Brief Explanation of the Project

In this assignment, I created a generic Naive Bayes binary classification algorithm that allows users to input their chosen classes and data files.

The application was developed from scratch with no predefined layout, working solely from the algorithm. The challenge involved identifying and applying use cases, and ensuring compatibility with various scenarios in a generic manner.

Users can provide CSV files formatted appropriately for use in the algorithm. Based on each class, the algorithm calculates the win, loss, and draw probabilities using the provided classes (refer to the "How to Run" PDF for a detailed explanation of the algorithm's operation and execution).

The algorithm runs successfully and meets the required functionality, achieving a high initial grade.

Design Decisions

- **Thorough Documentation**: The code is extensively documented with comments explaining each step, making it more understandable. An additional how to run document provided with this, explaining the algorithm, how to run and further expansion of the project.
- **Modular Approach**: The design separates classes, functions, and globals from the main functionality of the algorithm. This structure was implemented logically due to the lack of a template. For example, Laplace Smoothing, which is repeatedly called but not a core part of the Naive Bayes algorithm, was encapsulated in a function.
- **Data Organization**: Data storage and access are handled through classes as a logical way to group information. Most functions are not specific to classes or their data, so methods are not used within the class structure.
- **Exception Handling**: Implemented to ensure safe failure of interruptions or errors, allowing the application to continue functioning despite errors.
- **User Interface**: A CLI was used for basic user input, providing a simple interface. A GUI was deemed unnecessary for this small application given the time constraints.
- **Relevant Libraries**: Libraries such as pandas and copy were imported for numerical data handling and CSV file import functionality.

Additionally, the code demonstrates the use of loops, conditionals, namespaces, input and output components, logical operations, class instantiation, and library imports.