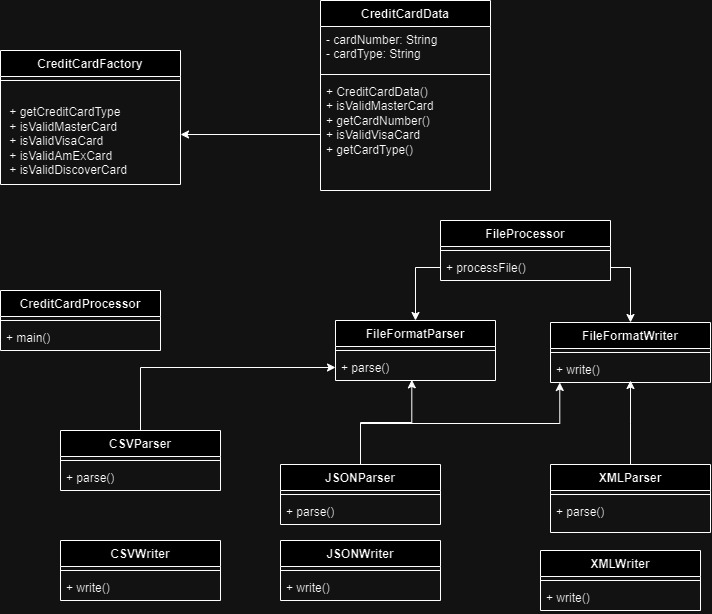
**Title**: Credit Card Processing System

1. **Primary Problem**:

The primary problem addressed in this system is the determination of the credit card type based on the given card number. This involves identifying whether a card number corresponds to a Visa, MasterCard, American Express, or Discover card, based on predefined criteria such as the starting digits and the length of the card number.

1. **Secondary Problems**:
   1. **File Processing**: Reading and writing different file formats (CSV, XML, JSON) to handle credit card data.
   2. **Data Representation**: Creating appropriate data structures to represent and manage credit card information.
2. **Design Patterns Used**:
   1. **Factory Pattern (Creational Pattern)**:
      1. **Usage**: *CreditCardFactory* is used to encapsulate the logic for determining the type of a credit card. It provides a static method *getCreditCardType* that returns the type based on the card number.
   2. **Strategy Pattern (Behavioral Pattern)**:
      1. **Usage**: *FileProcessor* utilizes the Strategy pattern through *FileFormatParser* and *FileFormatWriter* interfaces, allowing for flexible file processing strategies (CSV, XML, JSON).
3. **Consequences of Using These Patterns**:
   1. **Factory Pattern**:
      1. **Pros**: Encapsulates card type determination logic, making the system easy to modify and extend. Reduces coupling by not requiring clients to know the specifics of card type determination.
      2. **Cons**: Can lead to complex code structure if many card types with different validation criteria are added.
   2. **Strategy Pattern**:
      1. **Pros**: Provides flexibility to change file processing strategies without modifying the client code. Promotes the open/closed principle by allowing new file formats to be added without changing existing code.
      2. **Cons**: The strategy interface can become large or complex if there are too many different file formats to support.

**DIAGRAM:**

  
  
  
**GITHUB LINK:**  
<https://github.com/gopinathsjsu/individual-project-Dead-Stone>