iNeuron

Low Level Design

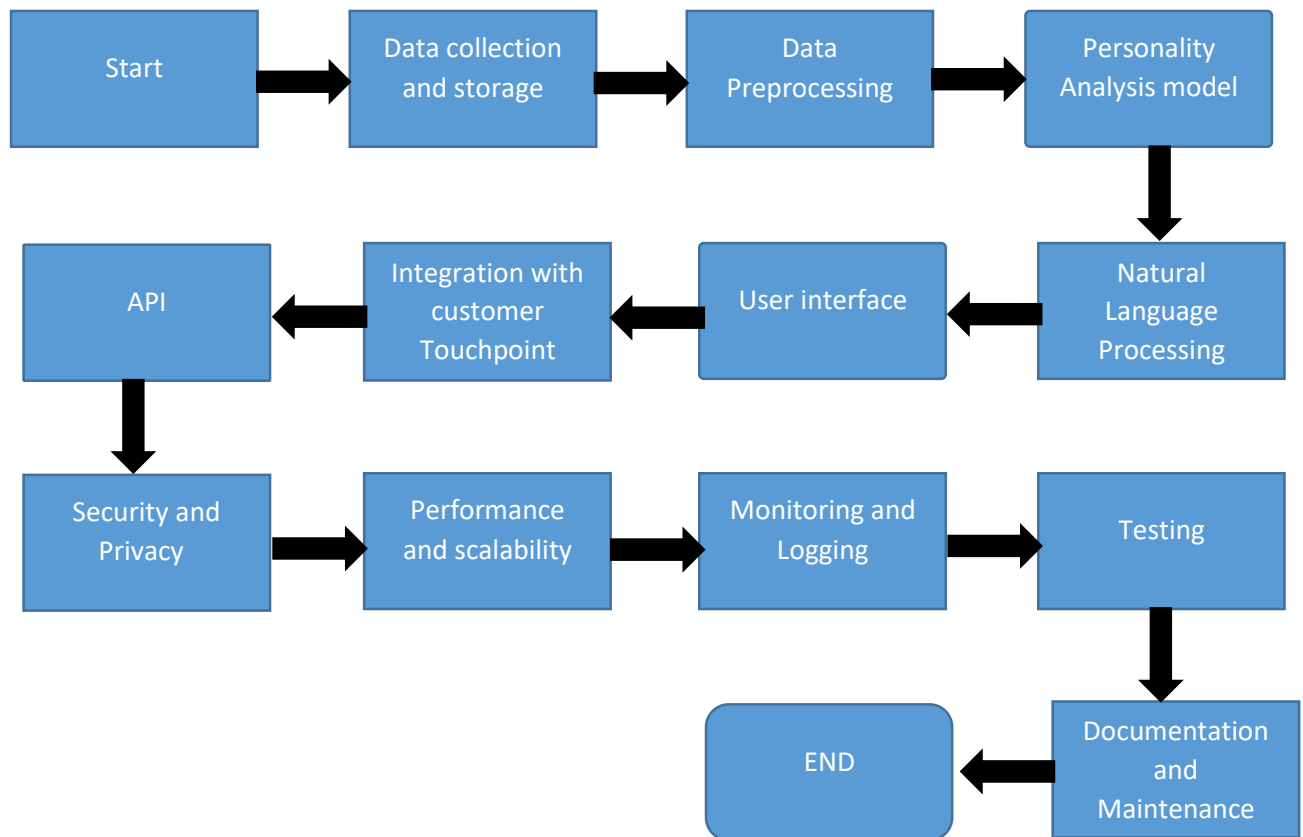# Customer Personality analysis (CPA)

Content

1. INTRODUCTION

1.1 What is Low-Level design document

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Food Recommendation System. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document.

1.2. Scope

Low-level design (LLD) is a component-level design process that follows a step-by-Step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work

2. Architecture

```
Start → Data collection and storage → Data Preprocessing → Personality Analysis model
                                                                      ↓
API ← Integration with customer Touchpoint ← User interface ← Natural Language Processing
 ↓
Security and Privacy → Performance and scalability → Monitoring and Logging → Testing
                                                                                ↓
                           END ← Documentation and Maintenance
```

3. Architecture Description

3.1 Data Collection and Storage
- Define the data sources from which customer data will be collected (e.g. social media).
- Choose appropriate data storage technologies (e.g. NoSQL database) to store customer data securely.

3.2 Data Preprocessing
- Develop data preprocessing modules to clean and transform raw customer data.
- Handle missing values, outliers, and data normalization.
- Perform feature engineering to extract relevant attributes that can contribute to personality analysis.

3.3 Personality Analysis Model
- Select a suitable personality analysis model or algorithm (e.g., machine learning models like SVM, Random Forest).
- Train the model on labeled data to predict customer personalities.
- Implement techniques for model evaluation and fine-tuning.

3.4 Natural Language Processing (NLP)
- If working with text data, employ NLP techniques to extract textual information and sentiment analysis.
- Use NLP libraries or pre-trained models for tasks like named entity recognition and part-of-speech tagging.

3.5 User Interface (UI)
- Create a user-friendly interface for users to input data and view the results.
- Implement visualizations and graphs to represent personality insights effectively.

3.6 Integration with Customer Touchpoints
- Integrate the personality analysis system with customer-facing touch points like websites, applications, or chatboats to collect data in real-time.

- Ensure privacy and data protection compliance while gathering customer data.

3.7 API and Micro services

- Create APIs to allow other systems to interact with the personality analysis service.
- Consider deploying the system as microservices for better scalability and maintainability.

3.8 Security and Privacy

- Implement robust security measures to protect customer data from unauthorized access.
- Comply with privacy regulations like GDPR, CCPA, etc., and ensure customer consent for data usage.

3.9 Performance and Scalability

- Optimize algorithms and code for performance and efficiency.
- Design the system to scale horizontally and handle increasing amounts of data and users.

3.10 Monitoring and Logging

- Implement monitoring and logging mechanisms to track system performance and diagnose issues.
- Set up alerts for critical events and errors.

3.11 Testing

- Develop comprehensive test cases to validate the system's functionality.
- Perform unit testing, integration testing, and performance testing.

3.12 Documentation and Maintenance

- Maintain detailed documentation of the system architecture, APIs, and data flow.
- Plan for regular maintenance and updates to keep the system up-to-date.