

**MEKELLE UNIVERSITY**

**ETHIOPIA INSTITUE OF TECHNOLOGY MEKELLE-EITM SCHOOL OF COMPUTING**

**DEPARTEMENT: SOFTWARE ENGINEERING**

***COURSE TITLE: SOFTWARE ENGINEERING TOOLS AND PRACTICES COURSE* CODE: SENG5331**

**GROUP ASSIGNMENT TITLE: BANKING SYSTEM MODEL**

**Name ID**

1. FILLIMON G/TSADIK UGR/178838/12
2. HENOK MEKONNEN UGR/179052/12
3. BRHINA WUBET UGR/170168/12
4. MESELE SHISHAY UGR/178259/12
5. MEBRAHTOM G/HIWOT EITM/TUR181590

SUBMISSION TO INS:

**SUBMISSION DATE: 17-12-2024**

**BANKING SYSTEM MODEL**

**Introduction to the Banking System Model**

the **Banking System Model** is often designed using **UML diagrams** to visually represent its structure, processes, and interactions. This approach helps in analyzing, designing, and implementing a robust banking application.

Key UML diagrams used include:

1. **Use Case Diagram**: Highlights the interactions between actors (e.g., customers, bank staff) and the system, showing use cases like deposit, withdrawal, fund transfer, and loan application.
2. **Class Diagram**: Represents the system's data structure, showing entities such as

**Account**, **Customer**, **Transaction**, and their relationships.

1. **State Diagram**: Highlights the states an object can occupy, such as an account transitioning from active to suspended.
2. **Sequence Diagram**: Describes the interaction between system components, such as a customer initiating a transaction and the bank verifying and processing it.

**Example:**

* + **Use Case**: A customer logs in to check their balance.
    - **Actors**: Customer, System
    - **Use Case Flow**:
      1. Customer enters credentials.
      2. System validates input.
      3. System retrieves and displays balance.

UML diagrams are created using tools like **Lucidchart**, **Microsoft Visio**, **Enterprise Architect**, or open-source options like **StarUML** and **Draw.io**, enabling clear communication among stakeholders and aiding in effective system development.