1. **Analysis and Identification of the suitable process models.**

1. Define the scope and boundaries of your project. What parts of the process are included, and what are the intended outcomes?

|  |  |
| --- | --- |
| **Project Scope Statement** | |
| **Project Name:** | **FINALyze** |
| **Description:** | The web application calculates the Gross margin, Net margin and ROI for products.  This app provides insight into products Finance health and Analyse its Investment decision. |
| **Deadline:** | 29th March 2025 |
| **Student Name:** | Mohammed Aiman Khan |
| **Goal:** | To create a user specific web app to calculate gross & net margin for product and analysing the Return on Investment for user for each product |
| **Deliverable(s):** | * Software Requirement Specification * Project Plan * High Level Design * Low Level Design * Web App * Unit Test Cases and Report * Integration Test Cases and Report * Deployment Procedure |
| **Tasks: (In-Scope Items)** | * User Registration * User Login * Financial Data entry for product * Profit and ROI Report * To be setup in local host * Unit testing * Integration testing * GitHub Repository |
| **Out-of-scope Items:** | * Master data Management * Data Analysis * AI and suggestions * Hosting * Mobile and Tablet app   Project Costing  Communication details  Scalability and portability |
| **Constraints:** | * Need to have a working prototype by mid-march and this semester being only 3 working months. * Resource limitation – knowledge limitation based on framework finalized |
| **Challenges** | * Framework Identification |
| **Risks** | * Framework choice can have huge impact on scope and time. |

2. **Clarify Your Goals:**

• List your primary objectives

• Set measurable target outcomes: Aim for specific, achievable benchmarks to assess your success.

**PRIMARY OBJECTIVES:**

1. **Adhere to all Project Management Process**
   * **Create needed documentation and plans for**
     1. **Scope**
     2. **Time**
     3. **Cost (Excluded)**
2. **Develop a Functional Web Application**:
   * Create a user-friendly interface that allows users to input product-related data
   * Implement calculation formulas for **Gross Margin**, **Net Margin**, and **ROI**.
   * Ensure accurate results based on the user’s inputs.
3. **Ensure Accurate Calculations**:
   * As per the given formulas in the excel sheet
4. **Optimize User Experience**:
   * Ensure the app is responsive, easy to navigate.
5. **Scalability & Security**:
   * <not in scop>
6. **Feedback & Iteration**:
   * Collect feedback from professor in charge and in cooperate accordingly

**MEASURABLE TARGET OUTCOMES:**

1. **Accuracy of Calculations**:
   * Achieve 100% accuracy in Gross Margin, Net Margin, and ROI calculations.
   * Ensure that all formulas are correctly implemented and tested with various real-world data sets.
2. **User Adoption and Engagement**:
   * Aim for good user interface.
3. **Usability and Satisfaction**:
   * Create a MVP to replace current Excel sheet
4. **Security Standards**:
   * User can see only their related products

**3. Assess Your Constraints:**

• Define the available timeline: What is the deadline for achieving your desired outcomes?

• Consider resource limitations: What skillsets are available to execute the process?

**CONSTRAINT1:** Need to have a working prototype by mid-march and this semester being only 3 working months.

**ASSESSMENT:**

* **Tight Deadline**: With only 3 months (approximately 12 weeks) for development, having a working prototype ready by mid-March is a challenge. This leaves limited time for thorough planning, development, testing, and iteration.
* **Prioritization Needed**: The team will need to focus on delivering the core functionality first—calculating Gross Margin, Net Margin, and ROI—and potentially save secondary features (like advanced visualization, advanced UI/UX, or additional integrations) for later versions or post-launch improvements.
* **Risk of Compromising Features**: In such a short timeframe, some features may need to be compromised or excluded to meet the prototype deadline.
* **Testing & Debugging**: Time for testing and debugging might be limited, which could impact the quality of the initial prototype. It's crucial to allocate specific time for this to ensure accuracy in the calculations and basic functionality.

**RECOMMENDATIONS:**

* **Agile Approach**: Use an **Agile** or **Iterative** development model, which allows for flexibility in adjusting scope and features as you progress. You can focus on core features first and release them as you complete each sprint.
* **MVP (Minimum Viable Product)**: Prioritize the most essential features required to meet the core goals of your prototype. This ensures that you have a working version by mid-March, even if some enhancements or advanced features are deferred.
* **Set Milestones**: Break the 3 months into smaller phases (e.g., planning, development, testing, iteration) with clear milestones. This will keep the project on track and ensure timely delivery.

**CONSTRAINT 2:** Resource limitation – knowledge limitation based on framework finalized

**ASSESSMENT:**

 **Limited Framework Choices**: The chosen framework might restrict the types of technologies or tools you can use for development. If you have already finalized the framework, the resource limitations could mean that certain libraries, features, or functionalities might not be easily integrated.

 **Skillset Constraints**: The team's expertise with the chosen framework could impact development speed. If the framework is unfamiliar or complex, it may lead to learning curves, reducing productivity

**RECOMMENDATIONS:**

* **Evaluate the Chosen Framework**: Ensure that the framework supports the core features you need (e.g., calculations, UI design, scalability). If it's too limiting, consider alternatives that align better with project goals, or investigate potential plugins/extensions that may help.
* **Optimize Resources**: Ensure to quickly learn the needed skill by referring training materials and tutorial.
  1. **Additional Considerations (Optional):**

• Mention any specific requirements or preferences you have for the process model (e.g., emphasis on flexibility, need for strong risk management).

• Provide any further details relevant to your project or desired process.

**Strong Focus on Risk Management:**

* **Risk Identification**: Early identification of risks is crucial due to the constraints on time and resources. The team should assess potential risks such as technical challenges with the framework, performance issues with calculations (Gross Margin, ROI), or delays due to lack of familiarity with the chosen tools.
* **Proactive Mitigation**: Include risk mitigation strategies within the development phases. That is to care for
  + **Technology Risk**: Mitigate by selecting well-documented frameworks and tools that the team is familiar with or can quickly learn.

**5. Identify the suitable process model for your project. Justify your answer**

**AGILE MODEL**

* **Overview**:   
  The Agile model is iterative and incremental. It allows for flexibility in the development process, making it easy to adapt to changing requirements over time. The project is broken down into smaller modules or sprints, each producing a functional part of the web application.
* **Why it fits**:  
   Project requires frequent updates as it is based on lots of calculations, feedback integration as it being developed with evolving requirements, Hence Agile model would be ideal.
* Agile emphasizes collaboration with stakeholders, delivering features in increments, and flexibility in adapting to changes.
* **Key Phases**:
  1. **Sprint Planning**: Define a list of features or tasks for the sprint (e.g., developing calculation logic, user interface design).
  2. **Design & Development**: Build and implement the features planned for that sprint.
  3. **Testing**: Perform testing after each sprint to ensure functionality and usability.
  4. **Review & Feedback**: Review results with stakeholders and incorporate feedback.
  5. **Repeat**: Continue with the next sprint until the application is complete.
  6. **For this project, you will be using GitHub to manage your code.**

• Create a new GitHub repository for this project.

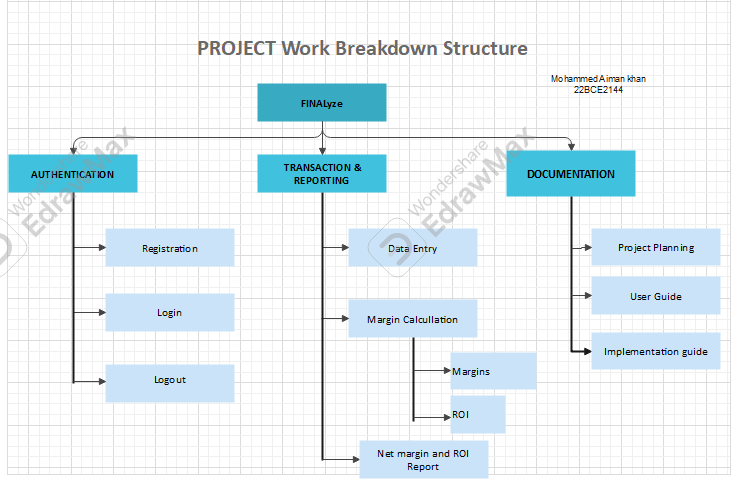
• Commit your code to the repository frequently, using descriptive commit messages.

• Submit the link to your GitHub repository now and during your final project demo

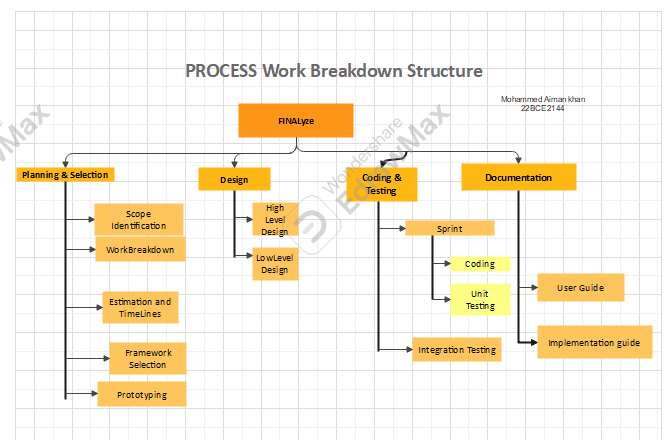
<https://github.com/Aiman-VIT/Software-engineering-project>

1. **Work Break-down Structure & Estimation**

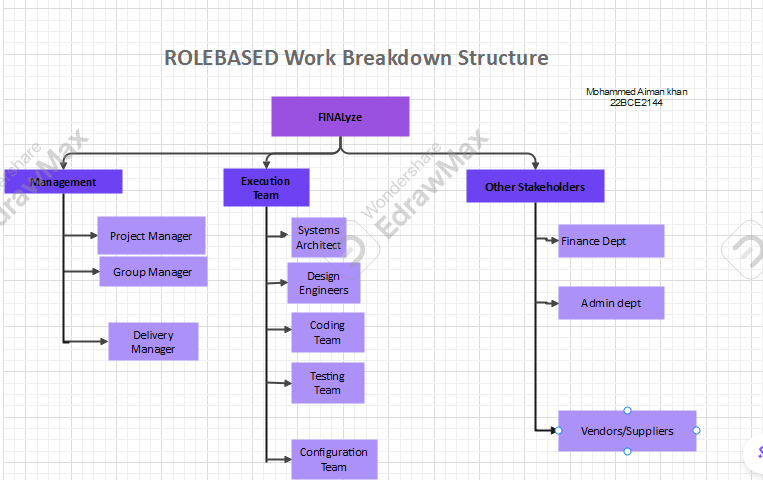
**PRODUCT BASED:**

****

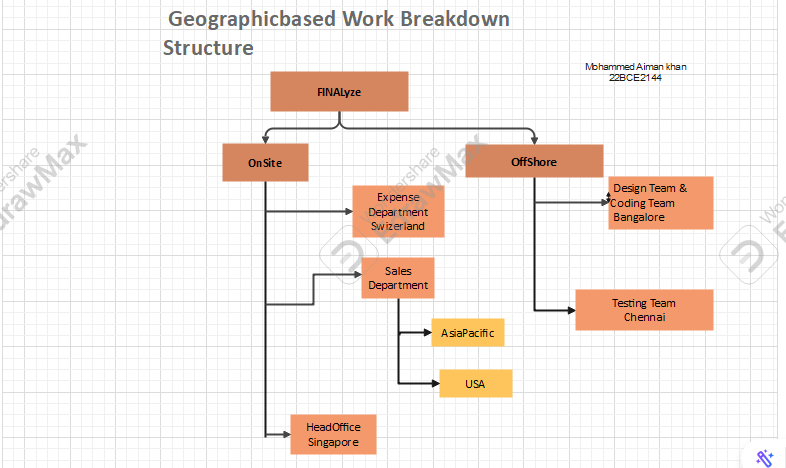
**PROCESS BASED:**

****

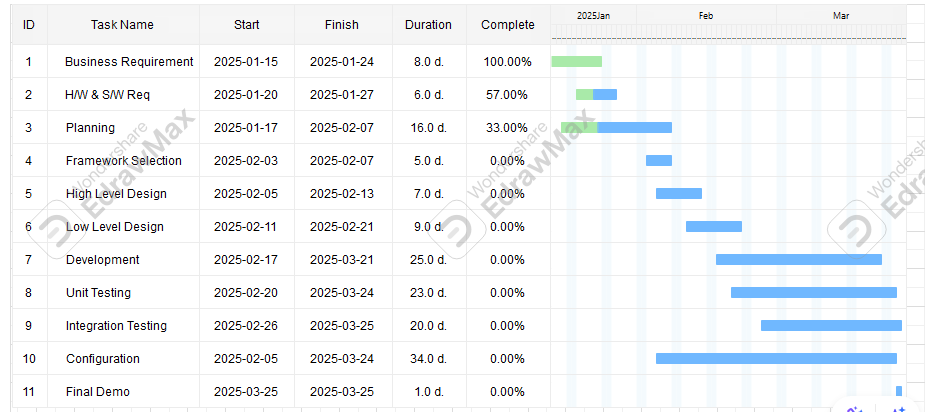
**ROLE BASED:**

****

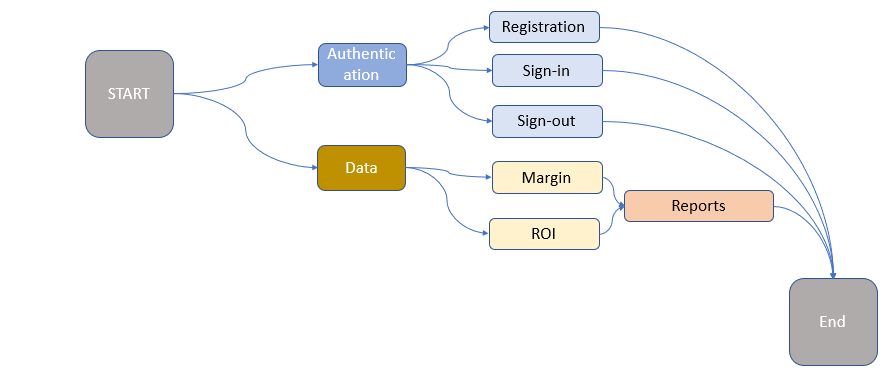
**GEOGRAPHIC BASED:**

****

**GANTT chart**

****

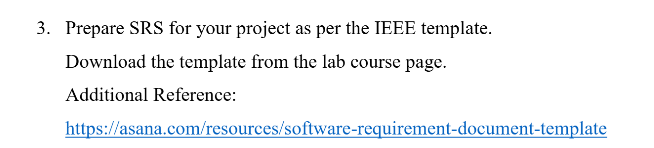
**PERT chart**

****

**Estimation:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Bottom-Up Approach Estimation** | | | |
| **Activities** | **Days** | **hrs/day** | **Effort (hrs)** |
| Requirement gathering | 7 | 2 | 14 |
| H/W & S/W Req | 10 | 1 | 10 |
| PLANNING | 16 | 0.25 | 4 |
| FRAMEWORK SELECTION | 5 | 2 | 10 |
| HIGH LEVEL DESIGN | 8 | 0.5 | 4 |
| LOW LEVEL DESIGN | 9 | 0.75 | 6.75 |
| CODING | 25 | 1 | 25 |
| UNIT TESTING | 22 | 0.25 | 5.5 |
| INTEGRATION | 10 | 0.5 | 5 |
| CONFIGURATION Mgmt | 30 | 0.1 | 3 |
| FINAL DEMO | 1 | 2 | 2 |
| FEEDBACK | 1 | 5 | 5 |
| CLOUSER | 1 | 5 | 5 |
| TOTAL HOURS |  |  | **99.25** |
| MAN DAYS (4hrs = 1md) |  |  | **24.8125** |

**III SRS: software requirement specifications**

****

****