cyclical change management philosophy is also useful for dealing with stakeholders, making them central during implementation. It should be noted that the study had a short time frame, but despite this, a general idea emerged, and during the process, not everything went as anticipated by the theoretical analyzes.

All this is demonstrated by the fact that, despite the positive feedback from both quantitative and qualitative results, with the continuation of the experiment the general commitment to following the principles and events of the project management methodologies has diminished. Among the feedbacks that can be drawn:

too many activities marked on the software are difficult to manage and create confusion. There is a need to have an actor who deals exclusively with updating the Kanban, thinking about the organization of events and testing.

In this specific case, there are KPIs that can be considered in the near future to assess whether the introduction of the library has actually brought benefits. The indicators to be evaluated and compared with the previous ones are the time of launching a challenge, the time spent by the PMs to create a challenge, and the time spent by the EPMs in creating Landing Pages and assisting the PMs to create graphic elements.

In the near future, the possibility of extending the use of this platform also externally is not excluded, for example for Saas customers, i.e. those customers who decide to implement the challenge independently, and also for the Sales department which can propose different prices depending on the difficulty of the graphic structures that will be created.

References

- **Abrahamsson P., Conboy K., Wang X.,** (2009). Lots done, more to do: the cur- rent state of Agile systems development research. *Eur. J. Inf. Syst.*, Vol.18 No. 4, pp. 281–284.
- **Ahmand E., Dennehy D., Conboy K, Oivo M** (2017), Kanban in Software Engineering: A systematic Mapping Study, *Journal of System and Software,* Vol 11, No 45, pp. 97-109.
- **Al-Ratrout S.,** (2019), "Practical Implementation of Agile Approaches in Teaching process", *International Journal of Engineering and Advanced Technology (IJEAT)*, Vol. 8 No. 4, pp. 278-284.
- **Bhavsar K.**, **Shah V., Gopalan S.,** (2020), Scrumban: An Agile Integration of Scrum and Kanbn in Software Engineering, *Internation Journal of Innovative Technology and Exploring Engineering,* Vol 9 No 4, pp. 2075-2084.
- **Boehm B.** (1988), A Spiral model of Software Development and Enhancement, *ACM SIGSOFT Software Engineering Notes,* Vol 11 No. 4, pp. 61-72.
- **Chesbrough H.W.** (2003) *Open innovation: The new Imperative for Creating and Profiting from Technology,* Harvard Business School Press, Harvard.
- **Chinneck J.** (2016). *Practical Optimization: a Gentle introduction*, Charleton University Ottawa, Ottawa.
- **Cohn M.** (2009), *Succeeding with Agile: Software Development Using Scrum*, Boston: Addison- Wesley, Cork.
- **Dennis A., Wixom B., Roth R.** (2012), *Systems Analysis and Design,* John Wiley & Sons, Inc, Hoboken.

- **Gaio L.** (2010) *Project management: elementi teorici e applicazioni. Metodi ed evidenze empiriche per il turismo: Metodi ed evidenze empiriche per il turismo,* FrancoAngeli, Milano.
- **Garcia A., Miguel I., Eugênio J., Vilela M, Marcondes G**, (2019) Scrum-Based Application for Agile Project Management, *Journal of Software,* Vol.15 No.4, pp. 106-113.
- **Gould, F., Joyce, N**. (2009). *Construction Project Management.* Third Edition, Prentice Hall, Upper Saddle River.
- **Hart, G.** (2002). The five W's: An old tool for the new task of audience analysis. *Technical Communication,* Vol. 48 No. 2, pp. 139-145.
- **Hazzan O., Dubinsky Y.** (2014). The Agile Manifesto. In: *Agile Anywhere*. SpringerBriefs in Computer Science. Springer, Cham.
- **Karlesky M., Voord M.** (2008), Agile Project Management**,** *Embedded Systems Conference Boston*, Vol.1 No.1, pp. 247-267.
- **Kernzer H.** (2014). *Project Management: pianificazione, scheduling e controllo dei progetti*, Hoepli, Milano.
- **Ladas C.,** (2009), *Essays on Kanban Systems for Lean Development,* Modus Cooperandi Press, Seattle.
- **Leffingwell D.** (2010). *Agile Software requirements: Lean requirements Practices For teams, programs, and the Enterprise*. Addison-Wesley Professional, Boston, MA.
- **Leybourne S.A.** (2009). Improvisation and agile project management: a comparative consideration. *International Journal of Managing Projects in Business*, Vol. 2 No. 4, pp. 519-535.
- **Mircea E. (2019),** Project Management Using Agile Frameworks, *Economy Informatics,* Vol.19 No. 1/2019, pp. 34-44.
- **Nelson R., Winter S.** (1982). *An Evolutionary Theory of Economic Change,* The Belknap Press of Harvard University Press, Cambridge.

- **Paul A., Rahman S.,** (2018), Study on Agile Management in Construction Project Using Scrumbam Methodology, *International Research Journal of Engineering and Technology (IRJET*), Vol. 5 No.11, pp. 774-777.
- **Reddy A.** (2016), *The Scrumban [R]evolution: Getting the Most Out of Agile, Scream and Lean Kanban,* Addison Wesley, New York.
- **Robbins J.** (2012). *Learning Web Design,* O'Reilly, Cork.
- **Schwaber K.** (1997) SCRUM Development Process. In: Sutherland J., Casanave C., Miller J., Patel P., Hollowell G. (eds) *Business Object Design and Implementation*. Springer, London. https://doi.org/10.1007/978-1-4471-0947-1_11
- **Schwaber K.** (2004). *Agile project management with scrum*. Microsoft Press, Redmond, Washington.
- **Sutherland, J., Schwaber K.** (2011). *The Scrum papers: Nuts, bolts, and origins of an agile process*, Scrum. Inc., Cambridge.
- **Tereso, A., Ribeiro, P., Fernandes, G., Loureiro, I., & Ferreira, M.** (2018). Project Management Practices in Private Organizations. *Project Management Journal*, Vol 50 No.1, 6-22. doi: 10.1177/8756972818810966.
- VersioOne. 10th Annual State of Agile Report (2016), *http://www.agile247.pl/wp-content/uploads/2016/04/VersionOne-10th-Annual-State-of-Agile-Report.pdf,* 31/01/2021.
- VersioOne. 11th Annual State of Agile Report (2017), *https://www.agile247.pl/wp-content/uploads/2017/04/versionone-11th-annual-state-of-agile-report.pdf,* 30/01/2021.