Better Implementation

Currently, the system's design is too rigid. It forces all versions of the system to use certain functions, like "experiment" and "report," even if they don't fit well. This setup makes the system hard to update and maintain because changes to one part might affect others unnecessarily. The design also breaks several key programming rules, which aim to make software designs more flexible and easier to manage.

To improve the design, I would suggest separating the reporting tasks from the main functions of the system. This means creating different classes or modules for handling data reporting separately from the main logic. It is also recommends using specific design patterns that help manage different algorithms as interchangeable options, simplify the creation of new system instances, and allow parts of the system to communicate more freely without being tightly connected. These changes would make the system easier to extend and maintain over time.