**CAP 405: Data Analytics - Capstone Project Requirements Document**

## Objectives

The capstone project serves as a culmination of the knowledge and skills acquired throughout the course. It is an opportunity for learners to showcase their abilities in a practical setting.

By leveraging a combination of Python (utilizing Pandas and Numpy), SQL, Apache Spark (Spark Core and Spark SQL), and the Power BI Visualization tool, learners will showcase their ability to manipulate, analyze, and visualize data effectively. Through this project, learners will gain hands-on experience in setting up development environments, performing installations, and applying a range of technologies to derive actionable insights from real-world datasets. This capstone serves as a comprehensive assessment of the knowledge and skills acquired throughout the course, highlighting learners' capabilities in data management, analysis, and visualization within a practical context.

**Overview**

This capstone project is your opportunity to demonstrate the knowledge and abilities you have acquired throughout the course.

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, Numpy), SQL, Apache Spark (Spark Core, Spark SQL), and the Power BI Visualization tool. Learners are expected to set up their environments and perform installations on their local machines.

## Instructions for Python Environment

You can work in a notebook while developing your project, but you must export it to a **.py** file that works appropriately. We ***expect*** your front-end user interface program to be fully functional when you demonstrate your final project.

## Capstone Project Submission Guidelines

1. Submit a zipped copy of your entire capstone repository and all related files to the Canvas assignment submission.
   * Upload all Python codes (Jupyter notebook or VS notebook), PySpark codes, database scripts, and databases that are part of the project.
   * Save a copy of the visualization, making sure it is PROPERLY NAMED! and include the final (.pbix) file with all visual components confined in each section of this file.
   * Report:
     + *Option A - Google Doc:* 
       - Use the following report template to upload all of your capstone findings: [405 Capstone Report Template](https://docs.google.com/document/d/1vv-haYF1muwOsgO6dYnKse88EHwWAIVez0QY9QSPXjU/edit#heading=h.aczyuw2yex2w)
     + Option B - GitHub:
       - Upload your capstone project to your private GitHub repository, which needs to have a minimum of one separate branch off of the main branch.
         * Ensure that any sensitive information (*secret.text* files etc.) has been .gitignore.
         * Upload all Python codes, PySpark codes, database scripts, and databases that are part of the repo.
         * Create a readme file in your GitHub repository that will be utilized as the “Report” requirement.
         * Submit the GitHub repository link to the Canvas assignment submission.
2. Perform a technical walkthrough of your project, focusing on the following components (it should not take more than 30 minutes):
   1. Technical Requirements
      1. The user interface of your front-end program.
      2. Database along with clean data.
      3. Your visualizations (Power BI).
      4. All functionalities and features of your solution
   2. Highlight how your application works from the technical perspective (high level)
   3. Be prepared to see additional visitors (potential employer/s and Per Scholas leaders) during your technical walkthrough of the capstone project.
   4. Be prepared to have discussion and Q&A regarding your capstone project.

## 

## Capstone Primer Activities *(Rubric 3% out of 14%)*

The following files are a set of activities to help you prepare for each task of your capstone project.

[ACT 304 - The Data Detective Agency: Relational Databases and SQL](https://docs.google.com/document/d/1uyvHKmW4plGsyqA6u79UDdfmygrPONxEa6f1u2n40Kc/edit#heading=h.eu15ixxpdcoh)

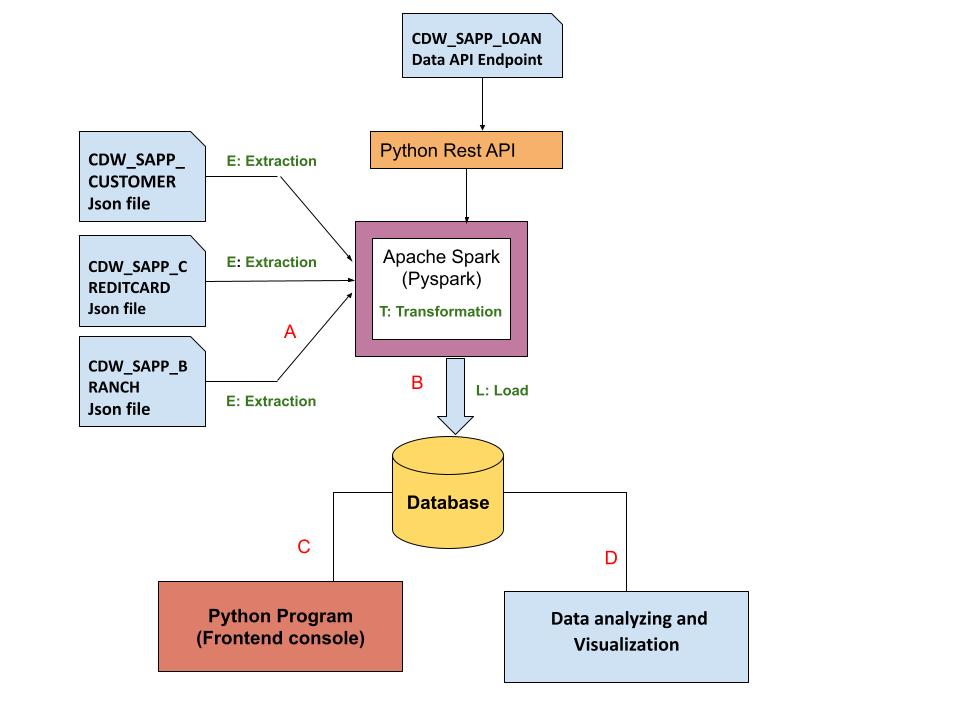
[ACT 400 - The Data Detective Agency:Advanced RDBMS and SQL](https://docs.google.com/document/d/1cKSxu4vGgO5o7TTpbz8L-YdChECpws9W3a5vmrfJ7Y0/edit)  
[ACT 401 - The Data Detective Agency:Python Programming and Database integration](https://docs.google.com/document/d/1OgCf8A--4nDnOTalpw4Dm8KHIICqWWoZsw_ra3zbgeE/edit)

[ACT 402 - The Data Detective Agency: Overview of Spark and Spark SQL](https://docs.google.com/document/d/1LwC73K5EfEw2HTWq96cuS_uEMr-ppCti2ZUjlQdYu9Y/edit)  
[ACT 403 - The Data Detective Agency: Intro to Data Analysis](https://docs.google.com/document/d/1z2g53g4jmyt9P7s0NUzH1rjdKsq9_FZPn80F0a-wDhg/edit)  
[ACT 404 - The Data Detective Agency: Power BI](https://docs.google.com/document/d/1BmTG4_FCcIdNhacc1nP2MOBtcvwDyOW5ieDesvUD-yU/edit)

## 

## Workflow Diagram of the Requirements.

The workflow diagram below will help you visualize the flow and scope of this capstone project at a high level.

****

## Credit Card Dataset Overview.

The Credit Card System database is an 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.

1. **CDW\_SAPP\_CUSTOMER.JSON:** This file has the existing customer details.
2. **CDW\_SAPP\_CREDITCARD.JSON**: This file contains all credit card transaction information.
3. **CDW\_SAPP\_BRANCH.JSON:** Each branch’s information and details are recorded in this file.

[**Click here to download the Credit Card system files**](https://drive.google.com/drive/folders/1J4a2UndLvVWszHAL2VxJeVXyAHm3xYIp?usp=sharing)**.**

## Business Requirements - ETL *(Rubric 11% out of 14%)*

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

| **Req-1.1** | **Data Extraction and Transformation with Python and**  **PySpark** |
| --- | --- |
| **Functional Requirement 1.1**  ***Rubric: - (18%)*** | 1. 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**](https://docs.google.com/spreadsheets/d/1t8UxBrUV6dxx0pM1VIIGZpSf4IKbzjdJ/edit?usp=sharing&ouid=109108037194607248998&rtpof=true&sd=true)**.**     1. CDW\_SAPP\_BRANCH.JSON  2. CDW\_SAPP\_CREDITCARD.JSON  3. CDW\_SAPP\_CUSTOMER.JSON  **Note**: **Data Analyst will be required to transform the data based on the requirements found in the** [**Mapping Document.**](https://docs.google.com/spreadsheets/d/1t8UxBrUV6dxx0pM1VIIGZpSf4IKbzjdJ/edit?usp=sharing&ouid=109108037194607248998&rtpof=true&sd=true)  Hint: [You can use PySQL “select statement query” or simple Pyspark RDD]. |
| **Req-1.2** | **Data loading into Database** |
| **Function Requirement 1.2**  ***Rubric: - (9%)*** | Once PySpark reads data from JSON files, and then utilizes Python, PySpark, and Python modules to load data into RDBMS(SQL), perform the following:     1. Create a Database in SQL(MySQL), named **“creditcard\_capstone.”** 2. Create a Python and Pyspark Program to load/write the “Credit Card System Data” into RDBMS(**creditcard\_capstone**).   Tables should be created by the following names in RDBMS:  ***CDW\_SAPP\_BRANCH***  ***CDW\_SAPP\_CREDIT\_CARD***  ***CDW\_SAPP\_CUSTOMER*** |

### 

### 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).

*Note: You can work in a notebook while developing your project, but you must export it to a .py file that works appropriately. We* ***expect*** *your front-end user interface program to be fully functional when you demonstrate your final project.*

**2.1 Transaction Details Module**

| **Req-2.1** | **Transaction Details Module** |
| --- | --- |
| **Functional Requirements 2.1**  ***Rubric: - (10%)*** | 1)Create a function that accomplishes the following tasks:   1. Prompt the user for a zip code, provide contextual cues for valid input, and verify it is in the correct format. 2. Ask for a month and year, and provide contextual cues for valid input and verify it is in the correct format. 3. Use the provided inputs to query the database and retrieve a list of transactions made by customers in the specified zip code for the given month and year. 4. Sort the transactions by day in descending order.   **Remember**: this function should be callable from the main application interface and the output should be screen-reading friendly for the user.  2) Used to display the number and total values of transactions for a given type.  3) Used to display the total 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**  ***Rubric: - (9%)*** | 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. Hint: What does YOUR monthly credit card bill look like? What structural components does it have? Not just a total $ for the month, right?  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.

**Use Power BI for the below requirements:**

| **Req - 3** | **Data Analysis and Visualization** |
| --- | --- |
| **Functional Requirements 3.1**  ***Rubric: - (4%)*** | * Export following tables as a CSV file, [**click here for the direction**](https://dev.mysql.com/doc/workbench/en/wb-admin-export-import-table.html)   + ***CDW\_SAPP\_BRANCH***   + ***CDW\_SAPP\_CREDIT\_CARD***   + ***CDW\_SAPP\_CUSTOMER*** * Import all CSV files into Power BI. * Create a PowerBI report and dashboard for the visualizations below. |
| **Functional Requirements 3.2**  ***Rubric: - (8%)*** | *Create an appropriate visualization to perform the following task -***Calculate and plot which transaction type has the highest transaction count.** *Note: Take a screenshot of the graph. Save a copy of the visualization, making sure it is PROPERLY NAMED!* |
| **Functional Requirements 3.3**  ***Rubric: - (5%)*** | *Create an appropriate visualization to perform the following task -*  * **Calculate and plot which state has a high number of customers.**   *Note: Take a screenshot of the graph. Save a copy of the visualization, making sure it is PROPERLY NAMED!* |
| **Functional Requirements 3.4**  ***Rubric: - (5%)*** | *Create a single appropriate visualization to perform the following task -*  * **Calculate the total transaction sum for each customer based on their individual transactions. Identify the top 10 customers with the highest transaction amounts (in dollar value). Create a plot to showcase these top customers and their transaction sums**.**F**   Hint (use CUST\_SSN).  *Note: Take a screenshot of the graph. Save a copy of the visualization, making sure it is PROPERLY NAMED!* |

## 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 for 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**  ***Rubric: - (5%)*** | Create a Python program to GET (consume) data from the above API endpoint for the loan application dataset. |
| **Functional Requirements 4.2**  ***Rubric: - (4%)*** | Calculate the status code of the above API endpoint.  Hint: status code could be 200, 400, 404, 401. |
| **Functional Requirements 4.3**  ***Rubric: - (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**.**  NOTE: you will also be using this data for the visualization portions below. You would be wise to ALSO save this as a CSV file to cut out step 5.1.2 |

### 

### 5. Functional Requirements - Data Analysis and Visualization for LOAN Application

Create visualizations in Power BI based on the exported CSV file listed below in 5.1.

Ensure that your choice of visualization is best suited to the data being displayed and the target audience.

Ensure that your scale, labels, axis ordering, and data points are well thought out.

**Use Power BI for Visualization**

| **Req-5** | **Data Analysis and Visualization** |
| --- | --- |
| **Functional Requirements 5.1**  ***Rubric: - (1%)*** | * Export “CDW-SAPP\_loan\_application” table as a CSV file,[**click here for the direction.** You should be able to do this with python.](https://dev.mysql.com/doc/workbench/en/wb-admin-export-import-table.html) * Import the CSV file into Power BI. * Using Power BI, create a Power BI report and dashboard to showcase your visualizations. |
| **Functional Requirements 5.2**  ***Rubric: - (4%)*** | *Create an appropriate visualization to perform the following task -*Calculate and plot the percentage of applications approved for self-employed applicants. Use the appropriate chart or graph to represent this data. *Note: Take a screenshot of the graph. Save a copy of the visualization, making sure it is PROPERLY NAMED!* |
| **Functional Requirements 5.3**  ***Rubric: - (4%)*** | *Create an appropriate visualization to perform the following task -*Calculate the percentage of rejection for married male applicants. Use the ideal chart or graph to represent this data. *Note: Take a screenshot of the graph. Save a copy of the visualization, making sure it is PROPERLY NAMED!* |
| **Functional Requirements 5.4\***  ***Rubric: - (4%)*** | *Create an appropriate visualization to perform the following task -*Calculate and plot the top three months with the largest volume of transaction data. Use the ideal chart or graph to represent this data. ***(hint: use CDW\_SAPP\_CREDIT\_CARD.csv)***  *Note: Take a screenshot of the graph. Save a copy of the visualization, making sure it is PROPERLY NAMED!* |
| **Functional Requirements 5.5\***  ***Rubric: - (4%)*** | *Create an appropriate visualization to perform the following task -*Calculate and plot which branch processed the highest total dollar value of healthcare transactions. Use the ideal chart or graph to represent this data. ***(hint: use CDW\_SAPP\_CREDIT\_CARD.csv)***  *Note: Take a screenshot of the graph. Save a copy of the visualization, making sure it is PROPERLY NAMED!* |

**References:**

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

Apache Spark - Spark SQL:  
<https://spark.apache.org/sql/>