UVIC Universitat de Vic
Escola Politècnica
Superior

Bachelor Thesis

Integration of the Scrum methodology in mechatronic product development

Joan Josep Mauri Also

Bachelor degree in Mechatronics Engineering, Bachelor degree in Industrial Electronics and Automation Engineering.

Tutor: Dr. Esteve Gallego Jutglà

Vic, September 2015

Universitat de Vic
Escola Politècnica
Superior

SUMMARY BACHELOR THESIS

BACHELOR DEGREE IN MECHATRONICS, BACHELOR THESIS IN INDUSTRIAL AUTOMATION ENGINERING

Title: Integration of the Scrum methodology in mechatronic product development

Keywords: Product development, Scrum, Agile, Workshop, Packaging, Mechatronic

Author: Joan Josep Mauri Also

Tutor: Dr. Esteve Gallego Jutglà

Date: September 2015

The purpose of this study was to demonstrate if it would be possible for a mechatronic product development team to use Scrum, an Agile Development framework with both, the students of UVIC-UCC and the company ITQ GmbH, behind the student project called **Mi5**. The Agile philosophy and methods have revolutionized the software development industry in the last decade, and therefore it was of interest to see if this new way of working would be applicable in other disciplines.

Thus, the study focuses on if Scrum can be applied in mechatronic product development, and if it needs any adaptations. Two case studies of agile methods used in mechanics, electronics and software development are presented. The primary case study follows a mechatronic product development team through their 5-month experiment of using Scrum. In this context a five-member team in question co-operates to develop an intelligent prototype for packaging food products in a production line. The methods used in order to form the case studies were observations, interviews, documentary analysis and informal conversations.

The results indicate that it is possible for a mechatronic development team to use the Scrum framework, with some adaptions of the framework. The team in the case study made use of all main aspects of the framework, but the deviations were not a working product or product increment is not produced every iteration and not all the members finish until the end of the project. The results also indicate that using the framework improved the team’s work and progress, and the team decided to continue using the framework after the experiment. The conclusion is that Scrum can be used in a mechatronic environment, by mechatronic development teams. Some adaptations might be needed, depending on each case, but this is also recommended in the literature to software development teams.

Acknowledgements

Firstly, I would like to express my sincere gratitude to my advisor Dr. Esteve Gallego for the continuous support of my bachelor thesis and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my bachelor thesis study.

Besides my advisor, I would like to thank the rest of my thesis committee: Dr. Moises Serra and Dr. Jaume March Amengual, for their insightful comments and encouragement, but also for the hard question which incented me to widen my research from various perspectives.

My sincere thanks also goes to Prof. Jordi Serra, Dr. Àngels Crusellas, Dr. Miquel Caballeria, Dr. Judit Molera, Dr. Montserrat Corbera and Dr. Jordi Villà, who provided me an opportunity to join in this project, and who gave access to the laboratory and research facilities. Without they precious support it would not be possible to conduct this research.

I thank my fellow teammates of the project Mi5 in for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last year. Also I thank my friends in the following institution ITQ GmbH. In particular, I am grateful to Dr. Rainer Stetter to give me this opportunity project-based in international cooperation.

Last but not the least, I would like to thank my family: my parents, my grandparents, my aunt and to my cousins for supporting me spiritually throughout writing this thesis and my life in general.

Table of Contents

1. Introduction

Delivering products in less time, at lower prices and with new, innovative functions is a vital prerequisite for success in an increasingly though global market. Product reliability, however must not suffer as a result of the measures that are introduced to achieve those goals. Improving the development process is key to "squaring the circle".

An additional challenge is the adaption of the continual growth in the proportion of software engineering in mechatronic developments (see Figure 1). Coordinating mechanics, electronics, and software is becoming even more importance because of mutual interdependencies. Therefore these projects present difficulties to plan and difficulties of adaption in rapidly changing environments.

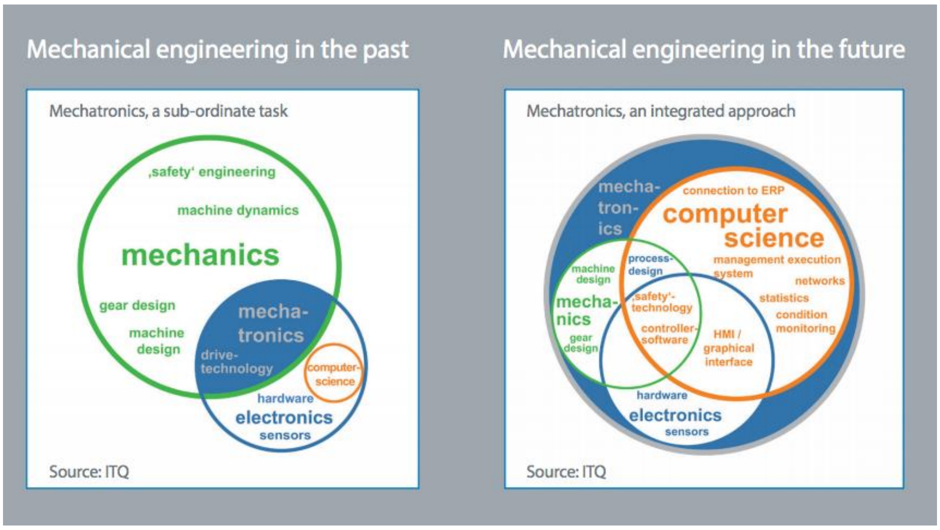


Figure 1. Increasing software proportion.

By the late 1990, a majority of software development was criticized as slow, bureaucratic and overly regimented, in reaction to these heavy software methods new ways of project management appeared in the beginning of 21st century, which changed the way of developing software. The term of Agile Software development emerged as a collective set of methods formed by Agile Manifesto in 2001, based on iterations and incremental development with cross-functional and self-organizing teams. Many important companies of today use Scrum such as Google, Microsoft, Yahoo, Siemens and more.

The topic remains if Scrum as a framework, is only suitable in software development. Nonetheless when observing the Scrum system, its pillars are not directly designed for software development, instead is for product development. Thus, the conclusion is Scrum is suitable for any kind of development where required strategy, organization, plan creation, evaluation and innovative approaches.

1.1. Purpose

The purpose of this study is to see if Scrum methodology can be used for mechatronic product development with or without adaptions. The fact of how a student team can co-operate and synchronize themselves and with other students and how they can teach other students will be studied.

The investigation was conducted at the UVIC-UCC (Spain), but also with the international cooperation of ITQ GmbH (Germany). The Vic team within the offices and laboratories of UVIC-UCC started using Scrum at the end of February 2015. This team worked in closed connection with the Munich team that was giving support on the communication, development and methodology. This thesis is a study of the results of this experiment of 5 month-period, from February to July 2015. One additional case is studied in order to understand widely the application of Scrum in mechatronic workshops for children.

1.2. Delimitations

The borders of this study are the main case studied in detail, complemented with one less detailed case. The results of this study might not be applicable to other teams or companies. The length of this empirical study was 5 month, as a result in great amount of ideas and data. The close cooperation of the author as Scrum Master and team member might have a significant impact on the results, for the team the fact to spend during this experiment in close environment could affect the conclusions of the study.

1.3. Outline of thesis

The outline of the thesis is the following:

In chapter 2, the problem formulation and research questions will be presented.

In chapter 3, the methods referent to the literature review and the empirical study will be described. The field of the literature review is described, include the covered areas.

In chapter 4, the theoretical framework of the thesis is explained. The topics involved are: Agile software development, waterfall method, the comparison between agile and waterfall processes and the Scrum framework.

In chapter 5, the empirical study will be presented along four iterations.

In chapter 6, the results will be analyzed and discussed in relation to the Scrum theoretical framework.

In chapter 7, the conclusions will be presented based in the analysis and discussions.

2. Problem formulation

In this chapter, the problem will be formulated through introductions of the existing project and the methods of development.

2.1. Introduction to ITQ GmbH

ITQ is an independent services company, they provide software engineering and interdisciplinary project management for fifteen years, also giving advices and support companies and plant construction during all stages of software development such analysis, design, implementation, test and development. They integrate the disciplines of mechanics, electronics, and software on the functional and human levels with flexibility, reliability and integrity. In the education aspect, ITQ promotes the student projects as the mentioned below.

Product development at ITQ GmbH mainly consist of software and mechanical design teams. What is important to note is that these teams have to cooperate to develop a single product, making essential the cooperation, communication and the decision making between teams, through the constant training on a combination of agile and traditional methods of development.

2.2. Project Mi5

The project Mi5 is the result of a multi-dimensional approach started in 2014, Munich founded in ITQ and mostly managed by motivated and talented students with a common goal to develop an industrial and mechatronic process for the food industry. In the first period of the project, from March to November 2014, it was form four functional teams, which mainly consisted of mechanical design and software development. Regarding the process development, the fact to manage four functional teams, to develop a new single product, makes essential the synchronization between them.

Before to start the second period of the project, the author was participating in the first period of the project in Munich. After that, he started and promoted a new project in Vic. Therefore, the second period was initiated on the 25th February and finalized the 31st July in 2015. Thus, the team in Vic was created and one project leader from the software team in Munich was the responsible person to give support to the new team via teleconference. In addition, other team members from Munich were cooperating as well (see Figure 2).

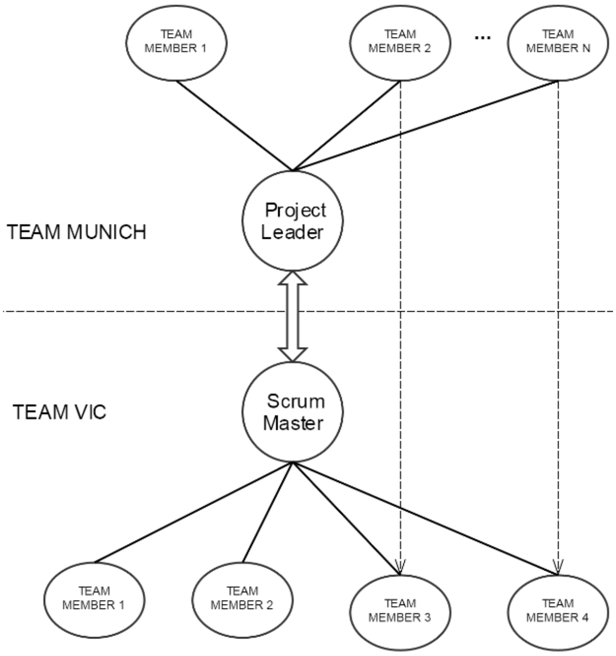


Figure 2. Team structure cooperation.

2.3. **The problem definition**

Before to start this project study case, in the first period of the project Mi5 in Munich, four development teams within project Mi5 were working paralelly with different work methods. Two software teams and one mechanical team used Scrum, while the other mechanical team used traditional project management methods. As noted above, to develop a new single product, the four teams needed to synchronize and cooperate. In other words, the four team were highly depended on each other's development. In broad the synchronization between the work methods of the teams was lacking.

Firstly, a chance to improve the work methodology in one mechanical team was observed, when looking the successful progress of the Scrum teams. Secondly, the synchronization between the four teams was often limited, regarding the fact to go in the same development phase at the same time. Thirdly, the issue was how to manage between the different work methods to the requirements of the product development.

After the problems had been identified, it was made a decision to use Scrum as a work method for the new mechatronic team in Vic, focusing with the issues commented above. Concerning the second issue the synchronization in this case study was via teleconference. However, it was decided to not cover the third issue, because in this case the work methods were Scrum in both institutions.

2.4. Research questions

Concerning the main scope defined, as the most important research question (RQ) of this thesis is:

1. RQ: Can a mechatronic product development team use Scrum?

In order to support the main research question of this study, the following questions were posed

1.1. RQ: Does Scrum framework need adaption for mechatronic teams?
1.2. RQ: What are the important factors to succeed with Scrum a mechatronic team?

Referent the second issue in the problem definition, where in this study case is via teleconference, the following research question was posed

1.3. RQ: What effects does it have that an international cooperation between teams, in communication aspect are both using Scrum?

3. Methodology

In the following chapter will be explained what methods were used to collect the data. First the research approach and the literature review will be explained, therefore different methods will be explained in detail.

3.1. Research design

A descriptive research methodology was used for this study. To evaluate if mechatronic product development team can use Scrum, literature was conducted as theoretical part through both empirical cases.
As it is observed in the Figure 3, the research follow two different path, however the basis of the empirical study is based in the theoretical part.

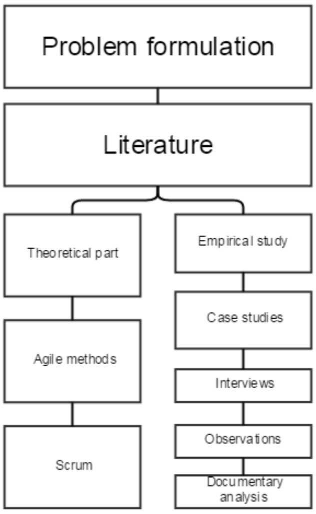


Figure 3. Research structure.

The theoretical part consist of the agile methods, the comparison between agile and traditional methods, but is mainly focus in Scrum method. The empirical study focus in one case study; the case study was conducted at UVIC-UCC, where an investigation of a mechatronic team was followed. The data collection of the study case study were performed through interviews, observations, documentary analysis and informal communication.

3.2. Literature review

The literature review of this thesis is mainly focus on the Scrum, as a type of agile methodology. To perform a good research valuable sources were evaluated taking into an account the experiences of the authors and early publications.

The most valuated literature sources found were the following: *Essential Scrum* (Kenneth S. Rubin, 2012), *The professional Scrum Master's Book* (Stacia Viscardi, 2013), *Agile retrospectives making good teams great* (Esther Derby and Diana Larsen).

3.3. Observations

Observations can form a part either quantitative o qualitative research. The researcher can collect during observation the behaviors of people under certain circumstances, being an active participant or being a complete observer. According to Hennink et al. (2011) participant observation is then categorized further into four levels of participation: passive, moderate, active and complete participation. The observations were conducted with complete participation in the primary case study and with passive and moderate participation in the additional case study.

3.4. Interviews

Interviews are one of the data collection methods for qualitative research. Interviews consist of meeting with participants one on one and asking them open-ended questions. Interviews can be structured or semi-structured. In the structured interview, the researcher has a predetermined set of questions to ask and does not deviate from those questions. In a semi-structured interview, the researcher will have prepared questions but has the freedom to ask additional questions.

Throughout the primary case study were mostly used semi-structured interviews, as part of the scrum appliance requires. In addition, informal conversations with the team and people from outside of the project were used to collect important information. Concerning the additional case study was used just informal conversations to gather information.

3.5. Documentary analysis

Documentary analysis involves obtaining data from existing documents without having to question team members through interview, questionnaires or observe their behavior. Documentation is tangible material in which facts or ideas have been recorded. In addition media files can also be part of the subject of documentary analysis, including films, presentations and photographs.

In the primary case study were mostly used presentations, photographs, videos and a special time management document. In the additional case study were used presentations and photographs. In both case studies all data were recorded and transcribed.

3.6. Case study

According Kardos & Smith (1979) a case study is an account of an activity, event or problem that contains a real or hypothetical situation and includes the complexities that is encountered in the workplace. Case studies are used to help the complexities of decision making. Analyzing a case study requires practice to apply knowledge and thinking skills to a real situation. To learn from a case study analysis you will be "analyzing knowledge, reasoning and drawing conclusions".

The case study is about face complexities, apply knowledge and thinking skills in real project situation. Most of the circumstances in the case study were based in problem-solving activities, decision-making in real and simulated scenarios.

The realization of the study case was performed with a parallel approach, using a mixture of interviews, observations and documentary analysis.

The primary case study was performed in four iterations. The length of this case study were five month development. During the four iterations the author was observing the process as a ScrumMaster and participating as an active member of development team. During this process, were used semi-structured interviews, informal conversations with team members from Vic and team members from Munich via teleconference.

4. Theoretical framework

In this chapter the Agile and waterfall methodologies will be explained and compared in broad terms.

4.1. Agile software development

In the following sub chapter will be presented; the concept, origins, the principles, the waterfall method and a comparison between agile and traditional methods.

4.1.1. Agile concept

Agile software development is a group of software development methodologies based on iterative development, where requirements and results evolve through collaboration between self-organizing, cross-functional teams. It promotes a disciplined project management that encourages the frequent inspection and adaption, teamwork, self-organization, accountability, a set of engineering best practices intended to allow fast deliveries of high-quality software.

According to Highsmith and Cockburn (2001), "what is new about agile methods is not the practices they use, but their recognition of people as the primary drivers of project success, coupled with an intense focus on effectiveness and maneuverability. This yields a new combination of values and principles that define an agile world view". The following points describes the most remarkable characteristics of Agile:

- People oriented
- Adaptive
- Balancing Flexibility and Planning
- Empirical process
- Decentralized Approach
- Simplicity
- Collaboration
- Small Self-organizing teams

The advantages of this methodology are the following:

- Allows for changes to be made after the initial planning. Re-writes to the program, as the client decide to make changes, are expected.
- Close, daily cooperation between business people and developers.
- Regular adaption to changing circumstances.
- Customer satisfaction by rapid, continuous delivery of useful product.
- People interactions are emphasized rather than process and tools.
- Continuous attention to technical excellence and good design.

The disadvantages of this methodology are the following:

- Lack of emphasis on necessary documentation and designing.
- Intensive work for the team.
- Harder to understand for new starters to integrated in the team.

- Difficult to assess the effort required.

To conclude, agile development processes have capability to create response to change, capability to balance flexibility and structure, capability to lead organizations through complex environments, capability to innovate and improve the product with external feedback. In addition the key of this strategic method is focus on delivering customer value.

4.1.2. Agile origins

The concepts behind Agile go even further back in time. In 1950s, a management consultant by the name of W. Edwards Deming created the PDCA (Plan-Do-Check-Act) cycle as a framework for continuous improvement. PDCA, also known as the Deming or Shewhart cycle, had an early influence on Toyota's lean approach to manufacturing. These ideas map one-to-one that of a Scrum sprint, and even to a sprint's daily scrum, as indicated in the picture xx, but Deming didn't know he was doing scrum. Or more appropriately, is that today's Scrum teams don't realize that they're applying the Deming cycle (see Figure 4).

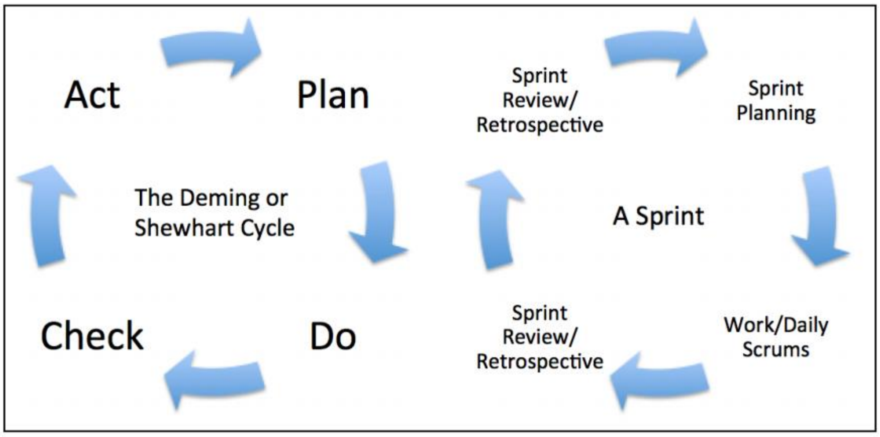


Figure 4. PDCA and Agile comparison.

4.1.3. Manifesto for Agile Software development

In February 2001, seventeen software developers met to discuss lightweight development methods. They published the *Manifesto for Agile Software Development.* Even though having different perspectives of software methodologies, all of them agreed on the treaty of the Agile Manifesto, which is as follows:

"We are uncovering better ways of developing software by doing it and helping others to do it. Through this work we have come to value:

- **Individuals and interactions** over processes and tools.
- **Working software** over comprehensive documentation.
- **Customer collaboration** over contract negotiation.
- **Responding to change** over following a plan.

That is, while there is value in the items on the right, we value the items on the left more. "

The Agile Manifesto is based on twelve principles:

1. Customer satisfaction by early and continuous delivery of useful software.
2. Welcome changing requirements, even in late development.
3. Working software is delivered frequently.
4. Close, daily cooperation between business people and developers.
5. Projects are built around motivated individuals, who should be trusted.
6. Face-to-face conversation is the best form of communication.
7. Working software is the principal measure of progress.
8. Sustainable development, able to maintain a constant pace.
9. Continuous attention to technical excellence and good design.
10. Simplicity-the art of maximizing the amount of work not done-is essential.
11. Self-organizing teams.
12. Regular adaption to changing circumstance.

In summary of the twelve principles, the highest priority is the customer satisfaction through continuous deliveries and feedback sessions, the best designs and products emerge from self-organizing teams, communication is the best way to achieve the goal through the close cooperation between business people and developers, the continuous attention to simplify and improve the process, ensures the quality product development.

4.1.4. Waterfall model

The waterfall model is a sequential design process that was the most common approach of developing software. The process is seen as flowing steadily downwards through the phases of conception, analysis, design, construction, testing, deployment and maintenance (see Figure 5).

The waterfall development model originates in the manufacturing and construction industries: highly structure physical environments in which after-the fact changes are prohibitively costly, if not impossible. Since no formal software development methodologies existed at that time, this hardware-oriented model was simply adapted for software development.

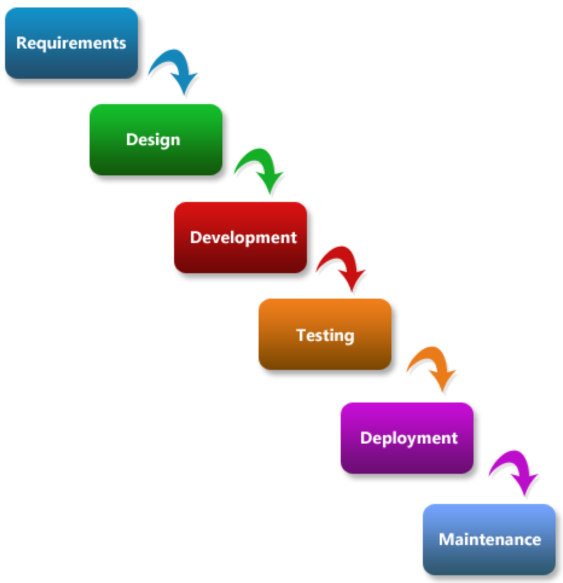


Figure 5. Waterfall model.

Thus, the waterfall method maintains that one should move to a phase only when it's preceding phase is review and verified.

The advantages of this model are the following:

- It is simple, easy to understand and use.
- It is easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
- The phases are processed and completed one at time. Phases do no overlap.
- It works well for small projects where requirements are clear at the beginning.

The disadvantages of this model are the following:

- Once an application is in the testing stage, it is very difficult to go back and change something that was not well thought out in the concept stage.
- Highs amounts of risk and uncertainty.
- Lack of opportunity for customer feedback.
- Hard to respond to changes.
- High effort and costs for writing and approving documents for each development phase.
- Not a good model for complex and object-oriented projects.
- Poor model for a long and ongoing projects.
- Not suitable for the projects where requirements are not clear and fixable at the beginning.

To conclude, the waterfall methodologies are quite rigid, as there are not ideal projects that allow to define all the requirements in the beginning and predict the future events, as well as there is not a clear path to follow strict steps and achieve the success. In today's projects the key of the success remains in the ability to response in constant changing environments.

4.1.5. Waterfall vs. Agile

A comprehensive understanding of the main differences between both development processes, can be understood by comparing the iterative nature of the agile method with the waterfall model over time.

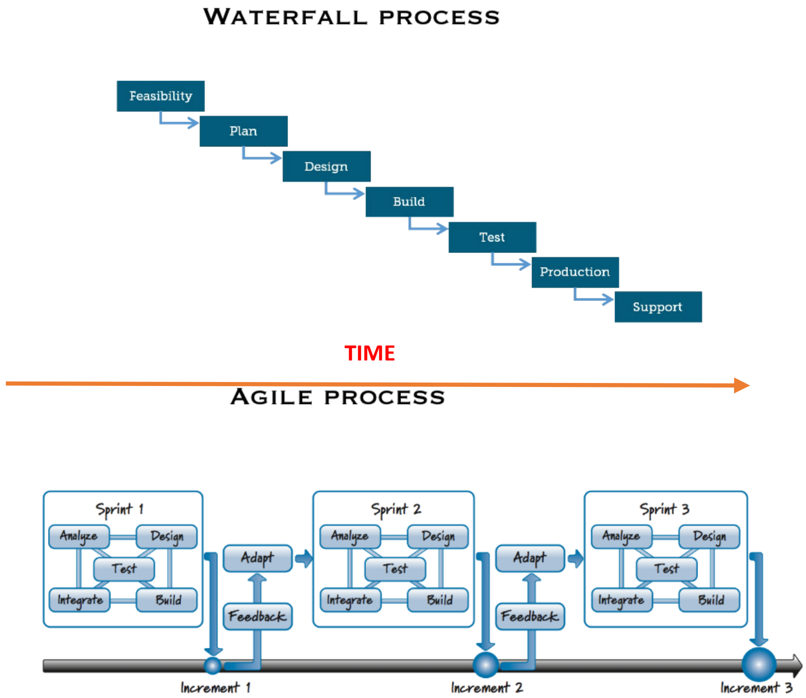


Figure 6. Waterfall and Agile comparison over time.

In the Figure xx, the Waterfall process is presented on the top where the steps of the process are carried one-by-one in a sequentially way over time. In the beginning of the project is studied the viability taking into an account the resources. When the Feasibility step is done, the plan step starts identifying all the requirements carefully, once the plan step is fully completed the design starts and so on successively. As a sequential method, it is no possible perform a new step without finish the previous one. The most problematic aspects of this method is the obligation of define all the requirements in the beginning of the project and the rigid restriction to no be able to return in a previous step in case of failure. This process does not interact well to rapid changing environments.

In comparison, the Agile process is based on iterative cycle, where in each iteration, an increment of the tasks is analyzed, designed, built and tested. By the end of the iteration this increment is evaluated with people from outside and inside of the project, as a result a fully increment usable. This method allows to change quickly the possible errors throughout the process as well as modify not appropriate conducts of the team. As it is observed in the Figure 6, both processes have the same direction conditioned by time, while in the waterfall process there is no usable product or code ready until the built step, in Agile process there is usable product in each iteration. This method promotes the importance of testing continually throughout the process.

4.2. Scrum

In the following sub chapter will be presented the Scrum methodology; concept; origins; the domains; framework; high level values and planning principles.

4.2.1. Concept

Scrum is an agile approach for developing products and services. The process starts by creating a product backlog, a prioritized list of features and other capabilities needed to develop a successful product. Guided by this list, users always work on the highest priority items first. Therefore when there are circumstances run out of resources (such as time), any work not completed will be of lower priority than completed work.

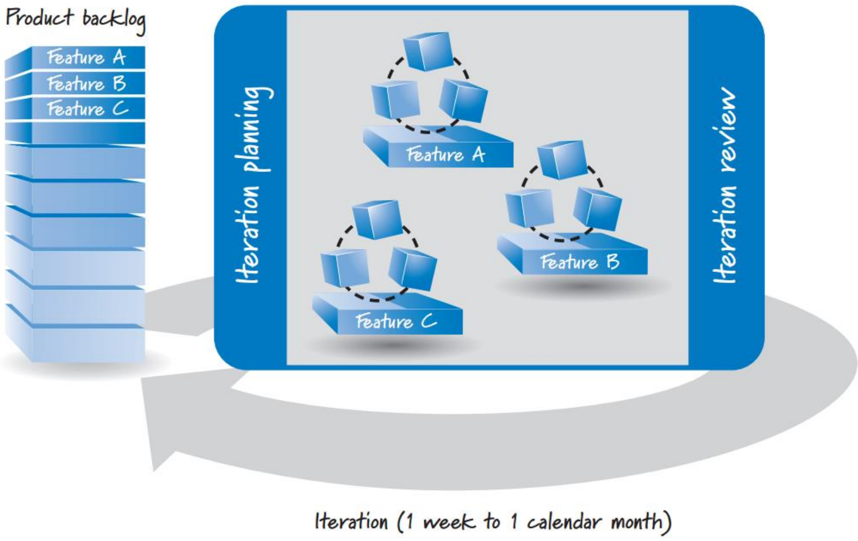


Figure 7. Scrum development overview.

The work itself is performed in short development cycles called "iterations", which usually is a period of time from a week to a calendar month in length. The amount of work to do inside the product backlog based in box features is much greater than can be completed by a team in one short-duration iteration.

At the beginning of each iteration the team plans which features of the product backlog to create in the upcoming iteration. In the figure 7, the team has chosen that it can create features A, B and C.

During each iteration the team does all the work, such as researching, designing, constructing, coding and testing.

At the end of each iteration the team should have a potentially shippable product, one that can be released if appropriate. If releasing after each iteration isn't appropriate, a set of features from multiple iterations can be release together.

At the end of the iteration, the team reviews the completed features (work done) with people from outside to get their feedback. Based on the feedback the product owner, the scrum master and the team can plan the next iteration improving and adapting the possible changes. For example, if the people from outside see a completed feature and the realize that another that they never considered must also be considered in the product backlog, the product owner can simply create a new item representing that feature and insert it into the product backlog in the correct order to be worked on in a future iteration.

As each iteration ends, the whole process is begun a new with the planning of the next iteration.

4.2.2. Scrum origins

In 1986, Harvard Business Review published an article by Hirotaka Takeuchi and Ikujiro Nonaka entitled "The new Product Development Game" in which authors wrote that development organizations must extend their focus beyond scope, time, and cost to find ways to increase speed and flexibility of product delivery in order to win in the new competitive landscape. This article was the first mention of Agile as a new paradigm for product development. Takeuchi and Nonaka set out to describe: built-in instability, self-organizing project teams, overlapping development phases, multi-learning, subtle control and transfer of learning.

In 1993, Jeff Sutherland and his team at Easel Corporation created the Scrum process for use on software development effort by combining concepts from the 1986 article with concepts from object-oriented development, empirical process control, iterative and incremental development, software process and productivity research and complex adaptive systems.

In 1995, Ken Schwaber published the first paper on Scrum at OOPSLA. Since then, Schwaber and Sutherland, together and separately, have produced several Scrum-specific publications, including *Agile development with Scrum* (Schwaber and Beedle 2001), *Agile project Management with Scrum* (Schwaber 2004), and "The Scrum Guide" (Schwaber and Sutherland 2011).

4.2.3. Why Scrum?

According to Rubin (2012), Scrum focus on delivering working, integrated, tested, business-valuable features each iteration leads to results being fast. Scrum is also suited to help organizations succeed in a complex world where it is necessary a quick adaption based on the interconnected actions of customers, users, developers, sponsors and other stakeholders. And Scrum provides more joy for all participants. Not only are customers delighted, but also the people doing the work actually enjoy it! They enjoy frequent and meaningful collaboration, leading to improved interpersonal relationships and greater mutual trust among team members.

Though Scrum is an excellent solution for many situations, it is not the proper solution in all the circumstances. It helps to understand the situation in which we have to operate and decide on a situation-appropriate approach. The Cynefin framework defines five different domains: simple, complicated, chaotic, complex and the fifth domain, disorder, which occurs when you don't know which other domain you are in (see Figure 8).

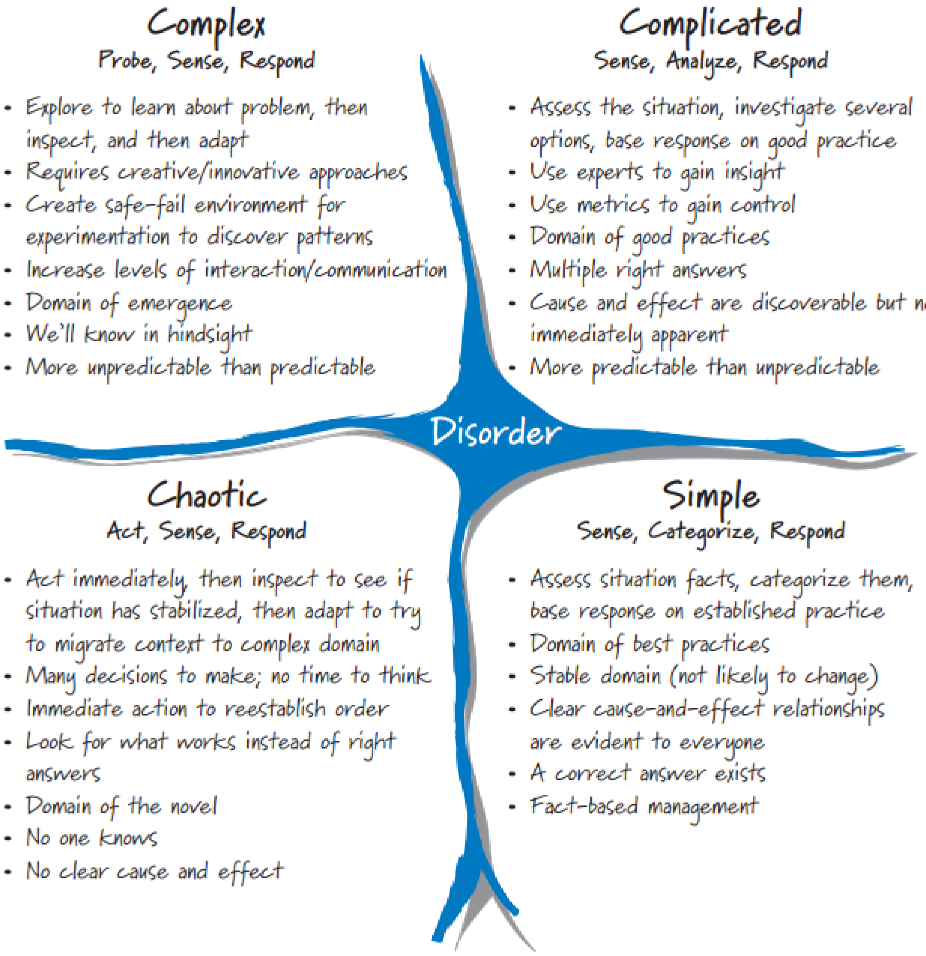


Figure 8. Cynefin framework.

Dealing with complex problems, where things are more unpredictable than predictable. If there is a right answer, it will be detected only with hindsight. This is the domain of emergence. It is necessary explore to learn about the problem, then inspect and adapt based on the learning. Working in complex domains requires creatives and innovative approaches. Thus, is remarkable to create a safe-fail environment high levels of interaction and communication are essential.

Scrum is particularly well suited for operating in a complex domain. In such situations demands ability to probe (explore), sense (inspect), and respond (adapt) is critical.

4.2.4. Scrum framework

Scrum is not a standardized process where you methodically follow a series of sequential steps that are guaranteed to produce, on time and on budget, a high-quality product that delights customers. Instead, Scrum is a framework for organizing and managing work. The Scrum framework is based on a set of values, principles, and practices that provide the foundation to which any organization will add its unique implementation of relevant engineering practices and any specific approaches for realizing the Scrum practices. The result will be a version of Scrum that is adaptable in any case.

The Scrum practices themselves are embodied in specific roles, activities, artifacts, and their associated rules (see Figure 9).

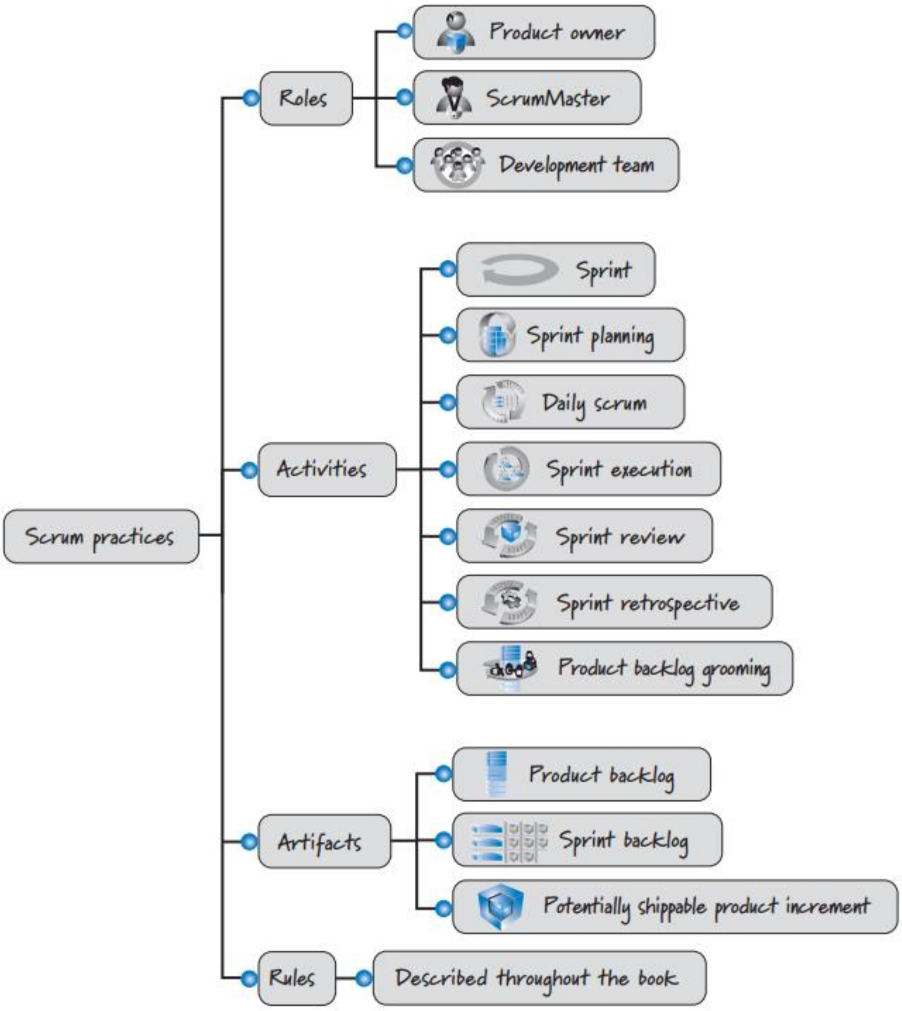


Figure 9. Scrum practices

4.2.4.1. Roles

Scrum development efforts consist of one or more Scrum teams, each made up of three Scrum roles: product owner, Scrum Master and the development team. (See Figure 10). There can be other roles when using Scrum, but the Scrum framework requires only the three listed.

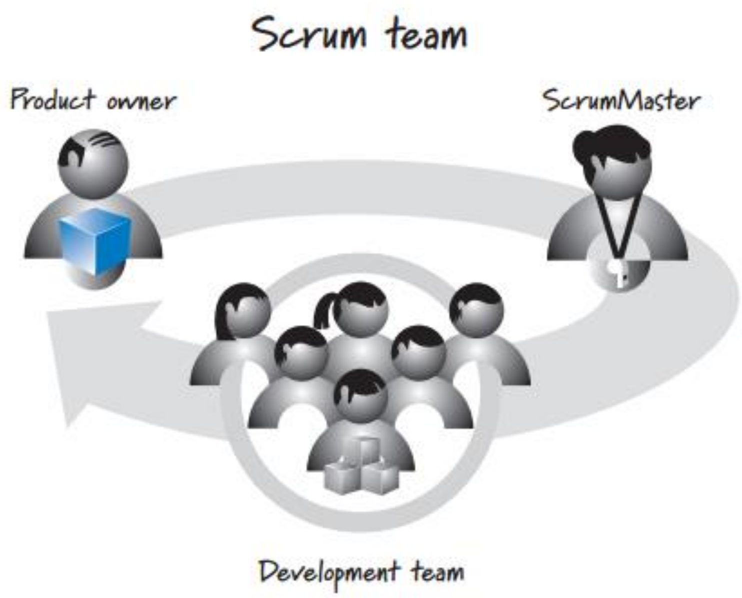


Figure 10. Scrum team.

4.2.4.2. Product owner

The product owner is the empowered central point of product leadership. He is the single authority responsible for deciding which features and functionality to build and the order in which to build them. The product owner maintains and communicates to all other participants a clear vision of what the Scrum team is trying to achieve. As such, the product owner is responsible for the overall success of the solution being developed.

It doesn't matter if the focus in on an external product or an internal application; the product owner still has the obligation to make sure that the most valuable work possible, which can include technically focused work, is always performed.

To ensure that the team rapidly builds what the product owner wants, the product owner actively collaborates with the Scrum Master and development team and must be available to answer questions soon after they are posed.

To be an effective Product owner, it is necessary manage domain skills, people skills, decision making and accountability (see Figure 11).

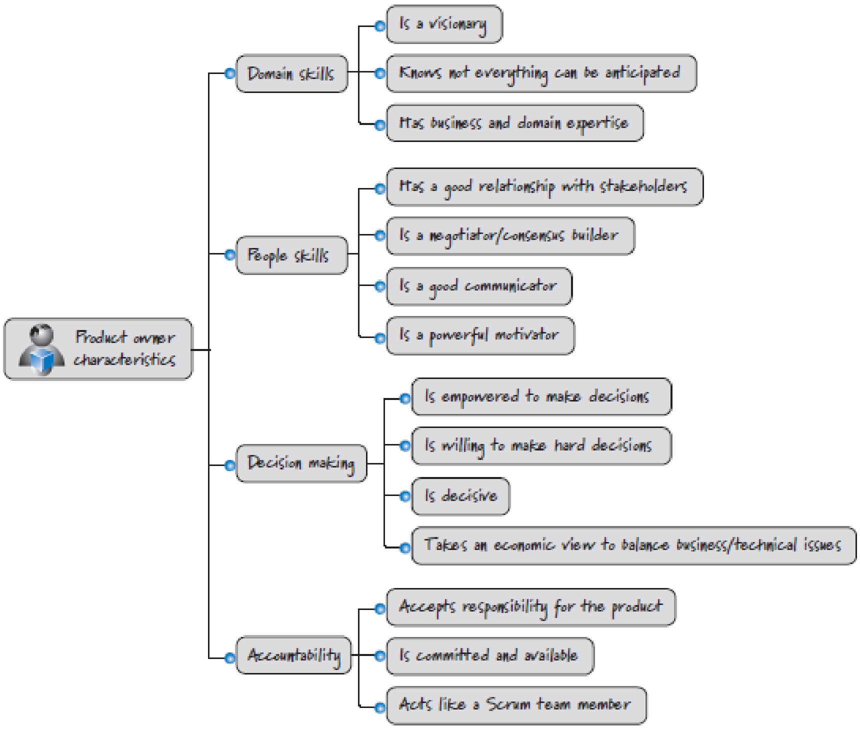


Figure 11. Product owner characteristics.

4.2.4.3. Scrum Master

The Scrum Master helps everyone involved understand and embrace the Scrum values, principles and practices. He acts as a coach, providing process leadership and helping the Scrum team and the rest of the organization develop their own high performance, organization-specific Scrum approach. At the same time, the Scrum Master help the organization through the challenging change management process that can occur during a Scrum adoption.

As a facilitator, the Scrum Master helps the team resolve issues and make improvements to its use of Scrum. He is also responsible for protecting the team from outside interferences and takes a leadership role in removing impediments that inhibit team productivity (when the individuals themselves cannot reasonably resolve them). The Scrum Master has no authority to exert control over the team, so this role is not the same as the traditional role of project manager.

To be an effective Scrum Master, it is necessary be knowledgeable, questioning, patient collaborative, protective and transparent (see Figure 12).

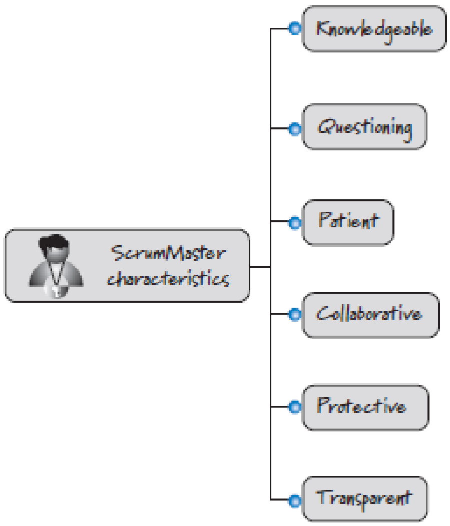


Figure 12. Scrum Master Characteristics.

4.2.4.4. Development team

The development team self-organizes to determine the best way to accomplish the goal set out by the product owner. The development team is typically five to nine people in size. Its members must collectively have all of the skills needed to produce good quality, working software. Scrum can be used on development efforts that require much larger teams. However, rather than having one Scrum team with 35 people, there would more likely be for o more Scrum teams, each with a development team of nine or fewer people.

To be an effective development team, it is necessary be self-organize, functionally diverse, T-shaped skilled (ability to work outside of the core area and to work in one functional area, specialty), musketeer attitude, high-bandwidth and transparent communication, right-sized (small teams), focus and committed, works at sustainable pace and long lived (see Figure 13).

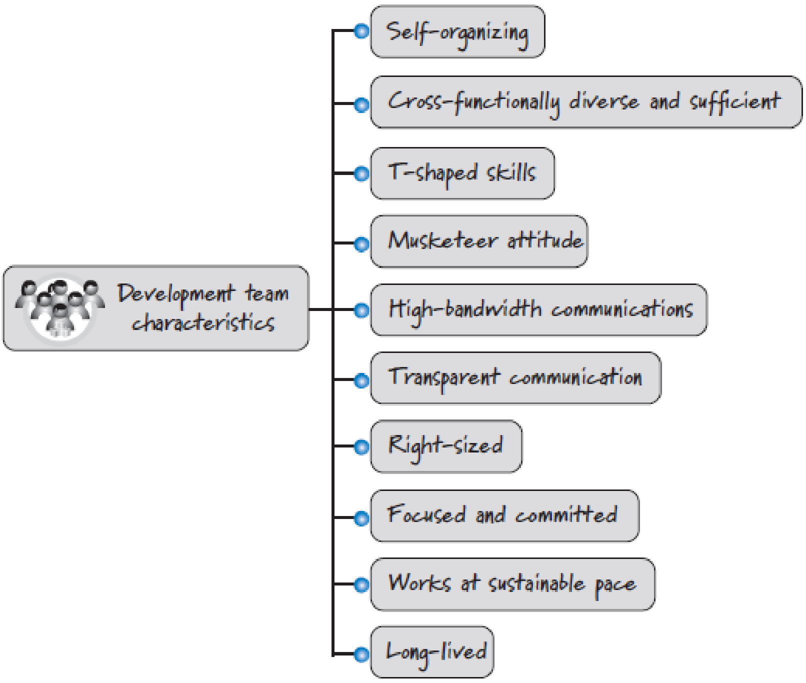


Figure 13. Development team characteristics.

4.2.5. Activities and artifacts

4.2.5.1. Overview

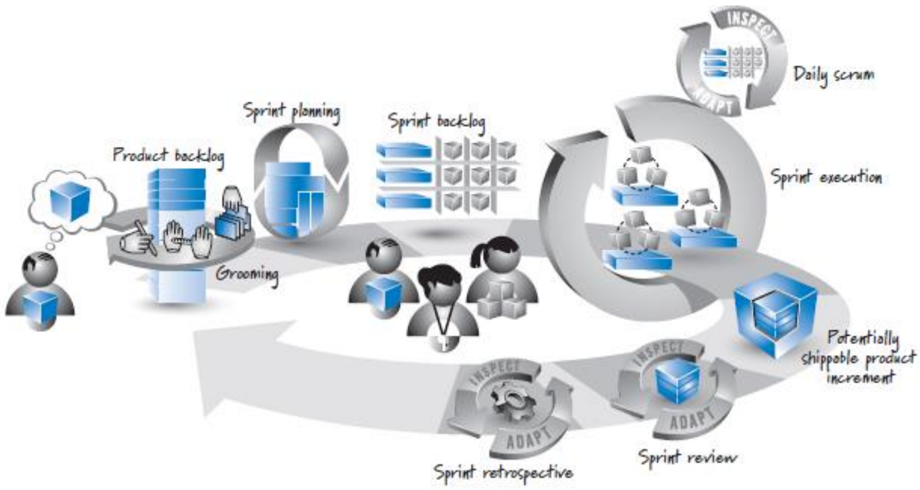


Figure 14. Scrum framework

The sprint starts in the Figure 14 on the left side, working clockwise around the main looping arrow.
The product owner has a vision of what is necessary to create (big cube). Because the cube can be large, through an activity called grooming it is split into a set of features that are collected into a prioritized list called product backlog.

A sprint starts with sprint planning, encompasses the development work during the sprint (called sprint execution), and ends with the review and retrospective. The sprint is represented by the large, looping arrow that dominates the center of the figure. The number of items in the product backlog is likely to be more than a development team can complete in a short-duration sprint. For that reason, at the beginning of each sprint the development team must determine a subset of the product backlog items it believes it can complete, this activity is called Sprint planning.

Therefore, the development team has made a reasonable commitment, the team members create a second backlog during sprint planning, called the sprint backlog. The sprint backlog describes, through a set of detailed tasks, how the team plans to design, build, integrate and test the selected subset of features from the product backlog during that particular sprint.

Next sprint execution, where the development team performs the tasks necessary to realize the selected features. Each day during sprint execution, the team members help manage the flow of work by conducting a synchronization, inspection and adaptive planning activity known as the daily scrum. At the end of sprint execution the team has produced a potentially shippable product increment that represents some, but not all, of the product owner's vision.

The Scrum team completes the sprint by performing two inspect-and-adapt activities. In the first, called sprint review, the stakeholders and Scrum team inspect the product being built. In the second, called the sprint retrospective, the Scrum team inspects the process being used to create the product. The outside of these activities might be adaptions that will make their way into the product backlog or be included as a part of the team's development process.
At this point the Scrum sprint cycle repeats, beginning anew with the development team determining the next most important set of product backlog items can be complete.

4.2.5.2. Product backlog

The product backlog is defined as a prioritized list of project requirements with estimated times to burn them into completed product functionality. The items can be functional or non-functional requirements as well as issues and technical improvements. The product owner is ultimately responsible for determining and managing the sequence of this work and communicating it in the correct priority (see Figure 15).

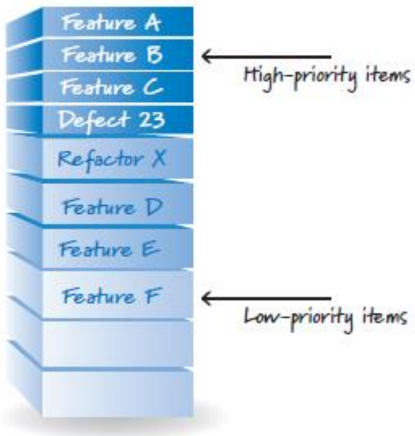


Figure 15. Product backlog.

4.2.5.3. Sprints

The work is performed in iteration cycles of up to a calendar month called sprints. The completed work in each sprint should create something of tangible value to the customer or user.

Sprints are timeboxed (see Figure 16) so they always have a fixed start and end date, and generally they should all be of the same duration. It is not allow any goal-altering changes in scope o personnel during a sprint.

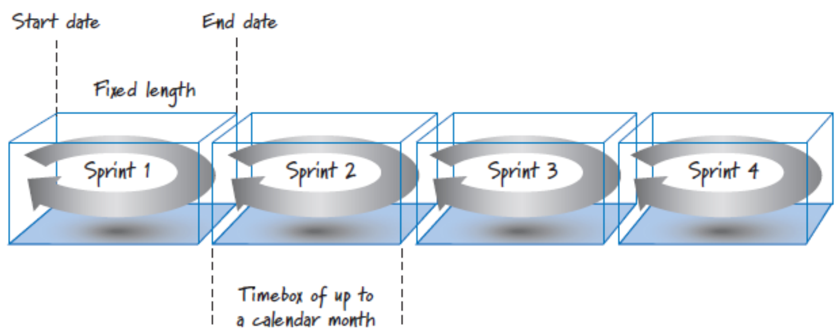


Figure 16. Timeboxed sprints.

4.2.5.4. Sprint planning

During the sprint planning, the product owner and the development team agree on a sprint goal that defines what the upcoming sprint is supposed to achieve, using this goal the development team reviews the product backlog and determines the high-priority items that the team can realistically accomplish in the upcoming sprint while working at a sustainable pace.

To acquire confidence in what it can get done, many development teams break down each targeted feature into a set of tasks. The collection of these tasks, along with their associated product backlog items, forms a second backlog called the sprint backlog (see Figure 17).

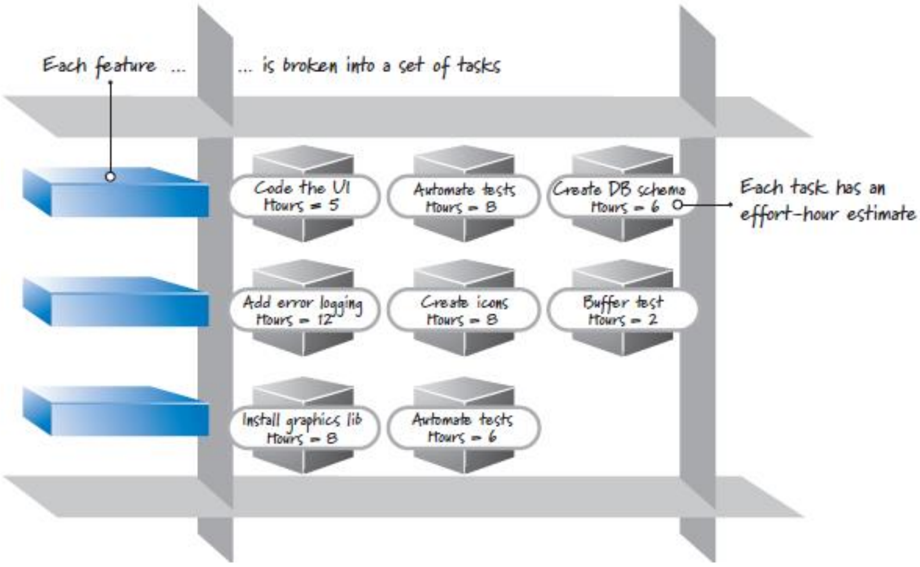


Figure 17. Sprint backlog.

The development team then provides an estimate of effort required to complete each task. Breaking the product backlog items into tasks is a form of design and just-in-time planning for how to get the features done.

To perform sprints of two weeks to a month in duration must be complete sprint planning in about four to eight hours. A one-week sprint should take no more than a couple of hours to plan. Then the approach is select a product backlog item, break the item down into tasks, and determine if the selected item fit within the sprint, if it does fit and there is more capacity to complete work, repeat the cycle until the team is out of capacity to do any more work.

4.2.5.5. Sprint execution

Once the Scrum team finishes sprint planning and agrees on the content of the next sprint, the development team, guided by the ScrumMaster's coaching, performs all of the task-level work necessary to get the features done (see Figure 18), where done means there is a high degree of confidence that all of the work necessary for producing good-quality features has been completed.

Exactly what tasks the team performs depends of course on the nature of the work. Nobody tells the development team in what order or how to do the task-level work and the self-organize in any manner they feel is best for achieving the sprint goal.

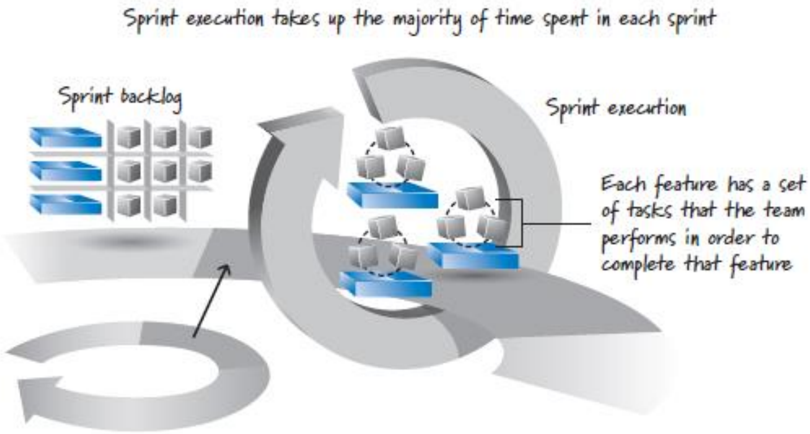


Figure 18. Sprint execution.

4.2.5.6. *Daily Scrum*

Each day of the sprint, ideally at the same time, the development team members hold a timeboxed (15 minutes or less) daily scrum (see figure 19). This inspect-and-adapt activity is sometimes referred to as the daily stand-up because of the common practice of everyone standing up during the meeting to help promote brevity.

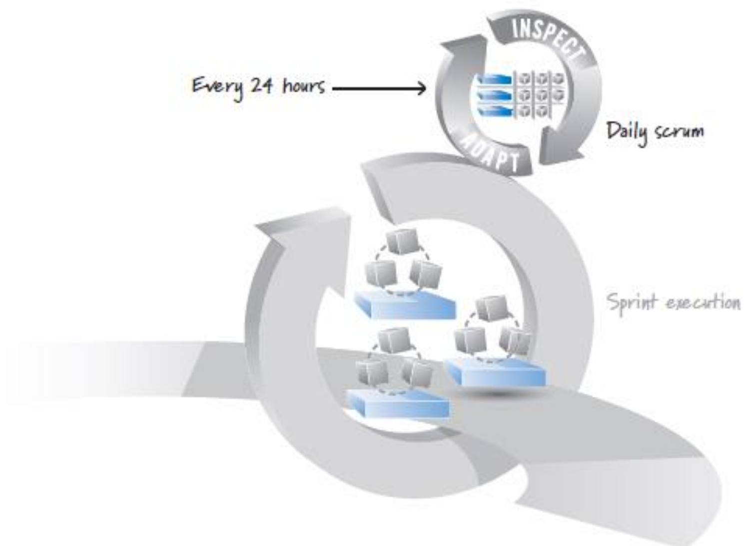


Figure 19. Daily scrum.

A common approach to performing the daily scrum has the ScrumMaster facilitating and each team member taking turns answering three questions for the benefit of the other team members:

- What did I accomplish since the last daily scrum?
- What do I plan to work on by the next daily scrum?
- What are the impediments or obstacles that are preventing me from making progress?

By answering these questions, everybody understands the big picture of what is occurring, how they are progressing toward the sprint goal, any modifications they want to make to their plans for the upcoming day's work, and what issues need to be addressed. The daily scrum is essential for helping the development team manage the fast, flexible flow of work within a sprint.

The daily scrum is not a problem-solving activity. Rather, many teams decide to talk about the problems after the daily scrum and do so with a small group of interested people. It can be useful to communicate the status of sprint backlog items among the development team members. Mainly, the daily scrum is an inspection, synchronization, and adaptive daily planning activity that helps a self-organizing team do its job better.

4.2.5.7. Done

In Scrum, the concept of done means a potentially shippable product increment. This definition specifies the degree of confidence that the work completed is of good quality and is potentially shippable (see Figure xx).

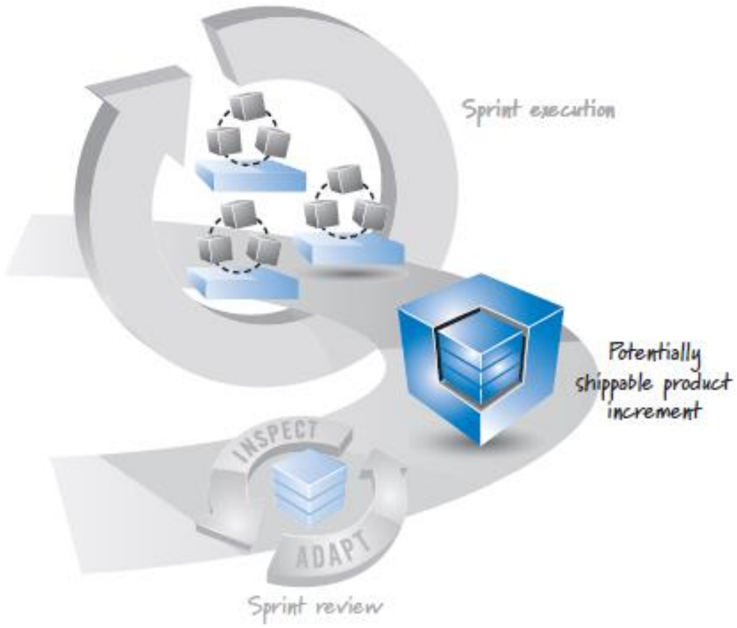


Figure 20. Potentially shippable product increment.

To be clear "potentially shippable" does not mean that what got built must actually be shipped. Shipping is a business decision, which is frequently influenced by things such as "Do we have enough features or enough of a customer workflow to justify a customer deployment?"

Potentially shippable is better thought of as a state of confidence that what got built in the sprint is actually done, meaning that there isn't materially important undone work that needs to be completed before ship the results of the sprint, if shipping is our business desire.

4.2.5.8. Sprint review

The goal of this activity is to inspect and adapt the product that is being built (see Figure 21). Critical to this activity is the conversation that takes among its participants, which include the Scrum team, stakeholders, sponsors, customers, and interested members of other teams. The conversation is focused on reviewing the just completed features in the context of the overall development effort. Everyone in attendance gets clear visibility into what is occurring and has an opportunity to help guide the forthcoming development to ensure that the most business-appropriate solution is created.

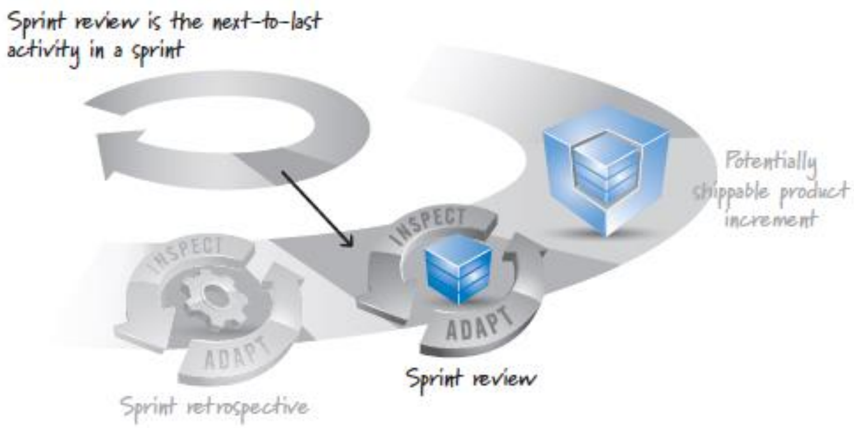


Figure 21. Sprint review.

A successful review results in bidirectional information flow. The people who aren't on the Scrum team get to sync on the development effort and help guide its direction. At the same time, the Scrum team members gain a deeper appreciation for the business and marketing side of their product by getting frequent feedback on the convergence of the product toward delighted customers or users. The sprint review therefore represents a scheduled opportunity to inspect and adapt the product.

As a matter of practice, people outside the Scrum team can perform intra-sprint feature reviews and provide feedback to help the Scrum team better achieve its sprint goal.

4.2.5.9. Sprint retrospective

The second inspect-and-adapt activity at the end of the sprint is the sprint retrospective (see Figure 22). This activity occurs after the sprint review and before the next sprint planning.

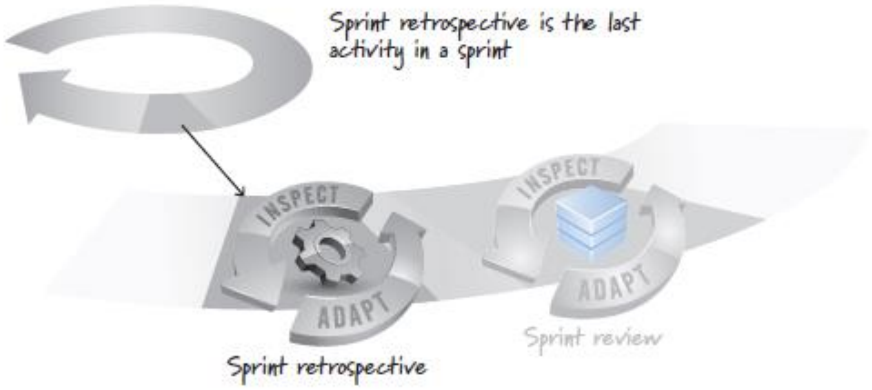


Figure 22. Sprint retrospective.

Whereas the sprint review is a time to inspect and adapt the product, the sprint retrospective is an opportunity to inspect and adapt the process. During the sprint retrospective the development team, ScrumMaster, and Product Owner come together to discuss what is and is not working with Scrum and associated technical practices.

The focus is on the continuous process improvement necessary to help a good Scrum team become great. At the end of a sprint retrospective the Scrum team should have identified and committed to a practical number of process improvement actions that will be undertaken by the Scrum team in the next sprint.

After the sprint retrospective is completed, the whole cycle as repeated again, starting with the next sprint-planning session, held to determine the current highest-value set of work for the team to focus on.

4.3. Scrum Values

All work performed in Scrum needs a set of values as the foundation for the team's processes and interactions. And by embracing these five value (see Figure 23), the team makes them even more instrumental to its health and success.

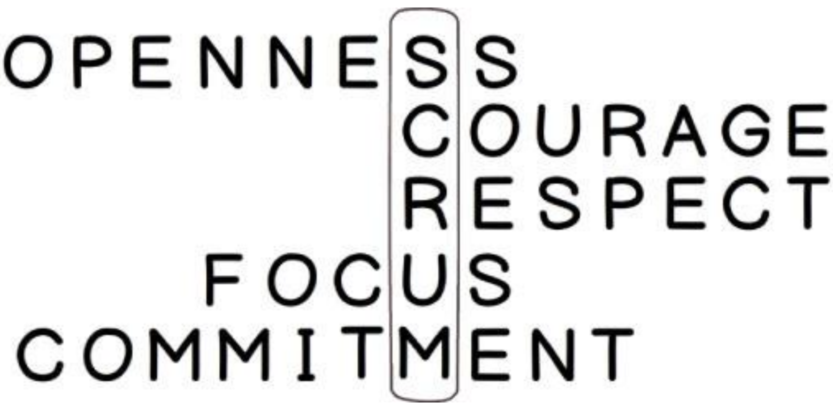


Figure 23. Scrum Values.

- **Commitment:** It is a promise or a serious pledge made by a person to do something for somebody. A commitment says, “You can count on me” and means dedicating yourself to work responsibly.
- **Focus:** It is the ability to give undivided attention to a problem or tasks, without interruption, until that problem is solved or the task is complete.
- **Openness:** It is an honest way of behaving that promotes to accept new ideas, methods and changes.
- **Respect:** It is an admiration for a person, and more specifically the qualities, characteristics, or talents of that person.
- **Courage:** It is the ability to confront fear. A bit stronger than risk-taking, which is calculated undertaking, courage is the decision that you make in order to face something head on, whether or not it knows the consequences.

5. Main Case Study: Project Mi5

In this chapter will be presented the contents of the main case study along with a short introduction of the project Mi5 and an extended explanation of the mechatronic product development divided in four sub-chapters.

5.1. Introduction to Project Mi5

The project Mi5 is an educational project with international cooperation, which consist in the implementation of modern engineering tools in the food industry. The main objective of this project is develop a modular and intelligent production-system that consist of several production modules, a central transportation system and a decentralized production control. Each has its own intelligence and autonomy, with the capabilities to produce products and work cooperatively with other modules through the production control and the transport system (see Figure 24).

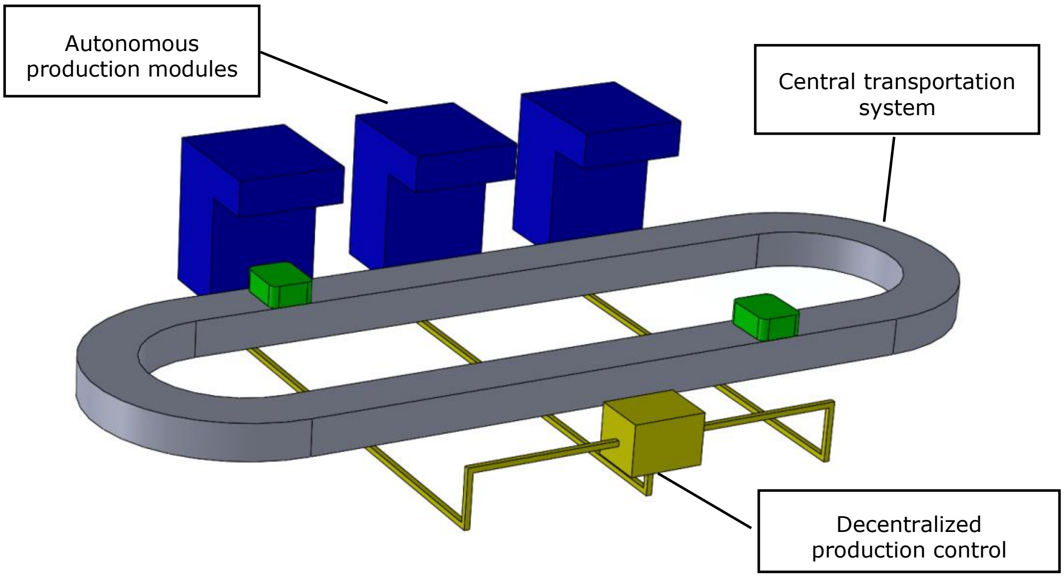


Figure 24. Modular system.

To make this complex development process more understandable, a realistic development project is put into reality in an engineering demonstrator (see Figure 25), in the production line, modern communication, simulated systems and processing technology is used. This helps to display, how a combination of current standards can realize the idea of *Industry 4.0* (Fourth Industrial Revolution).

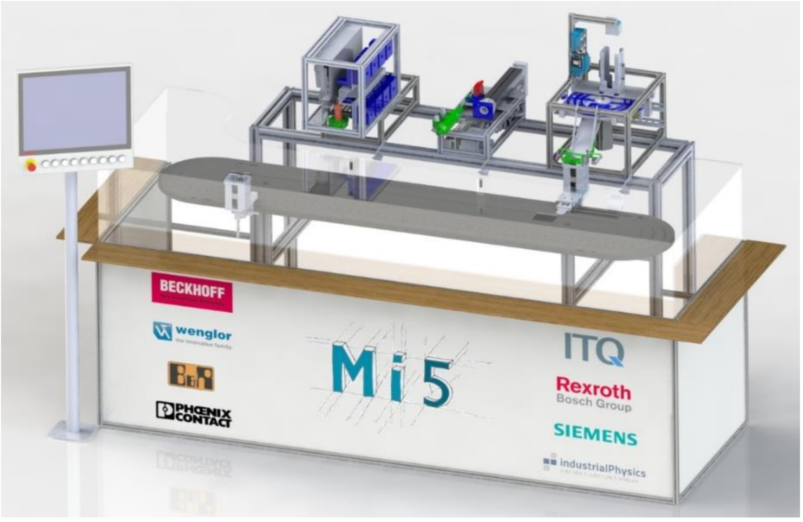


Figure 25. Demonstrator.

In this context, a group of 20 students at ITQ GmbH in Munich, developed a modular demonstrator through advanced engineering methodologies and highly interdisciplinary environment. In addition to this, a new cooperation with UVIC-UCC, consolidated a new project group of 5 students in Vic with the challenge to develop a module adaptable to the demonstrator.

5.1.1. Scrum at UVIC-UCC

This project is the first start-up team using Scrum methodology at UVIC-UCC in February 2015. The motivation of the author to implement this methodology, remains from his experience of the Erasmus internship in ITQ GmbH, which he had the first contact with methods of development, experimenting with four Scrum teams. He realized the potential of this development method based in short deliveries and he experimented improvements in his performance and work development.

Now there are two students from the UVIC-UCC doing an internship at ITQ GmbH, whose are in the same situation that was the author, getting in touch with a new system work, methods of development and cooperation between several teams. The intention of ITQ GmbH is continue this international cooperation, promoting more projects for students, making a closer relation between company and university, as the actual relation with TUM (Technical University of Munich).

In addition, some researchers from UVIC-UCC are interested to apply this methodology in subjects project-based such as integrated projects, this fact catches the attention of the author to continue researching with agile methodologies and helping the university community to raise the knowledge level.

5.1.2. General Start-Up and Education

According to the Agile ITQ Manager when starting up a Scrum team at ITQ, management does have not enough time training when starting teams and want to cut down the time spent on teaching the teams. Then the initial process to educate the team has had to be shortened by half, from what is generally recommended and the timeframe training is shorter than the team would like it to be. This means that there is less training in the Agile development.

To starting up in teams, the sessions starts on Monday morning and finish on Friday. Along the two days, the theory is given with practical exercises and going through slight presentations. After these two days, the team starts the Sprint planning and the first iteration starts. At the end of this session, the team members are invited to think about what they want to achieve with this methodology, what are the expected results and the motivations to take benefits of the new method system.

Basically, this is a short description of a Scrum start-up, however can be explained in longer periods or different ways to transfer the Scrum knowledge, it always depends on a lot of factors, such as quantity of people, team response, how motivated is the team, and so on successively.

5.1.3. Experiences from other Mechatronic Development teams

In conversations with the ScrumMaster and the project leaders of the team Munich the following observations were elucidated.

The ScrumMaster mentioned the missing role of the Product Owner as a weakness in the project development, who is supposed to push the team, and is the responsible of the planning prioritization. The connection to the market, sponsors and stakeholders are affected without a Product Owner. Therefore the ScrumMaster suggested to manage the situation without the Product Owner because for educational purpose like this kind of project it is not strictly necessary.

He also said that without the Product Owner, all the responsibilities are supported by the ScrumMaster, therefore the ScrumMaster become a “Coaching Leader” and he is the responsible to make the critical decisions and prioritizes the features and work items for the team.

The project leaders noted that the ScrumMaster is more complicated role that it seems, especially in such situations where is taking part in the development process. Thus, if the ScrumMaster participates in the development process as other team member, he should be high-qualified, otherwise it can cause failures in the project management. In addition, the ScrumMaster of the team Munich suggested to focus with the role and not participate in the development.

5.2. The Case Study Introduction: Scrum in a Mechatronic product development

In this case study is investigated the integration of Scrum methodology in a Mechatronic product development. The structure of the case study is split up into four iterations, where each iteration correspond to one phase of the project. The length of this study was five months.

In the first iteration was conducted the Scrum start-up, where the team was introduced to Scrum methodology and it was observed their development. In the second iteration was defined the design evaluation and technical features of the machine. In the third iteration was started the prototyping phase. In the fourth iteration was develop the functional prototype and the corresponding validation testing (see Figure 26). During the four iterations the author was observing the process as a ScrumMaster and participating as member of development team.

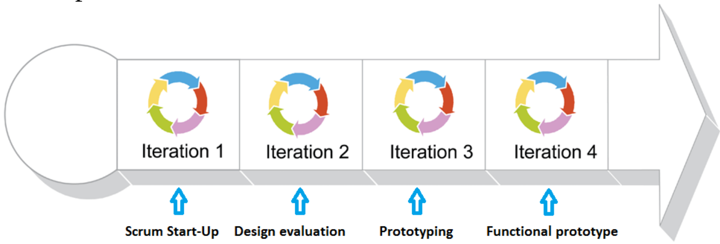


Figure 26. Development iterations.

The team evaluated in this case study is the development team in Vic. Concerning project organization, the team consisted of five members, two mechanical designers, a programmer, an agile developer and a web designer (see Figure 30). The team disciplines involved were Mechatronic Engineering, Multimedia Science and Industrial Electronic and Automation from the UVIC-UCC. In addition, two members of the team Munich were participating via teleconference as a consultants.

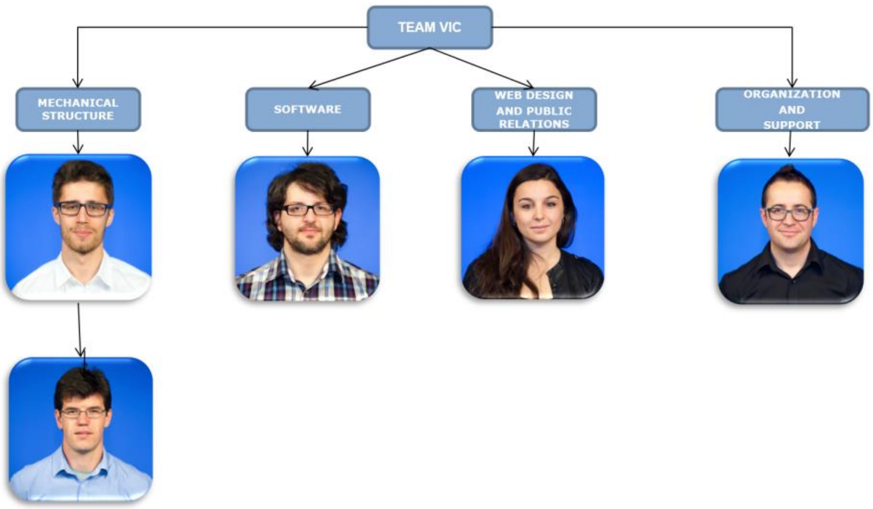


Figure 27. Project organization.

In addition, it was evaluated the experience of each the team member through informal interviews. Regarding the results of the interviews, two members had project-based experiences, other member had experiences working individually as official engineer and the other member did not has experiences in projects, but she affirmed she had experience working in pairs, which is demanded in most of the assignments. After that, the team Vic started the first iteration.

5.3. First iteration

In this section will be presented the first phase of the project development. The contents consist of the Start-Up, the first days of Scrum, extension of the first iteration, modification, the results, observation and discussions.

5.3.1. Scrum start-Up

The project started on a Wednesday in February. Before, there was a week of individual interviews, where the author was explaining and interviewing to the new team members the project planning, the educational aspects, and the objectives.

Therefore, a week after, the kick off meeting started. The findings found will be presented in this section.

5.3.2. First day

The Kick off meeting started on Wednesday 25th, it was split up in two sessions, because some members were not able to attend at this time. Thus, in the morning was the first session and in the afternoon the second session.

In both sessions, the ScrumMaster introduced to the development team the following concepts:

- Industry 4.0
- Modularity
- Modern education

After the introduction of the main concepts, the goals of the projects using Scrum were described with the help of the ScrumMaster. These were as follows:

- Develop a food module
- Promote team cooperation.
- Learn methods of development
- Improve project organization
- Knowledge distribution
- Roles distribution
- Make problems more visible
- Make progress more visible

Next step was start the first activity called “6-3-5 Method“. In brief, it consist of 6 participants supervised by a moderator who are required to write down 3 ideas on a specific worksheet within 5 minutes. The outcome after 6 rounds, during which participants swap their worksheets passing them on the team member sitting at their right, is 108 ideas generated in 30 minutes. Before start this activity, the ScrumMaster explained different use cases in the food industry, according to the topic of the project. The team with the help of the Scrum Master decided to choose the packaging technology.


Figure 28. 6-3-5 Method.

The results of the “6-3-5 method” (see Figure 28) were based in ideas of packaging, where the ideas were split up in mechanical system structure, electronic devices and software functionalities. The ScrumMaster noted few experience in methods between the team members, this fact have an important impact in the results.

Concerning the mechanical system structure, it was defined to develop the following parts:

- System to roll and unroll the material.
- Mechanism to catch and leave products.
- Transport system.
- Mechanism to cut the material in layers.
- Mechanism to model the material from layers to packages.
- Printing system to decorate the package.

Regarding the electronics, it was defined to implement the following devices:

- Camera sensor to take pictures of the customers.
- Image sensor to check the quality of the packaging.

Referent the software functionalities, it was defined the following features:

- Customize the format of the boxes by the interaction of the customers with the HMI.
- Register users in order to know preferences and recollect important data to improve the system.
- Order products via one smartphone app, to give access everyone.

The general feeling when observing the team was highly motivating for them, as they never used this kind of this technique to think cooperatively in a team.

5.3.3. Second day

The observation of this study started on the second session, on the 3rd Tuesday in March, after the kick-off meeting. At the start of the day, the team and the ScrumMaster reviewed the points of the day before. The ScrumMaster presented the summary of the brainwritting breaking the global feature "Packaging" in smaller sub-features in a diagram format (see Figure 29).

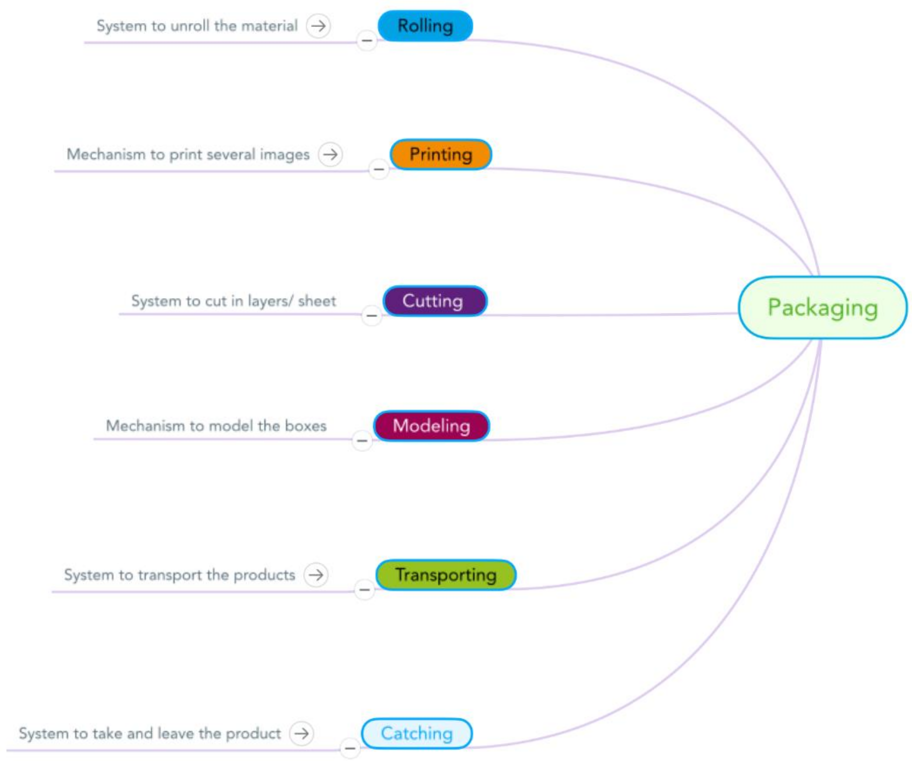


Figure 29. Diagram summary of brainwritting.

The ScrumMaster suggested start to develop the system, doing sketches by hand. The activity was perform by choosing the sub-features of the packaging, where the tasks were assigned by the team preferences and each team member chose the favorite subtopic.

The first prove of iteration planning was presented, as it is observed in the Figure 30, the four types of meetings were explained in order to plan all the tasks in the sprint planning, do all the amount of work during the daily planning, check what are the completed tasks in the Sprint review and finally check all the process in the Sprint retrospective.

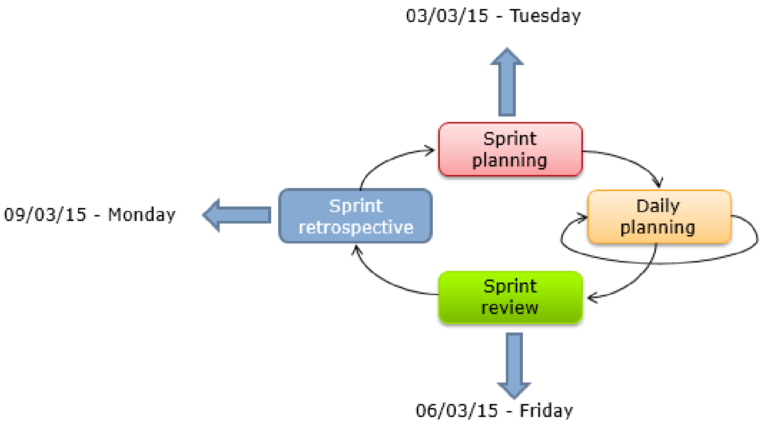
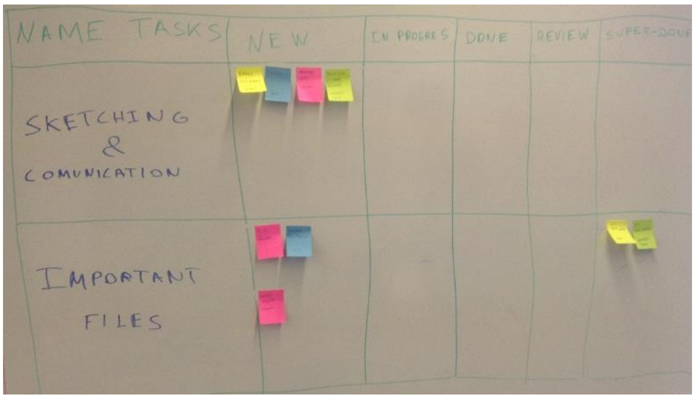


Figure 30. First planning.

Therefore, to start the sprint planning, it was drawn a table, as an example of the sprint backlog, which the global features were sketching, communication and important files. The ScrumMaster included in this sprint backlog the important files to make an easy understanding of the tasks responsibility, however these files were not part of the development. As it is observed in the Figure 31, the team stuck all the sub-features in the new column, which each color represents a team member tasks. In this case two sub-features were completely done.



MAIN TASK	NEW	IN PROGRESS	DONE	REVIEW	SUPER-DONE
Sketching	Rolling-Matheus Printing-Matheus Cutting-Sergi Modelling-Gulu Transporting-Carles Catching-Carles				
Communication	Daily Document-Joan Website post-Laura				
Documents	Skill profile-Laura Skill profile-Gulu Skill profile-Carles Skill profile-Sergi				Skill profile-Joan Skill profile-Matheus
	CV Europass-Laura CV Europass-Gulu CV Europass-Carles CV Europass-Sergi				CV Europass-Joan CV Europass-Matheus

Figure 31. Sprint Backlog.

At the end of the session, the ScrumMaster noted that the team was excited with the tasks assignment and also the methodology had a good impact, even though the high demanding requirements of the deliveries.

5.3.4. Sprint review

The observation of this study started on the third session on the 6th Friday, three days after the sprint planning. The development team presented all the amount of work done. In this session the Scrum master acted as Stakeholder as well. The development team presented some difficulties to deliver their assigned tasks. Therefore just one team member and the ScrumMaster presented results and decided to give two days more, in order to finish the tasks.

The ScrumMaster presented to the team a model of technical diary, instead of the daily stand-up, because it was not possible meet every day with everybody.

The technical diary was the first solution to face a problem of the daily meetings, composed by three parts. The first section was designed for the team member doubts, where everybody could write questions and answers. The second section was designed for specify changes in the development. The third section was an application of the three stand-up questions such as what I did today? (Done); what problems I had to face? (Problems); what is the planning for tomorrow? (Planning), as it is observed in the Figure 32.

1. Questions:

1. **Q**<jm>: Everybody understand this document?
   A:

2. Specification Changes:

When? (Date)	What? Specification	Who? (Author)	Why? (Reason)
03.03.15	Establish a fix meeting day (Monday at 10:00)	Joan	For better organization
09.03.15	Photo day (Monday at 11:00)	Everybody	For the website, CV and communication.

3. Scrum Diary:

Dear Diary, today I did .... <3

Tuesday 03.03.2015 - Meeting day

Name	Done	Planning	Problems
Joan	Daily document	Write an article for MI5. Prepare interview. Establish contacts. Help team members. Ask for the simulation license.	Explain better the features
Laura			
Guiu			
Matheus			
Carles			
Sergi			

Figure 32. Technical diary.

Concerning the development, the team presented the sketches of all the parts of packaging machine. In the Figure 33 are represented two examples of transport systems, which could be implemented to transport the food products from the demonstrator to the packaging module.

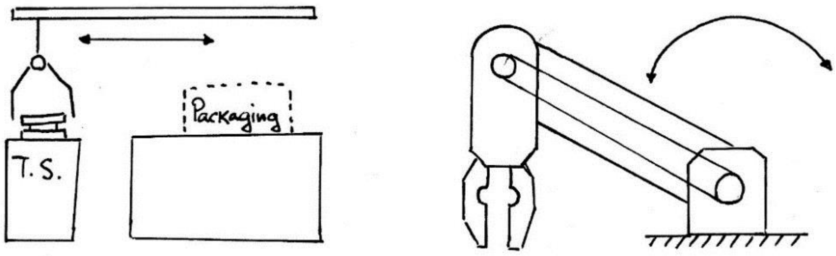


Figure 33. Transport systems.

In the Figure 34 is represented one example of a pick-up system, which is a part of the transport system. The objective of this system was pick up and leave the food product.

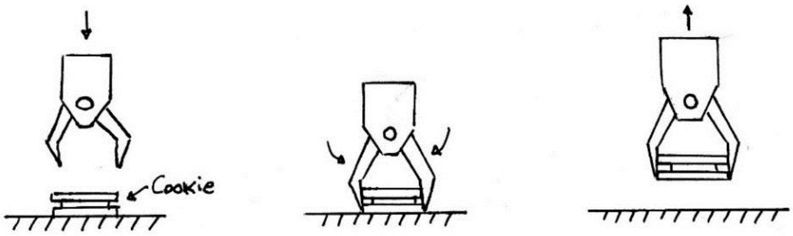


Figure 34. Pick-up system.

In the Figures 35 and 36, are represented two examples of modelling systems. These system were drawn with the possibility to use carton or plastic as a primary matter to package the food products.

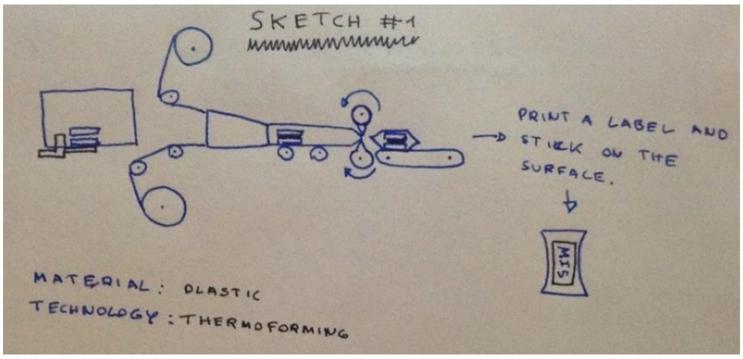


Figure 35. Modeling system handling plastic.

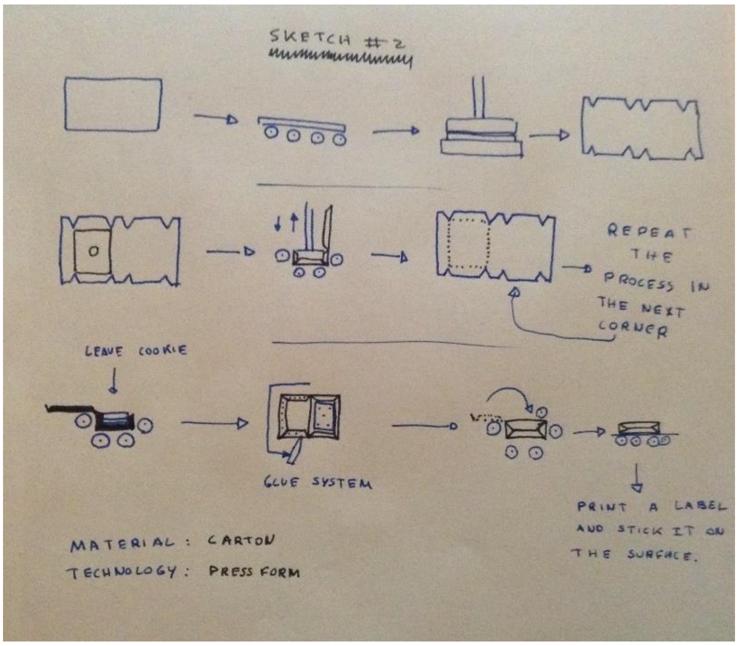


Figure 36. Modelling system modeling carton.

5.3.5. Sprint retrospective

The observation of this study started on Friday 9th, three days after the sprint review. In this session the ScrumMaster presented a grid structure (see Figure 37) in order to identify the possible issues during the development of the tasks.

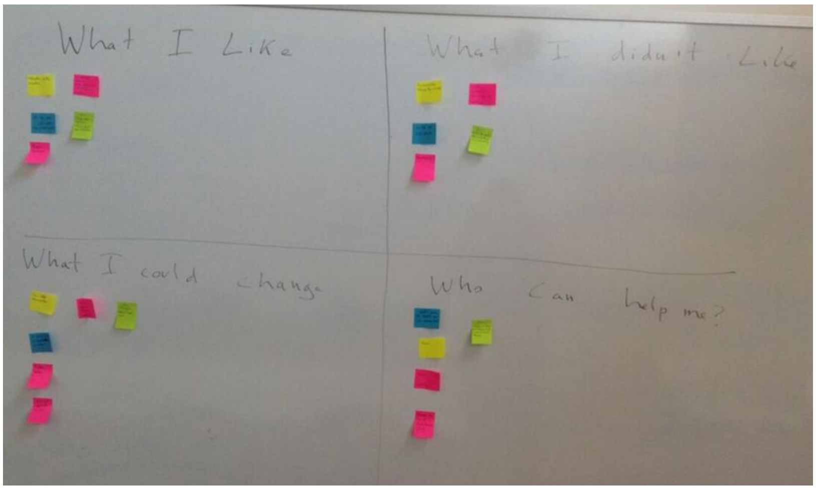


Figure 37. Grid structure.

According to the results of the grid structure, the ScrumMaster noted that some members were not used to work individually, especially under the pressure of a deadline. Others members presented difficulties to do tasks, which were not inside in their specialty. One member presented lack of communication in problem-solving environment and difficulties to focus with the tasks. The conclusion of ScrumMaster was that the team needed more time to adapt in scrum methodology.

Concerning the team opinions of the Grid structure method, it was noted a good feeling, analyzing what were the parts to improve and the way to solve their problems through the cooperation of the other team members.

Therefore, with the positive feedback of the team about the iterative process, the ScrumMaster suggested to extend the first iteration until the 19th in April, in order to give more time-adaption to the team.

5.3.6. Extension of the first iteration

As it is mentioned above, the first iteration was extended and the timeframe was from 25th February to 19th April, in total 9 weeks of development. It is important to note that scrum does not recommend extend an iteration more than 4 weeks, however in this especial case the ScrumMaster noted that the student team needed more time to understand all the concepts. The ScrumMaster made the decision to implement a multi-approach technique, giving some theoretical concepts and implement them in a practical way. Therefore, all the sessions contained some theory and practical approaches. The process was performed and controlled by doing checkpoints every Monday, revising the progress, planning and difficulties of the week.

5.3.7. Modifications

During that period, some changes were approved by the team and the ScrumMaster, such as fix the weekly meeting on Monday morning and the inclusion of a new software tool called “Redmine”. This software tool was demanded by the ScrumMaster, in order to give more flexibility to the development team in the tasks management. As it is observed in the Figure 38, all the tasks were introduced in Redmine, taking into account that this figure corresponds to the sprint backlog, where on the left first column there are the global features and the rest of columns represents the different progress parts such new, in progress, done, review and super-done, in which are the sub-features of the global tasks. Each color represents one team member.

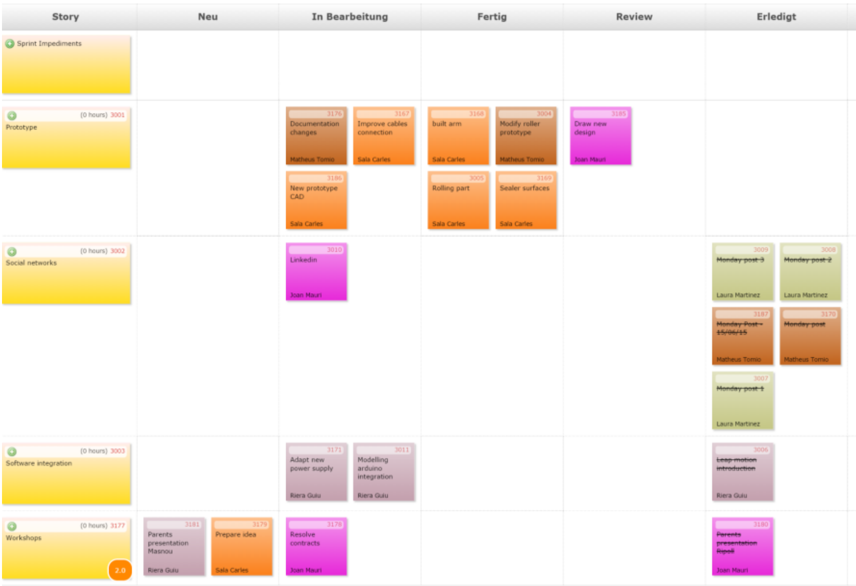


Figure 38. Backlog in Redmine.

The development team worked up with this software tool, because the development team had access to see how tasks were flowing through the different states. It is important to note that each member had the possibility to include new tasks individually, giving more flexibility to plan and work. Moreover this tool provided information to the different teams about the overall progress of project development.

Concerning the technical diary, it was presented a new modification of the Daily Scrum, instead of have three sections like the old model, this model presented one section in a daily table format, as it is observed in the Figure 39. In addition this new model presented detailed information about the tasks management, such as the name of each member and each task, the typology of tasks, the time spent in each task, the percentage of the tasks and the typical sections of the standard daily scrum (Obstacles, planning and done).

Name	Task	Obstacles	Planning	Type of tasks	Time	Done?	%
	Meeting	Burocracy obst.	Write article	Communication	140	Yes	100
	Query workstation		System specification	Organization	60	No	30
ScrumMaster	E-mailing		Update social networks	Organization	25	No	65
	Meeting		Design first CAD model	Communication	140	Yes	100
Member A	Lego testing	SVN config.	Test with different Lego prototypes and plastics	Development	60	No	20
Member B	Meeting	Windows acces		Communication	140	Yes	100
Member C	Meeting	Computer problems	Make stylish changes requested in meeting. Read information about OpcUA and Socket.io	Communication	140	No	30
	Meeting		Finish this week	Communication	140	Yes	100
	Printing		Find a printer	Development	120	No	30
Member E	Research	Svn configuration, define design difficulties		Research	60	No	40

Figure 39. Technical diary.

As additional approach, it was considered the data of the columns "Type of tasks" and "Time" to control the quantity of time spent in a graphical and numerical format. This approach was completely helpful to analyze the weekly results with the time spent and decide to do changes or not.

It is important to note that this model of technical diary was the method to collect data commented in the chapter 3.5 called "Documentary Analysis".

5.3.8. End of the first iteration

In the end of the iteration were presented the final results. Firstly it was presented the sketches integration of two possible packaging machines as it is observe in the Figures 40 and 41.

The figure 40 represents the first model of packaging machine, which is handling plastic material in film format. It is possible to observe all the necessary parts such as rolling, transport and pick up, feeding, modelling, sealing and printing system according to the results of the brain writing (see Figure xx). It is important to note that this system is a continuous system.

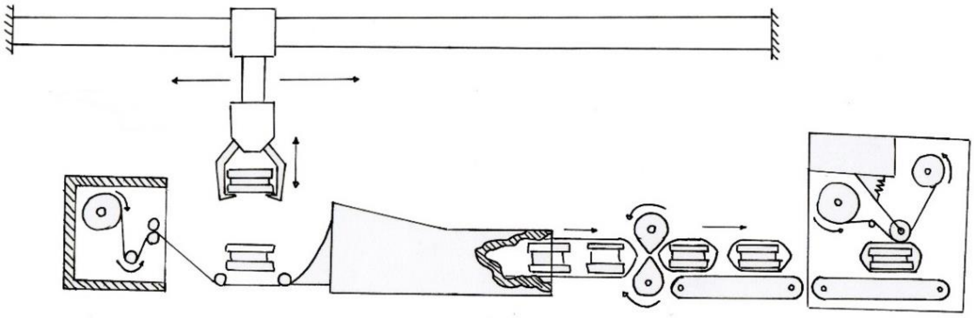


Figure 40. Sketch handling plastic.

In the Figure 41 is shown the second model of packaging machine, which is handling carton material in box format. This is the result of the Figure 36, taking into account that the problem of modelling the carton was solved using pre-fabricated boxes, therefore the process was reduced to a machine box-modeler. It is important to consider that this system is not continuous.

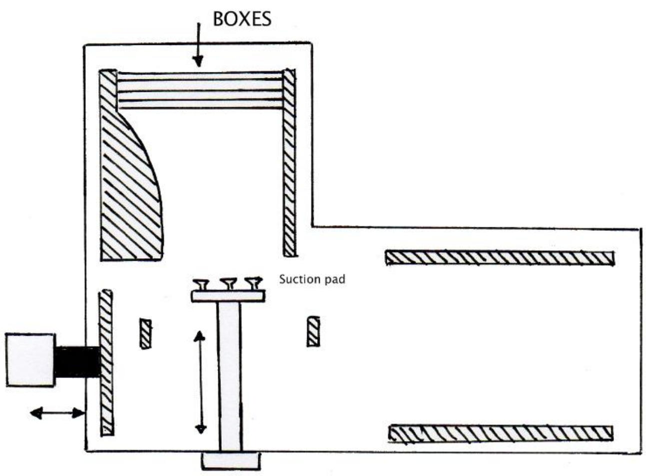


Figure 41. Sketch handling carton.

After the sketches integration, the development team and the ScrumMaster sat together to discuss what of these two models was the most appropriate one to develop. In order to carry out the decision-making, it was performed a session of pros and cons and the results were the following:

Modelling with plastic

PROS

- Package individual
- Continuous system
- Plastic material appropriate for cookie-burger
- Package in few steps
- It’s adaptable to the MI5 Munich requirements

CONS

- Complex to develop
- Difficulties to control the plastic material

Modelling with carton

PROS

- Prefabricated boxes solve the complicated part of modelling
- Carton material is more consistent than plastic

CONS

- Not appropriate for cookies
- Air pressure and vacuum pump is needed
- Requires an especial tape system to close the tabs
- Not adaptable for individual production
- Discontinuous system

Therefore, based in the results and the opinion of the Munich team, it was made the decision to choose the modelling with plastic material. The most important reasons were select a continuous system, which allows to wrap individual cookie-burger in few steps and adaptable to the Munich demonstrator.

Finally, the development team proceed to sketch the first model of the packaging module and to design the first CAD draft (see Figure 42). In that moment, the ScrumMaster and one member of the Munich team had the suggestion to use film in bag format instead of in film layer. The main reasons to choose film in bag format were reduce the number of plastic rollers from two to one and the simplicity of the machine steps. It is important to note that the process with film format required to seal four sides and the process with bag format required to seal two sides of the package.

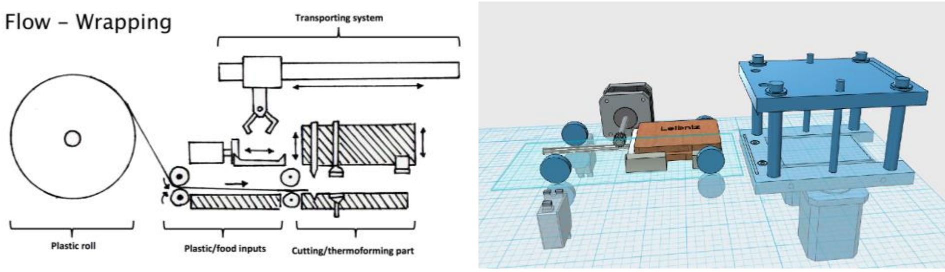


Figure 42. Final sketch handling plastic bags and basic CAD draft.

Another important topic was what plastic typology choose, based in the engineering requirements. In this case, one team member was researching about the materials used to package food products, and the most appropriate was the polypropylene, for its bio-oriented features.

5.3.9. Observations and informal discussions

During the execution of the first iteration, some interesting comments and discussions were took into consideration.

In a discussion with one of the team leaders from Munich, he noted that there were very good ideas arising and creativity during the kick-off meeting even though the few experience of the team in this type of project. He also noted that sometimes the team were a little bit confused about the importance of decision making and the hierarchy, in other words, the last decision was always made by the responsible members of team Munich. This fact caused some troubles in the team Vic, mainly in the motivation to develop systems that in the end were not enough appropriate for the team Munich.

The ScrumMaster as a double viewer, had to moderate the background differences between the authority of the project leaders from Munich and the feelings of the development team in Vic. It is important to note that the author gained experience next to the project leaders, he had to adapt to new culture of working, thanks to this fact, he was able to solve all the international and communication issues.

Concerning the team members' opinions about Scrum, the following points were considered:

- One team member recognize that Scrum provided a good workflow. He noted a slow adaption to the Scrum methodology but effective.
- Two team members noted that the sprint retrospective provided a communication improvement, where everybody can give the feedback of the others without having any conflict.
- During a chat with one team member, it was noted lack of experience in project methodology, he commented that the fact of review the development cause to him pressure, because he didn't know how perform his tasks in success.
- During some sessions the ScrumMaster noted that the team were not used to follow systematic processes,

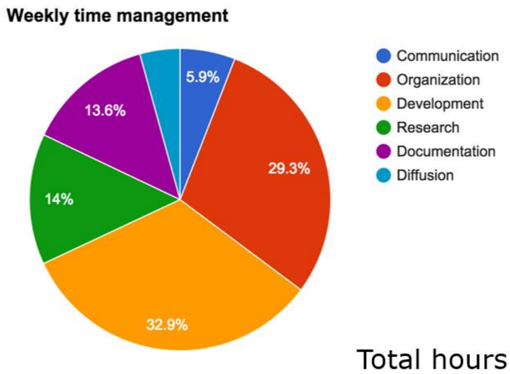
Regarding the work distribution and cooperation, the ScrumMaster invited to the team to think over about the importance of the project progress, in other words, if everyone felt as an important member. Therefore one member confessed that she felt few participative because in her case engineering was not her specialty. Other team member confessed that he felt participative even though he didn't feel making progress, giving reasons of his new environment project-based. The rest of the members confessed to feel highly-participative and making a good progress.

Referent the knowledge distribution, all team members expressed a big increment of acquired knowledge about Scrum and project management, the ScrumMaster noted good global results with the communication in English even the difficulties of some team members.

5.3.10. Documentary analysis

During the first iteration was implemented the technical diary to collect numerical data, therefore in this section it is not possible to analyze and obtain certain results from the beginning of the iteration. Nevertheless, there is one-week of development that demonstrates the following results about the time spent in the different tasks:

Task	Hours
Communication	2,75
Organization	11,58
Development	15,25
Research / Training	6,5
Documentation	6,33
Diffusion	2



Total hours=44,41

Figure 43. Data collection from the technical diary.

As it observed in the Figure 43, the evaluation of the week from 13th to 17th in April was a total of 41 hours. The time spent in communication was 2.75 hours, in organization 11.58 hours, in development 15.25 hours, in research and training 6.5 hours, in documentation 6.33 hours and in diffusion 2 hours. It should be considered that this analysis was very helpful to focus more in one task than another. The conclusion of this analysis determine that there was too many time invest in organization which could be focus more in development, research and training.

5.4. Second iteration

In this section will be presented the second phase of the project development. The contents consist of the procedure, the iteration execution, the sprint review, the sprint retrospective and the documentary analysis.

5.4.1. Procedure

This iteration started on the 20th in April and it finished on the 8th in May, with a length of 3 weeks development. It is important to note that there is no emphasis in the sprint planning, because at that period were not useful planning more than a week. Therefore, every Monday meeting were checked all the tasks and the possible modifications.

5.4.2. Iteration execution

During the iteration execution were presented several results. On one hand, it was presented the first CAD design of roller prototype, which the basic functionality was provide film (see Figure 44 on the left).

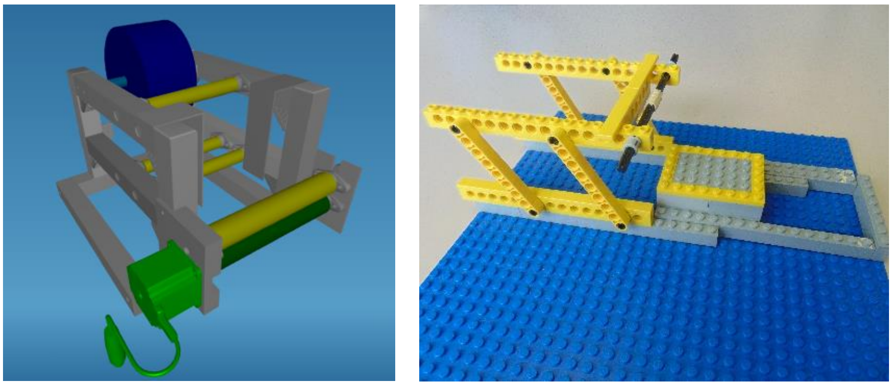


Figure 44. On the left CAD prototype, on the right LEGO prototype.

On another hand, concerning the sealing part, which the basic functionality was modelling the plastic, it was used the LEGO platform for the easy and fast construction, to get the first ideas to handle the plastic, because in this cases the CAD design was uncertain to predict the plastic behavior (see Figure xx on the right). However when this model was built in real pieces was discarded because of its movement complexity.

Therefore, it was built up a system that consist of two wood pieces with one hot wire attached on one wood surface. Thanks to this system was possible to start the testing operations in order to fix the sealing parameters and achieve the desired results. Once the sealing parameters were fixed, the first cookie was sealed with very good results (see Figure 45).

Figure 45. Sealed cookie.

Concerning the construction, it was started the construction of the roller system based in the CAD design (see Figure 46). This construction allowed to determine what were the basic actuators and components, and define the engineering requirements.

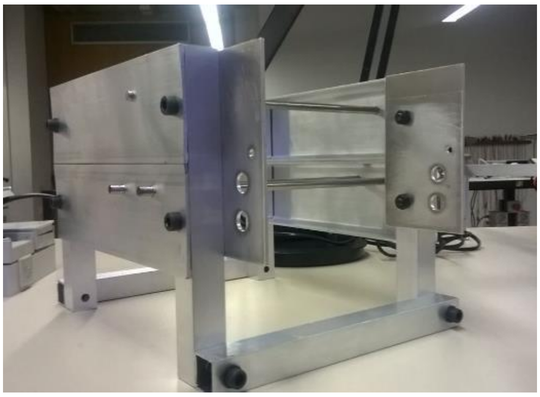


Figure 46. Construction of the roller system.

In this case the important aspects were provide film with the correct tension, discharge the static electricity of the film through the friction of the metal bars and provide appropriate guiding system.

5.4.3. Sprint Review

The observation of this session started on the Monday 4th in April, three weeks later of starting the second iteration (see Figure 47). The development team presented all the amount of work done and in this session were invited to three professors to give their feedback about the product development.

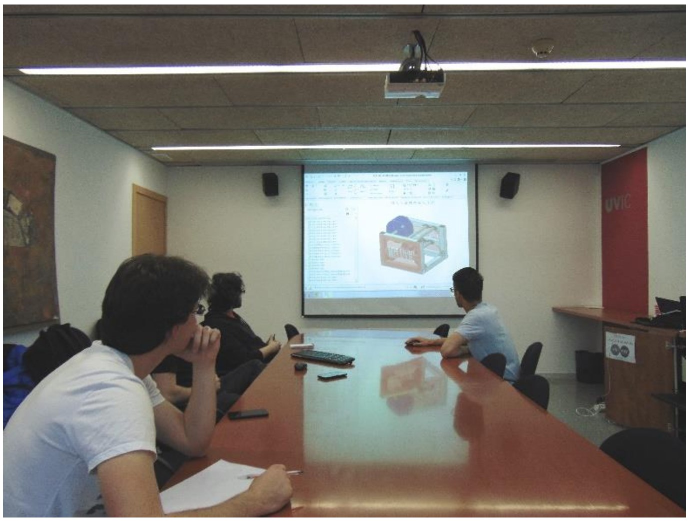


Figure 47. Second Sprint review.

As it was the first time for the professors, they were introduced to this type of session by the ScrumMaster. All team members participated in this activity, showing and explaining the development carried-out by them. It is important to note that the purpose of this session is inspect the product, however in this student project, apart of inspect the product it also was considered important the fact of how each student could carry-out his own tasks, and then explain and defend his results.

Concerning the observations and informal discussions some interesting topics were commented and took it into consideration.

All professors participated asking the following questions:

1. It is the CAD design based in similar machines?
2. Did you choose the actuators based in theoretic calculus?
3. Why this machine has three rollers?
4. Where is the innovation in this project?
5. Have you got any kind of documentation to follow?

Then, the mechanical designer in order to respond the questions 1, 2 and 3, he explain that before to perform the CAD design, the development visited an exclusive packaging fair, where it was looked in depth the requirements specification of the project and it was found one machine with the same requirements but with vertical disposition instead horizontal.

Referent the question 2 he explained that his selection was based for randomly, because it wasn't know at that moment what type of plastic should be used and what pressure was needed to provide film.

Regarding the question 3, he answered that the rollers were important to provide film tension and avoid the static electricity.

The ScrumMaster answered the question 4 and 5, explaining that the innovation in this project is for the following topics:

- Work with international cooperation
- Integrate Agile methodology in mechatronic development and in a student project
- Perform the project with sophisticated software tools
- Develop a machine based in the fourth industrial revolution (Industry 4.0)

In reference of the documentation he said that in the first period of the project Mi5, the team Munich created a website where is possible to follow a clear documentation.

In general, there was a good cooperation environment between teachers and students and it was considered important for the team and the ScrumMaster all the feedback provided by the teachers. In this case, it was suggested to the mechanical designer to do the concerning calculus before to start the design.

5.4.4. Sprint Retrospective

The observation of this session was on the Friday 8th in April, four days later of the sprint review. In this occasion the Scrum Master presented a table, which consist in the following three columns (see Figure 48):

- Keep doing
- Stop doing
- Start doing

In Keep doing the team and ScrumMaster stuck on the white board what was working during execution sprint, which was analyzed what processes should be held. In Stop doing was studied what wasn't working during the execution sprint, which was analyzed what processes should be removed. In Start doing was studied what was possible to change, which was analyzed what it could be change to improve the process.

Figure 48. Second Sprint Retrospective.

The results of this session indicated that write the technical diary allowed to give more information to the team and was helpful for everybody because provided a good thinking structure. In addition, it was commented that the

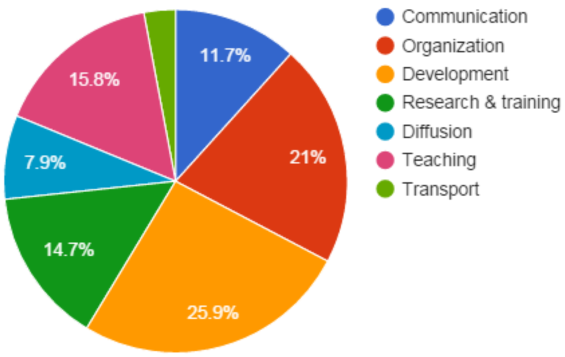
Scrum Master attitude was motivating and it helped to the team to get more involved in the project. This results were evaluated in Keep doing column.

Referent to Stop doing column, the results indicated that one team member presented lack of communication and difficulties to focus with the tasks. Other team member commented that in certain moments the ScrumMaster was controlling too much the progress. Another team member confessed to be few involved and very relaxed. The ScrumMaster confessed that he did too much work outside of his specialty, because some members were not working enough.

Concerning Start doing column, one team members decided to get more committed and productive, other team member decided to be more communicative and ask for help, when necessary. The ScrumMaster decided to do less work, however he keep the control over the progress, arguing the necessary attention to this topic.

5.4.5. Documentary analysis

During the second iteration, the technical diary was used in full-time conditions. Therefore, the data collection allowed to analyze with reliability the time spent in each task and progress.



Task	Hours
Communication	11,83
Organization	21,25
Development	26,24
Research & training	14,83
Documentation	0
Diffusion	7,99
Teaching	16
Transport	3
Total iteration hours	101,14

Figure 49. Technical diary.

According to the results (see Figure 49), in that period the development team spent more time in development 26.24 hours, representing the 25.9% of the time spent. In the second place with 21.25 hours in organization, representing the 21% of the time spent. In the third place with 16 hours in teaching activities (workshop), representing 15, 8% of the time spent considered the most important task to analyze in depth.

The conclusions of this analysis determined that the average of time spent per week was around 33 hours and the ScrumMaster suggested to spend more hours, because 33 hours managed for 5 members, resulted 6 hours per week for each member. The ScrumMaster suggested to arrive until 60 hours/week, resulting 12 hour per week for each member.

Other conclusion was to focus more in development tasks and less with organization. Otherwise the fact to prepare one workshop caused more time than expected. In the communication aspect, the average of hours per week were around 4 hours, it was discussed to decrease the number of hours per week until 3. In this session the development team realize the importance of the data collection and the corresponding analysis, in order to improve over the project development.

5.5. Third iteration

In this section will be presented the third phase of the project development. The contents consist of the procedure, the iteration execution, the sprint retrospective and the documentary analysis.

5.5.1. Procedure

This iteration started on the 11th in May and finished on the 12th in July, with a length of 9 weeks development. It is important to take into consideration that during this iteration one team member left the team, this fact caused work overload for rest of the team. Otherwise the environment became more predictable and the tasks were not changing, therefore it was not necessary perform sessions dedicated for the planning, each member knew exactly what tasks were necessary to do. However the ScrumMaster was controlling the tasks progress through the technical diary and the Redmine.

5.5.2. Iteration execution

During the iteration execution was presented a new solution for the sealing system (see Figure 50). It is important to note that the previous sealing system with LEGO platform was discarded and it was not easily-attained. Therefore, as it is observe in the Figure xx on the sketch, the idea consist in provide film until the right extreme, move the rotatory part following the arrow direction and cover the cookie through hooking the plastic layers.

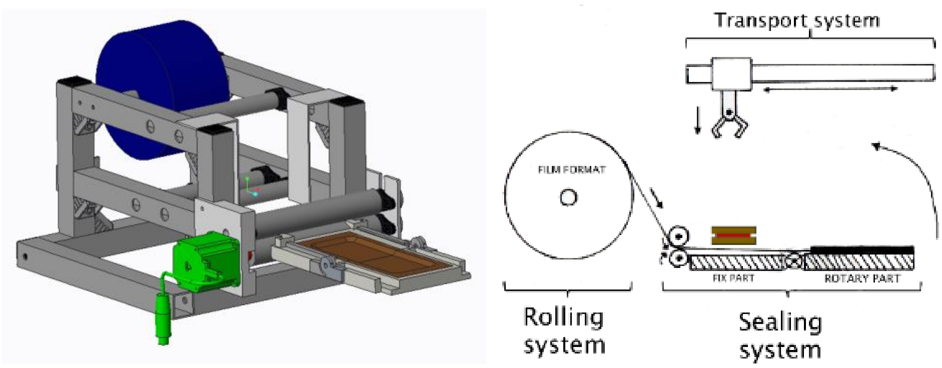


Figure 50. Advanced CAD prototype with the concerning sketch.

Therefore, after sketch and perform the CAD design, it started the rapid-prototyping.

Firstly, it was integrated a stepping motor with an Arduino interface on the roller system, therefore the testing was performed giving the desired results such as provide the correct tension, avoid the static electricity and guide the film straight without fluctuations.

Secondly, the sealer system was built using two different materials, one was wood, which resisted the high temperatures of the sealing operations. The other material was the delrin, which provided good mechanizing properties.

Thirdly, it was integrated a servo-motor on the sealer system, which provided the rotatory movement to the mobile part.

Fourthly, as it is observe in the Figure 51, the sealer and roller system were integrate in one system. It is important to note that this integration was from the mechatronic disciplines, unifying structural mechanics pieces and systems with electronic motors and programming all the mechanisms with the computer platform. In this operation were participating two mechanical designers and the programmer, in order to adjust all the arising issues.

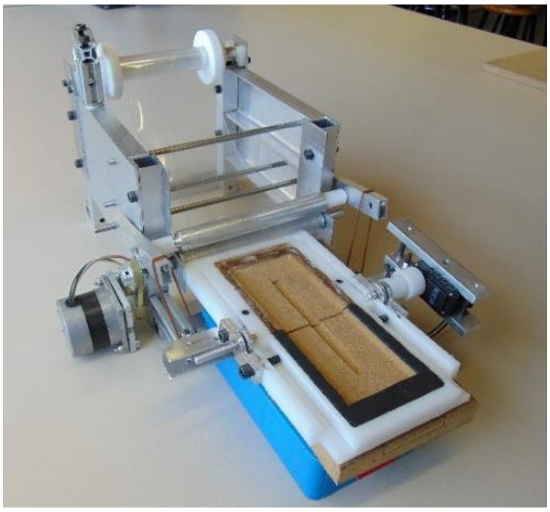


Figure 51. Rapid prototyping.

Finally, it was perform the testing, unfortunately the results were not satisfactory, because when the film had impediments to go forward went through the surfaces. The main reason of this failure was assume that the plastic was enough rigid to go through the surface without any kind of film collector.

Therefore, the mechanical designers tried to change the system with a film collector, nevertheless the system became more complex and more problems were arising.

At that complicated moment the ScrumMaster suggested to ask for external help and he met with head of laboratories from UVIC-UCC, explaining the evolution of the prototype and the engineering requirements. Therefore, the ScrumMaster and him sat together to discuss the possible solutions to face this problem. Thus, the main problem was found in the engineering requirements, the problem was the horizontal disposition of the machine and the ScrumMaster and him suggested together to change for a vertical disposition. It is important to take into consideration that the horizontal disposition was a requirement commented by one team member from Munich, with the logical reason to package cookie-burgers (cookies with cream) the most appropriate way to do it without any kind of risk is in horizontal disposition.

Furthermore, the ScrumMaster demand to the member of Munich to perform a cookie burger testing to make sure, the possible change in the engineering requirements (see Figure 52).

Figure 52. Cookie burger testing.

The result of the testing indicate the following features:

- The irregular parallelism between layers caused the cream drop over 20 minutes.
- The good parallelism between layers did not cause cream drop.

Then, based in the testing results, the development decided to re-design completely the system in vertical disposition.

5.5.3. Sprint retrospective

The observation of this session was on 8th in July (see Figure 53). In this session the ScrumMaster presented the same table configuration of the last sprint retrospective, composed by the three columns (Keep doing, Stop doing, Start doing).

Figure 53. Third Sprint Retrospective.

The results of this session indicated that the Scrum Master attitude improved from last retrospective, keeping the behavior between the balance and tolerance. One team member commented that he improved his English during the last month. Other team member confessed to be comfortable with the technical diary and he noted improvements in his performance. This results were evaluated in Keep doing column.

Referent to Stop doing column, the results indicated that one team member worked in his own without communicate to team. Other team member commented that he was not really involved with the prototyping. Another team member confessed that he was frustrated with the Munich communication, because this fact cause the re-design of the process. The ScrumMaster confessed to spend too many time with organization, give too many pressure to the team.

Concerning Start doing column, one team member commented to do more external activities together to unify the team, other team member promised to help in the prototyping activity. The ScrumMaster decided to focus more with the development rather than organization, give less pressure to the team and improve the communication between Munich and Vic, in order to reduce the miss-understandings of the engineering requirements.

5.5.4. Observations and discussions

During the execution of the third iteration, some interesting events occurred, such as one team member left the team and one team leader from Munich visited the team development in Vic.

In a discussion with the team member who left the team, she affirmed that the main reason to leave the team was an external full-time job offer. During the last informal conversation with her, she commented the following negative points:

- I didn’t feel in the same wavelength than the other engineering students.
- The project did not motivate me enough.
- The tasks to do were not appropriate.
- The ScrumMaster is really strict.
- I don’t like work under pressure.

She also commented the following positive points:

- It was a good experience with Scrum and the team members.
- I like the retrospective sessions of Scrum.
- I would like to see the final machine.
- It was impressive the development of the prototype.
- I learn a lot of engineering concepts.

After this informal conversation, the ScrumMaster though that is very complicated to perform a retrospective session with success, in the end she was not enough brave, to comment all these problems in the retrospective sessions and solve the problems. Therefore, the ScrumMaster noted one of the biggest issues in projects, which is communication.

In addition, the ScrumMaster noted that there are team members, who adapts very well and fast, and others team members, who have a slow process or cannot adapt.

Regarding the visit of one team leader from Munich, after one week of development with him and the team development, he noted some weaknesses in the team. He commented the following points:

- Lack of autonomy referent to solve a problem in group and then share the work equally. He heard several times the sentence “What Can I do now?” from two team members.
- Too many responsibilities for the ScrumMaster.
- Lack of engineering processes. He noted that the basis of the mechanical field was not enough consistent.
- Lack of motivation to learn. The ScrumMaster commented to him that he had to insist a lot with basic work habits such as write every day the daily scrum, check the e-mails, be punctual, save every day the files on the repository and updated Redmine.

He also commented the following positive points:

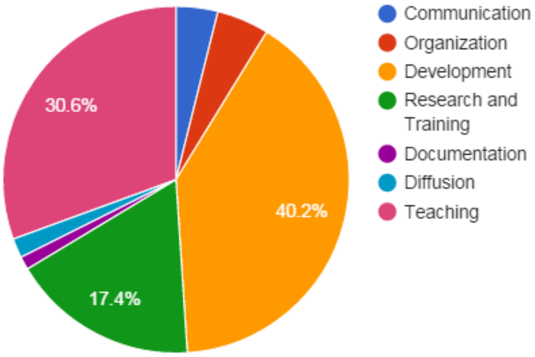
- Tailored system with the technical diary. He noted a good control of the time spent.
- Some members of the development team with experience in job-environment.
- The head of the laboratories of UVIC-UCC is very useful help for team.
- Big range of resources, such as electronic stuff, adequate rooms and appropriate works tools.

After this informal conversation, the ScrumMaster noted that there is big difference, between the university and the company concerning experience in projects, as well as the education level in Germany is higher.

Other point related to this topic is the different backgrounds between the German and the Spanish culture, for instance in Munich is relatively normal that a group of students can manage a part of a company and this is relatively complicated could happen in Vic.

5.5.5. Documentary analysis

During the third iteration, the technical diary was used during nine weeks of development. Therefore, the data collection allowed to analyze with reliability the time spent in each task and progress.



Task	Hours
Communication	25,41
Organization	31,78
Development	261,67
Research and Training	113,52
Documentation	8
Diffusion	11,83
Teaching	199,08
Total iteration hours	651,29

Figure 54. Technical diary.

According to the results (see Figure 54), in that period the development team increased the time spent in development until 261.67 hours, representing the 40.2%. In the second place with 199.98 hours teaching activities (workshops), representing 30.60% of the time spent. In the third place with 113. 52 hours in research and training, representing 17.4% of the time spent. These were the most significative data related task and time.

The conclusions of this analysis determined that the average of time spent per week was around 70 hours and the ScrumMaster valuated this as a successful result, remembering the last iteration that it was around 33 hours. This new value represented to work 20 hours per week for four team members.

Other result, concluded that the time spent in organization decreased from 21% to 4.9% comparing with the previous iteration as a positive result. And the importance to keep more or less the same percentage of research and training, in which is include testing, research information, training with new software tools and activities related to learn and train.

Another conclusion, determined the necessity to focus less in teaching activities, because the fact to prepare and perform the workshops, represented around 200 hours. In that session the development team and the ScrumMaster realized that the preparation of this events represents too many time and the most important in that case was the development, considered the scope of the project.

5.6. Fourth iteration

In this section will be presented the fourth phase of the project development. The contents consist of the procedure, the iteration execution and the documentary analysis.

5.6.1. Procedure

This iteration started on the 13th in July and finished on the 31th in July, with a length of 3 weeks development. It is important to take into consideration that during this iteration another team member left the team, this fact caused frustration for rest of the team. Otherwise the fact to re-design the system was a big challenge to face, therefore the team members were focus in the mechanical field. It was not considered to realize any sprint planning, as it was not necessary. The significant strategy was concentrate the team in the mechanical design.

5.6.2. Iteration execution

During the iteration execution was presented a new CAD design in vertical disposition rather than horizontal. It is important to note that the vertical disposition was the solution for the plastic advance through the different steps machine (see Figure 55).

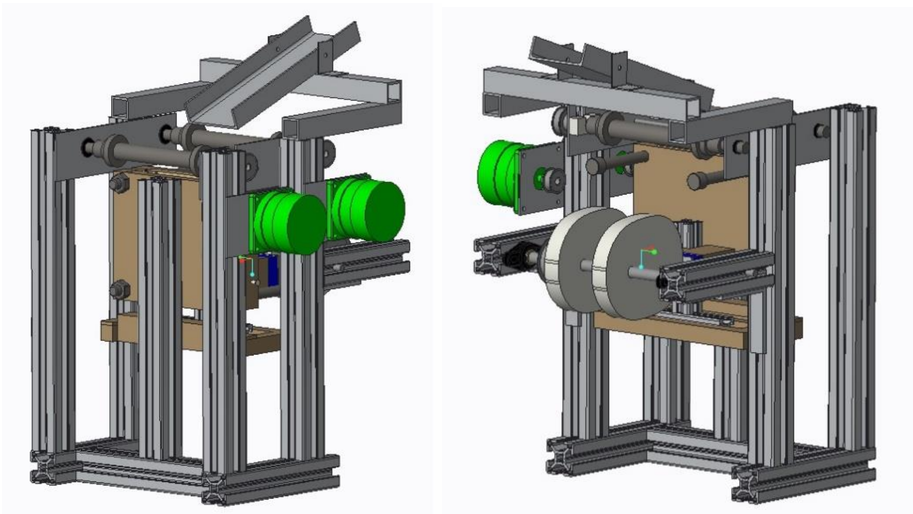


Figure 55. Final CAD design.

Referent the sealing system, the mechanical designer suggested to use a resistant wood called "Sapelly". The ScrumMaster with the cooperation of the head of the laboratories suggested to integrate a hot resistance to the wood with thermic glue and afterward, put Teflon on the hot resistance surfaces (see Figure 56, on the right). Therefore, after adjust all the parameters the first cookies were successfully packaged.

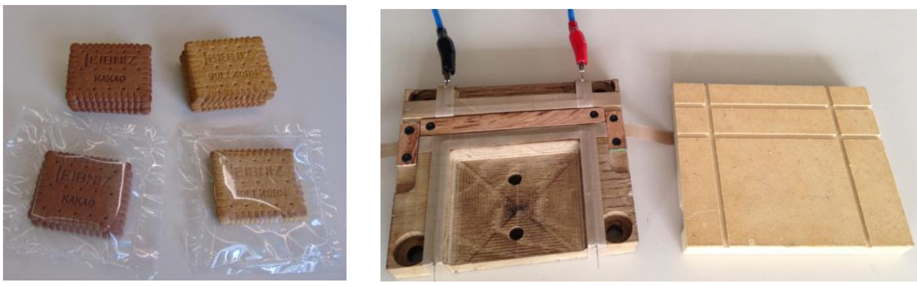


Figure 56. Sealing testing.

Additionally, it was performed a burning test, in order to know the resistance of the wood, Teflon and the thermic glue. The results of this test concluded that was not possible to burn the wood, however the thermic glue and the Teflon were completely non-usable after the test. The maximum temperature obtained was 353 degrees Celsius with more than 20 minutes and 6 amperes of current (see Figure 57).

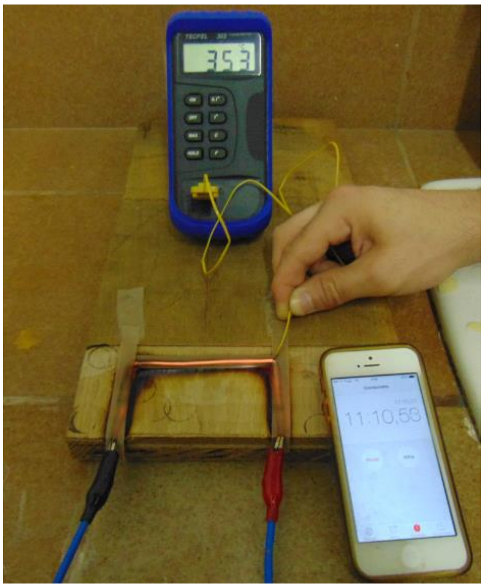


Figure 57. Burning test resistance.

After that, it was built the peripheral mechanical structure with the aluminum rollers, the support layers and the ramp. After, the sealing system was attached to the peripheral structure, which consisted of two platforms (see Figure 58). Once the system was built without actuators, it was perform a manual test, to make sure that the prototype was fully functional with successful results.

Figure 58. Peripheral structure.

Therefore, in order to open and close the sealing platforms, it was assembled a cam system pushed by a motor, which provided a linear movement based in rotatory movement. It is important to note that one of the important requirements was use only electrical supply. Then, without the chance to use pneumatic or hydraulic supply, the mechanical designer had to develop this cam system. In addition, two stepping motors were integrated (see Figure 59) with the aluminum rollers, in order to provide the film.

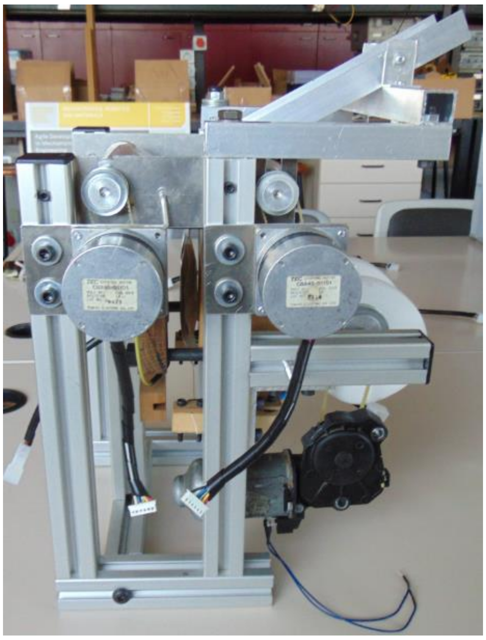


Figure 59. Final prototype.

Moreover, it was implemented an electronic board, in order to control the actuators, the sealing operations and the steps of the machine with interrupters rather than sensors. As it is observed in the Figure xx, it was used the Arduino platform, which in this prototype was the most adequate interface base in the requirements (see Figure 60).

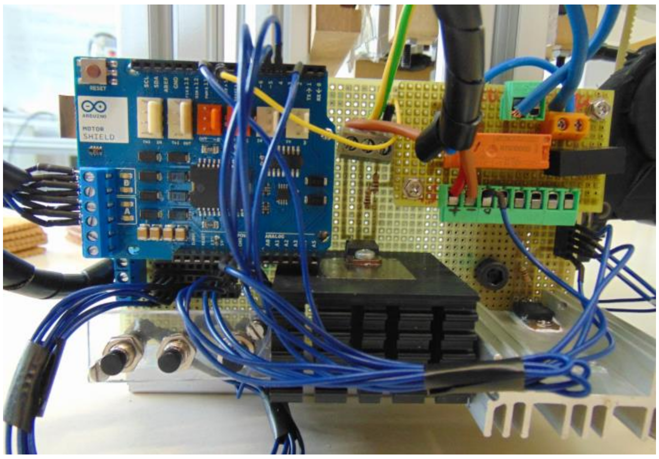
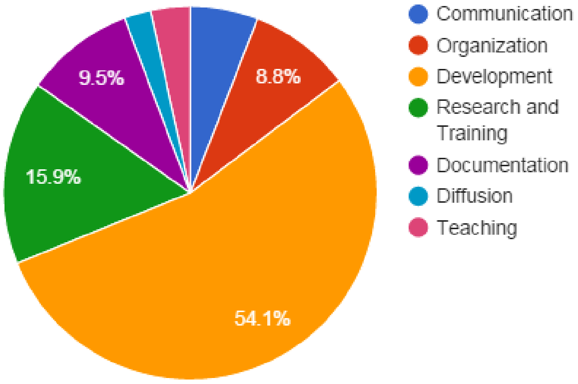


Figure 60. Electronics integration.

Finally, the electronic board was integrated to the peripheral structure and it was performed the validation test. It is important to take into consideration that this machine was a functional prototype, the purpose of a functional prototype is validate a solution that solve all the problems defined in the engineering requirements. In addition, it does not work perfectly, the goal is achieve the purpose of the machine that in this case is package a cookie-burger.

5.6.3. Documentary analysis

In the fourth iteration, the technical diary was used during the last three weeks of development. Therefore, the collection data concerning that period, represents the time spent in the end of the project development, changing significantly the average hours of each task.



Task	Hours
Communication	10,33
Organization	15,58
Development	95,33
Research and Training	28,08
Documentation	16,67
Diffusion	4,09
Teaching	6
Total iteration hours	176,08

Figure 61. Technical diary.

According to the results (see Figure 61), in the first place the time spent in development was 95.33 hours, representing the 54.1%. In the second place with 28.08 hours was research and training, representing 15.9% of the time spent. In the third place with 16.67 hours was documentation, representing 9.5% of the time spent. These were the most significative data comparing tasks with its corresponding time spent.

The conclusions of this analysis determined that the average of time spent per week was around 60 hours and the ScrumMaster valuated this as a good result, remembering the last iteration that it was around 70 hours. This new value represented to work 20 hours per week for three team members.

Other conclusion, determined that the time spent in organization increased from 4.9% to 8.8% comparing with the previous iteration as a negative result. It is important to note that in that period the team consist of three members and the work overload affected directly to the quantity of hours spent in organization.

Another result, conclude that the time spent in teaching decreased from 30.6% to 3.4% comparing the previous iteration considered as a positive result. This changes allowed to the team members focus on the real scope.

6. Analysis and Discussions

In this chapter will be discussed and analyzed the appliance of Scrum through the framework comparison and answering the research questions.

6.1. Comparison to the Scrum framework

In this chapter will be described the analysis and discussions of the Mi5 case study, which is compared with the Scrum framework.

6.1.1. Previous to Scrum

The development team explained the situation before start this project. The main points were no-one was working in product development, instead each member had experience in a particular specialty, without integrate the development, without inspections and adaptions. Another important point noted, that the fact to planning was a new subject, each member knew the importance of the project planning but was not enough remarkable. The ScrumMaster noted that was difficult for the team deliver the tasks on time.

The Scrum Master noted that this method prove some benefits: the frequent meetings involvement connected the team as a unit, the discipline growth with the technical diary, the preparation of the team in the Sprint Reviews and the confidence environment with the team in the Sprint Retrospective.

6.1.2. Scrum roles

Three roles define Scrum as different hierarchy structure, keeping everybody on the same level but with different functionalities. The most relevant point in this study was the missing and necessary Product Owner role, which was compensated, giving more responsibilities to the ScrumMaster and the members of team Munich, who act as consultants. It is important to take into consideration that the members of team Munich were in the highest hierarchy level as they had more experience in general terms. Thus, it can be considered the fact of the team members from Munich were a kind of a Product Owner.

According to Schwaber and Sutherland (2011) describe the Scrum Master as a servant-leader who's responsibility is to ensure that Scrum is understood and enacted, by making sure that the Scrum Team complies with Scrum theory, practices and rules. Servant-leader is a good description of the role, and that is what the Scrum Master has been in for the mechanical team. The Scrum Master is not a superior to the team, rather their friend in a way, but should drive them forward and guides them in using the system, by teaching and coaching. In the case study the team was happy with the ScrumMaster development and responsibility, even though he carried out too many responsibilities.

Referent the development team is described by Scrum as cross-functional, self-management, multi-shaped skilled, with excellent communication skills, focus and committed, in small size and cooperative. In this case study, most of the development team were students without experience, this means that the skills were relatively medium. However, the development completely cross-functional, cooperative and right-sized.

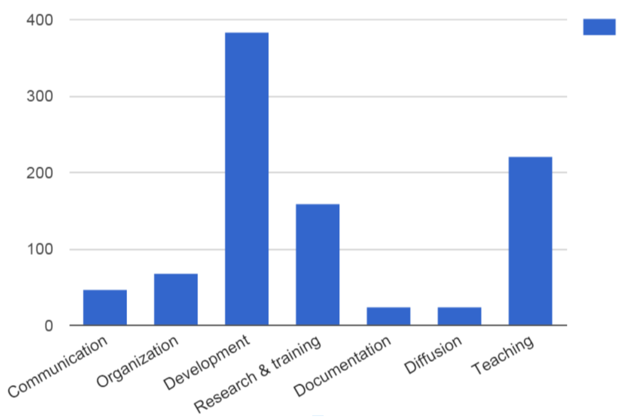
6.1.3. Activities

Scrum define the activities as a set of meetings, the Sprint Planning, Daily Scrum, Sprint Review and the Sprint Retrospective. Each meeting with a specific purpose, following a repeating process of planning, doing, checking and act.

In the Sprint Planning, was not planned more than a week further, as the events and circumstances were not possible to predict, instead every Monday the development team with the ScrumMaster help, were checking the tasks progress and including when necessary new features. It was also considered to remove not appropriate task, as well as the possibility to share feature between team members when necessary.

In the Daily Scrum, was replaced by a technical diary, instead of meet every day, the development team and the ScrumMaster were writing their tasks with detailed information about the time spent in the different task typology. In addition, this technical diary was used to analyze how the team spent the time and improve the possible issues related. Thanks to this collection data technique, the development was studied with the following results:

According the data was collected from the 13th in April to the 21st in July, as a result of 16 development weeks, the following parameters is shown (see Figure 65):



Task	Hours
Communication	47,57
Organization	68.61
Development	383.24
Research & training	159.43
Documentation	24.67
Diffusion	23.91
Teaching	221.29
Total time spent	928,72

Figure 62. Technical diary results.

According the Figure 62, the task which was dedicated more time and effort was development with 383.24 hours. In the second place was dedicated in teaching activities with 221.29 hours followed by research and training with 159.43 hours as the most relevant. At the bottom, organization with 68.61 hours, communication with 47.57 hours, documentation with 24,67 hours and diffusion 23,91 hours. Thus, the total time spent during that period was around 1000 hours. In addition, it could be calculated an estimation of the eight missing weeks with the following criteria:

In the beginning of the project the team were introduced to Scrum, the team had to do some tasks but none of the very intensive in the sense of spent too many hours. Therefore, looking at the technical diary, in the beginning of the second iteration and the weeks with lower time spent the values correspond around 30 per week. Thus, an estimation of this missing weeks would be, **8 weeks x 30h/week = 240 hours**.
By sum with the total spent time = **928.72 + 240 = 1168.72 hours.**

To conclude, the results indicate that the necessary spent time was around 1168 hours in broad terms, where the most significative task was the development, which in this case it is observed as negative results, because that indicates the problems had in the design, prototyping, re-design and prototyping again. A better approach to the success would be spending more time with training and research, with comprehensive and validated documentation and communication.

In the Sprint Review, the amount of work done was showed several times to the people from outside of the project such as teachers, salesman and sponsors. In this occasions was achieved the purpose of the meeting, which is mainly receive feedback, but not less important give a good impression, serious development and a professional staff. In this meetings was inspected and adapted the product development, therefore the technical aspect was presented and analyzed. The ScrumMaster noted that most of the members were well-qualified to present the work done, but not all the members well-qualified to carry-out a task which required a relatively high point of responsibility in the technical aspect.

In the Sprint retrospective, was performed in most of the iterations, as key factor in the communication aspect. Thanks to this meeting was possible to isolate the problems from the personal development of the product development, it seems not very important, but in fact it is. The ScrumMaster as an impediment remover should know from where is the problem, in order to find the best innovate technique to solve it. The ScrumMaster noted that two team members had problems related the personal development in which the Scrum Values define as commitment, respect, openness, focus and courage. In both cases, the members demonstrated lack of commitment and focus.

6.2. Research questions

In this chapter will be answered the research questions posed in the chapter 2.4.

6.2.1. Can a mechatronic product development team use Scrum?

With reference on the case study the answer is: **yes, it is possible.** Because the framework, which was created for product development rather than a specific discipline such as software engineering. Thus, it is completely suitable for a mechatronic, mechanical, electronic and software. However, it might need some adaptions in the activities depending on the development environments.

Concerning to determine if Scrum is better than other methodologies, can be complicated to affirm, however it is demonstrated through the case study that constants deliveries and inspections is better for the mechatronic team at UVIC-UCC, which had no defined work system previous to Scrum.

It is important to take into consideration that the definition of the ScrumMaster role requires a high level in the Scrum framework and a medium level of mechatronic knowledge. In the case study was very helpful that the ScrumMaster had a previous experience in this methodology approach and the other team members had experience with their specialty.

6.2.2. Does Scrum framework need adaption for mechatronic teams?

Based on the four development iterations the answer is: **yes, it is needed some adaptions.** It should be considered that the difference between mechatronic and software development is that mechatronic development deals with physical parts, this fact makes restriction to the changes. Otherwise, the software development give less restrictions.

It is important to consider that to develop a mechatronic product, first it should be defined how is the general set-up, what actuators are needed, make it real through the rapid prototyping and then start programming. It does not make sense to start the programming without following the order of these steps. Nevertheless, what is useful is to think about what type of actuators make an easiest implementation of the software development or with a more complex software implementation can reduce the cost of the hardware.

The necessary adaptions in this case study were the following:

- The length of the iterations can vary depending on the circumstances, such as the phase of the project, the team response and acceptance and the progress.
- It is needed a previous training, between two and for weeks, to introduce the development team to Scrum.
- The daily stand-up is modified to write a technical diary, because not everybody was able to meet every day.
- There is not sprint planning to predict all the tasks in one day for a longer time than a week.
- The development team is not completely cross-functional in the sense that all members have the adequate skills to develop the single new product.
- For a mechatronic team the most important value is the communication, especially between the different disciplines such as mechanics, programming and organization.
- The Product Owner role is needed in the technical aspect.
- It is not necessary perform in each iteration all the meetings, just when necessary.

6.2.3. What are the important factors to succeed with Scrum a mechatronic team?

There are few key factors that could improve the chances to succeed in Scrum development, however it might be difficult to ensure the success completely. First, the fact to have team members who already know each other, otherwise the process may take a bit longer. Second the face-to-face communication, which make easier the connection to the team and enjoy the experience to improve through give and receive feedback. These factors could be suitable for the teams in broad terms.

Concerning the Mechatronic terms, the following factors indicate to affect in a positive way to succeed:

- Experienced Product Owner that provides technical knowledge and makes a clear the direction of the project.
- Previous training with Scrum and experienced ScrumMaster that supports the team keeping them on the correct way.
- Highly-band communication between team members that makes sure everybody is on the same wavelength.
- Commitment rather than involvement, if all team members are committed the quantity of work is equal for each member, there is no possible frustration that can affects the development process.
- A ScrumMaster as an active team member rather than just a coach.
- Previous experiences with other methodologies that require systematic development and work discipline.
- Maturity level in the sense that no one is essential, but each team member opinion is important.

6.2.4. What effects does it have that an international cooperation between teams, in communication and organization aspects are both using Scrum?

Based in the case study development, the international cooperation had positive effects in the organization level and some negative effects in the communication aspect. It is important to take into consideration that the negative effects were solved through the feedback interchange of both teams.

Regarding the communication aspect the following points had negative effects:

- Lack of face-to-face communication between both teams, caused miss-understandings and confusions.
- The fact of work with co-workers that the team does not know, caused lack of confidence.
- The communication in English caused some speaking issues to some members.

Concerning the organization level the following points had positive effects:

- The coordination, work methodology and cooperation has been increased.
- The work methodology promoted discipline habits.
- The work system has been improved by follow systematic processes.
- The fact to save all the data in one very organized repository, increased the easy access to the project information and the performance in documentation tasks.

7. Conclusions

The objective of this project study was to prove the viability to integrate the Scrum methodology in a mechatronic product development team. The case study was conducted at UVIC-UCC in Vic, where a cross-functional team, developing a new packaging module, became agile. The length of the project was five months and the team decided to continue using the methodology even though the team had some frustrations with the strictness of the requirements through the four iterations. The result of the experiment can be observed as a successful, as the team developed a functional prototype and followed the methodology activities despite not fulfilling some scrum values and rules.

The conclusion of this study is that Scrum can be integrated in mechatronic teams. Even though that it is difficult to ensure if the best way to manage a mechatronic projects, it has been demonstrated that Scrum is helpful and promoted to show the constant progress and improve the process frequently, especially in the Sprints Review and Retrospective, which the team achieved confidence and close connection.

This study shows that using Daily Scrum as technical diary rather than a daily meeting improved the original approach of it. The additional improvement in this research demonstrated the total time spent was around 1168 hours and the most dedicated tasks in the cope of this study were development, research & training and organization.

The study demonstrate that Scrum methodology provides several benefits such as an increment of team cooperation and coordination, the promotion of discipline habits, an increment of systemic procedures and an improvement to organize and manage the documentation of the project and the software tools. Nonetheless, it demonstrates in this special case, it is needed an improvement in the international communication aspect. It is also needed a Product Owner to make a clear project direction and a ScrumMaster well-trained to promote and coach the Scrum training.

In addition, the case study demonstrated that the integration of Scrum methodology in an education project, provides a suitable training program for engineering students and a promotion of project-based experiences.

8. Bibliography

KENNETH, Rubin (2012). *Essential Scrum: A practical guide to the most popular agile process.* US: Stanford.

VISCARDI, Stacia (2013). *The professional Scrum Master's Book.* UK: Birmingham

DERBY, Esther; LARSEN, Diana (2013). *Agile retrospectives making good teams great*. US: Texas, North Carolina.

MERRIAM, S.B. (2002) Introduction to qualitative research. *Qualitative research in practice: Examples for discussion and analysis*.

Hennink, M.M. (2011) *Qualitative research methods*. London; Thousand Oaks, Calif: SAGE.

9. Webgraphy

RISING, L., Janoff. *The Scrum Software Development Process for Small Teams*. 2000

http://jeffsutherland.com/ScrumPapers.pdf

[Accessed April 12, 2015].

SCHWABER, k. Advanced Development Methods. Scrum Development Process. 2010.

http://jeffsutherland.com/ScrumPapers.pdf

[Accessed August 20, 2015].

SUTHERLAND, Jeff. In *Scrum Tuning: Lessons learned from Scrum implementation at Google*. 2010.

https://www.youtube.com/watch?v=9y10Jvruc_Q

[Accessed August 15, 2015].

AGILE ALIANCE. *The Agile Manifesto*. 2013

http://www.agilealliance.org/the-alliance/the-agile-manifesto/

[Accessed September 10, 2015].