SOFTWARE PLANNING AND UML

for

ESPOLTEL HIRING MANAGER

Version 1.0

Jeremy Rodrigo Poveda Gorotiza
José David Ramos Rios
Diego Fernando Flores Rengifo
Ariana Valentina Palacios Saenz
Alex Javier Vizuete Pereira

Submitted to: Francisco Ramirez

January 12, 2025

Revision History

Name	Date	Reason for Changes	Version
Team 3	2025-1-10	Initial draft	1.0

Contents

List of Figures

List of Tables

1 Introduction

1.1 Summary

This document presents a comprehensive framework for the design, planning, and execution of the ESPOLTEL HIRING MANAGER system. This product integrates a robust risk management strategy, a detailed project execution timeline, and a structured Sprint Backlog plan. Through the inclusion of Unified Modeling Language (UML) diagrams, we provide a thorough representation of both the static and behavioral logic of the system, ensuring that the architecture adheres to SOLID principles and eliminates implementation inefficiencies.

Our primary objective is to meticulously define the planning and breakdown of the system's static structure, logical flow, behavioral processes, implementation strategies, and activity sequences. These components collectively support the realization of a user-centric, scalable, and maintainable product.

1.2 Key features and Objetives

The ESPOLTEL HIRING MANAGER product is designed to streamline and enhance the recruitment process, leveraging a combination of web and mobile modules for maximum efficiency. Key objectives include:

1. **Risk Mitigation:** Developing a proactive risk management plan to address potential challenges in implementation and deployment.
2. **Comprehensive Planning:** Structuring the project execution into manageable phases using Agile methodologies.
3. **System Design:** Crafting static and behavioral UML diagrams to visualize the architecture, interactions, and workflows.
4. **Adherence to SOLID Principles:** Ensuring maintainability and scalability by avoiding anti-patterns and promoting clean code practices.

2 Risk management, product and sprint backlogs and scheduling

2.1 Risk management

In this section, we will identify, quantify, and classify the various risks that may arise during the software development process. Additionally, we will provide a detailed assessment of the likelihood of occurrence, the potential impact of each risk, and the corresponding protocols to be followed in the event they materialize.

Description	**Probability Range**
Not Probable: The event is highly unlikely to occur.	0% - 20%
Low Probability: The event is unlikely but possible.	21% - 40%
Moderate Probability: The event has an even chance of occurring.	41% - 60%
High Probability: The event is likely to occur.	61% - 80%
Very High Probability: The event is almost certain to occur.	81% - 100%

Table 2.1: Probability of Occurrence

Impact Level	**Description**
Low Impact	Minimal effect on the project. No significant changes required.
Moderate Impact	Some delays or adjustments needed but manageable within the team.
High Impact	Significant disruptions, requiring immediate attention and resource allocation.
Critical Impact	Severe consequences on project delivery, with major delays or failure possible.

Table 2.2: Impact Levels

The following table outlines the identified risks associated with the project, including their probability of occurrence, potential impact, and the corresponding action protocol.

Id	Name	Probability	Impact	Action Protocol
001	Changes in requirements after development completion	High Probability	High Impact	Establish a communication protocol to clarify that no new requirements will be accepted after the design phase is finalized.
002	Discovery of implicit requirements not considered in the design	Very High Probability	High Impact	Accept and address the risk by updating the design and implementing the missing requirements.
003	Need for developer training	High Probability	High Impact	Provide immediate training on the required frameworks to minimize delays and ensure smooth development progress.
004	Difficulty understanding prior implementation, causing delays	Low Probability	Critical Impact	Reduce the probability by consulting previous implementers to gain insights into the system before development begins.
005	Schedule misalignment affecting task timelines	Not Probable	High Impact	Mitigate the risk by redistributing tasks and holding regular progress meetings to stay on track.
006	Performance drop due to prior monolithic architecture	Low Probability	High Impact	Accept the risk, inform the client, and propose alternative solutions to improve performance.
007	Database schema not designed for extensions	Low Probability	Moderate Impact	Accept the risk and adapt the existing schema to accommodate the new requirements.
008	Insufficient documentation provided by the client	High Probability	Critical Impact	Reduce probability by maintaining active communication with the client to obtain necessary documentation.

Table 2.3: Risk Assessment and Action Protocols

2.2 Product backlog

2.3 Sprint backlog

2.4 Scheduling

3 Individual Contributions

Student's Names	**Contributions**
Jeremy Rodrigo Poveda Gorotiza	Project Scope, Introduction, User Stories, Creation of GitHub Repository, prototype: web application for director and managers
Diego Fernando Flores Rengifo	Non functional requirements both Web and Mobile Application, prototype in figma: Authentification module and Applicants Platform
José David Ramos Rios	Product Overview, Product Features, Module Featuring: Mobile App, First Preview of Module Featuring: Web Application, and prototype in figma of Mobile App
Ariana Valentina Palacios Saenz	Revision, User Stories, and prototyping flows and module integration
Alex Javier Vizuete Pereira	Web Application Modules Breakdown, Mobile Application Modules Breakdown, prototype in figma: Applicants Platform, screens, and flow of application process

Table 3.1: Responsibilities of each member of team 3

4 Appendix

4.1 Appendix A: Github Repository

The versioning of the project prototype has been managed using Github. You can find it through the following link ESPOLTEL's versioning project:
Repository link

4.2 Appendix B: Commitment Agreement

4.3 Appendix C: Evidence of requirements gathering

Initial interview for requirements gathering with the client

Template Questions for the Interview

1. Are "Human Talent" and "Human Resources" distinct roles within the company?
If yes, is the Human Resources area responsible for generating documents such as contracts and confidentiality agreements?

What we understand is as follows:
Human Talent:

- Requests documents and information from applicants.
- Verifies that applicants meet the position requirements.
- Sends the data of candidates who meet the requirements to Human Resources.

Human Resources:

- Generates documents such as contracts and confidentiality agreements.
- Sends the generated contracts or agreements to the applicant.
- Verifies the applicant's signature.
- Sends the documents to managers for their signatures.

2. Must the contracts and confidentiality agreements be signed not only by the managers and applicants but also by the project director?

3. In addition to requesting basic information such as names, surnames, cell phone numbers, etc., should the Human Talent area request specific documents according to the profile, such as copies of the ID, voting card, etc.?

4. Who is responsible for entering the templates of the contracts or confidentiality agreements into the system: Human Talent or Human Resources?

5. Should these templates be created directly within the system? If yes, would the data be in plain text, such as names, surnames, ID numbers, and the positions for electronic signatures (of managers, applicants, and possibly project directors)?

6. Would the stages of the applicant acceptance process be as follows?

- Application for a profile by submitting information (plain text data and documents). Waiting for a response from Human Talent to verify if the applicant meets the requirements. If the applicant meets the requirements, waiting for the contract and confidentiality agreements to sign, generated by Human Resources.
- Signing the documents.
- Waiting for signature validation by Human Resources.
- Waiting for signatures from managers and directors.
- Confirmation of participation in the project.