Data Analysis Capstone Process Guidelines & Expectations

**Overview:**

Your projects are your number one way to show off your skills and land a job as a developer. This document will provide an outline of the minimum requirements to pass Code:You. There will be a base project that outlines standards for your project then a list of additions you can add after the base project is complete.

**Capstone Process Module 3**

**1. Project Planning (Begins in Module 3)** During this stage, you will select a dataset and begin defining the overall direction of your capstone project. The dataset you choose must meet the following criteria:

* Contains **at least 1,000 rows**
* Contains **at least 10 columns**
* Includes a mix of **quantitative and qualitative data**
* You will use at least **two datasets**.

Once the dataset is selected, complete the following documentation in your README.md file:

* **Data Dictionary** – Describe each variable in the dataset.
* **Data Summary** – Provide basic statistics and an overview of the data.
* **Data Source** – Cite where the data came from, including any relevant links or attributions.

**2. Project Objective** Define a clear project objective or guiding question based on your dataset. This objective should outline the main problem you're aiming to solve or the insight you hope to uncover.

**3. Project Plan Submission** Create a brief project plan to submit to your mentors. This plan should include:

* The methods you intend to use to analyze the data
* Your rationale for choosing these methods
* An explanation of why your approach is a suitable and effective way to address the project objective

Mentors will review this plan to provide feedback and approve your capstone direction.

### **Module 4: Capstone Project Development**

Module 4 is fully dedicated to executing your capstone project. Once your project plan has been submitted and approved, you will begin developing your project in a professional, organized, and presentable format.

#### **Project Implementation**

* All work must be completed in a **Jupyter Notebook** and uploaded to your **GitHub repository**.
* The **project deadline is final**—no extensions will be granted.
* You must follow all requirements and project baselines outlined in the official Capstone Project Requirements document.

#### **README.md Requirements**

Your README.md should serve as a user guide for your project and must include:

1. **Project Setup Instructions**
   * Step-by-step directions on how to download, install, and run your project
   * Instructions for setting up the virtual environment
   * You should **test this process** by cloning your repo into a new folder or onto a different machine
   * Have others test your project and test theirs.
2. **Project Overview**
   * A description of what the user should expect from the project once it's running
   * Assume the reviewer has **no coding background**, so keep language clear and simple
3. **Technologies Used**
   * List all key tools/libraries used and explain their role in the project
     + Example: *“Pandas was used to clean and manipulate the dataset to identify trends in customer behavior.”*
     + Example: *“The project was developed in Jupyter Notebooks to allow for clean, narrative-driven presentation of both code and results.”*

**Data Handling Requirements**

* All data must be imported from a **public API** or using a **relative file path** (e.g., ./data/file.csv)
* **Do not** use absolute file paths (e.g., C:/Users/YourName/...), as these will not work on other systems

#### **Version Control and GitHub Practices**

* Your GitHub repository must include **at least 10 separate commits**
  + It's best practice to commit every time you complete a work session to avoid data loss
* All commits must be made using the **command line**
  + **Do not** use the file uploader, as this will result in project failure

#### **Notebook Guidelines**

* Use **Markdown cells** to explain the objective or logic of your code. This doesn't mean annotating every code block, but the narrative should clearly walk the reviewer through your process.
* Organize your notebook to be **visually appealing** and easy to follow:
  + Remove unused or test code
  + Present finalized code cleanly and professionally
  + Use **consistent formatting and correct spelling** throughout
  + Apply a **cohesive color palette** to any visualizations for a polished cohesive appearance.

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### **Project Guidelines**

To pass your capstone, your project must meet the following **minimum requirements**. These criteria are designed to demonstrate your understanding and application of the skills learned throughout the cohort.

#### **Environment & Dependencies**

* Use a **virtual environment** for your project.
* Include a requirements.txt file in your repository.
  + Only list the **libraries required** to run your project.

#### **Data Ingestion & Cleaning**

* Read your data using one of the following methods:
  + From a **local file**
  + Using a **public API**
  + Through **web scraping**
* Perform thorough data cleaning, including:
  + Handling **null or missing values**
  + **Normalizing** data where applicable
  + **Renaming columns** for clarity
  + Verifying that **column data types** accurately represent the data

#### **Feature Engineering & Functions**

* Perform **feature engineering** by creating at least one **new column** based on existing data.
* Implement **at least three custom functions** in your code.
  + Use **best practices**, including:
    - **Type hinting**
    - **Docstrings** to explain function purpose and parameters

#### **Database Integration**

* Store your cleaned data in a **SQLite database**.
  + Create **at least two tables**
  + Define a **primary key** for each table to allow joins
* Perform a **SQL join** between your tables to retrieve only the data needed for your visualizations

#### **Visualizations**

* Create **three different plots** to illustrate your analysis.
  + Each plot must be a **different type** (e.g., bar, line, scatter, etc.)
  + All plots must include:
    - A **title**
    - Clearly labeled **axes**
    - A **unified color theme** (avoid using default colors)

### **Optional Enhancements (For Going Above and Beyond)**

If you'd like to further showcase your skills and stand out, consider adding one or more of the following **optional features** to your project. These are not required to pass, but they demonstrate initiative, creativity, and real-world application.

#### **Interactive Dashboard**

* **Build a dashboard** to present your findings in an interactive and visually engaging way.
  + Recommended tool: **Tableau** (easy to use and great for storytelling)
  + Alternatives: **Power BI**, **Plotly Dash**, or **Streamlit** (for more technical control)

#### **Project Website**

* **Create a simple website** to display your project summary, visualizations, and conclusions.
  + **Basic (Front-End Only)**: Use **HTML/CSS** to design a static site showcasing your results.
  + **Advanced (Full Stack)**: Build a dynamic web app using:
    - **Flask** or **Django** (Python web frameworks)
    - Combine with your existing data processing scripts

Note: Your project must still meet all required components. Any additional features should use your **existing codebase**, organized in a way that separates core logic from interface code (e.g., functions and models in reusable modules).

#### **User Interface (UI)**

* Build a **simple user interface** for interacting with your analysis or visualizations.  
  Input fields for user-defined filters (e.g., selecting a region or category)
  + Buttons to trigger analysis or update graphs
  + Frameworks: **Tkinter**, **PyQt**, **Streamlit**, or front-end tools like **JavaScript with D3.js**

#### **Bonus Feature Ideas**

* **Model Deployment**: Train and deploy a basic ML model (if applicable) using tools like **scikit-learn**, **TensorFlow**, or **FastAPI**
* **API Creation**: Wrap your cleaned data or analysis in a REST API using **Flask** or **FastAPI**
* **Export Options**: Allow users to **download visualizations** or **filtered data** as CSV/PDF
* **Mobile-Friendly Design**: If building a web app, ensure it's **responsive** on mobile devices

### **Project Summary**

You are required to write a **clear and concise project summary** focused solely on your **problem and findings**. Avoid detailing the development process—this summary should be written for a **non-technical audience** and should clearly communicate the insights from your analysis.

Your summary should include:

* A description of the **problem or question** you aimed to solve.
* A summary of the **key findings** from your data analysis.
* Any **data inconsistencies, limitations, or issues** that could have influenced your results.

**Helpful references for writing your summary:**

* [DashThis: Analytical Report Guide](https://dashthis.com/blog/analytical-report-guide/)
* [Carnegie Mellon Paper Structure Guide (PDF)](https://www.stat.cmu.edu/~brian/701/notes/paper-structure.pdf)

### **Project Presentation**

You will also create a **visual presentation** that complements your project summary. This presentation is intended to communicate your analysis clearly and effectively using visuals and simple language.

Your presentation must include:

* A **brief overview** of the issue or question that led to your analysis.
* **Key findings**, displayed with **visuals** (e.g., charts or graphs) and short, clear explanations of why these findings matter.
* Mention of any **data inconsistencies or quality issues** that could have affected the results.
* A **concise conclusion** that summarizes your overall takeaway, without repeating all the individual points or visuals.

The goal is to make the project accessible and engaging, even for those with limited data experience.

**Project Review**

Your final project will be reviewed by **Code:You staff and mentors**, including some who are **outside the data pathway**. To ensure a smooth review process, complete the following:

* Make sure your **GitHub repository is public** and can be cloned without issues.
* Double-check that the **setup instructions in your README.md** are accurate and easy to follow.
  + *Tip: Ask a cohort peer to clone and run your project to verify it works on a different machine.*
* Be aware that **only minimal troubleshooting** will be provided. Your project should run as intended on its own.
* Confirm that **all project requirements** (notebook quality, documentation, visuals, commits, etc.) are fully met and properly documented.
* Schedule A time to conduct an interview on your project.