

**CS6003 – Advanced Software Engineering – Coursework 2**

*Project Plan Report*

Name: Chiemela Ndukwe

ID Number: 21034871

Date: Saturday, 25 February 2023

Lecturer: Dr Tingkai Wang

# Declaration

**Module:** CS6003 **Deadline:** 3pm Tuesday 21st March 2023

21034871

**Module Leader:** Dr Tingkai Wang **Student ID:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PLAGIARISM**

You are reminded that there exist regulations concerning plagiarism. Extracts from these regulations are printed below. Please sign below to say that you have read and understand these extracts:

**Chiemela Ndukwe**

Student signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: 25/02/23

This header sheet should be attached to the work you submit. No work will be accepted without it.

Extracts from University *Regulations* onCheating, Plagiarism and Collusion

Section 2.3: "The following broad types of offence can be identified and are provided as indicative examples…

1. Cheating: including taking unauthorised material into an examination; consulting unauthorised material outside the examination hall during the examination; obtaining an unseen examination paper in advance of the examination; copying from another examinee; using an unauthorised calculator during the examination or storing unauthorised material in the memory of a programmable calculator which is taken into the examination; copying coursework.
2. Falsifying data in experimental results.
3. Personation, where a substitute takes an examination or test on behalf of the candidate. Both candidate and substitute may be guilty of an offence under these Regulations.
4. Bribery or attempted bribery of a person thought to have some influence on the candidate's assessment.
5. Collusion to present joint work as the work solely of one individual.
6. Plagiarism, where the work or ideas of another are presented as the candidate's own.
7. Other conduct calculated to secure an advantage on assessment.

(viii) Assisting in any of the above.

Some notes on what this means for students:

1. Copying another student's work is an offence, whether from a copy on paper or from a computer file, and in whatever form the intellectual property being copied takes, including text, mathematical notation and computer programs.

2. Taking extracts from published sources *without attribution* is an offence. To quote ideas, sometimes using extracts, is generally to be encouraged. Quoting ideas is achieved by stating an author's argument and attributing it, perhaps by quoting, immediately in the text, his or her name and year of publication, e.g. "e = mc2 (Einstein 1905)". A *references* section at the end of your work should then list all such references in alphabetical order of authors' surnames. (There are variations on this referencing system which your tutors may prefer you to use.) If you wish to quote a paragraph or so from published work then indent the quotation on both left and right margins, using an italic font where practicable, and introduce the quotation with an attribution.

# Summary/Abstract

Developing a project plan for an event management system involves a series of steps that help to ensure that the project is well-defined, well-organized, and well-managed. The project plan report proposes the development of an event management system that aims to simplify the process of organizing and managing the time and resources. The proposed system is expected to increase efficiency, streamline the implementation of the event management process. The project plan will include a detailed timeline, budget, and risk management strategy to ensure successful project delivery. The first step in this project planning process is to define the project's scope, which involves identifying the goals, objectives, and deliverables of the project. Once the project scope has been defined, the next step is to create a timeline for the project which involves breaking the project down into smaller tasks and assigning each task a deadline. In addition to the project timeline, a budget for this project is included which involves identifying the resources needed to complete the project. Additionally, this project plan report will identify risks and develop a risk management plan including project roles and responsibilities. Overall, developing a project plan for an event management system requires careful planning, communication, and coordination among all team members involved in the project.

**Keywords**: Project plan report; Event management system; Budget; Timeline; Risk management

# Table of Contents

[Declaration 2](#_Toc128183768)

[Summary/Abstract 3](#_Toc128183769)

[Table of Contents 4](#_Toc128183770)

[List of Tables 6](#_Toc128183771)

[List of Figures 7](#_Toc128183772)

[Chapter 1: Generic Report Information 8](#_Toc128183773)

[1.1 Date of issue 8](#_Toc128183774)

[1.2 Issuing organisation 8](#_Toc128183775)

[1.3 Background of project 8](#_Toc128183776)

[1.4 Project scope, goals, and objectives 8](#_Toc128183777)

[1.4.1 Objectives 8](#_Toc128183778)

[1.4.2 Inclusions 8](#_Toc128183779)

[1.4.3 Exclusions 9](#_Toc128183780)

[1.4.4 Deliverables 9](#_Toc128183781)

[1.4.5 Constraints 9](#_Toc128183782)

[1.5 Methodology 9](#_Toc128183783)

[1.5.1 Agile methodology 10](#_Toc128183784)

[1.5.2 Scrum framework 10](#_Toc128183785)

[1.5.3 Project planning 10](#_Toc128183786)

[1.5.4 Agile development 10](#_Toc128183787)

[1.5.5 Testing and quality assurance 10](#_Toc128183788)

[1.5.6 Deployment and maintenance 10](#_Toc128183789)

[1.5.7 Justification of methodology 11](#_Toc128183790)

[1.6 Glossary 11](#_Toc128183791)

[Chapter 2: Project overview 13](#_Toc128183792)

[2.1 Project timeline 13](#_Toc128183793)

[2.2 Budget 14](#_Toc128183794)

[2.3 Major milestones 16](#_Toc128183795)

[Chapter 3: Project Scope 18](#_Toc128183796)

[3.1 Goals 18](#_Toc128183797)

[3.1.1 Establishing project objectives and scope 18](#_Toc128183798)

[3.1.2 Identifying project stakeholders 18](#_Toc128183799)

[3.1.3 Developing a project schedule and timeline 18](#_Toc128183800)

[3.1.4 Defining project resources and budget 18](#_Toc128183801)

[3.1.5 Developing a risk management plan 18](#_Toc128183802)

[3.1.6 Defining communication and reporting protocols 19](#_Toc128183803)

[3.2 Objectives 19](#_Toc128183804)

[3.2.1 To clearly define the scope of the project 19](#_Toc128183805)

[3.2.2 To identify project stakeholders and their roles 19](#_Toc128183806)

[3.2.3 To establish a realistic project schedule and timeline 19](#_Toc128183807)

[3.2.4 To define project resources and budget 19](#_Toc128183808)

[3.2.5 To develop a risk management plan 20](#_Toc128183809)

[3.2.6 To establish communication and reporting protocols 20](#_Toc128183810)

[3.2.7 To monitor project progress 20](#_Toc128183811)

[3.2.8 To ensure stakeholder satisfaction 20](#_Toc128183812)

[3.3 Deliverables of the projects 20](#_Toc128183813)

[Chapter 4: Project Schedule and Budget 22](#_Toc128183814)

[4.1 Deadlines 22](#_Toc128183815)

[4.2 Organisational structure 22](#_Toc128183816)

[4.2.1 Team structure description 23](#_Toc128183817)

[4.2.2 Team structure tree 24](#_Toc128183818)

[4.2.3 Stakeholders and each person’s role 24](#_Toc128183819)

[4.3 Budget breakdown structure 25](#_Toc128183820)

[Chapter 5: Work Breakdown Structure (WBS) 28](#_Toc128183821)

[5.1 Work breakdown structure based on the ISO 12207-2008 tasks 28](#_Toc128183822)

[5.2 Work breakdown structure rationale 28](#_Toc128183823)

[5.3 Gantt chart 29](#_Toc128183824)

[5.4 Project milestones 29](#_Toc128183825)

[5.5 Project milestones rationale 30](#_Toc128183826)

[Chapter 6: Risk Management and Software Quality Management Strategies 32](#_Toc128183827)

[6.1 Risk identification 32](#_Toc128183828)

[6.2 Risk analysis 32](#_Toc128183829)

[6.3 Risk planning 33](#_Toc128183830)

[6.4 Risk monitoring 34](#_Toc128183831)

[Conclusion 35](#_Toc128183832)

[Bibliography 36](#_Toc128183833)

# List of Tables

**Table 1: Internal deliverables** 21

**Table 2: External deliverables** 21

**Table 3: Deadline table** 22

**Table 4: Team structure description** 23

**Table 5: Stakeholders and each person’s role** 24

# List of Figures

**Figure 1: Project timeline** 14

**Figure 2: Budget overview** 15

**Figure 3: Budget overview** 16

**Figure 4: Project milestone overview** 17

**Figure 5: Deadline on timeline** 22

**Figure 6: Team structure tree** 24

**Figure 7: Cost Details** 25

**Figure 8: Resource cost** 26

**Figure 9: Cost status** 27

**Figure 10: Resource cost variance** 27

**Figure 11: WBS** 28

**Figure 12: Gantt Chart** 29

**Figure 13: Project milestones** 30

# Chapter 1: Generic Report Information

## 1.1 Date of issue

Project start date is 11th February, 2023. Project duration is 11 weeks. Project deadline is 28th April, 2023.

## 1.2 Issuing organisation

This project is carried out by Best Software Ltd.

## 1.3 Background of project

This project management is for Better-Than-At-Home catering company. Better-Than-At-Home provides catering services specifically to vegan customers. Mrs A. Cook who is the owner of Better-Than-At-Home requires a web-based event management system for her business. The goal for creating a web-based event management system is to reduce the frequent calls received by Better-Than-At-Home. However, this project plan will include a detailed timeline, budget, and risk management strategy to ensure successful project delivery.

## 1.4 Project scope, goals, and objectives

By defining the project scope, project managers can effectively plan and manage the project to ensure that it meets the defined objectives, stays within budget and is completed on time. The project scope provides a clear understanding of what is included in the project, what is excluded, and what is expected as a deliverable. It also helps to manage stakeholder expectations and minimize project risks.

## 1.4.1 Objectives

* To develop a user-friendly and efficient event management system that meets the needs of event organizers, attendees, and vendors.
* To improve event planning and management processes and reduce administrative workloads.
* To increase event attendance and engagement by providing attendees with easy access to event information and registration.

## 1.4.2 Inclusions

* Development of a web-based event management system that allows event organizers to create and manage events.
* Development of a user interface that allows attendees to easily search for and register for events.
* Development of a payment processing system to allow for secure online payments for event registration and purchases.
* Development of a reporting system to allow system admin to generate reports on event attendance, revenue, and other metrics.

## 1.4.3 Exclusions

* The project does not include the development of a mobile application for the event management system.
* The project does not include the development of any hardware or infrastructure.

## 1.4.4 Deliverables

* A fully functional event management system that meets the defined requirements and objectives.
* User manuals and training materials for event organizers and attendees.
* Technical documentation, including system architecture, data models, and system requirements.

## 1.4.5 Constraints

* The project must be completed within a timeframe of 11 weeks.
* The project must be completed within the defined budget of £80,000.
* The project must comply with all relevant legal and regulatory requirements.

## 1.5 Methodology

The project management methodology defines the processes and procedures that will be used to manage the project from start to finish. For the development of an event management system for Better-Than-At-Home, Agile methodology shall be used. Furthermore, the section shall explain these methodologies in more detail.

## 1.5.1 Agile methodology

The agile methodology is a flexible and iterative approach to project management that emphasizes collaboration and continuous delivery. This methodology is suitable for software development projects like the development of an event management system, which involves frequent changes and updates based on user feedback.

## 1.5.2 Scrum framework

The Scrum framework is a popular agile methodology that is widely used in software development projects. It involves dividing the project into short sprints, typically two to four weeks long, during which the development team works on a set of predefined tasks. At the end of each sprint, the team reviews progress, identifies any issues, and adjusts their plan for the next sprint.

## 1.5.3 Project planning

The project planning phase includes defining project scope, objectives, and requirements, as well as creating a project schedule and budget. The planning phase also involves identifying project risks and developing a risk management plan to mitigate those risks.

## 1.5.4 Agile development

The agile development phase involves breaking down the project into small, manageable tasks, and prioritizing those tasks based on their importance and complexity. The development team works on each task in short sprints, and each sprint produces a working software component that is integrated into the system.

## 1.5.5 Testing and quality assurance

The testing and quality assurance phase involves testing the system at each stage of development to ensure that it meets the defined requirements and objectives. The testing phase includes unit testing, integration testing, system testing, and user acceptance testing.

## 1.5.6 Deployment and maintenance

The deployment and maintenance phase involves releasing the system to end-users and providing support and maintenance services to ensure that the system continues to function properly. This phase also includes monitoring system performance and making necessary updates and enhancements based on user feedback.

## 1.5.7 Justification of methodology

By using an Agile methodology with a Scrum framework, project managers can more effectively manage the development of an event management system. This approach allows for frequent collaboration and feedback, and ensures that the project stays on track and within budget. The methodology also provides a structured approach to project management, with clear roles and responsibilities for each team member, and a focus on delivering a high-quality, user-friendly system that meets the defined objectives.

## 1.6 Glossary

**Event Management System**: A web-based software system designed to manage the planning, promotion, registration, and execution of events.

**Agile Methodology**: An iterative and flexible approach to project management that emphasizes collaboration, adaptability, and continuous delivery.

**Scrum Framework**: A specific methodology within the Agile framework that involves dividing a project into short sprints, with each sprint resulting in a working software component.

**Work Breakdown Structure (WBS)**: A hierarchical breakdown of the project tasks that defines the scope of the project, the tasks required to complete it, and the interrelationships between tasks.

**Scope**: The boundaries of the project, including its goals, objectives, requirements, deliverables, and exclusions.

**Deliverables**: Tangible products or services that must be produced or delivered as part of the project.

**Stakeholder**: Any individual or organization that has an interest or role in the project, including project team members, sponsors, customers, and users.

**Risk Management**: The process of identifying, assessing, and mitigating potential risks to the project.

**User Acceptance Testing**: A type of testing that involves end-users evaluating the system to ensure that it meets their needs and expectations.

**Quality Assurance**: The process of ensuring that the system meets defined quality standards and requirements.

**Deployment**: The process of releasing the system to end-users and making it available for use.

**Maintenance**: Ongoing support and maintenance services to ensure that the system continues to function properly after its release.

**Budget**: The amount of money allocated to the project, including funding for personnel, equipment, software, and other resources.

**User Interface (UI)**: The graphical interface that allows users to interact with the event management system.

**Payment Processing System**: The software system that allows for secure online payments for event registration and purchases.

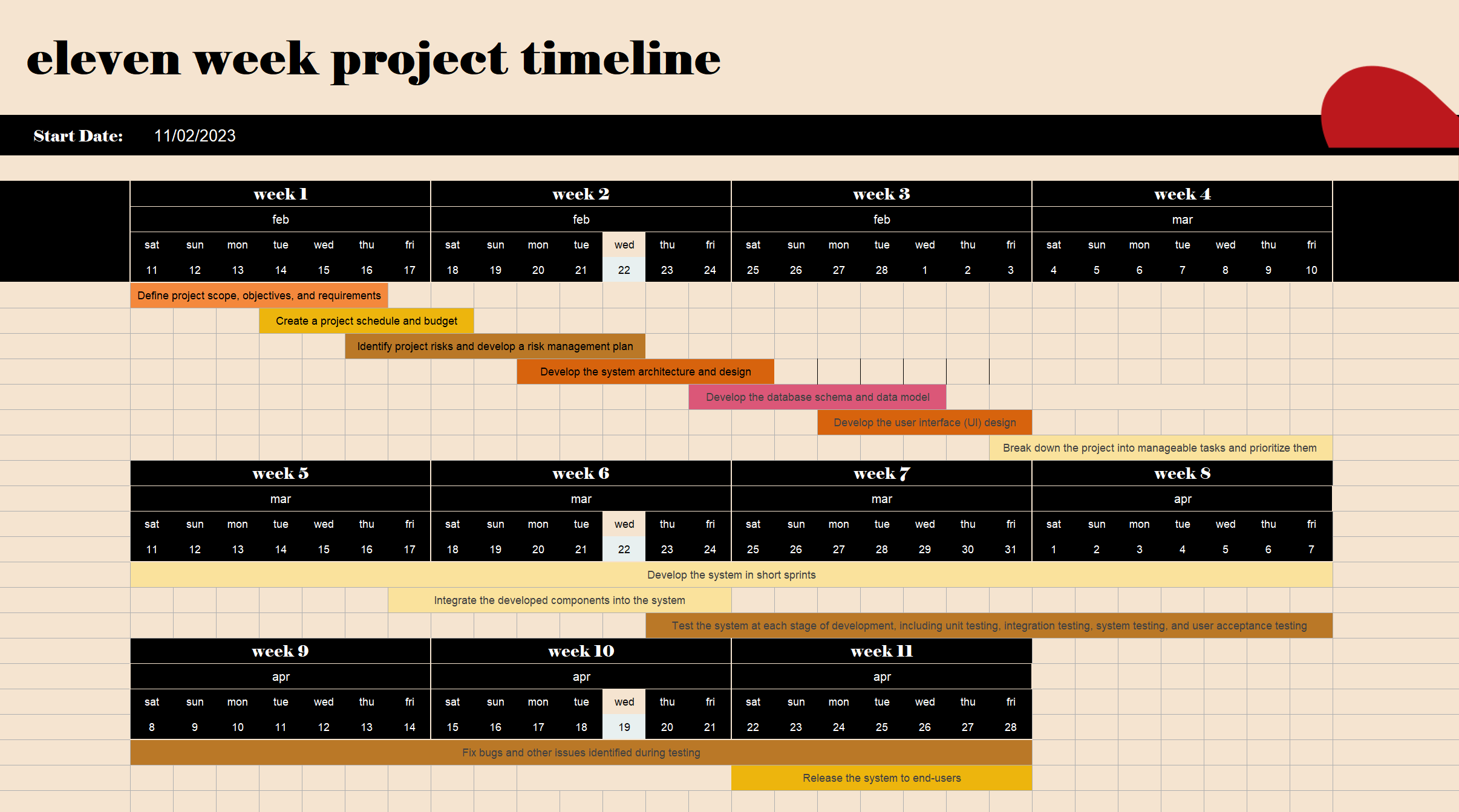
**Gantt Chart**: Horizontal vertical representation of all project activities. Gantt chart also illustrates the project dependencies.

# Chapter 2: Project overview

## 2.1 Project timeline

The project timeline for the development of an event management system depends on the specific requirements, scope, and resources of the project. Overall, this timeline assumes a development team of 7 individuals with experience in software development and project management. The project timeline shall include:

* Define project scope, objectives, and requirements.
* Create a project schedule and budget.
* Identify project risks and develop a risk management plan.
* Develop the system architecture and design.
* Develop the database schema and data model.
* Develop the user interface (UI) design.
* Break down the project into manageable tasks and prioritize them.
* Develop the system in short sprints.
* Integrate the developed components into the system.
* Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing.
* Fix bugs and other issues identified during testing.
* Release the system to end-users.

  
Figure 1: Project timeline.

## 2.2 Budget

The budget for the project is £80,000, and the project duration is 11 weeks. To determine the total cost of staffing for the project, we need to calculate the number of hours worked by each team member and multiply that by their hourly rate.

As for the project manager, his involvement shall be part-time, which is not more than 25% of their time. Using the formular 25/100 \* 40 hours = 0.25 \* 10 = 2.5 hours. 2.5 hours \* 5 weekdays = 12.5 hours per week. The total number of hours worked by the project manager over the project period is 11 x 12.5 = 137.5 hours.

Full-time team member works an average of 40 hours per week, the total number of hours worked by the Full-time team over the project period is 11 x 40 = 440 hours.

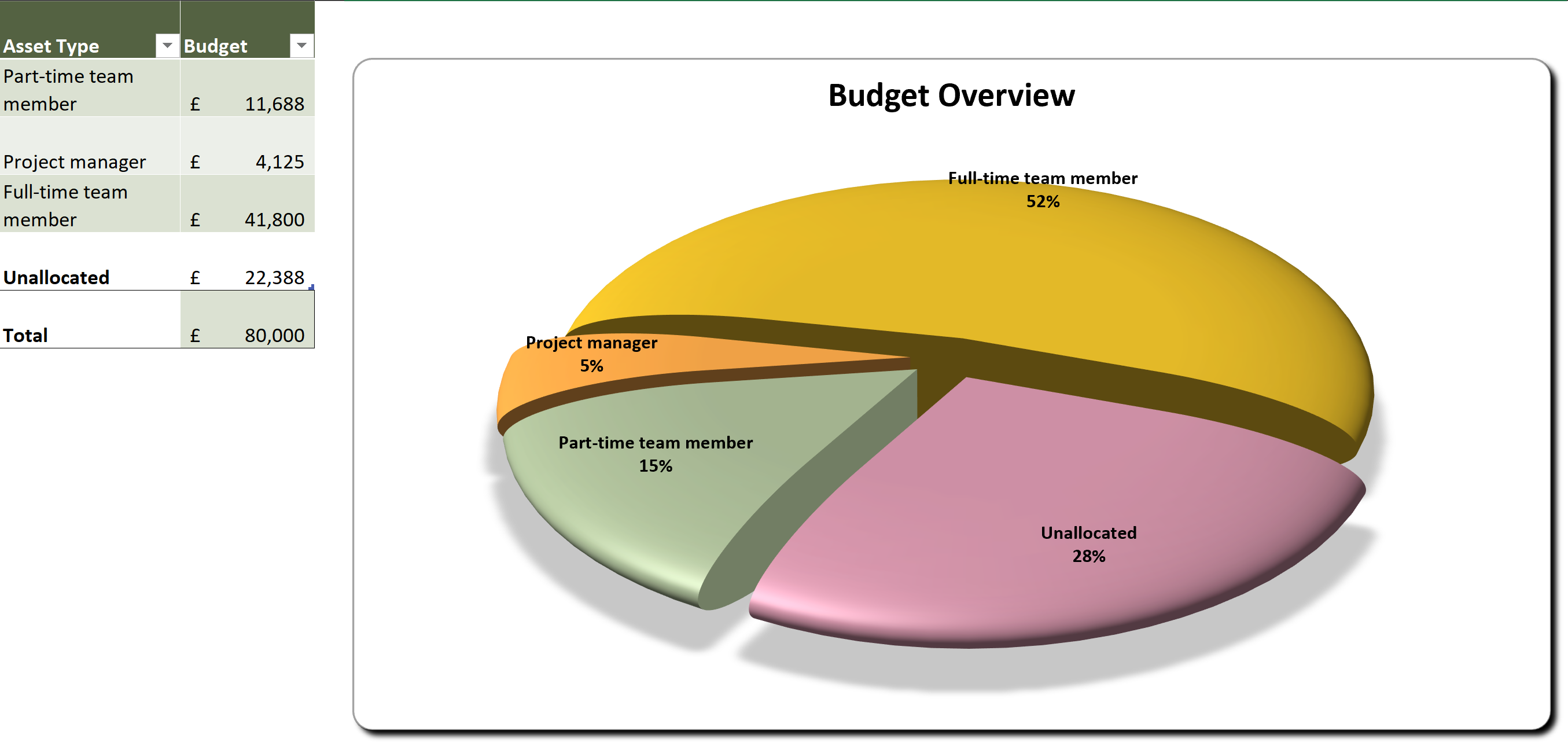
Whereas Part-time team member work only 25% of their time which is an average of 12.5 hours per week, the total number of hours worked by the Part-time team over the project period is 11 x 12.5 = 137.5 hours.

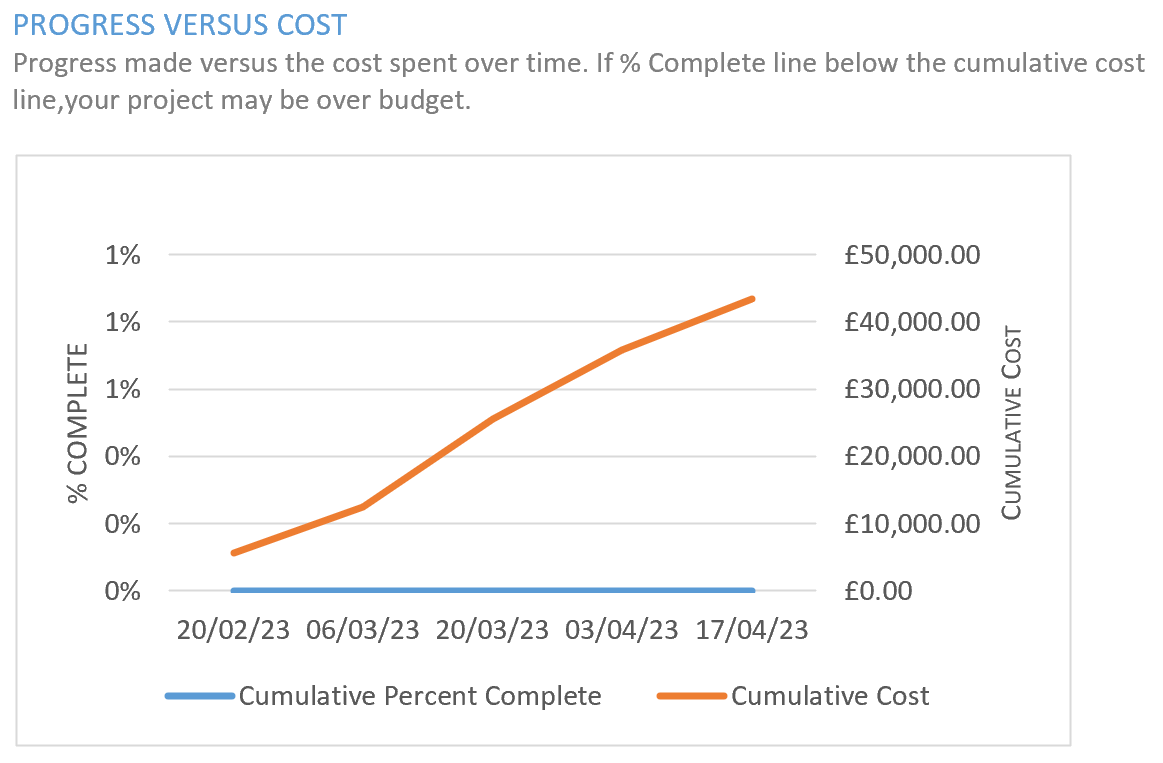
Based on the given salaries, the cost estimate of staffing the project would be as follows:

* Project manager: £30 per hour x 137.5 hours = £4,125
* Full-time team member 1: £35 per hour x 440 hours = £15,400
* Full-time team member 2: £35 per hour x 440 hours = £15,400
* Full-time team member 3: £25 per hour x 440 hours = £11,000
* Part-time team member: £35 per hour x 137.5 hours = £4,812.50
* Part-time team members: £25 per hour x 137.5 hours = £3,437.50
* Part-time team members: £25 per hour x 137.5 hours = £3,437.50

Estimate cost of staffing: £57,612.50

The total cost of staffing is below the project budget of £80,000, indicating that the team has sufficient budget to cover the cost of staffing. Overall, the project team can use the budget to ensure that the project is adequately resourced and delivered within the available budget. If the project costs are expected to exceed the budget, the team may need to re-evaluate the project requirements, timeline, or scope to ensure that the project remains financially viable.

  
Figure 2: Budget overview.

  
Figure 3: Budget overview.

## 2.3 Major milestones

The project milestone is supposed to guide the project management team to stay on track with the requirements and objectives indicated for this project. The project milestones overview will include the following:

  
Figure 4: Project milestone overview.

# Chapter 3: Project Scope

## 3.1 Goals

The goal of project planning is to establish a clear and comprehensive roadmap for the event management system project. This will help ensure that the project is completed on time, within budget, and meets the needs of all stakeholders. The goals of doing a project planning for the event management system are further broken down below.

## 3.1.1 Establishing project objectives and scope

The first goal of project planning is to establish clear and well-defined project objectives and scope. This will help ensure that the project team has a shared understanding of the project goals, deliverables, and timeline.

## 3.1.2 Identifying project stakeholders

Another goal of project planning is to identify project stakeholders and their respective roles and responsibilities. This will help ensure that all stakeholders are involved in the project and have a clear understanding of their contributions.

## 3.1.3 Developing a project schedule and timeline

The project planning process aims to develop a comprehensive project schedule and timeline that outlines key milestones and deliverables. This will help ensure that the project is completed on time and within budget.

## 3.1.4 Defining project resources and budget

Project planning involves defining the necessary resources and budget required to complete the project successfully. This includes identifying the necessary personnel, equipment, and materials required for the project and estimating the associated costs.

## 3.1.5 Developing a risk management plan

The project planning process also involves developing a risk management plan that identifies potential risks and outlines strategies for managing them. This will help the project team proactively manage risks that could impact project success.

## 3.1.6 Defining communication and reporting protocols

Project planning aims to establish clear communication and reporting protocols for the project. This includes identifying who needs to be kept informed about the project and how information will be shared among stakeholders.

## 3.2 Objectives

The objectives of project planning for the event management system project are to establish a clear and comprehensive roadmap for the project, ensure effective communication and collaboration among stakeholders, and deliver a high-quality product that meets stakeholder needs and expectations. The objectives of doing a project planning for the event management system are further broken down below.

## 3.2.1 To clearly define the scope of the project

The first objective of project planning is to clearly define the scope of the event management system project. This will ensure that all project stakeholders have a shared understanding of the project goals, deliverables, and timeline.

## 3.2.2 To identify project stakeholders and their roles

Another objective of project planning is to identify project stakeholders and their respective roles and responsibilities. This will ensure that all stakeholders are involved in the project and have a clear understanding of their contributions.

## 3.2.3 To establish a realistic project schedule and timeline

The project planning process aims to establish a realistic project schedule and timeline that outlines key milestones and deliverables. This will help ensure that the project is completed on time and within budget.

## 3.2.4 To define project resources and budget

Project planning involves defining the necessary resources and budget required to complete the event management system project successfully. This includes identifying the necessary personnel, equipment, and materials required for the project and estimating the associated costs.

## 3.2.5 To develop a risk management plan

The project planning process also involves developing a risk management plan that identifies potential risks and outlines strategies for managing them. This will help the project team proactively manage risks that could impact project success.

## 3.2.6 To establish communication and reporting protocols

Project planning aims to establish clear communication and reporting protocols for the project. This includes identifying who needs to be kept informed about the project and how information will be shared among stakeholders.

## 3.2.7 To monitor project progress

Another objective of project planning is to establish mechanisms for monitoring project progress, such as regular status reports, team meetings, and performance metrics. This will help the project team identify and address issues that may arise during project implementation.

## 3.2.8 To ensure stakeholder satisfaction

The final objective of project planning is to ensure stakeholder satisfaction by delivering a high-quality event management system that meets the needs of all project stakeholders. This will help establish positive relationships with stakeholders and set the stage for future project success.

## 3.3 Deliverables of the projects

**Internal Deliverables**

The internal deliverables are intended for the internal project team and stakeholders. The internal deliverables are essential for managing the project effectively.

|  |  |  |
| --- | --- | --- |
| **Deliverable name** | **Description** | **Delivery time** |
| Project Plan | A comprehensive project plan outlining the scope, goals, timeline, and budget for the project. | 22nd Feb |
| Requirements Document | A detailed requirements document that outlines the specific features and functionality of the event management system. | 25th Feb |
| Design Documents | Detailed design documents that outline the system architecture, database schema, and user interface design. | 1st Mar |
| Functional Prototype | A functional prototype of the event management system that demonstrates the key features and functionality. | 3rd Mar |
| Testing and Quality Assurance Documents | Documents outlining the testing and quality assurance processes used to ensure the event management system meets requirements and operates correctly. | 7th Apr |
| Training Materials | Training materials for users and administrators to ensure they can effectively use and maintain the event management system. | 21st Apr |
|  | | |

Table 1: Internal deliverables.

**External Deliverables**

The external deliverables are intended for external stakeholders, such as end-users and customers. The external deliverables are essential for ensuring the successful implementation and adoption of the event management system.

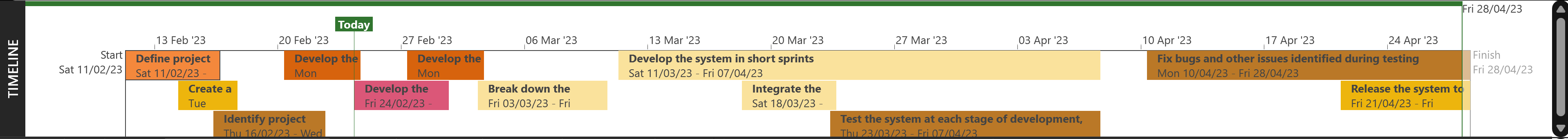
|  |  |  |
| --- | --- | --- |
| **Deliverable name** | **Description** | **Delivery time** |
| Final System Implementation | The final implementation of the event management system, including installation, configuration, and deployment. | 10th Mar |
| User Manual | A user manual that provides instructions for using the event management system. | 17th Mar |
| Technical Documentation | Technical documentation that provides information on the technical aspects of the system, including system architecture and database schema. | 10th Apr |
| Project Closure Report | A project closure report that summarizes the outcomes and results of the project, including any lessons learned and recommendations for future projects. | 28th Apr |
|  | | |

Table 2: External deliverables.

# Chapter 4: Project Schedule and Budget

## 4.1 Deadlines

Deadlines are a critical component of any project plan as they help to ensure that the project is completed on time and within budget. The project manager is responsible for setting and tracking deadlines for each task and milestone in the project plan. In the event management system project, the deadlines are set according to the project timeline. The project timeline outlines the start and end dates for each task and milestone, as well as the overall project completion date. It is essential to ensure that each team member understands the importance of meeting the set deadlines. This includes communicating the deadlines clearly, monitoring progress regularly, and addressing any issues that may arise promptly.

  
Figure 5: Deadline on timeline.

|  |  |  |
| --- | --- | --- |
| **S/N** | **Task** | **Deadline** |
| **1** | Define project scope, objectives, and requirements | 18th Feb |
| **2** | Create a project schedule and budget | 20th Feb |
| **3** | Identify project risks and develop a risk management plan | 24th Feb |
| **4** | Develop the system architecture and design | 27th Feb |
| **5** | Develop the database schema and data model | 3rd Mar |
| **6** | Develop the user interface (UI) design | 5th Mar |
| **7** | Break down the project into manageable tasks and prioritize them | 12th Mar |
| **8** | Develop the system in short sprints | 9th Apr |
| **9** | Integrate the developed components into the system | 26th Mar |
| **10** | Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing | 9th Apr |
| **11** | Fix bugs and other issues identified during testing | 30th Apr |
| **12** | Release the system to end-users | 30th Apr |
|  |  |  |

Table 3: Deadline table.

## 4.2 Organisational structure

The organizational structure is hierarchical, with Chiemela as the project manager responsible for the entire project, while the other team members report to him. John and Peter are senior software engineers with years of experience who led the development team, while Felix, Dan, Aaron, and Ben are software engineers who work under their leadership. The mix of full-time and part-time positions allows for flexibility in the development process while maintaining quality control.

## 4.2.1 Team structure description

The team consists of the project manager, three full-time members and three people working not more than 25% of their time.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Name** | **Specialisation** | **Experience** | **Availability** |
| **1** | Chiemela | Project manager | 10 years | Part-time |
| **2** | John | Senior software engineer | 8 years | Full-time |
| **3** | Peter | Senior software engineer | 8 years | Full-time |
| **4** | Felix | Full-stack developer | 4 years | Full-time |
| **5** | Dan | Junior developer | 4 years | Part-time |
| **6** | Aaron | Junior software engineer | 3 years | Part-time |
| **7** | Ben | Junior software engineer | 3 years | Part-time |
|  |  |  |  |  |

Table 4: Team structure description.

**Below is a short description of team structure.**

**Project Manager: Chiemela**

Responsible for overseeing the entire project, managing the project team, and ensuring the project is completed on time, within scope, and within budget.

**Full-Time Senior Software Engineer: John**

Responsible for leading the development team, designing and implementing software solutions, and ensuring the quality of the code. He is also responsible for testing and debugging code, collaborating with the development team to ensure code quality, and releasing the final software.

**Full-Time Senior Software Engineer: Peter**

Responsible for leading the development team, designing and implementing software solutions, and ensuring the quality of the code. He is also responsible for testing and debugging code, collaborating with the development team to ensure code quality, and releasing the final software.

**Full-Time Full-Stack Developer: Felix**

Responsible for developing and testing full-stack software solutions, working with the development team to ensure code quality, and collaborating with other teams to integrate software components. He is also responsible for testing and debugging code, collaborating with the development team to ensure code quality, and releasing the final software.

**Part-Time Junior Developer: Dan**

Responsible for assisting with software development tasks, testing and debugging code, and collaborating with the development team to ensure code quality.

**Part-Time Junior Software Engineer: Aaron**

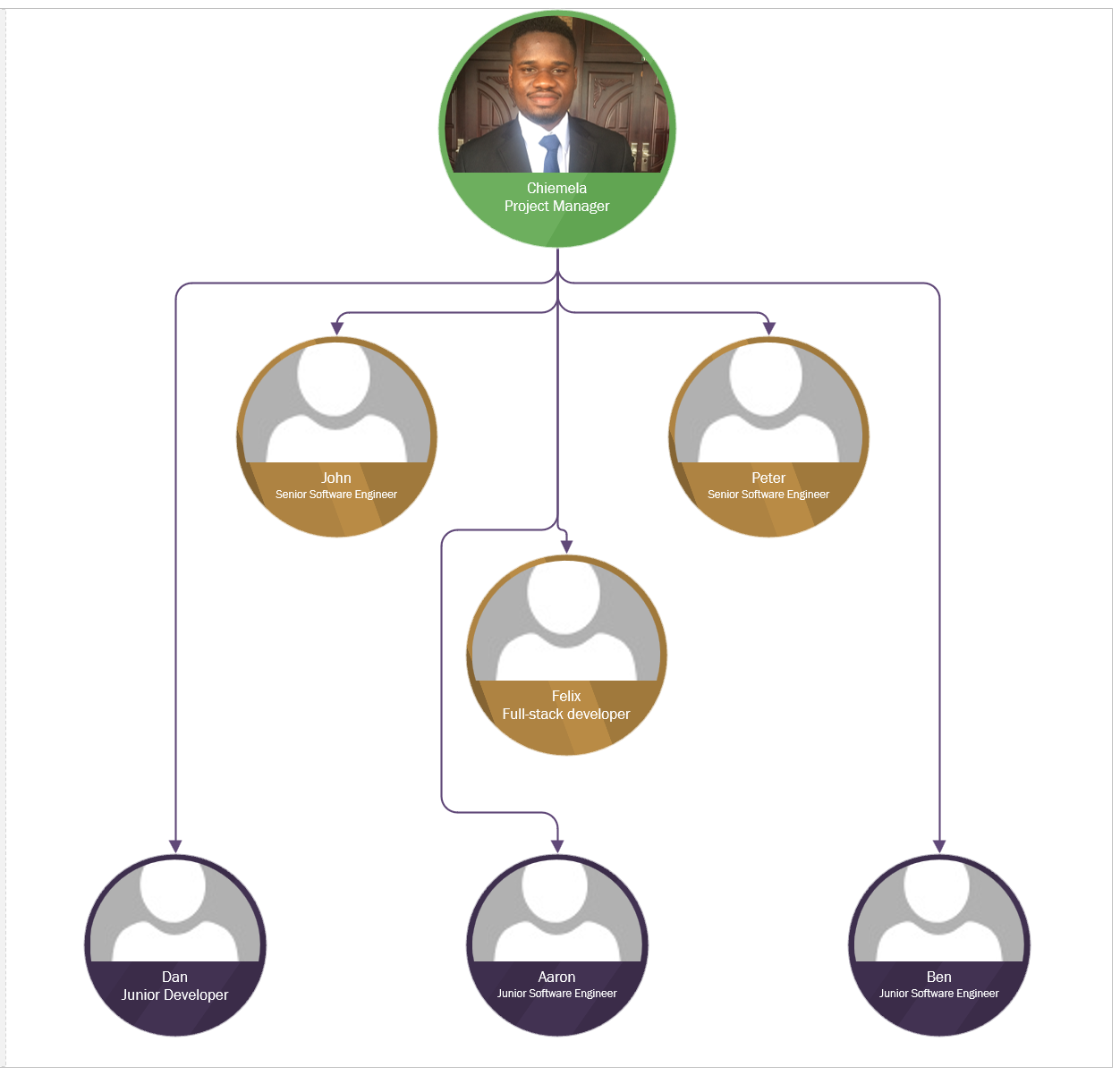
Responsible for assisting with software development tasks, testing and debugging code, and collaborating with the development team to ensure code quality.

**Part-Time Junior Software Engineer: Ben**

Responsible for assisting with software development tasks, testing and debugging code, and collaborating with the development team to ensure code quality.

## 4.2.2 Team structure tree

This structure represents the hierarchical relationship between the project manager and the team members. The project manager is responsible for overseeing the entire project and ensuring that it is completed on time, within budget, and to the required quality standards. The other team members report to the project manager and are responsible for specific aspects of the project. The full-time team members typically have more experience and expertise in their respective roles and may also be responsible for managing and supervising the part-time team members. The part-time team members may work on specific tasks or projects and report to the full-time team members or the project manager. This team structure provides a clear chain of command and facilitates efficient communication and decision-making within the team. It also helps to ensure that each team member has clearly defined responsibilities and understands their role in the project.

  
Figure 6: Team structure tree.

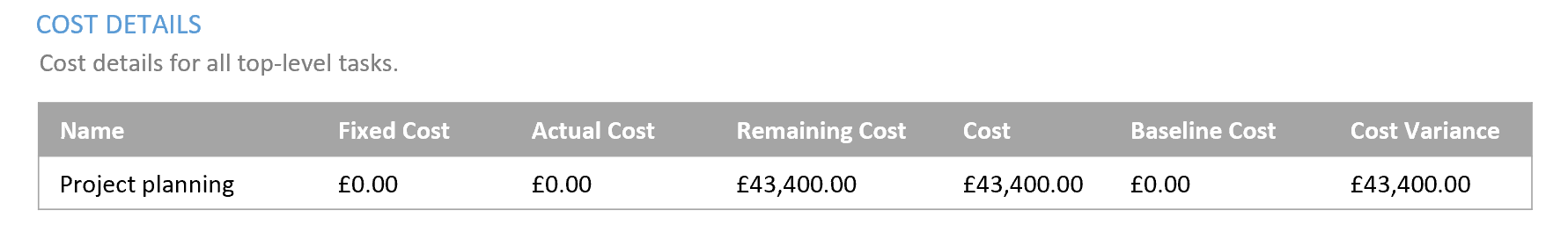
## 4.2.3 Stakeholders and each person’s role

|  |  |  |
| --- | --- | --- |
| **S/N** | **Name** | **Roles/Task** |
| **1** | Chiemela | 1. Define project scope, objectives, and requirements |
| 2. Create a project schedule and budget |
| 3. Identify project risks and develop a risk management plan |
| **2** | John | 1. Break down the project into manageable tasks and prioritize them |
| 2. Develop the system in short sprints |
| 3. Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing |
| 4. Fix bugs and other issues identified during testing |
| **3** | Peter | 1. Develop the system architecture and design |
| 2. Develop the system in short sprints |
| 3. Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing |
| 4. Fix bugs and other issues identified during testing |
| **4** | Felix | 1. Develop the database schema and data model |
| 2. Develop the system in short sprints |
| 3. Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing |
| 4. Fix bugs and other issues identified during testing |
| **5** | Dan | 1. Develop the user interface (UI) design |
| 2. Develop the system in short sprints |
| 3. Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing |
| **6** | Aaron | 1. Develop the system in short sprints |
| 2. Integrate the developed components into the system |
| 3. Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing |
| **7** | Ben | 1. Develop the system in short sprints |
| 2. Integrate the developed components into the system |
| 3. Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing |
|  |  |  |

Table 5: Stakeholders and each person’s role.

## 4.3 Budget breakdown structure

The total budget for the project is £80,000. The breakdown of this budget among the various categories will depend on the specific requirements and scope of the project. It is important to track expenses and adjust the budget as needed throughout the project to ensure that it stays within the allocated amount.

  
Figure 7: Cost Details.

Hours breakdown per allocated task is as follows:

Chiemela – 18h + 16h + 30h = 64h

John – 48h + 112h + 48h + 96h + 24h = 328h

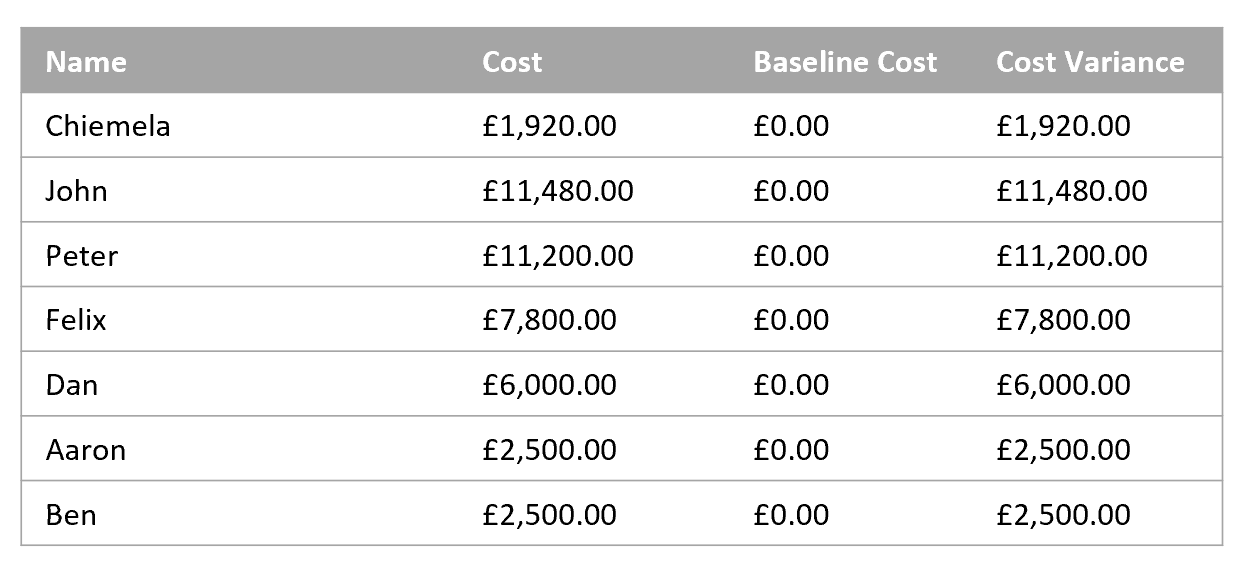
Peter – 40h + 112h + 48h + 96h + 24h = 320h

Felix – 32h + 112h + 48h + 96h + 24h = 313h

Dan – 40h + 112h + 48h = 200h

Aaron – 40h + 16h + 44h = 100h

Ben – 40h + 16h + 44h = 100h

  
Figure 8: Resource cost.

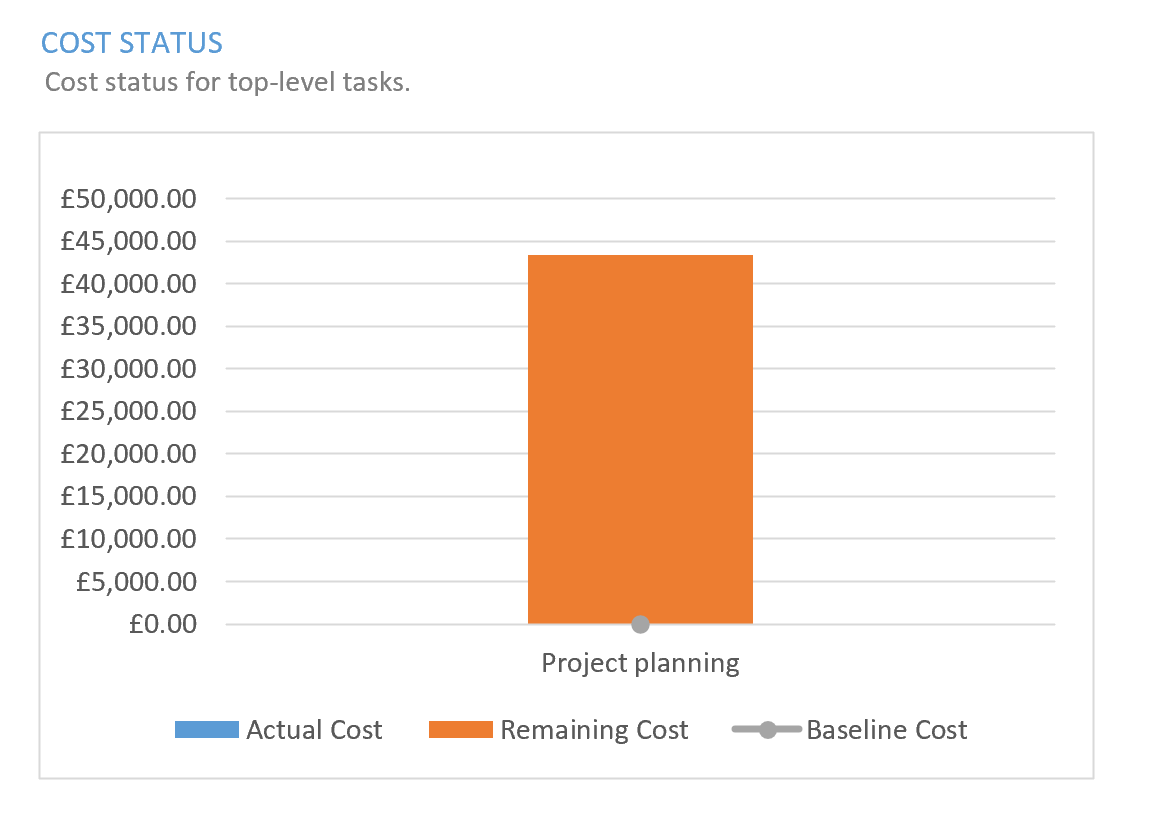
The budget breakdown structure for this project is as follows:

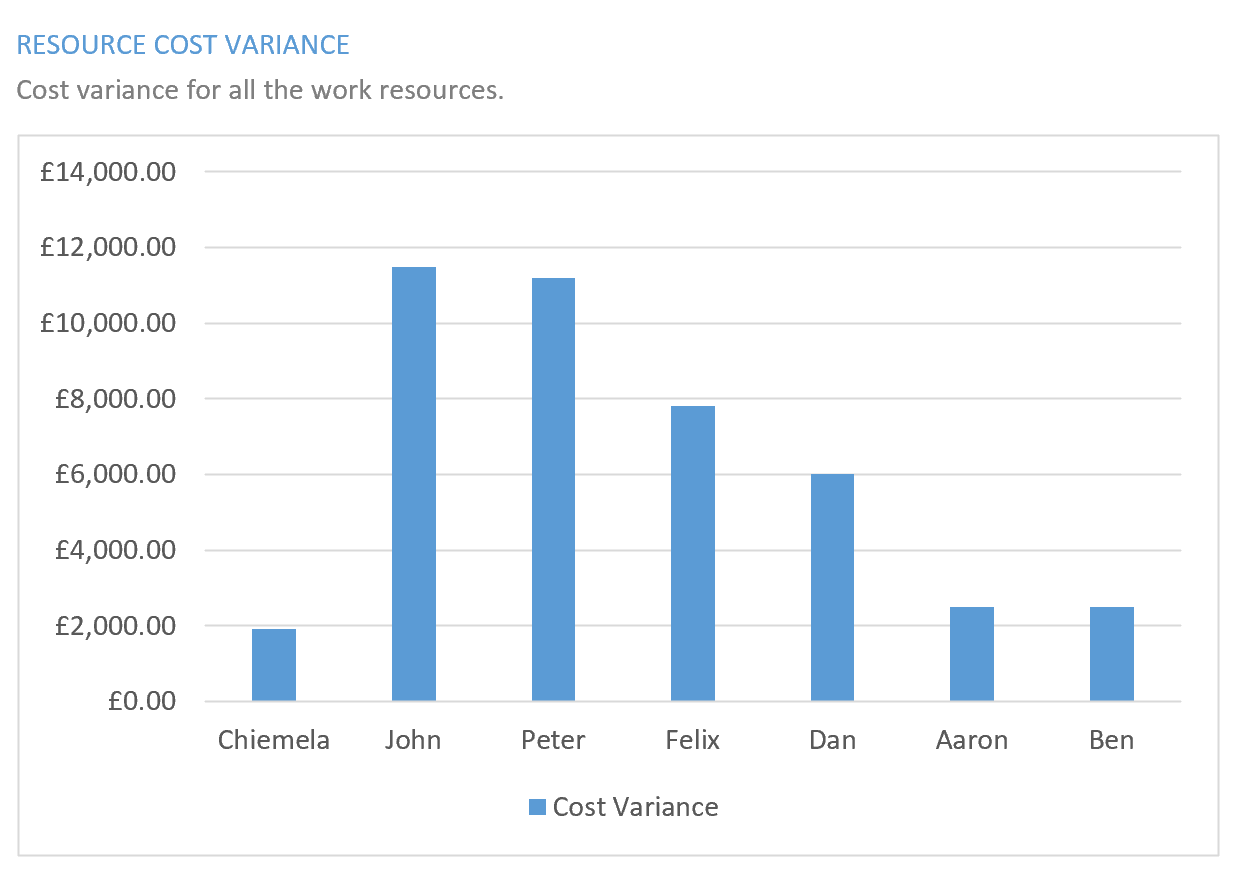
**Salaries**

* Project Manager: £30 per hour, 64 hours, 2 weeks.
* Full-Time Senior Software Engineer (John): £35 per hour, 328 hours, 8 weeks.
* Full-Time Senior Software Engineer (Peter): £35 per hour, 320 hours per week, 8 weeks.
* Full-Time Full-Stack Developer (Felix): £25 per hour, 312 hours, 8 weeks.
* Part-Time Junior Developer (Dan): £30 per hour, 200 hours, 5 weeks.
* Part-Time Junior Software Engineer (Aaron): £25 per hour, 100 hours, 3 weeks.
* Part-Time Junior Software Engineer (Ben): £25 per hour, 100 hours, 3 weeks.

**Contingency Fund**

* A percentage of the total budget set aside for unforeseen expenses or emergencies.

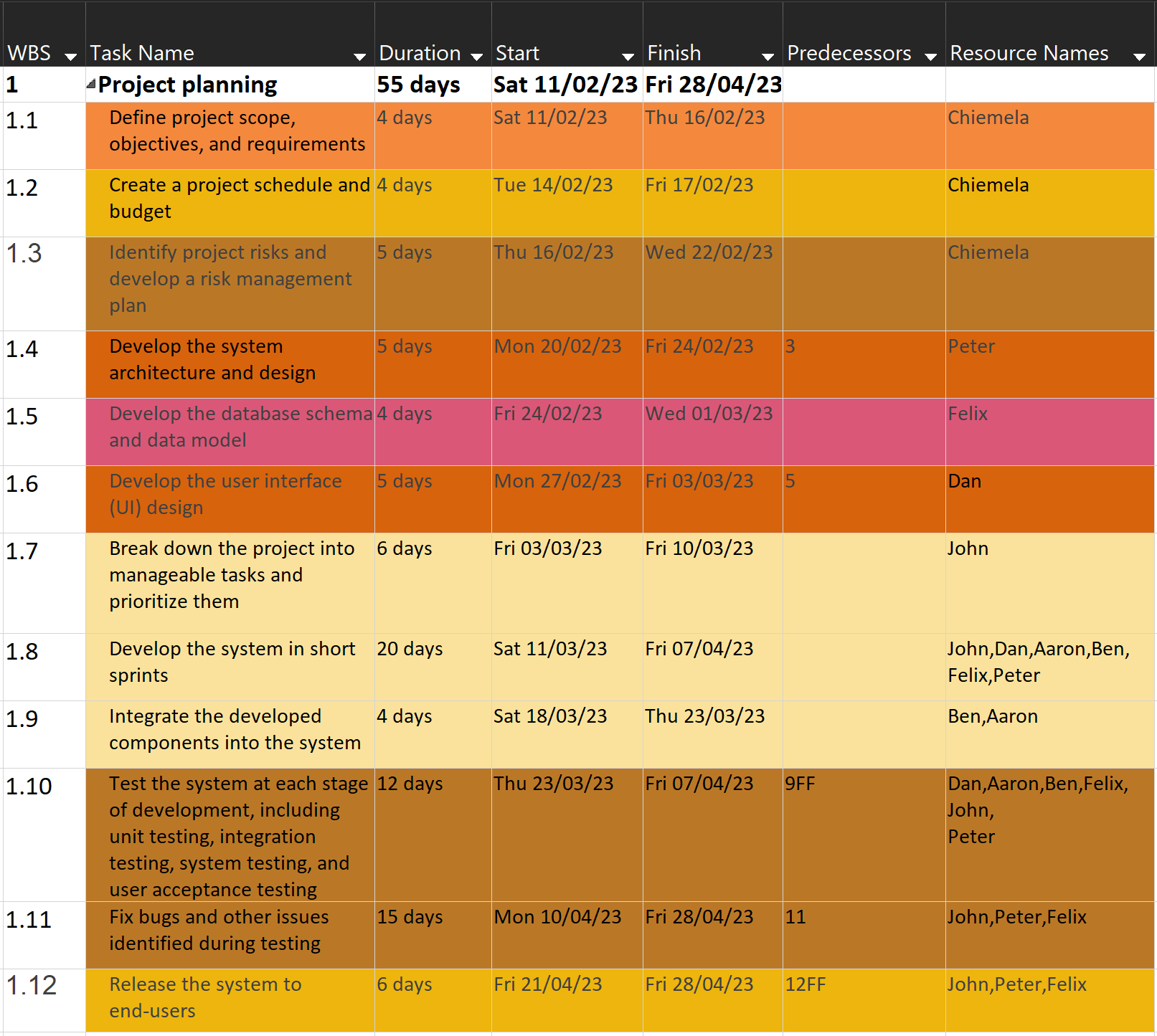
  
Figure 9: Cost status.

  
Figure 10: Resource cost variance.

# Chapter 5: Work Breakdown Structure (WBS)

## 5.1 Work breakdown structure based on the ISO 12207-2008 tasks

The Work Breakdown Structure (WBS) is a hierarchical decomposition of the project into smaller, more manageable work components. Each of these work components can be further broken down into smaller tasks, creating a comprehensive plan for the entire project. By breaking down the project into smaller, more manageable tasks, it becomes easier to estimate costs, track progress, and allocate resources.

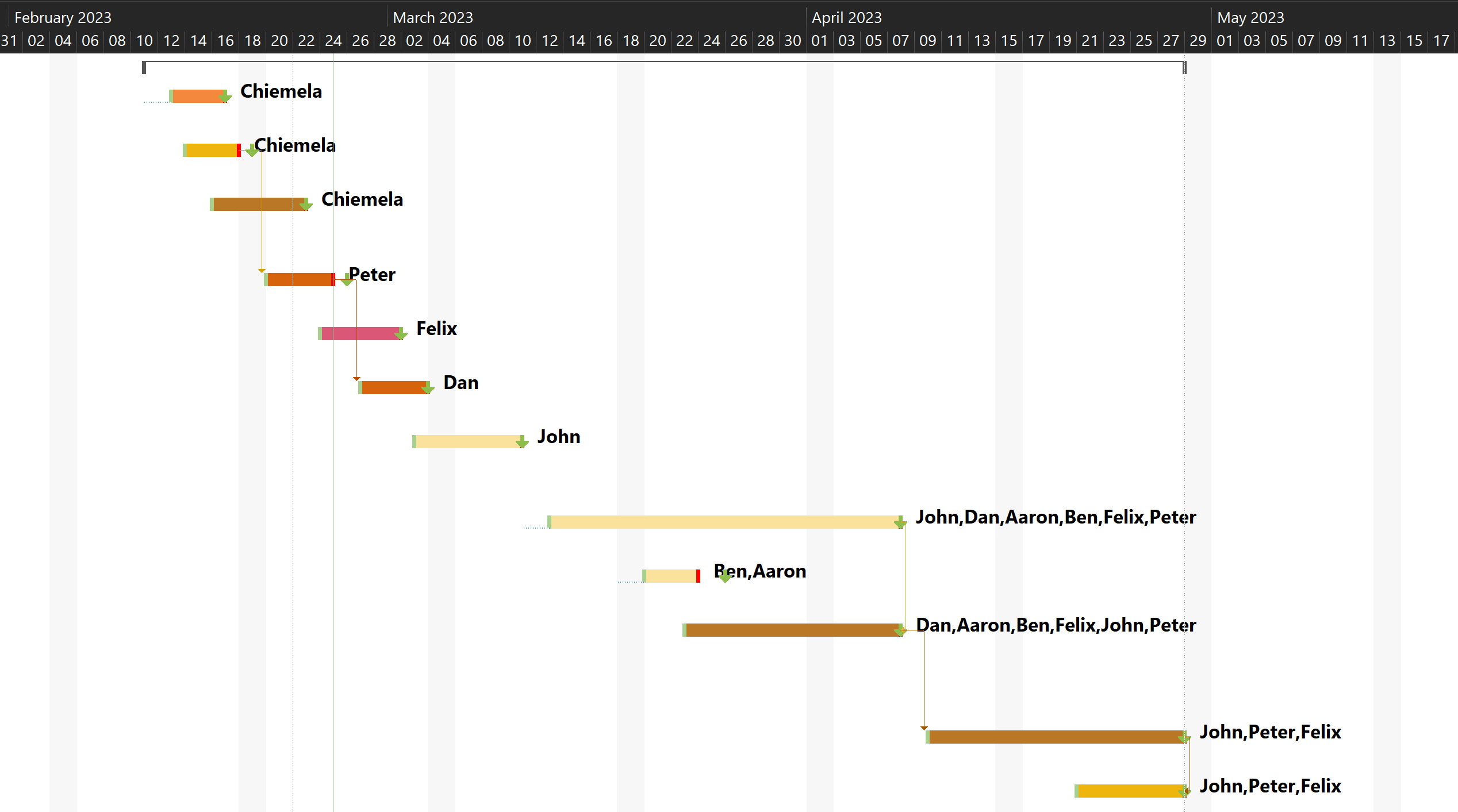
  
Figure 11: WBS

## 5.2 Work breakdown structure rationale

The Work Breakdown Structure (WBS) is an essential part of project planning as it breaks down the entire project into smaller, more manageable components. The WBS allows for a more detailed understanding of the project scope, schedule, and budget. It helps to identify all the tasks and deliverables required to complete the project successfully. The WBS provides a framework for project management that makes it easier to manage the project and keep track of progress. The breakdown structure provides clarity in terms of the project's objectives and goals, while also providing a roadmap to achieving them. The WBS enables the team to focus on the specific tasks that need to be completed, which helps to ensure that the project is completed on time, within budget, and to the expected quality. Additionally, the WBS can serve as a communication tool between the project team and stakeholders, helping to ensure that everyone has a shared understanding of the project scope and requirements. Overall, the WBS is a critical tool for project planning and execution, helping to ensure that the project is completed successfully.

## 5.3 Gantt chart

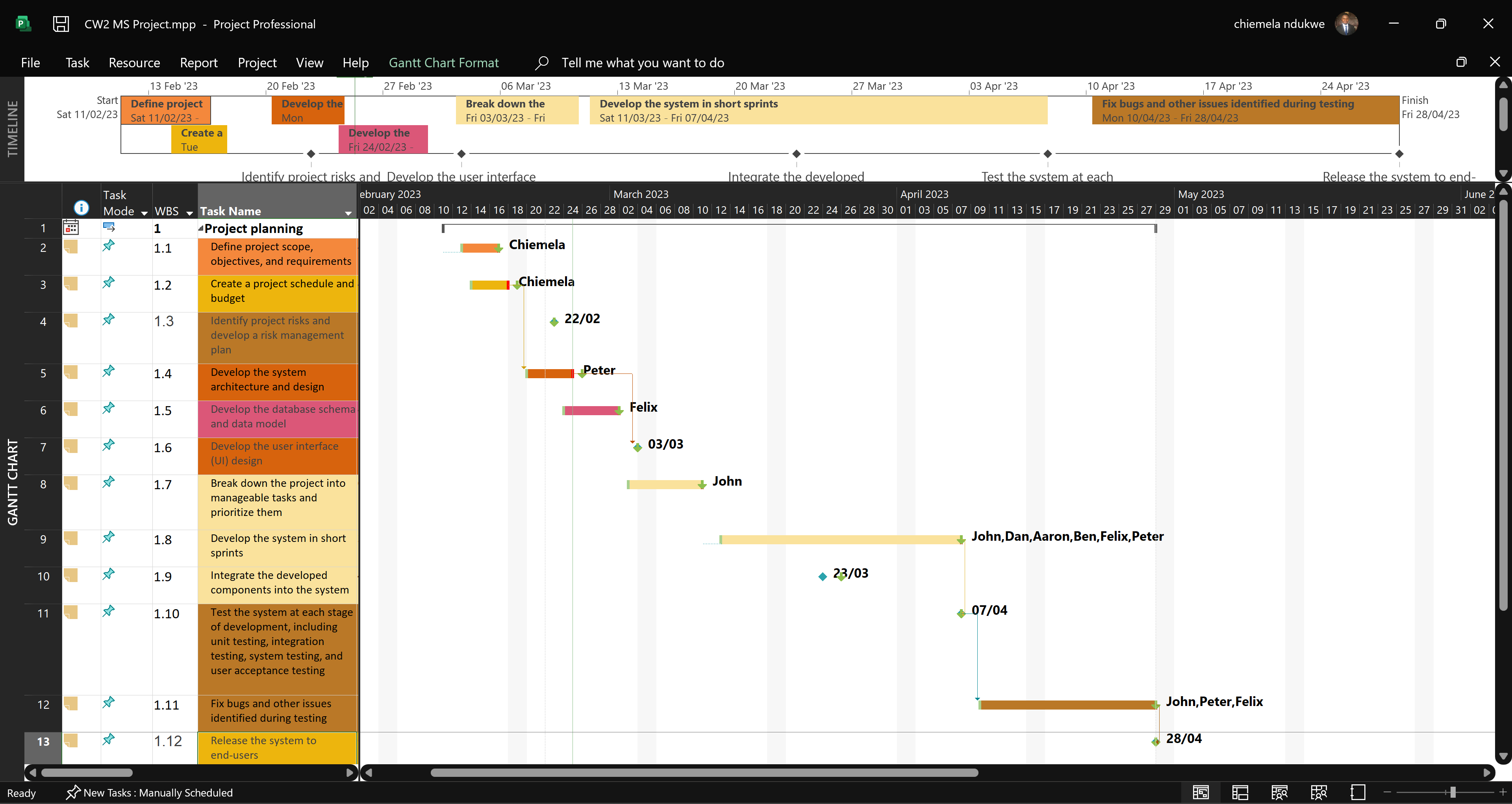
The Gantt Chart is a visual tool used in project management to represent the project schedule. It provides a timeline view of the project tasks, their dependencies, and their duration. For this project, the Gantt Chart would be an essential tool in tracking progress and ensuring that the project is completed on time. The Gantt Chart for this project would include all the tasks identified in the Work Breakdown Structure (WBS). Each task would be represented as a horizontal bar, with the length of the bar representing the duration of the task. The Gantt Chart would also indicate the dependencies between the tasks, with arrows connecting the bars to show which tasks need to be completed before others can start. Additionally, the Gantt Chart can also be used to track progress. As the project progresses, the Gantt Chart can be updated to show the actual progress made against the planned schedule. This can help the project manager identify any delays or issues and take corrective action as necessary. The Gantt Chart would also be an essential communication tool between the project team and stakeholders. It provides a clear and easy-to-understand view of the project schedule, which can help stakeholders understand the project timeline and when they can expect to see deliverables. In a whole, the Gantt Chart is a critical tool in project management, providing a visual representation of the project schedule and helping to ensure that the project is completed on time.

  
Figure 12: Gantt Chart.

## 5.4 Project milestones

Milestones are important points of progress in a project that represent the completion of a major phase or achievement. They are useful in project management as they provide a way to track and measure progress, and to identify any delays or issues that need to be addressed. These milestones are represented on a Gantt chart to provide a visual representation of the project timeline and the progress towards each milestone. By identifying and tracking milestones, the project manager can ensure that the project is staying on track and meeting its objectives and can take corrective action if necessary to address any delays or issues. The milestones are as follows:

* Identify project risks and develop a risk management plan.
* Develop the user interface (UI) design.
* Integrate the developed components into the system.
* Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing.
* Release the system to end-users.

  
Figure 13: Project milestones.

## 5.5 Project milestones rationale

The project milestone rationale outlines the reasons for selecting specific milestones in a project plan. Milestones are important because they provide clear targets for progress, help keep the project on track, and allow stakeholders to measure progress and assess the success of the project. By identifying and prioritizing these milestones, the project team can focus their efforts on the most critical components of the project, and stakeholders can assess progress and make informed decisions about the project's direction and resource allocation. In the case of an event management system project, the following are the rationales for the selected milestones:

* Identify project risks and develop a risk management plan: This milestone is critical for ensuring that the project is completed on time, within budget, and meets quality standards. By identifying potential risks early on and developing a plan to mitigate them, the project team can minimize the impact of these risks and avoid costly delays or rework.
* Develop the user interface (UI) design: This milestone is important for ensuring that the system is user-friendly and meets the needs of end-users. A well-designed UI can improve user adoption and satisfaction and reduce the need for training or support.
* Integrate the developed components into the system: This milestone is necessary for ensuring that the system functions as intended and that all components work together seamlessly. Integration testing is essential for identifying and resolving any issues that arise during the integration process.
* Test the system at each stage of development, including unit testing, integration testing, system testing, and user acceptance testing: Testing is critical for ensuring that the system meets functional and non-functional requirements, is free of defects, and is reliable and secure. Testing at each stage of development helps to identify and resolve issues early on, which can save time and reduce costs.
* Release the system to end-users: This milestone marks the successful completion of the project and the delivery of a working system to end-users. It is important to ensure that the system is thoroughly tested and meets all requirements before it is released to minimize the risk of issues arising after deployment.

# Chapter 6: Risk Management and Software Quality Management Strategies

## 6.1 Risk identification

The risk identification process involves identifying potential risks that could impact the success of the project. Some of the risks identified for this project are as follows:

**Technical Risks**

The project involves developing a complex event management system that requires sophisticated technical skills. Technical risks include the possibility of software bugs, hardware malfunctions, or network issues.

**Scope Creep**

This is a common risk that can occur in any project. Scope creep is the gradual expansion of the project beyond its original scope. This can result in delays, budget overruns, and the failure to meet project objectives.

**Resource Constraints**

The project budget and resources are limited, which could impact the timely completion of the project. Resource constraints could include a lack of skilled personnel, inadequate funding, or insufficient technology.

**Stakeholder Management**

Effective stakeholder management is critical to the success of any project. This risk involves ensuring that the project meets the needs and expectations of all stakeholders, including customers, project sponsors, and team members.

**Change Management**

Changes are an inevitable part of any project. Managing changes effectively requires a formal change management process to ensure that changes are documented, approved, and implemented in a controlled manner.

**Integration Risks**

The project involves integrating several components, which could result in integration risks. Integration risks include the possibility of data loss, software conflicts, or system downtime.

**Security Risks**

The event management system will store sensitive data, which could make it a target for cyber-attacks. Security risks could include data breaches, hacking, or other forms of cybercrime.

**User Acceptance**

User acceptance is a critical success factor for the project. This risk involves ensuring that the end-users are satisfied with the system's functionality, usability, and performance. Failure to achieve user acceptance could result in project failure.

## 6.2 Risk analysis

Risk analysis is the process of evaluating the likelihood and potential impact of identified risks. The aim of risk analysis is to prioritize risks, determine the severity of the consequences, and develop strategies to mitigate or avoid the risks. In the context of the event management system project, the risk analysis involves assessing the potential risks that could affect the project's success, as well as the likelihood of those risks occurring. The risk analysis process will help the team to identify and prioritize potential risks, develop strategies to mitigate or avoid those risks, and ensure that the project stays on track and within budget. The risks that have been identified are as follows:

**Technical Risks**

These risks include problems with software, hardware, and other technical issues that could cause delays or failures in the project. To mitigate these risks, the team will conduct regular testing and quality assurance checks throughout the development process.

**Personnel Risks**

These risks include issues with team members such as illness, turnover, or loss of expertise. To mitigate these risks, the team will ensure that there is sufficient coverage for each team member's responsibilities and provide cross-training to ensure that all team members are familiar with the project's requirements.

**Financial Risks**

These risks include budget overruns, unexpected expenses, and other financial issues that could disrupt the project's progress. To mitigate these risks, the team will closely monitor the project budget and track expenses to ensure that they are within budget.

**Schedule Risks**

These risks include delays or changes to the project schedule, which could impact the project's completion date. To mitigate these risks, the team will regularly review and update the project schedule, communicate any changes to stakeholders, and adjust the project plan as needed.

## 6.3 Risk planning

Risk planning involves identifying potential risks and creating a plan to mitigate or manage them. The following steps will be taken in risk planning. By following these steps, a comprehensive risk management plan can be developed to minimize the impact of potential risks on the project.

**Risk identification**

This involves identifying all possible risks that could impact the project.

**Risk assessment**

After identifying the risks, they need to be assessed to determine their likelihood and impact on the project.

**Risk prioritization**

The risks should be prioritized based on their level of impact and likelihood, and a plan should be created to address the high-priority risks.

**Risk response planning**

This involves developing a plan to mitigate or manage the identified risks. This may include developing contingency plans, transferring risks, avoiding risks, or accepting risks.

**Risk monitoring and control**

The risks should be monitored throughout the project, and any changes or new risks should be assessed and managed accordingly.

## 6.4 Risk monitoring

Risk monitoring is an ongoing process that involves tracking, reviewing, and updating the risk management plan as needed throughout the project. This is to ensure that risks are managed effectively and that the project remains on track. By monitoring risks throughout the project, the project team can take proactive steps to minimize the impact of risks on the project and ensure its successful completion. The risk monitoring process steps are as follows:

**Tracking risks**

This involves keeping track of identified risks and their status. The risks should be monitored regularly to determine if they have changed in likelihood or impact.

**Reviewing risk responses**

The risk management plan should be reviewed periodically to ensure that risk responses are still appropriate and effective.

**Updating the risk management plan**

As the project progresses, new risks may arise, and existing risks may change. The risk management plan should be updated to reflect any changes.

**Reporting risks**

Any significant changes in risks should be reported to the project team and stakeholders, along with any recommended actions.

# Conclusion

In conclusion, the event management system project requires careful planning, organization, and execution to achieve its objectives successfully. The project plan includes the project scope, goals, objectives, budget breakdown structure, work breakdown structure, and Gantt chart, among others. The team structure is well-defined, and the project manager will oversee the team's activities to ensure the project's success. The project milestones, including identifying project risks and developing a risk management plan, developing the user interface (UI) design, integrating the developed components into the system, testing the system, and releasing it to end-users, are crucial in ensuring the project's success. By following the project plan and addressing identified risks, the project team can achieve the project's objectives within the set timeframe and budget.

# Bibliography

ISO/IEC/IEEE, 2017. *Systems and software engineering -- Work breakdown structures for software systems.,* s.l.: ISO/IEC/IEEE 12207.

Smith, J., 2018. *Risk Management in Projects.* s.l.:Routledge.

Smith, J., 2020. *Work Breakdown Structure: Definition, Examples and Uses.* [Online]   
Available at: https://www.projectmanager.com/work-breakdown-structure  
[Accessed 2023].

Smith, J, 2022. *Project Schedule and Budget for XYZ Project..* [Online]   
Available at: https://www.example.com/project-schedule-budget-xyz-project  
[Accessed 2023].