Automated Reporting System with Python GUI

User & Setup Guide

Version: 1.0

Date: 2023-10-27

# 1. Overview

This document describes an automated system for generating interactive, multi-sheet Excel reports from an Oracle database. The system is designed to be both powerful for developers and user-friendly for operators, featuring a custom Graphical User Interface (GUI) for setting report parameters.  
  
Key Features:  
- User-Friendly GUI: A simple desktop application allows users to define date ranges and report parameters without editing configuration files or using the command line.  
- Optimized Database Queries: Connects directly to Oracle using Power Query, fetching only the specific data required for the given date range, minimizing database load and refresh times.  
- High-Performance Excel Engine: Utilizes a hidden Pivot Table as a high-speed data cache, allowing for instantaneous report interaction for the end-user.  
- Fully Automated Report Generation: Creates a separate, pre-filtered, and interactive dashboard sheet for each specified region or category.

# 2. System Architecture & Workflow

The system separates the user interface from the core automation logic for robustness and maintainability.  
  
A flowchart diagram (not displayed here) illustrates the sequence from GUI interaction, configuration setup, automation, database querying, and Excel reporting.

# 3. Prerequisites

Before setting up the project, ensure the following are installed:  
- Python 3.8+  
- Oracle Client for Microsoft Tools (bitness must match Excel)  
- Required Python Libraries: xlwings, configparser (install via `pip install xlwings configparser`)

# 4. Project Setup

Project folder structure:  
/Report\_Automation/  
|-- Report\_Template.xlsx  
|-- config.ini  
|-- gui\_launcher.py  
|-- main.py  
|-- /Generated\_Reports/

# 5. Component Configuration

Detailed steps are provided for:  
- Excel Template (setting up parameters, Power Query, pivot tables, and dashboard)  
- Configuration File (config.ini)  
- GUI Launcher (gui\_launcher.py)  
- Main Reporting Script (main.py)

# 6. How to Use the System

End-user steps:  
1. Launch gui\_launcher.py  
2. Enter date range and regions  
3. Click "Generate Report"  
4. The report is saved in /Generated\_Reports/

# 7. Troubleshooting

Common issues and resolutions:  
- "Oracle provider is not registered": Check Oracle client installation and bitness.  
- "Config file not found": Ensure config.ini is present.  
- "Error during report generation": Check console output from main.py.