Low Level Design (LLD)

Google Analytics Customer Revenue Prediction

Soubhagya Nayak

**Contents**

[Document Version Control 2](#_gjdgxs)

[**Abstract** 4](#_30j0zll)

[1](#_1fob9te) Introduction 5

[1.1](#_3znysh7) Why this Low-Level Design Document? 5

[1.2](#_2et92p0) Scope 6

[1.3](#_tyjcwt) Constraints 6

[1.4](#_3dy6vkm) 1.3 Risks 6

[1.5](#_1t3h5sf) 1.4 Out of Scope 7

[2](#_4d34og8) Technical specifications 7

[2.1 Predicting Revenues 7](#_2s8eyo1)

[2.2 Logging 7](#_17dp8vu)

[2.3 Database 7](#_3rdcrjn)

[3](#_26in1rg) Technology stack 8

[4](#_lnxbz9) Proposed Solution 9

[5](#_35nkun2) Model training/validation workflow 10

[6](#_1ksv4uv) User I/O workflow 12

[7](#_44sinio) Exceptional scenarios 13

**Abstract**

Creating an automated system for predicting potential future business, finding potential customers based on the various parameters as decided by the machine learning algorithm. The purpose of the document is to explain the High architecture that would be used for developing the Google Store revenue prediction system. We’re challenged to analyze a Google Merchandise Store customer dataset to predict revenue per customer. Google provided Merchandise customer dataset and no of transactions per customer. We will build a predictive model using G-store data set to predict the total revenue per customer that helps in better use of marketing budget and we will also interpret the most impacting element on the total revenue prediction using different models.

# Introduction

## Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Google Analyst Customer Revenue System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the higher management for its approval.

The main objective of the project is to our goal is to predict the revenue that is going to be generated by those potential customers in the near feature. So that marketing teams will invest appropriate money on promotional strategies to attract potential customers.

This system is the vital part of marketing team:

* Contain a transaction history, Geographical location, transaction date, t plans, immunization dates, allergies, radiology images, and laboratory and test results
* Automate workflow

This project shall be delivered in two phases:

Phase 1: All the functionalities with PyPi packages.

Phase2: Integration of UI to all the functionalities.

## Scope

This software system will be a Web application This system will be designed to automate all the process and generate predicted revenues .

## Constraints

We will only be selecting a few of the transaction revenue.

## Risks

Document specific risks that have been identified or that should be considered.

## Out of Scope

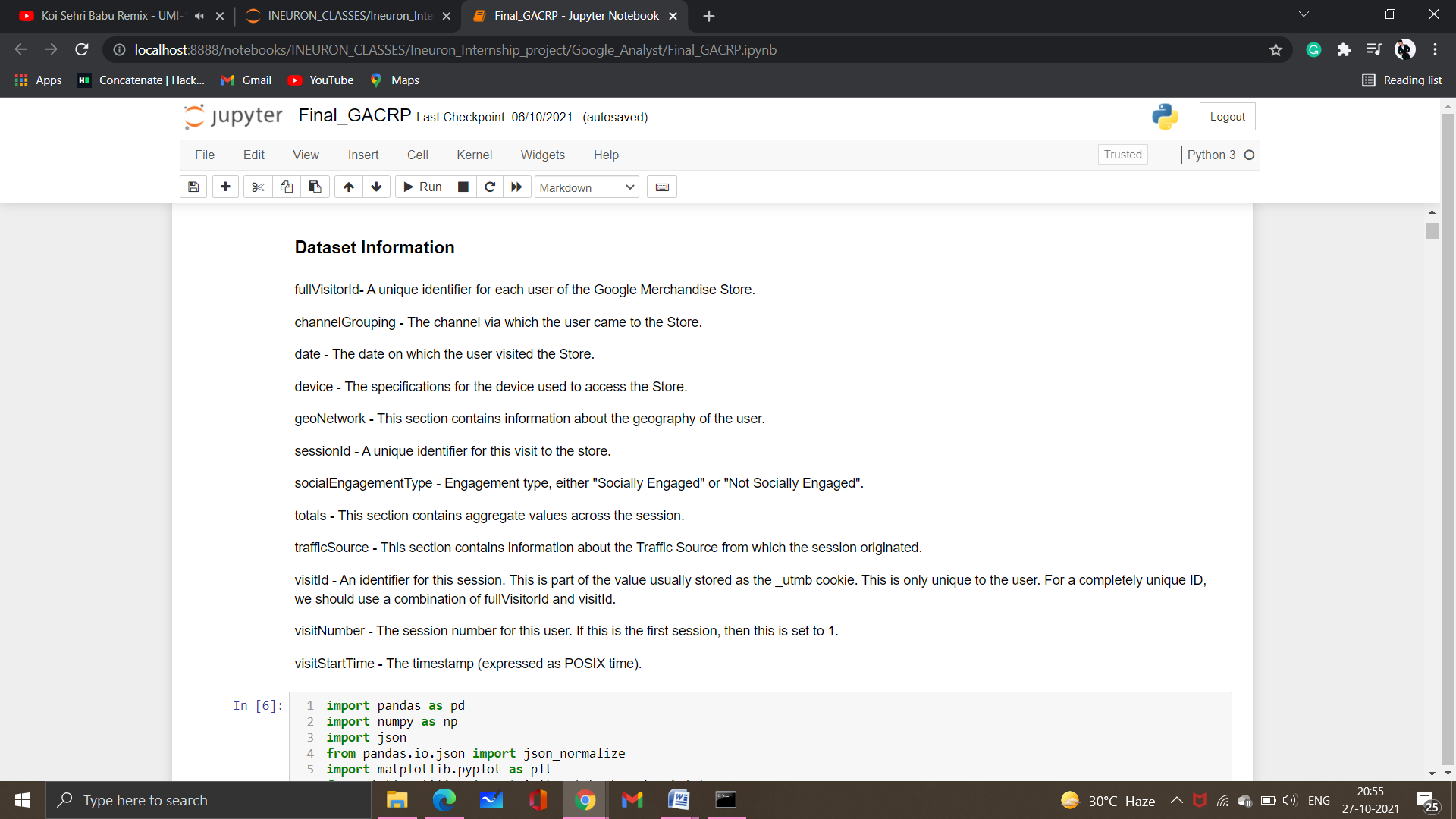
Delineate specific activities, capabilities, and items that are out of scope for the project.

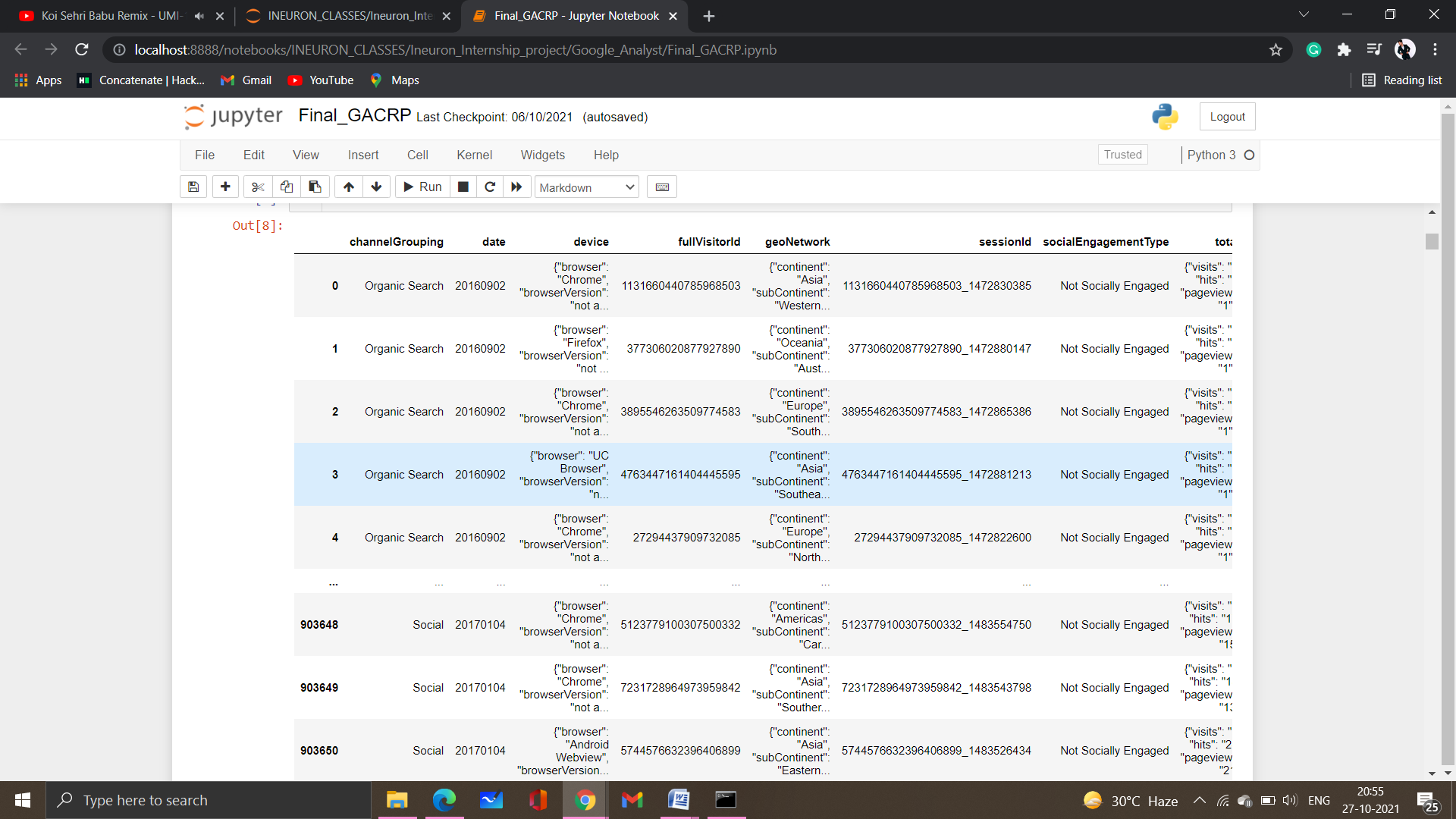
# Technical specifications

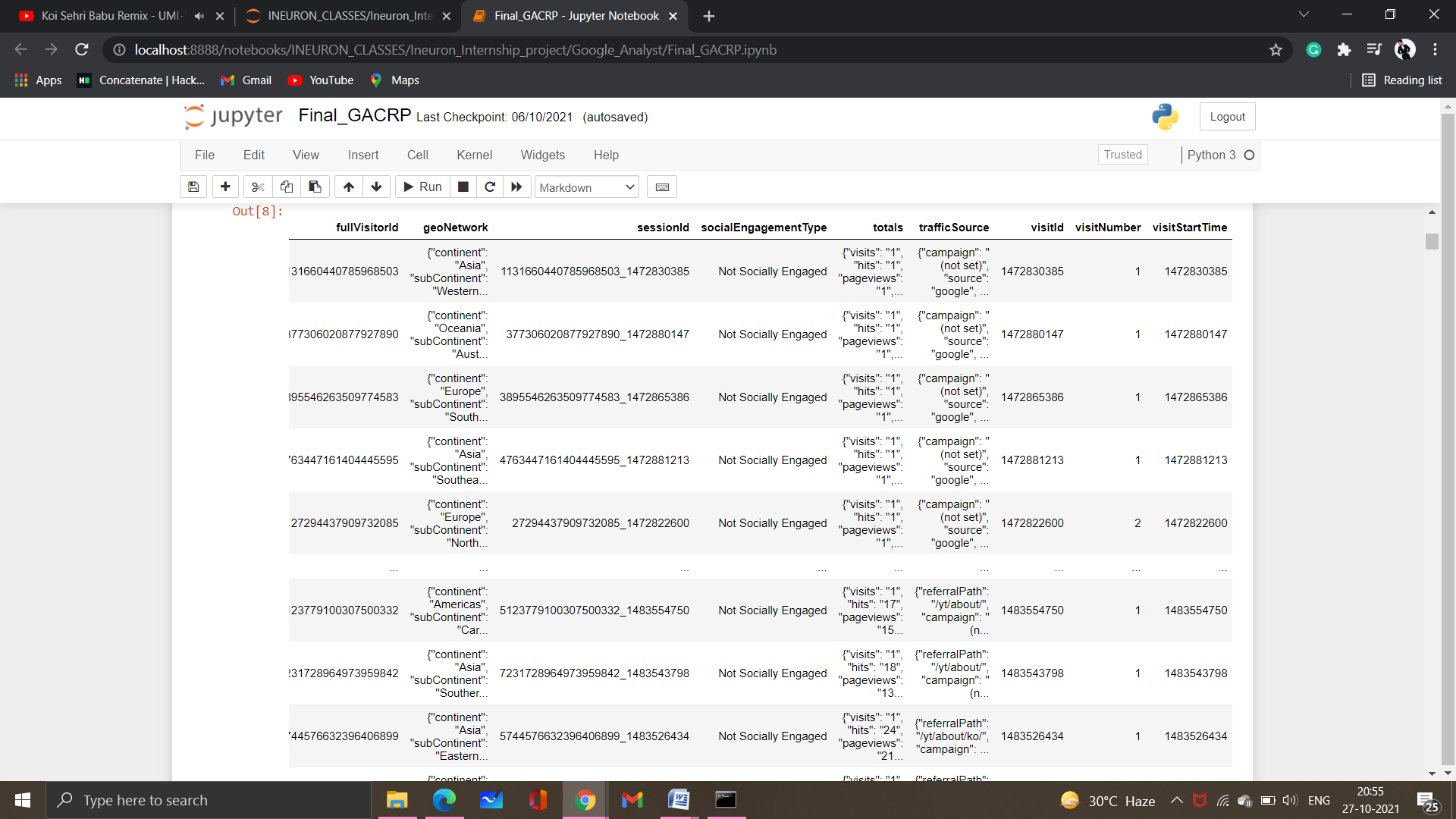
## 2.1 Dataset

Google provided Merchandise customer dataset and no.of transactions per customer

## 2.1.1 dataset overview







## 2.1.2 Input schema

## 2.2 Predicting Revenues

* The system will generate the revenue as predicted data.
* Through this revenue we will be able to find out potential customers.
* The system presents the set of file required from the user.
* The user gives required information and data to be predicted.
* The system should be able to predict which customer will be doing the no of transaction and generating the revenue.

## 2.3 Logging

We should be able to log every activity done by the user.

* The System identifies at what step logging required
* The System should be able to log each and every system flow.
* Developers can choose logging methods. You can choose database logging/ File logging as well.
* System should not be hung even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

## 2.4 Database

System needs to store every request into the database and we need to store it in such a way that it is easy to retrain the model as well.

The system stores each and every data given by the user or received on request to the database. Database you can choose your own choice whether MongoDB/ MySQL.

**2.5 Deployment**

1. Heroku



# Technology stack

|  |  |
| --- | --- |
| **Front End** | HTML/CSS/JS |
| **Backend** | Python Flask |
| **Database** | MongoDB/MySql |
| **Deployment** | Heroku |

# Proposed Solution

The solution proposed here is this prediction system based on some machine

learning algorithms (XGBoost ,LightGBM) can be implemented to perform predictive

analysis for finding the potential customers. After fine tune the model the algorithm

which have the lower Mean Square Error (MSE) will be selected for prediction . So we

have to predict the revenue that is going to be generated by those potential customers

in the near feature. So that marketing teams will invest appropriate money on

promotional strategies to attract potential customers.

# Model training/validation workflow



# Exceptional scenarios

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Exception | Mitigation | Module |
| 27th OCT 2021 | 1.1 | First Draft | Soubhagya Nayak |
| 28th OCT 2021 | 1.2 | Added Workflow chart | Soubhagya Nayak |

# Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Steps to perform test case | Module | Pass/Fail |
|  |  |  |  |

# Key performance indicators (KPI)

* Time and workload reduction using the GAnalyst model.
* Comparison of MSE of model prediction on revenue.
* Number of times customer visits the online store.
* Find potential customers for in the near feature