

# The benefit of implementing the Agile methodology in the development of software projects

Cipatly David Rodriguez Isidro, Clemente Hernandez Arias, Víctor Manuel Arias Peregrino,  
Dulce María León de la O, Hugo del Ángel Delgado

Tecnológico Nacional de México Campus Villahermosa.

## Abstract

Throughout the history of project development there have been different ways to manage and develop a project or to carry out a process, this is increasingly simple and effective for both the client and the work team, today there are different types of methodologies, but what are these methodologies, what are they for, and how can they benefit our projects?

Currently, we have the so-called agile methodologies, which allow us to develop software projects quickly and efficiently through a series of specific steps that allow the team and the client to interact in a beneficial way for both parties, allowing better management of the resources they have to work, it is extremely important to understand the differences, advantages, disadvantages, stages, and actors involved in these, to choose the most appropriate for the work we want to develop and implement the agile methodology properly.

Therefore, we seek to share the experience in the implementation of the agile methodology Scrum in the project "Sistema Web Para El Seguimiento De Los Egresados Del Tecnológico Nacional de México Campus Zona Olmeca" and thus give to understand a little more about these agile methodologies.

## Resumen

A lo largo de la historia del desarrollo de proyectos han existido diferentes maneras de administrar y desarrollar un proyecto o de llevar a cabo un proceso, esto llegando a ser cada vez más simple y efectivo tanto para el cliente como para el equipo de trabajo, al día de hoy existen diferentes tipos de metodologías, pero ¿que son estas metodologías?, ¿para qué nos sirven? y ¿cómo pueden beneficiar a nuestros proyectos?

Actualmente contamos con las denominadas metodologías ágiles, las cuales nos permiten desarrollar proyectos de software de manera rápida y eficiente mediante una serie de pasos específicos que permiten al equipo de trabajo y al cliente interactuar de una manera beneficiosa para ambas partes, permitiendo una mejor gestión de los recursos que se tienen para trabajar, es de suma importancia entender las diferencias, ventajas, desventajas, etapas y actores que intervienen en estas, para poder elegir la más adecuada para el trabajo que deseamos desarrollar e implementar la metodología ágil de manera adecuada.

Por lo que se busca compartir la experiencia en la implementación de la metodología ágil Scrum en el proyecto "Sistema Web Para El Seguimiento De Los Egresados Del Tecnológico Nacional de México Campus Zona Olmeca" y así dar a entender un poco más acerca de estas metodologías ágiles.

**Keywords:** Agile methodologies, follow-up of graduates, Scrum.

**Palabras Clave:** Metodología Ágil, Seguimiento de egresados, Scrum.

## 1. INTRODUCTION

The methodologies allow to carry out projects with organized and systematic processes that obtain effective results, agile methodologies renew these processes improving the process and the experience of developing projects, implementing an agile methodology is a good choice for the development of software projects since it

gives structure to the work plan, these methodologies are based on a manifesto established in 2001 where the criteria to follow of the new methodologies are mentioned, so it is very important to obtain the knowledge to implement it in our developments.

## 2. BACKGROUND

### Methodology

The methodology is a series of scientific rigor techniques applied systematically during a research process, which allows reaching theoretically valid results. We could say that the conceptual support governs the way of applying the procedures in research.

### Agile methodology

Agile methodology is considered a set of techniques and practices that allow the realization and completion of projects in a dynamic way. Created in 2001 by IT (Information Technology) programmers derived from the "Manifesto for Agile Software Development", the main idea is to perform the relevant activities of a process in a more dynamic way and with greater results.

The agile methodology in project management consists of the application of techniques that optimize the delivery of quality results to the client. These practices allow the project to be more flexible and adapt to changes that may arise during its implementation.

## 3. AGILE MANIFESTO

On February 17, 2001, the first Agility met at the Snowboard ski resort in the mountains of Utah (USA) and coined the term 'Agile Methods' to define the working models that were emerging as an alternative to previous methodologies. The Agile Alliance was then created, a non-profit organization dedicated to promoting agile software development and supporting organizations that wanted to adopt this new way of working. To achieve this, they first took the Agile Manifesto, a document that summarizes the agile philosophy in four values and 12 principles.

The values defined in the Agile Manifesto do not focus on practices, methodologies, or work procedures, but rather advocate a change of mindset, a new organizational culture based on four pillars:

1. Individuals and interactions over processes and tools.
2. 'Software' running on comprehensive documentation.
3. Collaboration with the customer on contract negotiation.
4. Response to change over following a plan.

### Agile Manifesto Principles

These four values are embodied in 12 principles, which define the framework of any agile team:

1. Our highest priority is customer satisfaction through early and continuous delivery of valuable software.
2. We accept that requirements change, even at late stages of development. Agile processes leverage change to provide a competitive advantage to the customer.
3. We deliver functional software frequently, between two weeks and two months, preferably in the shortest possible time frame.
4. Business managers and developers work together daily throughout the project.
5. Projects are developed around motivated individuals. Give them the environment and support they need, and entrust them with the execution of the work.
6. The most efficient and effective method of communicating information to the development team, and between team members, is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The promoters, developers, and users must be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design improves agility.
10. Simplicity, or the art of maximizing the amount of work not done, is essential.
11. The best architectures, requirements, and designs emerge from self-organized teams.
12. At regular intervals the team reflects on how to be more effective and then adjusts and refines its behavior accordingly. [1]

#### 4. AGILE METHODOLOGIES

##### Scrum

This methodology based on empirical process control is a framework designed to achieve the effective collaboration of teams in projects, using a set of rules, artifacts, and defining roles that generate the structure for its proper functioning.

Using an incremental approach that is based on the theory of empirical process control. This is based on transparency, inspection, and adaptation; transparency guarantees the visualization of possible problems that affect the result; inspection helps with the detection of undesirable variations in the process, and adaptation allows adjustments to be made to minimize their impact.

Scrum teams are self-managed, cross-functional, and work in iterations. Self-management allows choosing the most optimal way to do the work, so as not to follow guidelines from people outside the team. Team members have the necessary knowledge to perform the work because each team member is cross-functional. The delivery of results is done through interactions; each interaction allows the creation of new functionalities or modifications required by the product owner.

Scrum describes four main events that make up each of the deliverables, Sprint Planning Meeting, Daily Scrum, Sprint Review, and Spring Restrospective; the methodology focuses on dividing the work into different blocks or sections into short periods of time in a maximum of 4 weeks, this is called a sprint.

## Equipo de desarrollo Scrum



**Figure 1.** Scrum Development Team Roles

**Scrum Master:** is in charge of ensuring that the team works based on the theories, practices, and rules of the Scrum methodology, as well as helping the team and customers to understand the interactions that are really useful for the project.

**Product Owner:** The person in charge of transmitting the vision of the desired product, providing the business perspective to the development team.

**Stakeholders:** are people to be taken into account who do not belong to the development process but who have an interest in the project, such as directors, managers or business people, etc.

**Users:** are the people who receive the final product, i.e. the target audience of the project, and may be in charge of the review phase if deemed necessary.

**Team Members:** are the members responsible for the development and delivery of the product, maintaining a horizontal organization in which each team member is self-managed and organized in defining and executing the various sprints.

### Extreme programming (Programación Extrema, XP)

This methodology is based on rules and principles used throughout the history of systems development, applying each one to create an agile process, emphasizing the tasks that add value and removing processes that delay the project.

## PROGRAMACIÓN EXTREMA (XP)

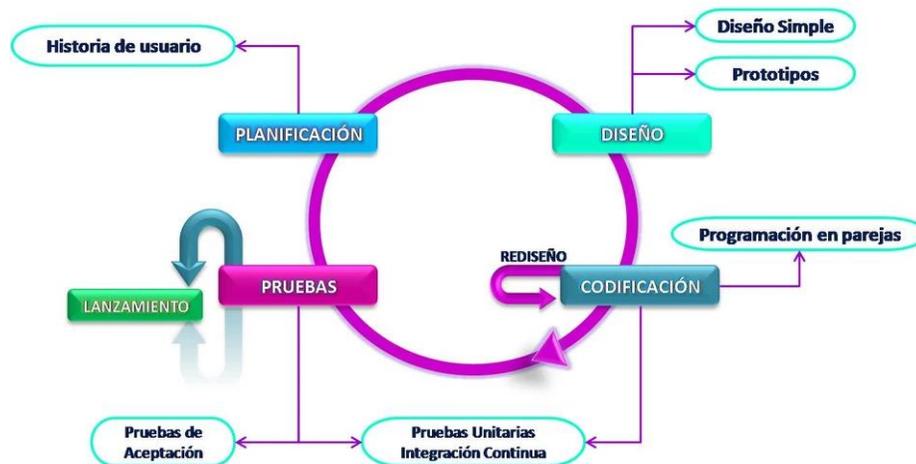


Figure 2. XP methodology framework [2]

The XP methodology is based on the following points:

- It is implemented for projects with imprecise and highly changing requirements, where there may be a high technical risk.
- It has as a principle to continuously feedback to the client and the development team, based on fluid communication between participants, simplify solutions and cope with changes.
- It structures and works values and principles that are taken into account and practiced in the development of the project.
- It is considered a discipline supported by values and principles of the methodologies.

XP has four values that are pillars of software development:

**Communication:** it is the environment of collaboration and communication with the work team and the client to the extent that the client could be considered as part of the team.

**Simplicity:** it is applied in all aspects from design, functionalities, code, refactoring, and methodology, it seeks to develop what the client demands most simply.

**Feedback:** it is found from the beginning to the end of the project, it is done in two ways both the work team and the client to provide information for the evolution of the project.

**Courage:** the development team is prepared to face the continuous changes that arise in the course of the planned activities, each member dares to expose the problems or doubts in the realization of the project, and each day provides maximum performance.

XP starts with meetings with the customer to clarify the needs through user stories, in which they are presented:

- Ongoing meetings and effective communication.
- Requirements may change.
- Prioritization of requirements.
- Small, integrated group (12 people max).
- Highly trained team and ability to learn.
- Methodology based on trial and error.
- Grounding of values and practices.
- Team has general ideas of the application to be implemented.

At the end of this process, a delivery plan must be established with the customer where the initial number of iterations and their duration is defined, where the tasks to be performed are defined, based on the delivery plan.

**Crystal Clear**

It is a methodology that is established according to a color code that defines the complexity of the project, the more critical the system is it is necessary to implement more rigor, also crystal defines a color for each project according to the complexity, there is not a methodology in general, but there is a crystal methodology for each type of project.

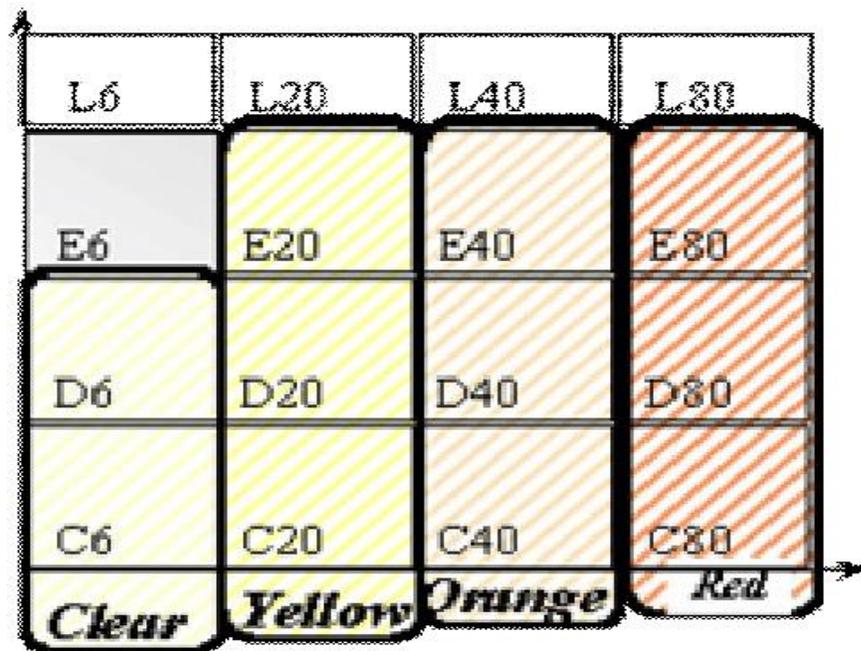


Figure 3. Criticality of the crystal methodology [3]

Each letter represents potential risks:

- C: Loss of comfort due to system failure.
- D: Loss of discretionary money.

E: Loss of essential money.  
 L: Loss of pathways due to system failure [4].

The numbers indicate the number of people that are coordinated in the project according to the following:

Clear is for teams of 8 people or less.  
 Yellow for teams of 10-20 people.  
 Orange for teams of 20-50 people.  
 Red for 50-100 teams and so on through brown and violet [4].

This methodology is governed by principles that make its use efficient which can be:

Each project has a different degree of compensation (Trade Off).  
 A small project is better coordinated.  
 Each project requires different means of communication.  
 Provide feedback to the customer and maintain good communication to reduce problems in failed deliveries.  
 Dedicated and concerned staff must be available to make deliveries on time and according to the customer's request.

**Roles and Products in Crystal Methodology**

In this methodology there are 9 roles that are assigned to the different members of the development team.

Roll- Last responsible	Products
<b>Sponsor (Sponsor, financier)</b>	Mission statement with trade-off of priorities.
<b>Team</b>	The structure and conventions of the team. The results of the reflection work.
<b>Coordinator, with the help of the team</b>	The project map. The delivery plan. The project status. The list of risks. The iteration plan and status. The visualization of the schedule calendar.
<b>Business expert and expert user</b>	The list of objectives per stakeholder. The case studies. The requirements file. The user role model.
<b>Design leader</b>	Description of the architecture
<b>Designers - Programmers</b>	Screen drafts. Common domain model. Schematics and design notes. Source code. Migration code. Testing. The packaged system.
<b>Tester</b>	On-the-fly error reporting.

Figure 4. Roles and Products in Crystal Methodology [4].

5. COMPARATIVE TABLE

Table 1. Comparative table of agile methodologies

Points/Methodology	Scrum	XP	Crystal
Creator/Author	Kent Beck	Jeff Sutherland, Ken Schwaber	Alistar Cockburn
Year	1999	1995	1990
History	Its roots begin in the Smalltalk community. Formulated by Kent Beck, author of the first book on the subject, Extreme Programming Explained: Embrace Change (1999).	It starts in 1986, from the Harvar Business Review article "The new new Product Development Game" by Hirotaka and Ikujiro Nonaka, from there Schwaber and Sutherland formalized the Scrum process in 1995.	Created by one person in particular Cockburn, who created it based on the analysis of different SW development projects and his own experience.
Important Features	<ul style="list-style-type: none"> <li>• Unit testing</li> <li>• Re-manufacturing</li> <li>• Pair programming</li> <li>• Communication between user and developers</li> <li>• Feedback</li> </ul>	<ul style="list-style-type: none"> <li>• Defines a set of practices and roles</li> <li>• Iterative and incremental approach that emphasizes Project management practices and values over other development disciplines</li> </ul>	<ul style="list-style-type: none"> <li>• Agile methodology with emphasis on cycle modeling</li> <li>• Handles short iterations with frequent feedback from users/customers</li> </ul>
Faces/Stages	<ol style="list-style-type: none"> <li>1. Project planning.</li> <li>2. Project design.</li> <li>3. Coding.</li> <li>4. Testing.</li> </ol>	<ol style="list-style-type: none"> <li>1. Pre game: Approach.</li> <li>2. Pre game: Staging.</li> <li>3. Game or Development.</li> <li>4. Post game: Release.</li> </ol>	<ol style="list-style-type: none"> <li>1. Frequent delivery.</li> <li>2. Osmotic communication.</li> <li>3. Reflective improvement.</li> <li>4. Personal safety.</li> <li>5. Focus.</li> <li>6. Easy access to expert users.</li> <li>7. Technical environment with automated testing.</li> </ol>

6. CONCLUSION

After the research conducted and the above mentioned agile methodologies, we defined according to their characteristics, advantages and disadvantages, which would be appropriate for implementation in software development for the project "Web System for The Follow-up of the Graduates of the Tecnológico Nacional de México Campus Zona Olmeca".

This consists of developing a system for the follow-up of graduates that allows us to have a catalog of professionals from the Instituto Tecnológico de la Zona Olmeca (ITZO) and other companies that have agreements with the institution, this in order to generate a tool to help alumni in the world of work, All of this

converges in a labor exchange provided by the system, making it easier for the institution to get closer to its graduates after completing their education and providing the necessary feedback for future improvements.

For this project we decided to use an agile methodology specifically Scrum, which allows us to work together with the client, to deliver a final product that meets all their needs, in each scheduled iteration the client sees the progress of the project and allows constant feedback and facilitates the changes that may arise without having to modify the entire project.

This is achieved thanks to the objectives set and organized by priority according to the needs of the project, also to the daily reviews and reports that the work team makes internally before each iteration allowing to fine tune the details so that each delivery is substantial to the progress of the project.

It is to perform a repetitive but functional process according to the diagram shown below of the scrum development process:

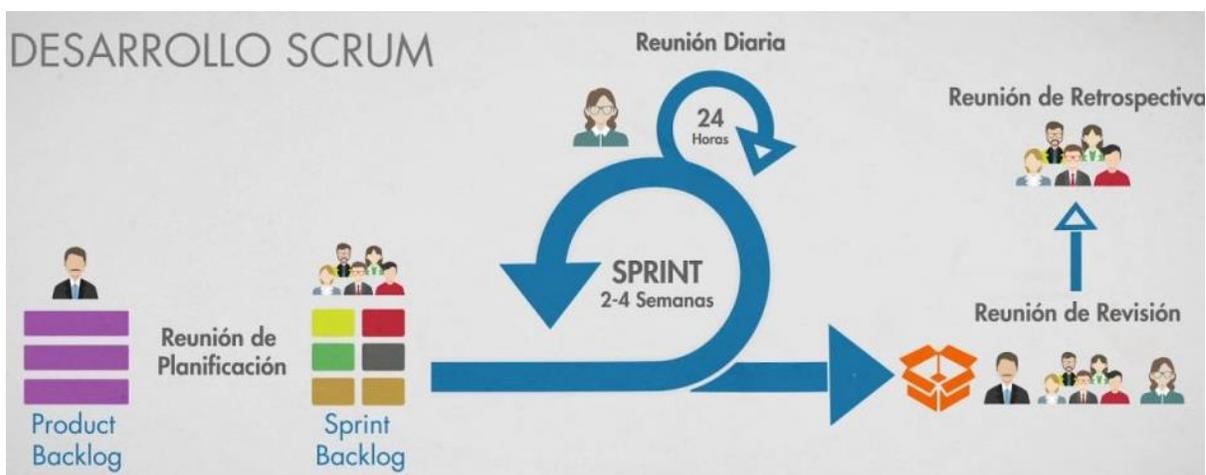


Figure 5. Scrum development diagram [5]

**Sprint:**

At this stage we deliver the progress report to the client, which allows for feedback, this meeting should not be prolonged as it will lose the objective

**Sprint Planning:**

At this stage, we plan the tasks to be addressed and the main objective of the meeting, having the first meeting with the team can be extended if our deliveries are a month.

We rely on some simple questions, the first one is What is going to be done in the sprint, according to this we choose which will be the tasks of the Product Backlog, that is, which are the points that we want to address in the meeting, the second question is How are we going to do it, here we define which tasks need to be completed by the team, according to what was chosen with the previous question.

The moment we define what to do implies that the development team has a goal and is committed to delivering the values that will be shown to the customer at the end of the Sprint.

#### **Daily meeting:**

This meeting involves the work team and the Scrum master so we define the following:

What did I do yesterday, what am I going to do today, do I have any impediment that I need to be solved?

In other words, it is a daily review of what has already been completed, what is next and if there is any impediment that allows me to continue working, it is a constant review of the daily progress of the project.

#### **Sprint Review:**

Finally, this meeting is attended by the customer who evaluates the results delivered by the development team and gives feedback on new tasks or possible changes to be added in the new work cycle.

#### **Sprint Retrospective:**

This event is the last one before the cycle starts again and has a duration of up to 3 hours depending on the time of the Sprint, in this meeting the team performs a self-assessment of how the methodology has been implemented in the last Sprint and allows to propose improvements for the next iteration, as a result, we have a list of improvements to be applied the next day since, at the end of the retrospective, the cycle starts again.

According to the experience obtained in the implementation of the scrum methodology, we find considerable benefits for the development of software projects:

- Integrate the customer during the process to provide feedback on the results obtained in each progress.
- Provide constant feedback to the work team before each delivery.
- Plan the objectives and their priority in the project.
- Establish short delivery periods to speed up the project process.
- Facilitate changes and modifications, if any.
- Keep the team on the same page.

As a conclusion, implementing an agile methodology may not be so easy at the beginning, especially if we come from a traditional methodology or we have never implemented one, however it provides remarkable benefits for the development team and improves the relationship with the customer by involving them constantly with the process, it also allows us to carry out projects in shorter periods of time and thus deliver products that satisfy the customer quickly and efficiently.

## REFERENCES

- [1] Alejandra Duran Gravan. (29 de noviembre de 2008). Manifiesto 'agile', ¿qué es? Obtenido de <https://www.bbva.com/es/agile-manifiesto-que-es/>.
- [2] Yanina Muradas (08 de marzo de 2018). Conoce las 3 metodologías ágiles más usadas Obtenido de [openwebinars.net/blog/conoce-las-3-metodologias-agiles-mas-usadas](https://openwebinars.net/blog/conoce-las-3-metodologias-agiles-mas-usadas).
- [3] Silvia Gabriela Rivadeneira Molina (Jun 2014) Metodología ágiles enfocadas al modelado de requerimientos, Obtenido de [researchgate.net/figure/Figura-4-Criticidad-de-Crystal-Methods-fig2\\_326948456](https://researchgate.net/figure/Figura-4-Criticidad-de-Crystal-Methods-fig2_326948456).
- [4] Rina Elizabeth López Méndez de Jiménez (8 de enero 2015) Metodologías Ágiles de Desarrollo de Software Aplicadas a la Gestión de Proyectos Empresariales, Obtenido de [http://fcaenlinea.unam.mx/anexos/1728/Unidad\\_1/u1\\_act2\\_2.pdf](http://fcaenlinea.unam.mx/anexos/1728/Unidad_1/u1_act2_2.pdf).
- [5] Staff HDC (21 de mayo del 2020) Metodología Scrum, una herramienta útil para agilizar tus proyectos, Obtenido de : <https://hondurasdigitalchallenge.com/2020/05/21/metodologia-scrum-una-herramienta-util-para-agilizar-tus-proyectos/>
- [6] Ulf, M. W., MariaDB Corporation AB, & MariaDB Foundation. (29 de octubre de 2009). Obtenido de MariaDB Foundation - MariaDB.org: <https://mariadb.org/>.

Corresponding author's mail: [rodriguezCD93@outlook.com](mailto:rodriguezCD93@outlook.com)