High Level Design (HLD)

Analysing Expenditure

Date: 04-01-2025

Final Date of Revision: 29-12-2024

Document Version Control

Date Issued	Description	Author
04-01-2025	Created Initial HLD	Sohel Datta
28-12-2024	Revised HLD with Process Flow Chart	Sohel Datta
29-12-2024	Final Revision	Sohel Datta

Contents

Abstract

This document outlines the High-Level Design (HLD) for the project Analysing Expenditure. The goal is to analyse organizational costs and provide actionable insights to help management reduce costs and improve operational efficiency. The HLD focuses on the project's purpose, scope, problem statement, and proposed solutions with tools, design details, KPIs, and deployment plans.

1. Introduction

1.1. Why this High-Level Design Document?

This document provides a structured approach to understanding, designing, and implementing the expenditure analysis project. It ensures clarity, alignment, and effective collaboration among stakeholders.

1.2. Scope

The scope of this project includes:

- Analysing expenditure data from provided datasets.
- Identifying key cost contributors.
- Proposing optimization models to reduce costs.
- Implementing insights using tools like Excel and Python.

2. General Description

2.1 Project Perspective

The project is a data-driven analysis aimed at empowering the management team with insights to optimize expenditure and reduce costs, ensuring profitability and sustainability.

2.2 Problem Statement

No business can survive in this competitive market without managing its cost. High revenues mean little if costs are disproportionately higher, posing a significant red flag. The management has tasked us with creating and establishing new structures and models to reduce costs effectively.

2.3 Proposed Solution

The proposed solution involves:

- **Data Analysis:** Understanding expenditure patterns using data analytics tools.
- **Cost Segmentation:** Categorizing costs into fixed, variable, and operational costs.
- **Model Development:** Creating predictive models to forecast high-cost areas.
- **Recommendations:** Providing actionable insights to reduce expenditures.

2.4 Tools Used

- Microsoft Excel
- Python (Pandas, NumPy, Matplotlib, Seaborn)
- Jupyter Notebook
- Data Visualization Tools (e.g. Excel, Python (if Required))

2.5 Libraries Used

- **Pandas**: A powerful library for data manipulation and analysis, providing data structures like Data Frames for handling structured data efficiently.
- **NumPy**: Essential for numerical computations, NumPy supports large multidimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays.
- **Matplotlib**: A widely-used plotting library that allows for the creation of static, animated, and interactive visualizations in Python.
- **Seaborn**: Built on top of Matplotlib, Seaborn provides a high-level interface for drawing attractive statistical graphics, making it easier to create complex visualizations.
- **SciPy**: This library is used for scientific and technical computing, providing modules for optimization, integration, interpolation, eigenvalue problems, and other tasks.

- **Scikit-learn**: A machine learning library that provides simple and efficient tools for data mining and data analysis, including clustering and predictive modelling.

- **Statsmodels**: Useful for statistical modelling, this library allows users to explore data, estimate statistical models, and perform hypothesis tests.

- **Jupyter Notebook**: While not a library per se, Jupyter Notebook is an essential tool that provides an interactive computing environment where you can combine code execution with rich text elements like paragraphs, equations, figures, and links.

These libraries collectively support various aspects of the data analytics process from data cleaning and transformation to visualization and modelling.

3. Design Details

3.1 Functional Architecture

i. Data Collection:

- Import expenditure datasets (Excel, CSV).
- Clean and preprocess the data.

ii. Data Analysis:

- Identify trends, anomalies, and high-cost areas.
- Apply statistical techniques to segment costs.

iii. Model Building:

- Develop optimization models using Python.
- Use regression and clustering techniques to identify cost-saving opportunities.

iv. Reporting and Visualization:

- Create dashboards for easy understanding.
- Generate reports for stakeholders

3.2 Optimization

- **Cost Segmentation**: Split costs into categories for better analysis.
- **Outlier Detection**: Identify unusual cost patterns.
- **Predictive Modelling**: Use machine learning to forecast costs and identify areas for reduction.
- **Scenario Planning**: Simulate scenarios to evaluate the impact of cost-cutting measures.

4. Key Performance Indicators (KPIs)

- **Percentage reduction in overall expenditure.**
- **Number of cost anomalies identified.**
- **Accuracy of predictive models in forecasting costs.**
- **Percentage increase in operational efficiency.**
- **Stakeholder satisfaction with actionable insights.**

5. Deployment

1. **Development Environment:**
   - Tools: Python (Jupyter Notebook), Excel.
   - Dataset Integration: Load and preprocess datasets.
2. **Testing:**
   - Validate models on sample datasets.
   - Ensure output accuracy and consistency.
3. **Production:**
   - Deploy models and dashboards for management use.
   - Provide training for stakeholders to interpret and use the insights.

6. Conclusion

Analysing expenditure is critical for businesses to remain competitive and profitable. This project aims to leverage data analytics to uncover cost-saving opportunities and establish a structured approach to cost management. The insights derived will support informed decision-making and foster a culture of continuous improvement.

7. References

- Provided datasets: "DA Assignment.xlsx," "salary data.xlsx," "zomato-schema.xlsx."
- Python libraries: Pandas, NumPy, Matplotlib, Seaborn.
- Analytics concepts and tools.