5 design principles intended to make understandable , flexible , robust and maintainable .

To avoid code smelling

S - The single responsibility principle

O- Open close Principle

L – Liskov substitution principle

I – Interface segregation principle

D – Dependency Inversion principle

Graphical user interface, text, application

Description automatically generated

Open Closed Principle

Software Entities (classes ) should open for extension and closed for modification.

Graphical user interface, text, application, email

Description automatically generated

Liskov Substitution Principle

* Derived classes must be able to substitute for their base classes without interrupting the current behaviour of the program
* Text

  Description automatically generated

Interface segregation :

* Client should never be forced to implement an interface that doesn’t use or client shouldn’t be forced to depnednet on methods they do not use
* Text

  Description automatically generated with medium confidence

Dependeny Inversion :

* Entities must depend on abstractions and not on concrete implementation(classes) , also the high level module must no depend on the low-level but both should depend on abstractions