TÉCNICO LISBOA

Information Analysis and Management for Agile Project Management with the ITLingo-Cloud Platform

André Nobre Revés Barrocas

Thesis to obtain the Master of Science Degree in

Computer Science and Engineering

Supervisors: Prof. Alberto Manuel Rodrigues da Silva
Prof. João Paulo Pedro Mendes de Sousa Saraiva

Examination Committee

Chairperson: Prof. Daniel Jorge Viegas Gonçalves

Supervisor: Prof. Alberto Manuel Rodrigues da Silva

Member of the Committee: Prof. Paula Cristina Negrão Ventura Martins

September 2022

Abstract

Considering the market's competitiveness and the complexity of organizations and projects, analyzing and managing information is crucial to support software development and project management processes. These practices are essential to increase performance, reduce costs and risks of failure, and guarantee the quality of results, keeping the work organized and controlled. ITLingo-Cloud is a multi-organization and multi-workspace collaborative platform to analyze and manage information that can support translating project performance knowledge into improved decision-making. This platform allows users to quickly set up their environment, manage workspaces and technical documentation, and analyze and observe statistics to aid both technical and business decisions. ITLingo-Cloud was designed considering the solutions available in the market, specifically standard features found in project management tools and inspired by emerging cloud IDE tools. ITLingo-Cloud supports multiple technologies and languages, promotes data synchronization with templates and reusable libraries (that have been studied in related research projects), as well as automation tasks, namely automatic data extraction, automatic validation, or document automation. We apply and evaluate ITLingo-Cloud with two experiments and compare it with other related approaches.

Keywords: ITLingo-Cloud, Agile Project Management, Data Visualization, Data Analytics, Information Management

Resumo

Considerando a competitividade do mercado atual e a complexidade que as organizações e projetos acarretam, analisar e gerir informação é fundamental para apoiar desenvolvimento de software e processos de gestão de projetos. Práticas como estas são essenciais para aumentar o desempenho, reduzir custos, riscos de falhas e garantir a qualidade dos resultados mantendo o trabalho organizado e controlado. ITLingo-Cloud é uma plataforma colaborativa multi-organização e multi-workspace para analisar e gerir informação que pode suportar a tradução de informação sobre o desempenho do projeto em melhores tomadas de decisão. Esta plataforma permite que os utilizadores possam configurar rapidamente o seu ambiente, gerir espaços de trabalho, documentação técnica e analisar e observar estatísticas para ajudar em decisões técnicas e de negócio. O ITLingo-Cloud foi concebido tendo em conta as soluções disponíveis no mercado, especificamente para incluir funcionalidades encontradas em ferramentas de gestão de projetos e inspirado em ferramentas de ambiente de desenvolvimento integrado na nuvem emergentes. Esta solução suporta múltiplas tecnologias e linguagens, promovendo a sincronização de dados com templates e bibliotecas reutilizáveis que têm sido estudados em projetos de investigação relacionados, bem como tarefas de automatização, nomeadamente extração automática de dados, validação automática ou automatização de documentos. Aplicamos e avaliamos ITLingo-Cloud com dois estudos de caso e comparamos com outras abordagens relacionadas.

Palavras-chave: ITLingo-Cloud, Gestão Ágil de Projetos, Visualização de Dados, Análise de Dados, Gestão de Informação

Acknowledgments

Firstly, I would like to express my sincere gratitude to professor Alberto Silva for his guidance throughout the process. His help was crucial, and I would not have made it through my master's degree without his support. His constant availability and permanent guidance and his advice were always essential. I want to thank professor João Saraiva for all the tips and guidance in the development of this project. Both contributed with their knowledge to make this project successful.

Many thanks to my family and friends who supported me throughout my studies. Without them, nothing would be possible. Their encouragement and support were fundamental for me to achieve this point.

Finally, I want to thank everyone who participated in the testing sessions, thanks for the availability and feedback.

I am grateful for this opportunity and want to thank all the people I had the chance to work with and learn from.

Table of Contents

List of Figures

List of Tables

List of Acronyms

API	Application Programming Interface
ASL	Application Specification Language
BI	Business Intelligence
BPMN	Business Process Model and Notation
CRUD	Create, Read, Update and Delete
DB	Data Base
DBMS	Database Management System
DSL	Domain Specific Language
DSRM	Design Science Research Methodology
DW	Data Warehouse
ITC	ITLingo-Cloud
KPI	Key Performance Indicator
MDE	Model-Driven Engineering
ML	Machine Learning
MTV	Model-Template-View
MVC	Model-View-Controller
OMG	Object Management Group
ORM	Object-Relational Mapping
PM	Project Management
PSL	Project Specification Language
RE	Requirements Engineering
RSL	Requirements Specification Language
SPM	Software Project Management
SRS	Software Requirements Specification
UC	Use-Case
UI	User Interface

1. Introduction

This chapter introduces the general context and motivation of this project, discusses the limitations of the research, and presents the key research goals, the followed methodology, and the document structure.

1.1. Context

IT organizations are complex entities that intend to satisfy their customer's needs and adapt to market competitiveness, aiming to adapt to advancements in technology development. Project management (PM) practices have contributed to increasing the performance of organizations with better control of these processes. The lack of project management practices can bring huge losses to organizations, such as [1] high costs, the product does not fit the needs, or the profit may not compensate the investment, compromising the quality of the product and customer satisfaction. The failure of projects is often due to a lack of communication between stakeholders, poor planning, disregard for requirements engineering practices, and lack of technical documentation [2], [3].

Giving support to companies during the software development process is vital to reducing risks, uncertainties, and costs at the different stages of that process and to help to meet the deadlines. Hence, it is crucial to use software systems to support organizations' current activities and particularly support software development life cycles [4]. Since companies generate data daily, data analysis and visualization techniques are essential to better understand actual or past project performance based on the analyzed data. These tools are crucial to help in planning work, maintaining the information available to the stakeholders to reduce inconsistencies through these processes [5]. In this scope, ITLingo is a research initiative that focuses on improving the rigor and consistency of technical documentation [3], promoting the productivity based on reusability of information and templates, and reducing inconsistencies and ambiguities. In the ITLingo project, some templates were developed to specify project plans, user requirements, and user interfaces aspects of software applications, among other business factors [3], [6]. ITLingo-Cloud emerged inspired by this research focusing on overcoming the abovementioned situations supporting software development processes.

1.2. Problem Addressed

Efficient software development processes are the key to the success of IT-based projects. Project management practices are essential to their functioning, reducing the risks and possible unnecessary costs. Technical documentation is used daily by IT organizations. For instance, project plans or requirements specification documents are crucial in developing and managing the software, reducing software errors, and saving time and costs, which is essential, considering the market's competitiveness [7]. In addition to requirements specifications, other crucial types of documentation, such as test specifications and different types of project data, need to be organized and visualized. Therefore, organizations must have their information stored and available to the involved parties. However, as organizations grow, the amount of information becomes massive and may bring inconsistencies and related problems. Storing,

collecting, and analyzing vast amounts of information is part of each organization's life cycle. Lack of information and technical documentation availability among the stakeholders at the right time may delay the decision-making processes [8]. When people are working simultaneously, it is difficult to visualize, explore, and track project information and the need to comprehend the organization's situation and project performance to form a shared and common opinion [9]. From these artifacts and information emerge the need for tools that would store, aggregate, and visualize that information to communicate insights to the stakeholders involved and improve communication.

As mentioned, there are templates, namely the ITLingo templates (e.g., PSL, RSL, and ASL Excel templates), that have been developed, and that allow a systematic and rigorous specification of technical documentation. However, these templates do not allow collaborative work, which is a barrier to finding better solutions in teams context.

In this research, we propose the "ITLingo-Cloud" platform (or just "ITC" for brevity), which shall be a web-based application to mitigate the problems mentioned above. ITLingo-Cloud shall offer a collaborative environment where it shall be possible to organize, filter, store and synchronize information based on pre-defined templates. Furthermore, ITLingo-Cloud shall provide an intuitive and easy way to manage workspaces, evaluate organization and project performance, track projects with statistics and dashboards to aid in business decisions, supporting software development and project management processes in a unique and integrated platform.

1.3. Research Goals

In this research, we propose an approach to support organizations and their projects, namely to increase performance, manage and analyze information, providing mechanisms to guide software development processes. ITLingo-Cloud platform intends to provide a collaborative environment with reusability, allowing data synchronization features with other technologies and approaches from the ITLingo initiative, namely PSL, RSL, and ASL Excel templates [3]. This research involves analyzing and studying collaborative platforms and cloud tools to identify a solution to manage organizations and workspaces according to agile processes and support software development life cycles. We shall evaluate this approach by exploring the results with experiments and discuss the results produced with tests mainly considering the following qualities: learnability and usability.

So, the main research goals (RG-i) of this project are:

RG-1. Design and implement the ITLingo-Cloud as a web platform that would include the following features: (a) support the management of organizations, (b) support the management of workspaces, (c) support collaborative work in particular agile processes, (d) support the storage, visualization, and analysis of information, (e) support synchronization of data from Excel files, in particular, the PSL Excel Template, as well as document automation for project management, (f) integration with ITOI, a cloud IDE that allows editing and developing code through the browser.

RG-2. Evaluate and discuss the results through tests and compare the ITLingo-Cloud with other related tools to make conclusions about the developed work.

1.4. Research Methodology

This project follows the Design Science Research (DSR) methodology, a well-established research approach in Information Systems (IS) and other disciplines. The DSR is an iterative methodology that combines principles, practices, and procedures [10]–[12].

Hevner et al. propose a set of guidelines for the application of DSR in the area of information systems, namely including the following aspects [10]:

Design as an Artifact: The main goal of DSR is to create viable artifacts in the form of a construct (e.g., a software application or tool), a representation (e.g., a new language or extension of a previous notation), a technique (e.g., a process or method), or an instantiation (e.g., a case study that applies such artifacts). In this research, the key artifact is the ITLingo-Cloud, a multi-organization and multi-workspace collaborative platform.

The problem's relevance: This methodology focuses on addressing unsolved business problems. As discussed, this specific project intends to solve collaborative issues in project management and organizations in general, namely data synchronization aspects, problems in versions, and the need to support and guide agile software projects. In this sense, ITLingo-Cloud contributes to accelerating the software development process and managing projects intuitively, according to standards guides.

The design evaluation: The design artifact's quality (e.g., measures in terms of utility or efficacy) is demonstrated rigorously through a well-executed evaluation method.

Research contribution: Effective DSR offers a clear and demonstrable contribution to the design artifact's application, such as design foundations and design methodologies. ITLingo-Cloud contributes to guiding and accelerating the software development processes through analysis and management of information, supporting agile PM methodologies.

Research Rigor: The DSR depends upon rigorous methods application in the evaluation and the construction of the design artifact. We evaluated our artifact based on two experiments and a user session assessment.

Design as a search process: The search for a compelling artifact depends on the available ways to reach desired outputs while the rules in the problem environment are still satisfied. We iterated several times to implement and evaluate our solution.

Communication of the results: The research presentation is effective from technology and business perspectives. The communication of the results of this research involves the writing and presenting the master's thesis.

Complementary, DSR methodology recommends a process based on six phases [10]:

Problem identification and motivation: Define the specific research problem, justify the solution's value, and motivate the researcher to investigate the answer. In this investigation, this phase took place in October and November of 2021, with the study and analysis of this research's context, motivation, and background.

Define the objectives of a solution: Infer the goals of a solution for the defined problem and the knowledge of what is achievable. In this project, we propose a cloud-based platform to

improve the project management process by mitigating some of the discussed problems. This phase took place in November and December of 2021.

Design and development: Create the artifacts, the core part of the project. This phase includes multiple iterations with several tasks. These iterations started in October 2021 and took place until June 2022.

Demonstration: Demonstrate the efficacy of the artifacts in solving the problem. In this phase, we apply the solution to two experiments with different complexity levels: Experiment A (The ITLingo-Cloud Project) and Experiment B (The BIER's Audit System). This phase occurred between June and July of 2022.

Evaluation: Analyze and measure how well the artifacts support the problem initially defined, comparing different approaches using objective metrics. We evaluated the solution and compared it with the related work based on two experiments and a user session assessment. This phase occurred between June and August of 2022.

Communication: Communicate the problem, the artifacts, and the design, considering relevance, utility, novelty, and effectiveness to researchers and other relevant audiences. This phase occurred between June and September of 2022 with the writing and discussion of the MSc dissertation.

1.5. Dissertation structure

This document is structured in the chapters and appendices:

Chapter 1 (Introduction) explains the context of this research, the motivation, the importance of a platform as ITLingo-Cloud, and the problems this tool solves. Then, the general research goals are established, followed by a discussion of the research methodology and document structure.

Chapter 2 (Background) presents and discusses the investigation's fundamental theoretical and technological concepts, including project management theory, data analysis and visualization aspects, the ITLingo initiative, as well as the developed technologies used to build this solution (e.g., Django, Chart.js).

Chapter 3 (Related Work) presents and discusses tools related to ITLingo-Cloud, namely collaborative project management tools, and cloud IDEs, exploring relevant features from each approach.

Chapter 4 (ITLingo-Cloud Requirement Aspects) describes the solution proposed, presenting the ITLingo-Cloud platform, its architecture, implemented requirements, and technological aspects.

Chapter 5 (ITLingo-Cloud Design Aspects) presents some design aspects of the ITLingo-Cloud platform, namely some screens and data visualization aspects, among other supported features.

Chapter 6 (Validation) discusses how the evaluation and testing of this research were conducted, namely through two experiments where we applied the ITLingo-Cloud platform. This chapter also compares our solution with related approaches, according to different functionalities and characteristics.

Chapter 7 (Conclusion) presents the main conclusion explaining the research's contributions and referring open issues and future work.

Complementary, Appendix A includes the user assessment guide used during the evaluation phase.

2. Background

This chapter introduces and discusses general concepts and technologies underlying this research, namely agile project management and data analysis aspects. This chapter also introduces the ITLingo initiative, mainly focusing on PSL and RSL languages. Finally, it briefly introduces the technologies used in the project.

2.1. Agile Project Management

Project management is a set of practices essential to maintain control of the scope, quality, schedule, budget, resources, and risk. These practices are crucial to compliance with the project objectives. PM is known as "the application of knowledge, skills, tools, and techniques to meet the project requirements", according to PMBOK. PM practices involve identifying requirements and addressing the stakeholders' needs as the project is planned and carried out [2]. In this way, the performance increases consistently, improving customer satisfaction and speeding up the product's delivery according to the expectations [13]. Well-known international PM frameworks and guidelines are PMBOK Guide [14], ISO 21500 [15], IPMA ICB [16], and PM² [17].

The process to be followed depends on the project's type, size, complexity, and duration [18].

Traditional approaches follow a "waterfall" model: The client communicates the plan and expects relatively clear results at the project's beginning [19]. In this approach, projects must define boundaries, and the plan should be followed as precisely as possible, finalizing within the planned time, budget, and scope. However, the reasons for the inappropriateness of these traditional approaches are the difficulty of dealing with the uncertainty and the change needs in current IT projects [20]. For instance, status values and principles such as the Manifesto for Agile Software Development were created to solve these restrictions[21]. This principle was written for agile software development but can also be applied to agile project management. Traditional approaches are more appropriate for projects with precise initial requirements and goals. On the other hand, agile project management approaches are better for projects with unclear or incomplete goals and where change is frequent [20].

Agile processes are focused on adaptability to changes during the project lifecycle and promote more communication and collaboration between stakeholders. In this approach, projects are managed iteratively, with frequent modifications in the project plan, focusing on fast implementation. Each interaction is usually short, achieving better control of uncertain projects. Agile project management also reduces risks and guarantees product quality [20].

Scrum and Kanban methods are popular agile methodologies used globally.

Scrum defined few roles (Product Owner (PO), Scrum Master (SM), and the Development Team) with specific responsibilities. The typical size for sprints is one, two, or four weeks. The team manages the project with two artifacts: a Product Backlog, and a Sprint Backlog. To achieve the goals, typical activities such as Backlog Refinement, Sprint Planning, Daily Scrum Meetings, Sprint Reviews, and Sprint Retrospectives are performed.

On the other hand, Kanban is not as prescriptive as Scrum. In Kanban, there are no defined roles and no emphasis on meetings or artifacts [22]. This methodology has its basis in the Just-in-Time (JIT) premise. It uses a visual Kanban board, limiting work in progress by reducing the number of tasks to be implemented [22].

2.2. Data Analysis and Visualization

There are a large number of applications to whom data analysis is crucial. For example, in health domain, the capacity to present and analyze data understandably is critical to the success of public health surveillance. Also, in project management, data visualization software allows to summarize and present information collected from different sources and assists developers and managers in visualizing performance in a shared way [23].

The human brain responds better to visual information when compared to plain text [9]. Since organizations generate a massive amount of data, it is easier to analyze and understand it visually. Visual information improves communication, reduces misinterpretation, and clarifies massive or complex information, being essential to sharing ideas with accuracy and efficiency. Data visualization techniques use visual representation of data and usually perform data reduction, and transformation [9]. For instance, dashboards enable users to investigate and track trends, predict outcomes, and discover insights, using quick scanning and understating key metrics [23].

Standard visualization techniques are [24] line charts, bar charts, scatter plots, and pie charts. The choice of the chart or visual representation depends on the data type that needs to be represented as well as the relationship among elements of the data, so it is vital to understand the concrete problem to choose the better representation to be used and, so, to show patterns, trends, and relationships correctly.

Descriptive statistics are used simply to describe and summarize the data logically, meaningfully, and efficiently, usually reported numerically in text, tables, or graphical forms, focusing on summarizing the characteristics of a data set (e.g., distribution, central tendency, variability). Inferential statistics involves using the available or representative sample data about a sample variable to make a valid inference or estimation about its corresponding unknown population parameter. This type of statistics uses a sample to make reasonable guesses about the larger population. (e.g., randomly selecting a sample of student graders to make estimates and test hypotheses about the whole population) [25].

2.2.1. Data Types

To analyze, report, and interpret data, as well as to understand and apply the findings, is essential to have a basic understanding of data and variables. The combination of the data types that compose the dataset influences the choice of the data visualization to be implemented. At the highest level, data can be separated into two categories: Quantitative and Qualitative. Quantitative data is countable or measurable, while qualitative data is usually interpretation-based, descriptive, and not easily measured [25].

Quantitative data can be continuous and discrete (ratio) data. Continuous data are measured, have a constant sequence, or exist in a continuous range. This data type can be meaningfully divided into smaller or finer increments. Height, weight, temperature, and length are all examples of continuous data. Ratio data, considered another form of continuous data, have the same properties as interval data but the distinguishing property of ratio data is that it has a true definition of an absolute zero point. Age, height, weight, heart rate, and blood pressure are also Ratio data.

Qualitative data can be classified as nominal or ordinal. Nominal data (also called categorical data) represent types of data that may be divided into groups (e.g., race, sex). This data type can be classified as dichotomous (two categories) or polytomous (more than two categories).

Ordinal data is data in which the values follow a natural order, while discrete data can be counted (e.g., age, educational level). Nominal data can only be classified, while ordinal data can be classified and ordered [25].

2.2.2. General Data Visualization Techniques

Data visualization type depends on the datasets, the proposal we want to present, and the findings we want to show. Hence, it is crucial to understand the specific data and the most common data visualization techniques to transform the data into an effective visualization. Only a good choice of the chart type allows to project and present patterns, trends, as well as relationships among elements of the datasets [24].

Common visualization techniques are the followings: Bar Chart, Line Chart, Pie Chart, Scatter Plot, and Histogram.

Bar Chart. Bar charts are one of the most common ways to visualize data and are used to compare quantities of different categories, revealing highs and lows at a glance and helping to understand trends in the data.

Line Chart. Line charts are one of the most frequently used chart types and can be used to compare changes and observe trends over a period of time. Line charts connect individual numeric data points being useful to visualize a sequence of values.

Pie Chart. Pie charts are helpful to compare the parts of a whole, showing relative proportions or percentages of information, being a fast way to understand proportional data. However, the pie wedges must be limited to six once it becomes too hard to meaningfully interpret the pie pieces when the number of wedges gets too high. Bar charts are a better option in cases that need more than six proportions to communicate.

Scatter Plot. Scatter plots are adequate to investigate relationships between different variables. This visualization technique provides an effective way to detect trends, concentrations, and outliers being helpful to guide where to focus some investigation efforts in the future. The story behind some data can be enhanced with a relevant shape or by adding a trend line.

Histogram. Histograms are adequate to analyze how the data are distributed across groups, helping to understand the distribution of the data, for example, the number of customers by company size or student grade on a specific exam. Histograms and bar charts look almost identical but display different types of data. Histograms present quantitative or numerical data, whereas bar charts depict categorical variables. The numerical data in a histogram is usually continuous [24].

2.2.3. Project-Specific Data Visualizations Techniques

PM dashboards typically support various features that foster collaboration and decision-making and are crucial in monitoring projects' success, allowing for summary and centralizing of information. These types of dashboards help track the progress of a project, working hours, outcomes, etc. Performance dashboards integrate several organizations' data, reducing redundancy and enabling people to measure, monitor, and track the tasks and processes needed to achieve project goals, supporting the development process analysis [26].

Dashboards may display individual developer performance as well as metrics that indicate how many lines of code or features software developers contributed to a specific repository. When used by the developers, this information is relevant to support personal performance monitoring and improving productivity. These dashboards also aggregate information for other

stakeholders, such as team leads or project managers, to improve the working environment and development process [27].

Agile teams have to deal with a lot of data to help them manage and reflect on their process, namely to track their performance across sprints. If monitored and used correctly, this information helps detect problem areas and improve processes efficiency and may detect and avoid project failures [26]. To support agile teams, dashboards provide an essential role, in particular, to visualize all open issues for a specific project allowing track who are assigned and what is the priority for its solving. Furthermore, current projects nowadays rely on continuous integration and deployment services. In this sense, dashboards are also valuable for DevOps to visualize the performance of running services or tracking outages [27].

Relevant charts in the scope of our research are Gantt Chart, Burn-up chart and Burn-down charts:

Gantt Chart. Gantt chart is a typical chart in project management tools, providing a complete overview of tasks, and dates, allowing project managers to drill down as needed to make informed decisions and hit deadlines for a successful project. This chart type is helpful to see what needs to be accomplished, representing the milestones, highlighting the start and finish dates elements for each deliverable, and illustrating resource planning to see how long it took project stakeholders to hit specific milestones and how that was distributed the tasks over the time, displaying the project schedule intuitively [24].

Burn-down Chart. A burn-down chart indicates how much work remains to be completed, keeping the end goal in mind and allowing to predict the likelihood of completing work during a period of time [28].

Burn-up Chart. A burn-up chart shows the project's progress over time and is an easy-to-understand figure of the status and rate of work done. This chart presents the increasing amount of work accomplished as a function of time that is reported on a regular basis [28].

2.3. ITLingo Initiative

ITLingo research initiative is an approach to specify technical documentation consistently and rigorously, focusing on reducing inconsistencies and ambiguities among stakeholders and promoting reusability and productivity [3]. This research explores ways to automate human-intensive tasks in producing technical documentation and related artifacts, focusing on implementing computational tools, platforms, or reusable libraries of technical specifications. This initiative includes different languages such as Requirement Specification Language (RSL), Application Specification Language (ASL), Project Specification Language (PSL), and Enterprise Architecture Specification Language (EASL). Some templates are provided as MS-Excel templates like PSL and RSL Excel templates or with the Xtext framework in an Eclipse-based tool called "ITLingo-Studio" [3].

ITLingo-Cloud platform shall integrate some of these technologies, in particular, the ITLingo excel template files. ITLingo-Cloud shall use these templates to store data, reuse artifacts, synchronize information, and generate dashboards and statistics based on stored data.

ITLingo PSL. PSL (short name for "Project Specification Language") is a language to specify projects strictly according to PM standards and frameworks. PSL approach has strong characteristics that are distinct from other related tools and allows consistently defining project-related aspects, such as product backlog, sprint backlog, stakeholders, scope, costs, time, quality, and risks. These templates include analysis reports, graphics, and document automation utilities [29]. The main goal of these templates is to reduce inconsistencies and specify project plans rigorously and consistently, saving time with features like the generation of documentation automatically as well as project management reports used during the software building life cycle, providing the automation of tasks in project management and software engineering activities. Figure 2.1 presents an example of a PSL Excel Template filled with data during the development of the ITLingo-Cloud platform.

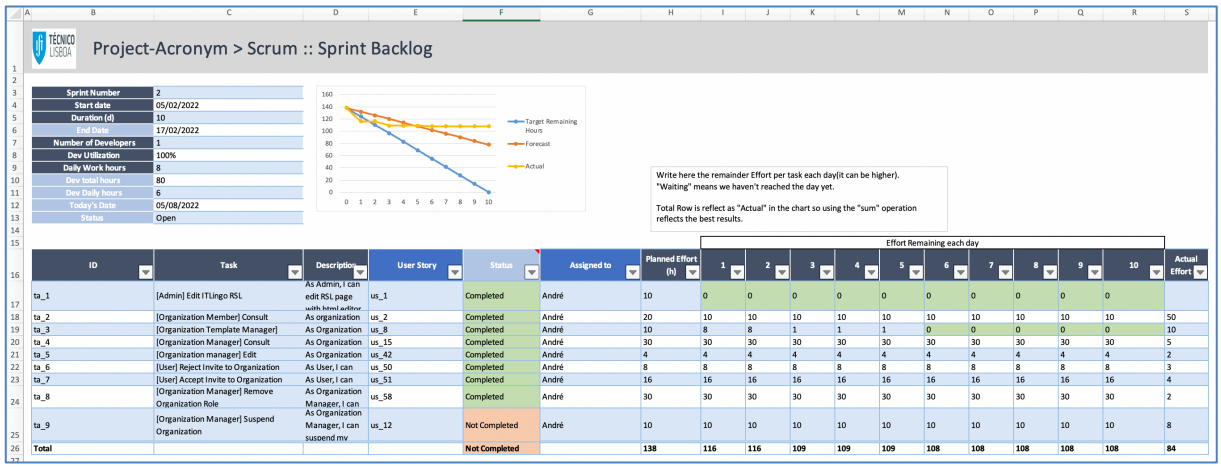


Figure 2.1 - Example of a PSL Excel Template page filled with project data

ITLingo RSL. RSL (short name for "Requirement Specification Language") is a specification language developed to mitigate problems that may arise when writing requirements. It can be used and adapted by multiple users and organizations with different processes/methodologies. RSL language allows the specification of stakeholders, actors, data entities, use cases, and goals, among other business concerns. Attractive advantages have been developed in this scope, namely with artificial intelligence techniques mainly to support the use of natural language for writing rigorous requirements specifications that can be automatically validated [3], [13]. To specifications are structured in a mathematical way using constructures according to concerns that exist at different abstraction levels. Figure 2.2 illustrates a simple example based on the ITLingo-Cloud project of an RSL specification that defines the actor "Workspace Editor", who participates in the use-case "Manage Workspaces", which involves the management of the data entity "Workspace".

```
Actor w_Editor "Workspace Manager": User
DataEntity e_Workspace "Workspace": Document [
    attribute Id "Workspace ID": Integer [isNotNull isUnique]
    attribute State "Workspace State": DataEnumeration enum_WorkspaceState
    attribute Title "Workspace Title": String (30) [isNotNull]
[...] ]


UseCase uc_1_ManageWorkspace "Manage Workspaces": EntitiesManage [
    actorInitiates w_Editor
    dataEntity e_Workspace
    actions Create, Read, Update, Delete]
```

Figure 2.2 – Example of an RSL specification

2.4. Used Technologies

ITLingo-Cloud platform is developed using several technologies to build the platform structure and implement its data extraction, storage, and visualization features, namely: Python, Django, and Chart.js.

2.4.1. Python

Python is a high-level programming language designed to empathize code readability with the use of significant indentation. This programming language is commonly used to develop software, automate tasks, and support data analysis and visualization functionalities. Python supports various everyday tasks, has vast applications, and is widely used. It has a lot of libraries and modules available, is easy to learn, and for that reason, is adopted by many universities as the first programming language to be taught. Python also is used by some of the largest companies in the world, such as NASA, Google, YouTube, Yahoo!, and Apple in their applications [30].

Python libraries are a suit of code used to aid in developing software programs, supporting and helping depending on the development needs. Several python libraries were used during ITLingo-Cloud implementation to implement specific system functionalities. The python library most used during this research was openpyxl to load, handle and extract data from excel files stored on the platform. Another essential and powerful python library used during the development of our solution was python-docx, which was used to develop the report automation features. This library was used for manipulating word documents and templates pre-filled with information stored on the database, in particular, to generate new reports with workspaces' and organizations' data.

Openpyxl. Openpyxl is a python library to read and write Excel files, being used for data analysis, data copying and extraction, data mining, drawing charts, styling sheets, adding formulas, and other relevant functions. This library can leverage business activities according to specific needs, automating repeated tasks, and supports extensions such as xlsx, xlsm, xltx, etc. Each excel spreadsheet is represented as a workbook in this library and can be manipulated with a few lines of code [31].

Python-docx. Python-docx is a python library to manipulate word documents (with docx extension). This library allows the modification of existing word files or the creation of new ones, providing several functions to handle pictures, index tables, footers, and headers, among other functionalities available in word documents [32].

2.4.2. Django

Django is a web development framework based on Python that enables the rapid development of software. This Framework is a powerful and flexible framework that supports the development of dynamic web applications and web services in a short time, focusing on automating as much as possible and adhering to the DRY ("don't repeat yourself") principle [30]. Django has some advantages compared to other related frameworks. Since Django is based on Python, this tool promotes flexibility and facilitates integration with different languages and tools. It was designed for integration and reusability. It has an automatically generated administrative interface, Django Admin, which is easy to use, allowing the creation, editing, and deleting of objects in a database in a flexible and customizable way, providing an interface with all the database tables.

Using Django Admin, we can deal with Django permissions. This feature enables the definition of specific permissions to users through the link to the Django user system, decreasing possible errors or inconsistencies. Additionally, Django provides another user management facility, the concept of Groups, which allows the organization of users who share the same permissions [33].

Django promotes agile software development patterns and practices. This web framework follows the Model-View-Template (MVT) architecture, which helps build clean and maintainable web applications [30].

Django framework provides a powerful built-in database management GUI to handle and access data in the database named object-relational mapping (ORM). This functionality is designed to facilitate data processing, data management, and data visualization and is used directly with python to interact with the database. ORM abstracts the access to database tables through Python classes providing an API to store, manipulate, and retrieve objects from the database. SQL knowledge is helpful but not required to use this tool [33].

Each model represents one database table that may be specified inside each file *models.py*. The database can be generated with a few simple commands. ORM also provides methods to manipulate and create querysets to the database allowing to model and perform functions to the data.

Although Django is adequate to interact with the database easily, there are better tools to perform operations with large amounts of data with better performance [34].

2.4.3. Chart.js

Chart.js is a popular JavaScript library used for data visualization. Chart.js uses HTML5 canvas to render different chart types, supporting the most common types such as bar, line, area, radar, polar area, scatter, bubble, and pie. These common types of charts also can be mixed (for example, match a bar and line charts) to provide a clear visual distinction between datasets [35].

To create a visualization is necessary to set up the canvas element in which the chart will be rendered, configure the options of the chart, and supply the data to be visualized.

This library allows to save time and effort, allowing to reuse of code to implement quickly new types of visualizations, for example, by changing the configuration and adjusting the settings for the desired chart type and binding the new datasets.

The main advantages are that this library is open source, the charts are automatically resizable according to the device used and the actual window size, and easy to use compared with other

approaches. For example, D3.js, another similar Javascript library, has a much steeper learning curve.

The configuration involves set properties that dictate the chart's behavior and type. In this step is possible to configure the chart type, fonts, styles, and tooltips and assign the datasets to be visualized. The data property is set during the configuration and contains the datasets to be visualized and their labels (for example, the label for the x-axis and y-axis). Chart.js supports multiple X and Y axes and multiple scale types, cartesian (linear, logarithmic, category, time, and time-series) or radial (linear radial axis) [35], [36].

Chart.js automatically matches the data with the label of the same index. Using this library with object-oriented programming languages is advantageous once the data can be parsed, specifying the parsing property. This is one of the reasons why this library is adequate to use with the Django framework.

Another valuable feature of this library is its interactivity options, including clicking on the legend to toggle a dataset's visibility on and off and data points tooltips triggered by hover showing the selected data point according to the user mouse coordinates, such as observed in Figure 2.3.

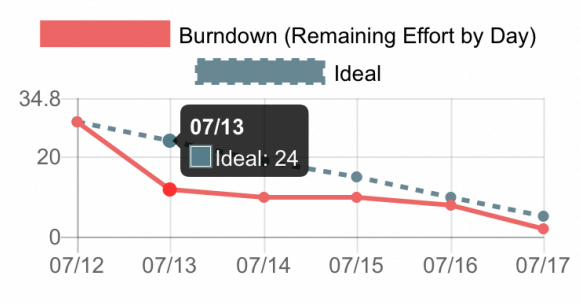


Figure 2.3 – Interactivity element in Chart.js

3. Related Work

This chapter presents and discusses the related work. This includes the analysis of collaborative project management tools, (including their reports and dashboards features) and cloud IDE tools.

3.1. Collaborative and Project Management Tools

This section analyses collaborative project management tools, namely: Microsoft Project, Jira, Trello, and ClickUp. We chose these tools because they are well known, and we already have previous experience with some of them. We also decided to cover tools with different characteristics to compare their features with the ITC. We overview the following features: organization management, costs management, risk management, communication tools, issue tracking, data import, artifacts storage, report generation, and agile process supported. Moreover, we analyze the reports and the data visualization mechanisms provided by each tool.

Organization Management. Organization management allows the creation of organizations to store and manage project-related information.

Costs Management. Cost management involves estimating, allocating, and controlling project costs. Tools with this feature may allow tracking, viewing, or analyzing costs over time as the project progresses [37].

Risk Management. Risk management involves identifying, analyzing, and prioritizing project risk factors. This feature also may allow the definition of contingency plans and the methods to be used in tracking the various risk factors [38].

Communication Tools. Communication mechanisms involve changing information among stakeholders.

Issue Tracking. Issue Tracking involves organizing, prioritizing, and tracking these issues and bugs until they are resolved.

Data Import. Data Import involves importing records with data from Excel files, namely to create dashboards or manage project information.

Artifacts Storage. Artifacts storage allows users to store files in a hierarchical structure organized by projects or organizations.

Report Generation. Report generation involves extracting data from the database and using it to export reports.

Agile Process. The agile process supported feature (in our analysis) involves tools that support scrum and kanban processes previously explained in Section 2.1.

We analyze the following tools:

Microsoft Project. Microsoft Project (also named MS Project) is a pioneer project management tool used in enterprise and academic contexts. This tool provides a Gantt chart to manage the work packages, activities, milestones, and dependencies. Resource Pool can be created to manage resources' information such as name, maximum resource units, and standard rate. Additionally allows the creation of the map of resources allocations per activity. These aspects

provide a complete view of the project schedule and costs. Pre-built reports can help visualize the project progress, resources, earned value analysis, and other key performance indicators (KPIs) [37].

Jira. Jira is a web-based application designed for agile teams to optimize project planning execution and tracking, used for Scrum and Kanban methodologies. With Scrum methodology, Jira allows the easy creation of the product backlog and sprints. On the other hand, following the Kanban methodology, this tool provides a simple board with a drag and drop functionality. Jira allows tracking issues, bugs, stories, or tasks and can integrate with other systems like Github, Slack, Google Calendar, or Zoom. Additionally, it provides customizable dashboards and pages to share information with other stakeholders (e.g., meeting notes or requirements) [39].

Trello. Trello is a collaborative project management tool. The main advantages of this platform are that it is simple, flexible, and customizable for managing projects or intuitively organizing tasks because it is user-friendly and can be better for beginners. Besides that, Trello is available on the web, desktop, and mobile versions. It is based on Kanban methodology and provides the typical board to create cards to organize, manage, track, and share with teammates. Trello also offers other relevant features to automate tasks and actions, namely, configuring automatic emails or defining rules with actions the system performs. With this tool, there is the possibility to integrate with platforms like Slack, Google Drive, Dropbox, or Evernote. The main disadvantages are the limited functionalities and dashboards compared to related systems. Furthermore, this tool is only adequate for the Kanban methodology [40].

ClickUp. ClickUp is a collaborative and project management platform that can integrate with other tools (e.g., Asana, Trello, Jira Software) or with excel files. This web-based application is customizable and may adapt to different needs, not focusing on agile methodologies such as Scrum and Kanban. This tool supports documentation management, goals definition, and team management. ClickUp provides templates for specific areas, not focusing only on Boards and Sprint templates for project management but also offers pre-built templates for education, sales, finance, or other fields. This approach is characterized by providing a hierarchy composed of Workspaces. Inside Workspaces, customizable Spaces (e.g., teams or departments) can break down further into Lists (e.g., Sprints) created to group related Tasks. The Lists can also be grouped in Folders. Inside Lists can be created Tasks, fundamental pieces of work, that can be visualized through different views (e.g., List, Board, Box, Calendar, or Gantt) [41].

3.1.1. Main Features

We compare these tools and summarize the main findings in Table 3.1. We investigate whether each tool supports specific project management and software engineering aspects. If a tool supports a particular functionality or feature, we put a "Y" for yes and “N”, no, otherwise.

Table 3.1 - General features comparison of each tool.

Feature		Microsoft Project	Jira	Trello	ClickUp
Organization Management		N	Y	N	Y
Risk Management		N	Y	N	N
Communication Tools		Y (1)	Y	Y	Y
Issue Tracking		N	Y	Y	Y
Data Import		Y	Y	Y	Y
Artifacts Storage		Y (1)	Y	Y	Y
Reports Support		Y	Y	Y	Y
Costs Management		Y	N	N	N
Agile Process Supported	Scrum	N	Y	N	Y
	Kanban	Y (1)	Y	Y	Y

(1) Only supported in web-version.

Some of the analyzed features exist in almost all of these tools. These include task scheduling, resource management, collaboration, and artifacts management. Microsoft Project has many features to control costs [37], [38]. All contain suitable communication mechanisms. Jira and ClickUp are better for supporting several workspaces inside the same organization, providing a hierarchy to manage and track at the organization level [39], [41]. Most of these tools are suitable for agile processes, namely scrum and kanban. However, MS Project and Trello only support the kanban framework [40]. All these tools include web and desktop versions, while some also support mobile ones.

3.1.2. Reports Supported

In project management, reporting offers multiple views on management information. Collaborative project management tools provide mechanisms to automatically generate such reports. Table 3.2 shows the types of reports commonly provided by these tools.

Table 3.2 – Report types supported by each tool

Report Type	Microsoft Project	Jira	Trello	ClickUp
Project Status Report	Y	Y	N	Y
Project Charter	Y	Y	N	N
Team Charter	Y	Y	N	Y
Risk Report	N	N	N	N
Variance Report	Y	N	N	N
Baseline Work Report	Y	Y	Y	Y
Baseline Cost Report	Y	N	N	N
Cash Flow Report	Y	N	N	N
Resource Report	Y	N	N	N
Earned Value Report	Y	N	N	N

By analyzing Table 3.2, MS Project is the most complete tool in what concerns reports generating. MS Project supports more report mechanisms related to resource management, being the only one that supports cash flow, and earned value reports [37], [42]. All approaches provide baseline work reports, allowing to show the planned schedule over time. Furthermore, all tools allow the creation of personalized reports, being possible to reuse them [37], [39]–[41].

3.1.3. Dashboards Supported

Dashboards are essential to summarize several project information, enabling understanding, monitoring, and tracking the project's performance as well as guiding in possible necessary project decisions. Table 3.3 presents an analysis of data visualizations supported by each tool.

Table 3.3 – Chat types supported by each tool

Chart Type	Microsoft Project	Jira	Trello	ClickUp
Burn-down Chart	Y	Y	Y	Y
Gantt Chart	Y	Y	N	Y
Burn-up Chart	Y	Y	N	Y
Cost Overview	Y	N	N	N
Kanban View	Y (1)	Y	Y	Y
Table View	Y	Y	Y	Y

(1) Only supported in web version.

Our findings concluded that all tools support Burn-down charts and table data views. MS Project supports several cost visualization mechanisms, allowing the visualization of planned costs, remaining costs, actual costs, cumulative costs, and baseline costs [38], [42]. The remaining tools also support useful visualizations despite not being related to project costs. All tools allow the visualization of tasks in both table and kanban views [37], [39]–[41].

3.2. Cloud IDE Tools

We explore some cloud tools specialized in different areas, in particular these mostly close to ITLingo-Cloud, such as: cloud IDE support, workspace hierarchy, and information storage.

Google Cloud. Google Cloud Platform (GCP) is a tool to develop, deploy, and manage resources and data. This tool has many benefits, such as autoscaling computer power, distributed memory cache, task queues, and databases, allowing hosted applications on the cloud. Large companies such as The Home Depot, Coca-Cola, Evernote, or Vimeo use tools and services from the Google Cloud Platform [43], [44].

GCP provides the Google Cloud Console to developers, an intuitive tool that supports many high-level management tasks. The Google Cloud Shell is a free and interactive web-based shell that allows the development of applications directly on the browser. Includes version control with source repositories, container builder, and deployment manager for build processes and continuous delivery applications, IDE integrations, and cloud service emulators for local development. Cloud Shell runs on top of a g1-small Google Compute Engine instance supporting the Linux operative system, and comes preconfigured with many developer tools such as Google Cloud command-line tools and SDKs. Additionally, the Cloud Shell supports many programming languages, including Java, Go, Node.js, Python, Ruby, PHP, and NET Core [44].

GCP Platform also offers several cloud-based data management tools, in particular, to fuel data-driven transformation. Key Google Cloud components, namely the BigQuery, a cloud data serverless data warehouse that allows tabular data storage, queries in large datasets, and built-in machine learning capabilities. GCP can connect with various other data sources, supporting cloud data transfer functionalities. This platform is also useful to report templates to visualize data, design compelling reports, and share them with your team or the world [43], [44].

Code Ocean. Code Ocean is a cloud-based platform that allows researchers and developers to share, discover, manage files, and run code. The main goal of this platform is to increase teams' workflow enabling them to reproduce results and share knowledge and discoveries. This tool allows search for published repositories for projects, results, and data allowing users to view and download for free. This approach offers a cloud-based IDE pre-configured with R, Python, Matlab, and C/C++ programming languages, including mechanisms to manage information as simple folders and collaborate in groups with access permissions. Code Ocean allows users to create workspaces called Compute Capsules. Each Compute Capsule includes an environment to ran code and store data, and visualize it using a browser, promoting reusability between different projects. It is possible to invite collaborators and share software projects within teams or organizations with access levels of view only or edit. Cloning previous research projects allow developers to directly modify code or data without wasting time on initial setup [45].

AWS Cloud9. AWS Cloud9 is another cloud-based IDE that provides flexibility to the developers to create the environment, write, run, and debug code with a browser. Cloud9 can develop serverless applications supporting popular programming languages (e.g., JavaScript,

Python, or PHP) and is preconfigured with SDKs, libraries, and plugins needed to accelerate the development process without installing or configuring files. This approach is helpful to run code without managing servers directly, facilitating the management of the resources and the deployment with Lambda functions. This platform provides a collaborative development environment, able to be shared with the team in real-time, running on an Amazon EC2 instance or any existing Linux server that supports SSH. This approach is helpful to accelerate the software development process, configure the environment easily, and deploy applications quickly [46].

Github Codespaces. GitHub Codespaces, as previous tools, provides a development environment hosted in the cloud and focuses on accelerating the software development process, in particular, to start a new project quickly and add new developers to the project. This approach is based on Visual Studio Code (VS Code) and works directly with GitHub repositories. Like the previous tools, it is available on the browser. This approach provides a text editor, terminal, debugger, version control, and the entire ecosystem of extensions like VS Code. GitHub Codespaces easily enables the creation of an environment (or dev container) for the GitHub repository. The developers can specify a Docker environment using a dev container for development. With this technology, anyone using the project repository can access the same tools, clone the repository into an environment, start a Docker container on a remote server, and install the entire software stack required by the project [47].

Each technology provides distinct advantages. Google Cloud is one of the most complete software available on the market today, developed to transmit security and practicality for doing business. This tool allows the development of applications using its cloud IDE and establishes Customer Relationship Management (CRM) management programs. Google Cloud Shell Editor has pre-installed and up-to-date tools for maximizing development productivity, notably simplifying the use of Kubernetes. This approach also provides a deployment manager, an infrastructure deployment service that automates the creation and management of Google Cloud resources. AWS Cloud9 also preconfigures the development environment with SDKs, libraries, and plug-ins. Furthermore, it easily allows resource management for serverless applications with Lambda functions and offers a collaborative chat to communicate without leaving the IDE. Github Codespaces allows working directly with GitHub repositories, isolating dependencies between projects with containers and docker-compose. The main advantage of this tool is that it is customizable with the same extensions of Visual Studio Code, allowing it to synchronize with other technologies. On the other hand, Code Ocean promotes computational reproducibility, particularly in research projects, allowing a quick share of results. This approach enables the storage of information, allowing software development and data analysis.

4. ITLingo-Cloud Requirement Aspects

This chapter introduces the main aspects of the proposed solution. ITLingo-Cloud is a multi-organization and multi-workspace collaborative platform to store, manage and analyze projects-related data, providing visual insights to aid in business decisions. Users can easily set up their environment and intuitively manage workspaces and technical documentation. This tool has been designed and developed considering the solutions available on the market, specifically to include standard features included in agile project management tools presented in chapter 3.1 and inspired by emerging collaborative solutions like cloud IDE tools presented in chapter 3.2, which shall provide the necessary conditions to develop software quickly and with quality. ITLingo-Cloud shall support multiple technologies and languages studied in previous research projects (e.g., PSL Excel Template, RSL Excel Template), promoting data synchronization with predefined templates filled with project data. ITLingo-Cloud shall support automation tasks, automatic text extraction, or document automation, storing organizations' information and project data and using it to provide valuable dashboards. The ITLingo-Cloud architecture encloses a relational database that stores these data and uses a popular Django framework (as discussed in Section 2).

Figure 4.1 illustrates the partial domain model of the ITLingo-Cloud, which introduces its main concepts: organization, workspace, library, template, document, product backlog, sprint backlog, user, and user role. These concepts are explained in detail in the following sections.

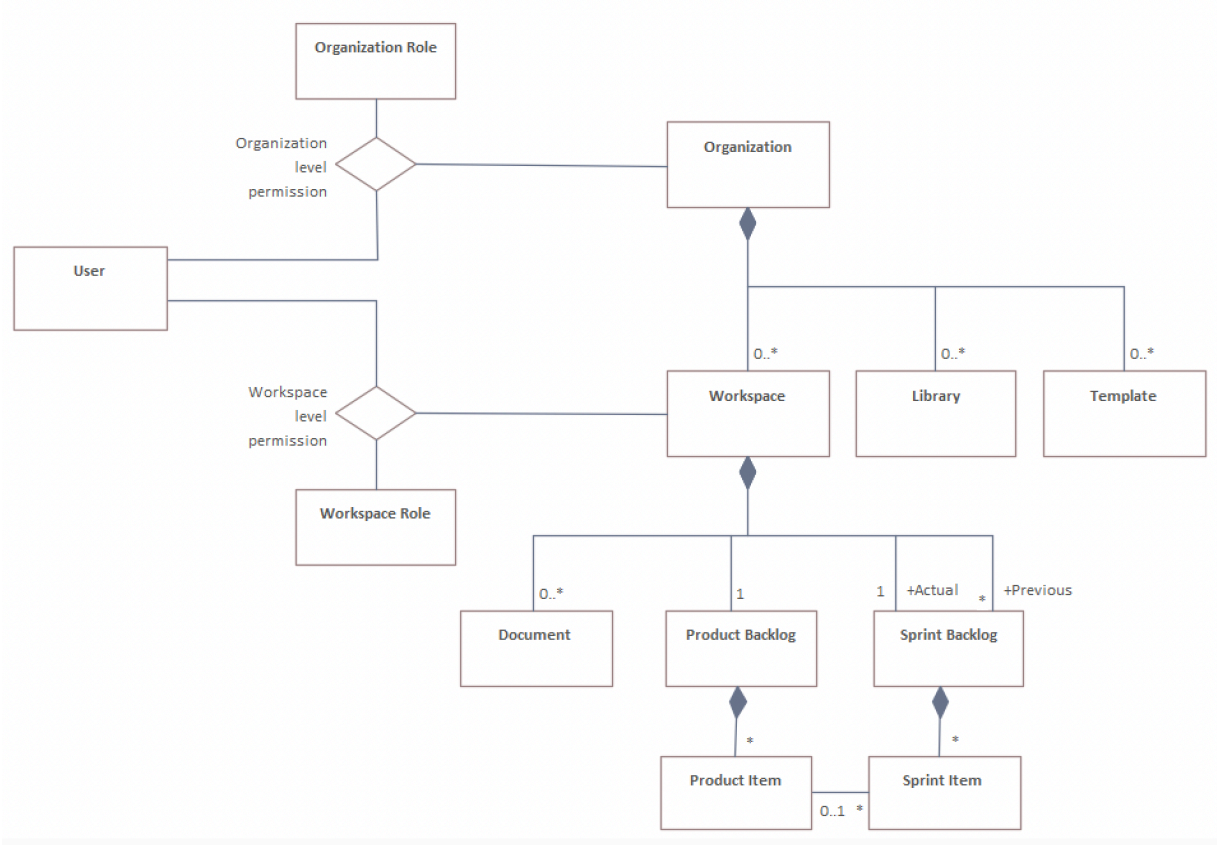


Figure 4.1 - ITLingo-Cloud partial domain model (UML class diagram)

4.1. Users and Roles Management

Users may have different roles, determining the permissions to access the platform's pages, access certain information, and perform system actions that guarantee the access and management of the information. Figure 4.2 presents the ITLingo-Cloud domain model from the users' perspective, showing the different types of roles that the system supports, namely platform roles, organization-specific roles, and workspace-specific roles.

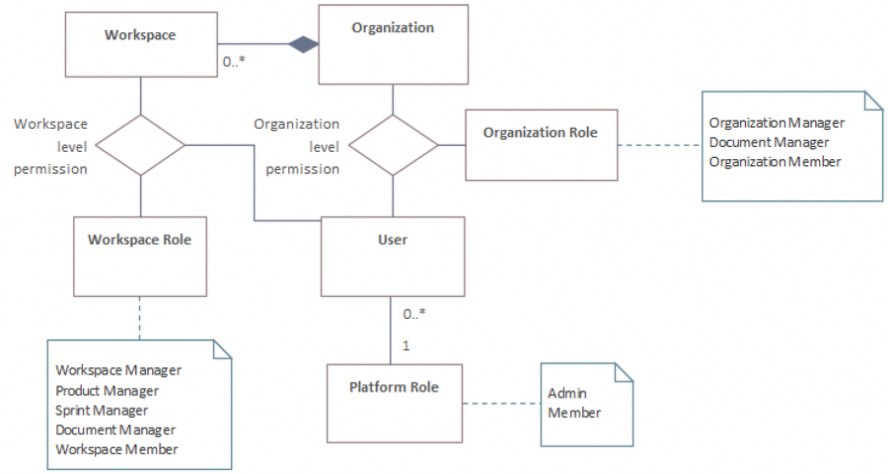


Figure 4.2 - Users and Roles domain model (UML class diagram)

User roles may be changed easily at the platform, organization, and workspace levels, and the users' information may be searched, filtered, and sorted. User roles also can be chosen when some invitation is sent to be part of one specific organization or workspace. Invitations and system notifications are explained in detail in Section 4.6.

Table 4.1 summarizes the user roles supported by ITLingo-Cloud at the platform, organization, and workspace levels.

Table 4.1 - User roles supported by the ITLingo-Cloud platform

Level	User Role	Description
Platform	Admin	Access and manage platform users and all the registered organizations on the platform. This role also allows the creation of new organizations.
	Member	Access common pages at the platform, create organizations and manage workspaces in its organization. This role allows receiving invites to be member of different organizations and workspaces.
Organization	Organization Manager	Complete control of the organization.
	Document Manager	Manage organization files (e.g., templates, libraries, documents). (This role also includes permissions granted to the Organization Member role).
	Organization Member	Consult the analytics page and the main page of the organization
Workspace	Workspace Manager	Complete control of the workspace.
	Product Manager	Enables the management of the product backlog. (This role includes permissions granted to the Workspace Member roles).
	Sprint Manager	Manage the Sprints or Kanban board, creating new tasks and assigning items depending on the project management process. (Moreover, this role also includes permissions granted to the Workspace Member role).
	Document Manager	Manage workspace files, synchronize, import data from the stored files. (This role also includes permissions granted to the Workspace Member role).
	Workspace Member	Consult the information about the workspace, such as its analytics page, workspace main page, stakeholders, or requirements, as well as generate reports with workspace data.

4.2. Multi-Organization Management

ITLingo-Cloud allows users to store, search and visualize several organizations' information.

To better structure teams and facilitate collaborative work is possible to create several workspaces inside each organization space, named as "organization". Each organization may store files, namely libraries and templates (e.g., PSL Excel Template, RSL Excel Template), that may be used and filled with projects' data at the workspace level. The system stores and processes projects' data to provide intuitive statistics about organizations, intending to present and track their performance, such as: to analyze and compare budgets between project costs, to

track projects to aid in business decisions. Figure 4.3 shows the ITLingo-Cloud domain model from the organization's perspective.

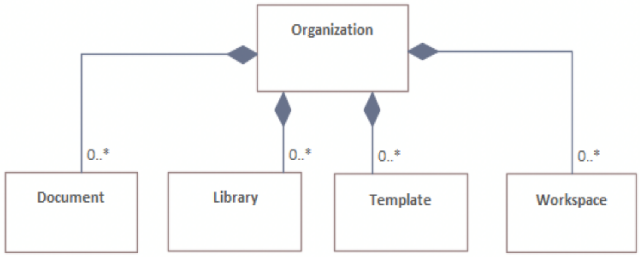


Figure 4.3 - Organization domain model (UML class diagram)

During the registration process, the user who registers in the system can create a new organization, thus becoming the organization manager of that organization. After the creation, the user has immediate access and control over it. A user can be added or invited to be a member of an organization that already exists on the platform. In this case, the organization manager of that organization needs to accept the access request to that organization. At any moment, a user with an organization manager role may access the list of the organization users from this organization, invite a new one or change user roles. Figure 4.4 summarizes the many use cases mentioned that could be made at an organizational level.

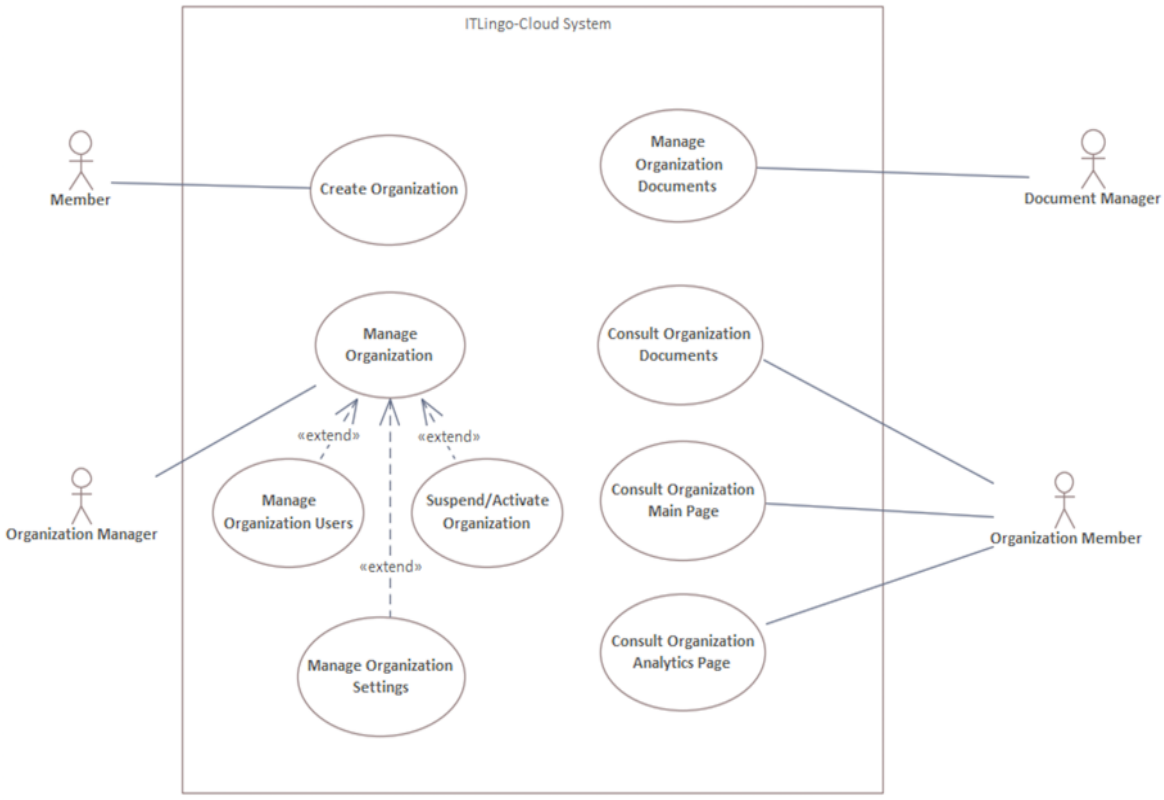


Figure 4.4 - Organization-Specific Use Cases Diagram (Simplified)

Each organization provides statistics to analyze its data, namely on the analytics page and on its main page. It is possible to edit information related to the organization's name, organization activity type, or country on the settings page.

4.3. Multi-Workspace Management

Workspaces are created in the scope of an organization. The main workspaces' properties are: project management process, status, project benefits, success criteria, general costs, and schedule information. These properties can be defined manually or automatically imported from files uploaded on the platform. Users can manage documents, the product backlog, and sprint backlogs. Figure 4.5 presents the workspace domain model.

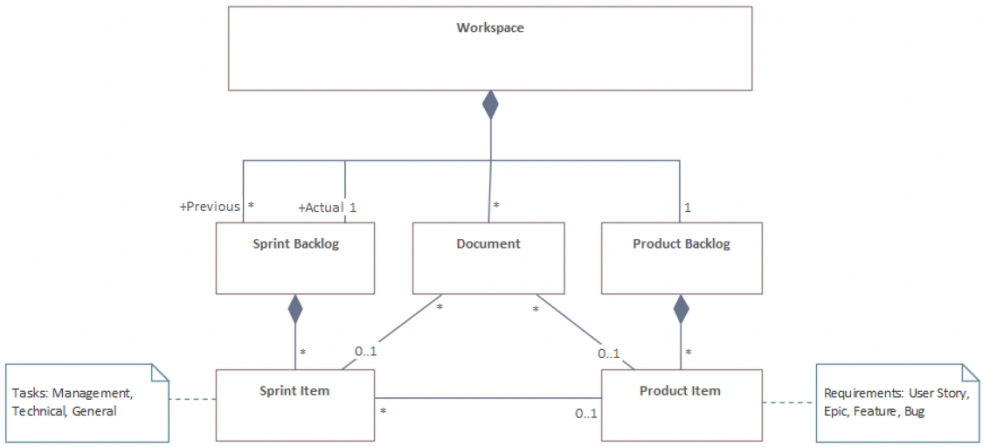


Figure 4.5 - Workspace domain model (UML class diagram)

A workspace is a helpful space to support collaborative project management and contains several pages with several features. The system shall allow the storage of files (e.g., PSL Excel files) filled with project data. This data can be automatically imported from these files and consequently saving time. The users may select or merge the relevant data stored in these Excel files (e.g., related to the product backlog, sprint backlog items, project costs) and choose what they want to import into the database. This feature is particularly useful to promote better collaboration between teams or stakeholders since they may share dispersed files. ITLingo-Cloud shall provide visual insights such as statistics and intuitive dashboards to analyze workspace data, that may be imported or not. Users may generate project management reports by using the data stored in the database in the scope of each workspace.

Figure 4.6 shows the use case model that summarizes the main use cases considered at the workspace level.

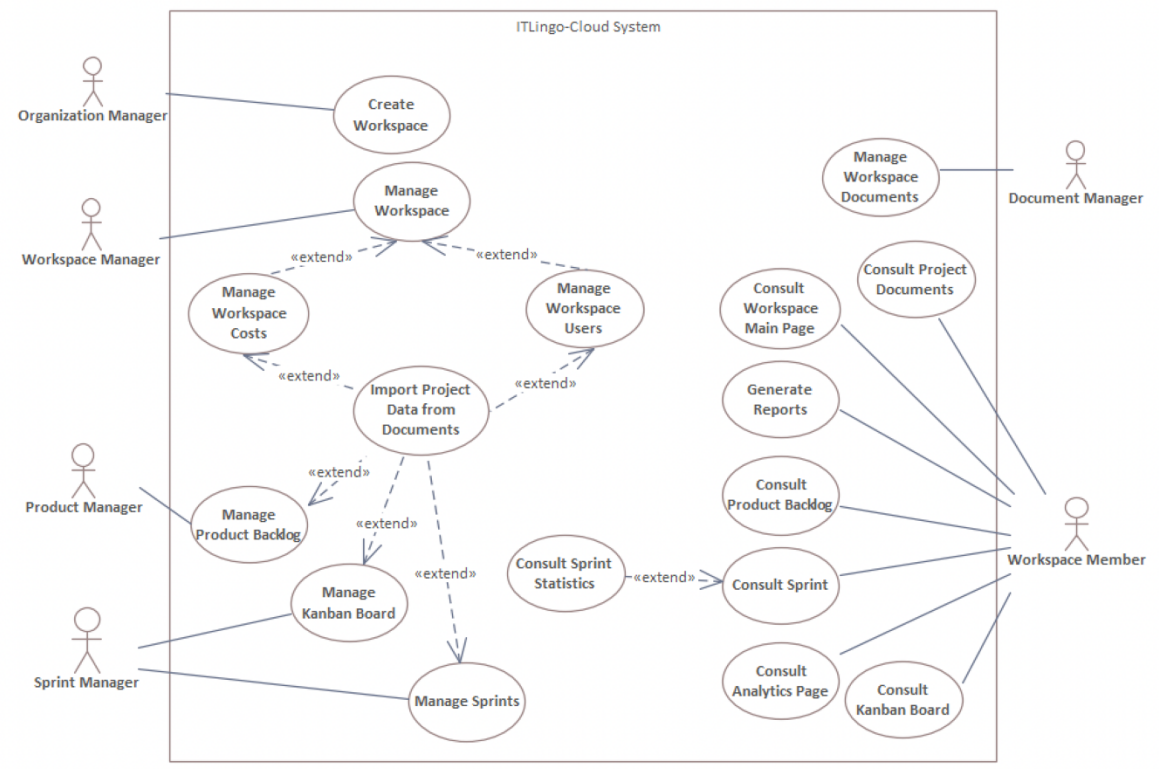


Figure 4.6 - Workspace-Specific Use Cases Diagram (Simplified)

4.4. Agile Processes Support

The product backlog, sprint backlog, and Kanban board can be managed according to agile processes, namely Scrum and Kanban [22]. It shall be allowed to store information associated with each product backlog item (or task), including files, images, effort, and other relevant information.

Scrum Support. If the chosen process in (the workspace settings) is Scrum, users shall have access to the Sprint Backlog and Sprint History pages.

The users shall manage sprints and respective tasks. The system shall allow to search the information, including sprint history and each sprint statistics page. If there is no initialized sprint, the system shall redirect to the sprint page creation. During the sprint creation, users shall specify relevant information, namely the sprint name, sprint schedule, and the stakeholders involved.

Sprint information may be imported from Excel files or filled manually. The loading of data from the PSL Excel files detects if the task id already exists in the database or not to create or update its information on the database (data importation aspects are explained in detail in the following chapters). If the tasks are created manually on the platform, it is necessary to provide all the information (e.g., name, type, priority, allocated people, task effort).

ITLingo-Cloud shall allow the creation of tasks by importing items from the product backlog. With this functionality, the backlog item information is used to create a new task with these data, assigning a unique task ID.

There are different types of effort defined for each task: (1) planned effort, (2) remaining effort, and (3) actual effort, which may be filled daily on the platform or imported from excel files.

The planned effort is the number of hours expected to conclude the task and may be registered during the task creation and edited after, if necessary. The remaining effort is the number of hours expected to complete the task and may be reported daily during each workday. Actual effort is the number of hours spent working on one specific task and is defined daily during each workday. All this information may be imported directly from the excel files, allowing dispersed data to be merged.

ITLingo-Cloud shall provide a sprint board, which displays the tasks organized by status (Not Stated, Not Completed, and Completed). With Sprint Board users can drag and drop and filter tasks. Each concluded sprint is stored in history and all the associated information, including planned and actual schedule, tasks performed, stakeholders involved, effort, and associated images or files. The system shall display dashboards summarizing each sprint's information.

Kanban Support. If the preferred software development process chosen is Kanban, users can choose this process in workspace settings. In this case, the Sprint Backlog page and Sprint History page are replaced by the Kanban page, which only displays the Kanban board with filters and the functionality to create new tasks. Kanban data also can be imported by Excel files, automatically generating the charts for these data. (These concepts are explained better in the following chapters.)

4.5. Files Management

ITLingo-Cloud shall support the storage of multiple files, promoting the reusability and integration with ITLingo templates (e.g., PSL Excel Template, RSL Excel Template).

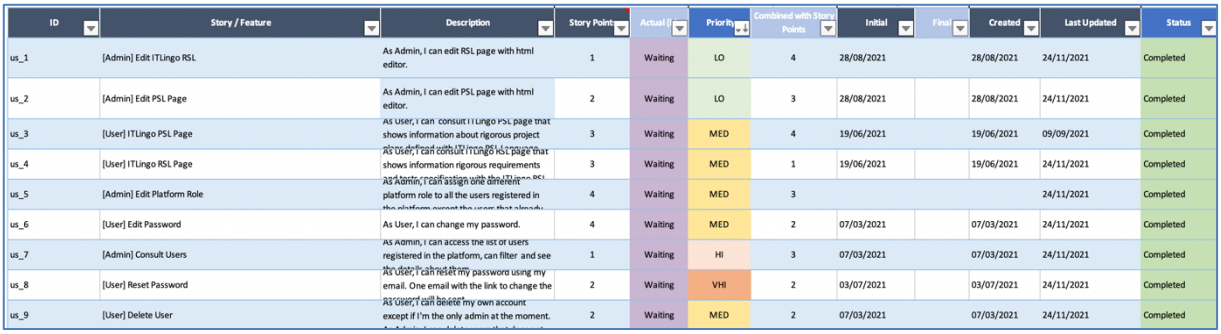
ITLingo libraries contain reusable specifications (e.g., defined in PSL or RSL languages), which can be used to create new specific specifications. ITLingo-Cloud shall support reusing these templates, importing data, and saving time by using these data in the scope of each organization and respective workspaces.

ITLingo-Cloud shall ensure that users can access the same documents' versions that stakeholders may share. ITLingo-Cloud shall allow selecting and merging data from these Excel files, allowing to select the content that the user may need (e.g., stakeholders, use cases, open issues, product backlog items).

4.5.1. Data Import from PSL Excel Files

The stakeholders involved in each software project constantly generate data in organizations, including information related to requirements, bugs, budget, risks, and effort. When multiple stakeholders work simultaneously, there are problems with versions, data consistency, outdated information, and erroneous data. ITLingo-Cloud system shall allow import specific data from Excel files (namely based on PSL Excel templates) to speed processes, adding new information or updating existing data stored on the database. Furthermore, it shall be possible to partially select specific information from these Excel files and synchronize it with the information in the database.

Figure 4.7 presents an example of a Product Backlog worksheet from a PSL Excel template with project data that could be imported to the platform.



ID	Story / Feature	Description	Story Point	Actual	Priority	Combined with Story Points	Initial	Final	Created	Last Updated	Status
us_1	[Admin] Edit ITLingo RSL	As Admin, I can edit RSL page with html editor.	1	Waiting	LO	4	28/08/2021		28/08/2021	24/11/2021	Completed
us_2	[Admin] Edit PSL Page	As Admin, I can edit PSL page with html editor.	2	Waiting	LO	3	28/08/2021		28/08/2021	24/11/2021	Completed
us_3	[User] ITLingo PSL Page	As User, I can consult ITLingo PSL page that shows information about rigorous project	3	Waiting	MED	4	19/06/2021		19/06/2021	09/09/2021	Completed
us_4	[User] ITLingo RSL Page	As User, I can consult ITLingo RSL page that shows information rigorous requirements	3	Waiting	MED	1	19/06/2021		19/06/2021	24/11/2021	Completed
us_5	[Admin] Edit Platform Role	As Admin, I can assign one diferent	4	Waiting	MED	3				24/11/2021	Completed
us_6	[User] Edit Password	As User, I can change my password.	4	Waiting	MED	2	07/03/2021		07/03/2021	24/11/2021	Completed
us_7	[Admin] Consult Users	As Admin, I can access the list of users registered in the platform, can filter and see	1	Waiting	HI	3	07/03/2021		07/03/2021	24/11/2021	Completed
us_8	[User] Reset Password	As User, I can reset my password using my email. One email with the link to change the	2	Waiting	VHI	2	03/07/2021		03/07/2021	24/11/2021	Completed
us_9	[User] Delete User	As User, I can delete my own account except if I'm the only admin at the moment.	2	Waiting	MED	2	07/03/2021		07/03/2021	24/11/2021	Completed

Figure 4.7 - PSL Excel File Filled With Project Data

ITLingo-Cloud shall avoid duplicate information, detecting duplicated IDs of data elements stored in these Excel. Hence, the ITLingo-Cloud shall update the information on the database if the ID already exists in the specific workspace, not duplicating data, otherwise will create the data on the database.

It shall be possible to select a specific PSL Excel worksheet(s) to be imported, namely: (1) all the project data (several worksheets) or only, (2) product backlog items, (3) sprint backlog or kanban tasks, and (4) project stakeholders.

Openpyxl library [31] shall be used to import the data, as mentioned in Section 2.4.1.

4.5.2. Document Automation Support

Automatic report generation is an emerging technology that generates documents or reports containing text, tables, and figures about a specific topic [3]. ITLingo-Cloud shall provide a report automation system that shall use the data stored in the database to generate different reports. This feature is handy and saves time once the production of these technical artifacts can be a very monotonous and repetitive activity. Standard project management reports must follow certain practices or contain certain types of information. Some of them are identified in PMBOK [14].

Domingos Bragança carried out a study to understand what are the common reports in project management [48]. During his research, he designed the functionality to generate automatic reports in the PSL Excel template. The document templates collected and produced by Domingos Bragança were adapted to be used in ITLingo-Cloud.

The ITLingo-Cloud shall adopt the pre-defined templates stored on the platform and replace specific tags by organization and workspace data stored on the database. (To do this we shall use a python library called Python-docx [32]).

4.6. Notification and Alerts Support

ITLingo-Cloud shall include a notification system to share relevant information and events that happen in the scope of organizations and projects. That notification system shall enable real-time notifications allowing directly read messages from the notification bar. Important information shall be automatically sent, namely invites to members of organizations or of specific workspaces. Also, accepting or rejecting the invite directly on the notification message shall be possible. Besides that, the system shall inform when one invite is accepted or rejected

and when the project's planned end date is approaching. Invites to be member of organizations or workspaces are made even if a person does not have yet an account on the platform. In these cases, an email is sent to invite the user to register in the system. Depending on the invitation type, the invite to be member of a specific organization or workspace is made at the organization or workspace level. Supposing the user's email is already registered on the system, one invitation notification shall be sent, and it shall be possible to accept or reject the invite directly from the notification message.

4.7. Data Analysis

ITLingo-Cloud shall support the analysis of project-related data by summarizing and centralizing information. ITLingo-Cloud shall help to make more informed correct decisions with less uncertainty. The user shall explore the data through interaction with charts and obtain a summary of the characteristics and statistics of the organizations' and projects' data. ITLingo-Cloud supports dashboards at the organization and workspace levels to track progress, working hours, outcomes, and other relevant information.

4.7.1. Organization-Level Data Analysis

ITLingo-Cloud shall include (i) an organization main page that summarizes the organization's data and (ii) an organization analytics page that analyzes the data at a detailed level.

The data analysis mechanisms in the scope of each organization can be summarized as the following:

R-1. Compare organizational productivity changes over time, namely work completed over time.

R-2. Track completed work per workspace.

R-3. Analyze workspaces' status.

R-4. Compare users involved per workspace.

R-5. Analyze workspaces' costs, namely to better understand how to use the available budget, comparing planned cost and the current cost among workspaces.

4.7.2. Workspace-Level Data Analysis

ITLingo-Cloud shall include (i) a workspace main page to summarize data among users and (ii) a workspace analytics page to show data at a detailed level.

The data analysis mechanisms in the scope of each workspace can be summarized as the following:

R-1. Analyze the amount of the remaining work versus the time required to complete it in each sprint.

R-2. Visualize the actual effort per day progressively in each sprint.

R-3. Track the workspaces' productivity over time, illustrating the number of tasks in each sprint, and the actual number of work hours.

R-4. Analyze product backlog items' status and types.

R-5. Analyze the number of people involved in each sprint and individual performance.

R-6. Analyze sprints' historical data, namely each sprint's open issues, the progression, the type of items solved, and tasks' remaining and actual effort per day.

4.8. Quality Requirements

ITC shall consider some quality requirements, namely, to provide a secure, intuitive, and user-friendly system.

QR-1. Usability. The system must offer a user-friendly, easy-to-learn, and intuitive interface for all users with search filters.

QR-2. Privacy. The system must protect the privacy of users and organizations with appropriate mechanisms to safeguard sensitive data (i.e., users' personal information, organization geographic data, workspace financial data).

QR-3. Security. The system must provide access control, taking all reasonable steps to prevent unauthorized parties from accessing confidential information. Furthermore, the system must use encryption to protect stored personal and confidential data (i.e., passwords).

4.9. Other Requirements

We define other requirements to be considered in the implementation of the ITC platform, namely the following general goals:

Goal 1. Integration with ITOI. The system must provide integration with the ITOI system, an online integrated development environment (IDE), allowing to open and edit ITLingo-Cloud workspace artifacts stored in the database. Furthermore, this integration will enable software development and deployment in each workspace's scope. A user logged in ITLingo-Cloud will access and modify files directly with the ITOI system.

Goal 2. Multi-Language Support. The system shall support the storage of multiple libraries and templates, supporting ITLingo specification languages such as PSL or RSL.

Goal 3. ITLingo PSL Excel Integration. ITLingo-Cloud shall integrate with the ITLingo PSL Excel template, allowing to import PSL Excel files (see Section 4.5.1).

5. ITLingo-Cloud Design Aspects

This chapter describes the architectural and design aspects of the ITLingo-Cloud system. Section 5.1 introduces the platform architecture, database architecture, and the integration with ITOI.

Section 5.2 describe the design aspects of the main modules developed in the ITLingo-Cloud platform: users, organizations, workspaces, agile processes, notifications and project alerts, files management and data import, document automation, and data analytics features.

5.1. Architecture

The ITLingo-Cloud system is implemented following a modular, iterative, and incremental approach. As illustrated in figure 5.1, ITLingo-Cloud has a client-server-style architecture with three main elements: (i) a front end available to the client through a web browser, (ii) a back-end server, and (iii) a relational database.

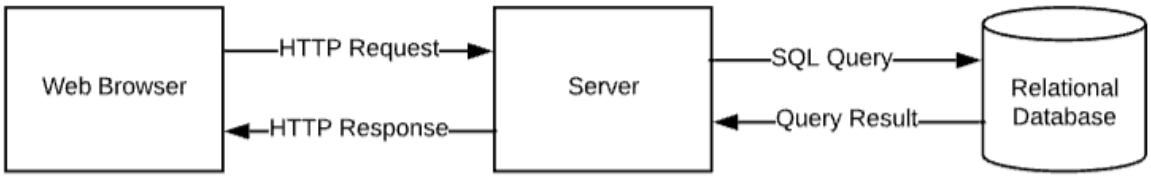


Figure 5.1 – High-level Architecture of the ITLingo-Cloud system.

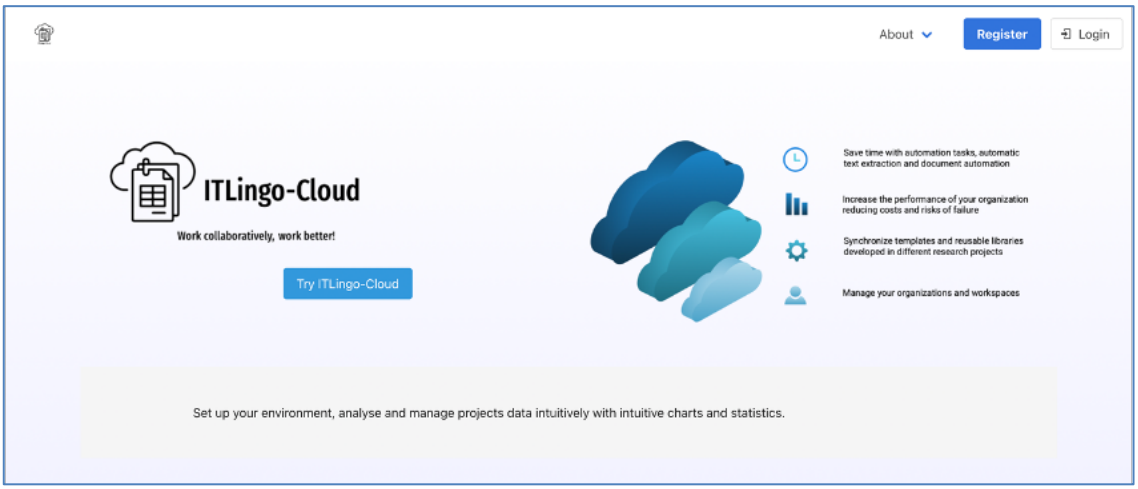


Figure 5.2 - Screenshot of the ITLingo-Cloud main page.

As discussed in Section 2, the ITLingo-Cloud architecture encloses a relational database that stores these data and uses a popular Django framework [49]. Users interact with the system through a web browser. The browser sends the HTTP request to the web server. The server translates the queries to SQL and sends them to the Database Management System (DBMS).

The resulting query is returned to the Django application [49]; finally, the server returns an HTML page. In the front-end part we used Chart.js to develop the dashboards. The web pages are responsive, so the ITC can also be used on resized or smaller screens such as mobile devices (i.e., tablets or mobile phones). Figure 5.2 presents the ITLingo-Cloud main page.

5.1.1. Database Architecture

The ITLingo-Cloud architecture encloses a relational database, in particular, implemented in the PostgreSQL database management system. Some concepts were already presented in Chapter 4, however, it is possible to observe the architecture of the database (simplified view) without attributes in Figure 5.3.

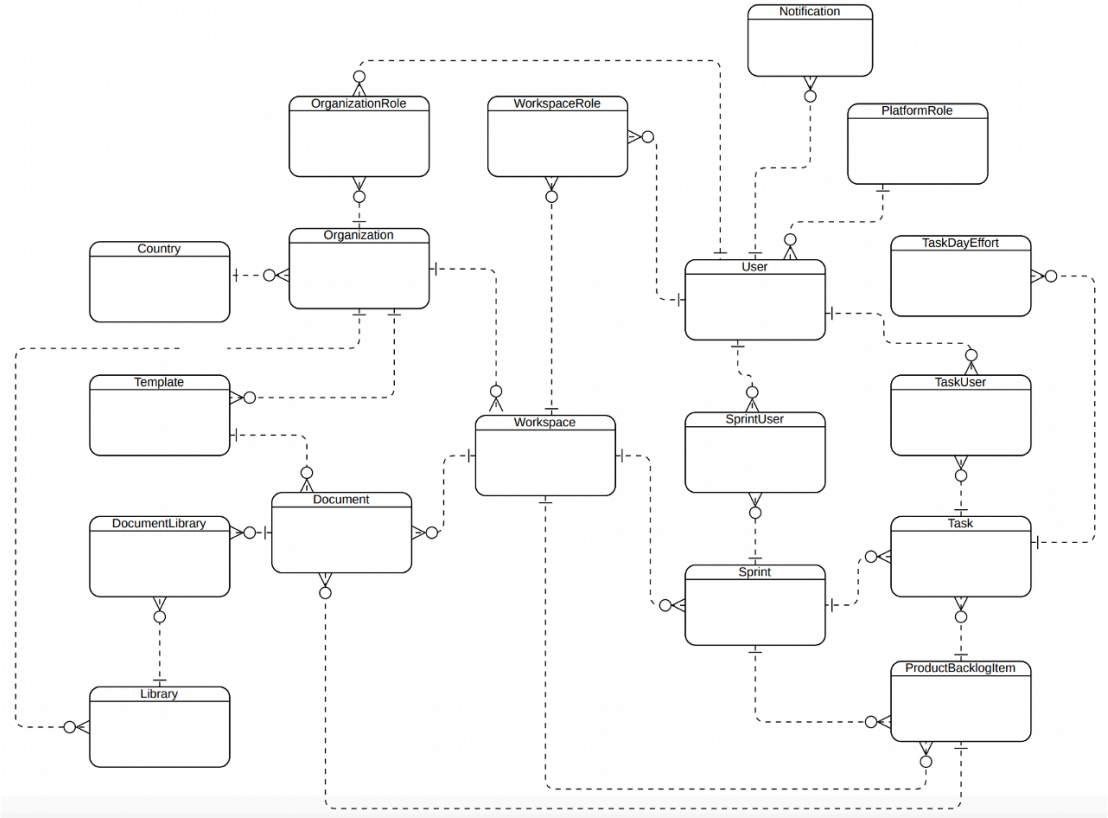


Figure 5.3 – Simplified Database Architecture of ITLingo-Cloud (Entity-Relationship Notation)

5.1.2. Integration with ITOI

Cloud IDEs are integrated development environments based in the cloud, crucial to software development nowadays.

Most software developers are familiarized with integrated development environments like Visual Studio Code (VScode) [50], a popular source-code editor produced by Microsoft. ITOI shall integrate typical features included in VScode, namely the workspace explorer, syntax highlighting and error highlighting, intelligent code completion, and code refactoring [50]. Its user interface shall be similar to VScode to be familiar to the end users.

Ricardo Silva has developed the ITOI platform that is integrated with the ITLingo-Cloud platform [51]. ITOI is an online integrated development environment (IDE) inspired by

emergent technologies like Google Cloud Shell Editor, Aws Cloud9, or GitHub Codespaces [44], [46], [47], as mentioned in Section 3.2.

This integration allows to create and edit of the workspace artifacts stored in the ITLingo-Cloud database through a web browser. Furthermore, this integration will enable the development and deployment of software more efficiently and collaboratively. These functionalities will facilitate organizations in their software development process, integrating several important functionalities in one unique platform. A user logged in ITLingo-Cloud will access to the ITOI system to directly edit such artifacts (e.g., RSL and ASL specification files). In this way, ITOI will update the database with any file changes, as described in Figure 5.4.

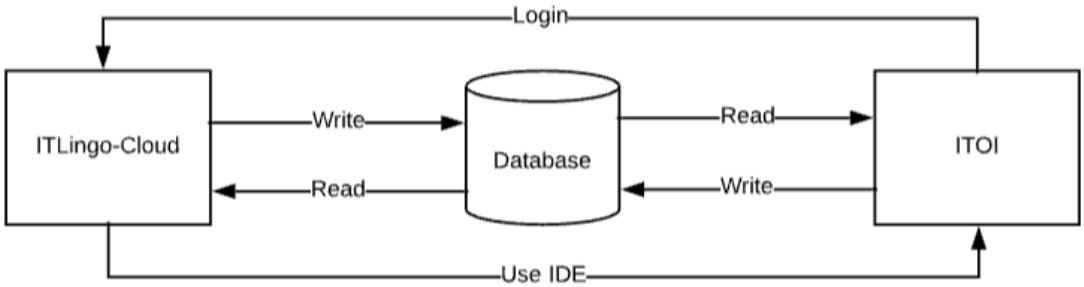


Figure 5.4 - ITOI and ITLingo-Cloud Integration

The ITOI system may be accessed in the scope of a workspace by selecting the “Specifications” option, as shown in Figure 5.7.

5.2. ITLingo-Cloud Modules

ITLingo-Cloud system is designed and implemented using a modular approach. This section describes the design aspects of its main modules, namely: (1) users, (2) organizations, (3) workspaces, (4) agile processes, (5) notifications, (6) files management and data import, (7) document automation, and (8) data analytics.

5.2.1. Users

The Users module is responsible for user registration, authentication, resetting passwords, changing passwords, changing account data, deleting accounts, or listing users registered on the platform.

Users need to log in to access the platform. A user logged into the system may change her password: the user must enter the old password, the new password, and its confirmation. If any of these passwords are not correctly entered, an error message is displayed. An error message is also displayed if the passwords have less than six characters or are too similar to the username.

If a user forgets the password, the system allows resetting it and regaining access to the platform: The user can enter her email; If the email is valid and associated with a user, an

automatic email is sent to that email. Otherwise, an error message will be displayed. Upon receiving the email, the user can click on the link and thus access the password reset page.

To be able to access the platform, it is necessary to register. During the registration or sign-in process, the user can create a new organization, thus becoming the organization manager of her organization. The user can also request to join an existing organization during this process. This request needs to be accepted by the organization manager of that organization.

A user can delete his account. In this functionality, warnings are shown, and the user needs to confirm her intent. There are restrictions on deleting accounts, namely when the user is the unique administrator registered on the platform.

A user can change her account data, namely her first name, last name, username, and email. The email and username attributes have to be unique on the database.

Users with the admin role can consult the table with all registered users on the platform. ITLingo-Cloud allows filtering and sorting by any user's attribute (i.e., name, username, email, role). A user with an admin role can also grant other users' permission to the admin. Additionally, system admins can remove registered users on the platform (except other users with admin roles) and assign specific organizations to them.

5.2.2. Organizations

The Organizations module is responsible for organization registration, editing organizations' data, deleting organizations, or listing organizations registered on the platform and its users.

An organization's main page provides statistics about the organization and its workspaces, as observed in Figure 5.5.

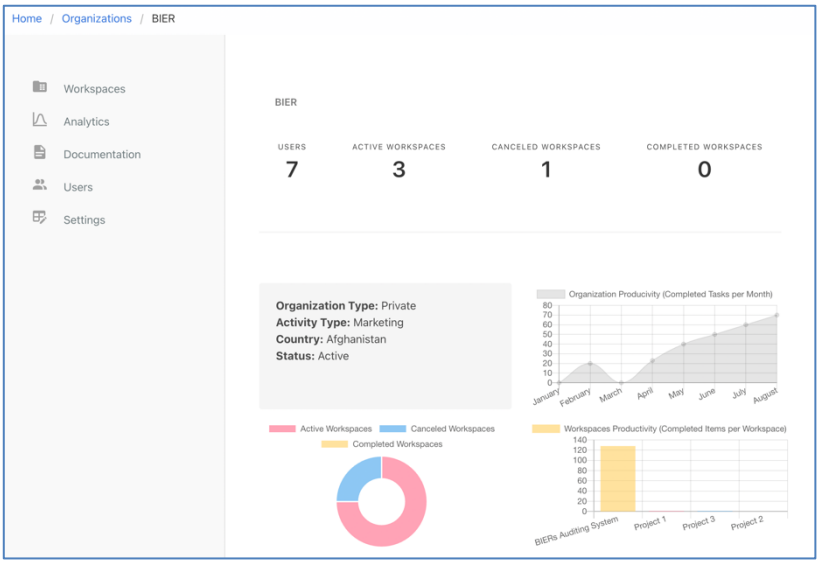


Figure 5.5 - Screenshot of the organization's main page

On the option settings, it is possible to edit data related to the organization, namely its name, privacy, activity type (i.e., sales, marketing, design, product, engineering, etc.), or country.

The user who registers in the system can create a new organization, thus becoming the organization manager of that organization. A user can also be added or invited to be a member of an organization that already exists on the platform.

Users with the admin role can also create organizations. Users with this role can access the table with all organizations registered on the database and filter, sort, and search this data by organization attributes. In addition, each user can also access the table with the list of their own organizations. This list includes all the organizations the user is associated with. The organizations' table presents the list of organizations ordered, by default, by creation date. However, organizations' data can be sorted by any attribute. Figure 5.6 shows an example of a table and filters provided to list and search organizations' data with the links to visit each organization page.

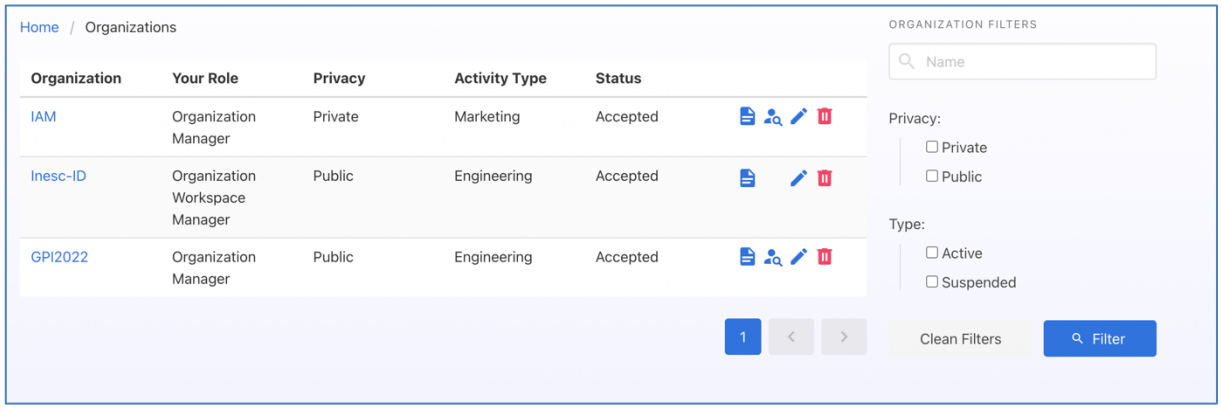


Figure 5.6- Screenshot of the organizations' data page

Users with an organization manager role may access the list of the organization users from this organization, invite a new one or change user roles. All the users' data can also be sorted, filtered, or searched. Users accessing this table can change roles or remove users (except other organization managers). This page grants access to the functionality of inviting new people to the organization.

To invite a new user, the role to be assigned to the new user must be specified (i.e., organization manager, document manager, or organization member). Invitations can be made to people already registered on the platform or not. If the email of the person to be invited is already registered in the database, the system sends an automatic notification with the invitation. Otherwise, an email is sent with the invitation to register on the platform to access the organization.

Organization managers can delete their organizations, permanently deleting the data from the database. This functionality is available in its own organizations' table.

5.2.3. Workspaces

The Workspaces module is responsible for creating new workspaces, editing workspaces' data, deleting workspaces, or listing workspaces registered on the platform and its users.

A workspace allows tracking software project information and contains several pages with several functionalities to support software engineering and project management activities. Figure 5.7 presents the workspace main page.

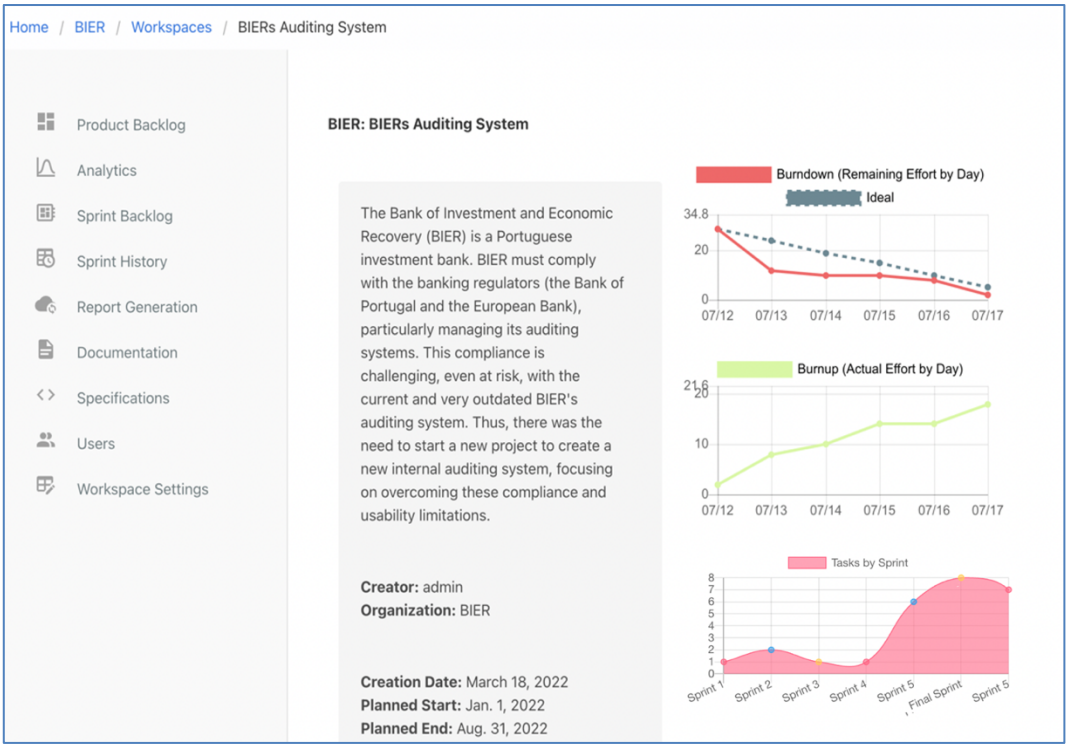


Figure 5.7 - Screenshot of the workspace's main page

The workspace-related data stored in the database (including data imported from PSL Excel template files) is used to provide visual insights such as statistics and intuitive dashboards to analyze up-to-date project data or used to generate documents on the report generation page, namely project management reports.

A workspace can be created on two pages: on the page that contains the workspaces of a specific organization (available on the organization page) or on the page that displays all the workspaces of the current user. In both cases, it is necessary to fill in some data related to this workspace for its creation. The workspaces list is available on two different pages: on a specific organization page or the workspaces page accessible on the navbar.

The workspaces to which the current user has access are listed on both pages. The list is ordered, by default, by creation date. However, the workspaces data can be filtered and sorted by several attributes. Figures 5.8 presents a list of workspaces stored on the platform.

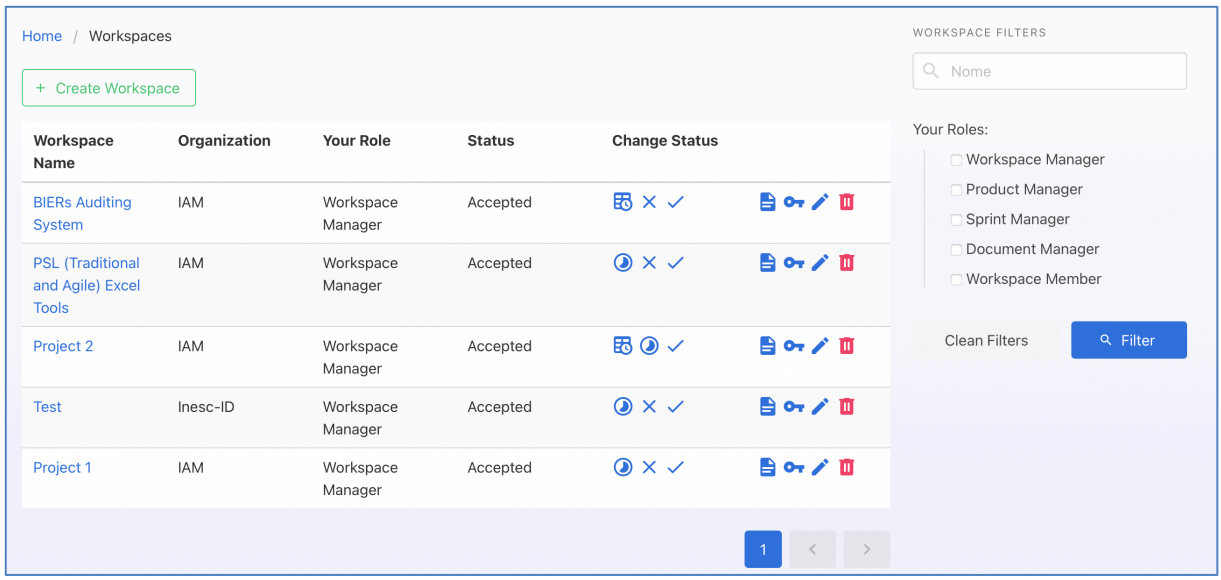


Figure 5.8 - Screenshot of the workspaces' data page

The tables of workspaces, depending on the workspace role of the current user, allow to change the status, consult documents, consult and change user permissions, edit data or delete the workspace. By deleting workspaces, the data on the database is permanently deleted. These features are also available on the main page of each workspace, as presented in Figure 5.8.

Depending on the current user's role, it is possible to access the table with the list of users associated with the respective workspace. This table presents some of the main attributes of each user (i.e., name, email, status, and role). If the logged-in user is the workspace manager, it is possible to change the workspace role or remove a user using this page. As in all pages containing tables, it is possible to filter, search and sort the data by attribute. Users can also invite or add more people to the current workspace on this page. The invitations can be made even if users are registered on the platform or not. In both cases, the user role and the email must be specified. The system detects whether the email is associated with a user registered in the database or not. If the email does not correspond to registered users, an email is sent with the invitation to register on the platform. Otherwise, an automatic notification is sent with the invitation. Users can also be added by importing a list of emails and automatically send invitations from a PSL Excel file or other Excel file with a similar structure.

ITLingo-Cloud also allows to edit of workspace-related data on the settings page, namely: (1) general information (workspace name, description), (2) schedule (i.e., planned start, planned end, actual start, actual end), (3) costs (i.e., planned costs, current costs), or even other (4) general settings (i.e., workspace status, project management methodology).

5.2.4. Agile Processes

The Agile Processes module is responsible for the management of the product backlog, sprint backlogs, sprint history, and kanban board.

The product backlog page includes several functionalities: create, edit, or delete items and list, sort, and filter all the product backlog items associated with the current workspace.

To create new items, the user needs to specify the respective attributes (i.e., identifier, name, type, story point, priority, or description). Each product backlog item has its page where it is possible to attach files, images, etc. On each product backlog item page, the system allows editing product backlog items' data, adding or removing attachments, and presents all the associated sprint backlog items. The system uses cookies to keep the actual product backlog page whenever the user visits the page of a specific product backlog item so that the user returns to the previous product backlog page. Each Product backlog item has a unique identifier in the database, and the system does not allow duplicate identifiers. Each identifier may be used to detect duplicated data imported from PSL Excel files.

The sprint backlog page allows users to create, edit, or delete sprint items (or tasks), import items from the product backlog, and manage the effort for each sprint task. On this page also is possible to consult the sprint board and start, conclude, edit or delete the actual sprint. The system redirects to the sprint page creation when there is no current sprint. To create a sprint, users must specify the schedule, users involved, and name. The sprint attributes can be edited later. After the sprint creation, the sprint backlog can be managed, and the system redirects to the product backlog table, as presented in Figure 5.9.

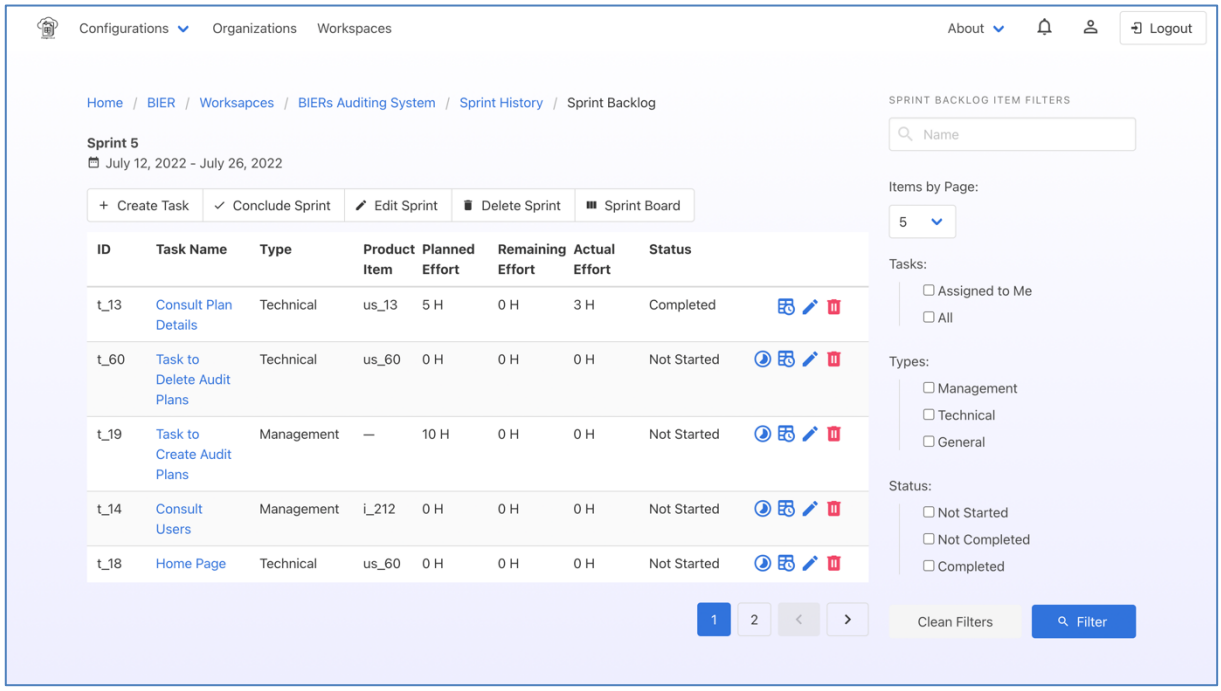


Figure 5.9 - Screenshot of the sprint backlog page

This page list all the sprint backlog items associated with the current sprint. In this table, the data can be sorted and filtered. The Sprint Backlog table has similar functionalities to the product backlog table. Tasks can be imported from the product backlog, from excel files, or created manually. To create a task, specify its name, type (technical, management, or general), priority, planned effort, and description. Like product backlog items, tasks have a unique identifier in the database that allows the detection of duplicated data imported from excel templates. Each task can be associated with specific product backlog items, and users can be assigned to perform particular tasks. Tasks' data can be edited later.

In the sprint backlog table (presented in Figure 5.9), it is possible to delete the task, change its status or consult the remaining effort and the current effort for each sprint day. The system allows daily recording of the remaining effort and the actual effort.

The sprint can be started, concluded, or deleted at any time. Supposing a user wants to finish the sprint but still has unfinished tasks the system displays a message explaining that the open tasks will be moved to the next sprint or kanban board, depending on the project management process chosen.

The sprint board was implemented mainly in javascript. In this board, all sprint tasks are grouped by their status (Not Started, Not Completed, Completed). The system also allows filtering tasks by their type (management, technical or general tasks). The user can drag and drop each task for other sprint board columns, updating the status attribute of the respective task in the database. Figure 5.10 presents the sprint board page.

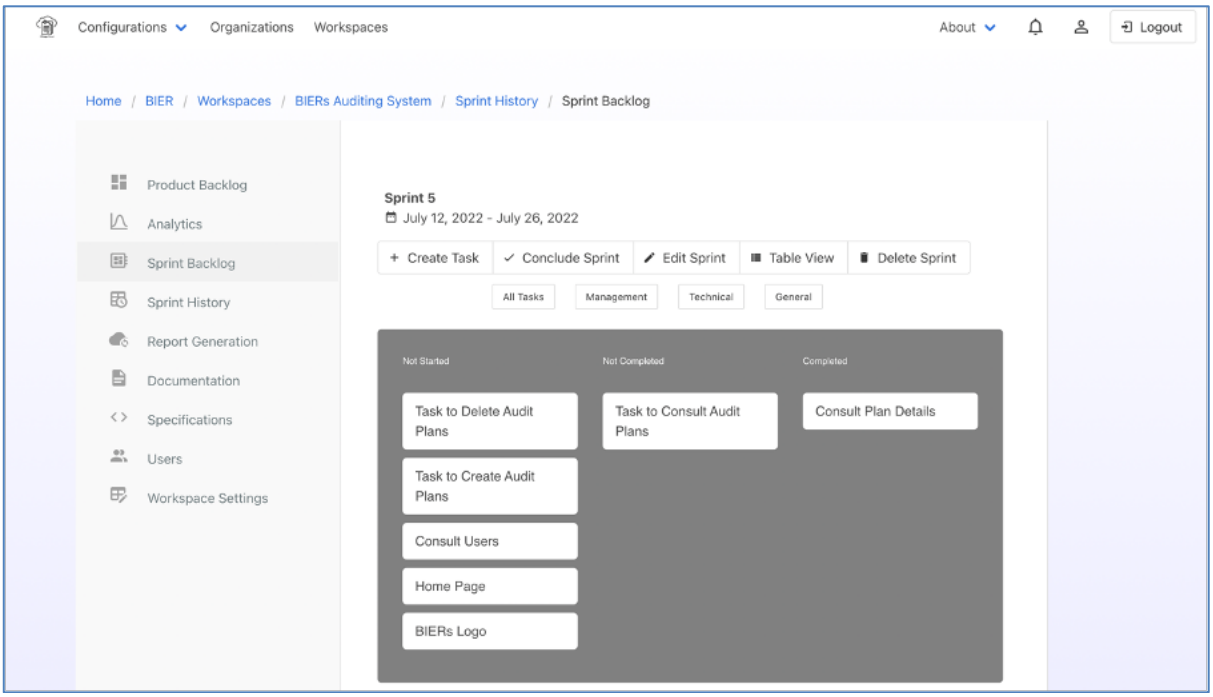


Figure 5.10 - Screenshot of the sprint board page

For the Kanban Board page, available when the project management process is Kanban, the system provides similar functionalities, including filters and allowing to update of the tasks' data. However, with the Kanban board, the tasks are not assigned to any sprint.

5.2.5. Notifications

The Notifications module is responsible for sending invitations, accepting or rejecting invitations, and sending alerts.

The system is designed to send automatic notifications in the following situations: (1) whenever an invite is made to a new member of a workspace, (2) whenever an invitation is made to a new member of an organization, (4) whenever an invited user accepts or rejects a workspace

invitation, in this case, the notification is sent to the person who invited, or (5) whenever an invited user accepts or rejects an organization invitation, in this case, the notification is sent to the person who sent the invite.

Figure 5.11 presents a notification example. As is possible to observe, the invitation can be accepted or rejected directly in the notification bar. In this case, the system sends a notification to the person who sent the invite informing them about the answer.

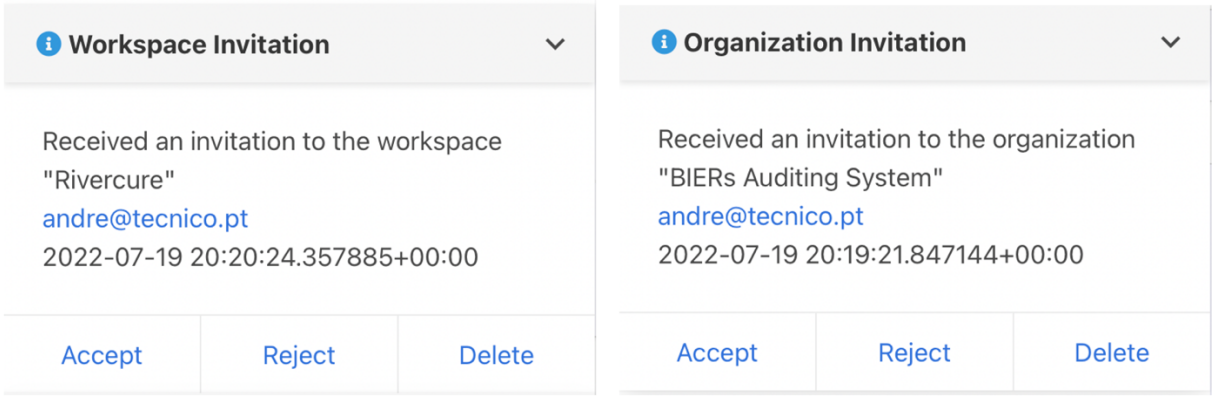


Figure 5.11 - Screenshot of the notification messages

Invitations are classified by the system as informative messages and therefore have an informative icon. Notifications about accepted or rejected invites have a success or unsuccess icon, respectively.

These notifications are received in real-time, so the user does not need to refresh the page, showing the user that there are unread notifications at any time.

Furthermore, some notifications are only sent conditionally, alerting for specific situations, namely when the planned date for the end of the project is close. In this case, the system sends a warning five days before the project's planned end date. This notification has an informational icon. These notifications alert the user to possible dangers of missing deadlines, functioning as reminders.

If a notification is not read, its color is slightly different from a read notification. In this case, the notification background is shaded, so the user detects that the message has not yet been read. The system allows to mark notifications as read. Additionally, users can delete them, erasing data from the database permanently.

5.2.6. Files Management and Data Import

The Files Management and Data Import module is responsible for the creation, storage, editing, deleting, and consulting files as well as importing data from PSL Excel files.

Figure 5.12 shows the ITLingo-Cloud documentation page, in which storing, downloading, sorting, and filtering stored documentation is possible.

A document can be created by uploading a specific file and specifying its type and name. ITLingo-Cloud allows editing file properties at the organization or workspace level, namely file name, file type, or uploading a new version.

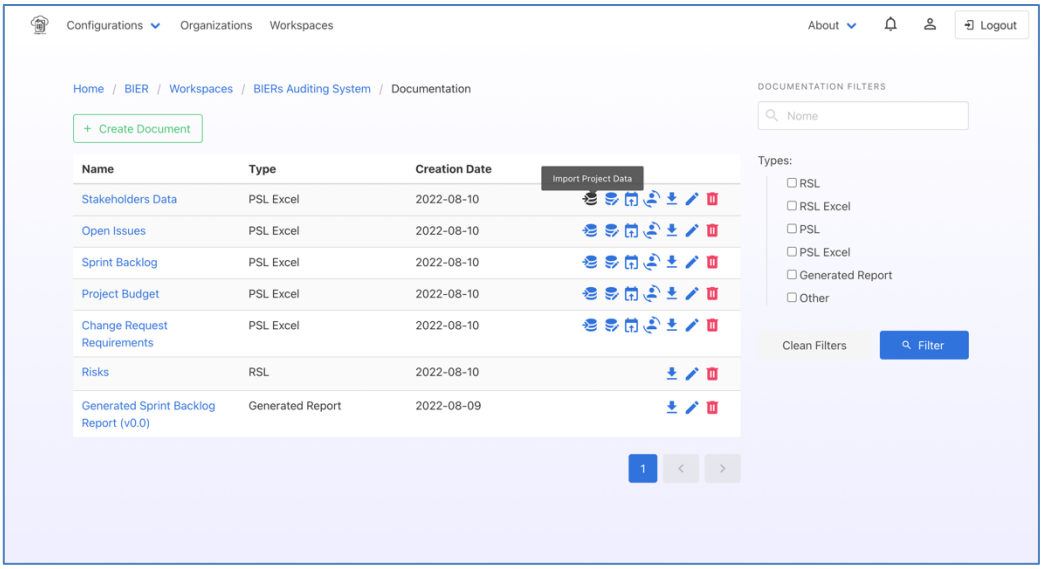


Figure 5.12 - Screenshot of the documentation page

The system allows users to select partial data from these Excel files, allowing them to select the specific worksheet that the user may need (e.g., stakeholders, use cases, open issues, product backlog items). As observed in Figure 5.12 user may select a specific PSL Excel worksheet(s) to be imported, namely: (1) all the project data (several worksheets) or only, (2) product backlog items, (3) sprint backlog or kanban tasks, and (4) project stakeholders.

The system detects duplicated data, comparing the IDs of data elements. The system detects if the ID already exists in the specific workspace to update or create the data on the database.

Import Product Backlog. To import the product backlog, the Excel worksheet called "ProductBacklog" is loaded using the openpyxl library [31]. To avoid problems or system exceptions that different PSL versions may cause, the algorithm starts by checking if the worksheet's loaded attributes correspond to the attributes expected to be received. This functionality is essential because it allows loading data from normal Excel files (with the same attributes of the worksheet to be imported on PSL excel) and allows to import of isolated PSL Excel sheets. To check if the attributes match as expected, the names are compared, and even if the columns are in a different order, it is possible to import that data. After checking the attributes, the excel sheet is scanned line by line, and each row of information is saved to the database. Then the algorithm checks if the ID of the expected object to be created exists in the database for this workspace. If the ID is already stored in the database associated with any data object of the current workspace, all attributes of that object are updated. Otherwise, it is created in the usual way in the database.

Import Sprint Backlog or Kanban Tasks. To import the sprint backlog or kanban tasks, the idea of the algorithm is the same as the previous but with some particularities. The system loads the excel worksheet called "SprintBacklog" or "Kanban" (depending on the software

development process chosen in the workspace settings). If Scrum is chosen, Sprint Backlog will be imported, otherwise will be imported the Kanban tasks.

Import Stakeholders. The algorithm is similar to the previous approaches to importing the workspace stakeholders. The system loads the excel worksheet called “StakeholdersPeople”. This algorithm only tries to find email attributes, avoiding possible inconsistencies between the people's data. For instance, the names of the stakeholders stored in the excel file may contain differences regarding the system users' data registered in the database. The algorithm checks if the emails correspond to users who already have an account on the platform but do not yet have access to the workspace. Only are sent invites to users registered on the platform and without access to the workspace.

Import Full Project. The system also allows importing all the new project data that were not imported since the last import. This functionality is helpful to speed up the software development process, saving time, for instance, to start a new project that does not yet exist in the platform and the data was only registered in an Excel file. This functionality is also handy, for example, to add new data that have been later filled in an Excel file but not on the platform. In this case, the system will call all the previous algorithms and import all the data, displaying a message that the project data is being imported.

5.2.7. Document Automation

The Document Automation module is responsible for generating word documents using the data stored in the database.

Project management reports must follow certain practices and include data as, for example, explained in PMBOK [14]. These reports shall be produced automatically. Figure 5.13 shows the report generation process, which includes: (1) selecting the report generation page, (2) choice of the report to generate, (3) redirecting to the documentation page, and (4) consulting the generated report.

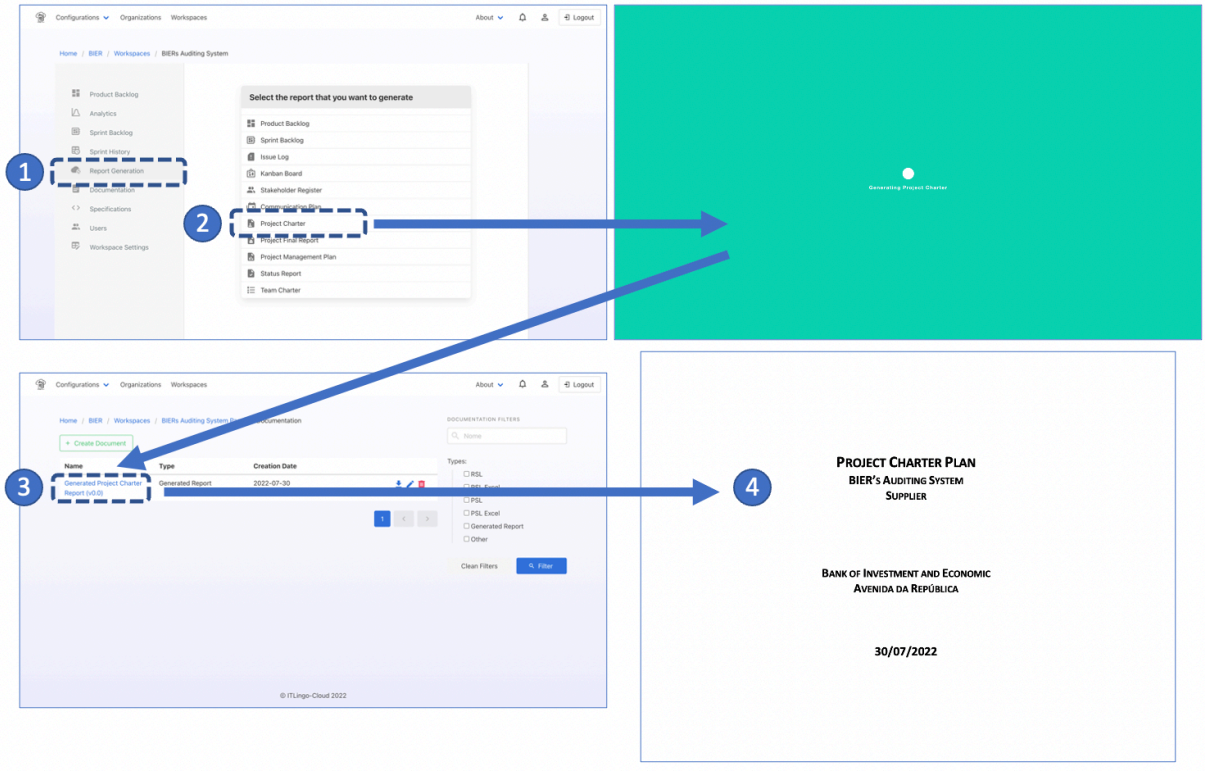


Figure 5.13 - Report generation process in ITLingo-Cloud platform

Although the automatic report generation functionality has not been fully completed nor tested in the validation section, the report generation algorithm has been implemented in ITC.

As defined in Domingos' research [48], each document template has special tags (defined as "{{ tag_name }}") indicating attributes that need to be replaced on the word documents with data stored in the database. Figure 5.14 shows a word template adapted from Domingos' research.

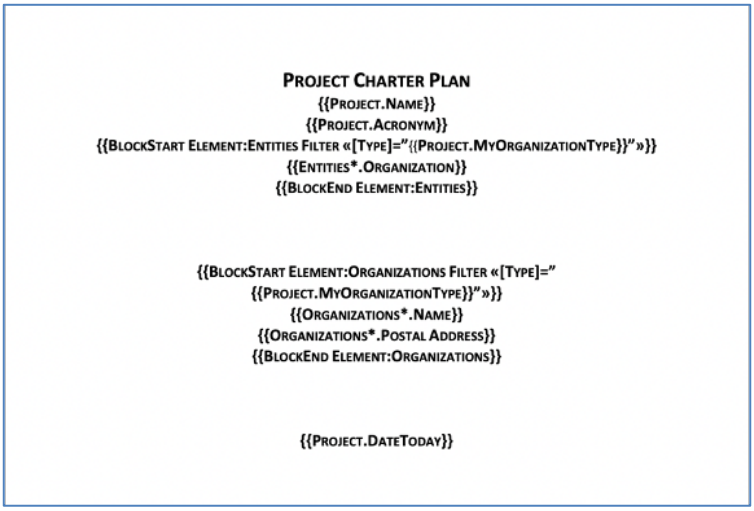


Figure 5.14 - Screenshot of a template used to generate project reports.

This algorithm starts by loading a report template. The Python-docx library is used to go through the file from beginning to end and replace all the tags found (with the following format: "*{{ tag_name }}*"), in which *tag_name* corresponds to specific information related to the current workspace or organization.

A new word file is created from the original word template but with the latest changes made by the algorithm and saved in the database, being associated with the workspace and organization in which the current user is working.

The generated report is available in the respective workspaces' documentation section, and the user is redirected to that page with a success message. After this process, ITLingo-Cloud allows downloading the generated report or sharing it with other users.

5.2.8. Data Analysis

The Data Analysis module is responsible for generating dashboards at the organization, workspace levels using the data stored in the database.

5.2.8.1. Organization-Level Data Analysis

Performance dashboards at the organization show information that enables understanding, monitoring, and tracking the organization's performance.

Main Page

The organization's main page summarizes the organization's data, as observed partially in Figure 5.5.

General Data. This page contains relevant information provided in a non-graphical way, namely the total number of users, status as well the number of workspaces/projects active, canceled, and completed.

Organization Productivity Chart. Line charts are a good option for analyzing the organization's productivity over time. This visualization helps analyze the evolution of organizational productivity over time, namely the number of completed tasks per month, as presented in Figure 5.15.

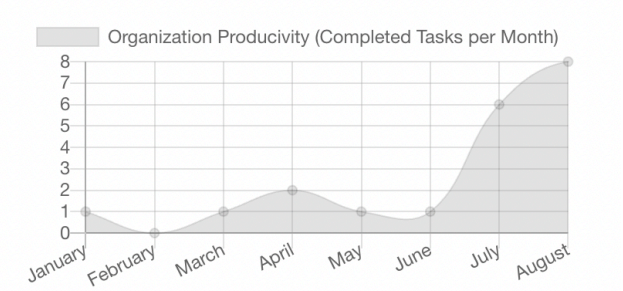


Figure 5.15 - Organization Productivity (Completed Tasks per Month)

Workspaces Productivity Chart. Bar charts allow comparing quantities of different categories, revealing highs and lows at a glance. Figure 5.16 shows a bar chart to understand and monitor the productivity of each workspace of the organization, namely the number of

product backlog items completed per workspace. (By moving the mouse over the data, the user can find the number of product backlog items completed in each bar).

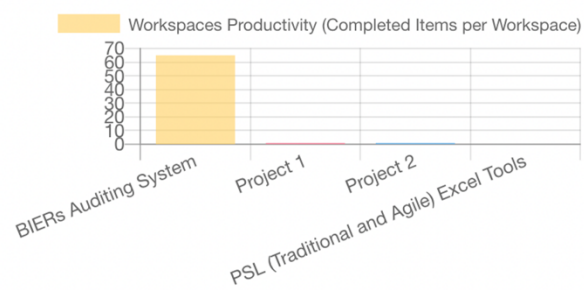


Figure 5.16 – Workspaces Productivity (Completed Items per Workspace)

Workspaces Status chart. Donut charts are similar to pie charts and are used to show the proportions of categorical data, with the size of each piece representing the proportion of each category. Due workspace status attribute is a nominal data type (also called categorical data). Pie charts or donut charts well depict these types of attributes. In this sense, we use a donut chart to illustrate the portion of active, canceled, or completed workspaces, as presented in Figure 5.17.

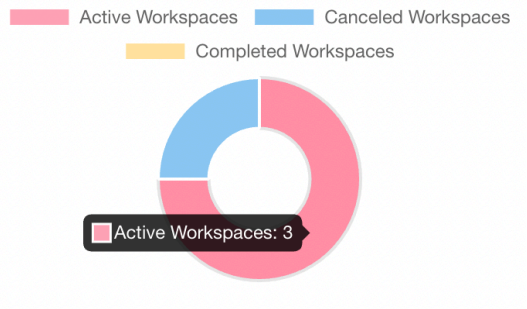


Figure 5.17 – Workspaces Status (Number of Organization's Workspaces per Status)

Analytics Page

The organization analytics page is designed to track all the organization's data. The organization analytics provides all the visualizations available on the organization's main page mentioned previously and two more charts to compare efforts and costs among different workspaces.

Workspaces' Users Overview chart. The system includes a bar chart to compare the number of users involved in each workspace of the current organization, as presented in Figure 5.18.

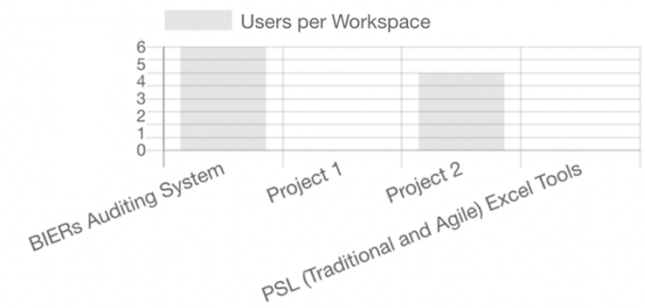


Figure 5.18 – Workspaces Users Overview (Number of Users per Workspace)

Costs Overview Chart. ITLingo-Cloud provides visualizations to track costs among workspaces, namely to compare planned costs with current costs. Line charts may be combined with bar charts to compare two different types of datasets: the planned cost and the current cost of each project from this organization. Figure 5.19 presents a chart to track the costs per workspace. (By moving the mouse over the data, the user can better compare the two values).

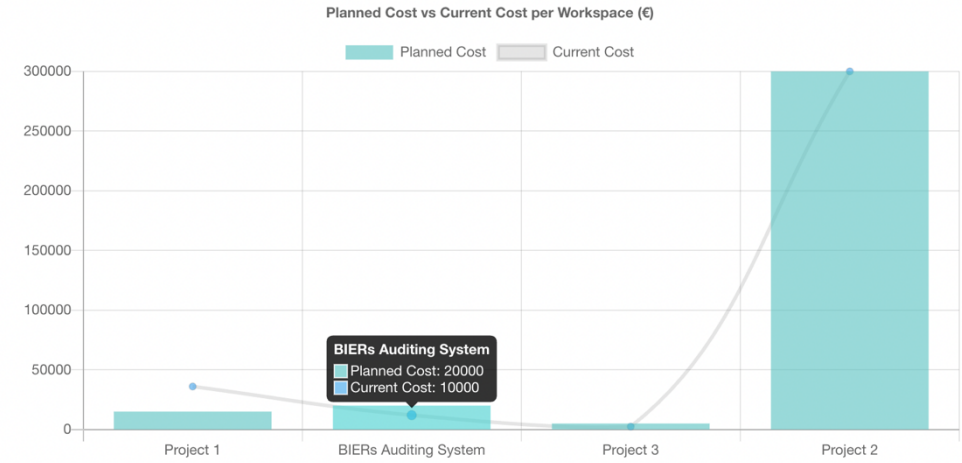


Figure 5.19 – Costs Overview (Planned Cost vs Current Cost per Workspace (€))

5.2.8.2. Workspace-Level Data Analysis

Dashboards and data analysis at the workspace level integrate several project information and allows to track of projects' performance.

Main Page

The workspace main page, presented partially in Figure 5.7, centralizes information among stakeholders.

General Data. The workspaces' main page contains information provided in a non-graphical way, namely about the workspace description, creator, schedule (creation date, planned start date, planned end date, actual start date, and actual end date), and costs (e.g., planned cost, current cost, etc.).

Burn-down Chart. To analyze the amount of remaining work versus the time required to complete it, ITLingo-Cloud provides a burn-down chart. This chart is used to predict the team's likelihood of achieving their work in time, displaying the remaining effort by day. Figure 5.20 presents the burn-down chart provided by the ITLingo-Cloud platform. (By moving the mouse over the data, the user can better compare the remaining effort of the current day and the value that would be ideal for that day.)

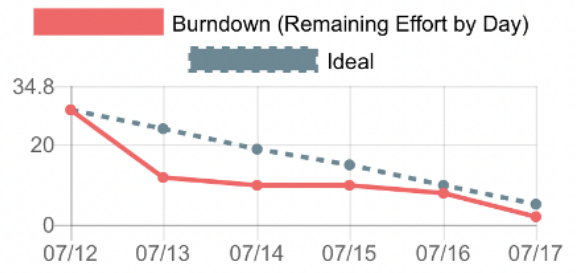


Figure 5.20 – Burn-down Chart (Remaining Effort by Day)

Burn-up Chart. ITLingo-Chart also provides a chart to analyze the project progress over time. To do it, we used a burn-up chart to represent the actual effort per day progressively, that is, the real effort spent on each day of the current sprint. Figure 5.21 illustrates the burn-up chart provided by the ITLingo-Cloud system.

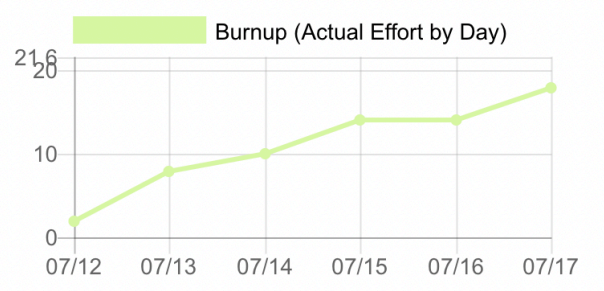


Figure 5.21 – Burn-up Chart (Actual Effort by Day)

Workspaces Productivity Chart. The system provides a line chart to analyze the workspaces' productivity, illustrating the number of tasks per sprint over time, as shown in Figure 5.22.

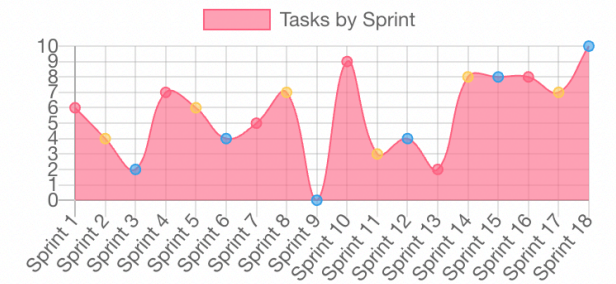


Figure 5.22 – Workspaces Productivity (Tasks by Sprint)

Items Status Chart. To show the percentage of backlog items not started, not completed, or completed, we chose to use a pie chart since these visualization techniques are suitable for comparing the parts of a whole, being a fast way to understand proportional data. Figure 5.23 shows the pie chart representing the status of the product backlog items globally at the current workspace level. (Hovering the mouse over each portion of the chart provides each type's concrete number of items.)

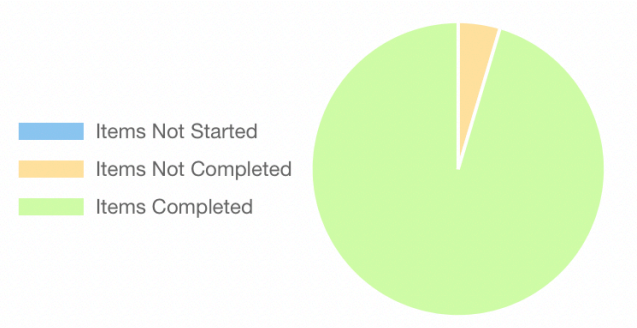


Figure 5.23 - Items Status (Number of Product Backlog Items per Status)

Items Types Chart. The system provides a bar chart to compare what types of items are included in the product backlog, as presented in Figure 5.24.

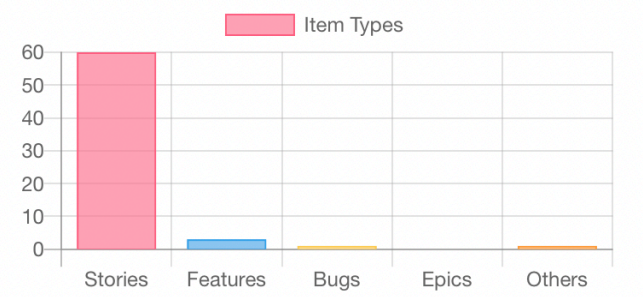


Figure 5.24 - Items Types (Number of Product Backlog Items per Type)

Analytics Page

The Analytics page at the workspace level summarizes workspace information and includes all the charts available on the workspace main page. Furthermore, it allows the analysis of the individual developers' performance, sprint data, and work hours data.

Users Performance Chart. Since several stakeholders may be involved in each workspace and its name is a nominal data type, a horizontal bar chart was built to display the intended datasets. Figure 5.25 illustrates a chart in which it is possible to analyze how many tasks users contributed to a workspace, supporting personal performance monitoring and improving productivity.

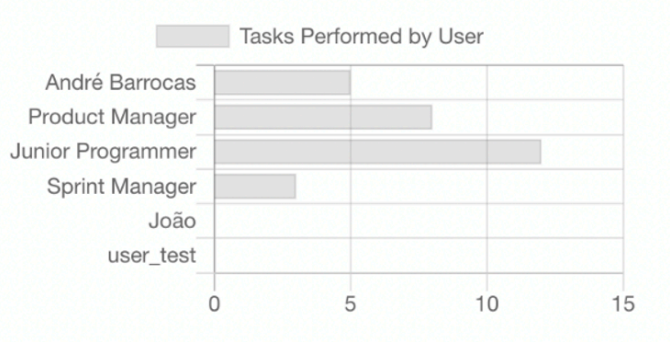


Figure 5.25 - Users Performance (Tasks Performed by User)

Workspace Work Hours Chart. The system includes a line chart to analyze the workspaces' productivity, illustrating the actual number of hours spent working per sprint, as presented in Figure 5.26. Since sprints follow a temporal order, we choose a line chart representing these datasets.

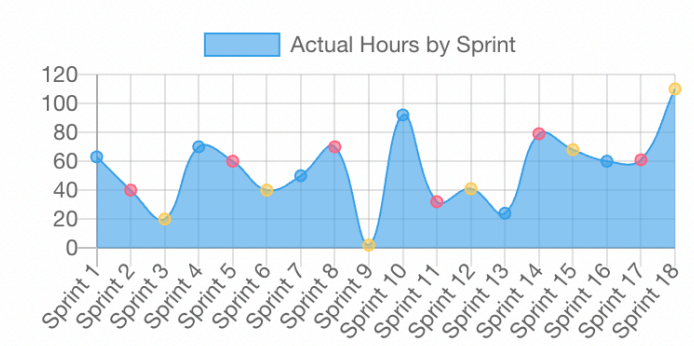


Figure 5.26 - Workspace Work Hours (Actual Hours by Sprint)

Sprints Users Chart. ITC provides a line chart to analyze the number of people involved in each sprint, as presented in Figure 5.27. With this visualization, it is possible to know which sprints involved more people and to detect potential relationships with the previous charts, namely with the productivity chart over time, for example.

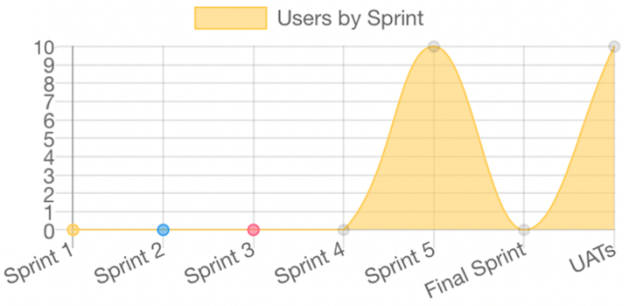


Figure 5.27 - Sprints Users (Users per Sprint)

Sprint Statistics Page

Agile teams must deal with much information to help them manage and reflect on their process, which is crucial to understanding and tracking their performance. To support these teams and all software developers in general, we provide a dashboard presented in Figure 5.28. to follow the sprints' historical data analysis and relevant information to visualize each sprint's open issues, the progression, type of items solved, and effort. To build this dashboard, the charts mentioned previously were used but related to a specific sprint and not the current one.

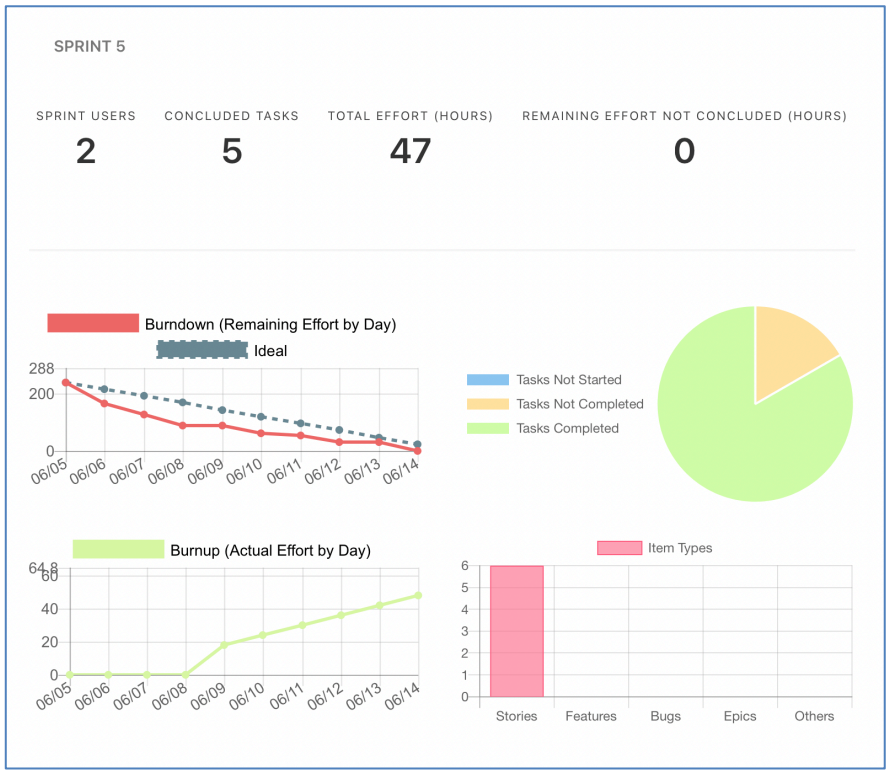


Figure 5.28 - Sprint History (Statistics Page)

6. Validation

This chapter discusses how we evaluated and validated the ITLingo-Cloud, mainly considering learnability, usability, efficiency, user feedback and satisfaction. This evaluation involves two experiments with different levels of complexity to test various functionalities of the ITC. Experiment A is based on development in the ITC in the scope of this research. This experiment tests platform features related to storing, importing, and analyzing data from excel files are tested. Experiment B is based on the BIER's audit system project. This experiment focuses on testing the collaborative features of the platform. Additionally, we evaluate the ITC platform through a user session assessment focused on Experiment B with other researchers and students, mainly to test collaborative and data analytics features of ITC.

6.1. Experiment A: The ITLingo-Cloud Project

In agile projects, stakeholders generate and share many artifacts with relevant information (product backlog, open issues, business information, etc.). Many of these documents can be used for subsequent projects since one principle of agile processes is to reuse work to accelerate software development processes. In this sense, ITC is very useful for managing the generated and stored data and existing Excel templates.

The first experiment for evaluating the ITLingo-Cloud platform is based on the development process of our solution according to agile principles, namely for accelerating the development process and reusing artifacts. Since it is a research project, requirements were often changed, so its development followed the Scrum framework with sprints with two weeks of duration. The requirements for this system were refined and updated several times, as well as some of the objectives initially planned in this project. Use cases, tasks, bugs, and system features were tracked using excel templates, particularly the most updated PSL Excel template previously mentioned.

In this experiment, we used real data to specify ITLingo-Cloud project requirements, use cases, project goals, project schedule, and stakeholders involved. Figure 6.1 shows an outdated version of a spreadsheet of the PSL Excel template used to manage one specific sprint.

Sprint Number	2
Start date	05/02/2022
Duration (d)	10
End Date	17/02/2022
Number of Developers	1
Dev Utilization	100%
Daily Work hours	8
Dev total hours	80
Dev Daily hours	6
Today's Date	21/08/2022
Status	Open

Write here the remainder Effort per task each day(it can be higher).
"Waiting" means we haven't reached the day yet.

Total Row is reflect as "Actual" in the chart so using the "sum" operation reflects the best results.

ID	Task	Description	User Story	Status	Assigned to	Planned Effort (h)	Effort Remaining each day 1	2	3	4	5	6	7	8	9	10	Actual Effort
ta_1	[Admin] Edit ITLingo RSL	As Admin, I can edit RSL page	us_1	Completed	André	10	0	0	0	0	0	0	0	0	0	0	
ta_2	[Organization Member] Consult Organization Document	As organization Member, I can	us_2	Completed	André	20	10	10	10	10	10	10	10	10	10	10	50
ta_3	[Organization Template Manager] Create Organization Document	As Organization Template	us_8	Completed	André	10	8	8	1	1	1	0	0	0	0	0	10
ta_4	[Organization Manager] Consult Organization Permissions	As Organization Manager, I can	us_15	Completed	André	30	30	30	30	30	30	30	30	30	30	30	5
ta_5	[Organization manager] Edit Organization Permission	As Organization Manager, I can edit organization	us_42	Completed	André	4	4	4	4	4	4	4	4	4	4	4	2
ta_6	[User] Reject Invite to Organization	As User, I can receive one automatic notification with	us_50	Completed	André	8	8	8	8	8	8	8	8	8	8	8	3
ta_7	[User] Accept Invite to Organization	As User, I can receive one	us_51	Completed	André	16	16	16	16	16	16	16	16	16	16	16	4
ta_8	[Organization Manager] Remove Organization Role	As Organization Manager, I can remove users	us_58	Completed	André	30	30	30	30	30	30	30	30	30	30	30	2
ta_9	[Organization Manager] Suspend Organization	As Organization Manager, I can suspend my	us_12	Not Completed	André	10	10	10	10	10	10	10	10	10	10	10	8
Total				Not Completed		138	116	116	109	109	109	108	108	108	108	108	84

Figure 6.1 - PSL Template Excel (Product Backlog)

During the development process of ITC, this platform also was used to test some features as they were being implemented, namely, to import and manage all the project information, including the artifacts (PSL Excel templates) filled with project data.

As shown in Figure 6.2, a workspace named "ITLingo-Cloud" was created and started (Tasks 1 and 2). Some dispersed PSL Excel files were uploaded to the platform (Task 3), and all the project data were imported as soon as the ITLingo-Cloud system was stabilized (Task 4). When importing the data, the system automatically creates the product backlog by registering each row of data in the database and associating it with the current workspace. In this process of importing all the project data, ITLingo-Cloud also stores the emails of the stakeholders in the database and automatically sends a notification with the invitation to the users who do not yet have access to the workspace. Finally, the system saves in the database the tasks being carried out in the current sprint, recorded in the file to be imported. The dashboards with the new data stored in the database can be consulted (Task 5). ITC also provides the data in the tables, which can be filtered, sorted, and searched intuitively and quickly.

During the development process, we updated the data in the PSL Excel files. After iterating for several months, we uploaded new excel files and imported the new data, and the system updated the data elements already registered in the database and added the latest data.

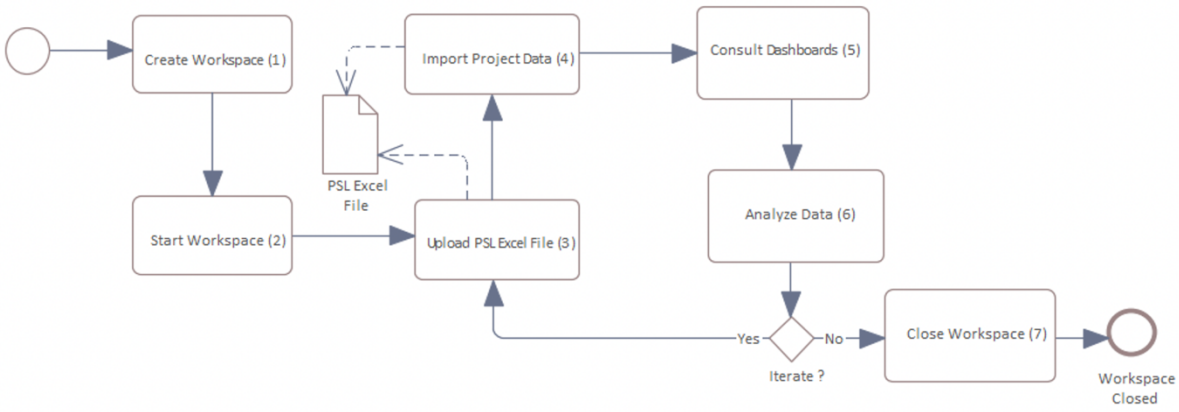


Figure 6.2 – Experiment A Workflow (BPMN Diagram)

After each data import, dashboards were consulted, and we analyzed the project situation and history based on the stored data (Task 6).

This experiment data does not involve the analysis at the organization level. For simplicity, only data from the workspace analytics page were analyzed. Figure 6.3 presents the analytics page of the ITLingo-Cloud project during the last iteration of the project, using ITC.

By analyzing the analytics page after the last project iteration, we can understand relevant information about the current sprint, the progress of the project, working hours, items solved, and users.

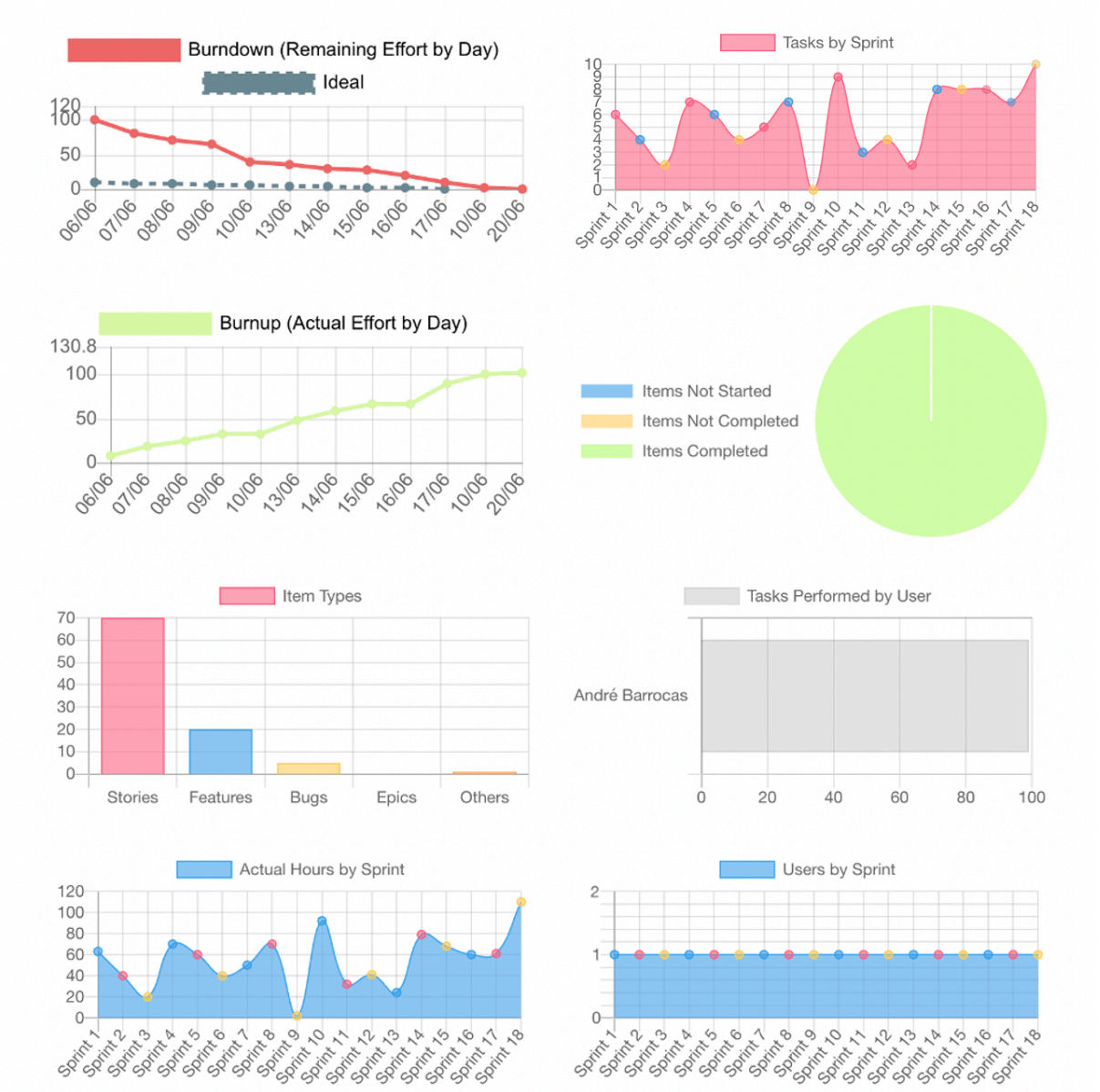


Figure 6.3 – Screenshot of the analytics page after the last sprint

Visualizations related to the last sprint performed present how the work was forecasted with the burndown chart. We noticed that the initial forecast was to give 100 hours for the complete sprint (with two weeks of duration). After each day of the sprint was reduced as the work was being developed. By analyzing burnup, it is possible to explore the actual number of hours spent per day working, concluding that, in total, 104 hours were spent working during the last two weeks. Data relating to past sprints can be found in history. Analyzing other charts, we can verify that there were registered in the product backlog a total of 70 stories (or use cases), 20 features, 6 bugs, and 1 item of type "other". It is also possible to analyze that the total number of tasks completed by André Barrocas was 99. In addition, the dashboard presents the number of users involved in each sprint of this project. There is a relationship between the number of tasks completed and the actual number of hours spent in each sprint. ITLingo-Cloud allows the

analysis in detail of how many hours were spent in each sprint and how many tasks were completed.

6.2. Experiment B: The BIER's Audit System

Experiment B intends to validate ITC in what concerns the functionalities that involve collaboration between stakeholders. This experiment is based on a fictitious organization named Bank of Investment and Economic Recovery (BIER) and a fictitious project discussed in the Information Systems Project Management course in 2022 [52].

6.2.1. Experiment Overview

The Bank of Investment and Economic Recovery (BIER) is a Portuguese investment bank. BIER must comply with the banking regulators (the Bank of Portugal and the European Bank), particularly managing its auditing systems. This compliance is challenging, even at risk, with the current and very outdated BIER's auditing system. Thus, there was the need to start a new project to create a new internal auditing system, focusing on overcoming these compliance and usability limitations. ITLingo-Cloud supported this project. Considering the project's foreseen difficulties in specifying the requirements for this component and dealing with the expected permanent requests for additional functionalities, it was decided to implement the reports and dashboards based on the Scrum process.

6.2.2. Approach Overview

Our approach involves supporting the development of BIER's audit system, namely carrying out some planning tasks on the platform, ending with an analysis of the data generated by the organization not only in this project but also in other existing projects carried out by BIER organization. This organization and this project are fictitious. The data used for analysis was manually stored in excel files and subsequently imported into the platform. The case study is divided into two parts: (1) the first part of the approach supported in this case study involves creating and setting up the environment to start developing this new project, managing the product backlog, and managing the sprints, while (2) the second part of the case study involves the analysis of the organization's and respective workspaces' data to understand and summarize and analyze all the organization's information.

The following sections describe the tasks involved in this case study using the ITLingo-Cloud platform.

Figure 6.4 presents the first part of this experiment as a simplified business process model, supported by the ITLingo-Cloud platform. The workspace manager creates the organization and workspace (Tasks 1 and 2), starts the workspace (Task 3), and invites users to participate (Task 4), giving them specific workspace roles. Then, there are two main continuous flows in parallel: (i) the product manager manages the product backlog (Tasks 6), and (ii) the sprint manager creates and manages each sprint (Task 7).

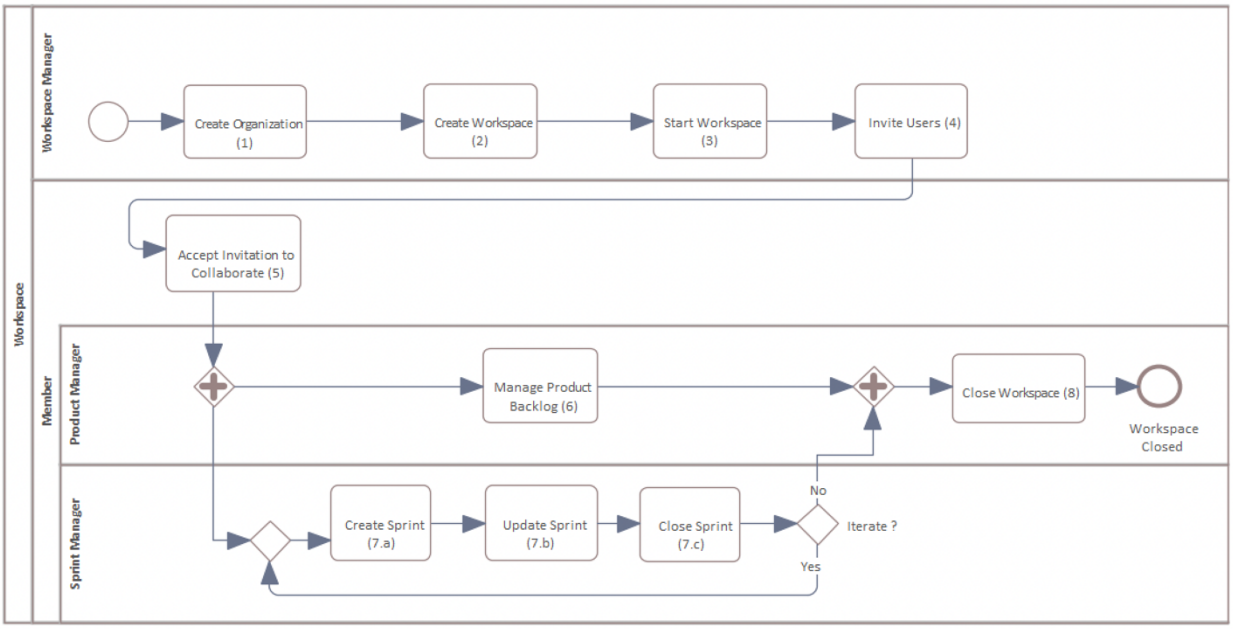


Figure 6.4 - Experiment B Workflow (BPMN Diagram)

Create and Setup the Environment (Tasks 1 to 4)

After the creation of the organization (Task 1), a workspace named "BIER's Auditing System" is created and is configurated the environment with project information (Tasks 2 and 3). The workspace manager invites the project stakeholders (Task 4). ITLingo-Cloud sends an automatic notification with the invite to be part of the workspace, with the appropriate role for each user. The users can accept the invitations (Task 5) directly from the notification bar, making the process quick. Figure 6.5 shows the BIER's Auditing System workspace page.

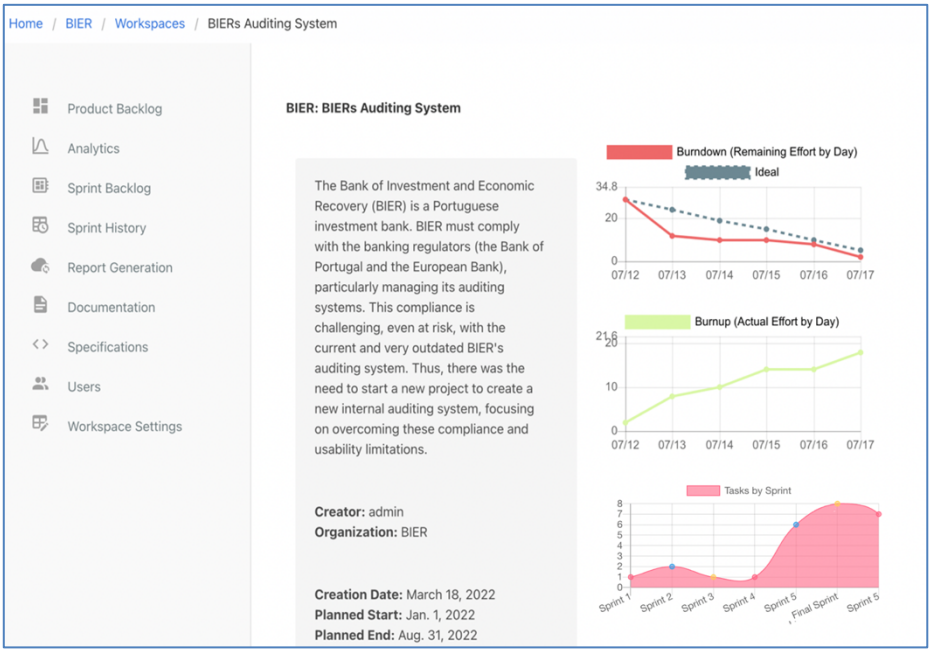


Figure 6.5 - Screenshot of the BIER's Auditing System page

Manage Product Backlog (Task 6)

The requirements were mainly defined during the project's initial phase, and ITLingo-Cloud was used to create the product backlog (Task 6). Some of the requirements were imported directly from excel files, while others were created manually on the platform by the product manager. Figure 6.6 shows the ITLingo-Cloud Product Backlog page.

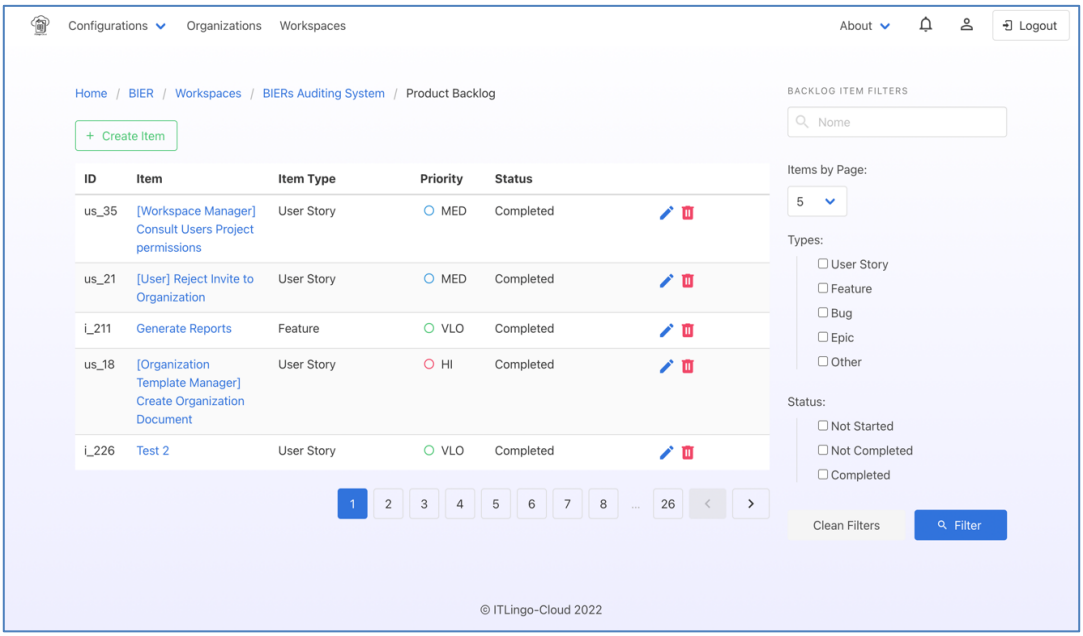


Figure 6.6 - Screenshot of the product backlog page

Manage Sprints (Task 7)

Some weeks after the beginning of the project, the sprint manager created Sprint 5, specifying the start and end date and the users involved (Task 7.a), as shown in Figure 6.7.

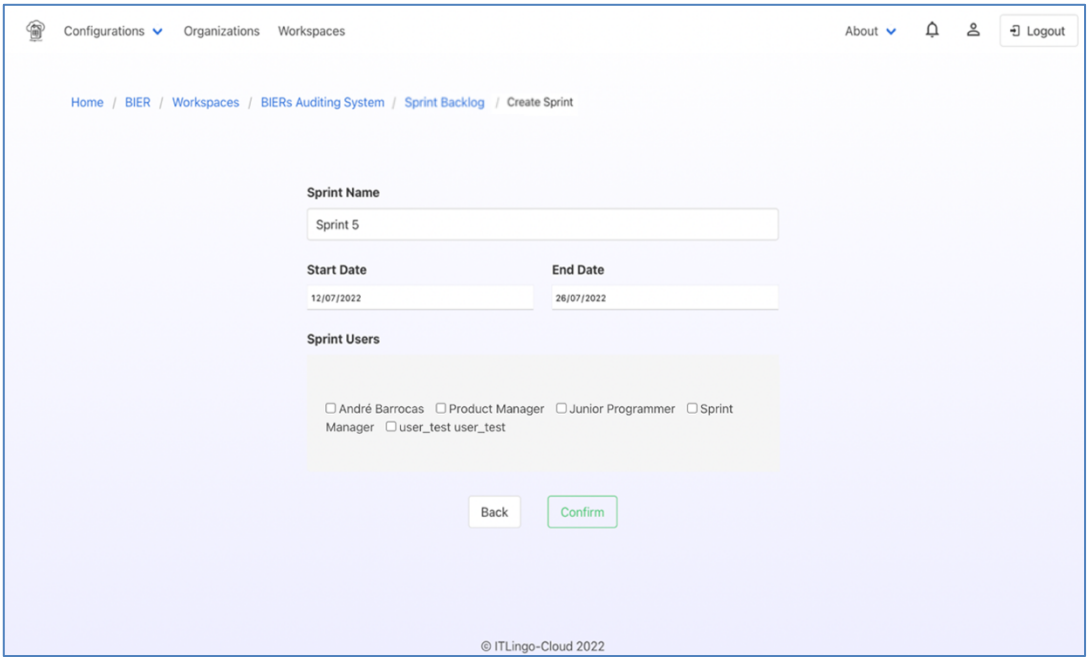


Figure 6.7 - Screenshot of the sprint creation page

To plan Sprint 5 properly, some tasks were imported directly from the Product Backlog, while others were created manually. Each sprint task was assigned the member responsible for its execution, specifying the remaining and actual effort by workday.

Several technical documentation, images, and files with relevant data were assigned to each product backlog item to organize and share information among all parties involved. Each task was managed by dragging in drop using the kanban board or changing its status on the sprint backlog table.

During the development of BIER's auditing system, several iterations were made, and several change requests were made for this project. New data were imported from excel files to the platform (related to existing open issues, new use cases, and changes in the project schedule and budget). The system updated and added the information with the latest data, automatically providing dashboards to analyze the data.

In the second part of this case study, we intend to analyze the data generated by the organization, as well as understand its content and compare it with other workspaces from the BIER organization. For simplicity, we do not analyze the data in the history of each sprint. Figure 6.8 presents the analytics page at the organizational level.

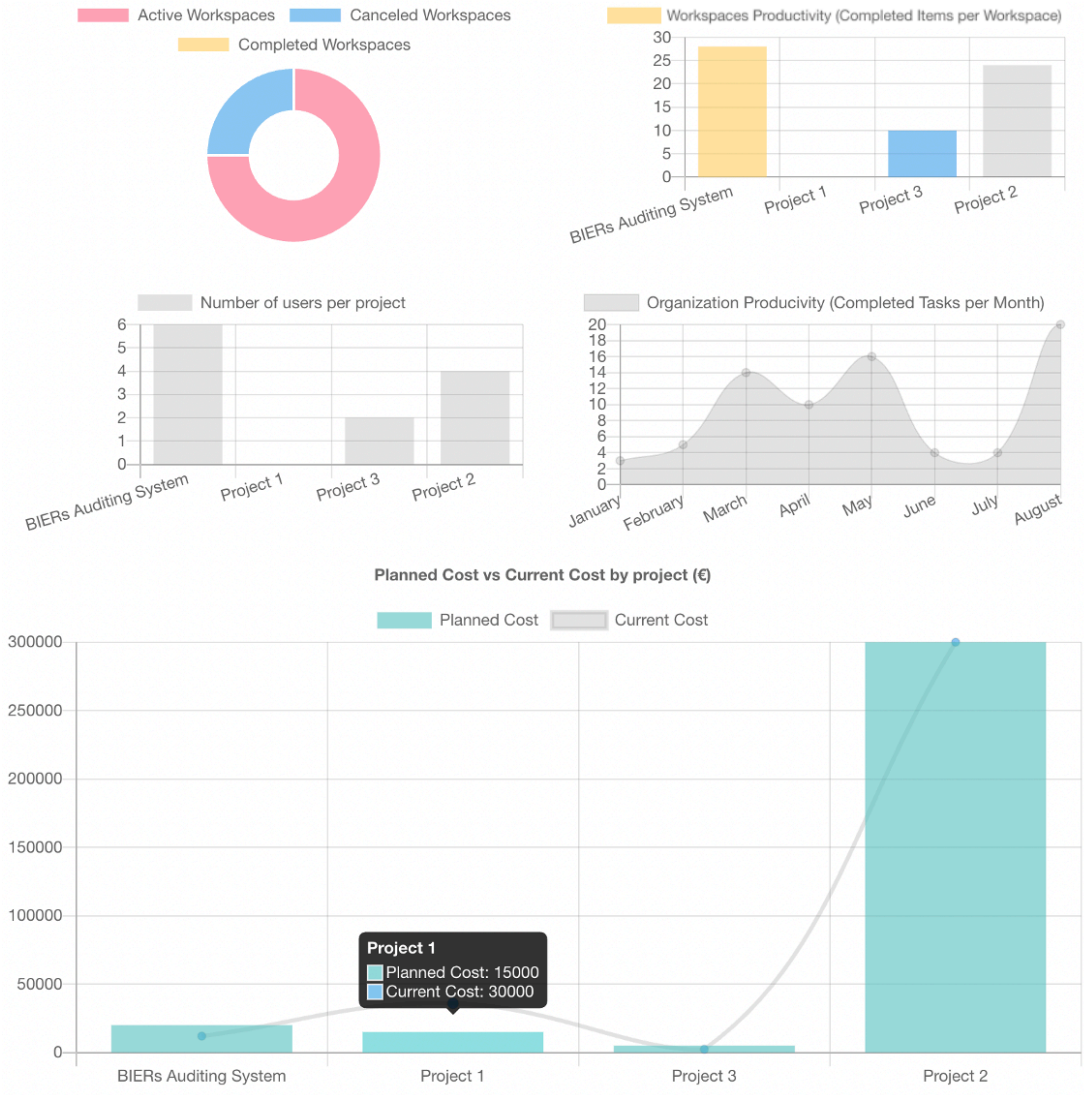


Figure 6.8 – Screenshot of the analytics page at the organization level

By analyzing the data at the platform level, it is possible to take some conclusions and make some comparisons. There are 3 active workspaces and 1 canceled. BIER organization solved a total of 28 items in the Bier's Auditing system workspaces, 0 in Project 1, 10 in Project 3, and 24 in Project 2. The most productive workspace is Bier's Audit System which involves more users. The most productive month of the organization was August, with more work concluded.

By comparing the costs among workspaces, we concluded that Project 2 involved a higher current cost and Project 3 had a lower current cost. Furthermore, it is possible to conclude that Project 1 has a current cost higher than the planned cost. When analyzing the costs related to the BIER's Audit System workspace, it is possible to conclude that the planned cost is higher than the current cost.

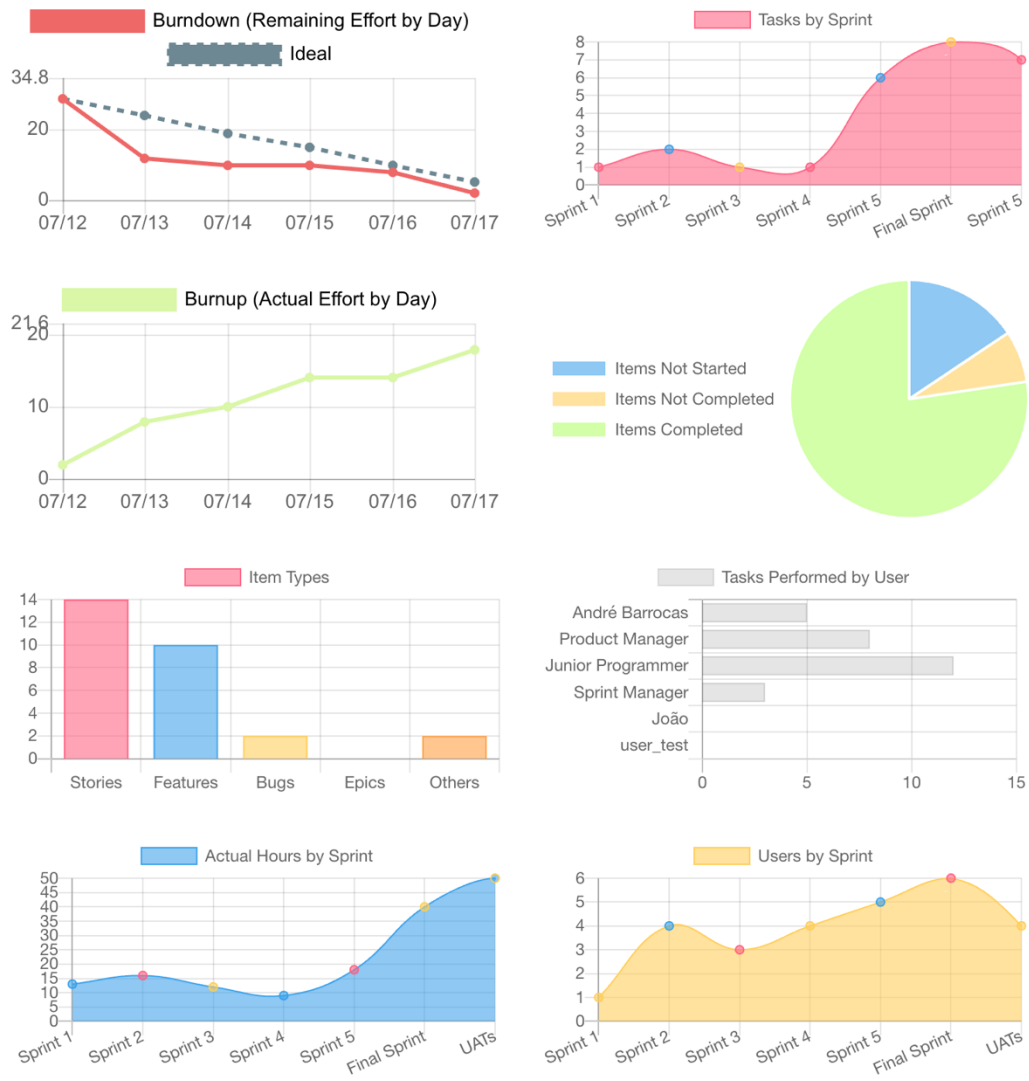


Figure 6.9 – Screenshot of the analytics page at the workspace level

The analytics page was consulted in the last sprint of the project (named "UATs"), as presented in Figure 6.9. When analyzing the workspace data, we can reach several conclusions about the BIER's Audit System project.

By observing the charts related to the last sprint of this workspace (burn-down and burn-up charts), it is possible to analyze how the work was forecasted throughout the last week with the burn-down chart, in particular, to see how the forecast of hours to conclude the sprint was decreasing as time progressed and the work was being done. We can analyze the burn-up chat and observe the accumulation of the number of working hours as the last sprint progressed. Observing the remaining charts related to the complete project life cycle, we can conclude that not all items in the product backlog were completed. It is also possible to observe the summary of the different types of items registered in the product backlog.

By analyzing charts related to project performance, productivity, and people involved, we can take some conclusions and understand the relationships between them. Comparing the tasks completed by sprint and the current hours by sprint, we can conclude that the most productive

sprints, which involved more work and with more completed tasks, are distributed at the end of the project. ITLingo-Cloud allows analyzing in detail the number of tasks implemented in each sprint, the number of hours, and even more details stored in the sprint history. We can also conclude that the user who completed more tasks during the BIER's Auditing System project was the junior programmer, with 12 tasks completed. Finally, we can conclude which sprints involved more people and understand the relationship with other productivity chars.

6.3. User Assessment

We conducted a user assessment with other researchers, teachers, and students to evaluate the system and receive preliminary feedback from people not directly involved in this research.

This activity was based on the BIER organization and BIER' audit project mentioned in the Section 6.2. These tests were helpful to find problems and solve them, and evaluate the usability, some collaborative features, as well as the capacity to understand the data visualization mechanisms provided by ITLingo-Cloud, in particular, to test the ability of the users to analyze BIER's organization data and take some conclusions about it. In this evaluation process, we used usability tests with users in individual sessions to interact directly with the platform. Finally, we elaborated a questionary for the users to gather vital feedback to guarantee that the system meets the user's needs and analyze the obtained results.

The tests phase was performed between June and August of 2022. The questionary was answered by a group of 21 participants with ages ranging from 18 to 60 years old and at least a Bachelor of Science degree, namely 9 with a BSc, 9 with an MSc, and 3 with a Ph.D. degree.

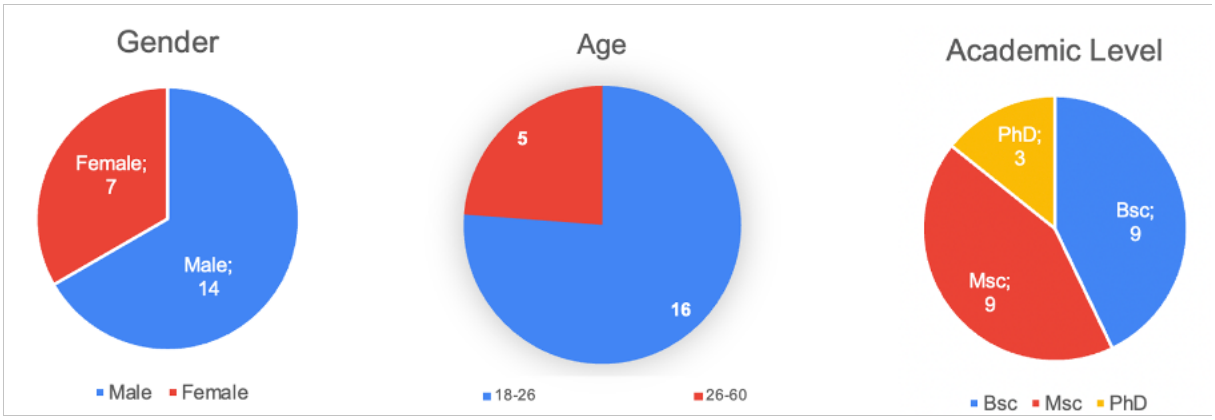


Figure 6.10 - Demographic analysis of the group of participants in the user assessment

Most participants had little professional experience, in particular, 13 participants with less than 1 year, 5 participants between 1 and 5 years, 1 participant between 5 and 10 years, and 2 participants with more than 10 years of experience.

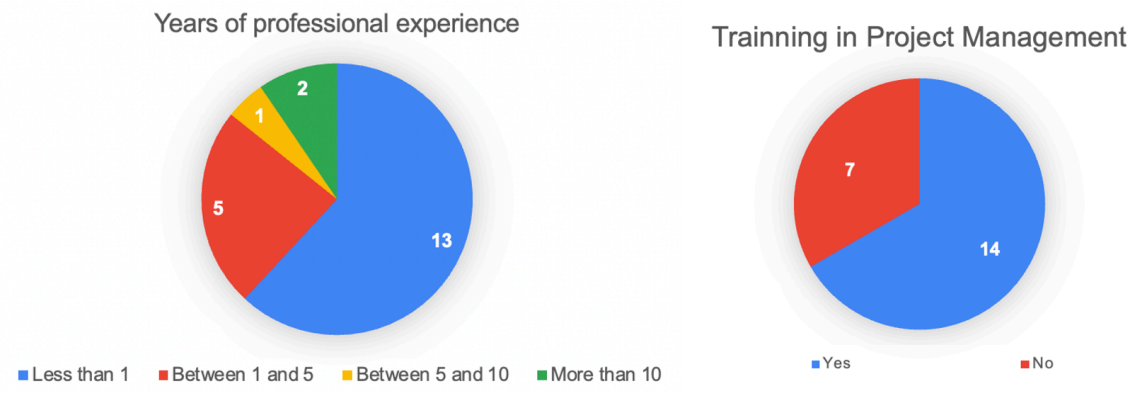


Figure 6.11 – Background analysis of the group of participants in the user assessment

By analyzing the users' background, it is possible to understand that most users have previous training in the project management field. In particular, 14 of the participants have previous training while 7 have no. By comparing the answers, it is possible to understand that most of the users have experience with project management tools while the minority have no experience with these tools.

The user pilot-user test session was conducted under the following conditions:

- The tests were conducted in a controlled environment without distractions (e.g., office or home environment).
- Realization of the tasks without previous use and learning of the system.
- The users were required to have an internet connection and a browser with good JS support.
- Users were free to think out loud and share ideas.
- The evaluator didn't interact with the users until the tests are finished (except in case of blocking errors).
- The session had a maximum of 45 minutes of duration.

To guide the user assessment test, participants received a script (available in Appendix A) briefly describing the ITLingo-Cloud platform, explaining the goals of the activity, the BIERs' organization, and a fictitious project to be tracked with our system. After contextualizing the users, the steps to be followed to carry out the evaluation tasks were explained.

As in experiment B, the user assessment session was divided into two parts: the first part involved testing some of the collaborative features of the platform, while the second involved the ability of users to use the data visualization features, in particular, to understand whether they were able to apprehend the preliminary information that the dashboards intend to transmit.

The first part included tasks 1 to 24 of the script, which involved the registration on the platform, creation of the organization and workspace, and management of the product backlog and sprints. The second part is referent to tasks 25 to 33 of the script. The users were asked to analyze the general statistics related to the BIER organization, its analytics page, the available workspace statistics, and its summarized analytics page, as well as the data analysis and visualization of a stored specific sprint in the historic. The organization data used for test purposes were mainly imported from excel files, including information related to fictitious

projects, to make it possible to compare data at an organization's global level, comparing information among its projects, stakeholders, productivity, effort, and performance.

In the end, participants were asked to fill in a questionnaire to rate the platform, suggest improvements, report errors, and answer the data analysis questions.

The questionary is divided into the following sessions:

Section 1: Respondent Characterization. The first three questions (Q1 to Q3) were focused on the general characterization of the participants with the following aspects: age, gender, and academic level.

Section 2: User Background Analysis. Four questions (Q4 to Q7) were directly related to user background analysis, namely related to the years of experience, previous experience in project management subjects, and related tools.

Section 3: User Experience Overall Assessment. Eight questions (Q8 to Q15) focused on test usability and learnability aspects of ITLingo-Cloud. We asked the percentage of completed the tasks proposed and asked participants to rate on a 5-Likert scale (i.e., from 1 to 5, 0-completely disagree, 5-completely agree) aspects related to the interface complexity, usability, feature integration, interface inconsistencies, and learnability.

Section 4: Data Analysis Questions. We asked five questions (Q16 to Q20) to evaluate the users' capacity to interpret the dashboards' information. There are a lot of possible aspects to ask related to the dashboards. Still, we focused on testing only two questions related to organization data analytics, two related to workspace data analytics, and one to interpret historic sprint data. We asked about organization productivity, costs, effort, hours of work, and completed items to check if the users understood the charts asking for a concrete answer.

Section 5: ITLingo-Cloud General Evaluation: This section has the highest number of questions (Q21 to Q30). In this part of the questionnaire, we asked about aspects related to the usefulness of specific features provided by the platform. We first asked participants to rate in a 5-Likert scale (i.e., from 1 to 5, 0—Do not know, 1-Very Low, 2 -Low, 3-Medium, 4-High, and 5-Very High) how useful the functionalities are to create and manage organizations, workspaces, import project-related data, manage the sprints and tasks, as well as the functionalities to analyze organization's and workspace's statistics, and sprint dashboards. Then, we include an open-ended question for users to provide additional comments (suggestions, problems, or bugs).

6.3.1. Survey Analysis

As mentioned above, the questionary is divided into five sections. The first and second sections were already analyzed previously, describing participant-related aspects, namely the general characterization and background.

Section three analyzed ITLingo-Cloud usability and learnability aspects. Responses in this section revealed good interface results. When analyzing the answers, we concluded that 100% of the tasks were completed by the participants. For the remaining questions, we asked participants to rate on a 5-Likert scale (i.e., from 1 to 5, 1-completely disagree, 5-completely agree). Table 6.1 summarizes the average scores for these questions, based on which we may verify the following findings: Most participants would use the product again. When analyzing the results for the question Q11 - The system is more complex than necessary, we found that

although most people disagreed with the question, there were still three people who answered 4, and one who answered 3. The vast majority of people agreed that the system is easy to use and easy to learn, feeling confident when using the product.

Some of the feedback provided by the participants on the open-ended question about the interface and usability aspects were: “The tool is very visual and intuitive, allowing a very interesting analysis of information.” and “The platform is very complete and very visual, which I find great. Good job!”.

Table 6.1 – User experience overall assessment results

Questions	Average
I think I would use this product again.	4.43
The system is more complex than necessary.	1.81
The system is easy to use.	4.57
The various features of this product were well integrated.	4.62
This system has inconsistencies.	1.52
I suppose most people would quickly learn to use this product.	4.71
I felt very confident using this product.	4.67

(values on a 1–5 scale, 1-completely disagree, 5-completely agree)

Section 4 evaluates the users’ capacity to understand the dashboards, as well as the ability to understand the information whose graphics are intended to convey and verify that the data analysis was done correctly, asking for concrete answers. Regarding the first question: Q16 - "Which project has more items solved in this organization?" all participants got it right. In the remaining questions, the vast majority got the answer right. The percentage of correct answers was 95.2% in the question Q17 - “Which project has the current cost higher than planned cost?”, 95.2% in the question Q18 - “Which was the remaining effort foreseen in day 12?”, 90% in the question Q19 - “Which sprint involved more hours of work?” and 90.5% in the question Q20 - “How many items were completed?”.

The participants provided the following feedback on the open-ended question about the data analysis features: “Good monitoring and data analysis tool. You can monitor users and projects (active, canceled, and completed), among other tasks. One note would be to increase the size of the graphs to allow better visibility of the numbers on the scales of the graphs.”. The last suggestion was followed, and the numbers of some of the graphs were increased as well as some colors were refined for a better understanding of the values.

Section 5 was very important to detect bugs and system improvements, particularly at the beginning of the testing phase. Users who tested the platform more intensively found a bug in the system that caused a message with no text when no description is provided. Another bug found was related to the drag-and-drop provided by the kanban board. This bug was related to the need to manually refresh the page for the kanban board items to be updated. These bugs were fixed at the beginning of the evaluation phase.

Other improvement suggestions were: "I think the description should be an optional field (I ended up placing the name of the task in the description just to fill)" and "It would be interesting

to add more integration features with other management tools, for example ". The first suggestion was followed, and task description is no longer a required field. The second suggestion was not implemented because this feature requires extensive future work. Nonetheless is an option to consider for future work.

In section 5, we also asked about aspects related to the usefulness of specific features provided by the platform. Participants rated, in a 5-Likert scale (i.e., from 1 to 5, 1-Very Low, 2 -Low, 3-Medium, 4-High, and 5-Very High), the usefulness of the functionalities of creating and managing organizations, workspaces, import project-related data, manage the sprints and tasks, as well as the functionalities to analyze organization's and workspace's statistics, and sprint dashboards. Table 6.2 summarizes the average scores for these questions based on which we may verify the following findings: All answers were between 4 and 5 (4-High, and 5-Very High) in all questions, with the majority being 5. In question Q23 - How do you rate the usefulness of import product backlog data? there was one person who answered 3 (medium), and the remaining answers were between 4 and 5. With these results, we can conclude that users found all the features they evaluated useful.

Table 6.2 – ITLingo-Cloud usefulness assessment results

Questions	Average
How do you rate the usefulness of creating and managing organizations?	4.95
How suitable is the platform for creating and managing workspaces?	4.81
How do you rate the usefulness of importing product backlog data?	4.71
How do you rate the usefulness of managing the sprints and the tasks?	4.80
How do you rate the usefulness of the sprint board and its filters?	4.71
How do you rate the usefulness of the organization's data analysis?	4.90
How do you rate the usefulness of the workspace's data analysis?	4.95
How do you rate the usefulness of the sprints' data analysis?	4.95

(values on a 1–5 scale, 1-very low, 5-very high)

Regarding the open-ended questions of section 5, most of the participants did not answer. However, in addition to the feedback, improvements, and bugs that have already been mentioned before, those that answered provided encouraging comments and feedback, such as: "Very good Congratulations!", and "Congratulations on your work!", among the other positive feedback mentioned previously. The analysis of these comments led to the conclusion that this tool is useful and that the work was carried out successfully. To sum up, the results collected and analyzed in all sections had very positive scores. Usability experts like Nielsen and Landauer observed that a group of 5 testers is enough to uncover over 80% of the usability problems [53]. Since our questionnaire focuses on the usability and general evaluation of the ITLingo-Cloud platform, we may conclude that 21 participants are a fair number for an exploratory assessment, allowing us to identify significant flaws in the usability of such proposals. Furthermore, this assessment was also handy for detecting and solving a few bugs.

6.4. Comparison with the Related Work

Comparing project management aspects, Microsoft Project surpasses other tools with mechanisms related to visualizing and managing resources, in particular, to track time and costs [37], [42]. Jira provides helpful dashboards and customizable pages to share information among stakeholders and is useful in collaboration aspects. Trello is user-friendly and offers communication mechanisms but limited reports and dashboards compared to the other systems, only supporting the Kanban framework. ClickUp is customizable and adapts to different needs, providing different ways to visualize data. ITLingo-Cloud, Jira and ClickUp are better for supporting several workspaces inside the same organization, providing a hierarchy to manage and track at the organization level [39], [41].

Despite of all tools provide dashboards at the workspace level, ITC provides more detailed data analysis at the organizational level, allowing the comparison of productivity over time, cost analysis, and other data among workspaces.

All tools support the creation of personalized reports, being possible to reuse them. However, ITLingo-Cloud has other advantages over these tools, namely supporting document automation features crucial in each project life cycle (e.g., Communication Plan, Product Backlog, Project Charter, Project Final Report, Status Report) and includes an IDE to allow the rigorous specification of technical documentation based of textual domain specification languages (DSL) such as RSL and PSL.

ITLingo-Cloud provides other features that go beyond these project management tools. In particular, it supports features that allow software development directly through a cloud IDE. Besides that, ITC supports a workspace hierarchy and information storage common with other cloud tools. Due to these reasons, ITLingo-Cloud can also be compared with cloud IDE tools such as Google Cloud, AWS Cloud9, Github Codespaces, and Code Ocean.

However, these IDE tools are specialized or suitable for specific areas despite of all approaches include a cloud IDE to develop software and their interfaces are very similar. Google Cloud, AWS cloud9, and Github Codespaces surpass other tools in allowing the installation of more dependencies and including more features. All approaches allow the storage of files. Google Cloud and AWS cloud9 provide more services than other approaches, while ITC is more suitable for managing and analyzing organizations' and projects' information.

7. Conclusion

This dissertation proposes the ITLingo-Cloud system, a collaborative platform that supports multiple software development processes and project management activities, allowing to manage and analyze organizations' data. ITLingo-Cloud is a platform that provides an appealing and easy-to-use interface that software developers, managers, and other stakeholders can use to keep a large amount of project-related information always accessible. With this work, we research and address the following concerns: (1) increase the performance in organizations; (2) managing projects more intuitively, allowing people to synchronize information and work collaboratively; (3) generating and managing project-related data, keeping it accessible to all interested parties; (4) improve collaboration among stakeholders within organizations and workspaces; and (5) provide visual mechanisms to directly translate that knowledge on project performance into decision making support. Thus, ITLingo-Cloud platform intends to support these issues by providing a collaborative environment with reusability and adaptability features, supporting synchronization with other tools and technologies.

7.1. Main Contributes

The most important contributes of this research are the following:

First, the design, development, and test of the ITLingo-Cloud platform, which the primary objective is to support project management processes, as well as to manage and analyze information for better decision making.

Second, integrate Excel template files developed in the ITLingo initiative (in particular, the latest version of the PSL Excel template). These templates are used to manage and synchronize data directly with the ITLingo-Cloud platform.

Third, integrate the word template files (previously studied by Domingos Bragança in the scope of the ITLingo PSL research project), these word documents were used in ITLingo-Cloud to generate reports saving time and aiding in project management processes.

Fourth, integrate ITOI, a cloud IDE tool that allows to develop and test code directly in the browser (developed by Ricardo Silva).

7.2. Future Work

This research identifies yet several aspects that may be addressed in future work.

First, the ITC may improve its data analysis capabilities, including machine learning algorithms, namely to better monitor project costs through predictive data analytics models to increase project productivity, helping managers make more informed decisions. Furthermore, although the ITC already provides alerts when the project's end is approaching (e.g., a few days before the project's planned end date), the system could forecast planned dates based on data from past projects.

Second, the ITC support for multiple processes could be extended and improved. For example, the support of the Kanban process already implemented in the platform could be improved to be able to define customizable lanes and WIP limits.

Third, this research may also involve the completion of the document automation module to include more reports on the platform.

Finally, the integration of ITC with ITOI may still involve relevant research, namely including more common cloud IDEs features such as more developer tools, allowing software development and project management in an integrated platform.

Appendix A

User Session Guide:

ITLingo-Cloud Platform: Support for Agile Project Management

User Assessment, Jun 2022

André Barrocas (MSc Student)

Alberto Rodrigues da Silva and João Saraiva (Supervisors)

INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal

ITLingo-Cloud is a multi-organization and multi-workspace collaborative platform to analyze and manage business data.

Its users can easily set up their environment, manage workspaces and technical documentation and analyze and observe statistics to aid business decisions. ITLingo-Cloud has been designed and developed considering the solutions available on the market, specifically to include standard features found in agile project management tools and inspired by emerging collaborative solutions like cloud IDE tools.

This user test session aims to assess ITLingo-Cloud focused on its agile project management features. This session will be conducted by researchers, students, or professionals unfamiliar with this ITLingo-Cloud.

The main goal is to receive helpful feedback to check if the requirements were achieved and detect problems and significant constraints.

Thanks for your participation and feedback!

This activity is based on the IAM project studied in detail during the GPI course. The case study is described below.

Case Study – BIER's auditing system

The Bank of Investment and Economic Recovery (BIER) is a Portuguese investment bank, 51% owned by the Government and 49% owned by a consortium of private banks. With around 500 employees, BIER has 30 branches located in the continent and autonomic regions' major cities. BIER must comply with the banking regulators (the Bank of Portugal and the European Bank), particularly managing its auditing systems. This compliance is challenging, even at risk, with the current and very outdated BIER's auditing system. The current audit system presents some problems. It is not user-friendly, challenging to maintain, and does not fulfill the regulatory requirements. In this way, there was the need to start a new project to create a new internal auditing system, focusing on overcoming these compliance and usability limitations. ITLingo-Cloud supported this project. Considering the foreseen difficulties of the project to specify the requirements for this component and deal with the expected permanent requests for additional functionalities, it was decided to implement the reports and dashboards based on the Scrum process.

Consider that your company (the contractor) started developing the project. During the project's initial phase the requirements were specified using ITLingo-Cloud to create the product backlog. As planned, the first sprint needs to be initialized. Please use ITLingo-Cloud to import and manage project data and analyze the project data with intuitive statistics. Click on the following link to access the ITLingo-Cloud platform: https://itlingocloud.herokuapp.com

Conditions

The user pilot-user test session will be conducted under the following conditions:

- The tests are conducted in the laboratory (controlled environment) without distractions.
- Realization of the tasks without previous use and learning of the system.
- The user must have an internet connection and one browser with good JS support.
- Users can think out loud and share ideas.
- The evaluator does not interact with the users until the tests are finished (except in case of blocking errors).
- The session will last 45 minutes (at most).

General Instructions

To simplify the user assessment, only a small part of the ITLingo-Cloud platform will be tested.

User Registration / Create Organization

1. Register on the platform by choosing the organization “IAM-XX” (XX is your name) in the

organization field.

2. Log in to the platform.
3. Select “Organizations”.
4. Choose the organization “IAM-XX” (XX is your name). Organization statistics and charts are not provided yet because your organization has no data imported or created.

Create Workspace

5. Select “Workspaces” in the menu.
6. “Create Workspace” named "BIER's Auditing System".
7. Consult your workspace page.

Manage Product Backlog

8. Select “Product Backlog”
9. Create the following backlog items directly on the Product Backlog (can use the default values or change them according to your preferences):
    a. “Delete Audit Plans”
    b. “Create Audit Plans”
    c. “Consult Audit Plans”

    (To simplify and speed up this user assessment, the data will not be imported from excel files)

Create Sprint

10. Back to "BIER's Auditing System" Page
11. Choose the “Sprint Backlog” option.
12. Create a new sprint named “Sprint 1” starting today with one week of duration and select the user(s).

Manage Sprint

13. Create the following tasks aligned with the Product Backlog, “Create Task | From Product Backlog”:
    a. “Create Audit Plans”
    b. “Consult Audit Plans”
14. Create a new task manually, “Create Task | New Task”:
    a. With the name “Create IAM System Logo”.
    b. With the planned effort of 8h.
    c. Specify the task type, priority, member(s) to implement, effort, and a brief description (can use the default values or change according to your preferences).
15. Start the sprint.
16. Edit the Sprint by clicking on the “Edit Sprint” option.
    a. Change the sprint name to “Initial Sprint”
    b. Click “Confirm”

17. Edit task “Create Audit Plans” by clicking on the task name.
    a. Change the task description.
    b. Click “Confirm”
18. Change task “Create Audit Plans” remaining and actual effort.
    a. Click on the “Effort” icon in the “Actions” section for this task.
    b. Write 16 hours in the remaining effort.
19. Select the “Sprint Board” button.
20. Conclude the “Create Audit Plans” task by dragging the task.
21. Select the “Table View” button and note the task status changes.
22. Close the sprint by clicking on the “Conclude Sprint” button.
23. Consult the sprint history by clicking on the “Sprint History” option.
    a. Consult the sprint history tasks of sprint 1 by clicking on the “Consult Sprint Tasks” option.
    b. Back to sprint history.
    c. Consult the sprint statistics of sprint 1 by clicking on the “Consult Sprint Statistics” option.
24. Logout on the platform.

Data Analysis

Consider that some organization data were imported from excel files, others were created manually. This data is regarding this project and other projects, including about stakeholders, product backlog data, and sprints.

25. Log in to the platform using the following credentials:
    a. Username: user_test
    b. Password: itlingo123456
26. Select “Organizations”.
27. Choose the organization “IAM”.
28. Consult the general statistics related to this organization in home page.
    a. Observe the organization productivity.
    b. Analyze productivity by project.
        i. Which project has more items solved in this organization? Please register the project name to answer in the questionnaire below.
    c. Organization’s projects status.
29. Click in “Analytics” and analyze the organization’s data through statistical dashboards provided.
    a. Observe the organization productivity and projects productivity.

b. Number of users by project.

c. Compare planned cost vs current cost by project.

    i. Which project has the current cost higher than planned cost? Please register the project name to answer in the questionnaire below.

30. Click on the “Workspaces” button and select “BIERs Auditing System”.

    a. Observe the project productivity.

        i. Burndown chart.

            1. Which was the remaining effort foreseen in day 12? Please register the value to answer in the questionnaire below.

        ii. Burnup chart.

        iii. Tasks by sprint.

        iv. Items statistics.

31. Click in “Analytics” and analyze the workspaces' data through statistical dashboards provided.

    a. Analyze workspace productivity charts.

    b. Observer items statistics.

    c. Tasks performed by user.

    d. Current hours by sprint.

        i. Which sprint involved more hours of work? Please register the value to answer in the questionnaire below.

    e. Users involved in each sprint.

32. Click in “Sprint History” and click on the “Consult Sprint Statistics” icon on Sprint 5.

    a. Consult historical data from Sprint 5.

        i. Burndown chart.

        ii. Burnup chart.

        iii. Items statistics.

            1. How many tasks were completed? Please register the number to answer in the questionnaire below.

33. Logout on the platform.

Questionnaire

After performing the steps described in the general instructions section, please give us your feedback, by answering the following questionnaire: https://tinyurl.com/2cswfama

References

[1] P. Patanakul, B. Iewwongcharoen, and D. Milosevic, "An empirical study on the use of project management tools and techniques across project life-cycle and their impact on project success," *Journal of General management*, vol. 35, no. 3, pp. 41–66, 2010.

[2] K. el Emam and A. G. Koru, "A replicated survey of IT software project failures," *IEEE Softw*, vol. 25, no. 5, pp. 84–90, 2008.

[3] A. R. da Silva, "ITLingo Research Initiative in 2022," *arXiv preprint arXiv:2206.14553*, 2022.

[4] M. Nayebi, G. Ruhe, R. C. Mota, and M. Mufti, "Analytics for Software Project Management--Where are We and Where do We Go?," in *2015 30th IEEE/ACM International Conference on Automated Software Engineering Workshop (ASEW)*, 2015, pp. 18–21.

[5] V. Novitzká *et al.*, *Informatics 2017 : 2017 IEEE 14th International Scientific Conference on Informatics : proceedings : November 14-16, 2017, Poprad, Slovakia*.

[6] I. Gamito and A. R. da Silva, "From Rigorous Requirements and User Interfaces Specifications into Software Business Applications," in *International Conference on the Quality of Information and Communications Technology*, 2020, pp. 459–473.

[7] D. Zowghi and Z. Jin, *Requirements engineering*. Springer, 2011.

[8] B. Hansoti, "Business Intelligence Dashboard in Decision Making," 2010.

[9] M. Sadiku, A. E. Shadare, S. M. Musa, C. M. Akujuobi, and R. Perry, "Data visualization," *International Journal of Engineering Research And Advanced Technology (IJERAT)*, vol. 2, no. 12, pp. 11–16, 2016.

[10] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004, doi: 10.2307/25148625.

[11] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A design science research methodology for information systems research," *Journal of management information systems*, vol. 24, no. 3, pp. 45–77, 2007.

[12] F. Gacenga, A. Cater-Steel, M. Toleman, and W.-G. Tan, "A proposal and evaluation of a design method in design science research," *Electronic Journal of Business Research Methods*, vol. 10, no. 2, pp. pp89-100, 2012.

[13] A. R. da Silva, "Rigorous Specification of Use Cases with the RSL Language, International Conference on Information Systems Development'2019, AIS," 2019.

[14] Project Management Institute., *A guide to the project management body of knowledge (PMBOK Guide).* Project Management Institute, 2008.

[15] R. Stellingwerf and A. Zandhuis, *ISO 21500 Guidance on project management–A Pocket Guide*. Van Haren, 2013.

[16] M. Vukomanović, M. Young, and S. Huynink, "IPMA ICB 4.0—A global standard for project, programme and portfolio management competences," *International Journal of Project Management*, vol. 34, no. 8, pp. 1703–1705, 2016.

[17] E. Commission and D.-G. for Informatics, *PM² project management methodology : guide 3.0.1*. Publications Office, 2021. doi: doi/10.2799/022317.

[18] J. Charvat, "Project management methodologies: selecting, implementing, and supporting methodologies and processes for projects," 2003.

[19] T. Thesing, C. Feldmann, and M. Burchardt, "Agile versus Waterfall Project Management: Decision model for selecting the appropriate approach to a project," in *Procedia Computer Science*, 2021, vol. 181, pp. 746–756. doi: 10.1016/j.procs.2021.01.227.

[20] M. Špundak, "Mixed agile/traditional project management methodology–reality or illusion?," *Procedia-Social and Behavioral Sciences*, vol. 119, pp. 939–948, 2014.

[21] K. Beck *et al.*, "Manifesto for agile software development," 2001.

[22] M. Alqudah and R. Razali, "A comparison of scrum and Kanban for identifying their selection factors," in *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)*, 2017, pp. 1–6. doi: 10.1109/ICEEI.2017.8312434.

[23] J. G. Zheng, "Data visualization in business intelligence," in *Global business intelligence*, Routledge, 2017, pp. 67–81.

[24] M. Hardin, D. Hom, R. Perez, and L. Williams, "Which chart or graph is right for you?," *Tell Impactful Stories with Data. Tableau Software*, 2012.

[25] T. R. Vetter, "Fundamentals of research data and variables: the devil is in the details," *Anesth Analg*, vol. 125, no. 4, pp. 1375–1380, 2017.

[26] W. W. Eckerson, *Performance dashboards: measuring, monitoring, and managing your business*. John Wiley & Sons, 2010.

[27] M.-A. Storey and C. Treude, "Software Engineering Dashboards: Types, Risks, and Future," in *Rethinking Productivity in Software Engineering*, Springer, 2019, pp. 179–190.

[28] A. Cabri and M. Griffiths, "Earned value and agile reporting," *AGILE 2006 (AGILE'06)*, pp. 6-pp, 2006.

[29] P. Batista, "Transformations and Alignments in Project Management Tools: The PSL-v.2020 Experiments: MSc dissertation. Instituto Superior Técnico, Universidade de Lisboa.," 2020.

[30] J. Plekhanova, "Evaluating web development frameworks: Django, Ruby on Rails and CakePHP," *Institute for Business and Information Technology*, vol. 20, p. 2009, 2009.

[31] "openpyxl", Accessed: Aug. 06, 2022. [Online]. Available: https://openpyxl.readthedocs.io/en/stable/

[32] "python-docx", Accessed: Aug. 06, 2022. [Online]. Available: https://python-docx.readthedocs.io/en/latest/

[33] A. Hourieh, *Learning Website Development with Django*. Packt Publishing Ltd, 2008.

[34] A. Ravindran, *Django Design Patterns and Best Practices*. Packt Publishing Ltd, 2015.

[35] "Chartjs", Accessed: Jul. 17, 2022. [Online]. Available: https://www.chartjs.org/

[36] S. Benbba, “Comparison of D3. js and Chart. js as visualisation tools (Bachelor’s thesis),” 2021.

[37] “Microsoft Project.” https://project-management.com/the-pros-and-cons-of-using-microsoft-project-software/ (accessed Jan. 04, 2022).

[38] M. Sajad, M. Sadiq, K. Naveed, and M. S. Iqbal, “Software Project Management: Tools assessment, Comparison and suggestions for future development,” *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 16, no. 1, p. 31, 2016.

[39] “Jira.” https://www.atlassian.com (accessed Jan. 07, 2022).

[40] “Trello.” https://trello.com/home (accessed Jan. 05, 2022).

[41] “ClickUp.” https://clickup.com (accessed Jan. 05, 2022).

[42] E. Marmel, *Microsoft Office Project 2003 Bible*. John Wiley & Sons, 2004.

[43] “Google Cloud Platform.” https://cloud.google.com/gcp?hl=en (accessed Aug. 18, 2022).

[44] T. Hunter and S. Porter, *Google Cloud Platform for developers: build highly scalable cloud solutions with the power of Google Cloud Platform*. Packt Publishing Ltd, 2018.

[45] “Code Ocean.” https://codeocean.com/ (accessed Aug. 18, 2022).

[46] “Cloud9.” https://aws.amazon.com/cloud9/ (accessed Jan. 05, 2022).

[47] “Github Codespaces.” https://github.com/features/codespaces (accessed Jan. 09, 2022).

[48] D. de Carvalho Bragança, “Document Automation in ITLingo PSL Excel Template: MSc dissertation. Instituto Superior Técnico, Universidade de Lisboa.,” 2021.

[49] A. Ravindran, *Django Design Patterns and Best Practices*. Packt Publishing Ltd, 2015.

[50] “VScode”, Accessed: Aug. 11, 2022. [Online]. Available: https://code.visualstudio.com

[51] Ricardo Silva, “ITOI: Towards a Collaborative and online IDE for ITLingo Languages: MSc Project Report. Instituto Superior Técnico, Universidade de Lisboa.,” 2022.

[52] “Information Systems Project Management. Instituto Superior Técnico, Universidade de Lisboa.”, Accessed: Aug. 25, 2022. [Online]. Available: https://fenix.tecnico.ulisboa.pt/disciplinas/GPI/2021-2022/2-semestre/pagina-inicial

[53] J. Nielsen and T. K. Landauer, “A mathematical model of the finding of usability problems,” in *Proceedings of the INTERACT’93 and CHI’93 conference on Human factors in computing systems*, 1993, pp. 206–213.