**UNIVERSITY OF ENERGY AND NATURAL RESOURCES SUNYANI**

**DEPARTMENT OF INFORMATION AND DECISION SCIENCES**



**GROUP 11**

**ADU JUSTIN BOATENG - UEB3211020**

**AMPOMAH MARK-HILL - UEB3200720**

**AWITI KUFFOUR CHRIS - UEB3207420**

**FORDJOUR HESKEY AMOAKO - UEB3202120**

**KUSI AKWASI NYAMEKYE - UEB3209620**

**MONJILIB BRIGHT NYANBADAL - UEB3208820**

**NYAMEKYE MELVIN OSEI - UEB3204820**

**OSEI BOATENG ISSABELLA (MS) - UEB3207820**

**SOLOMON WILLIAM NII OUT - UEB3205420**

**YELINGE JOYCE JEBUNI (MS) - UEB3218720**

**1. Inventory Management:**

**Objective:** To track and manage raw materials, work-in-progress inventory, and finished goods to ensure optimal stock levels and minimize holding costs.

**Functionalities:**

* Real-time inventory tracking.
* Automatic reorder alerts when stock levels fall below predefined thresholds.
* Inventory forecasting using historical data to predict future demand.
* Integration with barcoding or RFID systems for easy identification and tracking.

**Architecture:**

* Database to store inventory records, including quantity, location, and status of each item.
* User interface for inventory queries, updates, and reporting.
* API integration with procurement and sales modules to automatically update inventory levels based on incoming and outgoing goods.

**2. Production Planning and Control:**

**Objective:** To streamline production processes by scheduling equipment and manpower, managing production orders, and monitoring production status.

**Functionalities:**

* Production scheduling tool to allocate resources and plan production runs.
* Order management to track production orders from creation to completion.
* Real-time production monitoring to track the progress of manufacturing processes.
* Quality control integrations to ensure product standards are met.

**Architecture:**

* Integration with inventory management to pull raw materials based on production schedules.
* Database for storing production schedules, order details, and status updates.
* User interface for production managers to create, update, and monitor production orders.

**3. Sales and Order Processing:**

**Objective:** To manage customer orders from placement to delivery, ensuring accurate and timely order fulfillment.

**Functionalities:**

* Customer relationship management (CRM) to store customer information and order history.
* Order tracking from receipt through to delivery.
* Automated invoicing and payment processing.
* Sales forecasting tools to predict future sales trends and prepare inventory accordingly.

**Architecture:**

* Integration with inventory and production modules to ensure product availability and manufacturing capability.
* Database for storing customer data, order details, and transaction records.
* User interface for sales personnel to enter, update, and track orders.

**4. Purchasing and Supplier Management:**

**Objective:** To manage supplier relationships, procure raw materials, and ensure timely delivery of goods.

**Functionalities:**

* Supplier database to store contact details, contract terms, and performance metrics.
* Purchase order management, including creation, approval, and tracking.
* Automated reorder processes based on inventory levels and forecasts.
* Supplier performance tracking to evaluate and optimize supplier relationships.

**Architecture:**

* Integration with inventory management for seamless updating of stock levels upon receipt of goods.
* User interface for creating and managing purchase orders and supplier contracts.
* Database for recording all procurement activities and supplier information.

**5. Finance and Accounting:**

**Objective:** To manage financial transactions, including accounts payable and receivable, payroll, and financial reporting.

**Functionalities:**

* General ledger to track all financial transactions.
* Accounts payable and receivable management.
* Payroll processing and financial compliance tools.
* Financial reporting and analytics for strategic decision-making.

**Architecture:**

* Integration with sales, purchasing, and human resources modules for financial data consolidation.
* Secure database for storing sensitive financial information.
* User interface for financial management and reporting.

**6. Human Resources Management:**

**Objective:** To manage employee information, recruitment, payroll, and performance evaluations.

**Functionalities:**

* Employee database for storing personal and professional information.
* Recruitment and onboarding tools.
* Payroll and benefits management.
* Performance evaluation systems to assess and manage employee performance.

**Architecture:**

* Integration with finance module for payroll processing.
* User interface for HR personnel to manage employee data, recruitment, and appraisals.
* Database for recording all HR-related activities and employee records.

**7. Reporting and Analytics:**

**Objective:** To provide comprehensive insights into business operations through data visualization and analytics.

**Functionalities:**

* Customizable dashboards to monitor key performance indicators (KPIs).
* Reporting tools for generating periodic reports on sales, inventory, production, and financials.
* Data analytics for predictive insights and strategic planning.

**Architecture:**

* Integration with all other ERP modules to pull data for comprehensive analysis.
* Use of Tableau for advanced data visualization and dashboard creation.

**1. Introduction**

**1.1 Purpose**

The purpose of this Enterprise Resource Planning (ERP) system is to streamline and integrate the core business processes of a mini manufacturing company. By automating various functions across departments such as inventory management, production planning, sales, and human resources, the ERP system aims to enhance operational efficiency, improve data accuracy, and support informed decision-making.

**1.2 Scope**

This document outlines the design and functionalities of an ERP system tailored for a mini manufacturing company. The ERP system encompasses the following key modules:

* **Inventory Management:** Manages stock levels and tracks inventory movements.
* **Production Planning and Control:** Schedules and monitors manufacturing processes.
* **Sales and Order Processing:** Manages customer orders from placement to delivery.
* **Purchasing and Supplier Management:** Oversees supplier relationships and purchase orders.
* **Finance and Accounting:** Handles financial transactions and reporting.
* **Human Resources Management:** Manages employee information and payroll.
* **Reporting and Analytics:** Provides insights through data visualization and performance metrics.

These modules are designed to integrate seamlessly to provide a comprehensive view of the company's operations, facilitating better control and optimization of resources.

**1.3 Definitions, Acronyms, and Abbreviations**

* **ERP:** Enterprise Resource Planning.
* **CRM:** Customer Relationship Management.
* **KPI:** Key Performance Indicator.
* **UI:** User Interface.
* **API:** Application Programming Interface.

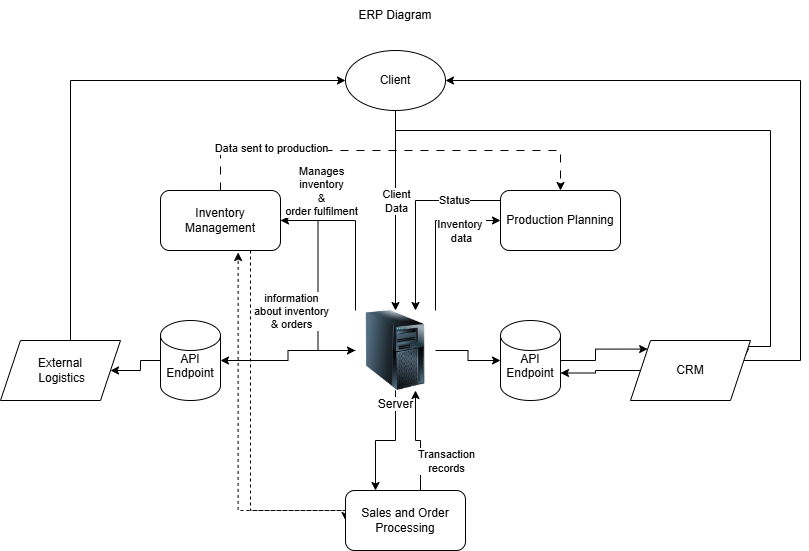
These terms will be used throughout this document to refer to the various components and functionalities of the ERP system.

**2. System Overview**

**2.1 System Architecture**

The ERP system is designed as a modular, integrated platform that allows seamless data flow between different functional areas of the company. Below is a high-level architecture diagram that illustrates the interactions between the core modules:

* **Architecture**:

*The architecture is built on a client-server model, with a central database that stores all transactional and operational data. This setup ensures that all modules access consistent and up-to-date information, facilitating real-time decision-making and reporting.*

Here's the detailed ERP System Architecture Diagram, which illustrates the client-server model with a central database. This diagram shows how various modules such as Inventory Management, Production Planning, Sales and Order Processing interact with each other and external systems, including API endpoints and data flow arrows that represent communication between modules.

**Explanation of the ERP System Architecture Diagram:**

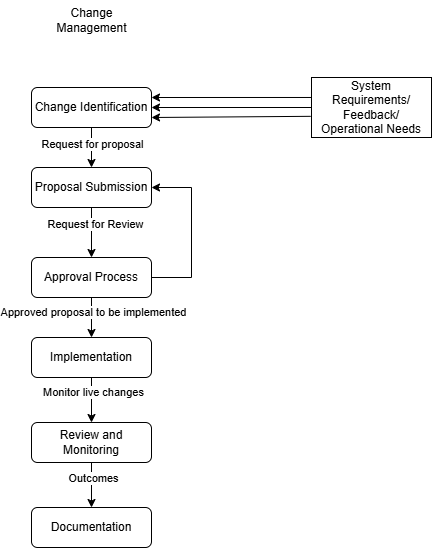
* **Client-Server Model:** The core of the system architecture, indicating that clients (user interfaces) communicate with servers that process data and manage business logic.
* **Central Database:** Central storage for all transactional and operational data, ensuring consistency and real-time accessibility.
* **Modules Interaction:**
  + **Inventory Management:** Manages stock levels and interacts with Sales for order fulfillment and Production for material requirements.
  + **Production Planning:** Receives data from Inventory Management to plan production tasks; sends status updates back to central management to adjust plans dynamically.
  + **Sales and Order Processing:** Processes customer orders, interacts with Inventory for stock checks, and updates the central system with transaction records.
* **External Systems:** Demonstrates integration with external systems like CRM for customer data and external logistics for shipping management.
* **API Endpoints:** Points of communication between modules and external applications, facilitating data exchange and functionality expansion through RESTful services.

**2.2 Technological Stack**

To build and support the ERP system, the following technologies will be utilized:

* **Frontend Development:**
  + **HTML5** and **CSS3** for basic webpage structures and styling.
  + **JavaScript** and frameworks such as **React** or **Vue.js** for dynamic user interfaces and client-side logic.
* **Backend Development:**
  + **Node.js** with **Express** for server-side logic and RESTful API development.
  + **Python** with frameworks like **Django** or **Flask** for complex backend processing.
* **Database:**
  + **PostgreSQL** for relational database management, known for its robustness and scalability.
  + **MongoDB** for handling unstructured data, providing flexibility with document-oriented storage.
* **Integration and Middleware:**
  + **Apache Kafka** for handling data streams and ensuring reliable message passing between different components of the system.
  + **Redis** for caching frequently accessed data to improve system performance.
* **Security:**
  + **OAuth 2.0** for secure authorization.
  + **JWT (JSON Web Tokens)** for secure transmission of information between parties as a JSON object.
* **Analytics and Reporting:**
  + **Tableau** for data visualization and creating interactive dashboards.
  + **Elasticsearch** for performing real-time analytics and searches across the diverse data sets.

**Change Management**



Here is the Data Flow Diagram for the Change Management process in your ERP system. This diagram visually represents the steps involved in managing changes, from identification through proposal, approval, implementation, review, and documentation.

Explanation of the Data Flow Diagram for Change Management:

* Change Identification: The process begins when a change is identified, which could stem from system requirements, user feedback, or operational needs.
* Proposal Submission: Once a change is identified, a detailed proposal is prepared, outlining the scope, impact, resources required, and a proposed timeline.
* Approval Process: The proposal is reviewed by relevant stakeholders, and decision points are included in the diagram to show where approvals are required.
* Implementation: Upon approval, the change is implemented. This stage may involve modifications to the software, updates to processes, or both.
* Review and Monitoring: After implementation, the change is monitored for effectiveness and compliance with expectations. Any issues are addressed, and necessary adjustments are made.
* Documentation: Finally, all changes and outcomes are documented for future reference, ensuring that there is a clear historical record of what was done and why.

This diagram will be particularly useful for stakeholders to understand the flow of activities in change management and to ensure they are aware of their roles in each step of the process.

**Risk Assessment**



Here is the detailed Risk Assessment Process Diagram for your ERP system. This diagram visually represents the steps involved in managing risks, from identification through to continuous monitoring.

**Explanation of the Risk Assessment Process Diagram:**

* **Risk Identification:** The process begins by identifying potential risks that could affect the ERP system, sourced from various inputs like system analysis, user feedback, and external factors.
* **Risk Evaluation:** Once risks are identified, they are evaluated using a risk matrix to determine their likelihood and potential impact on the system.
* **Mitigation Strategies Implementation:** Based on the evaluation, appropriate risk mitigation strategies are developed and implemented to manage or reduce the risks to acceptable levels.
* **Continuous Monitoring:** After implementing mitigation strategies, the process includes continuous monitoring to ensure the effectiveness of the risk management measures. Adjustments are made as necessary based on new information or changes in the system environment.
* **Feedback Loops:** The diagram includes feedback loops to indicate that the risk management process is iterative, with continual inputs and adjustments to improve the system's resilience.

This diagram will help stakeholders understand the structured approach to risk management within the ERP system, ensuring that risks are proactively identified, assessed, and managed to maintain system integrity and reliability.

**3. System Requirements**

**3.1 Functional Requirements**

For each module within the ERP system, specific functionalities are required to meet the operational needs of the mini manufacturing company:

* **Inventory Management:**
  + Track and report stock levels in real-time.
  + Automated alerts for low stock levels.
  + Detailed reporting on inventory turnover and aging.
* **Production Planning and Control:**
  + Schedule production runs based on resource availability and demand forecasts.
  + Track production progress against schedules.
  + Manage and report on production incidents and downtimes.
* **Sales and Order Processing:**
  + Manage customer orders from placement through to fulfillment and invoicing.
  + Provide a customer portal for order tracking and history.
  + Integrate with inventory management for real-time product availability.
* **Purchasing and Supplier Management:**
  + Create and manage purchase orders.
  + Track supplier performance and compliance.
  + Integrate with finance for payment processing.
* **Finance and Accounting:**
  + Support all standard accounting functions, including ledger management, accounts payable/receivable, and financial reporting.
  + Ensure compliance with financial regulations and reporting requirements.
  + Provide comprehensive financial analytics and forecasts.
* **Human Resources Management:**
  + Manage employee records, including personal data, employment history, and payroll.
  + Support recruitment, onboarding, and training processes.
  + Evaluate and report on employee performance.
* **Reporting and Analytics:**
  + Provide customizable dashboards for real-time data monitoring.
  + Support detailed analytical reports across all modules.
  + Include data export capabilities for further analysis.

**3.2 Non-functional Requirements**

To ensure the ERP system is efficient, reliable, and scalable, the following non-functional requirements must be met:

* **Usability:** The system should have an intuitive user interface that minimizes training requirements and enhances user adoption.
* **Performance:** The system must perform efficiently under load, with capabilities to handle increases in user numbers and data volume without degradation of performance.
* **Security:** Robust security protocols must be in place to protect sensitive data, including encryption of data in transit and at rest, and secure authentication mechanisms.
* **Scalability:** The system should be scalable both in terms of hardware and software to accommodate future growth.
* **Reliability:** High availability and fault tolerance should be built into the system to ensure continuous operation.
* **Compliance:** The system must comply with applicable laws and regulations regarding data protection and financial reporting.

**4. Module Descriptions**

**4.1 Inventory Management**

**Objective:** To manage stock levels efficiently, ensuring availability while minimizing excess inventory.

**Functionalities:**

* Real-time tracking of inventory levels.
* Automated reorder notifications based on predefined thresholds.
* Detailed inventory reporting and analysis tools.

**Data Flow:**

* Data flows between the purchasing module (for incoming stock) and sales module (for outgoing stock), updating real-time inventory levels.

**Database Design:**

* Tables for stock items, stock levels, reorder levels, and transaction logs.
* Relationships between stock items and orders to track inventory movements.

**User Interface Design:**

* Dashboard for viewing current stock levels.
* Screens for setting up and managing reorder levels and notifications.

**4.2 Production Planning and Control**

**Objective:** To optimize the production process, ensuring efficient use of resources and timely production output.

**Functionalities:**

* Production scheduling based on machine and manpower availability.
* Tracking of production status against planned schedules.
* Management of production incidents and resolutions.

**Data Flow:**

* Integrated with Inventory Management for raw material availability and Human Resources for manpower scheduling.

**Database Design:**

* Tables for production schedules, machine availability, production orders, and incident logs.

**User Interface Design:**

* Calendar view for production schedules.
* Detailed views for production order statuses and incident management.

**4.3 Sales and Order Processing**

**Objective:** To manage customer orders efficiently from placement to delivery, enhancing customer satisfaction.

**Functionalities:**

* Order entry, tracking, and fulfillment management.
* Customer relationship management integration.
* Invoicing and payment processing.

**Data Flow:**

* Direct interaction with the inventory management to check product availability and with finance for invoicing.

**Database Design:**

* Tables for customer details, orders, order statuses, and payment records.

**User Interface Design:**

* Interface for order entry and status tracking.
* CRM features for managing customer interactions and history.

**4.4 Purchasing and Supplier Management**

**Objective:** To streamline procurement processes and manage supplier relationships effectively.

**Functionalities:**

* Purchase order creation and management.
* Supplier performance tracking and management.
* Integration with inventory for automatic stock updates.

**Data Flow:**

* Receives data from Inventory Management for stock needs and sends data to Finance for payment processing.

**Database Design:**

* Tables for suppliers, purchase orders, delivery notes, and supplier performance metrics.

**User Interface Design:**

* Screens for creating and managing purchase orders.
* Dashboards for monitoring supplier performance.

**4.5 Finance and Accounting**

**Objective:** To manage all financial transactions and reporting with compliance and accuracy.

**Functionalities:**

* General ledger management, accounts payable and receivable.
* Financial reporting and compliance tools.
* Payroll and budgeting modules.

**Data Flow:**

* Interacts with all modules, collecting financial data for processing and reporting.

**Database Design:**

* Tables for financial transactions, accounts, budgets, and financial reports.

**User Interface Design:**

* Financial dashboards for real-time monitoring.
* Detailed financial reports and ledger management screens.

**4.6 Human Resources Management**

**Objective:** To manage all aspects of employee administration, from recruitment to retirement.

**Functionalities:**

* Employee database management.
* Recruitment, payroll, and performance evaluation tools.
* Employee self-service portal for personal data updates and benefits management.

**Data Flow:**

* Data exchange with Finance for payroll and with all modules for access control.

**Database Design:**

* Tables for employee records, payroll data, performance reviews, and benefits.

**User Interface Design:**

* Interfaces for HR management.
* Employee portal for personal and professional information management.

**4.7 Reporting and Analytics**

**Objective:** To provide insights and decision-support tools through data analysis and reporting.

**Functionalities:**

* Customizable dashboards for various KPIs.
* Reporting tools for operational, financial, and HR data.
* Data export capabilities for external analysis.

**Data Flow:**

* Pulls data from all modules to provide integrated reporting and analytics.

**Database Design:**

* Data warehouse architecture to support analytics.
* Integration with external analytics tools like Tableau.

**User Interface Design:**

* Interactive dashboards for real-time data visualization.
* Tools for creating and managing reports.

**5. Integration**

**5.1 Module Integration**

**Objective:** To ensure seamless data exchange and functionality between all ERP modules, enhancing the overall system efficiency and reducing redundancy.

**Strategies:**

* **Centralized Database:** All modules will interact with a centralized relational database which ensures data consistency and integrity. This setup minimizes data duplication and allows real-time data access across modules.
* **API Layer:** Implement a robust API layer using RESTful services, which allows different modules to communicate and exchange data efficiently. This layer acts as a bridge between frontend applications and backend services, ensuring that data flows are secure and scalable.
* **Event-Driven Architecture:** Utilize an event-driven architecture with a messaging system like Apache Kafka to handle events such as stock updates, order placements, or payroll processing. This approach ensures that all related modules receive updates in real-time and can react to changes immediately.
* **Common Services:** Develop common services for tasks that are used by multiple modules, such as authentication, logging, notification services, and data validation. This reduces the need to reimplement these functionalities in multiple modules and simplifies maintenance.

**5.2 External Integrations**

**Objective:** To enable the ERP system to interact with external systems, enhancing its capabilities and providing more comprehensive services.

**Strategies:**

* **Third-Party Services Integration:** Integrate with third-party services such as payment gateways, email services, and external CRM systems. Use standardized APIs to connect with these services, ensuring that data exchange is secure and efficient.
* **IoT Integration:** For modules like inventory management and production planning, integrate IoT devices directly into the ERP system. Use IoT gateways to collect data from sensors on production equipment or warehouse inventory systems, and feed this data into the ERP for real-time monitoring and decision-making.
* **Data Synchronization:** Implement data synchronization mechanisms to ensure that data shared with external systems remains consistent with the data in the ERP system. Employ techniques such as timestamping and checksums to validate data integrity during synchronization.
* **Compliance and Security Protocols:** Ensure that all external integrations comply with relevant security standards and regulations. Implement robust authentication and encryption protocols to protect data exchanged with external systems.

**6. Security and Compliance**

**6.1 Security Measures**

**Objective:** To safeguard sensitive data and protect the ERP system from unauthorized access and potential security threats.

**Strategies:**

* **Data Encryption:** Employ encryption both at rest and in transit. Use protocols such as TLS for securing data during transmission and AES for encrypting data stored in databases.
* **User Authentication and Authorization:** Implement robust user authentication mechanisms using OAuth 2.0 and OpenID Connect. Use role-based access control (RBAC) to ensure users can only access data and functionalities necessary for their roles.
* **Regular Security Audits:** Conduct regular security audits and penetration testing to identify and address vulnerabilities. Employ automated tools and engage third-party security firms to perform thorough assessments.
* **Security Patches and Updates:** Establish a routine for applying security patches and updates to software components. Monitor security advisories for the technology stack used in the ERP system and apply updates promptly.
* **Intrusion Detection and Prevention Systems (IDPS):** Install IDPS to monitor network traffic and system activities for suspicious behavior. Configure alarms to alert administrators of potential security breaches.

**6.2 Compliance**

**Objective:** To ensure that the ERP system complies with relevant legal, regulatory, and industry standards, particularly concerning data protection and financial reporting.

**Strategies:**

* **Data Protection Compliance:** Adhere to data protection laws such as GDPR in the EU or CCPA in California, depending on the geographical operation areas. Implement features for data anonymization, rights to access, and the ability to erase personal data.
* **Financial Compliance:** Ensure the finance and accounting modules comply with international financial reporting standards (IFRS) and local regulations. Integrate functionalities to support audit trails, transparent reporting, and accurate tax calculations.
* **Accessibility Standards:** Follow accessibility guidelines such as the Web Content Accessibility Guidelines (WCAG) to make the ERP system usable for people with disabilities. This includes providing alternative text for images, ensuring user interface components are accessible via keyboard, and supporting screen readers.
* **Industry-Specific Regulations:** Comply with industry-specific regulations, such as those pertaining to manufacturing, healthcare, or retail. This may involve tracking specific data points, maintaining certain operational standards, or reporting in particular formats.

**7. Implementation Plan**

**7.1 Deployment Strategy**

**Objective:** To ensure a smooth and effective deployment of the ERP system with minimal disruption to ongoing operations.

**Steps:**

* **Phase 1: Preparation and Planning**
  + Conduct a thorough needs assessment and finalize system requirements.
  + Prepare the infrastructure, including server setup, database initialization, and security measures.
* **Phase 2: System Installation and Configuration**
  + Install the ERP software components on designated servers.
  + Configure system settings, including user roles, permissions, and workflow rules.
* **Phase 3: Data Migration**
  + Migrate existing data from old systems or databases to the new ERP system.
  + Validate data integrity and accuracy after migration.
* **Phase 4: Testing**
  + Conduct comprehensive testing, including unit testing, system integration testing, and user acceptance testing (UAT).
  + Fix any issues identified during testing phases.
* **Phase 5: Go-Live and Monitoring**
  + Officially launch the ERP system for full operational use.
  + Monitor system performance and user feedback closely for the initial weeks.
* **Phase 6: Post-Implementation Review**
  + Conduct a post-implementation review to evaluate the system's performance and identify any areas for improvement.
  + Plan and implement necessary updates or enhancements.

**7.2 Training Plan**

**Objective:** To equip all users with the necessary skills and knowledge to effectively use the ERP system.

**Strategies:**

* **Role-Based Training:** Provide tailored training sessions based on user roles and responsibilities within the ERP system.
* **Online Training Modules:** Develop comprehensive online tutorials and documentation to facilitate self-learning and reference.
* **Hands-On Workshops:** Organize interactive workshops where users can practice using the system with real data.
* **Continuous Education:** Offer ongoing training sessions to cover updates, new features, and advanced functionalities.

**7.3 Support and Maintenance**

**Objective:** To ensure the ERP system remains functional, secure, and up-to-date with business needs.

**Strategies:**

* **Help Desk Support:** Establish a dedicated help desk for users to report issues, request assistance, and provide feedback.
* **Regular Maintenance:** Schedule regular maintenance for the ERP system, including updates, backups, and security checks.
* **Performance Monitoring:** Continuously monitor system performance and address any efficiency or reliability issues.
* **User Feedback Loop:** Implement a feedback loop to gather and analyze user feedback for ongoing system improvements.

**8. Appendices**

**8.1 Glossary**

This section will define key terms, acronyms, and abbreviations used throughout the document to aid understanding.

* **ERP (Enterprise Resource Planning):** A system of integrated software applications that manages business processes and automates many back-office functions related to technology, services, and human resources.
* **API (Application Programming Interface):** A set of routines, protocols, and tools for building software and applications, which allows different software entities to communicate with each other.
* **KPI (Key Performance Indicator):** A measurable value that demonstrates how effectively a company is achieving key business objectives.
* **UI (User Interface):** The space where interactions between humans and machines occur, aimed at effective operation and control of the machine from the human end.
* **TLS (Transport Layer Security):** A cryptographic protocol designed to provide communications security over a computer network.
* **AES (Advanced Encryption Standard):** A symmetric encryption standard used to secure data.
* **RBAC (Role-Based Access Control):** A policy-neutral access-control mechanism defined around roles and privileges.
* **IoT (Internet of Things):** The interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data.
* **GDPR (General Data Protection Regulation):** A regulation in EU law on data protection and privacy in the European Union and the European Economic Area.
* **CCPA (California Consumer Privacy Act):** A statute intended to enhance privacy rights and consumer protection for residents of California, United States.

**8.2 References**

* **React Documentation:** Available at https://reactjs.org/docs/getting-started.html
* **Node.js Documentation :** Available at https://nodejs.org/en/docs/
* **PostgreSQL Documentation :** Available at <https://www.postgresql.org/docs/>
* **Python Django Framework Documentation:** Available at <https://docs.djangoproject.com/>
* **Tableau User Guide:** Available at https://help.tableau.com/current/guide/en-us/index.html
* **ISO/IEC 27001 Information Security Management:** Available at https://www.iso.org/isoiec-27001-information-security.html