# **TelcoGenie 1.0 System Design**

### **Title Page**

**Project Name:** TelcoGenie 1.0  
 **Version Number:** 1.0  
 **Document Version:** 1.0  
 **Authors:** Abhishek Mulay  
 **Date of Creation/Last Update:** 4-Dec-2024

### 

### **Table of Contents**

1. Introduction  
    1.1 [Business Domain](#_tllpe4l4gib)  
    1.2 [Overview](#_vjt694vt9aqi)  
    1.3 [Scope](#_mhpe2moxaz5b)  
    1.4 [Objectives](#_teigxrq7sfbz)
2. System Overview  
    2.1 [System Context](#_hh6a93azadzb)  
    2.2 [Key Features](#_mq1s97tl0bzf)  
    2.3 [Assumptions and Constraints](#_65nnem4tl2l9)
3. Architecture Design  
    3.1 [Architectural Overview](#_slj0jqc0mbdj)  
    3.2 [System Components](#_wgv12rwgjhpz)
4. Detailed Design  
    4.1 [Module Details](#_dz91cok6f66)

### 

### **1. Introduction**

#### **1.1 Business Domain**

The telecom industry is highly competitive, with companies striving to enhance customer experience while reducing operational costs. Generative AI technology presents an opportunity to revolutionize customer support by providing quick, accurate, and personalized responses through conversational agents.

#### **1.2 Overview**

TelcoGenie 1.0 is a Generative AI chatbot designed specifically for telecom companies. It serves as a conversational agent for mobile subscribers, offering key functionalities such as balance inquiry, outstanding bill details, plan information, and plan recommendations. TelcoGenie aims to enhance customer engagement, provide instant support, and recommend suitable plans to potential customers.

#### **1.3 Scope**

* **Boundaries:**
  + The chatbot caters only to mobile customers (prepaid and postpaid).
  + Handles predefined features: balance inquiry, billing details, plan details, and recommendations.
  + Redirects users to human assistants for unsupported queries.
  + Uses CSV files as the database for version 1.0.
* **Target Users:** End customers of telecom companies with mobile subscriptions.

#### **1.4 Objectives**

* Improve customer service efficiency using AI-driven conversational agents.
* Provide instant access to essential mobile service information.
* Enhance user experience by offering tailored plan recommendations.
* Minimize reliance on human customer service representatives for basic queries.

### 

### **2. System Overview**

#### **2.1 System Context**

TelcoGenie operates as an AI-driven layer between customers and telecom service providers, reducing the need for human interaction in handling basic queries. It integrates with existing customer data and plan details via CSV files and uses OpenAI's APIs for natural language processing and function calling.

#### **2.2 Key Features**

* **Balance Enquiry:** Provides prepaid customers with their current balance.
* **Outstanding Bill Details:** Offers postpaid customers their billing information.
* **Pricing Plan Details:** Displays available plans for both prepaid and postpaid users.
* **Plan Recommendations:** Suggests suitable plans based on user preferences and requirements.

#### **2.3 Assumptions and Constraints**

* **Assumptions:**
  + CSV files contain accurate and up-to-date customer and plan data.
  + OpenAI APIs remain accessible and reliable.
* **Constraints:**
  + System limited to version 1.0 features.
  + Uses CSV files instead of a full-fledged database.
  + Moderation checks ensure no offensive or inappropriate interactions.

### 

### **3. Architecture Design**

#### **3.1 Architectural Overview**

A layered architecture is used to separate different responsibilities:

1. **User Interface Layer:** Handles interactions between the user and the bot.
2. **Application Logic Layer:** Manages dialogue and intent recognition.
3. **Data Access Layer:** Reads and processes data from CSV files.
4. **Integration Layer:** Uses OpenAI APIs for natural language understanding and function execution.

#### **3.2 System Components**

* **User Input Handler:** Captures user queries.
* **Moderation Module:** Ensures input and output adhere to content guidelines.
* **Intent Recognition Engine:** Identifies user intent to route the query to the appropriate function.
* **Function Module:** Executes defined functions based on user intent.
* **Response Generator:** Formats and delivers responses to the user.

### 

### **4. Detailed Design**

#### **4.1 Module Details**

1. **Module Name:** init\_conversation  
   * **Responsibilities:** Initiates and establishes the conversation with the user.
   * **Inputs:** User input string.
   * **Outputs:** Welcome message.
   * **Dependencies:** None.
2. **Module Name:** get\_customer\_details  
   * **Responsibilities:** Fetches customer details from the subscriber data.
   * **Inputs:** User-provided mobile number or identifier.
   * **Outputs:** Customer data or indication of potential customer.
   * **Dependencies:** Subscriber data CSV.
3. **Module Name:** get\_balance\_info  
   * **Responsibilities:** Retrieves the balance for prepaid customers.
   * **Inputs:** Customer identifier.
   * **Outputs:** Balance amount.
   * **Dependencies:** Subscriber data CSV.
4. **Module Name:** get\_billing\_info  
   * **Responsibilities:** Fetches outstanding bill details for postpaid customers.
   * **Inputs:** Customer identifier.
   * **Outputs:** Billing amount and due date.
   * **Dependencies:** Subscriber data CSV.
5. **Module Name:** get\_plan\_recommendations  
   * **Responsibilities:** Recommends suitable plans based on user input.
   * **Inputs:** User preferences or usage data.
   * **Outputs:** List of recommended plans.
   * **Dependencies:** Plan data CSV.
6. **Module Name:** moderation\_check  
   * **Responsibilities:** Filters out offensive or inappropriate content from user input or system response.
   * **Inputs:** User input string or response string.
   * **Outputs:** Moderated input/output.
   * **Dependencies:** OpenAI moderation API.
7. **Module Name:** dialogue\_management  
   * **Responsibilities:** Manages user interaction flow, maps intent to function calls, and generates appropriate responses.
   * **Inputs:** User queries, processed intent.
   * **Outputs:** Responses to user queries.
   * **Dependencies:** User Input Handler, Intent Recognition Engine, Function Module, Response Generator.

**Dialogue Management System Diagram:**

