**Document on Business Requirements**

**Project Title:**  Intelligent Automatic Voice Self-Service System in Call Center (Smart IVR)

**Project Objective:** Creating Use Case Diagram, Class Diagram, Activity Diagram, and Sequence Diagram based on the topic "Intelligent Automatic Voice Self-Service in Call Center (Smart IVR)" using the StarUML application.

**Project Description:** The Intelligent Automated Voice Self-Service Call Center (Smart IVR) is an innovative solution that enhances customer service and support while simplifying self-service for customers. The system encompasses several key features, including customer identification based on mobile phone number, customer query prediction, and dynamic main menu. The system automatically identifies customers through their mobile phone numbers, simplifying the identification process and routing calls to the appropriate department. It also predicts the purpose of the customer's call, providing them with relevant information such as product arrears, deactivated cards, upcoming payment dates, and recent incoming payments. The dynamic main menu of the intelligent IVR system adapts to the customer's banking products, presenting only the options that align with their specific products. This streamlines navigation and enables customers to quickly access the information they require. The primary goal of implementing the intelligent IVR system is to reduce the number of incoming calls to the call center by simplifying self-service and improving customer service and support. The system provides automated customer recognition, query prediction, and rapid access to information, creating a "wow" effect and increasing customer satisfaction.

**What does the project include?**

Key features of the automatic voice system:

Customer identification based on mobile phone number: The new Smart IVR will identify customers based on the mobile phone number they are calling from if that number is registered in our system. In other words, if you call from your registered mobile phone, the system will automatically identify you.

Expected outcomes for the customer: Once the system identifies the caller as a bank customer, it attempts to identify specific call center data: A. Checks for any product arrears. If available, the call will be automatically transferred to the respective department. B. Checks for inactive cards. If any are found, the customer will be prompted to retrieve them. C. Checks the next card payment dates. If the payment date falls within the next 10 days and there is an outstanding balance, the customer will be offered information on the statement and the minimum payment amount. D. Checks the next payment dates for open credits. If the payment date is within the next 10 days, the customer will be offered a review of the payment amount and date. E. Checks the recent payments made to the customer's current accounts. If funds have been received recently, the system will ask the customer for the date and amount of the payment.

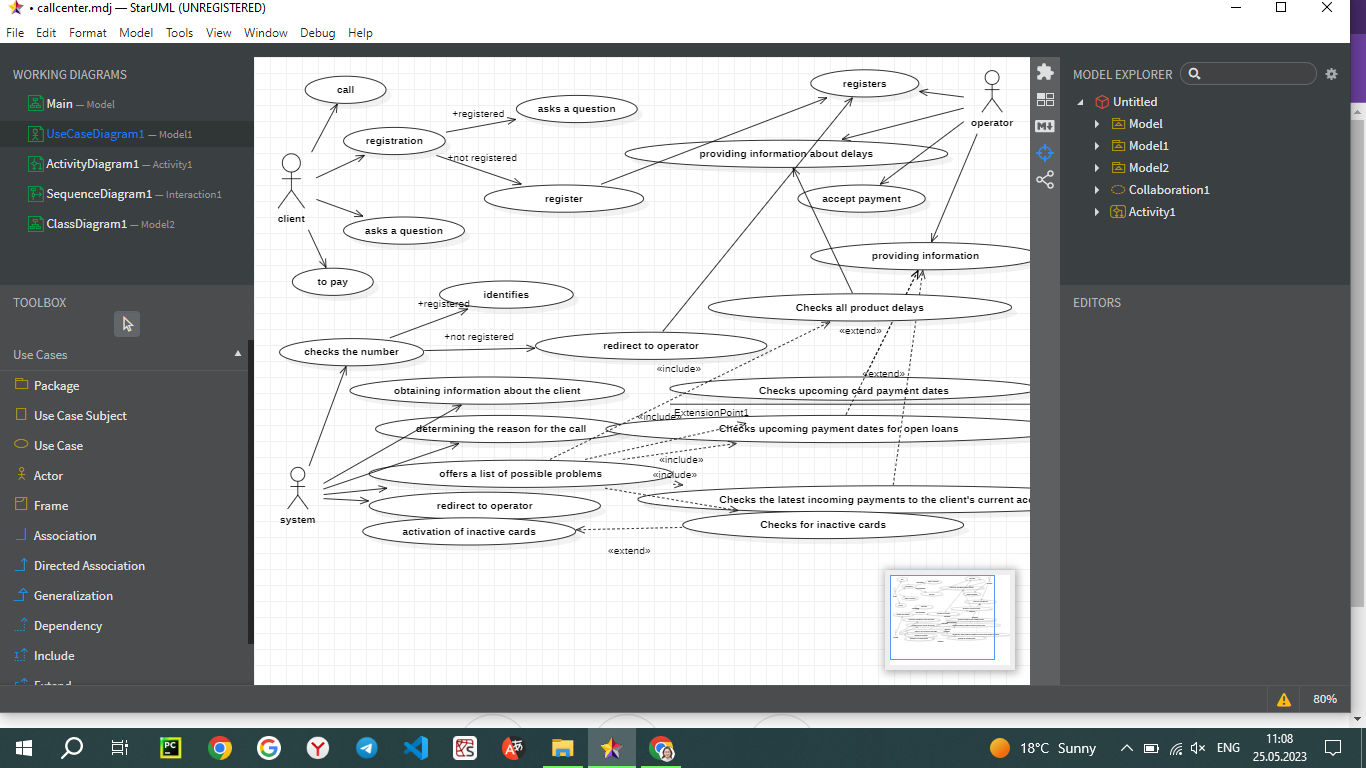
Dynamic main menu: Knowing all the customer's active products, the Smart IVR system will play a menu generator that only showcases banking product-specific options. For example, if loan accounts are not available, financial information related to loans will not be presented.

**1.Use Case Diagram:** It is a Unified Modeling Language (UML) diagram used to model the interactions and behavior of a system, application, or business process among different users or system components. A Use Case Diagram typically consists of the following components:

Actors: Users, other systems, or components that interact with the system or application.

Use Case Scenarios: Actions or scenarios representing the functional capabilities provided by the system.

Relationships: Links that show how actors interact with the use cases.



**2.Class Diagram** is a UML diagram that illustrates the structural organization of a system or software and how its components are interconnected. The Class Diagram showcases classes, objects, interfaces, relationships, and connections between these components within the system. A Class Diagram consists of the following components:

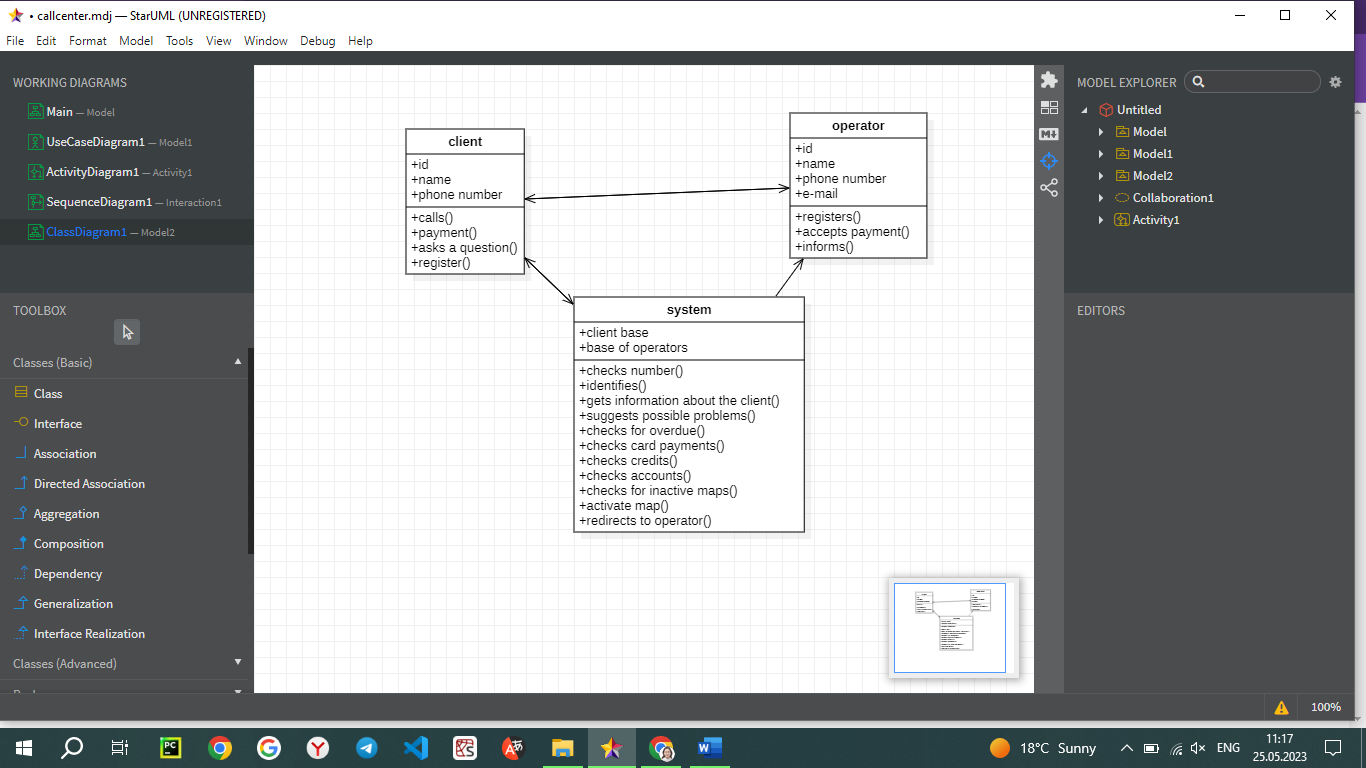
Classes: Structural units representing objects in the system.

Properties: Data elements or attributes of classes.

Methods: Functions or behaviors of classes.

Relationships: Links that demonstrate the associations between classes, such as inheritance, association, or dependency.

Please rephrase and rewrite the following text in English:



**3.Activity Diagram** is a UML diagram that visually represents the steps, decision points, parallel actions, and data flow of a process or workflow. It is a modeling tool used for understanding, analyzing, and communicating business processes and workflows.

The Activity Diagram includes the following components:

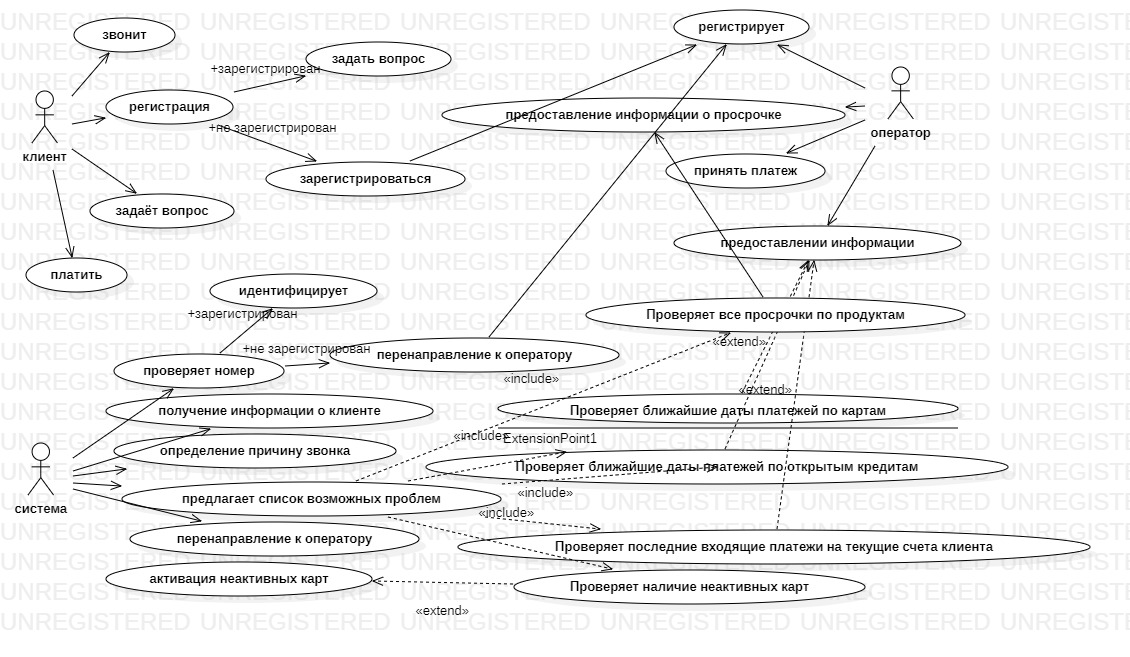
Actions: Nodes representing the steps of the process, actions, or activities.

Decision Points: Points that lead to different paths of the flow based on specific conditions or decisions.

Fork and Join Nodes: Points representing concurrent actions and indicating parallel flow.

States: State nodes representing a specific state of the process or system.

Flows: Arrows showing the transitions between actions, decision points, and states.



**4.Sequence Diagram** is a UML diagram that illustrates the interaction and message exchange between objects in a system in chronological order. It showcases how objects interact with each other and the sequence in which they exchange messages during a scenario or operation.

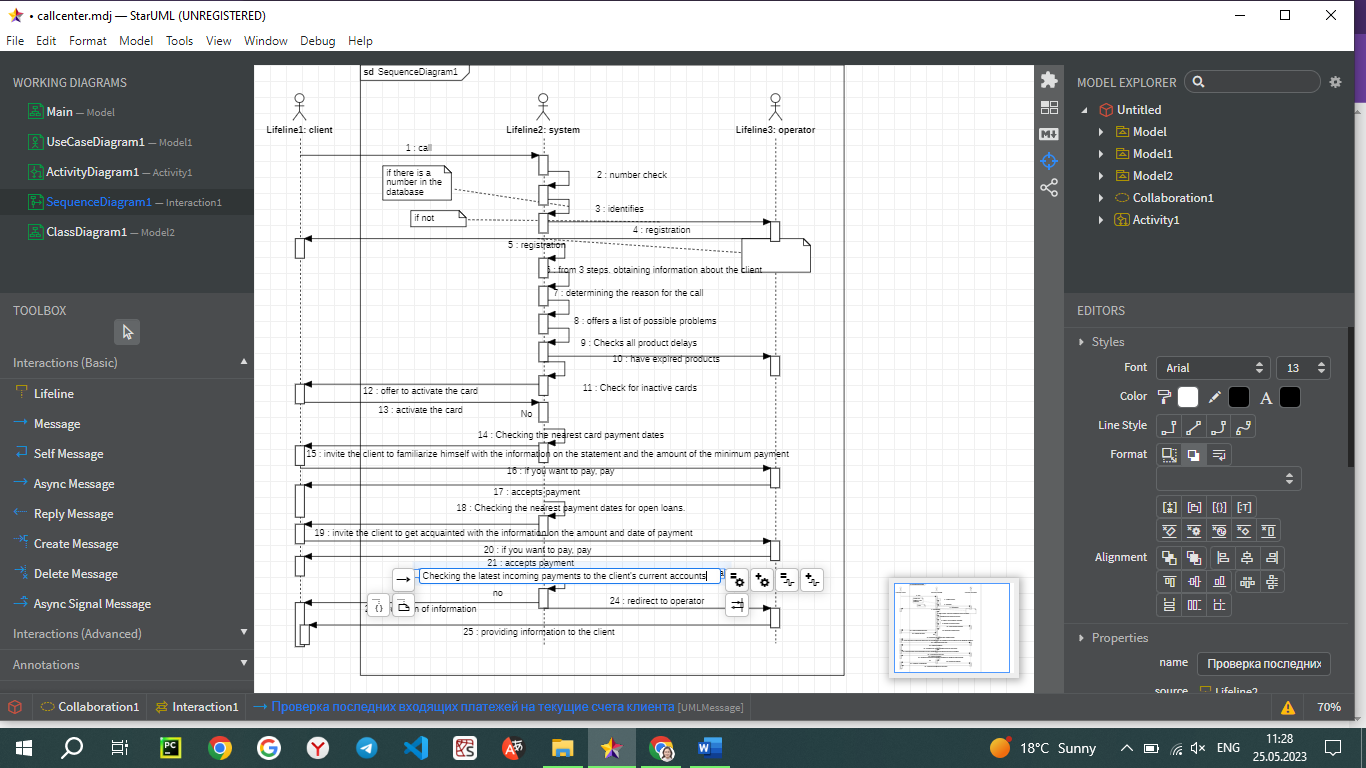
A Sequence Diagram consists of the following components:

Objects: Nodes representing objects in the system or software.

Lifelines: Horizontal lines that depict the lifespan of objects.

Messages: Arrows representing the communication between objects. Messages can be of different types, such as method calls, returns, or concurrent messages.

Activations: Nodes indicating the time when an object receives a message and responds to it.



**Conclusion:** In conclusion, in this project, we successfully modeled an intelligent automatic voice self-service system for a call center. We learned how to create Use Case Diagrams, Activity Diagrams, Class Diagrams, and Sequence Diagrams. Through these diagrams, we gained a deep understanding of the system's functionality and the step-by-step process flow. Additionally, we acquired the ability to establish connections between objects and comprehended their impact on the overall model.

Throughout the project, we gained valuable experience in modeling and developing voice self-service systems. This experience will enable us to create more efficient and innovative solutions in the realm of customer service. We witnessed the advantages that an intelligent automatic voice self-service call center, such as Smart IVR, can offer, including increased customer satisfaction and reduced workload on the call center.

We believe that the system we developed, Smart IVR, can be a valuable asset for any call center, and its implementation will bring significant benefits. We take pride in our work and are confident that our efforts will lead to improved customer service quality and optimized workflow processes.