|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Activity #** | **Activity Name** | **Brief Description** | **Duration** | **Responsible Team Members** |
| 1. **Initiating** | 1.1 Project Objective | 1. **Understand the Project Scope**: Clearly define the boundaries and deliverables of the project. [This helps in setting realistic objectives](https://www.atlassian.com/work-management/project-management/project-objectives). 2. [**Identify Stakeholder Needs**: Gather input from all stakeholders to understand their expectations and requirements](https://www.atlassian.com/work-management/project-management/project-objectives). 3. [**Set SMART Objectives**: Ensure that the objectives are Specific, Measurable, Achievable, Relevant, and Time-bound](https://asana.com/resources/how-project-objectives). 4. [**Align with Business Goals**: Make sure the project objectives align with the broader business goals and strategies](https://www.atlassian.com/work-management/project-management/project-objectives). 5. **Document Objectives**: Write down the objectives clearly and concisely. [This helps in communicating them effectively to the project team and stakeholders](https://www.atlassian.com/work-management/project-management/project-objectives). 6. [**Review and Validate**: Get feedback from stakeholders and make necessary adjustments to ensure the objectives are realistic and agreed upon](https://www.atlassian.com/work-management/project-management/project-objectives). | **(5 Days)** | **All Team Members**  Kate as Lead Chief Officer |
|  | 1.2 Project Stakeholders | 1. **Identify Stakeholders**: Determine who the stakeholders are. [This includes anyone who has an interest in or is affected by the project](https://www.teamwork.com/project-management-guide/project-stakeholders/). 2. **Analyze Stakeholders**: Understand each stakeholder’s interests, influence, and impact on the project. [This helps prioritize their needs and manage their expectations](https://www.projectmanager.com/blog/stakeholder-analysis-101). 3. **Engage Stakeholders**: Develop strategies to engage stakeholders based on their level of interest and influence. [This might include regular meetings, updates, and feedback sessions](https://bing.com/search?q=steps+in+Project+Stakeholders). 4. **Manage Stakeholder Expectations**: Continuously communicate with stakeholders to manage their expectations and address any concerns. [This helps build trust and ensures their support throughout the project](https://www.projectmanager.com/blog/stakeholder-analysis-101). 5. [**Monitor and Review**: Regularly review stakeholder engagement and adjust as needed to ensure ongoing alignment with project goals](https://www.teamgantt.com/guide/stakeholder-management) | **(14 Days)** | **All Team Members**  Kate as Lead Chief Officer |
|  | 1.3 Project Scope | 1. **Identify Project Objectives**: Clearly define what the project aims to achieve. [This includes setting specific, measurable, achievable, relevant, and time-bound (SMART) goals](https://www.indeed.com/career-advice/career-development/project-scope-example). 2. [**Gather Requirements**: Collect all the necessary requirements from stakeholders to understand what needs to be delivered](https://www.indeed.com/career-advice/career-development/project-scope-example). 3. **Define Deliverables**: List all the deliverables that the project will produce. [This helps in understanding the scope of work](https://www.indeed.com/career-advice/career-development/project-scope-example). 4. [**Create a Work Breakdown Structure (WBS)**: Break down the project into smaller, manageable tasks and activities](https://www.indeed.com/career-advice/career-development/project-scope-example). 5. [**Identify Exclusions**: Clearly state what is not included in the project to avoid any misunderstandings later](https://asana.com/resources/project-scope). 6. [**Establish Constraints**: Identify any limitations or constraints such as budget, time, and resources](https://asana.com/resources/project-scope). 7. [**Assumptions**: Document any assumptions that are made during the planning phase](https://asana.com/resources/project-scope). 8. [**Get Stakeholder Approval**: Ensure that all key stakeholders review and approve the project scope to confirm alignment and understanding](https://asana.com/resources/project-scope). | **(5 Days)** | **All Team Members**  Kate as Lead Chief Officer |
| **2 - Planning** | 2.1 - Work Breakdown Structure | 1. **Hierarchy**: The WBS is structured in a hierarchical manner, starting with the overall project at the top and breaking it down into smaller deliverables and tasks. 2. **Deliverables**: Each level of the WBS represents a more detailed breakdown of the project deliverables. 3. **Phases**: The project can be divided into phases, with each phase containing specific tasks and deliverables. 4. [**100% Rule**: This rule ensures that the WBS includes all the work required for the project, with no overlap or omission](https://asana.com/resources/work-breakdown-structure). | **(4 Days)**  Continues Updating &  End of Section Updates | **All Team Members**  Kate as Lead Chief Officer |
|  | 2.2 - Activity Definition Estimation | 1. **Purpose**: To determine the amount of time needed for each activity, which helps in developing the project schedule and estimating costs. 2. [**Factors**: Several factors influence these estimates, including the complexity of the activity, resource availability, team expertise, risk assessment, and external dependencies1](https://everythingprojectmanagement.com/learn/managing-project-schedules/activity-duration-estimation/). 3. **Techniques**: Common techniques include:    * **Expert Judgment**: Leveraging the experience of experts to estimate durations.    * **Analogous Estimating**: Using historical data from similar projects.    * **Parametric Estimating**: Applying statistical relationships between historical data and other variables.    * **Three-Point Estimating**: Considering optimistic, pessimistic, and most likely scenarios to calculate an average duration.    * [**Bottom-Up Estimating**: Breaking down activities into smaller components and estimating each one](https://project-management.info/estimating-activity-durations/) | **(5 days)**  Continues Updating &  End of Section Updates | **All Team Members**  Neveen as Lead Chief Officer |
|  | 2.3 - Gant Chart | 1. **Tasks and Time**: Tasks are listed on the vertical axis, and time intervals are shown on the horizontal axis. 2. **Bars**: Each bar represents a task, with the length of the bar indicating the duration of the task. 3. [**Dependencies**: Modern Gantt charts can show dependencies between tasks, indicating which tasks need to be completed before others can start](https://www.bing.com/aclick?ld=e8NrFjZ7KoKMumOoDcWwjsgTVUCUys-NRRXe-xFLuFyEjIvsyrRMrnwR-elgKkA65qcb9urTX0xb3LvlA_iZv2Z8PomI51BzCXiBtVW8pxwOD6BvPHg6Z1_af6mvQxIzUL9ODZ4cGXQkf-EuOApc_AwtjycLcrGNNBVONsgyKAPBAQzC1l&u=aHR0cHMlM2ElMmYlMmZhc2FuYS5jb20lMmZnbyUyZnRpbWVsaW5lJTNmdXRtX2NhbXBhaWduJTNkJTI2dXRtX3NvdXJjZSUzZCUyNnV0bV9tZWRpdW0lM2QlMjZnY2xpZCUzZDA3MzY5NjMwZmY3MjExNGRhMTI1ZjBlYWFlOWRhNzhjJTI2Z2Nsc3JjJTNkM3AuZHMlMjZtc2Nsa2lkJTNkMDczNjk2MzBmZjcyMTE0ZGExMjVmMGVhYWU5ZGE3OGMlMjZ1dG1fdGVybSUzZHRpbWVsaW5lJTI1MjBnYW50dCUyNTIwY2hhcnQlMjZ1dG1fY29udGVudCUzZEdhbnR0JTI1MjBDaGFydA&rlid=07369630ff72114da125f0eaae9da78c). 4. [**Progress Tracking**: They often include features to track the progress of tasks, such as shading or color-coding](https://asana.com/resources/gantt-chart-basics). | **(4 days)**  Continues Updating &  End of Section Updates | **All Team Members**  Naveen as Lead Chief Officer |
| **3- Executing** | 3.1 – Problem Background | **System capabilities/overview**   1. [**Identify Stakeholder Needs**: Gather and analyze the needs and requirements of all stakeholders to understand what the system must accomplish](https://www.dau.edu/sites/default/files/2024-02/Systems%20Engineering%20Guidebook.pdf). 2. [**Define System Objectives**: Clearly outline the goals and objectives of the system, ensuring they align with stakeholder needs and organizational goals](https://www.dau.edu/sites/default/files/2024-02/Systems%20Engineering%20Guidebook.pdf). 3. [**Develop System Requirements**: Translate stakeholder needs into detailed system requirements, specifying what the system should do and how it should perform](https://www.dau.edu/sites/default/files/2024-02/Systems%20Engineering%20Guidebook.pdf). 4. [**Create a System Architecture**: Design the overall structure of the system, including its components, their relationships, and how they interact](https://www.dau.edu/sites/default/files/2024-02/Systems%20Engineering%20Guidebook.pdf). 5. [**Define Capabilities**: Specify the key capabilities of the system, detailing the functions it will perform and the performance criteria it must meet](https://www.dau.edu/acquipedia-article/systems-engineering-process). 6. [**Develop a Concept of Operations (CONOPS)**: Describe how the system will be used in real-world scenarios, including its operational environment and user interactions](https://www.dau.edu/acquipedia-article/systems-engineering-process). 7. **Validate and Verify Requirements**: Ensure that the system requirements are complete, feasible, and testable. [This involves reviewing and validating requirements with stakeholders](https://www.dau.edu/acquipedia-article/systems-engineering-process). 8. [**Document the Overview**: Compile all the information into a comprehensive document that provides a clear overview of the system’s capabilities, architecture, and operational concept](https://www.dau.edu/acquipedia-article/systems-engineering-process).   **Benefits and Analysis**   1. **Establish a Framework**: Define the scope and objectives of the analysis. [This includes identifying what you aim to achieve and the criteria for success](https://online.hbs.edu/blog/post/cost-benefit-analysis). 2. **Identify Costs and Benefits**: List all potential costs and benefits associated with the project or decision. [This includes both direct and indirect impacts](https://online.hbs.edu/blog/post/cost-benefit-analysis). 3. **Assign Values**: Quantify the costs and benefits in monetary terms where possible. [This helps in comparing them on a common scale](https://online.hbs.edu/blog/post/cost-benefit-analysis). 4. **Compare Costs and Benefits**: Calculate the total value of the benefits and subtract the total costs. [This can be done using various methods such as net present value (NPV) or cost-benefit ratio](https://online.hbs.edu/blog/post/cost-benefit-analysis). 5. **Analyze Results**: Interpret the results to determine whether the benefits outweigh the costs. [Consider both quantitative and qualitative factors](https://online.hbs.edu/blog/post/cost-benefit-analysis). 6. [**Make Recommendations**: Based on the analysis, make informed recommendations on whether to proceed with the project or decision](https://online.hbs.edu/blog/post/cost-benefit-analysis). | **(3 days)**  Continues Updating &  End of Section Updates  **(5 days)**  Continues Updating &  End of Section Updates | **All Team Members**  Kate as Lead Chief Officer |
|  | 3.2 - User Requirements | **Use Case Diagram**   1. **Identify Actors**: Determine the external entities that will interact with the system. [These can be users, other systems, or hardware devices](https://www.geeksforgeeks.org/use-case-diagram/). 2. **Identify Use Cases**: Define the specific functions or actions that the system will perform. [Each use case represents a goal that an actor wants to achieve](https://www.geeksforgeeks.org/use-case-diagram/). 3. **Establish Relationships**: Determine the relationships between actors and use cases. [This includes identifying which actors are involved in which use cases and how they interact](https://www.businessanalysisinsights.com/diagramming-tools-use-case-diagrams). 4. **Draw the Diagram**: Use a UML (Unified Modeling Language) tool or draw by hand to create the diagram. Place actors outside the system boundary and use cases inside. [Connect actors to their respective use cases with lines](https://www.geeksforgeeks.org/use-case-diagram/). 5. **Review and Validate**: Ensure that the diagram accurately represents the system’s functionality and interactions. [Review it with stakeholders to confirm its accuracy and completeness](https://creately.com/guides/use-case-diagram-tutorial/).   **Use Case**   1. **Identify Actors**: Determine who will interact with the system. [Actors can be users, other systems, or devices](https://blog.logrocket.com/product-management/what-is-a-use-case-template-how-to-write/). 2. **Define Use Cases**: Identify the specific actions or functions that the system will perform. [Each use case represents a goal that an actor wants to achieve](https://blog.logrocket.com/product-management/what-is-a-use-case-template-how-to-write/). 3. **Establish Relationships**: Determine how actors and use cases interact. [This includes identifying which actors are involved in which use cases and how they interact](https://blog.logrocket.com/product-management/what-is-a-use-case-template-how-to-write/)[3](https://online.visual-paradigm.com/diagrams/tutorials/use-case-diagram-tutorial/). 4. [**Outline Preconditions**: Specify any conditions that must be met before the use case can be initiated](https://blog.logrocket.com/product-management/what-is-a-use-case-template-how-to-write/)[2](https://www.figma.com/resource-library/what-is-a-use-case/)[4](https://www.adaptiveus.com/blog/technique/use-case-and-senarios/). 5. [**Describe the Basic Flow**: Detail the standard sequence of steps that occur when the use case is executed successfully](https://www.figma.com/resource-library/what-is-a-use-case/). 6. [**Identify Alternate Flows**: Outline any variations or exceptions to the basic flow, including error conditions and alternative paths](https://www.figma.com/resource-library/what-is-a-use-case/). 7. [**Document Postconditions**: Define the state of the system after the use case has been completed](https://www.adaptiveus.com/blog/technique/use-case-and-senarios/). 8. **Review and Validate**: Ensure that the use case accurately represents the system’s functionality and interactions. [Review it with stakeholders to confirm its accuracy and completeness](https://www.figma.com/resource-library/what-is-a-use-case/). | **(3 days)**  Continues Updating &  End of Section Updates  **(2 days)**  Continues Updating &  End of Section Updates | **All Team Members**  I Wibawa as Lead Chief Officer |
|  | 3.3 - Structural Design | **Software Development**   1. System architecture design 2. Database schema design 3. UI/UX wireframing 4. Design review and approval   **Frontend Development**   1. Set up frontend framework 2. Implement user interface components 3. Develop search and results display 4. Create visualization components (charts and graphs) 5. Implement comparison view 6. Develop meal plan generator interface   **Backend Development**   1. Set up development environment 2. Implement data loading from CSV 3. Develop search functionality 4. Implement filtering and categorization logic 5. Create API endpoints for frontend communication   **Data Management**   1. Data cleaning and preprocessing 2. Database setup and configuration 3. Data import and validation 4. Implement data update mechanism | **10 days**  (3 days)  (2 days)  (3 days)  (2 days)  **25 Days**  (2 days)  (5 days)  (5 days)  (6 days)  (4 days)  (3 days)  **20 Days**  (1 day)  (3 days)  (5 days)  (7 days)  (4 days)  **8 Days**  (3 days)  (1 day)  (3 days)  (1 day) | **All Team Members**  I Wibawa as Lead Chief Officer |
|  | 3.4 - Documentations | **API Documentations**   1. **Create an Outline**: Start by developing a detailed outline of what the documentation will cover. [This helps organize the content and ensures all necessary topics are included](https://webflow.com/blog/api-documentation). 2. **Select a Platform**: Choose a platform or tool for hosting your documentation. [This could be a dedicated documentation site, a wiki, or a platform like GitHub](https://webflow.com/blog/api-documentation). 3. [**Write the Introduction**: Provide an overview of the API, including its purpose, key features, and how it can be used](https://webflow.com/blog/api-documentation). 4. [**Create Samples and Tutorials**: Include code samples and tutorials to help users understand how to implement the API in various scenarios](https://webflow.com/blog/api-documentation). 5. [**Explain Authentication and Endpoints**: Detail the authentication methods required to access the API and describe each endpoint, including the methods, parameters, and expected responses](https://webflow.com/blog/api-documentation). 6. [**Write the Reference Material**: Provide detailed reference material for each endpoint, including descriptions of parameters, request and response formats, and error codes](https://webflow.com/blog/api-documentation). 7. [**Troubleshoot**: Include a troubleshooting section to help users resolve common issues they might encounter](https://webflow.com/blog/api-documentation). 8. [**Clean Up**: Review and edit the documentation to ensure clarity, accuracy, and consistency](https://webflow.com/blog/api-documentation).   **User Manual Documentations**   1. **Define Your Audience**: Understand who will be using the manual. [This helps tailor the content to their needs and knowledge level](https://www.tango.us/blog/user-manual). 2. [**Build an Outline**: Create a detailed outline of the manual, including sections such as introduction, table of contents, step-by-step instructions, troubleshooting, and FAQs](https://www.tango.us/blog/user-manual). 3. [**Gather Information**: Collect all necessary information about the product or service, including features, functions, and common issues](https://www.tango.us/blog/user-manual). 4. **Write Clear Instructions**: Use simple, concise language to write step-by-step instructions. [Avoid jargon and ensure each step is easy to follow](https://www.tango.us/blog/user-manual). 5. [**Include Visuals**: Add annotated screenshots, diagrams, and other visuals to help users understand the instructions better1](https://www.tango.us/blog/user-manual)[2](https://document360.com/blog/creating-a-user-manual/). 6. **Test the Manual**: Have a few users test the manual to ensure the instructions are clear and accurate. [Gather feedback and make necessary adjustments](https://www.tango.us/blog/user-manual). 7. [**Review and Edit**: Proofread the manual for any errors and ensure consistency in formatting and style](https://document360.com/blog/creating-a-user-manual/). 8. **Publish and Distribute**: Make the manual available to users, either in print or online. [Ensure it is easily accessible](https://www.tango.us/blog/user-manual). 9. [**Maintain and Update**: Regularly update the manual to reflect any changes or new features in the product or service](https://www.tango.us/blog/user-manual)   **System Documentation**   1. **Define the Purpose and Scope**: Clearly outline the objectives of the documentation and what it will cover. [This helps set expectations and ensures all necessary information is included](https://draft.dev/learn/technical-documentation-what-it-is-and-how-to-do-it-well). 2. **Identify the Audience**: Determine who will be using the documentation, such as developers, users, or maintenance teams. [Tailor the content to meet their needs and knowledge levels](https://draft.dev/learn/technical-documentation-what-it-is-and-how-to-do-it-well). 3. [**Gather Information**: Collect all relevant information about the system, including its architecture, components, functionalities, and interfaces](https://draft.dev/learn/technical-documentation-what-it-is-and-how-to-do-it-well). 4. [**Organize the Content**: Structure the documentation logically, typically starting with an overview and then detailing each component and its interactions](https://draft.dev/learn/technical-documentation-what-it-is-and-how-to-do-it-well). 5. **Write Clear and Concise Content**: Use simple language and avoid jargon. [Ensure each section is easy to understand and provides the necessary details](https://draft.dev/learn/technical-documentation-what-it-is-and-how-to-do-it-well). 6. [**Include Visuals**: Add diagrams, flowcharts, and screenshots to help illustrate complex concepts and make the documentation more engaging](https://draft.dev/learn/technical-documentation-what-it-is-and-how-to-do-it-well). 7. **Review and Validate**: Have subject matter experts review the documentation to ensure accuracy and completeness. [Make necessary revisions based on their feedback](https://draft.dev/learn/technical-documentation-what-it-is-and-how-to-do-it-well). 8. [**Publish and Distribute**: Make the documentation accessible to all relevant stakeholders, either in print or online](https://draft.dev/learn/technical-documentation-what-it-is-and-how-to-do-it-well). 9. [**Maintain and Update**: Regularly update the documentation to reflect any changes or new features in the system](https://asana.com/resources/process-documentation). | **(10 Days)**  **(15 Days)**  **(12 Days)** | **All Team Members**  Kate as Lead Chief Officer |
|  | 3.5 - Deployment | **Integration & Testing**   1. Backend-Frontend integration 2. Unit testing 3. Integration testing 4. User acceptance testing 5. Performance testing   **Security Implementation**   1. Security requirements analysis (2 days) 2. Implement authentication and authorization (3 days) 3. Data encryption implementation (2 days) 4. Security audit (1 day) | **15 Days**  (3 days)  (4 days)  (3 days)  (3 days)  (2 days)  **8 Days**  (2 days)  (3 days)  (2 days)  (1 day) | **All Team Members**  I Wibawa as Lead Chief Officer |
| **4 - Controlling** | 4.1 – Status Report | 1. Measuring the ongoing project activities 2. Monitoring the project variable 3. Identifying corrective actions to address issues and risks properly 4. Influencing the factors that could circumvent integrated change controls | **30 Days**  Monitor and Report from a month’s worth of data | **All Team Members**  Naveen as Lead Chief Officer |
|  | 4.2 - Update Plans | 1. **Review Current Status**: Assess the current progress of the project against the original plan. [Identify any deviations or issues](https://www.atlassian.com/blog/project-management/write-an-effective-project-plan). 2. **Identify Changes**: Determine what changes are needed. [This could include scope adjustments, timeline shifts, resource reallocations, or budget updates](https://www.atlassian.com/blog/project-management/write-an-effective-project-plan). 3. **Analyze Impact**: Evaluate the potential impact of the changes on the project. [Consider how they will affect the schedule, budget, resources, and overall project goals](https://www.atlassian.com/blog/project-management/write-an-effective-project-plan). 4. **Update Documentation**: Revise the project plan and any related documents to reflect the changes. [Ensure all updates are clearly documented](https://www.atlassian.com/blog/project-management/write-an-effective-project-plan). 5. **Communicate Changes**: Inform all stakeholders about the changes. [Provide clear explanations and ensure everyone understands the reasons and implications](https://asana.com/resources/project-management-plan). 6. **Implement Changes**: Put the changes into action. [Adjust schedules, reassign tasks, and update any tools or systems used to manage the project](https://asana.com/resources/project-management-plan). 7. **Monitor and Review**: Continuously monitor the project to ensure the changes are having the desired effect. [Be prepared to make further adjustments if necessary](https://asana.com/resources/project-management-plan). | **15 Days** |  |
| **5 - Closing** | 5.1 - Final Performance Review | 1. **Preparation**: Gather all relevant data on the employee’s performance over the review period. [This includes performance metrics, feedback from peers and supervisors, and any self-assessments](https://hbr.org/2022/07/how-to-conduct-a-great-performance-review). 2. [**Set the Agenda**: Outline the main topics to be covered during the review, such as achievements, areas for improvement, and future goals](https://hbr.org/2022/07/how-to-conduct-a-great-performance-review). 3. [**Create a Positive Environment**: Start the review with a personal greeting and create a comfortable atmosphere for open and honest communication](https://hbr.org/2022/07/how-to-conduct-a-great-performance-review). 4. **Discuss Performance**: Review the employee’s performance against the set objectives and expectations. [Highlight both strengths and areas for improvement](https://hbr.org/2022/07/how-to-conduct-a-great-performance-review). 5. **Provide Feedback**: Offer specific, actionable feedback. [Use examples to illustrate points and ensure the feedback is balanced and constructive](https://hbr.org/2022/07/how-to-conduct-a-great-performance-review). 6. **Set Goals**: Collaboratively set goals for the next review period. [These should be SMART (Specific, Measurable, Achievable, Relevant, Time-bound) goals](https://hbr.org/2022/07/how-to-conduct-a-great-performance-review). 7. [**Document the Review**: Record the key points discussed, including feedback, goals, and any agreed-upon action plans](https://hbr.org/2022/07/how-to-conduct-a-great-performance-review). 8. [**Follow-Up**: Schedule follow-up meetings to monitor progress and provide ongoing support and feedback](https://honehq.com/resources/blog/importance-of-performance-review-and-follow-up/). | **10 Days** | **All Team Members** |
|  | 5.2 - Prepare Final Report | 1. **Define the Purpose and Audience**: Understand the purpose of the report and who will be reading it. [This helps tailor the content to meet the needs and expectations of the audience](https://clickup.com/blog/how-to-write-a-report/). 2. **Gather Information**: Collect all relevant data, findings, and insights from the project. [This includes any supporting documents, charts, and graphs](https://clickup.com/blog/how-to-write-a-report/). 3. **Create an Outline**: Develop a detailed outline to organize the report logically. [Common sections include an executive summary, introduction, methodology, results, discussion, conclusion, and recommendations](https://clickup.com/blog/how-to-write-a-report/). 4. **Write the Draft**: Start writing the report based on the outline. [Ensure each section is clear and concise, and use visuals to support the text where necessary](https://clickup.com/blog/how-to-write-a-report/). 5. **Review and Revise**: Review the draft for clarity, coherence, and completeness. [Make necessary revisions to improve the flow and accuracy of the information](https://clickup.com/blog/how-to-write-a-report/). | **5 Days** | **All Team Members** |