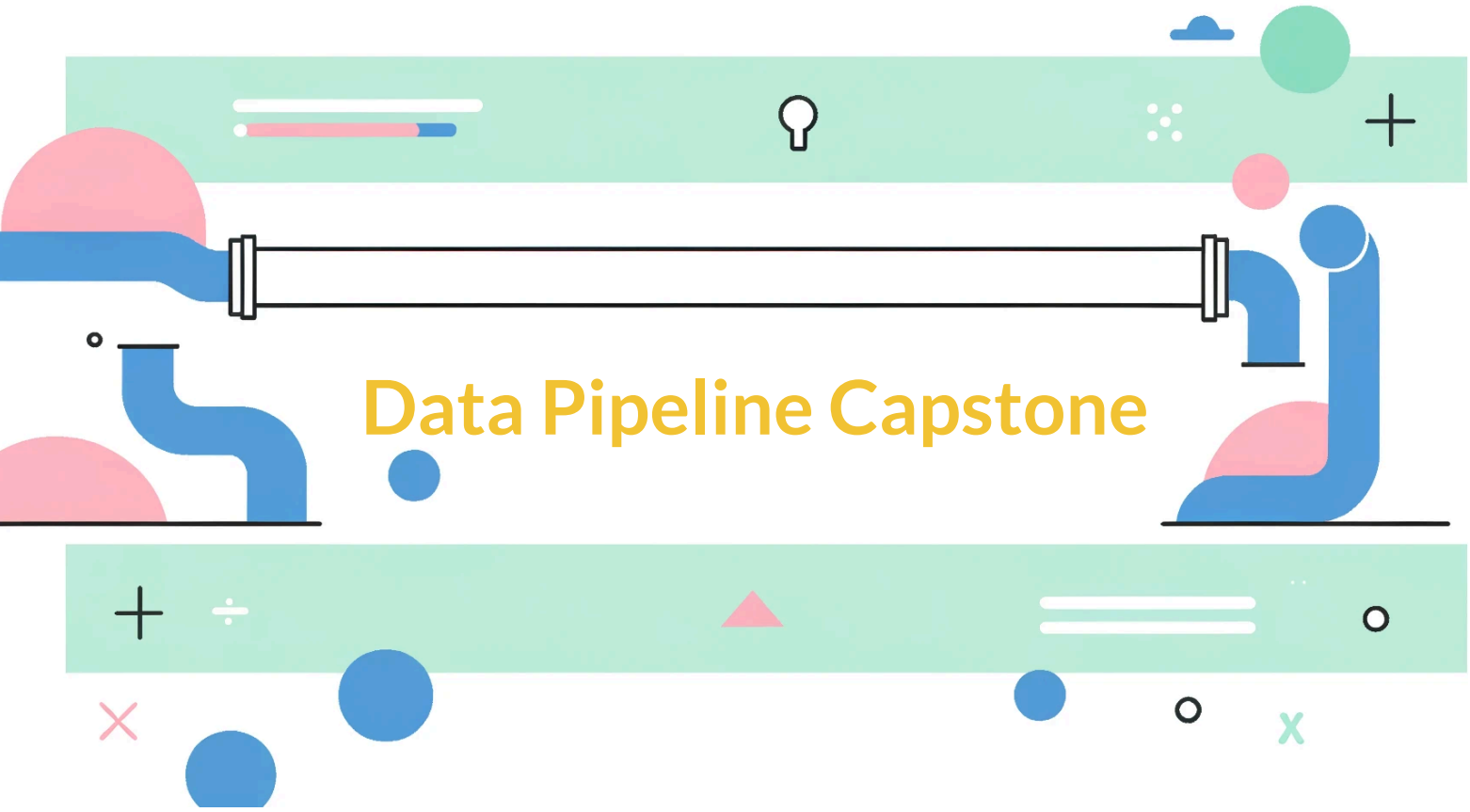


Data Pipeline Capstone

Introduction

In this capstone project, students will embark on a journey to architect and implement a data pipeline reflective of industry practices, albeit on a smaller scale. The project aims to simulate real-world data challenges, requiring students to source semi-structured data, which they will then process and analyze for insights. The choice of data source remains open-ended, encouraging creativity and exploration of domains such as social media analytics, e-commerce, public health, or any field of interest.

Functionality

The goal is to build a data pipeline capable of handling semi-structured

data through various stages: extraction, transformation, loading (ETL), and analysis. This process will illuminate the journey from raw data to actionable insights, a cornerstone in data-driven decision-making.

Tier 1 - Local: In this tier, students will leverage their own machines and local servers to set up the data pipeline. This involves more hands-on work with coding, database management, and manual setup of environments and tools.

Tier 2 - Cloud-based: This tier shifts the focus to utilizing cloud services for the data pipeline. Students will explore and integrate various cloud-based tools and services to manage the data lifecycle, reducing the need for local resource management and manual configurations.

Process and Checkpoints

- *Extraction of Semi-Structured Data into Data Lake*:
  - **Tier 1**: Manually script the extraction process to pull data from chosen sources, storing it locally or in a simulated data lake environment.
  - **Tier 2**: Use cloud storage services (e.g., AWS S3) to host the data lake, with automated or semi-automated data ingestion methods.

➢ *Transformation of Semi-Structured Data into Structured Data:*

- **Tier 1**: Employ Python libraries (e.g., Pandas) for data cleaning and transformation tasks.
- **Tier 2:** Leverage cloud-based ETL services (e.g., AWS Glue) to automate the transformation process.

➢ *Loading of Structured Data into Data Warehouse:*

- **Tier 1**: Load the transformed data into a locally hosted database or data warehouse solution.
- **Tier 2**: Utilize a cloud-based data warehouse (e.g., Amazon Redshift) for data storage, benefiting from the scalability and managed services.

➢ *Analysis on Warehouse Data:*

- **Both Tiers**: Conduct analytical queries on the structured data to derive insights. This could involve SQL queries, Python analysis scripts, or cloud-based analytics tools. For whatever you're analyzing, be sure to include analysis of both the dataset itself (via descriptive statistics) and correlations among the data (via inferential statistics).

Backlog / Deliverables

Each checkpoint must be accompanied by the following deliverables:

Data Extraction:

- ➢ Documentation on data source, extraction logic, and justification for chosen methods.
- ➢ Extraction code and a data sample.

Data Transformation:

- ➢ A report detailing transformation rules, challenges, and solutions.
- ➢ Transformation code along with data samples pre and post-transformation.

Data Loading:

- ➢ A description of the data warehouse architecture, schema design, and loading strategy.
- ➢ Code or configurations for the loading process, along with proof of successful execution.

Data Analysis:

- ➢ An analysis report with insights, including the queries or scripts used, visualizations, and interpretations of the data.

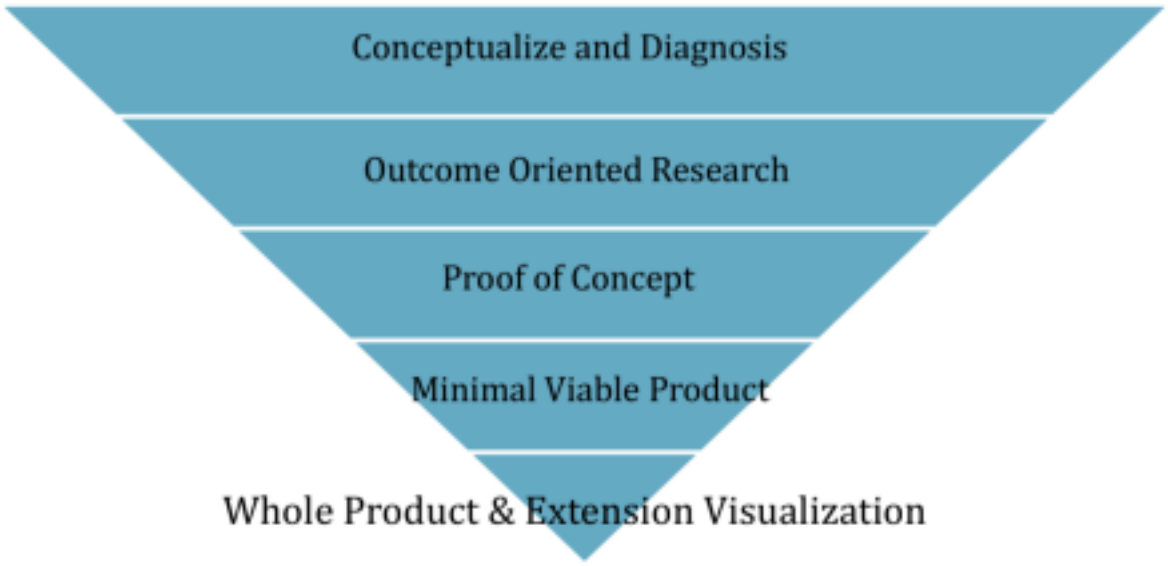
Project Reflection and Presentation:

- A reflective overview discussing the selected tier, utilized technologies, encountered challenges, and key takeaways. This should be in presentation format - your team will be given 30 minutes (maximum) to present, with questions from peers (not included in the 30 minutes) at the end of the presentation.

Considerations when Developing a PRD

A product requirements document (PRD) is an artifact used in the product development process to communicate what capabilities must be included in a product release to the development and testing teams.

The PRD will contain everything that must be included in a release to be considered complete, serving as a guide for subsequent documents in the release process. While PRDs may hint at a potential implementation to illustrate a use case, they may not dictate a specific implementation. The process diagram below showcases the steps considered when developing a PRD.



Whole Product & Extension Visualization