Here are the extracted technical terms along with their definitions:

1. **Single Responsibility Principle (SRP)**
   * **Definition:** A design principle stating that a class should have only one reason to change, meaning it should focus on a single functionality or address a specific concern.
2. **SOLID Design Principles**
   * **Definition:** A set of five design principles aimed at improving software design by making it more maintainable, scalable, and easier to understand.
3. **Class**
   * **Definition:** A blueprint for creating objects in object-oriented programming, encapsulating data and methods that operate on the data.
4. **Communication Protocol**
   * **Definition:** A set of rules or conventions that allow devices or systems to communicate with each other over a network.
5. **Message Format**
   * **Definition:** The structure or encoding used to define how messages are formatted, such as JSON or XML.
6. **JSON (JavaScript Object Notation)**
   * **Definition:** A lightweight data-interchange format that is easy for humans to read and write, and easy for machines to parse and generate.
7. **XML (Extensible Markup Language)**
   * **Definition:** A markup language used to encode documents in a format that is both human-readable and machine-readable.
8. **Authentication**
   * **Definition:** A process of verifying the identity of a user or system to ensure secure communication or access to resources.
9. **Module**
   * **Definition:** A self-contained unit of code that encapsulates specific functionality, often used to organize and separate concerns in software.
10. **Responsibility**
    * **Definition:** The specific functionality or task that a class or module is designed to address or handle.
11. **Change Pipeline**
    * **Definition:** The sequence of modifications or updates that occur in a software system as it evolves over time.
    * Ripple Effect
12. **Ripple Effect**
    * Definition: A phenomenon where changes made in one part of the code lead to unintended changes or issues in other parts of the system. This typically occurs when a class or module has multiple responsibilities, causing modifications to impact unrelated functionality.

These terms are central to understanding the concept of the Single Responsibility Principle and its application in software design.