

# Best Coding Practice

## Naming

- Interfaces
  - Should be a good Object Name
  - Never start with an 'i' (This is a dot net thing)
- Implementation
  - With only 1 implementation: user + Impl
  - With more than one the name should indicate the difference of implementation

## SOLID Principals

1. Single Responsibility Principal
  - Each class has a single responsibility
  - No more than a screen full of code per class
2. Open/Closed Principal
  - Classes should be open for extension but closed for modification
  - Use private getters and setters ONLY when you need them
3. Liskov Substitution Principal
  - Objects should be replaceable with instances of their subtypes without altering correctness
  - E.g. A square is a rectangle
4. Interface Segregation Principal
  - Make fine-grained interfaces that are client specific
  - Keep components focused and minimize shared dependencies
5. Dependency Inversion Principal
  - Abstraction should not depend on details
  - Important that high and low level objects depend on the same abstract implementation

---

Revision #1

Created 17 April 2022 00:36:00 by Elkip

Updated 17 April 2022 00:36:23 by Elkip