Introduction:

System analysis is a problem-solving technique that decomposes a system into its component pieces for the purpose of the studying how well those component parts work and interact to accomplish their purpose. It primarily addresses an entire corporation and its software development operation. To analyze a system, an analyst must be able to explain the various stages of the system development life cycle, list the components of system analysis, and explain the components of system designing. They must also be able to engage with internal users and clients. We chose to examine Bank Asia Ltd, Noapara Branch.

UML Diagram:

UML, short for Unified Modeling Language, is a standardized graphical language used for visualizing, specifying, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. It provides a set of notations and symbols to represent various elements and relationships in a system. A UML diagram is a visual representation of different aspects of a system using UML notation. UML diagrams help software developers, designers, and stakeholders to understand, communicate, and analyze the structure and behavior of a system in a clear and concise manner. There are several types of UML diagrams, each serving a specific purpose:

- Class Diagram
- Use Case Diagram
- Sequence Diagram
- Activity Diagram
- Component Diagram

Relationships in UML:

In Unified Modeling Language (UML), four types of relationships are used to depict the associations and interactions between different elements within a system. They are:

1. **Dependency:** Signifies a relationship between two elements where changes in one element can affect the other.
2. **Generalization:** Depicts an "is-a" relationship between classes, indicating that one class is a specialized version of another class.
3. **Association:** Represents a structural relationship between two or more classes or objects.

4. **Realization:** Represents the relationship between an interface (or abstract class) and the class that implements (or realizes) it.

Use Case Diagram:

Use case diagrams are a fundamental tool in the field of software development and systems analysis. They serve as a graphical representation of the functional requirements of a system from the user's perspective. Use case diagrams provide a clear and concise way to capture and communicate the interactions between users and the system, showcasing the specific functionalities offered by the system and the potential scenarios in which they are utilized. The primary purpose of a use case diagram is to outline the functionalities a system will perform and the actors (users, other systems, or external entities) that interact with the system. By using standardized symbols and notations, use case diagrams become a universal language that enables effective communication between different stakeholders involved in the development process. The main components of use case diagram are as follows:

- **Actor:** Actor represents the external entity that interacts with the system. An actor can be a human user, another system, or an external hardware device.

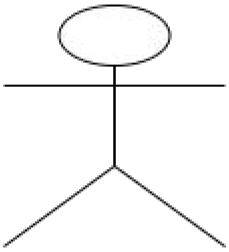

Fig-01: Actor

- **Use Cases:** Use cases represent the specific functionalities or services that the system provides to its actors. Each use case represents a distinct action that the system can perform.

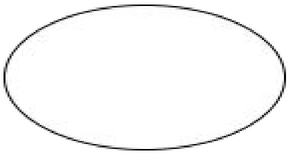

Fig-02: Use Case

- **Relationships:** The relationships between actors and use cases are represented by connecting lines between them.

Fig-03: Relationships

Overall UML Diagram of our system:

The use case diagram for several departments of Bank Asia Limited, Noapara Branch is represented below for better understanding the overall system.

Account opening Section:

Customer initiates account opening process by submitting an account application. The bank conducts verification to know customer identity and compliance. Required documents are verified for authenticity and accuracy. If verification is successful, the bank opens the account and assigns an account number.

Loan management:

Administrator initiates the loan application process by receiving Loan Requests from customers. The system verifies customer information and checks their credit score. If eligible, the loan request is sent to the head of the branch for approval. In case of huge loan, the application is also sent to a directory board for approval.

Cash withdrawals and deposits (through cheques):

Customer initiates transaction by visiting the bank branch. The customer provides account no. and identification to the cash officer. The cash officer verifies the provided documents. If verified, the customer provides the withdrawal amount. The cash officer checks the account balance if there is enough amount to withdraw. Then the account balance is updated accordingly after cash withdrawals or deposits.

ATM withdrawals:

Customer interacts with the ATM system by inserting ATM card. The system authenticates card verification. If the account holds enough balance for withdrawals, the customer can withdraw and his account balance is updated accordingly. If there's not enough balance, the request is rejected.

Conclusion:

In this report, we have presented the use case diagram of Bank Asia Ltd, Noapara branch. The use of UML (Unified Modeling Language) is a powerful and effective way to model, visualize, and document complex systems, whether they are software-based or non-software-based. UML diagrams provide a standardized and easily understandable representation of various elements, their interactions and relationships within a system, making it easier for stakeholders to communicate and collaborate throughout the development or analysis process. By utilizing UML diagrams such as use case diagrams, class diagrams, sequence diagrams, and others, software developers, designers, and stakeholders can gain a comprehensive understanding of the system's architecture, behavior and requirements. These visual representations help in identifying potential issues, clarifying design decisions and streamlining the development process, ultimately leading to the creation of more reliable, efficient and user-friendly systems.