

Capstone Project - Requirements Document

Overview

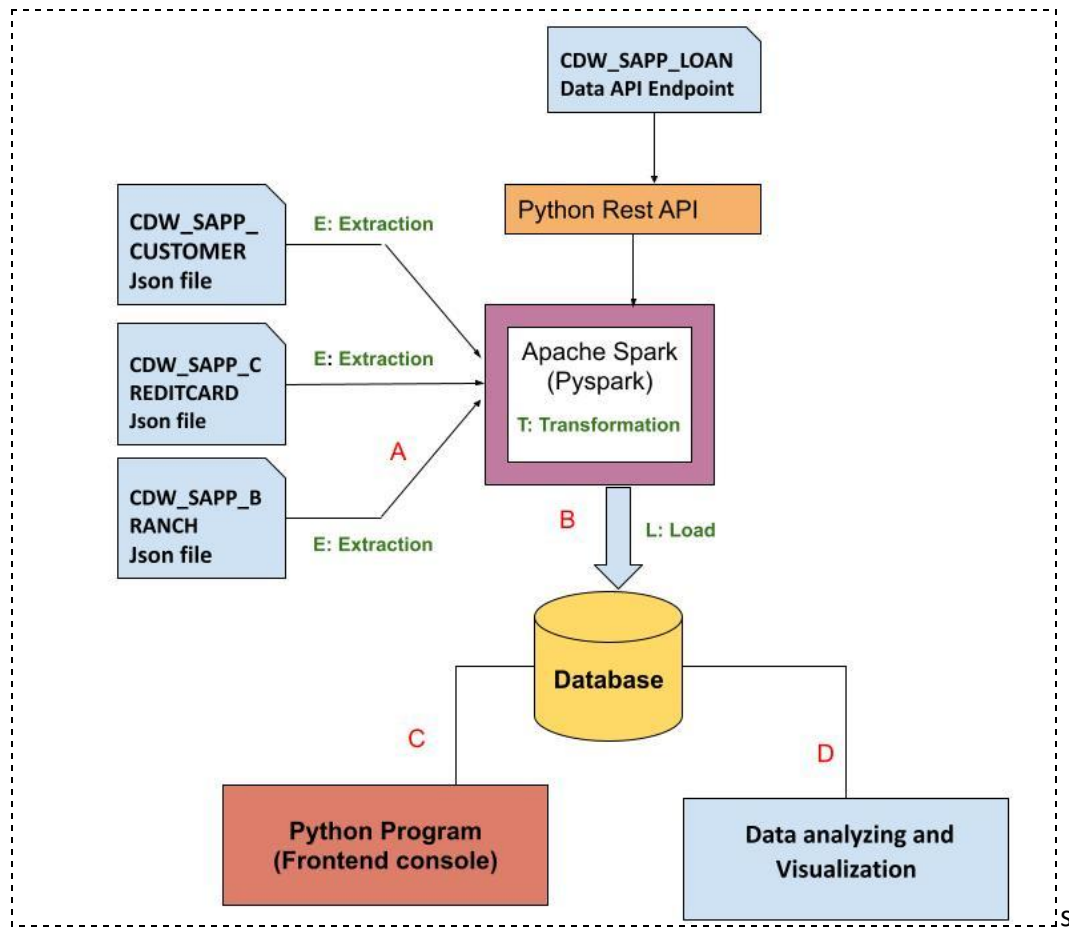
This Capstone Project requires learners to work with the following technologies to manage an ETL process for a **Loan Application dataset** and a **Credit Card dataset**: Python (Pandas, advanced modules e.g., Matplotlib), MariaDB, Apache Spark (Spark Core, Spark SQL), and Python Visualization and Analytics libraries. Learners will be expected to set up their environments and perform installations on their local machines.

Part One

Workflow Diagram of the Requirements

The workflow below will help you to understand the requirements and flow of the application.

Credit Card Dataset Overview



independent system developed for managing activities such as registering new customers and approving or canceling requests, etc., using the architecture.

A credit card is issued to users to enact the payment system. It allows the cardholder to access financial services in exchange for the holder's promise to pay for them later. Below are three files that contain the customer's transaction information and inventories in the credit card information.

- a) **CDW_SAPP_CUSTOMER.JSON:** This file has the existing customer details.
- b) **CDW_SAPP_CREDITCARD.JSON:** This file contains all credit card transaction information.
- c) **CDW_SAPP_BRANCH.JSON:** Each branch's information and details are recorded in this file.

[Click here to Download Credit Card system files](#)

Business Requirements - ETL

1. Functional Requirements - Load Credit Card Database (SQL)

Req-1.1	Data Extraction and Transformation with Python and PySpark
Functional Requirement 1.1	<p>a) For “Credit Card System,” create a Python and PySpark SQL program to read/extract the following JSON files according to the specifications found in the mapping document.</p> <ol style="list-style-type: none"> 1. CDW_SAPP_BRANCH.JSON 2. CDW_SAPP_CREDITCARD.JSON 3. CDW_SAPP_CUSTOMER.JSON <p>Note: Data Engineers will be required to transform the data based on the requirements found in the Mapping Document.</p> <p>Hint: [You can use PYSQL “select statement query” or simple Pyspark RDD].</p>
Req-1.2	Data loading into Database
Function Requirement 1.2	<p>Once PySpark reads data from JSON files, and then utilizes Python, PySpark, and Python modules to load data into RDBMS(SQL), perform the following:</p> <ol style="list-style-type: none"> a) Create a Database in SQL(MariaDB), named “creditcard_capstone.” b) Create a Python and Pyspark Program to load/write the “Credit Card System Data” into RDBMS(creditcard_capstone). <p>Tables should be created by the following names in RDBMS:</p> <p>CDW_SAPP_BRANCH</p> <p>CDW_SAPP_CREDIT_CARD</p> <p>CDW_SAPP_CUSTOMER</p>

2. Functional Requirements - Application Front-End

Once data is loaded into the database, we need a front-end (console) to see/display data. For that, create a **console-based Python program** to satisfy System Requirements 2 (2.1 and 2.2).

2.1 Transaction Details Module

Req-2.1	Transaction Details Module
Functional Requirements 2.1	<ol style="list-style-type: none">1) Used to display the transactions made by customers living in a given zip code for a given month and year. Order by day in descending order.2) Used to display the number and total values of transactions for a given type.3) Used to display the number and total values of transactions for branches in a given state.

2.2 Customer Details Module

Req-2.2	Customer Details
Functional Requirements 2.2	<ul style="list-style-type: none">1) Used to check the existing account details of a customer.2) Used to modify the existing account details of a customer.3) Used to generate a monthly bill for a credit card number for a given month and year.4) Used to display the transactions made by a customer between two dates. Order by year, month, and day in descending order.

3 - Functional Requirements - Data analysis and Visualization

After data is loaded into the database, users can make changes from the front end, and they can also view data from the front end. Now, the business analyst team wants to analyze and visualize the data according to the below requirements.

Use Python libraries for the below requirements:

Req - 3	Data Analysis and Visualization
Functional Requirements 3.1	<p>Find and plot transactions, showing which transaction type occurs most often.</p> <p>Note: Take a screenshot of the graphs.</p>
Functional Requirements 3.2	<p>Find and plot states, showing which state has the highest number of customers.</p> <p>Note: Take a screenshot of the graphs.</p>
Functional Requirements 3.3	<p>Find and plot the sum of all transactions for each customer, and which customer has the highest transaction amount. (First 20) hint(use CUST_SSN).</p> <p>Note: Take a screenshot of the graphs.</p>

Functional Requirements 3.4	Find and plot the top three months with the largest transaction data. Note: Take a screenshot of the graph.
Functional Requirements 3.5	Find and plot each branches healthcare transactions, showing which branch processed the highest total dollar value of healthcare transactions. Note: Take a screenshot of the graph.

Overview of LOAN application Data API

Banks deal in all home loans. They have a presence across all urban, semi-urban, and rural areas. Customers first apply for a home loan; after that, a company will validate the customer's eligibility for a loan.

Banks want to automate the loan eligibility process (in real-time) based on customer details provided while filling out the online application form. These details are Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History, and others. To automate this process, they have the task of identifying the customer segments to those who are eligible for loan amounts so that they can specifically target these customers. Here they have provided a partial dataset.

API Endpoint:

https://raw.githubusercontent.com/platformps/LoanDataset/main/loan_data.json

The above URL allows you to access information about loan application information. This dataset has all of the required fields for a loan application. You can access data from a REST API by sending an HTTP request and processing the response.

4. Functional Requirements - LOAN Application Dataset

Req-4	Access to Loan API Endpoint
Functional Requirements 4.1	Create a Python program to GET (consume) data from the above API endpoint for the loan application dataset.
Functional Requirements 4.2	Find the status code of the above API endpoint. Hint: status code could be 200, 400, 404, 401.
Functional Requirements 4.3	Once Python reads data from the API, utilize PySpark to load data into RDBMS(SQL). The table name should be CDW-SAPP_loan_application in the database. Note: Use the “ creditcard_capstone ” database.

5 - Functional Requirements - Data Analysis and Visualization for Loan Application

After the data is loaded into the database, the business analyst team wants to analyze and visualize the data according to the below requirements.

Use Python libraries for the below requirements:

Req-5	Data Analysis and Visualization
Functional Requirements 5.1	Create a bar chart that shows the difference in application approvals for Married Men vs Married Women based on income ranges. (number of approvals) Note: Take a screenshot of the graph.
Functional Requirements 5.2	Create and plot a chart that shows the difference in application approvals based on Property Area. Note: Take a screenshot of the graph.
CHALLENGE Requirement(Non-Graded)	Create a multi-bar plot that shows the total number of approved applications per each application demographic. Note: Take a screenshot of the graph.

Submission

- Upload your project to your private Github repository, which needs to have a minimum of one separate branch off of the main branch.
- Have a minimum of three commits (one line brief overview, plus a more descriptive paragraph underneath, if needed).
- The project needs to have a “readme” file documenting the project details, information, and technical challenges, along with how they were resolved.
- Ensure that any sensitive information (*secret.text* files etc.) has been .gitignore.
- Upload all Python codes, PySpark codes, database scripts, and databases are part of the repo.
- Take a screenshot of all the graphs.
- Submit the GitHub repository link during the Canvas assignment submission.
- All learners should present their projects individually. Your instructor will schedule the presentation slot for you.

References:

MariaDB Documentation

<https://mariadb.org/>

PySpark

<https://spark.apache.org/docs/latest/api/python/index.html>

Apache Spark - Spark SQL

<https://spark.apache.org/sql/>

Analyzing and Visualization

<https://www.analyticsvidhya.com/blog/2021/08/understanding-bar-plots-in-python-beginners-guide-to-data-visualization/>