S: A class should have one and only one reason to change, meaning that a class should have only one job. For my project, I made sure airQuality, waterQuality, and forestQuality are all classes that have only one job.

O: Objects or entities should be open for extension but closed for modification. For my project, airQuality, waterQuality, and forestQuality all extend environmental categories, but cannot be modified by it.

L: Every subclass or derived class should be substitutable for their base or parent class. For my project, user implements registeredUser and Guest.

I: A client should never be forced to implement an interface that it doesn’t use, or clients shouldn’t be forced to depend on methods they do not use. In my project, waterQuality, airQuality, and forestQuality are all categories. All waterQuality, airQuality, and forestQuality have separate probes that they pull their data from. This is separated from the categories.

D: Entities must depend on abstractions, not on concretions. It states that the high-level module must not depend on the low-level module, but they should depend on abstractions. For my project, the database connection interface, waterQuality, airQuality, and forestQuality do not depend on each other.