Improving Efficiency and Effectiveness in the Development of Customized Software Solutions: A Case Study in a Service-Oriented Company

Manuel Fernandes (1) and Américo Azevedo (1, 2)
(1) Faculty of Engineering, University of Porto
Rua Dr. Roberto Frias S/N
4200-465 Porto
(2) INESC TEC
Rua Dr. Roberto Frias S/N
4200-465 Porto
ruifrancofernandes@gmail.com, ala@fe.up.pt

Abstract

This paper presents the approach and results of an improvement project carried on by a company that provides business process outsourcing services based on customised software solutions. The critical objectives considered were to increase the efficiency and effectiveness of the processes involved in developing customised solutions. Thus, the focus of this research was to identify problems in current processes and practices and develop possible solutions to improve. The processes were mapped, the inefficiencies were identified, and suggestions for improvements were presented and analysed. It was concluded that the main problems with the project management and software development processes are related to the lack of visibility of the team load, lack of standardisation and inefficient management processes. These causes result in problems such as IT being unable to plan work and problems associated with quality that negatively influence the lead time of projects. Therefore, suggestions for improvement were formulated and prioritised to address each of these aspects. A more agile approach to software development and redesigning the processes for creating customised solutions were the solutions developed. There was also a need to develop the existing project management software changes to adapt it to these changes.

Keywords
Project Management, Agile Software Development, Process Redesign, Customised Software Solutions

1. Introduction

In response to market needs, to stand out and create a competitive advantage, many companies defined and developed initiatives scoping the digitalisation of their business processes. Indeed, as technology change accelerates and new digital solutions emerge, many companies feel the pressure to perform a digital transformation not to lose their competitive advantage. For Downs and Nunes (2013), digital transformation integrates digital technology into all areas of a business, fundamentally changing how companies operate and deliver value to customers. This is one of the biggest challenges that companies are facing today. No industry or organisation can stay indifferent to digital transformation. Therefore, entire business models are being rethought and reformulated.

For Urbach, Ahlemann and Böhmann, (2017), new service architectures and business models began to be widely adopted, such as Software as a Service (SaaS). This represents a shift from traditional software package license purchases to remotely "renting" specific software services where a business model is paid for usage. For Horlach, B., Drews, P., and Schirmer, I. (2016), these changes should allow the business to be better informed, more flexible, and faster in adapting services and products to market opportunities and customer needs. Realising these needs for new business models and strategies is essential not only for companies facing these changes but simultaneously to understand the challenges of companies that support this digital transformation. Through SaaS models and customised software solutions, these types of companies are responsible for the development and are responsible for the security, performance, availability, and stability of the service.

However, according to Agarwal, N. and Rathod, U. (2005), success is found relatively rare in the world of software projects. This success is assessed to attain the target cost, time, and the desired level of products' quality. Therefore, a software project is successful if it delivers the product with a pre-agreed level of quality within the given time and cost. Experience tells us that, unfortunately, just as a technologically inferior product can win in the market due to superior marketing, a technologically superior software project can fail if project management is lacking. So, in addition to changes in technology and the competitive environment, it is essential to highlight the evolution of mentalities. According to Kerzner (2006), PM came to be a means to achieve the objectives. The same author points out that companies are under pressure to introduce new products on the market nowadays because the product life cycles are getting shorter and shorter.

In particular, in business environments oriented to customisable solutions, as the company considered in our research, this new context creates an urgent need to adjust their internal core processes. To deliver customisable solutions to enable the digital transformation of their customers requires high efficiency and effectiveness, namely timely delivery and compliance with the costs and requirements established. To adapt to this "new" reality and guarantee the success of services, the company has focused on increasing internal efficiency and effectiveness, specifically in the processes related to the development and management of its clients' software projects. Thus, the main objective of this research was to identify problems in current processes and practices and develop possible solutions to improve the overall performance. The following research questions (RQ) were defined: RQ 1 - What are the relevant metrics in analysing the company's software development and project management processes? RQ 2 – What types of problems and inefficiencies affect the company's software development and project management processes? RQ 3 – What are the real causes of problems identified? RQ 4 – How to overcome the problems identified?

A multi-phase approach was defined to support the development of this research project. The first phase aimed to understand the operation of the entire company in terms of structure, processes, surrounding context, and evolution over the past few years by collecting information, holding informal workshops with the main stakeholders, and observing the process. This allowed obtaining qualitative data, specifically textual descriptions of observed facts, valid for the development of a generic characterisation of Management processes. Then, narrowing the focus of the analysis on the PM and software development process (SDP), the informal workshops, the analysis of documents and information existing in the current Software were used to obtain information that allowed the extensive representation and diagnosis of these processes. At the same time, the process of searching and analysing the relevant literature on the topics of the project was initiated.

In the second phase, a deductive approach through the application of concepts and methods already developed and studied by a vast set of authors in order to identify opportunities for improvement and discover their causes. At this stage, it was used workshops to complement and enrich the information we were collecting. In the final phase of the project, a set of solutions were designed and analysed in terms of the benefits and difficulties of their implementation to mitigate or eliminate the problems previously identified. Again, knowledge obtained from the literature consulted was used and complemented by additional workshops conducted. To do this and to structure this project, *kobetsu kaizen* was chosen as a methodology. According to Coimbra, A. (2016), this structured problem-solving methodology can be used to achieve different types of objectives: reducing defects or errors, costs, delivery times, increasing productivity or even the motivation of employees. The following stages were defined: problem Selection, problem description, objective definition, root-cause analysis, and solution.

2. Theoretical Background

It is commonly accepted that PM activities typically take place in parallel with software development activities. For Sheetal Sharma et al. (2012), software practices are techniques for "how" to develop software products or manage software projects effectively. In this section, we briefly explore the subjects relevant to this research, such as the most used models for software development and the processes and the mains topics inherent to the management of software projects.

2.1 Project Management for Information Systems

According to Kerzner H. (2002), to understand PM and specifically Software project management first, it is necessary to know how to recognise a project.

So, to Project Management Institute (PMI) (2013), in Project Management Body of Knowledge (PMBOK), a project is a temporary effort undertaken to create an exclusive product, service, or result. "Project management" is a broad field that can be applied to any development scenario. However, building software presents some unique challenges. In the book from 2008, Project Management for Information Systems, the authors Cadle J. and Yeates D., advocate that although PM's general principles are broadly common to all IS projects, there are nonetheless ways in which the different types of IS project do diverge from one another. The group IS projects in software development, package implementation, system improvement, business consulting and analysis designation, systems migration, infrastructure implementation, outsourcing (and supply), disaster recovery, and more minor IS projects. They question the need to apply the principles and management techniques necessary for each one. For these authors, the right approach is the one that ensures success. Spundak, M. (2014) states that to fully answer how the methodology can be designed for a specific project, the challenge is to define which characteristics of the project are essential for this decision. For this author, the case of the custom software development project shows that it is necessary to combine different approaches. Paulk, M., Weber, C. and Chrissis, M. (1991) report that the success of a software project is closely related to the ability to achieve the desired cost, time and desired level of product quality and therefore, a software project is considered successful if it delivers the product with an agreed quality level on time without exceeding the budgeted cost.

For the same authors, a successful software organisation can manage software development and maintenance processes. For these authors Key characteristics, that are directly related to management are: precise process communication, activities are carried out as planned, work should be based on consistent processes, but updated and improved where necessary, roles and responsibilities are clearly defined, and quality of the process and product is monitored. A disciplined process is consistently followed because all participants understand the value of doing so, and the necessary infrastructure exists to support the process.

2.2 Process Models for Software Development

According to Pinto, P. (2008), a software development process is a set of activities, methods and practices involving transformation that people use to develop and maintain Software. Process models for software development have evolved alongside advancements in computer science. At the same time, it may be tempting to adopt the latest innovations in software process models simply, not all process models fit all projects' needs. Some process models are more complex or comprehensive than needed for a specific project. According to Sheetal Sharma et al. (2012), there are several software developments models that have their characteristics, advantages, disadvantages, and applications.

3. Case Study Description

The studied company has evolved over the years from a document custody company with all the inherent logistical skills to position itself in the value chain, ensuring all the activities inherent in document management. Activities ranging from the collection of physical documents to a structured data delivery service to its customers supported by Software developed according to the needs of its customers. So, this company provides business process outsourcing services based on customised software solutions to ensure digital transformation from other companies.

First, it is essential to comprehend how the company provides its customers with a wide range of services based on information technology systems, namely: capacity to develop web portals based on big data, workflows, internal collaboratives, robotics, outsourcing of administrative back office, data extraction from documents, mail management, etc.

So, the company divides these solutions into six fundamental processes that characterise the global services of the IT department: parameterisation of physical files, parameterisation of digital files, developing customised solutions (projects), budgeting, developing robots and problem solving (support and maintenance).

Although there is an idea within the company that the main problems would be in customised solutions, we chose to briefly know each of these processes using a process modelling tool called SIPOC (Table 1).

Table 1 - SIPOC of the company' IT processes

Suppliers	*Inputs*	*Processes*	*Outputs*	*Clients*
Operations	Order Data and Files	**Parameterise Digital Files Portals** Creation of tables; Robot parameterisation; Test; Portal Parameterisation	Customised portal Files available on the portal Integration with the customer's IS.	Digital Archive Client
Operations	Order Data and Files	**Parameterise Physical Files Portals** Portal parameterisation (client image) Test	Customised portal	Physical Archive Client
Operations Commercial	Data Order email Specifications	**Develop customised solutions (Project)** Planning Back End Front End Integrations and Tests Go Live	Portal Modules API Data base Customised Software	Clients Operations
Comercial	Order Specifications	**Budgeting** Requirements Analysis Budgeting Functional analysis	Functional Dossier Budget	Commercial
Operations	Order Data and Files	**Develop Robot** Operative Analysis Development Test Monitoring	RPA Script	Operations
Internal Departments Clients	Order	**Troubleshooting (Support)** Problem analysis Resolution Test	Support	Internal Departments Clients

So, after this high-level analysis of the existing processes in the company's IT department, we try to understand the importance of invoicing and the impact each of these processes had on the perception of quality, cost and delivery of the service provided by the company's IT department. (Table 2).

Table 2 - Perception of processes' impact on a company's IT services

	Quality (3-1)	*Cost (3-1)*	*Delivery (3-1)*
Parameterise Digital Files Portals	1	2	1
Parameterise Physical Files Portals	1	1	1
Develop customised solutions	2	3	3
Budgeting	3	3	3
Develop Robots	2	1	2
Support	2	2	3

Then, we did a prioritisation exercise giving different weightings, considering the impact of these processes on invoicing, referred above, and their alignment with its strategy. The analysis and improvement of the processes associated with developing customised software solutions were then defined as the priority. Informal workshops with key stakeholders and observation of team performing their daily tasks and processes were helpful to develop a generic picture of this IT process to develop customised software solutions. The following phases or processes and their relationships were identified in figure 1.

Figure 1 - Phases for developing customised software solutions at the studied company

In order to support our project and to understand possible differences between the perception and the reality of the numbers, we decided to collect information on the field. A qualitative approach to research, informal workshops, and data analysis allowed the identification of relevant metrics in analysing the company's software development and project management processes, thus managing to answer our first research question (RQ1).

As previously mentioned by Agarwal N. and Rathod U. (2005), the performance of a software project is evaluated in terms of its ability to achieve the desired cost, the time and the desired level of quality of the products and, therefore, a software project is considered successful if deliver the product efficiently and with a pre-agreed quality level within the given deadline and cost.

Framed in our research work, time, quality and cost analyses will be presented for ten projects of customised software solutions developed by the company. However, it was also essential to limit the scope of our analysis; we only collected information about projects that developed between January 2019 and March 2020. Were also defined the characteristics that represent the case under study and the most common of the company projects, such as projects with more than ten days and up to 90 days of budget.

Although the studied literature points to indicators related to the time, quality, and cost of the projects, the two defined and analysed are related to time, such as project delivery time and on-time delivery, and quality, rework. This decision is due to the expectation that the impact of the solutions to reduce in 20% the lead time, reducing rework by about 57%, and the increase of on-time deliveries to 75% will have positive repercussions on the cost of the projects.

So, when we looked at the numbers of these indicators before the project, we perceived an average of 150 working days required since the customer requests the visit of the sales account until the solution is in a production environment and available for full use. It was possible to recognise the weight each phase has on the lead time of the different projects. Implementation showed a weight of 66%, while acceptance and budgeting presented very similar percentages, 16% and 17%, respectively. Other important conclusions were made through this analysis; we realised that 80% of the projects did not finish in the estimated time, and 70% of the projects needed rework.

4. Problems and root cause analysis

After having a comprehensive view of the company and its operation, specifically on the processes selected for the subsequent phases of the project, this chapter is dedicated to identifying opportunities for improvement (problems) and root cause analysis, thus responding to RQ2 and RQ3. So, the question arose that would serve as a guide for the following sessions: w*hy are we in the current state and not in the proposed objectives?*

This work resulted in 29 causes or opportunities listed by the team that were affecting the failure to achieve the expected results in each phase of the process. After the causes have been identified and bearing in mind the objective of this phase, to understand and analyse the root causes, we chose the diagram of Ishikawa, also known cause-effect diagram. This tool consists of dividing the causes into types, and the root causes arrive, successively asking "Why?". The team's exercise in realising a problem was always to debate the problems to understand why this happens until the root cause was almost confused with the solution.

As already mentioned, the three indicators are related, so it was decided to execute the Ishikawa diagram analysing only as a cause for the current situation of the lead time. With this approach, we can identify opportunities for improvement and find out the real causes of the problems. To better understand the problems, we will divide them into the problems that affect each phase. However, we realise that there are problems in the earlier stages of the process that stem from later stages and where the root cause is the same as in that earlier stage.

4.1 Budgeting phase

The main problems are mostly related to the low standardisation of the request and response to the budget. In fact, problems that we associated with the customer or the commercial department are due to the lack of a fragile budget request. When we look at a specific case like the problems associated with the lack of information, these are related to the lack of adequate budget request questions. Other issues are related to do not understand what the client needs or requirement specification. These problems lead to a wrong estimation of hours, or even the final solutions are presented that do not meet expectations which in turn leads to rework to meet the customer's expectations
The absence or poor communication between departments that lead to not understanding all the requirements and poorly defined scope are also related to the lack of standardisation of the budget request. Other root causes identified

as a result of the problems of this phase are the lack of standardisation in response to the budget, lack of processes after customer acceptance, lack of the process of creating documentation and lack of visibility of the team's capacity load. This last root cause addressed in the paragraph above comes up from the problem identified that IT does not give deadlines before acceptance. It happens because the IT manager is unaware of the team's capacity and load.

4.2 Documentation phase

The problems observed at this stage are related to the absence of a robust and more effective process for creating and accepting the functional dossier by the stakeholders. Simultaneously, there is no standard document to standardise the solution. Most of these problems will impact the solution's delivery time. The absence of a dossier built and accepted by all stakeholders would increase the probability of the solution being delivered not meeting the customer's quality expectations.

4.3 Planning phase

The first problem identified is that there is no time set aside for unforeseen events, and unforeseen circumstances will lead to more work. Moreover, the lack of visibility is essential to understand, which means that it is not possible to plan and that other departments think IT is able to start immediately and that it is not exactly true.

4.4 Development phase

In this phase, we are able to more clearly realise that the problems that occur are not just due to processual problems or lack of documentation of tasks in the phase itself, but problems of the entire process in previous or later phases. The main problem of a "bad" development is the result in the end. If, in the end, the solution does not work will force rework. If, on the one hand, the development of solutions that do not satisfy the customer directly impacts the lead time, this problem still gives rise to wrong priorities and the inability to perceive the team's load and consequently bottlenecks. In our opinion, this happens because of the lack of standardisation of task documentation.

Another critical issue related to the long lead time is that the support that should oversee one of the developers does not solve and assigns the task to another colleague who, given the priority of the request, may interrupt the planning. If all developments were documented, this problem would not occur as often. The problem related to the code review gets stuck in one person, which slows down the process (bottleneck) because of the lack of team organisation. Simultaneously, the managers do not have visibility of these bottlenecks.

4.5 Testing and Pilot phase

The root causes of the problems identified in these phases are mostly related to others' phases. The lack of the process of creating the functional dossier with stakeholders and the lack of the process of creating the project with internal departments will bring problems to these phases of the project. These problems will affect the lead time because if the output of this phase may lead to the need to get back to work on the solution.

5. Problems and root cause analysis

After all, the leading causes have been identified, and in order to answer the last of our research questions (RQ4), in this chapter, a set of suggestions for improvements was thought and analysed in terms of the benefits and difficulties of their implementation. For each problem that has its root cause, some suggested solutions are presented to improve the software development and project management daily activities. In addition, given the limited project period and the available internal resources, not all proposed solutions were selected to be designed. In this sense, before the final design of the solutions, we analysed the possible solutions in terms of two factors - the implementation effort and the expected impact - to prioritise those that will be put into action.

5.1 Proposal Solutions

The root causes of the problems identified in these phases are mostly related to others' phases. The lack of the process of creating the functional dossier with stakeholders and the lack of the process of creating the project with internal departments will bring problems to these phases of the project. These problems will affect the lead time because the output of this phase may lead to the need to get back to work on the solution. To solve the problems identified in the previous chapter, the following possible solutions were suggested in Table 3.

Table 3 - Suggested solutions

ID Solution	*Suggested Solution*
S1	Standardize budget request
S2	Create a training plan for the current team of sales accounts;
S3	Standardize the response to the budget
S4	Create handoff processes between the sales team and the IT
S5	Create a support tool for team management
S6	Create a process of documenting developments
S7	Standardize the functional dossier
S8	Create handoff processes between the IT and client
S9	Re-design the functional dossier creation process
S10	Re-design the planning process
S11	Create a process of testing
S12	Create a training plan for developers
S13	Re-design the project creation process

Their primary considerations about this table are the benefit of a solution from the implementation of the others. So, while the problems are related to the specific stages of the process, the suggested solutions may impact the different parts of the process. For example, when we suggest standardising the budget request, it will impact the overall lead time or the budgeting lead time and impact the other phases of the process. When we refer to the relationship between solutions, we think that if we create a support tool for management where the team can see the carrying capacity or the challenges it faces, we can act on the team's training needs.

5.2 Impact-Effort Analysis

After presenting the various improvement solutions that address the problems encountered, the next stage of this project included prioritising such solutions through the execution of an impact-effort analysis. An impact effort matrix is a decision-making tool that assists people to manage their time more efficiently. An organisation, team, or individual assesses activities based on the level of effort required and the potential impact or benefits they will have.

Ideally, solutions are expected to generate increased positive benefits for the company and, simultaneously, not be difficult to implement - do not take long to be implemented, does not have significant costs associated, and does not require the collaboration of many employees, namely from other functional units. However, natural trade-offs may be considered. So, the suggested solutions to improve the company software development and project management regarding the level of impact in lead time and the implementation effort are represented in Figure 2.

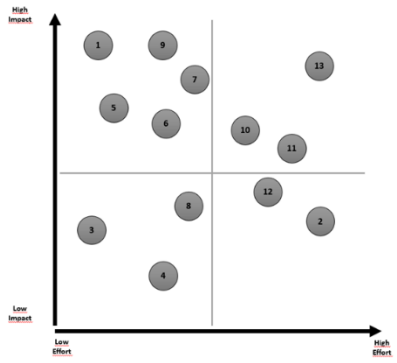

Figure 2 - Effort impact analysis for proposed solutions

Taking into account the assessment made for each proposed solution, the standardisation of the budget request (S1), the creation of a support tool for team management (S5), the creation of a process for documenting developments (S6), the standardisation of the functional dossier (S7) and the reformulation of the process of creating the functional dossier (S9) was selected to be designed, given the expected impact it will have on all stakeholders. Another important consideration is that after looking at the results of our impact effort analysis and each solution's problems, we can conclude that these five solutions cover 17 out of 28 problems found throughout the process.

5.3 Develop Solutions

The last step of this project is to design the solutions. Therefore, considering the several authors who address the different methodologies of software development and the characteristics and related activities for successful project management, we chose to analyse the solutions to be developed, such as team organisation, standardisation and process redesign. For Sharp, A. and McDermott, P. (2009) because successful process redesign projects usually involve significant information systems efforts, and the implementation of an effective information system usually involves redesigned processes. So, a more agile methodology for SD, creating new functionalities in the project management software (PMS), process redesign, and standardisation are the proposed solutions that we will develop in the following points.

a. Scrum Solution

According to Pinto, P. (2008), Scrum projects progress through a series of iterations called "sprints", and at the beginning of each sprint, the team determines the amount of work it can do during that sprint. The job is selected from a prioritised list called "product backlog". The team's work believes it can complete during the sprint is moved to a list called "Sprint Backlog". A brief daily meeting, the "daily scrum", is held to allow the team to inspect their progress and adapt as needed.

According to the same author, Scrum outlines four essential techniques within a sprint: sprint planning, daily scrums, sprint review and project retrospective. To implement this methodology in the company, we must adapt the ideas from the different authors to the company's reality. So, to design the solution, we need to consider the definition of roles and responsibilities, the ceremonies of the development cycle, and the strategies for implementing the model.

- Roles and responsibilities – We designate the current IT manager as the product owner responsible for the project because he is the most experienced person and able to list and prioritise the features better, and the current project manager assumes the functions and denomination of the Scrum Master in new projects. New fields have been added to the company' PMS, as we can see in the project creation view, to adapt to this reality.

- Sprints - According to Zamith, M. (2018), the duration of the sprint will depend on the reality of the company, project or product that is being developed. In this specific case, it was defined that the allocation and evaluation of tasks will be carried out every seven days in a meeting.

- Document *user stories* - A workshop where the criteria of correction, quality, clarity, consistency, viability, and traceability were addressed was then held with the entire sales and development team for the teams to be aligned on how the requirements should be documented.

- Sprint planning - Our proposal went through changes in the current Software with features that responded to this planning need, as shown in figure 3. This new version allows the IT manager to view the tasks in the backlog and their time estimated completion time, assign tasks to different developers according to different priority criteria and evaluate their completion. The main functionality of the developers is to be able to view the tasks assigned to them.

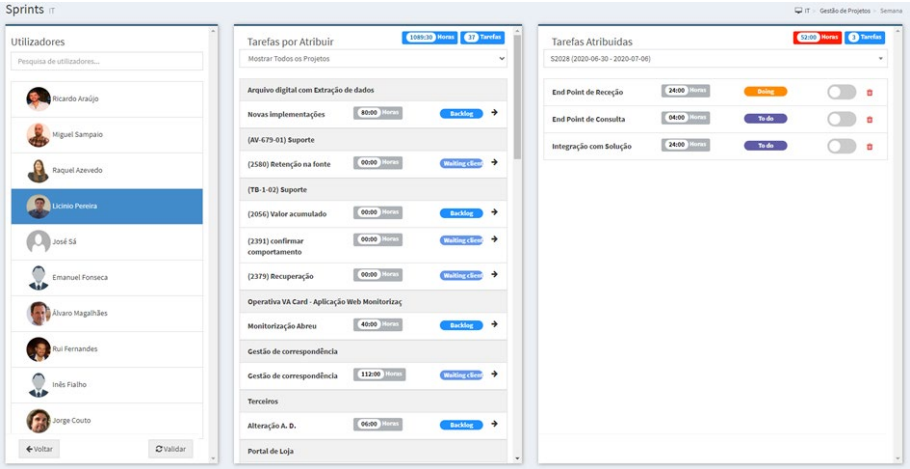

Figure 3 – New features in project management software

- Daily Meeting - To organise these meetings, our proposal created a team table, which includes a work plan for the days of the week, the task pipeline, and the waiting tasks.

- Sprint Monitoring - To this methodology, tasks should be placed in different categories on the task board, such as to-do, in progress, to verify, verify, and do. This functionality already existed in the previous version of the PMS. In the developments of this new version, the option was to make the status of tasks visible in the new view.

- Sprint Review - it was defined that every seven days, end of the sprint, the developments must be demonstrated by the developers, approved by the product owner (IT manager), the resources must be removed from the product backlog, and the parts stakeholders should provide feedback and suggest new features.

- Project Retrospective – It was decided that this moment will become part of the PM and development process. The IT Manager must summon those involved in the different areas. Moreover, reasons like the lack of time, the lack of desire to look at the past, and the fact that people do not want to expose themselves, cannot be an excuse for this more formal meeting not to happen.

a. Process Redesign

Considering this, we redesign the future process of creating a customised software solution with the team. In the following few paragraphs, we will detail the new process and documents to ensure the expected improvements. This process is triggered by the visit of the sales account to the customer's company. The first differences are in the budget phase. The customer will need to create the feature specification together with the sales account using user stories. Thus, when the IT department receives this document, their understanding of what is intended by the customer and the estimation of the development phase will be facilitated.

There was no standard document or place to attach it to the budget request interface. A functionality was then developed that allows the sales account to attach this document. The assumption in this new document is that it is filled in by the commercial and complemented by IT in order to be the basis of the proposal that will be presented to the client. So, there was a need to develop two new features. One that allows the document workflow and the other to attach that document to the client's file.

The process is thus no longer iterative between the client and the company but is only incremental between the two departments. All documentation and planning activities take place after the customer decides to proceed. At this moment, there was also a break from the past. The project manager's planning and documentation process will be carried out with the sales account and a team leader of the operations department. The results of this meeting will necessarily be a draft of the functional dossier containing the scope of the project, accepted features, macro planning,

and an appointment for a kick-off project meeting. In this kick-off meeting, the final document must be presented and signed by all stakeholders, including, of course, the client.

As we already mentioned during the development, the IT manager is responsible for the sprints planning. Every seven days, developments must be demonstrated and approved by the IT Manager, and suggestions for customer improvements must be evaluated. After each meeting, a minute of the meeting should be attached to the documents of the project. Simultaneously, an independent team must carry out an internal audit to check the main deliverables at the end of each sprint. As a result of this audit process, a non-conformity report must be prepared and delivered to the Product Owner to create corrective measures. These measures must be transformed into tasks and included in the product backlog so that non-conformities are corrected.

As the agile development process is iterative and incremental with the different stakeholders, the same sales team responsibility no longer exists. Then, in the last meeting where the developments are demonstrated, the test phase and pilot period dates must be defined considering the period in the budget.

The test phase already takes place in the production environment. What we proposed was the creation of a test environment before going to production. The code is a review, and the uniformity of the nomenclature and the documentation of the code is verified so that there is the slightest difference in the code of the different programmers. This is a change with the past delivery method because the code is now reviewed by the IT manager regarding efficiency and not just about effectiveness. The process finishes after the pilot, with the moment pre-defined to go live with a kick-off meeting.

6. Main Conclusions and future research

A qualitative approach to research, informal workshops, and data analysis allowed the identification of relevant metrics in analysing the company's software development and project management processes, thus managing to answer our first research question. When we identified problems that affect processes, opportunities for improvement, and the root causes of problems, we faced a significant challenge from the beginning. The procedures are mainly ad hoc. However, direct observation of the team's tasks, document analysis and informal workshops allowed the identification of 28 causes or opportunities affecting the failure to achieve the expected results in each phase of the process. After the causes have been identified, we can see that the main problems with the project management and software development processes are the lack of visibility of the team load, lack of standardisation, and deficient management processes. These causes result in problems such as IT being unable to plan the work in advance and problems associated with quality that negatively influence the lead time of projects.

So, we believe that with a more agile approach to SD, specifically using some of the techniques and events of Scrum, we managed to improve the current state of the situation concerning team management. Regarding inefficient processes and lack of standardisation, we opted for redesigning the entire process of creating customised solutions by the company. As successful process redesign projects often involve significant efforts in information systems, we have also implemented several changes to the current project management software to adapt it to new changes. All changes were accepted and prepared to be implemented, however not all were initiated given the time of the project.

Given the short project time and the company's priority to improve the development of customised solutions and the management of projects associated with this solution, it would be interesting to see if this development model would apply to its other solutions. Thus, it would be interesting to analyse the impact of improvements in the budgeting process on the percentage of conversion from a prospect to a client (hit rate).

As a proposal for future studies in this area, we recommend focusing on the impact of a change in the organisational structure. This company has a very vertical model of organisational structure, which, as the literature points out, can hinder an improvement process. Thus, future research can focus on assessing the impact of changing the current model to a more horizontal structure model.

Finally, in addition to the limitations mentioned throughout this chapter, we also point out that the context of our project is the specific context. Thus, future investigations may choose to evaluate the applicability of our solutions in different contexts because the literature is inconclusive and controversial.

References

Agarwal, N. and Rathod, U. (2005), "Defining success for software projects: An exploratory revelation", International Journal of Project Management, Vol. 24 No. 4, pp. 258-370.

Cadle, J. and Yeates, D. (2008), Project Management for Information Systems, 5th ed., Pearson Education Limited, England.

Coimbra, A. (2016), Kaizen: Uma estratégia de melhoria, crescimento e rentabilidade. Edited by McGraw-Hill.

Downs, L. and Nunes, P. F. (2013), "Big Bang Disruption," Harvard Business Review (91:3), pp. 44-56

Horlach, B., P. Drews, and I. Schirmer, (2016). "Bimodal IT: Business-IT Alignment in the Age of Digital Transformation", Proceedings on the Multikonferenz Wirtschaftsinformatik, pp.1417–1428.

Kerzner, Harold R. (2013). Project Management: A Systems Approach to Planning, Scheduling, and Controlling. Edited by John Wiley & Sons. 11th ed.

Paulk, M C, Curtis, B and Chrissis, M. B. (1991) Capability Maturity Model for Software, Software Engineering Institute

Pinto, P (2008). Software Improvement Process at CERN CO-AP. Master Thesis. FEUP-UP.

Sharp, A. and McDermott, P. (2009). Workflow Modeling: Tools for Process Improvement and Applications Development: Artech House.

Sheetal Sharma (2012). Agile Processes and Methodologies: A Conceptual Study. International Journal on Computer Science and Engineering (IJCSE).

Spundak, M. (2014). Mixed agile/traditional project management methodology. Reality or illusion. Social and Behavioral Sciences 119, 939-948.

Urbach, N., Ahlemann, F., Böhmann, T., (2017). The Impact of Digitalization on the IT Department. Bus Inf Syst Eng 61, 123–131

Zamith, M (2019). Desenvolvimento de Software para dispositivos méidocs: uma abordagem ágil. Master Thesis. FEUP-UP.

Biographies

Manuel Fernandes is a Senior Project Manager at TBFiles. He holds a Master's in Service Engineering and Management from the Faculty of Engineering of the University of Porto (FEUP), Portugal.

Américo Azevedo is an Associate Professor in the Department of Industrial Engineering and Management at the Faculty of Engineering of University of Porto (FEUP), Portugal. His research and teaching focus on operations strategy and business processes management. He teaches in the academic programs of FEUP and PBS (Porto Business School) and in specific programs such as EDAM (Engineering Design and Advanced Manufacturing) of the MIT-Portugal Program. He has been the author of many international journals and technical publications and is also active in preparing and participating in R&D projects involving industrial companies. He has been a reviewer and evaluator of several international R&D Industrial projects and a member of several scientific programs committees. Currently is head of Center for Enterprise Systems Engineering at INESC TE (applied research institute).