CONNECT-EZ

SOFTWARE PROJECT MANAGEMENT PLAN

1.1.0

November 21st, 2023

Team Cosmos

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	Kabir Bhakta, Bhautik Sojitra, Heejong Kim, Yirong Wang, Vrushil Patel	November 19, 2023	Group	November 19 2023	Initial Version
1.1	Kabir Bhakta	November 21, 2023	Group	November 21, 2023	Additional Plans

Name	Email	Period of being project manager
Kabir Bhakta	bhaktakb@myumanitoba.ca	November 11, 2023 – Nov 24, 2023
Yirong Wang	wangy27@myumanitoba.ca	October 30, 2023 – November 10, 2023
Bhautik Sojitra	sojitrbm@myumanitoba.ca	September 22, 2023 – September 29, 2023
Heejeong Kim	kimh4@myumanitoba.ca	October 14, 2023 – October 27, 2023
Vrushil Patel	patelvk1@myumanitoba.ca	September 30, 2023 – October 13, 2023

TABLE OF CONTENTS

1. INTRODUCTION

1.1. PROJECT OVERVIEW

This document is intended for the members of the project describing the managerial aspects and technical aspects. The document is intended for planning and scheduling purposes and serves as a summary document of the deliverables expected from each of the teams. In this section, we'll explore the purpose of this document and dive into the project's scope and objectives.

1.1.1. PURPOSE

The SPMP for the "ConnectEZ" project is a roadmap and guide for the team and stakeholders. It ensures a clear understanding of tasks, methods, and steps for success. This crucial tool facilitates effective communication, coordination, and decision-making throughout the project. By defining roles, responsibilities, and processes, the SPMP streamlines progress and contributes to successful completion.

1.1.2. SCOPE

The "ConnectEZ" project involves the design, development, testing, and deployment of an advanced virtual phone system. We're focusing on creating a service that allows users to make and receive calls seamlessly using a single business phone number across different devices. This includes desktop applications and mobile phones.

1.1.3. OBJECTIVES

The "ConnectEZ" project has diverse objectives. Our primary goal is to create a user-friendly virtual phone system that seamlessly adapts to desktop and mobile platforms. Simultaneously, we strategically aim to secure a market share surpassing 10% in the virtual phone service sector. A key performance indicator is maintaining a robust 95% system uptime rate for reliable service. Finally, we aspire to empower users to easily make and receive calls using a unified business phone number, enhancing communication efficiency.

1.2. PROJECT DELIVERABLES

The ConnectEZ team is dedicated to delivering a fully functional system that follows the guidelines set in the ConnectEZ software requirements specification (SRS) document. We aim to hand over all the necessary software and related documents to the client by March 7th, 2025. To ensure clarity, a final report and presentation will be shared with Pengfei He during this final delivery. Additionally, we plan to provide the client with a working Minimum Viable Product (MVP) by March 13th, 2024, offering an early version of our solution. The upcoming section outlines the specific elements and documents that make up these important project deliverables.

Software Deliverables
The team will deliver a working MVP by March 13th, 2024. Furthermore, the ConnectEZ team will also deliver a working virtual phone system at the end of the project that satisfies the requirements.

Team Documents

The following documents are for the team's use:

- Design Report
- Testing Report
- Security Report
- Final Project Report
- Financial Budget Report

Client Documents

The following documents will be delivered to the client:

- Training Materials
- Billing System
- Final Project Report
- Presentation

1.3. EVOLUTION OF SPMP

The SPMP for the ConnectEZ project will be under version control, so any changes will be made to the plan itself. The updated document will be made available to all project members and interested stakeholders.

1.4. REFERENCES

1. IEEE Std 1058-1998, IEEE Standard for Software Project Management Plans, IEEE 1998

2. "Software project management plan (SPMP)."
   Available: https://cs.uwaterloo.ca/~apidduck/se362/Assignments/A2/spmp.pdf

3. "SPMP Examples," *ase.in.tum.de*. Available:
   https://ase.in.tum.de/stars.globalse.org/stars1/docs/SPMP/Examples/Examples.html (accessed Nov. 20, 2023).

4. Bernd Bruegge, Allan Dutoit: *Model-Based Software Engineering: A Project-Oriented Approach*, Course Manuscript.

5. Robert Orfali, Dan Harkey, *Client/Server Programming with Java and CORBA,* Wiley & Sons, Inc, 1997.

2. PROJECT ORGANISATION

2.1. PROCESS MODEL

For the ConnectEZ project, we will follow the Agile software development approach. Agile is a highly adaptive process that enables teams to respond to the unpredictability of building software through incremental, iterative workflows.

2.2. ORGANISATIONAL STRUCTURE

The project team for ConnectEZ is structured to efficiently manage various aspects of the project. The team is divided into several sub-teams, each focusing on different components of the system.
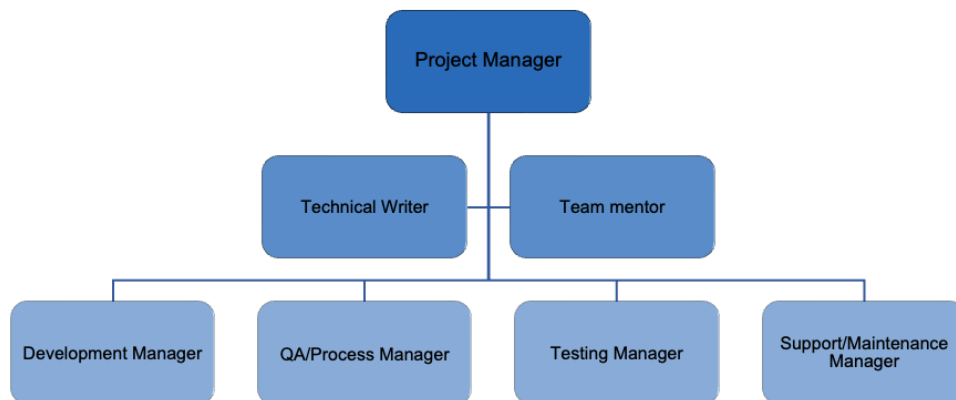
The ConnectEZ project's team structure is intentionally designed to make a clear hierarchy and efficient distribution of responsibilities with the project manager who provides direction and oversees the project's progress.

Supporting the project manager are specialized roles including a technical writer and team mentor. The technical writer's responsibility is to assist the team in producing high-quality documentation and ensuring consistency across all project materials. The team mentor offers guidance and support, fostering collaboration and aiding the team in achieving its goals. This setup helps everyone know what they need to do and helps the team work well as a group.

Team members are encouraged to actively participate in the decision-making process, with a democratic approach where each member's vote holds equal weight. This approach values different viewpoints, leading to innovative ideas and greater dedication from everyone on the team. The team will hold weekly meetings and use email for urgent matters, and we encourage team members to interact personally, strengthening relationships and sharing knowledge informally.

2.3. ORGANISATIONAL BOUNDARIES

Role	Relationships with organizations
Project Manager [Kabir Bhakta]	• Interface with Parent Organization: Reports on progress, and ensures the project aligns with the strategic adjectives. • Interface with Customer Organization: Get feedback from customers and ensure the project meets the requirements.
Development Manager [Bhautik Sojitra]	• Interface with Subcontracting Organizations: Work with any external development teams to put their work with the project's development strategy. • Interface with Documentation Organization: Work with technical writers to ensure accurate user documentation.
QA/Process Manager [Vrushil Patel]	• Interface with QA Organization: Work with external QA teams to integrate their quality assurance processes with the internal quality plan.
Testing Manager [Heejeong Kim]	• Interface with Customer Organization: Bring feedback from customers into testing strategies to ensure the product meets user needs.
Support/Maintenance Manager [Yirong Wang]	• Interface with End-user Support Organization: Create protocols and processes for end-user support. • Interface with Customer and IT Organizations: Make sure the service provided to users stays consistent and matches what users need and that the IT systems can handle.

2.4. PROJECT RESPONSIBILITIES

Role	List of responsibilities
Project Manager [Kabir Bhakta]	• Motivate the team members to perform their tasks. • Make strategic decisions and provide guidance to team members. • Monitor project progress and adjust plans as needed. • Resolve conflicts and manage team.
Development Manager [Bhautik Sojitra]	• Lead the team in producing the development strategy. • Lead the development team in designing and building software components. • Ensure code quality and commit to project specifications. • Lead the team in developing the test materials and running the tests. • Lead the team in producing the product's user documentation. • Oversee the integration of new features and systems. • Lead team in producing high-level design.

QA/Process Manager [Vrushil Patel]	• Lead the team in producing and tracking the quality plan. • Develop and enforce quality assurance policies and procedures. • Monitor bug reports and manage the issue resolution process. • Act as recorder in the team's meetings. • Alerts the team to quality problems.
Testing Manager [Heejeong Kim]	• Lead the testing team in executing test plans to ensure software functionality, performance, and security meet requirements. • Report on testing activities, including test results, coverage, and areas of risk, to the stakeholders. • Ensure that all release deliverables meet the quality standards set forth by the project requirements. • Collaborate with the development and operations team to continuous integration. • Implement a comprehensive testing strategy for the project, including test planning, test case, design, and test automation.
Support/Maintenance Manager [Yirong Wang]	• Lead the team in determining its support needs and in obtaining the needed tools and facilities. • Maintain the team's issue and risk tracking system. • Coordinate support services and manage customer support teams. • Oversee the creation of support documentation and resources. • Maintain the system glossary. • Manage the configuration management system.

3. MANAGERIAL PROCESS

3.1. MANAGEMENT OBJECTIVES AND PRIORITIES

3.1.1. OBJECTIVE

The primary management objective of the ConnectEZ project is to deliver a user-friendly, robust virtual phone system that aligns with the strategic goals of enhancing communication solutions and capturing a significant market share within the internet-based telephony industry.

3.1.2. PRIORITIES

- *Quality Assurance***:** Ensure the highest quality in both the development process and the end product, aiming for a 95% system uptime rate and seamless user experience.
- *Time-to-Market***:** Adhere to an aggressive timeline to release an MVP within 6 months and a final product within 18 months, to quickly capture market share.
- *Cost Efficiency***:** Manage the project within the allocated budget of $500K, maximizing resource utilization and ensuring financial viability.
- *Scalability***:** Design the system with scalability in mind to accommodate future growth in user base and functionality.
- *Security***:** Prioritize the security of the system to protect user data and maintain trust in the ConnectEZ brand.

- *Customer Satisfaction***:** Provide exemplary customer service and support to ensure user satisfaction and positive market reception.

3.2. ASSUMPTIONS, DEPENDENCIES, AND CONSTRAINTS

3.2.1 ASSUMPTIONS

- *Team Expertise*: The project assumes the availability of a skilled and experienced team capable of handling the diverse aspects of software development, including system design, development, testing, and deployment.
- *Technological Resources*: Adequate technological resources, including software development tools and testing environments, are presumed to be available and reliable throughout the project lifecycle.
- *Market Viability*: The project is based on the assumption that there is a significant market demand for an advanced virtual phone system, warranting investment and effort.
- *Stakeholder Commitment*: Continuous engagement and timely decision-making from stakeholders, including prompt feedback and necessary approvals, are assumed.

3.2.2 DEPENDENCIES

- *Stakeholder Feedback*: Timely input and feedback from stakeholders, including the Project Sponsor/Product Owner, are critical for the project's milestones and directional decisions.
- *Inter-Team Collaboration*: Effective collaboration and communication between the development, design, testing, and support teams are essential for the project's success.

3.2.3 CONSTRAINTS

- *Budgetary Limitation*: A fixed budget of $500K, requiring careful allocation to different project divisions and activities without overspending.
- *Timeframe Restriction*: The project is constrained by an 18-month timeframe, necessitating efficient management to ensure the timely delivery of the MVP and the final product.
- *Resource Allocation*: Limited availability of human resources and technological tools, necessitating efficient and strategic deployment of available resources.
- *Quality Standards*: Maintaining high-quality standards within the constraints of budget and schedule.
- *Software Integration*: Constraints related to incorporating existing software or components into the new system.
- *Technological Selection*: The project is limited to certain technologies and frameworks, which may restrict some design or feature choices.
- *External Interfaces*: Dependence on external interfaces and APIs, which could limit control over certain functionalities and integration aspects.

3.3. RISK MANAGEMENT

Below is the potential risk on top of the constraints mentioned above. It also mentions the mitigation plan to avoid these risks.

3.3.1 POTENTIAL RISKS

- *Unclear Requirements*: Ambiguous or changing requirements can lead to scope creep and project delays. Mitigation: Conduct thorough requirement analysis sessions with stakeholders. Use agile practices to adapt to changing requirements during the development process.

- *Technical Complexity*: The project may involve complex technical challenges that the team is not prepared for. Mitigation: Conduct a detailed technical feasibility analysis before the project begins. Ensure that the team has the necessary expertise and consider hiring external consultants if needed.

- *Integration Issues*: Difficulty integrating the call processing system with other existing systems. Mitigation: Conduct thorough integration testing. Collaborate with application owners and stakeholders to ensure compatibility.

- *Security Concerns:* Potential security vulnerabilities in the call processing system. Mitigation: Implement robust security measures, conduct regular security audits, and stay informed about the latest security best practices.

- *User Adoption*: The user may resist or struggle to adopt the new call processing system. Mitigation: Involve users in the development process through regular feedback sessions. Provide training and create user-friendly documentation.

- *Third party Reliability*: Dependence on third party applications for critical components. Mitigation: Choose reputable applications, conduct due diligence, and have contingency plans in case of issues.

There may be other situational-based risks hindering the project flow. The management should follow a detailed risk management plan to avoid such risks.

3.3.2 DETAILED RISK MANAGEMENT PLAN

- **Risk Identification**: regularly review project documentation, conduct brainstorming sessions, and involve team members to identify potential risks.

- **Risk Assessment:** Evaluate each identified risk based on its impact and probability. Prioritize risks that have higher likelihood of occurrence and significant impact on the project.

- **Risk Mitigation:** Develop specific strategies to address such identified risks. This may include implementing preventive measures, allocating additional resources, or adjusting the project plan.

- **Contingency Planning:** Create contingency plans for high-priority risks. These plans should outline the actions to be taken if the risk happens, minimizing the impact on the project.

- **Regular Monitoring:** Continuously monitor and reassess risks throughout the project lifecycle. Regularly update the risk documentation to reflect changes in risk status.

- **Communication:** Maintain transparent communication about potential risks with team members, stakeholders, and project sponsors. Encourage a culture of risk awareness and reporting.

- **Documentation:** Document the risk management plan, including identified risks, assessments, mitigation strategies, and contingency plans. Keep this documentation accessible to the project team.

3.4. MONITORING AND CONTROLLING MECHANISM

This section specifies the project management processes for the project. It defines the plans for project start-up, risk management, project control and project close-out.

3.4.1. REQUIREMENT CONTROL PLAN

- The Requirement Control Plan outlines the procedures and responsibilities for managing requirements throughout the project. It aims to ensure that requirements are accurately captured, analyzed, and maintained to meet project objectives.

- *Responsibility***:** The project manager is responsible for analyzing, gathering, and documenting project requirements.

- *Methods***:** The requirements will be gathered through stakeholder interviews and surveys. It will be documented in SRS documents with detailed user stories and use cases. Once the SRS document is approved, we should try not to change the requirements.

- *Tools***:** The requirements documents should be placed in Confluence with everyone having access to it. We should maintain the latest version in there.

- *Traceability***:** We should maintain traceability between requirements, design documents and project deliverables. Therefore, we should be able to track back to the user stories in SRS documents, after developing the relevant feature. It is also important to establish version control procedures for requirement documents including previous changes and updates to ensure that all team members are working with the latest version.

- *Analysis and Validation***:** Using prioritization, feasibility analysis and impact assessment, we should do comprehensive requirement analysis periodically.   We should conduct

regular validation sessions with stakeholders to ensure requirements align with their expectations to avoid scope creep.

- *Change Management***:** Normally, we should avoid requirement changes, unless necessary. However, in case of changes, they should be submitted through a formal change request process by the team member responsible. The change request should include a clear description, rationale, and impact on the project. The request should be reviewed by all the stakeholders. They should perform an impact analysis for proposed changes to assess the effects on the project schedule, scope, and resources.

3.4.2. SCHEDULE CONTROL PLAN

The Schedule Control plan outlines the processes, methods, and tools for controlling the schedule of the project. It ensures effective measurement of progress, comparison of actual progress to planned progress, and the implementation of corrective actions when needed.

Schedule Control Processes

- *Major and Minor Project Milestones:* Regularly assess the completion status of major milestones such as system design completion, development phase completion, testing completion, and deployment. Minor milestones, including sprint completion will be measured during bi-weekly reviews.
- *Comparison of Actual vs Planned Progress:* Regularly compare actual progress against the baseline schedule to identify any deviations.
- *Corrective Action Implementation***:** Establish a process for implementing corrective actions when actual progress deviates from the planned progress. Identify responsible parties for taking corrective actions and define escalation paths for unresolved issues.

Methods and Tools

- *Gantt Chart***:** For the project, we are maintaining a Gantt chart with a predefined timeline for each phase, major milestones and task dependencies.
- *Critical Path Analysis*: The project also has a critical path graphed out, which needs to be monitored regularly to ensure that we are on the right track.
- *JIRA:* The project employs JIRA for task and schedule management.

Measurement of Scope and Quality Criteria

- *Scope Measurement***:** Evaluate the completion of major milestones based on the delivered features and functionalities. Utilize acceptance criteria defined during requirements analysis to measure the completion of each sprint or development phase.
- *Quality of Work Completed***:** Assess the quality of work completed at each milestone by using predetermined quality metrics and standards. Implement testing and code review processes to ensure adherence to quality standards.
- *Objective Criteria for Each Milestone***:** Define objective criteria specific to each milestone to assess the achievement of planned progress. Example: Milestone 1 – all high-level design documents approved by stakeholders.

3.4.3. Communication plan

Tools

- *Video Conferencing Tool:* For daily stand-ups and status meetings, zoom software should be used. Everyone should be provided with the software. However, the team should emphasize in-person meetings when possible.
- *Regular Test-Based Tool***:** Each team member should be given access to a Slack channel for regular basis communication.
- *Documentation Tools***:** Confluence will be used for documentation purposes. The project manager should document every major decision. Developers are responsible for maintaining user guides and feature information. The customer support trainer is responsible for maintaining a trainee guide.
- *Other Available Communication Resources***:** Each team member will have access to an Outlook account to communicate with stakeholders outside the organizations, who do not have access to Slack channel.

Communication Methods

- *Team Meetings*
  - Purpose: Discuss project progress, updates, and address team concerns.
  - Frequency: Weekly (every Monday).
  - Tools: Video conferencing (Zoom), project management software (Jira).
- *Status Reports*
  - Purpose: Provide a summary of project status, accomplishments, and upcoming milestones.
  - Frequency: Bi-weekly (every other Friday).
  - Tools: Document sharing (Google Docs), email distribution.
- *Email Updates*
  - Purpose: Communicate important announcements, changes, and reminders.
  - Frequency: As needed.
  - Tools: Outlook.
- *Stakeholder Meetings*
  - Purpose: Engage with stakeholders to gather feedback and provide project updates.
  - Frequency: Monthly.
  - Tools: Video conferencing (Zoom)

3.4.4. PROJECT CLOSEOUT PLAN

The project closeup plan describes the steps to take upon project is finalized. The project manager is responsible for looking over all the tasks described below.

- *Documentation Review:* Conduct a thorough review of all the project documentation, including user manuals, technical documents and reports. Verify that all deliverables meet the acceptance criteria and are aligned with the project goal.
- *Documents Archiving:* Categorize all the project documents, including but not limited to requirements, design documents, project plans, and reports. Archive them in a secure and accessible repository. Ensure proper versioning and naming conventions for ease of retrieval.
- *Staff Reassignment Plan:* Identify new project assignments and roles for team members. Communicate the reassignment plans with each team member individually. Facilitate knowledge transfer sessions if team members are transitioning to new projects.
- *Review Project Matrics:* Analyze project metrics related to schedule adherence, budget variance, and quality. Evaluate the success of meeting project objectives and milestones.
- *Debriefing Project Personnel:* Conduct individual or team debrief sessions to discuss project experiences. Capture lessons learned, challenges faced, and suggestions to improve.
- *Document lesson learned:* Compile lessons learned documentation from debriefing sessions and individual reflections. Organize them into categories, such as project management, technical aspects and team collaboration.
- *Financial Closure:* Verify that all vendor invoices are received and processed. Ensure that reimbursement for project-related expenses is completed. Close out financial accounts and records.
- *Client Acceptance Sign off:* Share the final project deliverables with the client for review. Obtain client sign-off indicating acceptance for the delivered software.

3.5. STAFFING PLAN

Based on the cost and schedule estimates defined in section 5, the project requires the staff as follows:

Skill/Role	Total Number of Personnels	Duration of Requirement (in weeks)	Source of staff	Description of work
System Designer	2	6	Contracted	Product designing, documentation
Developer	4	40	2 internal, 1 new hire, 1 contracted	Back-end, front-end, database development, Networking experience, documentation
QA Engineer	2	22	internal	Unit testing, acceptance testing, integration testing

Customer Support Trainer	1	6	contracted	Customer service, Teaching, Documenting
Customer Service	3	6 + continued employment after the product has deployed	New hire / internal	Helping customers, basic technical work
DevOps Engineer	1	6	Internal	Releasing MVP and final product, provide other team members with networking support
Project Manager	1	18	Contract	Managing the people and the product

3.5.1. STAFFING BY PHASE

The table below shows the number of people required by skill in each project phase. The project manager will be required throughout the project to ensure that all the team members are provided with enough resources to get the work done.

Project Phase	Role: Number of personnel
Project Initialization	All the stakeholders
System Design	System Designer: 2
Application Development	Developer: 4 DevOps Engineer: 1 (for setup purposes)
Billing	System Designer: 2 Developer: 2 QA Engineer: 1
Training and Support	Customer Support Trainer: 1 Customer Service Team: 3
Testing	QA Engineers: 2
Project Finalization	DevOps Engineer: 1 (Releasing the final product) Developer: 1 (for documentation and in case of emergency)

4. TECHNICAL PROCESs

4.1. METHODS, TOOLS, AND TECHNIQUES

- The team decided to use Agile Methodology for the development of the Online Virtual Phone System, ConnectEZ. The team chose this approach because it offers flexibility to change requirements according to the stakeholders which will improve the relationship with the stakeholders. The team is looking forward to using an Object-Oriented Programming Language. This section will be updated once a language has been agreed upon.
- To keep track of the changes, GIT will be used as the version control system. The developers will create their own branches with appropriate names. Changes to the code will be reviewed to ensure code quality before making the changes permanent.
- For the quality plan, it will contain a list of tests that will be written and carried out. These include unit tests, integration tests, and end-to-end tests using various tools. Testing will be conducted continuously to maintain high code quality.
- A strategy has been developed for obtaining the necessary resources. This strategy involves in deciding whether to create or purchase components of the system, selecting the contract type and identifying back-up options as a form of risk management.

4.2. SOFTWARE DOCUMENTATION

The software documentation is important for future reference. There will be detailed documentation on important parts of the project. The documentation will have the following documents:

- **Requirements Document**: List of all functional and non-functional requirements.
- **Technical Documentation**: Information about all the tools and techniques used, description of the code structure, explanation of various libraries and functions and details on the testing of the system.
- **User Manual:** Important document for the end-users providing step-by-step guidance on how to use the system.
- **Design Documents:** This includes prototypes of user interfaces, database schema and the system architecture.
- **Code Naming Conventions:** Explains how to write variables, functions, and classes names to ensure consistency and clarity.
- **File Naming Conventions:** Descriptive names will be used to name files that will help in identifying the purpose of the file.

4.3. PROJECT SUPPORT FUNCTIONS

These functions ensure the smooth flow of the project. It covers various aspects such as communication, documentation, quality assurance, risk management and administrative support.

- **Quality assurance functions** are responsible for conducting regular reviews and testing the software to ensure coding standards are maintained.

- **Administrative support functions** are responsible for managing the flow of information to ensure effective communication, scheduling and coordinating team meetings and taking notes for future reference.
- **Contract negotiation support functions** include finding the right company, negotiating the contracts, and reviewing the contracts with legal experts.
- **Technical Support Services** include services provided to end-users on how to use the system.

5. WORK PACKAGES, SCHEDULE, AND BUDGET

5.1. WORK PACKAGES
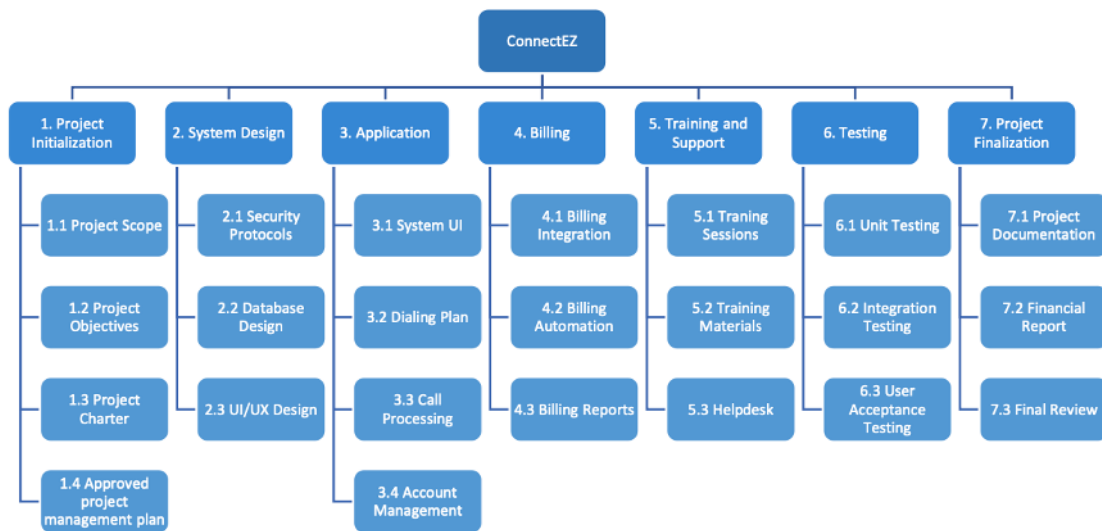
- **Project Initialization**
  - Necessary Resources: Project Manager, Stakeholders.
  - Estimated Duration: 1 month.
  - Products/Deliverables: Project Scope Statement, Project Objectives, Project Charter, Approved Project Management Plan.
  - Acceptance Criteria: Document approval by stakeholders.
  - Predecessors: None.
  - Successors: System Design.

- **System Design**
  - Necessary Resources: Security Analyst, Database Analyst, UI/UX Designer.
  - Estimated Duration: 2 months.
  - Products/Deliverables: Security Protocol Specification, Database Schema, UI/UX Design Mockups.
  - Acceptance Criteria: Designs meet technical and user experience standards as per requirements.
  - Predecessors: Project Initialization.
  - Successors: Application Development.

- **Application Development**
  o <u>Necessary Resources</u>: Developers, Telecommunications Specialist, System Analyst.
  o <u>Estimated Duration</u>: 6 months.
  o <u>Products/Deliverables</u>: System UI, Dialing Plan, Call Processing Module, Account Management System.
  o <u>Acceptance Criteria</u>: Functional modules as per design specifications, passing initial tests.
  o <u>Predecessors</u>: System Design.
  o <u>Successors</u>: Billing, Training and Support, Testing.

- **Billing**
  o <u>Necessary Resources</u>: System Analyst, Financial Analyst.
  o <u>Estimated Duration</u>: 3 months.
  o <u>Products/Deliverables</u>: Billing Integration Module, Billing Automation System, Billing Reports.
  o <u>Acceptance Criteria</u>: Accurate billing as per usage, report generation without errors.
  o <u>Predecessors</u>: Application Development.
  o <u>Successors</u>: Training and Support, Testing.

- **Training and Support**
  o <u>Necessary Resources</u>: Training Specialist, Technical Writer, Helpdesk Coordinator.
  o <u>Estimated Duration</u>: 2 months.
  o <u>Products/Deliverables</u>: Training Sessions, Training Materials, Operational Helpdesk.
  o <u>Acceptance Criteria</u>: Effective training leading to competent use of system, established support mechanism.
  o <u>Predecessors</u>: Billing.
  o <u>Successors</u>: Testing, Project Finalization.

- **Testing**
  o <u>Necessary Resources</u>: Test Manager, Quality Assurance Team.
  o <u>Estimated Duration</u>: 3 months. Products/Deliverables: Unit Tests, Integration Tests, User Acceptance Tests.
  o <u>Acceptance Criteria</u>: All tests passed, system meets business and user requirements.
  o <u>Predecessors</u>: Application Development, Billing.
  o <u>Successors</u>: Project Finalization.

- **Project Finalization**
  o <u>Necessary Resources</u>: Project Manager, Technical Writer, Financial Analyst.
  o <u>Estimated Duration</u>: 1 month. Products/Deliverables: Project Documentation, Financial Report, Final Review.
  o <u>Acceptance Criteria</u>: Documentation accurately reflects project execution, financial report matches budget and expenses, final review confirms project objectives met.
  o <u>Predecessors</u>: Testing.
  o <u>Successors</u>: None (Project Closure).

Project Initialization	This is about laying groundwork for the project. It includes defining the project scope, setting objectives, creating a project charter, and getting approval for the project management plan.
System Design	This is focused on designing the system. It consists of developing security protocols to ensure the system's integrity, designing a database that will store all the necessary data, and crafting the user interface and user experience.
Application	This covers the creation of the actual virtual phone application. It includes developing the system's user interface, establishing a dialing plan for the system, processing the calls made through the system, and managing user accounts within the system.
Billing System	This is for the financial aspect of the system. It involves integrating a billing system to handle charges for calls, automating the billing process, and generating billing reports for users and administrators.
Training and Support	This ensures that the users and administrators of the system are well-prepared to use it. It includes conducting training sessions, creating training materials, and setting up a helpdesk for ongoing support.
Testing	This is the testing stage before the system goes live. This involves unit testing of individual components, integration testing to ensure these components work together, and user acceptance testing to confirm the system meets the user's needs.
Project Finalization	This is wrapping up the project with documentation, financial reporting, and a final review process, including obtaining release approvals to officially conclude the project.

5.2. DEPENDENCIES

- **Project Initialization**
  - o The Project Scope must be defined first.
  - o Project Objectives depend on the completion of the Project Scope.
  - o Project Charter comes after Project Objectives
  - o The Approved Project Management Plan is the last step in this phase and is dependent on the Project Charter.
- **System Design**
  - o Design Security Protocols can start after the Approved Project Management Plan.
  - o Database Design also follows the Approved Project Management Plan and can be done in parallel with Design Security Protocols.
  - o UI/UX Design is dependent on the Database Design since the user interface needs to align with the database structure.
- **Application**
  - o System UI development is dependent on UI/UX Design.
  - o Dialing Plan can begin after Database Design.
  - o Calling Processing requires both System UI and Dialing Plan to be in place.
  - o Account Management can start after Calling Processing.
- **Billing**
  - o Billing Integration is dependent on Database Design.
  - o Billing Automation requires Billing Integration to be completed.
  - o Billing Reports depend on the Billing Automation system being in place.
- **Training and Support**

- Training Sessions need to wait until Billing Reports and Account Management are ready so that training can cover these aspects.
- Training Materials can start alongside Training Sessions.
- Helpdesk Setup can begin after System UI is developed and continue as Training Materials are being created.

- **Testing**
  - Unit Testing should start after the main features.
  - Integration Testing follows Unit Testing and must ensure all systems work together.
  - The User Acceptance Testing is the last testing phase and requires all previous testing to be completed.

- **Project Finalization**
  - Project Documentation is finalized after User Acceptance Testing.
  - Financial Reports can be prepared after Billing Reports are generated.
  - Final Review and Release Approvals are the last steps and ensure the project meets all requirements and is ready.

5.3. RESOURCE REQUIREMENTS

- **Human Resources**
  - Stakeholders
  - Project Manager
  - Security Analyst
  - Database Analyst
  - UI Designer and UX Designer
  - Graphic Designer
  - Telecommunications Specialist
  - System Analyst
  - Financial Analyst
  - Training Specialist
  - Technical Writer
  - Helpdesk Coordinator
  - Test Manager
  - End Users

- **Support Hardware**
  - Servers: To host database applications and support the backend.
  - Networking Equipment
  - Testing Devices: desktops, laptops, and mobile phones for testing the application.
  - Computers

- **Support Software**
  - Developing Environment (IDEs)
  - Design Software
  - Communication Tools
  - Security Tools
  - Billing Software
  - Testing Software
  - Documentation Tools

- **Facilities**
  - o Office Space
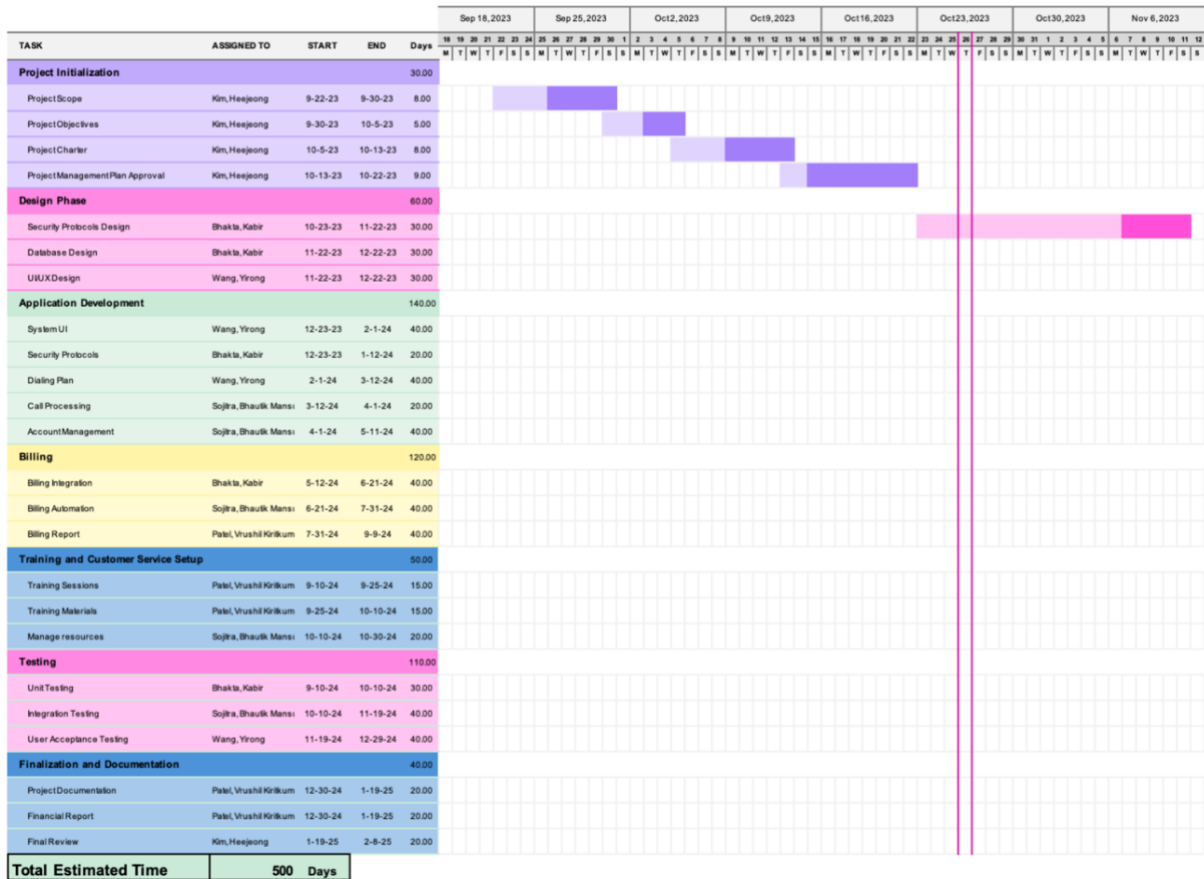 - o Meeting Rooms
  - o Training Facilities

5.4. BUDGET, RESOURCE ALLOCATION AND SCHEDULE

5.4.1. BUDGET ALLOCATION

Based on the total budget of $500,000 and the total estimated time of 500 days for the project.

Division	Estimated Cost	Basis of estimation	Resources required	Confidence Levels
System Design	$50,000	Detailed design documents.	System architects and product designers	High
Development	$200,000	Experience with similar projects.	Developers and development tools	Medium

Testing	$50,000	Complexity of the systems.	QA engineers, Testing tools	High
Customer Support Team	$60,000	Wages of the employees and historical data.	Administrative systems, and people	Medium
Documentation	$20,000	Cost of the tools required	Documentation software	High
Deployment	$20,000	System deployment plan and cost of the tools required.	Deployment software, Hosting service	Low
Infrastructure	$40,000	Based on the current costs	Servers, Network interfaces	Medium
Hardware Equipment	$40,000	Cost of the systems	Laptops/PCs, monitors	Medium
Contingency Reserve	$20,000	Emergency funds just to be on a safer side	--	Medium

5.4.2. RESOURCE ALLOCATION

There are two main types of required resources for this project to succeed, in addition to personnel. 1. Hardware Resources 2. Software Resources. The following is planning to acquire these resources.

- **Hardware Resources**
  - o Workstations: High performance desktops/laptops with MacOS
  - o Servers/Back-up Servers: For hosting call processing software
  - o Storage: Adequate storage for storing databases
  - o Network Equipment: Routers, Switches, and firewalls for security purposes.

All the hardware resources will be acquired and set up during the project initialization phase. The project manager and DevOps engineer are the go-to personnel in case of any concerns/issues.

- **Software Resources**
  - o IDE (Visual Studio)
    - ▫ Personnel: Developers, QA Engineers
    - ▫ Acquisition plan: Should be acquired before the application development phase starts.
  - o Version Control (Git/Bitbucket)
    - ▫ Personnel: Each team member
    - ▫ Acquisition plan: Should be acquired before the application development phase starts.
  - o Communication Tool (Slack)
    - ▫ Personnel: Each team member

- ▫ Acquisition plan: This should be acquired as soon as possible to ensure that each stakeholder on the project has a way of communicating the issues/ideas.
  - o Documentation Tool (Confluence)
    - ▫ Personnel: Each team member
    - ▫ Acquisition plan: This should be acquired as soon as possible to ensure that everyone has access to a common space where they can start documenting their ideas.
  - o Task Management/ Project Management (JIRA)
    - ▫ Personnel: Each team member
    - ▫ Acquisition plan: This should be acquired as soon as the project is approved to start planning the project.
  - o Deployment Automation Tool (Jenkins)
    - ▫ Personnel: DevOps Engineer, Developers, QA Engineers
    - ▫ Acquisition plan: This should be acquired before the application development phase starts so that devOps engineer can start setting up during that phase.
  - o Customer Service Software
    - ▫ Personnel: Customer Support Trainer, Customer Service Team
    - ▫ Acquisition plan: This should be acquired after the MVP product has been approved.

The project manager is responsible for maintaining the licenses for these software tools. The PM should also be ready to provide any additional tools required to complete the tasks.

- **Other Resources**
  - o There might be other resources needed to complete tasks. In that case, the team member should contact the project manager.

5.4.3. SCHEDULE

Project Initialization (Sep 22, 2023 – Oct 22, 2023)
- Project Scope: Sep 22 – Sep 30 (8 days)   - Project Objectives: Sep 30 – Oct 5 (5 days)   - Project Charter: Oct 5 – Oct 13 (8 days)   - Project Management Plan Approval: Oct 13 – Oct 22 (9 days)
Design Phase (Oct 23, 2023 – Dec 22, 2023)
- Security Protocols Design: Oct 23 – Dec 22 (60 days)   - Database Design & UI/UX Design: Nov 22 – Dec 22 (30 days)
Application Development (Dec 23, 2023 – May 11, 2024)
- System UI: Dec 23 – Jan 24 (40 days)   - Dialing Plan: Jan 2 – Mar 12 (40 days)   - Call Processing: Mar 1 – Apr 1 (20 days)   - Account Management: Apr 1 – May 11 (40 days)
Billing (May 4. 2024 – Sep 9, 2024)
- Billing Integration: May 5 – Jun 21 (40 days)   - Billing Automation: Jan 21 – Jul 31 (40 days)

[Insert appropriate disclaimer(s)]

• Billing Report: Jul 31 – Sep 9 (40 days)
Training (Sep 10, 2024 – Oct 30, 2024)
• Training Sessions: Sep 10 – Sep 25 (15 days) • Training Materials: Sep 25 – Oct 10 (15 days) • Manage Resources: Oct 10 – Oct 30 (20 days)
Testing (Sep 10, 2024 – Dec 29, 2024)
• Unit Testing: Sep 10 – Oct 10 (30 days) • Integrating Testing: Oct 10 – Nov 19 (30 days) • User Acceptance Testing: Nov 19 – Dec 29 (40 days)
Finalization and Documentation (Dec 30, 2024 – Feb 25, 2025)
• Project Documentation: Dec 30 – Jan 19 (20 days) • Financial Report: Jan 20 – Feb 19 (20 days) • Final Review: Feb 8 – Feb 25 (20 days)

6. ADDITIONAL SECTIONS

This section outlines the supplementary plans essential to meet product requirements and contractual terms associated with the "ConnectEZ" project. These plans are designed to ensure the comprehensive fulfillment of safety, privacy, and security requirements, as well as other critical aspects integral to the success of the project.

6.1. SAFETY, PRIVACY, AND SECURITY ASSURANCE PLAN:

The Safety, Privacy, and Security Assurance Plan details the strategies and measures in place to guarantee that the "ConnectEZ" system adheres to the highest standards of safety, privacy, and security. It outlines protocols, processes, and technologies employed to safeguard user data and ensure a secure operational environment.

6.2. PRODUCT SUPPORT PLAN:

The Product Support Plan details the strategies for providing ongoing support to end-users. It includes helpdesk services, troubleshooting procedures, and the escalation process for addressing user queries or issues effectively.

6.3. ENVIRONMENTAL IMPACT PLAN:

If applicable, the Environmental Impact Plan outlines strategies for assessing and mitigating the environmental impact of the project. It includes measures to ensure compliance with environmental regulations.

6.4. LEGAL AND COMPLIANCE PLAN:

The Legal and Compliance Plan details the strategies for ensuring that the "ConnectEZ" project adheres to legal requirements and industry regulations. It includes compliance checks, legal reviews, and mechanisms for addressing legal issues.