

| **TITLE: Project Plan document for** Mini Project |
| --- |

**AIM:** To learn and understand the way of developing the software by classical methods of software engg. planning and monitoring of the project using tools and prepare a document for the same by using the concept of software engineering **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected OUTCOME of Experiment:**

Analyse the software requirements and Model the defined problem with the help of UML diagram

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

1. Roger Pressman, Software Engineering: A practitioners Approach, McGraq Hill, 2010 ,6th edition

2. Ian Somerville , Software Engineering , Addison Wesley,2011,9th edition

1. http://en.wikipedia.org/wiki/Software\_requirements\_specification

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Software Project Management Plan**

**for**

***Le Miele Mists***

***Rohit Deshpande***

***1st October, 2024***

| **Version** | **Release Date** | **Responsible Party** | **Major Changes** |
| --- | --- | --- | --- |
| 0.1 | 10th October, 2024 | Rohit Deshpande | Initial Document Release for Comment |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Table of Contents

1. Introduction  
   1.1 Project Overview  
   1.2 Project Deliverables  
   1.3 Evolution of the SPMP  
   1.4 Reference Materials  
   1.5 Definitions and Acronyms
2. Project Organization  
   2.1 Process Model  
   2.2 Organizational Structure  
   2.3 Organizational Interfaces  
   2.4 Project Responsibilities
3. Managerial Process  
   3.1 Management Objectives and Priorities  
   3.2 Assumptions, Dependencies, and Constraints  
   3.3 Risk Management  
   3.4 Monitoring and Controlling Mechanisms  
   3.5 Staffing Approach
4. Technical Process  
   4.1 Methods, Tools, and Techniques  
   4.2 Software Documentation  
   4.3 User Documentation  
   4.4 Project Support Functions
5. Work Packages, Schedule, and Budget  
   5.1 Work Packages  
   5.2 Dependencies  
   5.3 Resource Requirements  
   5.4 Budget and Resource Allocation  
   5.5 Schedule
6. Additional Components  
   6.1 Index  
   6.2 Appendices

**1.  Introduction**

This section provides a comprehensive overview of the **Le Miele Mists** project, focusing on its key aspects and foundational components.

### 1.1 Project Overview

**Le Miele Mists** is an e-commerce platform specifically designed to cater to luxury perfume buyers. The platform aims to create a personalized and exclusive online shopping experience by integrating several advanced features, such as:

* **Personalized Perfume Recommendations:** The platform will recommend perfumes to users based on their previous purchases, preferences, and reviews.
* **Product Management and Inventory System:** Dynamic inventory tracking and product catalog management to ensure stock levels are up-to-date.
* **Secure Payment Gateways:** Integration with various payment methods, including credit cards, wallets, and UPI, ensuring safe and encrypted transactions.
* **Delivery Tracking:** Integration with a third-party delivery tracking system for real-time shipment updates.
* **Promotions and Discounts:** Users can avail of discounts and promotional offers, with tracking available for repeat customers.

The project adheres to a budget between ₹180,000 and ₹200,000, and it will span across three development phases:

* **Phase 1:** Prototype (Beta Launch)
* **Phase 2:** Full Website Release
* **Phase 3:** Post-Launch Feature Enhancements

### 1.2 Project Deliverables

The key deliverables for **Le Miele Mists** include:

1. **E-commerce Platform**: A mobile-responsive, feature-rich website with secure payments and user-friendly interfaces.
2. **Personalized Recommendation Engine**: An algorithm recommending products based on user preferences.
3. **Inventory Management System**: A backend to monitor stock levels, reorder products, and manage suppliers.
4. **Payment Gateway Integration**: Secure integration with popular payment methods.
5. **User Reviews and Ratings**: User-generated content and feedback for future product improvement and enhanced shopping experiences.
6. **Shipment and Delivery Tracking System**: Automated integration for tracking customer orders with real-time updates.
7. **Post-Launch Support**: Ongoing support for bug fixing, feature improvements, and performance optimization post-launch.

### 1.3 Evolution of the SPMP

The SPMP will undergo revisions as the project progresses. It will be updated at key milestones, such as:

* **After Phase 1:** Beta feedback and any scope changes.
* **After Phase 2:** Full-scale release updates, including real-world challenges.
* **After Phase 3:** Post-launch updates addressing feature requests and optimizations.

### 1.4 Reference Materials

The following reference materials are used to guide the project’s development:

* **Agile Development for E-commerce Platforms**, John Wiley & Sons, 2023
* **UX Best Practices for E-commerce** by Nielsen Norman Group, 2022
* **Payment Gateway Security Guidelines**, PCI DSS Compliance Documentation, 2023

### 1.5 Definitions and Acronyms

* **SPMP**: Software Project Management Plan
* **UI/UX**: User Interface/User Experience
* **API**: Application Programming Interface
* **SRS**: Software Requirements Specification
* **LMM**: Le Miele Mists

Here’s a more detailed version of the **Software Project Management Plan (SPMP)** for **Le Miele Mists**:

# Software Project Management Plan

**Project Name:** Le Miele Mists  
**Author:** Rohit  
**Date:** October 21, 2024  
**Version:** 0.2  
**Release Date:** TBD  
**Responsible Party:** Rohit  
**Major Changes:** Initial Detailed Document Release for Review

## Table of Contents

* **1. Introduction**
  + 1.1 Project Overview
  + 1.2 Project Deliverables
  + 1.3 Evolution of the SPMP
  + 1.4 Reference Materials
  + 1.5 Definitions and Acronyms
* **2. Project Organization**
  + 2.1 Process Model
  + 2.2 Organizational Structure
  + 2.3 Organizational Interfaces
  + 2.4 Project Responsibilities
* **3. Managerial Process**
  + 3.1 Management Objectives and Priorities
  + 3.2 Assumptions, Dependencies, and Constraints
  + 3.3 Risk Management
  + 3.4 Monitoring and Controlling Mechanisms
  + 3.5 Staffing Approach
* **4. Technical Process**
  + 4.1 Methods, Tools, and Techniques
  + 4.2 Software Documentation
  + 4.3 User Documentation
  + 4.4 Project Support Functions
* **5. Work Packages, Schedule, and Budget**
  + 5.1 Work Packages
  + 5.2 Dependencies
  + 5.3 Resource Requirements
  + 5.4 Budget and Resource Allocation
  + 5.5 Schedule
* **6. Additional Components**
  + 6.1 Index
  + 6.2 Appendices

## 1. Introduction

This section provides a comprehensive overview of the **Le Miele Mists** project, focusing on its key aspects and foundational components.

### 1.1 Project Overview

**Le Miele Mists** is an e-commerce platform specifically designed to cater to luxury perfume buyers. The platform aims to create a personalized and exclusive online shopping experience by integrating several advanced features, such as:

* **Personalized Perfume Recommendations:** The platform will recommend perfumes to users based on their previous purchases, preferences, and reviews.
* **Product Management and Inventory System:** Dynamic inventory tracking and product catalog management to ensure stock levels are up-to-date.
* **Secure Payment Gateways:** Integration with various payment methods, including credit cards, wallets, and UPI, ensuring safe and encrypted transactions.
* **Delivery Tracking:** Integration with a third-party delivery tracking system for real-time shipment updates.
* **Promotions and Discounts:** Users can avail of discounts and promotional offers, with tracking available for repeat customers.

The project adheres to a budget between ₹180,000 and ₹200,000, and it will span across three development phases:

* **Phase 1:** Prototype (Beta Launch)
* **Phase 2:** Full Website Release
* **Phase 3:** Post-Launch Feature Enhancements

### 1.2 Project Deliverables

The key deliverables for **Le Miele Mists** include:

1. **E-commerce Platform**: A mobile-responsive, feature-rich website with secure payments and user-friendly interfaces.
2. **Personalized Recommendation Engine**: An algorithm recommending products based on user preferences.
3. **Inventory Management System**: A backend to monitor stock levels, reorder products, and manage suppliers.
4. **Payment Gateway Integration**: Secure integration with popular payment methods.
5. **User Reviews and Ratings**: User-generated content and feedback for future product improvement and enhanced shopping experiences.
6. **Shipment and Delivery Tracking System**: Automated integration for tracking customer orders with real-time updates.
7. **Post-Launch Support**: Ongoing support for bug fixing, feature improvements, and performance optimization post-launch.

### 1.3 Evolution of the SPMP

The SPMP will undergo revisions as the project progresses. It will be updated at key milestones, such as:

* **After Phase 1:** Beta feedback and any scope changes.
* **After Phase 2:** Full-scale release updates, including real-world challenges.
* **After Phase 3:** Post-launch updates addressing feature requests and optimizations.

### 1.4 Reference Materials

The following reference materials are used to guide the project’s development:

* **Agile Development for E-commerce Platforms**, John Wiley & Sons, 2023
* **UX Best Practices for E-commerce** by Nielsen Norman Group, 2022
* **Payment Gateway Security Guidelines**, PCI DSS Compliance Documentation, 2023

### 1.5 Definitions and Acronyms

* **SPMP**: Software Project Management Plan
* **UI/UX**: User Interface/User Experience
* **API**: Application Programming Interface
* **SRS**: Software Requirements Specification
* **LMM**: Le Miele Mists

## 2. Project Organization

### 2.1 Process Model

The project will follow an **Agile Scrum** methodology to allow flexibility, adaptability, and continuous feedback loops. Sprint durations will be set for two weeks, and at the end of each sprint, the team will deliver increments in line with stakeholder priorities. The backlog will evolve with new tasks and features based on customer feedback during prototype and beta launches.

### 2.2 Organizational Structure

The organization consists of the following stakeholders:

1. **Project Manager (PM):** Rohit Deshpande
   * Ensures the project runs on time and within budget.
   * Coordinates between stakeholders and development teams.
2. **Technical Lead (TL):** Kunj Gohil
   * Oversees the technical aspects of the project, including architecture, technology stack, and coding practices.
3. **UI/UX Designer:** Jeet Gada
   * Responsible for the user interface and overall user experience.
4. **Frontend Developers:** Rohit Deshpande, Rishil Desai
   * Builds the customer-facing features and interfaces.
5. **Backend Developers:** Kunj Gohil
   * Develops the backend functionalities, including database design, API development, and payment integration.
6. **Quality Assurance (QA):** Jeet Gada
   * Tests the software for bugs and ensures it meets quality standards.

### 2.3 Organizational Interfaces

Key communication and decision-making interfaces:

| **Stakeholder** | **Interface** | **Responsibility** |
| --- | --- | --- |
| Product Owner | Project Manager | Defining scope, key decisions |
| Development Team | Technical Lead | Developing product features |
| QA | Project Manager | Testing, validating product quality |
| UI/UX Designer | Development Team | Ensuring design is implemented correctly |

### 2.4 Project Responsibilities

* **Project Manager:**
  + Manages scope, budget, and timeline.
  + Coordinates between internal and external stakeholders.
  + Manages risk and handles change requests.
* **Technical Lead:**
  + Guides technical decisions and ensures alignment with project objectives.
  + Oversees code reviews, architectural decisions, and deployments.
* **UI/UX Designer:**
  + Focuses on delivering a user-centric design.
  + Coordinates with developers to ensure the design is accurately implemented.
* **QA:**
  + Conducts manual and automated tests.
  + Provides detailed reports of bugs and performance issues.

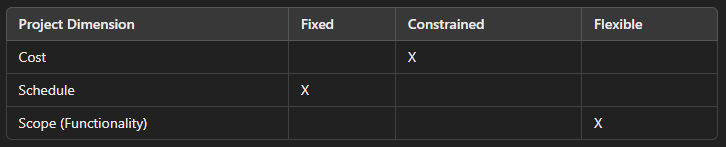
## 3. Managerial Process

### 3.1 Management Objectives and Priorities

The project prioritizes:

* **Delivering a fully functional and scalable platform** within budget and on schedule.
* **Ensuring a seamless and personalized user experience**, focusing on usability, performance, and ease of navigation.
* **Maintaining flexibility** to accommodate feature updates based on user feedback post-launch.

The primary priority matrix is:



### 3.2 Assumptions, Dependencies, and Constraints

* **Assumptions**:
  + Payment gateway approval will proceed smoothly without delays.
  + Availability of skilled team members.
* **Dependencies**:
  + Integration with third-party services, including payment gateways, delivery tracking APIs, and review systems.
* **Constraints**:
  + Budget constraints might limit advanced features during initial phases.
  + Scope changes may occur post-beta launch, affecting schedules.

### 3.3 Risk Management

Key risks and mitigation strategies:

* **Risk 1: Delays due to Third-Party Integration**
  + Mitigation: Start early communication with payment gateways and delivery systems.
* **Risk 2: Scope Creep**
  + Mitigation: Use a formalized change control process with stakeholder approval.
* **Risk 3: Security Vulnerabilities in Payment System**
  + Mitigation: Implement PCI DSS-compliant standards and conduct regular security audits.

### 3.4 Monitoring and Controlling Mechanisms

The project will be monitored using the following methods:

* **Weekly Scrum Meetings**: Review progress and plan next sprints.
* **Burn-Down Charts**: Track completion of tasks during each sprint.
* **Milestone Reviews**: Held at the end of each major phase to evaluate whether project goals are being met.

### 3.5 Staffing Approach

The team will consist of 5 core members:

* **Developers (3):** Two frontend, one backend.
* **UI/UX Designer (1):** Responsible for interface and experience design.
* **QA Specialist (1):** Ensures quality across platforms.

## 4. Technical Process

### 4.1 Technologies and Tools

For the successful development and deployment of the **Le Miele Mists** online perfume store, we will leverage a comprehensive set of technologies and tools, detailed as follows:

**Frontend: React.js**React.js will be our primary framework for building dynamic and interactive user interfaces. Its component-based architecture allows for reusable UI components, facilitating a smooth user experience. The benefits of using React.js include:

* **Virtual DOM:** Enhances performance by minimizing direct manipulation of the DOM.
* **Component Reusability:** Streamlines development and maintenance by allowing components to be reused across the application.
* **Rich Ecosystem:** Access to numerous libraries and tools that complement React, such as React Router for navigation and Redux for state management.

**Backend: Node.js with Express.js**Node.js, coupled with the Express.js framework, will serve as the backbone for our server-side operations. This environment is chosen for its asynchronous, event-driven architecture, which is ideal for I/O-heavy applications. Key features include:

* **High Performance:** Capable of handling numerous simultaneous connections with high throughput.
* **RESTful APIs:** Facilitate communication between the client and server, ensuring efficient data retrieval and manipulation.
* **Middleware Support:** Allows for easy integration of additional functionalities, such as authentication and error handling.

**Database: MongoDB Atlas**We will utilize MongoDB Atlas, a cloud-based NoSQL database service, to manage our data effectively. This choice offers:

* **Scalability:** Seamlessly handle increasing data loads as our user base grows.
* **Flexible Data Models:** Accommodate various data types and structures without the need for a predefined schema.
* **Real-time Data Access:** Enhance user experience through quick data retrieval and updates.

**Payment Integration: Razorpay or Stripe**To ensure secure transactions, we will integrate either Razorpay or Stripe, both of which provide robust payment processing solutions. They offer:

* **Multiple Payment Methods:** Support for various payment modes, including credit/debit cards, UPI, and net banking.
* **Security Compliance:** Adherence to PCI DSS standards, ensuring user payment data is handled securely.
* **Ease of Integration:** Comprehensive documentation and SDKs facilitate quick setup and deployment.

**Version Control: GitHub**We will use GitHub for version control and collaboration among team members. This platform provides:

* **Branching and Merging:** Allows developers to work on features independently before merging them into the main project.
* **Issue Tracking:** Helps in managing project tasks and bug fixes effectively.
* **Collaboration Tools:** Facilitates code reviews and discussions within the team, ensuring high-quality code development.

### 4.2 Documentation Plan

Clear and detailed documentation will be maintained throughout the project to ensure all stakeholders have access to essential information. The documentation will include:

**API Documentation**A comprehensive guide detailing all available API endpoints, including:

* **Endpoint Descriptions:** Clear explanations of what each endpoint does.
* **Request/Response Examples:** Sample requests and responses for better understanding.
* **Authentication Requirements:** Information on how to authenticate requests, including token usage.

**System Architecture Diagrams**Visual representations of the system's architecture will be created to provide a clear understanding of:

* **Component Interactions:** How different parts of the system communicate with each other.
* **Data Flow:** The movement of data through the system, highlighting key processes and databases.
* **Deployment Architecture:** An overview of how the application will be hosted and managed.

**User Guides**To empower end users, we will create user-friendly guides that cover:

* **Navigation Instructions:** Step-by-step processes on how to use various features of the website.
* **Common Tasks:** Tutorials for frequently performed actions, such as searching for products, placing orders, and managing accounts.
* **Troubleshooting Tips:** Guidance on resolving common issues users may encounter while using the website.

### 4.3 User Support

To ensure a positive user experience, we will establish robust support mechanisms, including:

**FAQs and Help Center**A dedicated section on the website will address common questions and concerns. This resource will include:

* **Product Information:** Details about different perfumes, including descriptions, ingredients, and usage.
* **Order and Shipping Queries:** Information regarding order processing, shipping options, and delivery times.
* **Account Management:** Assistance with account creation, password recovery, and privacy settings.

**Customer Support**We will offer comprehensive customer support through:

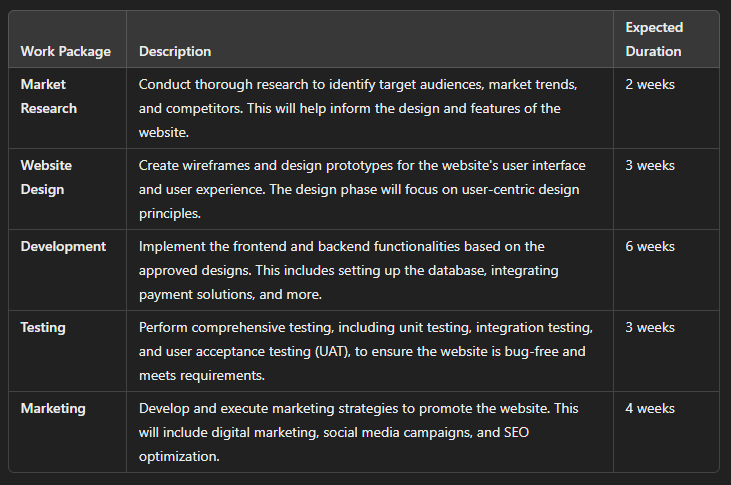
* **Email Support:** Users can reach out for assistance with technical issues or inquiries related to orders.
* **Live Chat Support:** Real-time help for urgent concerns, providing immediate assistance during business hours.
* **Response Time Commitment:** We will establish clear response time commitments to ensure users receive timely help.

### 5. Work Packages, Schedule, and Budget

This section outlines the work packages required for the project, their dependencies, resource requirements, budget allocation, and project schedule.

### 5.1 Work Packages

Work packages represent distinct tasks or activities that contribute to the overall project objectives. The following work packages have been identified:



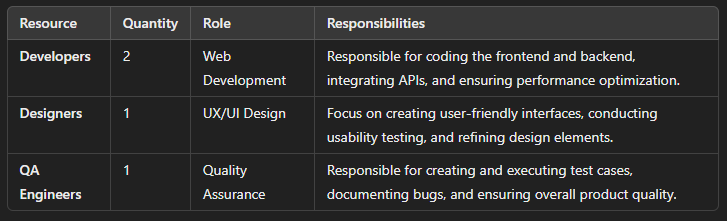
### 5.2 Dependencies

Understanding dependencies is crucial for managing the project timeline effectively. The following dependencies have been identified:

1. **Completion of the Design Before Development:** The design phase must be finalized and approved before development begins. This ensures that developers have clear specifications to work from, minimizing rework and design inconsistencies.
2. **Testing to Occur Before Deployment:** Comprehensive testing must be completed before the website is deployed to production. This includes validating functionality, performance, and security to ensure a seamless user experience.

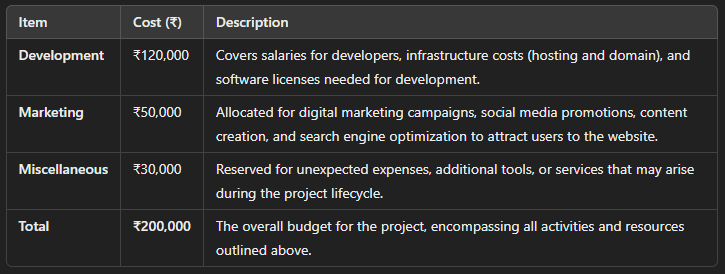
### 5.3 Resource Requirements

To successfully execute the project, the following resources will be required:

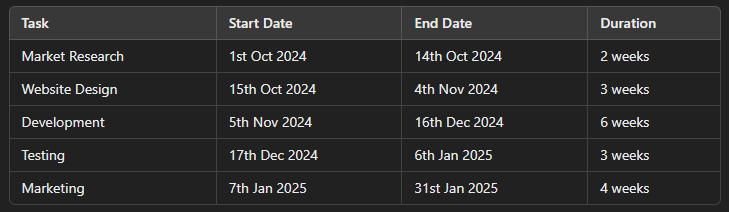


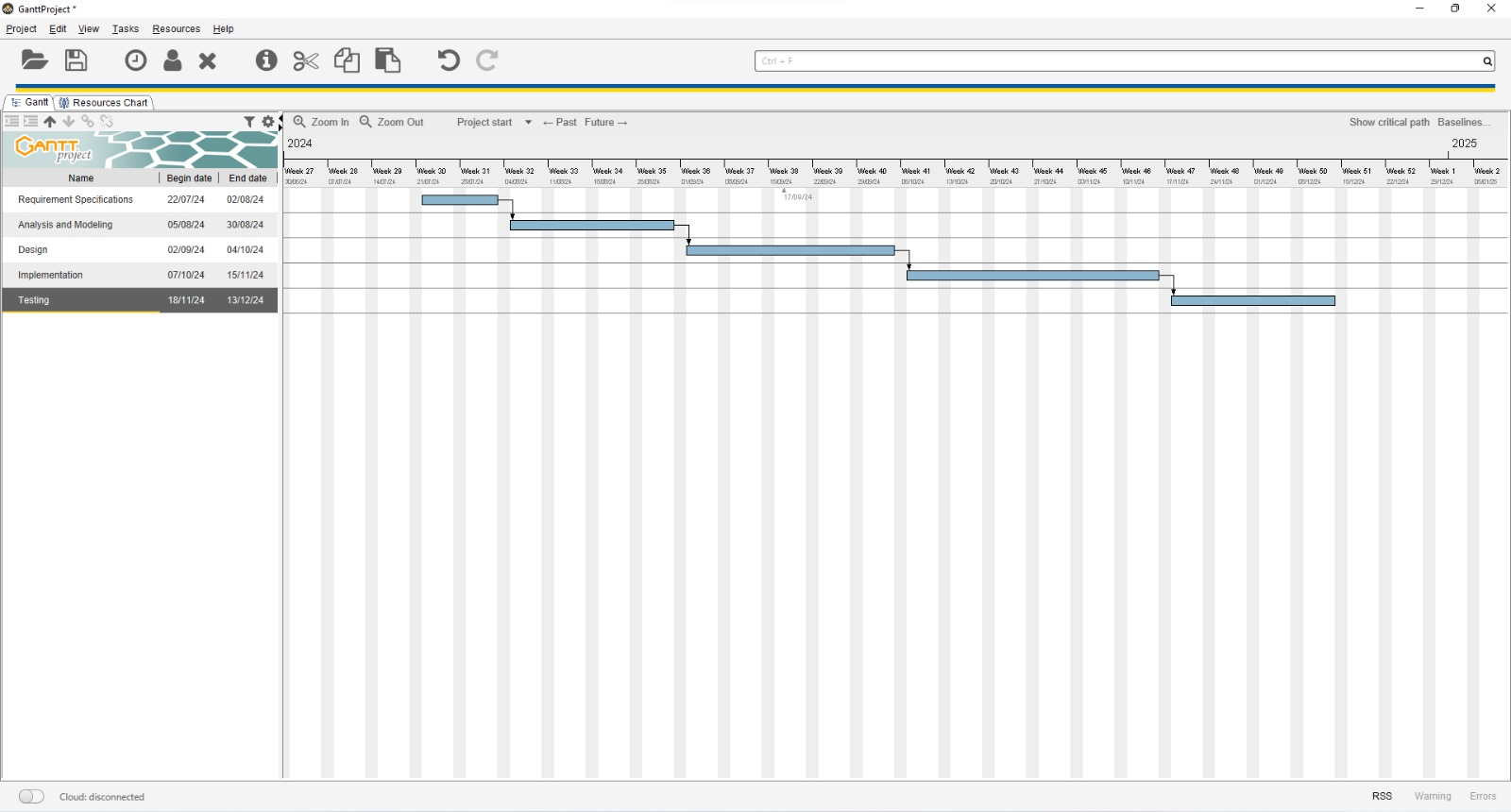
### 5.4 Budget and Resource Allocation

A detailed budget breakdown will ensure effective financial management throughout the project. The budget allocation is as follows:

  
**5.5 Schedule**

A Gantt chart will be utilized to visualize the project timeline, deadlines, and interdependencies of the work packages. The schedule will help track progress and ensure timely delivery of the project. Below is a simplified representation of the project schedule:





**6. Additional Information**

To facilitate navigation of this document, an index will be provided in future versions, allowing users to locate specific sections easily. Additionally, appendices will be included to document tools, third-party services, and other relevant information that supports the project’s execution.

**Conclusion:**

In conclusion, the successful execution of the "Le Miele Mists" project hinges on thorough planning, clear communication, and effective collaboration among all team members to deliver a high-quality online perfume store that meets user needs and market demands.

**Post Lab Descriptive Questions**

1. State various Scheduling principles and explain them in detail.

Scheduling is a crucial aspect of project management that involves planning the timing of project activities to ensure efficient use of resources and timely completion of tasks. Here are several key scheduling principles explained in detail:

### 1. Define Clear Objectives and Milestones

* **Explanation:** Before developing a schedule, it's essential to establish clear objectives for the project. Milestones are significant points in the project timeline that indicate the completion of key phases or deliverables. Defining these objectives helps in aligning the team’s efforts and provides a basis for measuring progress.
* **Implementation:** Break down the project into specific goals and set milestones for each major phase. This will help track progress and maintain focus.

### 2. Prioritization of Tasks

* **Explanation:** Not all tasks are equally important or time-sensitive. Prioritizing tasks based on their urgency and importance ensures that critical tasks are completed first, preventing potential bottlenecks.
* **Implementation:** Use methods such as the Eisenhower Matrix to categorize tasks into urgent and important, allowing for effective prioritization of project activities.

### 3. Resource Allocation

* **Explanation:** Efficient scheduling requires understanding the availability and skill sets of team members, as well as the resources required for each task. Proper allocation of resources ensures that tasks are completed without overburdening any team member or resource.
* **Implementation:** Create a resource management plan that outlines who is responsible for what tasks, and ensure that resources are available when needed.

### 4. Task Dependencies

* **Explanation:** Many tasks are interdependent, meaning that the completion of one task may be necessary for the start of another. Identifying these dependencies is critical for creating an accurate schedule.
* **Implementation:** Use tools like dependency diagrams or the Critical Path Method (CPM) to visualize task relationships and ensure that dependent tasks are scheduled appropriately.

### 5. Time Estimates

* **Explanation:** Accurate time estimation is vital for creating a realistic schedule. Overly optimistic or pessimistic estimates can lead to project delays or resource waste.
* **Implementation:** Utilize techniques like historical data analysis, expert judgment, and estimation software to derive accurate time estimates for each task.

### 6. Buffering for Uncertainties

* **Explanation:** No project goes exactly as planned. Unforeseen circumstances can arise, so it’s important to include buffers in the schedule to accommodate potential delays or resource shortages.
* **Implementation:** Add contingency time to critical tasks or phases to mitigate risks and ensure flexibility in the project schedule.

### 7. Regular Monitoring and Adjustment

* **Explanation:** A schedule is not static; it requires regular monitoring to assess progress and make adjustments as necessary. This helps in identifying potential issues early and allows for corrective actions to be taken.
* **Implementation:** Schedule regular check-ins and reviews, using project management tools to track progress against the plan, and adjust timelines and resources as needed.

### 8. Stakeholder Communication

* **Explanation:** Keeping stakeholders informed about the project schedule and any changes is essential for maintaining alignment and managing expectations.
* **Implementation:** Establish a communication plan that outlines how and when updates will be shared with stakeholders, ensuring transparency throughout the project.

### 9. Use of Project Management Tools

* **Explanation:** Leveraging project management software can significantly enhance scheduling effectiveness. These tools help visualize tasks, dependencies, and resource allocation, making it easier to manage complex projects.
* **Implementation:** Choose appropriate project management software (e.g., Microsoft Project, Trello, Asana) to create Gantt charts, task lists, and dashboards that provide real-time insights into project status.

### 10. Feedback Mechanism

* **Explanation:** Incorporating feedback from team members can improve scheduling accuracy and effectiveness. Team insights often reveal potential issues and improvement areas that may not be apparent at the planning stage.
* **Implementation:** Create a feedback loop by encouraging team members to share their thoughts on the schedule and its feasibility, and adjust the plan accordingly.

By applying these scheduling principles, project managers can create a more organized, realistic, and flexible schedule, enhancing the likelihood of project success while minimizing delays and resource wastage.