Low-Level Design Document

Team Details:
Name : Siddharth Gupta (justcallmesidd@gmail.com)
Project Name: Customer Satisfaction Prediction using Python and Tkinter

Table of Contents

1. **Introduction**

1.1. Purpose

This low-level design document provides a detailed explanation of the implementation of the machine learning model and the Tkinter-based graphical user interface for the project aiming to predict customer churn in a bank.

1.2. Scope

This document covers the technical aspects of implementing the machine learning model, data preprocessing, training, and evaluation. Additionally, it includes the design and implementation of the Tkinter-based GUI, error handling mechanisms, testing procedures, and deployment considerations.

2. **Machine Learning Model Implementation**

2.1. Data Preprocessing

- Data loading and cleaning.
- Handling missing values.
- Feature encoding for categorical variables.
- Data scaling and normalisation.

2.2. Model Architecture
- Define the architecture of the Artificial Neural Network (ANN).
- Specify the number of layers, neurons, and activation functions.
- Choose loss functions and optimization algorithms.

2.3. Training and Evaluation
- Split the dataset into training and testing sets.
- Train the model on the training data.
- Implement evaluation metrics (e.g., accuracy, precision, recall, F1-score).
- Save the trained model for integration with the GUI.

3. **Tkinter-Based GUI Implementation**

3.1. GUI Layout
- Design the graphical layout of the application.
- Create input fields and buttons for user interaction.
- Organise widgets in frames and windows.

3.2. User Input Handling
- Implement functions to capture and validate user inputs.
- Pass user inputs to the machine learning model for prediction.
- Display prediction results on the GUI.

3.3. Model Integration

- Load the pre-trained machine learning model.
- Integrate the model into the GUI application.
- Implement the functionality to trigger predictions based on user inputs.

4. **Error Handling**

4.1. Input Validation
- Define rules and validation mechanisms for user inputs.
- Display error messages for invalid inputs.

4.2. Model Prediction Errors
- Handle errors that may occur during model prediction.
- Provide clear error messages to users.

5. **Testing and Validation**

5.1. Unit Testing
- Develop unit tests for individual components (e.g., data preprocessing, model training).
- Ensure correctness and reliability of each component.

5.2. Integration Testing
- Test the interaction between the GUI and the machine learning model.
- Verify that the application functions as expected end-to-end.

6. **Deployment**

6.1. Deployment Environment
- Specify the target deployment environment (e.g., local machine, cloud server).
- Ensure compatibility with the chosen environment.

6.2. Installation and Configuration
- Provide instructions for installing and configuring the application.
- Document any dependencies or runtime requirements.

7. **Conclusion**

This low-level design document outlines the technical implementation details of the machine learning model and Tkinter-based GUI for the customer churn prediction project. By following the guidelines and specifications provided in this document, the development team can efficiently implement and test the components necessary for the successful completion of the project.