

EXPENSES REPORTING TOOL FOR AN IT CONSULTANCY

Project Management (GEP)
Assignment 1: Context and Scope

Bachelor Degree in Informatics Engineering
Specialization in Software Engineering

Bachelor Thesis



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona



Natalia Yike Wang Yang

Thesis Supervisor

Sergio Alejandro Lobig Machado (FSP Consulting Services Limited)

Thesis Director

Marc Oriol Hilari (Department of Software Engineering)

GEP Tutor

Eduard Balbuena Longo (Department of Project Management)

Universitat Politècnica de Catalunya
Facultat Informàtica de Barcelona
Computer Science Department

February 25, 2026

Contents

1	Context	5
1.1	Introduction	5
1.2	Key Concepts	5
1.3	Problems to Solve	7
1.4	Proposed Solution	7
1.5	Stakeholders	7
2	Justification	8
2.1	Market Analysis	8
2.2	Project Justification	10
3	Scope	10
3.1	Objectives	10
3.2	Requirements	11
3.3	Obstacles and Risks	13
4	Methodology	14
4.1	Monitoring Tools	15
4.2	Validation Methods	16

List of Figures

1	Agile Methodology Diagram (Wrike). Source:[11]	14
---	--	----

List of Tables

1	Comparative Analysis of Leading Expense Management Solutions (own creation)	9
---	---	---

1 Context

AThis is a Bachelor's Thesis belonging to the Software Engineering Specialization of the Computer Engineering Degree, which is provided by the Barcelona School of Informatics at the Polytechnic University of Catalonia.

This project corresponds to the B modality, which means it has been carried out in a professional context in a collaboration with FSP Consulting Services Limited. It addresses the company's operational needs and is supervised under a dual mentorship structure, namely by Marc Oriol Hilari, a lecturer in the specialization and Sergio Alejandro Lobig Machado, an Senior Cloud Engineer at FSP.

1.1 Introduction

Organizations tend to hold company events which require its employees to assist. Travel to these events inevitably leads to various expenditures incurred by employees, such as flights, accommodation, and meals. As long as these expenses are business-related and not for personal use, companies typically offer a reimbursement option.

The mxis is a common scenario within FSP's Barcelona team. FSP is a UK-founded private company and a leader in enterprise-level digital evolution and cybersecurity. FSP has successfully expanded its operations, establishing Barcelona as one of its several office locations. As expected, the Barcelona-based team frequently travels to the UK for corporate events, meetings and collaborative work, often struggling with high expenditures along the way.

The current method for managing these expense claims relies on a manual and document-based workflow. To do that, Barcelona's team needs to gather all tickets, record them in Excel spreadsheets, and email them to the Finance Department for processing and approval.

1.2 Key Concepts

The goal of this thesis is to apply and demonstrate the knowledge and competencies acquired throughout the author's Computer Engineering Degree. As a result, this document may contain jargon that could be challenging for readers without a background in this field. To facilitate the reading of this thesis, the definition of technical terms of this documentation is listed below and can be consulted as needed.

App /Application

A software solution designed to facilitate one or many tasks.

Software

Software is a collection of programs, data, and instructions that tells the computer or device to perform specific tasks[7].

Full-stack

Application composed of two parts: Frontend and Backend. Frontend handles the visual and interactive part of the application, while Backend manages its internal server logic.

Feature Specific ability that an app offers to its end-users and adds value to the application[12].

Client

The business stakeholder who requests and funds the development of the software application.

End-user

The individuals who will actively use the application.

UI (User Interface)

The visual part of the application with which the end-user interacts to use it.

UX (User Experience)

The overall feeling and satisfaction a user has while using the application[16].

Final product

The application that is delivered at the end of the project.

Pipeline

A sequence of automated tasks executed when a certain action is triggered[2].

DB (Database)

An organized system for storing and managing the app's essential data[13].

Cloud Computing Platform

A virtual service that offers virtual resources that were traditionally provided physically.

Dashboard

A method to display key data visually in the user interface.

User-friendly

An easy-to-use interface.

Page

Every distinct screen of the app's UI represents a different page.

Clean Code

Programming code that is clear, readable and easy to edit[4].

Reusable Code

Piece of code reusable across multiple contexts to avoid redundancy and reduce developer effort.

Maintainable Code

A code that is easy to understand and maintain as the project grows.

1.3 Problems to Solve

The existing hand-operated expense reporting system in FSP introduces significant inefficiencies. Submitting expense claims is time-consuming for employees, who must enter detailed information from receipts and whose goal is to get the exact reimbursement as soon as possible. At the same time, it becomes labour-intensive work for the Finance Department, which must verify and process refunds using outdated spreadsheets.

From FSP's perspective, this manual data entry and information submission is prone to human error—not only increasing the risk of inaccuracies in financial reports but resulting in expenses being submitted to the wrong person as well. Alternatively, the time invested in this over-consuming activity could be considerably reduced, enabling highly skilled employees to focus on activities that deliver higher value to the company.

1.4 Proposed Solution

FSP clearly needs a more intelligent, automated expense management strategy to prevent employee frustration and guarantee accurate reimbursement of business expenses. This project aims to develop an automated expense reporting web-app allowing employees to manage, report, and submit expenses seamlessly on a single platform. The app digitizes the full workflow—from effortless receipt uploads to AI-powered data extraction and efficient verification and approval processes.

The primary goal of this application is to simplify the previous expenses management workflow, optimize in terms of time and effort for all FSP parties, minimize human error, and ensure the information is transmitted securely and efficiently to the appropriate approver.

1.5 Stakeholders

This section identifies all parties involved in this project, technically recognized as stakeholders. In other words, anyone who can affect the product's final outcome is included in this group.

Product Owner

The Product Owner is responsible for understanding the client's interests and communicating them effectively to the app development team. Their principal role is to maximize the value of the final product from the client's point of view. They define the product's vision and requirements, aligning the app's features with the client's interest throughout the app development lifecycle. In this case Sergio Alejandro Lobig Machado performs as Product Owner, as is the only representative of FSP in this project.

Scrum Master

The Scrum Master's main goal is to lead the whole application development process by following Scrum principles and the Agile methodology (both concepts will be explained in more detail in Section 1.4.1). The author of this thesis assumes this role.

Thesis Director

The Bachelor Thesis Director oversees the academic aspects of this project, prioritizing the successful fulfillment of all degree requirements. Marc Oriol Hilari fills this role, consistently monitoring progress to verify that all required technical and transversal competencies are achieved and demonstrated at the end of this project.

Development Team

The app development team is responsible for implementing the entire application—from initial pipelines and database design and technology stack decisions to final testing and deployment of the product. At this moment, only the thesis author takes charge of this role. However, it does not preclude the possibility of extending and upgrading this project in the future, which would entail adding new developers to the team.

End-users

This application addresses FSP's Barcelona Office Team needs, as a consequence, they become the primary beneficiary of the product and are the application's end-users.

2 Justification

This section presents a market analysis of existing applications considered prior to developing the new application , while demonstrating this project's necessity.

2.1 Market Analysis

The market already offers millions of available expenses management applications. However, in this section only the most recognized and competent examples are pointed out. Analyzing today's market solutions helps identify market gaps while leading the ongoing application to meet industry standards and prioritize the most crucial features.

Evidently, the following competent platforms mentioned are assumed to include the basic features of a simplified expense management process. This includes centralized expense data, digital expense management and reporting, streamlined information submissions, and secure login access through the company's employee account.

To delve deeper into the main differences between available potential market offerings, consider Table 1, which shows a comparative overview of these platforms. Green cells indicate features that the application offers, while red cells indicate features that it does not. In the last row, the cost of each example is presented in detail.

Feature	SAP Concur	Zoho Expense	Expensify	Payhawk
AI-powered data extraction				
OCR technology				
Intuitive UI				
Limit employee's spending				
Approval processes customized				
Global reimbursements				
Virtual cards				
Automated VAT extraction				
Cost	High	Subscription-based	Cost-per-use	Subscription-based

Table 1: Comparative Analysis of Leading Expense Management Solutions (own creation)

As demonstrated in this table, all options share two common features: AI-powered data extraction and OCR technology. Both are indispensable for automated expense management. The AI model serves as an automatic extraction tool for selecting the relevant information from receipts, while the OCR technology automatically converts a text-content image into a machine-readable text[1].

In this project, incorporating an AI model is essential to simplify expense data extraction. The final product will include this tool, as discussed previously in Section 1.1.4. Nevertheless, since it is only a web-app (not a mobile-app), it is not considered to include the possibility to allow end-users to upload receipts via a camera, integrating OCR technology is unnecessary and so, will not be included in the final implementation.

However, going back to Table 1, there are evident differences in target markets and secondary features. For instance, SAP Concur dominates the large enterprise sector[5] but is criticized for its non-intuitive user interface and high-cost plans [3]. In contrast, Zoho Expense and Expensify target small businesses with more user-friendly interfaces, more affordable pricing, and additional features not included in SAP Concur[8][9]. The last example introduces Payhawk, which includes a new feature that has not been offered by the previous platforms yet: automated receipts VAT extraction[14].

Despite their strengths, this analysis clearly reveals common areas for improvement. An interesting feature for FSP and similar companies would be real-time email notifica-

tions—such as alerting employees when an expense report has been rejected or approved, and notifying approvers when a report is submitted and pending review.

2.2 Project Justification

At this point, the development of this application can be justified on several grounds, while aligning the author's academic objectives with the FSP's business needs.

First, FSP clearly needs a better solution. Although numerous excellent options exist, none of them align perfectly with FSP's specific context, not to mention the high costs and the excessive features that may not be useful for FSP. This project develops a tailored, efficient, and cost-effective platform to address the identified time-consuming and error-prone processes. In this way, FSP can manage employee expenses without relying on external platforms while reducing costs, as maintaining the in-house application is significantly more economical than paying for existing solutions.

Second, from a personal perspective, this project provides an ideal context to apply the author's prior Software Engineering knowledge and skills to a real-world problem. It covers all project stages—from requirements analysis and architecture design through full app development lifecycle, full-stack implementation, testing, and documentation. Therefore, it creates an optimal scenario for connecting the author's academic background with meaningful practical experience.

Last but not least, this is a perfect opportunity to demonstrate the author's technical competencies, problem-solving abilities, and project management skills acquired during the degree program. It also is the author's first chance to develop an full-stack application independently, which also introduces some obstacles that will be discussed in Section 3.3.

3 Scope

With the context and justification established, it is time to define the objectives of this project and the requirements that the final product must satisfy. It is also important to consider potential obstacles and risks before proceeding, so these will be discussed as well.

3.1 Objectives

The main objective of this project is to develop an expense reporting tool for FSP that addresses the company's priorities while tailoring web-app features to its specific context. To achieve this goal, it will be broken down into smaller and simpler sub-objectives to make its scope easier to understand while facilitating the evaluation of its completion. The specific sub-objectives are outlined below:

SG1. Apply author's previous academic knowledge to solve a real-world problem through a software solution

SG2. Design the technology stack and the database system of the application

SG3. Develop a full-stack web application

SG4. Design a scalable expense upload pipeline to extract relevant data directly using an external AI model

SG5. Extend programming skills and learn new programming concepts.

SG6. Use a new cloud computing platform to integrate external application resources.

SG7. Maintain a clean, reusable, maintainable, and high quality code throughout the application development

3.2 Requirements

In the field of Software Engineering, requirements are categorized into two main groups: functional and non-functional. The former defines what features the application provides, while the latter describes how those features are delivered. As a result, the requirements of this project are defined following this structure[10].

Functional Requirements

These are the functional requirements of the final product:

- FR1: The application must centralize expenses data in a single platform.
- FR2: The application must enable effortless uploads and drag-and-drop of receipts.
- FR3: The application must automate the extraction of relevant data from uploaded receipts using an AI model.
- FR4: The application must allow users to review and edit the automatically extracted expenses data.
- FR5: The application must provide a search bar and filters for uploaded expenses in order to search by name and filter by category.
- FR6: The application must allow users to gather multiple expenses in a single report.
- FR7: The application must allow users to complete and edit the report information.
- FR8: The application must provide a search bar to find reports by name.
- FR9: The application must enable simple submission of expenses report to the corresponding approver.
- FR10: The application must allow users to login through the company's employee account.
- FR11: The application must display a user account page showing the total budget of all submitted reports, categorized as:
 - Pending approval or rejection budget
 - Approved budget
 - Rejected budget
- FR12: The application must send email notifications when a submitted report is approved or rejected.
- FR13: The application must implement role-based access control for employees and admins:

- **Employees** can use all previously listed features.
- **Admins** can use all employee features plus the additional features listed below.
- FR14: The application must allow admin users to approve or reject submitted expenses reports from employees under their supervision.
- FR15: The application must display the history of all reviewed reports (approved or rejected).
- FR16: The application must enable admins to set the expenses budget limit for those employees under their supervision.
- FR17: The application must send email notifications to users when an employee under their supervision submits an expense report pending approval or rejection.
- FR18: The application must provide a monthly summary of the total submitted budget by employees under their supervision.

Non-Functional Requirements

These are the non-functional requirements of the final product:

- **NFR1: The application must provide real-time dashboards for tracking receipt status to improve UX.**

Users should be able to check the current status of their reports submissions.

- **NFR2: The application must offer an interactive, responsive, reactive, and user-friendly UI.**

The UI must be interactive—every user action should trigger a response. The UI must also be responsive, meaning it adapts automatically to any device or screen size. Additionally, the UI must be reactive, this behaviour is needed as the UI and the internal server are two separate components that communicate with each other. Updating the UI without repeatedly calling the server is an important practice to prevent server overloaded[6]. Finally, the UI should be intuitive and easy to use, a new user should not need more than five minutes to learn how to navigate the app.

- **NFR3: The application must ensure high application performance and availability.**

High performance delivers a fast and smooth user experience. Any action within the application should not take longer than four seconds to complete. The application should also maintain high availability, meaning users can access it 24 hours a day, 365 days a year.

- **NFR4: The application must support high scalability as the company grows.**

The app should remain stable and responsive even when the number of employees using the app increases significantly.

- **NFR5: The application must guarantee data storage compliance with GDPR.**

FSP is a UK-based company, therefore, the expenses application should also comply with the General Data Protection Regulation(GDPR).

- **NFR6: The application must ensure secure sensitive information storage, as well as secure information access and submission.**

Any sensitive data stored by the web-app should not be leaked. Additionally, the app should also control who can access that information, and also protect it from external parties during the submission.

- **NFR7: The application must enable efficient and intuitive expense management process.**

Employees should find the new expense managing process much faster and easier than the traditional one.

- **NFR8: The application must ensure efficient processing for approval and rejection of expenses reports.**

Approvers should also be able to review and manage submitted expenses more quickly and efficiently.

- **NFR9: The application must provide an integration-friendly architecture.**

Meeting this requirement facilitates the integration of new AI models in the future.

3.3 Obstacles and Risks

Evaluating possible obstacles will lead to a more accurate project time estimation. Furthermore, identifying the risks that could impact the application's development is crucial. The following section outlines these potential obstacles and risks in detail.

Coding Bugs

As the application development progresses, it is inevitable that certain coding bugs will emerge, requiring additional time to fix. Since encountering such risks is highly probable, minor delays in the project are to be expected.

Tight Project Deadlines

The project timeline is constrained by the thesis deadline. As a consequence, the final product may not fully meet all the defined requirements. However, this represents a meaningful challenge for the author, as prioritizing specific application features is essential to deliver a high-value solution for FSP.

Lack of Experience in Certain Technologies

The two previous situations are quite common in real-world engineering practice, and this one is no exception. Exploring and applying new technologies is a constant challenge to a software engineer's daily work. However, the time dedicated to learning and becoming familiar with these technologies should be taken into account in the project timeline.

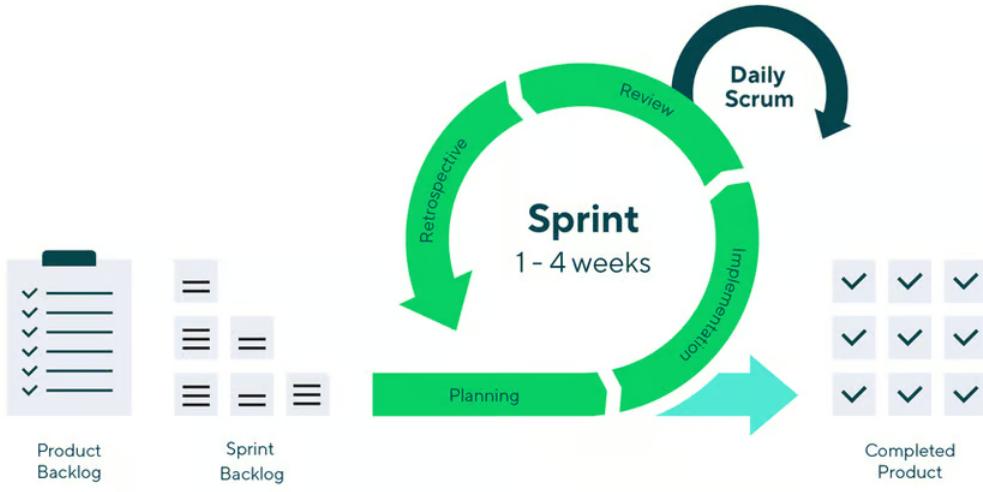


Figure 1: Agile Methodology Diagram (Wrike). Source:[\[11\]](#)

4 Methodology

Since this project involves the development of a software product, it follows a specific strategy commonly adopted by most companies in this field today. In addition, the tools selected for use and the validation strategies are also defined.

This project applies the Agile methodology, specifically based on the Scrum framework. Agile is a modern approach that has revolutionized traditional software development methods. It is an iterative and incremental process composed of short development cycles, with the main objective of continuously improving the product's value through each iteration. This strategy enables developers to adapt quickly to changes, monitor project progress effectively, increase productivity, prioritize essential features, prevent last-minute problems, meet deadlines, and maintain clear communication—ensuring that the final product aligns with the client's requirements from the very beginning of the project[\[15\]](#).

The key roles of Agile methodology are the Scrum Master, Product Owner, and Development Team. These roles were previously introduced in Section 1.6. At this point, it is not necessary to go into detail about the entire methodology workflow. However, there are key terms that will appear throughout this documentation and are important to understand.

- **Sprint:** Each Agile cycle is considered a sprint, which is repeated as many times as needed throughout the project. A sprint is composed of four phases (see Figure 1 for a context). In this project a sprint lasts two weeks.
- **Product Backlog:** A list of features, bugs or improvement that serves as a guide to the team to track the product's evolution.
- **Daily Scrum:** Short daily meetings with the Scrum Master where each development team member shows the work completed in the previous day and outlines the tasks planned for the current day. In this way, each team member has clearly

defined tasks to complete, with the option to reassign them if the workload is not balanced among all members.

- **Sprint Planning:** The initial phase of a sprint, during which the development team defines the objectives and the tasks to be done in that sprint.
- **Sprint Review:** The sprint phase during which the development team verifies whether all defined tasks have been completed and adjusts the workload for the next sprint if necessary.
- **Sprint Retrospective:** The final phase of a sprint, during which the development team discusses the challenges encountered, the solutions found as long as areas to improve in the next sprint.

4.1 Monitoring Tools

The use of monitoring tools makes it easier for the project to follow the Agile methodology. Moreover, most of the solutions are essential in software development projects, as they help prevent the loss of the work if any system or device fails unexpectedly during the development process.

Git

A version control system which allows multiple developers to work on the same project and enables a single user to develop several features at the same time. These capabilities make Git an essential tool for software product development.

In Git, a repository is the location where the application's source code is stored. It serves as a project space where developers submit their code, allowing changes to be tracked throughout the development process.

Azure DevOps

Azure is a cloud resource that offers multiple tools. This project makes use of two of them: the Git repository and the backlog dashboard.

In the backlog, four statuses are assigned to tasks:

- **New:** A task that has been defined but it is pending development, or a bug that is pending resolution.
- **Active:** A task that is currently in development.
- **Resolved:** A bug that has been fixed.
- **Closed:** A task that has been developed, integrated and tested.

Microsoft Teams

Communication tool used in FSP. It is used primarily by the Scrum Master and the author used primarily for chats and meetings.

Gmail

An email platform used for communication between the author and the thesis director.

4.2 Validation Methods

During each Agile sprint, the product is validated in every daily Scrum meeting and every sprint review as well. Agile facilitates the validation of this project, as it is reviewed continuously, keeping every new feature or change under control. Regarding technical validation, the application must pass several tests to ensure that its behaviour matches the discussed requirements. Finally, the thesis director will supervise the documentation throughout the project to ensure it meets the standards of a proper bachelor's thesis.

Bibliography

- [1] Adobe. *OCR technology*. URL: <https://www.adobe.com/acrobat/guides/what-is-ocr.html> (visited on 2026).
- [2] BrowserStack. *Automation Pipeline and CI/CD: A Guide to Testing Best Practices*. URL: <https://www.browserstack.com/guide/automation-pipeline#:~:text=An%20automation%20pipeline%20is%20a%20connected%20set%20of,previous%20one.%20CI%2FCD%20pipelines%20are%20a%20common%20example.> (visited on 2025).
- [3] Capterra. *SAP Concur vs Zoho Expense: Features and Cost Comparison 2025 — Capterra*. URL: <https://www.capterra.com/compare/380-142384/Concur-Expense-vs-Zoho-Expense> (visited on 2026).
- [4] Codacy. *What Is Clean Code? A Guide to Principles and Best Practices*. URL: <https://blog.codacy.com/what-is-clean-code> (visited on 2025).
- [5] S. Concur. *Concur Expense — SAP Concur España*. URL: <https://www.concur.es/products/concur-expense> (visited on 2026).
- [6] S. M. Corp. *Interactive, Responsive and Reactive - What's the difference?* URL: <https://sosmediacorp.com/interactive-responsive-and-reactive-whats-the-difference/> (visited on 2016).
- [7] Educative. *Computer science 101: Hardware vs software components*. URL: <https://www.educative.io/blog/hardware-vs-software-components-computer> (visited on 2024).
- [8] Z. Expense. *Travel and expense (TE) management - Zoho Expense*. URL: <https://www.zoho.com/us/expense/> (visited on 2026).
- [9] Expensify. *The easiest way to do your expenses — Expensify*. URL: <https://www.expensify.com/> (visited on 2026).
- [10] GeeksforGeeks. *Functional and Non-Functional Requirements*. URL: <https://www.geeksforgeeks.org/software-engineering/functional-vs-non-functional-requirements/> (visited on 2025).
- [11] A. Gurnov. *What is Agile methodology in Project Management?* URL: <https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/> (visited on 2025).
- [12] Harness. *Understand Software Features And Their Impact On Development*. URL: <https://www.harness.io/harness-devops-academy/mastering-software-features-tips-measurement> (visited on 2026).
- [13] Microsoft. *What Is a Database?* URL: <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-are-databases/#:~:text=A%20database%20is%20a%20structured%20system%20for%20storing%2C%20it%20easy%20to%20retrieve%2C%20manage%2C%20and%20analyze%20information.>?msclkid=3fb076b7d42168e23fd86320d566695a (visited on 2026).

- [14] Payhawk. *La gestión financiera inteligente, a tu alcance hoy — Payhawk*. URL: <https://payhawk.com/es> (visited on 2026).
- [15] E. University. *Agile and Scrum: differences and applications in each framework*. URL: <https://universidadeuropea.com/blog/agile-scrum/> (visited on 2025).
- [16] Userlytics. *User Experience (UX)*. URL: <https://www.userlytics.com/resources/glossary/user-experience-ux/#:~:text=User%20Experience%20%28UX%29%20is%20the%20overall%20feeling%20a,service%2C%20encompassing%20aspects%20of%20usability%2C%20accessibility%2C%20and%20delight>. (visited on 2025).