**Software Requirements Specification**

**For**

**UNIVERSITY ENTERPRISE RESOURCE PLANNING**

**Version 0.1 Approved**

**Prepared by: OKOMO PETER ODHIAMBO**

**REG NO: SIT/B/01-02925/2022**

**PHONE NO :0797561261 MASINDE MULIRO UNIVERSITY OF SCIENCE AND**

**TECHNOLOGY**

**29TH SEPTEMBER,2024**

Copyright © 1999 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.

Table of Contents

[1. Revision History vi](#_Toc50532)

[2. Introduction 7](#_Toc50533)

[2.1 Purpose 7](#_Toc50534)

[2.2 Document Conventions 7](#_Toc50535)

[2.2.1 Formatting and styling 7](#_Toc50536)

[2.2.2 Document templates 7](#_Toc50537)

[2.2.3 Storage and Version Control System: 7](#_Toc50538)

[2.3 Intended Audience and Reading Suggestions 8](#_Toc50539)

[2.4 Product Scope 10](#_Toc50540)

[2.5 References 12](#_Toc50541)

[3. Overall Description 14](#_Toc50542)

[3.1 Product Perspective 14](#_Toc50543)

[3.2 Product Functions 15](#_Toc50544)

[3.3 User Classes and Characteristics 16](#_Toc50545)

[3.4 Operating Environment 16](#_Toc50546)

[3.5 Design and Implementation Constraints 17](#_Toc50547)

[3.6 User Documentation 17](#_Toc50548)

[3.7 Assumptions and Dependencies 18](#_Toc50549)

[3.7.1 Assumptions include: 18](#_Toc50550)

[3.7.2 Dependencies include: 18](#_Toc50551)

[4. External Interface Requirements 19](#_Toc50552)

[4.1 User Interfaces 19](#_Toc50553)

[4.2 Hardware Interfaces 23](#_Toc50554)

[4.2.1 Hardware Interfaces for University ERP System - cafeteria 23](#_Toc50555)

[4.2.2 Logical Characteristics 23](#_Toc50556)

[4.2.3 Physical Characteristics 23](#_Toc50557)

[4.2.4 Supported Device Types 24](#_Toc50558)

[4.2.5 Nature of Data and Control Interactions 24](#_Toc50559)

[4.2.6 Communication Protocols 24](#_Toc50560)

[4.3 Interfaces 24](#_Toc50561)

[4.3.1 Connections Software to Other Software Components 24](#_Toc50562)

[4.3.2 Data Items and Messages 25](#_Toc50563)

[4.3.3 Purpose of Each Data Item or Message 25](#_Toc50564)

[4.3.4 Services Needed 25](#_Toc50565)

[4.3.5 Nature of Communications 26](#_Toc50566)

[4.3.6 Application Programming Interface Protocols 26](#_Toc50567)

[4.3.7 Data Sharing 26](#_Toc50568)

[4.3.8 Implementation Constraints 26](#_Toc50569)

[4.4 Communications Interfaces 26](#_Toc50570)

[4.4.1 communications interfaces for UNIVERSITY ERP System - cafeteria 26](#_Toc50571)

[4.4.2 Message Formatting 27](#_Toc50572)

[4.4.3 Communication Standards 27](#_Toc50573)

[4.4.4 Communication Security and Encryption 27](#_Toc50574)

[4.4.5 Data Transfer Rates 28](#_Toc50575)

[4.4.6 Synchronization Mechanisms 28](#_Toc50576)

[5. System Features: University Cafeteria 29](#_Toc50577)

[5.1 System Feature 1: Menu Planning and Management 29](#_Toc50578)

[5.1.1 Description 29](#_Toc50579)

[5.1.2 Stimulus and Response: 29](#_Toc50580)

[5.1.3 Functional Requirements: 29](#_Toc50581)

[5.1.4 Response: 29](#_Toc50582)

[5.2 System Feature 2: Ordering and Payment 29](#_Toc50583)

[5.2.1 Description: 29](#_Toc50584)

[5.2.2 Stimulus and Response: 30](#_Toc50585)

[5.2.3 Functional Requirements: 30](#_Toc50586)

[5.2.4 Response: 30](#_Toc50587)

[5.3 System Features 3: Inventory Management 30](#_Toc50588)

[5.3.1 Description: 30](#_Toc50589)

[5.3.2 Stimulus and Response 30](#_Toc50590)

[5.3.3 Functional Requirements: 30](#_Toc50591)

[5.3.4 Response: 31](#_Toc50592)

[5.4 System Features 4: Food Safety Management 31](#_Toc50593)

[5.4.1 Description: 31](#_Toc50594)

[5.4.2 Stimulus/Response Sequences: 31](#_Toc50595)

[5.4.3 Functional Requirements: 31](#_Toc50596)

[5.4.4 Response: 31](#_Toc50597)

[5.5 System Features 5: Customer Feedback and Reporting 31](#_Toc50598)

[5.5.1 Description 31](#_Toc50599)

[5.5.2 Stimulus/Response Sequences: 32](#_Toc50600)

[5.5.3 Functional Requirements 32](#_Toc50601)

[5.5.4 Response: 32](#_Toc50602)

[6. Other Nonfunctional Requirements 33](#_Toc50603)

[6.1 Performance Requirements 33](#_Toc50604)

[6.2 Safety Requirements 34](#_Toc50605)

[6.3 Security Requirements 35](#_Toc50606)

[6.4 Software Quality Attributes 38](#_Toc50607)

[6.5 Business Rules 40](#_Toc50608)

[7. Other requirements 42](#_Toc50609)

[7.1 Database Requirements 42](#_Toc50610)

[7.2 Internationalization Requirements 42](#_Toc50611)

[7.3 Legal Requirements 42](#_Toc50612)

[7.4 Reuse Objectives 42](#_Toc50613)

[7.5 User Documentation 43](#_Toc50614)

[7.6 Performance Testing 43](#_Toc50615)

[Appendix A: Glossary 44](#_Toc50616)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| **OKOMO PETER** | **29/09/202**  **4** |  | **0.1** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

This is the 0.1 approved version of the University ERP.The purpose of the University Enterprise Resource Planning (ERP) software is to develop and implement a comprehensive and integrated software solution that addresses the specific needs and challenges faced by universities and higher education institutions.

The product scope is Student Management, Student affairs, Faculty Management, Library Management, Fleet management, Financial Management, Human Resources Management, Clinic management, Timetable management and enterprise management.

## Document Conventions

In order to facilitate a clear and consistent understanding of this document, the following conventions are employed:

### Formatting and styling

1. Heading: font style – Times new roman, Font size - 16 pt.
2. Sub Heading: font style – Times new roman, Font size - 14 pt,
3. Standard text: font style – Times new roman, Font size - 12 pt,

### Document templates

1. Standard templates(SRS) is used for various types of project documentation, such as requirements documents, design documents, and test plans.
2. Templates will provide a consistent structure and format, ensuring that important sections and information are included.

### Storage and Version Control System:

1. A centralized document storage system, such as a shared drive or version control system, in this document are used to manage and organize project document.
2. This documents are stored in appropriate folders and categorized based on their type and purpose.

## Intended Audience and Reading Suggestions

The University Enterprise Resource Planning (ERP) software is intended for a diverse audience involved in different stages of the software development life-cycle and the university ecosystem. The primary target audience includes:

**i) Project Stakeholders:**

* University administrators and management responsible for strategic decision-making and resource allocation.
* IT department personnel involved in system implementation, maintenance, and support.
* Faculty members, academic staff, and department heads interacting with the ERP system for academic and administrative purposes.
* Students who will utilize the system for accessing information, managing their academic journey, and engaging in university activities.

**ii)Software Development Team:**

* Software engineers, developers, and architects responsible for designing, developing, and testing the ERP system.
* Business analysts who gather and document requirements from stakeholders.
* Quality assurance professionals involved in testing and ensuring the software meets specified requirements.
* Project managers overseeing the planning, execution, and delivery of the ERP software. **ii) System Users and End-Users:**
* Staff members and administrators across various departments who will use the ERP system to manage administrative tasks, such as admissions, finance, human resources, and student records.
* Faculty members who will utilize the system for course management, grading, and student interaction.
* Students who will engage with the ERP system for course registration, accessing grades, and other academic activities.

**Contents of the rest of the SRS**

I. Overview Section: II. Purpose:

1. Scope:
2. Definitions, Acronyms, and Abbreviations: V. References:
3. Overall Description:
4. Functional Requirements:
5. Non-Functional Requirements:
6. Appendix.

**Reading suggestions**

To maximize the usefulness of this document, readers are encouraged to follow a suggested sequence:

1. **Overview Section**:

 Reading the Overview section. This section provides a high-level summary of the entire document, offering a quick understanding of the project's scope and purpose.

1. **Purpose**:
   * Purpose section. This section elaborates on why the University ERP is being developed and what problems it aims to solve.
2. **Scope**:
   * The Scope section provides information about the boundaries of the project. Reading this section provides understanding on what features and functionalities are within the project's scope and what is not.
3. **Definitions, Acronyms, and Abbreviations**:
   * This section contains a list of terms, acronyms, and abbreviations used throughout the document. Familiarize yourself with these to understand the terminology used.
4. **References**:
   * The external documents or references that are relevant to the SRS are provided.
5. **Overall Description**:
   * In this section, there is more detailed description of the University ERP, including its functions and user profiles. There is a sense of the system's main features and goals.
6. **Functional Requirements**:
   * This is one of the most important sections. There is an outline of the specific functions and capabilities the University ERP must have. It defines what the system will do.
7. **Non-Functional Requirements**:
   * Non-functional requirements cover aspects like performance, security, scalability, and more. Understanding these requirements is crucial for designing and developing the system correctly.
8. **System Features**:
   * This section provides a detailed breakdown of the system's features and how they relate to the functional requirements. This helps in understanding the system's behavior.
9. **External Interface Requirements**:
   * How the University ERP interacts with external systems and users, this includes APIs, data exchange, and user interfaces.
10. **System Interfaces**:
    * There is brief on interfaces within the system itself, how different modules or components interact with one another.
11. **User Interfaces**:
    * Understanding on how the user interfaces look and function, This include descriptions, wireframes, or mockups.
12. **Hardware Interfaces**:
    * This section describes systems that interact with specific hardware components.
13. **Software Interfaces**:
    * Gives an explanation about the software and services the University ERP relies on and how they communicate with each other.
14. **Communication Protocols**:
    * It explains the specific communication Protocols used.
15. **Performance Requirements**:
    * Understanding the performance expectations, such as response times and system capacity.
16. **Security Requirements**:
    * This section details the security measures and requirements for protecting data and system functionality.
17. **Software Quality Attributes**:
    * This system adheres to attributes like reliability, maintainability, and usability.
18. **Other Requirements**:
    * This section contains additional requirements that don't fit into the previous categories.
19. **Appendices**:
    * Checks if there are any appendices with supplementary information, diagrams, or other relevant documents.
20. **Glossary**:
    * Finally, it reviews the glossary to ensure you understand all the technical terms and acronyms used in the document.

## Product Scope

The product scope for the University Enterprise Resource Planning (ERP) software encompasses the following functionalities and features:

**i)Student Management:**

* Student records management, including personal information, academic history, and grades.
* Student portal for accessing information, submitting assignments, and interacting with faculty.

**ii)Student affairs.**

* Registration and enrollment management for students.  Course registration, scheduling, and management.

**iii)Faculty Management:**

* Faculty profile management, including personal information, qualifications, and areas of expertise.
* Faculty portal for accessing student information, managing courses, and communication with students.

**iv) Library Management**

* Cataloging, lending, and tracking of library resources.

**v)Fleet management**

* Management of university vehicles, including maintenance scheduling and usage tracking.

**vi)Financial Management:**

* Budgeting and financial planning for the university.
* Tuition and fee management, including payment processing and tracking.
* Financial aid management, including scholarship and grant processing.
* Expense management, including reimbursement requests, vendor payments, and budget tracking.

**vii)Human Resources Management:**

* Employee profile management, including personal information, employment history, and qualifications.
* Recruitment and on-boarding processes.
* Leave and attendance management.
* Payroll processing and benefits administration.

**viii)Clinic management:**

* Management of university health clinic appointments, medical records, and health services.

**ix) Timetable management:**

* + Provide dashboards and data visualization tools for administrators, faculty, and staff to monitor and analyze university data.

**x)Enterprise management**

* + Creation and management of academic and event timetables

**Purpose**

The purpose of University ERP (Enterprise Resource Planning) software is to provide educational institutions with a centralized and integrated system that efficiently manages and automates various administrative, academic, and financial processes. It aims to streamline operations, enhance decision-making, and provide a seamless experience for students, faculty, and administrative staff.

**Goals of University ERP Software:**

1. **Operational Efficiency**: The goal of University ERP software aims to improve operational efficiency by automating manual processes, reducing paperwork, and eliminating redundant tasks. It aims to streamline administrative workflows, optimize resource allocation, and enhance productivity across the institution.

1. **Improved Decision-Making**: University ERP software aims to provide administrators with accurate and real-time data on student performance, faculty workload, financial trends, and other key metrics. This data-driven approach enables informed decisionmaking, helps identify areas for improvement, and supports evidence-based strategies for enhancing education quality and administrative effectiveness.

1. **Enhanced Student Experience**: A key goal of University ERP software is to enhance the student experience by providing self-service portals for admissions, course registration, fee payment, grade tracking, and access to academic resources. It aims to empower students with easy access to information, efficient communication channels, and personalized support, thereby improving engagement and satisfaction.

**Objectives of University ERP Software:**

1. **Streamline Administrative Processes**: University ERP software aims to automate and streamline administrative processes such as admissions, enrollment, student records management, course scheduling, faculty management, payroll, and financial operations. It seeks to eliminate manual errors, reduce paperwork, and ensure efficient data management across departments.

1. **Improve Academic Management**: University ERP software aims to improve academic management by providing modules for tracking student progress, managing course registration, grading, faculty workload, and communication. It aims to facilitate effective collaboration between faculty and students, enhance academic planning, and ensure timely feedback and support.

1. **Optimize Resource Utilization**: University ERP software aims to optimize resource utilization by providing insights into student enrollment, faculty workload, classroom allocation, and financial trends. It aims to enable universities to make informed decisions about resource allocation, course offerings, infrastructure planning, and budgeting, thereby improving efficiency and cost-effectiveness.

**Benefits of University ERP Software:**

1. **Efficiency and Productivity**: University ERP software improves efficiency by automating processes, reducing manual efforts, and eliminating duplication of tasks. It enhances productivity by providing streamlined workflows, easy access to information, and seamless collaboration among different departments.

1. **Accurate and Real-time Data**: University ERP software ensures accurate and real-time data management, reducing errors and providing administrators with up-to-date information for decision-making. It enables data-driven insights and reporting, leading to improved transparency and accountability.

1. **Improved Student Success**: With self-service portals, academic tracking, and personalized support, University ERP software enhances the student experience and success. It enables efficient communication, timely feedback, and access to academic resources, fostering engagement and facilitating student achievement.

1. **Cost Savings**: University ERP software helps reduce costs by eliminating manual processes, reducing paperwork, and optimizing resource allocation. It minimizes administrative overhead, improves financial management, and enables better budgeting and financial decision-making.

1. **Compliance and Reporting**: University ERP software ensures compliance with regulatory requirements, data privacy policies, and reporting obligations. It simplifies processes for financial reporting, accreditation reporting, and government compliance, reducing the risk of errors and penalties.

## References

The following external references may be useful for readers seeking additional context or related information:

1. *IEEE 12207: Standard for Software Life Cycle Processes.*

1. *IEEE 1012: Standard for System and Software Verification and Validation*

1. *ISO 24641: 2023: ISO reference on software engineering*

1. *ICTA-6.002:2019: ICT authority on system and application development.*

# Overall Description

## Product Perspective

The University ERP System is a context of a new, self-contained software product designed to streamline and optimize the administrative and academic operations of a university. It is a replacement of ABNO ERP existing system and it is being developed as a comprehensive solution to manage various university functions efficiently. The ERP system will integrate different modules that are essential for university management, including University Enterprises, Human Resource, Academics and financial operations. It will facilitate data sharing across different departments and provide a unified platform for managing university resources.

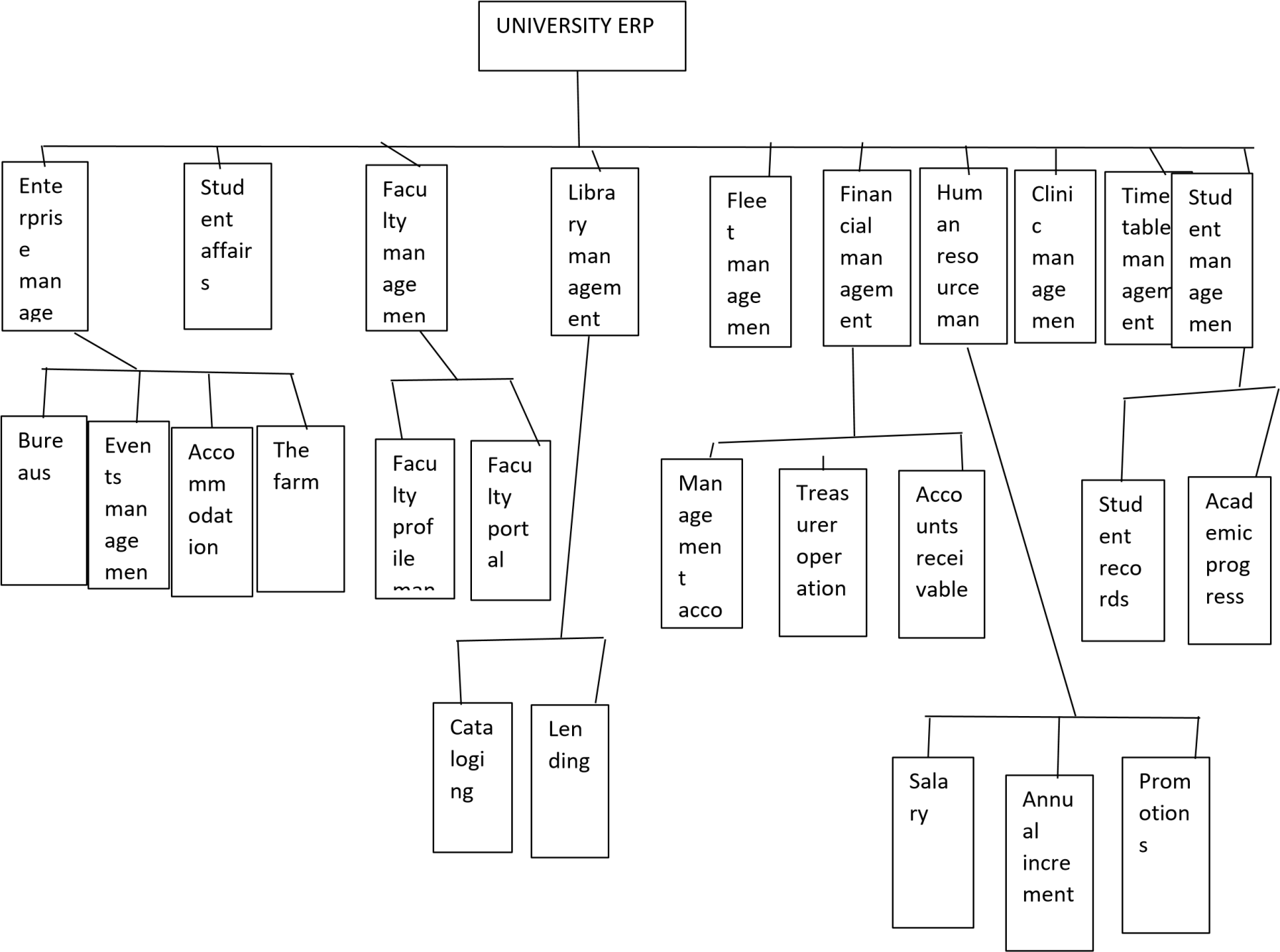


Figure 1: UNIVERSITY ERP system.

## Product Functions

The major functions of the University ERP System include:

University Enterprises: These are sources responsible for generating income to the university. Human Resource: Department that is responsible to ensure the university is able to achieve success through manpower.

Course Scheduling: Creating and managing course schedules, room assignments, and class registration.

Financial Operations: Managing tuition fees, financial aid, payroll, and budgeting.

Academics: Ensures students have Reported and registered for units among others.

Enterprise

Module

name,

description,

manager

manageBurea

us (),

manageAcco

mmodation (),

Bureau

name,

location,

services

~~provideServices~~

()

EventsManage

ment

name,

type,

capacity,

cost

Farm

name,

size,

Location

manageOrders ()

Accommodation

Name

type,

capacity,

cost

bookAccommoda

ti

on ()

cafeteria

meal,

cost,

manageOrder ()

Figure 2: Enterprise Module UNIVERSITY ERP system.

## User Classes and Characteristics

User classes and characteristics for the cafeteria under university enterprises include:

1. Student:

Characteristics: Interact with the system to make orders and payments. Relevant requirements: Access orders.

1. Cashier:

Characteristics: Interact with the system to approve orders and approve payments. Relevant requirements: Access orders, access payments.

1. Service person:

Characteristics: Interact with the system to obtain what a user requires from the café. Relevant requirements: Access orders, access table numbers

1. Chef:

Characteristics: Responsible for food preparation, menu planning, and ensuring kitchen efficiency.

Relevant Requirements: Access to recipe database, menu planning tools, kitchen inventory.

1. Store keeper:

Characteristics: Manage and organize inventory, track stock levels, and update product availability.

Relevant Requirements: Inventory management tools, order tracking, stock level alerts.

1. Finance officer:

Characteristics: Handle financial transactions, track expenses, and manage budget for the cafeteria.

Relevant Requirements: Access to financial reports, budgeting tools, expense tracking.

1. Procurement officer:

Characteristics: Responsible for sourcing suppliers, negotiating contracts, and managing procurement processes.

Relevant Requirements: Supplier management tools, contract negotiation features, order tracking.

## Operating Environment

The University Enterprise System will operate in the following environment:

**Hardware Platform**

The system will run on Amazon Web Services AWS.

**Operating System**

The system will run onRed Hat Enterprise Linux

**Software Components**

The system will rely on an Oracle Database 19c, Apache Tomcat 9 webserver, Java Development Kit 11 for its operation.

**Web Browsers**

Users will access the system through standard web browsers such as Google Chrome, Mozilla Firefox and Microsoft Edge.

## Design and Implementation Constraints

Constraints for the University ERP System include:

**Regulatory Compliance**

The system must adhere to data protection and privacy regulations specific to Kenya and university policies.

**Security**

Robust security measures, including encryption, access controls, and regular security audits, will be implemented to protect sensitive data.

**Database Technology**

The system will use a n Oracle database management system and will follow its data modeling standards.

## User Documentation

The University Enterprise System is a powerful tool that can help you streamline work and improve efficiency. But with any new software, there is a learning curve. That's why we provide comprehensive user documentation to help users get the most out of the system quickly and easily.

Our user documentation includes a variety of resources including

**User manuals**

User manuals provide step-by-step instructions on how to use all of the features of the University Enterprise System. They are written in clear and concise language, and they include screenshots and diagrams to help you visualize the steps involved.

**Online help resources**

The online help resources provide context-sensitive assistance as you work in the system. If you get stuck, simply click the help button and you will be taken to a relevant article or video tutorial.

**Tutorials**

The tutorials provide hands-on training on how to use specific features of the University Enterprise System. They are available in a variety of formats, including text, video, and interactive exercises.

All of our user documentation is available in electronic formats compatible with common document viewers and browsers. So, whether you are using a desktop computer, a laptop, or a mobile device, you can always access the information you need when you need it.

## Assumptions and Dependencies

### Assumptions include:

Availability of necessary hardware and software components.

Compliance with data protection and privacy regulations.

User training will be provided for effective system utilization.

### Dependencies include:

Integration with external authentication systems, google, for user authentication.

*Availability of technical support for system maintenance and troubleshooting*

# External Interface Requirements

## User Interfaces

The user interfaces (UIs) for the University Enterprise System will be web-based and accessible via standard web browsers. They will adhere to usability standards and include elements such as menu bars, buttons, search functions, and error message displays. In addition, the UIs will be designed to be consistent with the university's branding and style guide.

The UIs will be tailored to meet the needs of the university enterprise system in the following ways

**Role-based access control**

The UIs will provide different levels of access to different users, depending on their role in the university. For example, students will have access to different features of the UI than faculty or staff.

**Integration with other university systems**

The UIs will be integrated with other university systems, such as the Human Resource System and the Financial System. This will allow users to access information and complete tasks from a single location.

**a) User Authentication** – Authenticates students, finance officer, Store keeper, procurement officer and service person into the system

Figure 3:User Authentication Login Page

 LOGIN

Don’t’ have an account? Register

*EMAIL ADDRESS*

*Password*

Forgot password

LOGIN

Figure 4: User Authentication Register Page

REGISTER

Already have an account? Login

*F\_Name*

*L\_Name*

*EMAIL*

*Password*

Forgot password

Register

1. **Cafeteria Menu** 
   * Purpose: Provide students and customers with access to available meals and a way of making payments for them.
   * Target Users: Students.
   * Key Features and Functionality:

o Billing – A way of making payments for the meals ordered

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Menu |  | My order |  | View orders |

 Gathura Cafe

|  |
| --- |
| + |

|  |  |  |
| --- | --- | --- |
| - |  | + |

TEA - +

**Order Summary**

**Item**

**QTY**

**Price**

TEA

OMENA

UGALI

CHAPATI

1

1

1

1

k

10

sh

50

ksh

ksh

30

20

ksh

Total

110

ksh

Checkout

CHAPATI - +

UGALI -

OMENA

*Thank you for choosing J-squarecafé!!!*

Figure 5: Cafeteria Menu

1. **Store Keeper Inventory Management** 
   * Purpose: Enable store keeper to keep track of stocks used and purchased in the cafeteria.
   * Target Users: Store keeper.
   * Key Features and Functionality:

Add Product – Enables store keeper to add new products in for cafeteria.

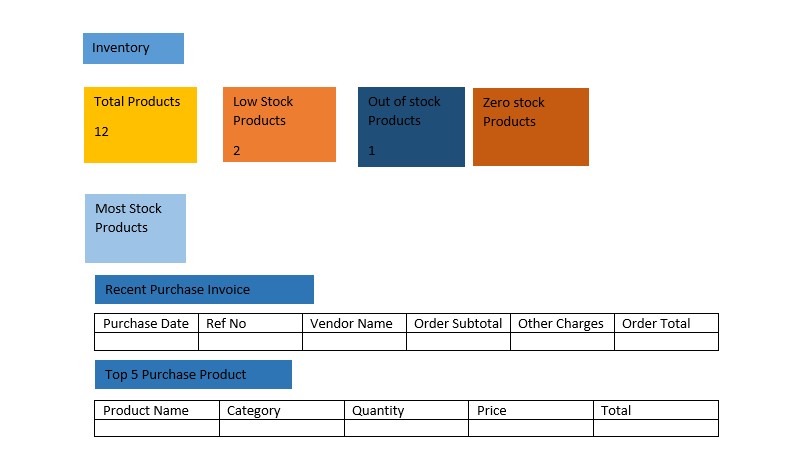


Figure 6: Store keeper Inventory management for Cafeteria.

1. **Cashier Finance Management**
   * Purpose: Enable cashier to confirm payments made by customers in the cafeteria.
   * Target Users: Cashier.
   * Key Features and Functionality:

Transactions – Enables Cashier to track transactions made in the cafeteria.

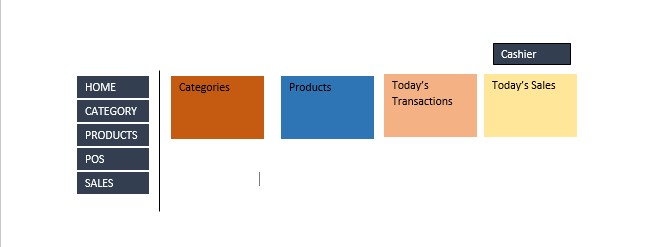


Figure 7: Finance manager for Cafeteria

## Hardware Interfaces

### Hardware Interfaces for University ERP System - cafeteria

The UNIVERSITY ERP System - cafeteria will interface with a variety of hardware components, including:

* Storage devices
* Network infrastructure devices
* Printers
* Scanners
* Mobile devices
* Desktop computers

### Logical Characteristics

The logical characteristics of the hardware interfaces will vary depending on the specific device type. However, all of the hardware interfaces will be designed to be efficient and reliable. The software will be able to communicate with the hardware devices in a timely manner and without errors.

### Physical Characteristics

The physical characteristics of the hardware interfaces will also vary depending on the specific device type. However, all of the hardware interfaces will be standard and well-documented. This will make it easy to integrate the system with existing hardware infrastructure.

### Supported Device Types

The UNIVERSITY ERP System – cafeteria will support a wide range of device types, including:

* SAN and NAS storage devices
* Ethernet switches and routers
* Network printers and MFPs
* USB and Bluetooth scanners
* Scanners
* Mobile devices running iOS, Android, and Windows

### Nature of Data and Control Interactions

The nature of the data and control interactions between the software and the hardware will vary depending on the specific device type. However, some common types of interactions include:

* Reading and writing data to storage devices
* Sending and receiving network packets
* Controlling the operation of printers and scanners
* Receiving input from barcode scanners and mobile devices

### Communication Protocols

The UNIVERSITY ERP System will use a variety of communication protocols to interface with the hardware devices. Some common protocols include:

* TCP/IP
* HTTP/HTTPS
* FTP
* SMTP
* SMB
* iSCSI
* Bluetooth

## Interfaces

### Connections Software to Other Software Components

The UNIVERSITY ERP System – Cafeteria Module will connect to the following software components:

* Database: MySQL 7.0
* Operating system: Windows Server 2016
* Programming language: Python 3.11 , Kotlin 1.7.20
* Web framework: Django 4.1
* ORM: SQLAlchemy 1.4
* Authentication and authorization library: Django REST Framework 3.13
* Emailing library: Django Celery 5.3
* PDF generation library: ReportLab 6.5
* Data visualization library: Plotly 6.10

### Data Items and Messages

The following data items and messages will come into and go out of the system:

Incoming:

* Student Purchases data
* Meals data
* Inventory data
* Service Person data
* Financial data

Outgoing:

* Student Orders
* Available Meals
* Financial statements
* Purchase orders
* Shipping labels

### Purpose of Each Data Item or Message

Each data item or message is used to support a specific business process within the university. For example, student purchases data is used to make students meal orders, Inventory data is used to create and manage Cafeteria Stocks, and financial data is used to manage the university's cafeteria budget and expenses.

### Services Needed

The UNIVERSITY ERP System will need the following services from the other software components:

* Database: The database will be used to store and manage the system's data.
* Operating system: The operating system will provide the platform on which the system runs.
* Programming language: The programming language will be used to develop the system's code.
* Web framework: The web framework will be used to create and deliver the system's user interfaces.
* Android application: will be built using Kotlin programming lamguage.
* ORM: The ORM will be used to map the system's data objects to the database schema.
* Authentication and authorization library: The authentication and authorization library will be used to manage user access to the system.
* Emailing library: The emailing library will be used to send notifications and reports to users.
* PDF generation library: The PDF generation library will be used to generate reports and other documents.
* Data visualization library: The data visualization library will be used to create charts and graphs for reports and dashboards.

### Nature of Communications

The communications between the UNIVERSITY ERP System and the other software components will be asynchronous. This means that the system will be able to send and receive data and messages without having to wait for a response. This will allow the system to be more scalable and efficient.

### Application Programming Interface Protocols

the UNIVERSITY ERP System will use the following application programming interface (API) protocols to communicate with the other software components:

* JDBC and FIREBASE for the database
* HTTP/HTTPS for the web framework
* SQLAlchemy Core's SQL expression language for the ORM
* Django REST Framework's JSON Web Token (JWT) authentication for the authentication and authorization library
* Django Celery's AMQP protocol for the emailing library
* ReportLab's PDF generation API for the PDF generation library
* Plotly's JavaScript API for the data visualization library

### Data Sharing

The UNIVERSITY ERP System will share data across software components using a variety of mechanisms, including:

* Database tables: The system will store its data in database tables. The tables will be designed to minimize data redundancy and maximize data integrity.
* API calls: The system will use API calls to request and receive data from other software components.
* Message queues: The system will use message queues to send and receive messages to and from other software components.

### Implementation Constraints

The following implementation constraints apply to the data sharing mechanisms:

* The database tables must be designed to be efficient and scalable.
* The API calls must be designed to be secure and reliable.
* The message queues must be designed to be able to handle high volumes of messages.

## Communications Interfaces

### communications interfaces for UNIVERSITY ERP System - cafeteria

The UNIVERSITY ERP System - cafeteria will use a variety of communications interfaces to communicate with users and other systems.

#### Email

The UNIVERSITY ERP System - cafeteria will use email to send notifications and promotions to users. The system will use the Simple Mail Transfer Protocol (SMTP) to send emails.

#### Web Browser and Android Mobile Application

The UNIVERSITY ERP System - cafeteria will be accessed by users through a web browser and android mobile phone. The system will use the Hypertext Transfer Protocol (HTTP) to communicate with web browsers.

#### Network Server Communications Protocols

The UNIVERSITY ERP System - cafeteria will use a variety of network server communications protocols, including:

* TCP/IP: The Transmission Control Protocol/Internet Protocol is the standard protocol for communication over the Internet.
* HTTP/HTTPS: The Hypertext Transfer Protocol is used to transfer web pages and other resources over the Internet. HTTPS is a secure version of HTTP that uses encryption to protect data in transit.
* FTP: The File Transfer Protocol is used to transfer files between computers over a network.

#### Electronic Forms

The UNIVERSITY ERP System - cafeteria will use electronic forms to collect data from users. The system will use a variety of formats for electronic forms, including HTML, PDF, and XML.

#### SMS

The UNIVERSITY ERP System - cafeteria will communicate short messages to the users to inform different users of their requests and replies to their orders.

### Message Formatting

The UNIVERSITY ERP System - cafeteria will use the following message formatting standards:

* JSON: JavaScript Object Notation is a lightweight data-interchange format.
* XML: Extensible Markup Language is a markup language used to represent data in a structured format.

### Communication Standards

The UNIVERSITY ERP System - cafeteria will use the following communication standards:

* HTTP/HTTPS: The Hypertext Transfer Protocol is the standard protocol for communication over the web. HTTPS is a secure version of HTTP that uses encryption to protect data in transit.
* FTP: The File Transfer Protocol is the standard protocol for transferring files between computers over a network.
* SMB: The Server Message Block protocol is the standard protocol for sharing files and resources between computers in a Windows domain.

### Communication Security and Encryption

The UNIVERSITY ERP System - cafeteria will use the following security and encryption mechanisms to protect data in transit:

* HTTPS: HTTPS uses encryption to protect data in transit.
* TLS: Transport Layer Security is a cryptographic protocol that provides secure communication over a computer network.
* SSL: Secure Sockets Layer is a cryptographic protocol that provides secure communication over a computer network. SSL is a predecessor to TLS.

### Data Transfer Rates

The UNIVERSITY ERP System - cafeteria will be designed to support high data transfer rates. The system will use a variety of techniques to improve data transfer rates, such as caching, compression, and load balancing.

### Synchronization Mechanisms

The UNIVERSITY ERP System - cafeteria will use a variety of synchronization mechanisms to ensure that data is consistent across all of the system's components. Some common synchronization mechanisms include:

* Database replication: Database replication replicates data across multiple database servers to ensure that the data is always available and consistent.
* Message queues: Message queues can be used to synchronize data between different components of the system.
* File synchronization: File synchronization can be used to synchronize data between different computers and devices.

# System Features: University Cafeteria

## System Feature 1: Menu Planning and Management

### Description

Description: This feature allows cafeteria staff to create and manage menus, including daily specials, dietary restrictions, and allergen information.

**Priority** High

### Stimulus and Response:

Staff selects "Menu Planning and Management" from the cafeteria management module.

System displays a calendar view of the upcoming menus.

Staff can create new menus, edit existing menus, and add items to menus.

System validates menu items to ensure that they are in stock and meet dietary restrictions.

### Functional Requirements:

* REQ 001- The system shall allow staff to create and manage menus for multiple cafeterias and stations.
* REQ 002- . The system shall allow staff to import and export menus in a variety of formats  REQ 003- The system shall generate reports on menu item popularity and sales.

### Response:

Success: The menu has been successfully created or updated.

Failure: The menu could not be created or updated due to an error.

## System Feature 2: Ordering and Payment

### Description:

This feature allows customers to order and pay for food at the cafeteria

Priority: High

.

### Stimulus and Response:

. Customer selects items from the menu and places an order.

System displays the order total and allows the customer to pay using a variety of payment methods. System prints a receipt and provides the customer with their order

### Functional Requirements:

* REQ 001- The system shall support multiple payment methods, including cash, credit/debit cards, and meal plans.
* REQ 002- The system shall allow customers to customize their orders and add special instructions.
* REQ 003 -The system shall generate reports on order volume and sales

### Response:

Success: Your order has been placed successfully.

Failure: Your order could not be placed due to an error.

## System Features 3: Inventory Management

### Description:

This feature allows cafeteria staff to track inventory levels and ensure that food is always available. Priority: High assistance."

### Stimulus and Response

Staff selects "Inventory Management" from the cafeteria management module.

System displays a list of all inventory items, including current quantities and reorder points.

Staff can add, edit, and delete inventory items. System generates reports on inventory levels and usage

### Functional Requirements:

* 33REQ 001-.The system shall support multiple inventory locations, including kitchens, storage rooms, and serving lines.
* REQ 002-.The system shall generate alerts when inventory levels are low or when items are about to expire.
* REQ 003-The system shall generate reports on inventory costs and usage.

### Response:

Success: The inventory item has been successfully added, edited, or deleted.

Failure: The inventory item could not be added, edited, or deleted due to an error.

## System Features 4: Food Safety Management

### Description:

This feature helps cafeteria staff to ensure that food is handled and prepared safely. Priority: High actions.

### Stimulus/Response Sequences:

* Staff selects "Food Safety Management" from the cafeteria management module.
* System displays a list of all food safety tasks, such as temperature checks and cleaning schedules.
* Staff can mark tasks as complete.
* System generates reports on food safety compliance.

### Functional Requirements:

.

* REQ-001The system shall notify students when they receive new messages or have new files shared with them. The system shall track the temperature of food at critical control points.
* REQ-002The system shall generate alerts when food temperatures are outside of the safe zone.
* REQ-003The system shall track employee food safety training and certifications.  REQ-004The system shall generate reports on food safety incidents and corrective

### Response:

Success: The food safety task has been marked as complete.

Failure: The food safety task could not be marked as complete due to an error.

## System Features 5: Customer Feedback and Reporting

### Description

This feature allows cafeteria staff to collect and analyze customer feedback. Priority: Medium

### Stimulus/Response Sequences:

Customer selects "Feedback" from the cafeteria website or app.

System displays a survey where the customer can rate their experience and provide feedback.

Customer submits the survey.

System generates reports on customer feedback and trends

### Functional Requirements

REQ-001The system shall allow customers to provide feedback on food quality, service, and cleanliness.

REQ-002The system shall generate reports on customer feedback by category and location.

REQ-003The system shall allow staff to respond to customer feedback and track resolutions.

### Response:

Success: Your feedback has been submitted successfully.

Failure: Your feedback could not be submitted due to an error.

# Other Nonfunctional Requirements

## Performance Requirements

1. **Response Time:**

The system should provide a responsive user interface with an average response time of less than 2 seconds for standard operations, even during peak usage.

1. **Scalability:**

The system should be able to handle a minimum of 100 simultaneous users without significant performance degradation.

It should be scalable to accommodate increased usage as the café grows, with provisions for load balancing.

1. **Database Performance:**

Database queries for common operations, such as ordering of food stuffs and payments, should execute within 1 second for datasets containing up to 10,000 records.

Database indexing and query optimization should be regularly maintained to ensure efficient performance.

Payment transactions within the database should have a worst case scenario of 300 seconds in which the transaction is cancelled in case of a roll back.

1. **Batch Processing:**

Background batch processes, such as financial reporting and order processing, should complete within specified timeframes to minimize disruption to daily operations.

1. **Data Retrieval Efficiency:**

Data retrieval for reporting and analytics should be optimized to provide results for complex queries within 5 seconds.

1. **Network Latency:**

Network latency between the system and external services, such as payment gateways, should not exceed 1000 milliseconds.

1. **Concurrent User Support:**

The system should support a minimum of 50 concurrent users during peak usage hours without a significant impact on performance.

## Safety Requirements

1. **Data Security:**

The system should implement strong encryption protocols (e.g., HTTPS) to ensure the security of data transmitted over the network.

User authentication and authorization mechanisms should be in place to protect sensitive information.

1. **Backup and Recovery:**

Regular automated backups of all system data should be performed, with data retention policies defined.

Data recovery procedures should be documented and tested to ensure rapid restoration in case of data loss or system failure.

1. **Access Controls:**

Role-based access controls should be enforced to restrict access to sensitive data and functionalities based on user roles and permissions.

1. **Data Privacy Compliance:**

The system should comply with data privacy regulations such as GDPR (General Data Protection Regulation) to protect the privacy of customers and employee data.

1. **Disaster Recovery:**

A disaster recovery plan should be in place, outlining procedures for system recovery in the event of a catastrophic failure or natural disaster.

1. **Logging and Auditing:**

Comprehensive logging and auditing mechanisms should be implemented to track system activities and user actions for security and compliance purposes.

1. **Firewall and Intrusion Detection:**

A firewall should be in place to protect the system from unauthorized access, and intrusion detection systems should monitor for potential security breaches.

1. **User Training and Awareness:**

User training programs and documentation should be provided to ensure that cafe staff are aware of security best practices and potential risks.

1. **Emergency Shutdown:**

In the event of a security breach or other critical incidents, the system should have the capability to initiate an emergency shutdown to prevent further harm or data loss.

## Security Requirements

1. **Access Control:**

Define role-based access control (RBAC) to restrict access to specific modules and data based on user roles (e.g., customers, staff, managers).

Implement a mechanism for access revocation and suspension when necessary.

1. **Authentication and Authorization:**

Require user authentication through strong password policies or multi-factor authentication (MFA).

Define authorization rules to specify what actions each user role can perform within the system.

1. **Data Encryption:**

Ensure that data transmission over the network is encrypted using industry-standard protocols (e.g., SSL/TLS).

Encrypt sensitive data at rest, including personal and financial information.

1. **Data Privacy:**

Comply with data protection regulations such as GDPR (General Data Protection Regulation).

Define data retention policies and mechanisms for data anonymization when required.

1. **Audit Trail:**

Implement an audit trail system to record and monitor all user actions and system events for auditing purposes.

Ensure logs are securely stored and regularly reviewed.

1. **User Identity Management:**

Define user registration and account recovery procedures.

Implement user identity validation and verification processes.

1. **External Policy Compliance:**

Adhere to external policies, such as PCI DSS (Payment Card Industry Data Security Standard) for financial transactions, as applicable.

1. **Security Certifications:**

Identify any industry-specific security or privacy certifications required (e.g., ISO 27001, SOC 2) and ensure compliance.

1. **Incident Response Plan:**

Develop an incident response plan to address security breaches, data leaks, or other security incidents promptly.

1. **Secure Software Development Lifecycle (SDLC):**

Integrate security into the entire software development lifecycle, including regular security testing, code reviews, and vulnerability assessments.

1. **Data Backups and Recovery:**

Implement regular data backups and define disaster recovery procedures to minimize data loss and downtime in case of system failures.

1. **User Training and Awareness:**

Provide training and resources to educate users about security best practices and how to safeguard their accounts and data.

1. **Security Updates and Patch Management:**

Establish a process for timely installation of security patches and updates to mitigate vulnerabilities.

1. **Secure API and Integration:**

Ensure secure API design and authentication for third-party integrations to prevent data breaches.

1. **Penetration Testing and Security Assessments:**

Conduct regular penetration testing and security assessments to identify and remediate vulnerabilities.

1. **Physical Security:**

Secure physical access to servers and data centers where the ERP system is hosted.

1. **Logging and Monitoring:**

Implement real-time monitoring of system activities for suspicious behavior and security incidents.

1. **User Privacy Preferences:**

Allow users to manage their privacy preferences, including opting in or out of certain data collection and communication.

1. **Legal and Regulatory Compliance:**

Stay up to date with changes in relevant laws and regulations and adapt the system to remain compliant.

1. **Data Ownership and Consent:**

Clearly define data ownership and consent mechanisms, especially when handling personal information.

**External Policies and Regulations:**

1. **GDPR (General Data Protection Regulation):**

The café should comply with GDPR when processing personal data of citizens. This includes obtaining explicit consent for data processing and ensuring the right to be forgotten.

1. **Payment Card Industry Data Security Standard (PCI DSS):**

If the ERP system handles credit card payments, adhere to PCI DSS standards to protect cardholder data and ensure secure payment processing.

1. **State and Local Privacy Laws:**

Comply with state and local privacy laws, which may include additional requirements for data protection and privacy.

**Security and Privacy Certifications:**

1. **ISO 27001:**

Achieving ISO 27001 certification demonstrates a commitment to information security management systems (ISMS) and ensures comprehensive security controls are in place.

1. **SOC 2 (Service Organization Control 2):**

SOC 2 certification attests to the security, availability, processing integrity, confidentiality, and privacy of data within the ERP system.

1. **CIPP (Certified Information Privacy Professional):**

This certification focuses on privacy regulations and best practices, ensuring that the ERP system complies with data privacy laws.

1. **FISMA (Federal Information Security Management Act):**

If the university collaborates with federal agencies, FISMA compliance may be required to meet federal security standards.

1. **FERPA Compliance Certification:**

Obtain certification or documentation to prove compliance with FERPA, demonstrating the protection of student records.

1. **HIPAA Compliance Certification:**

If healthcare data is involved, obtain HIPAA compliance certification to ensure the secure handling of health information.

1. **GDPR Compliance Certification:**

If applicable, obtain certification or documentation to confirm compliance with GDPR, especially if the university has European students.

1. **PCI DSS Compliance Certification:**

If handling payment card data, obtain PCI DSS compliance certification to demonstrate secure payment processing.

## Software Quality Attributes

**Adaptability:**

Requirement: The system should support the addition of new modules or functionalities within two weeks without affecting existing operations.

**Availability:**

Requirement: The ERP system should have a 99.99% uptime during peak usage hours, measured monthly.

**Correctness:**

Requirement: The software must achieve a defect rate of fewer than 0.1% in the final release version. **Flexibility:**

Requirement: The system should allow administrators to configure workflows and data fields without requiring custom coding, reducing configuration time by 50%.

**Interoperability:**

Requirement: The ERP software must integrate seamlessly using standard APIs.

**Maintainability:**

Requirement: Code changes and updates should take less than one hour to deploy without causing system downtime.

**Portability:**

Requirement: The system should be accessible on major web browsers (Chrome, Firefox, Safari, Edge) and mobile devices (iOS and Android) with full functionality.

**Reliability:**

Requirement: The system should not crash or freeze during normal operation, with a mean time between failures (MTBF) of at least 10,000 hours.

**Reusability:**

Requirement: Code modules should be designed with reusability in mind, with a goal of reusing 70% of code components in future system enhancements.

**Robustness:**

Requirement: The system should gracefully handle unexpected inputs and errors, with 99% of error scenarios leading to user-friendly error messages.

**Testability:**

Requirement: Test coverage should include 95% of all code paths, and automated test suites should cover all critical functionalities.

**Usability:**

Requirement: The system should achieve a System Usability Scale (SUS) score of 80 or higher in user testing.

In terms of relative preferences, it's important to clarify customer and developer priorities:

**Ease of Use vs. Ease of Learning:** Customers may prioritize ease of use, while developers may prioritize ease of learning. It's essential to strike a balance that makes the software accessible to both novice and experienced users.

**Performance vs. Resource Efficiency:** Customers often want high performance, while developers may focus on resource efficiency. Specify the desired performance levels and resource utilization targets.

**Functionality vs. Simplicity:** Balance the desire for feature-rich functionality with the need for simplicity and user-friendliness. Identify critical features that must not be sacrificed for simplicity.

**Scalability vs. Cost:** Customers may prioritize scalability, but developers must consider the associated infrastructure and maintenance costs. Specify scalability requirements based on expected growth.

**Security vs. Usability:** Security is paramount, but it shouldn't compromise usability. Define security requirements while ensuring a user-friendly experience.

**Interoperability vs. Independence:** Specify the degree of interoperability required with external systems and clarify any preference for independence or autonomy in certain functions.

## Business Rules

Business rules are essential for guiding the operation and behavior of a university ERP (Enterprise Resource Planning) software system - cafeteria. These rules dictate who can perform specific functions under various circumstances.

1. **User Authentication and Authorization:**

Only registered users with valid credentials can access the system.

Role-based access control (RBAC) determines which functions users can perform based on their roles (e.g., customers, staff , managers).

Customers can view their orders, but only administrators can modify them.

1. **Financial Transactions:**

Only authorized personnel (e.g cashier) can process financial transactions, such as order payments.

Payment processing adheres to PCI DSS standards for credit card transactions.

1. **Data Privacy and Security:**

Personal and academic data must be kept confidential and accessed only by authorized individuals as per FERPA regulations.

User data is stored securely, and encryption is used for data transmission over the network.

1. **Account Creation and Management:**

New user accounts are created by administrators after verifying the identity of customers, cashiers, store keepers and manager.

Users can reset their passwords through a self-service mechanism.

1. **Table Reservations:**

Customers can request table reservations e.g., for events, meetings etc., through the ERP system.

1. **Billing and Invoicing:**

Invoices are generated automatically for order payments and other charges and sent to customers' registered email addresses.

# Other requirements

## Database Requirements

1.Data Storage Capacity:

The database should have the capacity to store a minimum of 1 years’ worth of customers’ data, staff data, inventory records, and transaction history.

2.Backup and Recovery:

A backup schedule, i.e. daily at 2am, for the database to ensure data integrity, and define a robust recovery mechanism in case of data loss or system failure.

3.Database Performance:

The database system will be optimized for efficient data retrieval and updates, supporting the performance requirements as described in chapter 5.1 i.e. under database performance where database queries are used for common operations, Database indexing and query optimization is regularly maintained to ensure efficient performance and so on.

## Internationalization Requirements

1.Multilingual Support:

The system will be designed to support multiple languages i.e. English and Kiswahili to accommodate users from diverse linguistic backgrounds.

2.Currency Formats:

If applicable, the system will be capable of displaying and processing financial information in different currency formats.

## Legal Requirements

1.Compliance with Data Protection Laws:

The system will comply with the data protection act, 2019 in Kenya.

2.Transaction Record Keeping:

Implement features to adhere to legal requirements for record-keeping of transactions, ensuring compliance with tax regulations. The features to be implemented are complete and accuracy of transactions, reliability and authenticity, timely and accessible, secure and confidential record keeping as stated in records systems: characteristics and functions (2023).

## Reuse Objectives

1.Code Reusability:

Promote code reusability by documenting and organizing code modules in a way that facilitates easy integration into future projects or updates.

2.Third-Party Integration:

Design the system with a modular architecture that allows for seamless integration with thirdparty tools or APIs, promoting future scalability and adaptability.

## User Documentation

1.User Manuals:

Provide comprehensive user manuals and documentation to guide administrators, sales personnel, and other users on system functionality, troubleshooting, and best practices.

2.Training Materials:

Develop training materials, including video tutorials or guides, to assist users in learning how to effectively use the system.

## Performance Testing

1.Load Testing:

Conduct regular performance testing, including load testing, to ensure the system can handle peak usage without significant degradation in performance.

2.Scalability:

Design the system architecture to be scalable, allowing for easy expansion to accommodate growing data volumes and user loads.

# Appendix A: Glossary

**1.System:**

Refers to the university entity resource planning system developed for the university.

**2.User:**

Any individual interacting with the university entity resource planning system, including administrators, students, staff and any other personnel related to the university.

**3.Software:**

Refers to the application or program developed to manage the operations of the university.

**4.Administrative Privileges:**

The rights and permissions granted to administrative users for system management and configuration.

**5.University ERP:**

The acronym representing the university entity resource planning for which the software system is designed.

**6.Limitations:**

Constraints or restrictions on certain functionalities or features of the system.

**7.Interface:**

The point of interaction between the user and the software, including graphical user interfaces (GUIs).

**8.GUI (Graphical User Interface):**

The visual representation of the system that allows users to interact with various functionalities.

**9.SMTP (Simple Mail Transfer Protocol):**A protocol used for sending emails.

**10.HTTP (Hypertext Transfer Protocol):**

A protocol for transmitting hypertext requests and information between servers and browsers.

**11.HTTPS (Hypertext Transfer Protocol Secure):**

A secure version of HTTP that encrypts data for secure communication.

**12.Protocol:**

A set of rules defining the format and sequence of messages exchanged between systems.

**Appendix B: Analysis Models**

Menu

*Menu\_Id*

*Price*

*Start Date*

*End date*

*Foo*

*d\_Id*

Manager

*Manager\_Id*

*FName*

*LName*

*Username*

*Password*

*Status*

*Menu\_Id*

Food Item

*Food\_Id*

*Name*

*Quantity*

*Unit Price*

*ItemCategory*

Order Item

*Order\_Id*

*Food\_Id*

*Quantity*

*UnitPrice*

Order

*Order\_Id*

*OrderDate*

*CustomerID*

*Quantity*

*PickupDate*

Payment

*Payment\_Id*

*Customer\_Id*

*Order\_Id*

*PaymentDate*

*Amount*

*PaymentType*

Chef

*Chef\_Id*

*La*

*stName*

*FirstName*

*Username*

*PhoneNumbe*

*r*

*Password*

*Order\_ID*

Customer

*Customer\_Id*

*Email*

*Phone no*

*FName*

*LName*

*Payment\_Id*

*Food\_Id*

Reservation

*Reservation\_*

*Id*

*Customer\_Id*

*tableNumber*

*bookingDate*

*expectedDat*

*e*

*status*

Figure 8: UNIVERSITY ERP SYSTEM – CAFETERIA DATABASE DESIGN