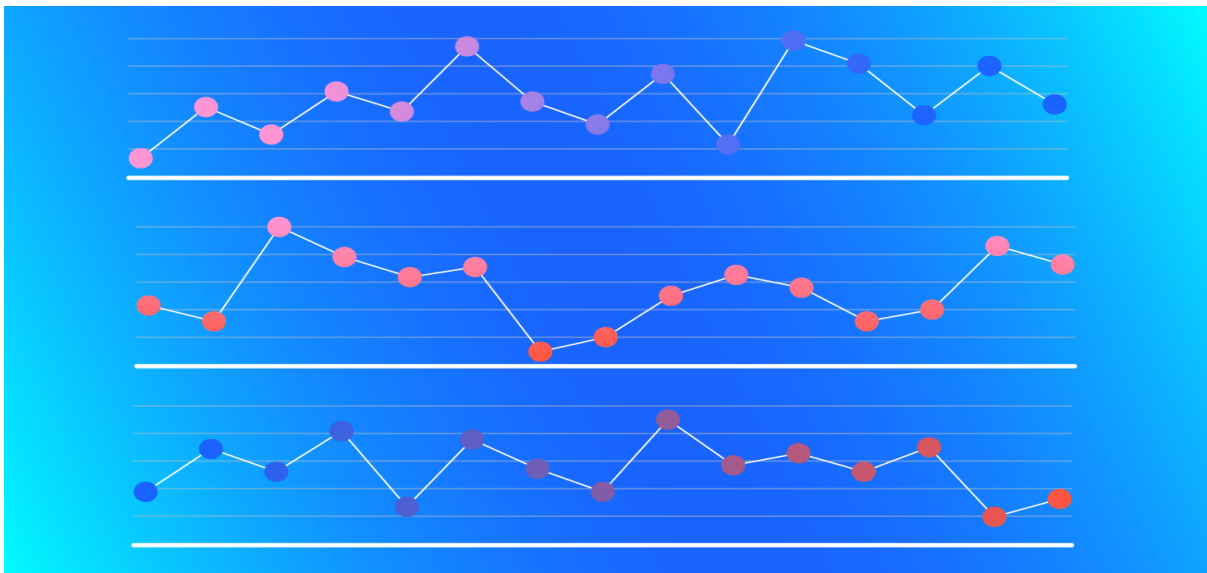


High Level Design (HLD)

BUDGET SALES ANALYSIS



ARPAN DATTA

Document Control

Date	Version	Description	Author
02/09/2022	1.0	Introduction, Problem Statement	Arpan Datta

Contents

Document Version Control	2
Abstract.....	4
1 Introduction	4
1.1 Why this High-Level Design Document?	4
1.2 Scope	5
2 General Description.....	5
2.1 Product Perspective & Problem Statement	5
2.2 Tools used	5
3 Design Details.....	6
3.1 Functional Architecture.....	6
3.2 Optimization.....	6
4 KPIs	8
4.1 KPIs (Key Performance Indicators)	8
5 Deployment.....	9

Abstract

Budget and Sales are by far most important attributes that defines a business's success and failure. Therefore, it is very important to keep a track on various features related to these attributes to keep on increasing the Sales and to allocate the Budget so that it can be utilized wisely and efficiently. There have been many examples where companies failed due to wrong allocation of budget and wrong marketing strategy. So, it is very important for businesses to dig deep into the customer, sales, budget and product data to make better marketing strategy, to know the target customers, to make market friendly product upgrades and to keep a strong track on the budget efficiency. Good data driven systems can help achieve these goals and take the businesses forward towards success.

1. Introduction

1.1 Why this High-Level Design Document?

=> The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- a. Present all of the design aspects and define them in detail
- b. Describe the user interface being implemented
- c. Describe the hardware and software interfaces
- d. Describe the performance requirements
- e. Include design features and the architecture of the project
- f. List and describe the non-functional attributes like:
 1. Security o Reliability o Maintainability
 2. Portability o Reusability
 3. Application compatibility
 4. Resource utilization
 5. Serviceability

1.2 Scope

=> The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

2 General Description

2.1 Product Perspective & Problem Statement

=> Our "Domain Sale" process is structured to help potential buyers purchase the domain they want immediately without the hassle of contacting the seller directly. A seller lists a domain for sale at a specific price in our Marketplace. An interested buyer sees this domain for sale and decides to buy it. Our main aim is to do the ETL and analyze, Visualize the relationship between Revenue, Sales, and Profit over various features like customers, time period, location, categories of items, and budget.

2.2 Tools used

=> Business Intelligence tools and libraries works such as Python Programming Language, NumPy, Pandas, Seaborn, Matplotlib, M S Excel, Jupyter Notebook, Power BI are used to build the whole framework.



3. Design Details

3.1 Functional Architecture



3.2 Optimization

How BI Really Works



- Minimize the number of fields
- Minimize the number of records
- Optimize extracts to speed up future queries by materializing calculations, removing columns and the use of accelerated views

Reduce the marks (data points) in your view

- i. Practice guided analytics. There's no need to fit everything you plan to show in a single view. Compile related views and connect them with action filters to travel from overview to highly-granular views at the speed of thought.
- ii. Remove unneeded dimensions from the detail shelf.
- iii. Explore. Try displaying your data in different types of views.

Limit your filters by number and type

- i. Reduce the number of filters in use. Excessive filters on a view will create a more complex query, which takes longer to return results. Double-check your filters and remove any that aren't necessary.
- ii. Use an include filter. Exclude filters load the entire domain of a dimension, while include filters do not. An include filter runs much faster than an exclude filter, especially for dimensions with many members.
- iii. Use a continuous date filter. Continuous date filters (relative and range-of-date filters) can take advantage of the indexing properties in your database and are faster than discrete date filters

4. KPIs

Dashboards will be implemented to display and indicate certain KPIs and relevant indicators for the disease.



As and when, the system starts to capture the historical/periodic data for a user, the dashboards will be included to display charts over time with progress on various indicators or factors

4.1 KPIs (Key Performance Indicators) Key indicators displaying a summary of the Housing Price and its relationship with different metrics

1. Influence of Category of the product on the Sales.
2. Influence of Sub-Category of the product category 'Bikes' on the Sales.
3. Influence of Country in which the product is been purchased on the Sales.
4. Influence of Color of the product on the Sales.
5. Influence of Yearly income of customers on the Sales.
6. Customer base distribution among different countries.
7. Change of total sales with respect to time.
8. Change of Budget 2016 with respect to months.

5. Deployment

=> Prioritizing data and analytics couldn't come at a better time. Your company, no matter what size, is already collecting data and most likely analyzing just a portion of it to solve business problems, gain competitive advantages, and drive enterprise transformation. With the explosive growth of enterprise data, database technologies, and the high demand for analytical skills, today's most effective IT organizations have shifted their focus to enabling self-service by deploying and operating Tableau at scale, as well as organizing, orchestrating, and unifying disparate sources of data for business users and experts alike to author and consume content.

