**DRY – Don’t repeat yourself (не повторяй себя)**

##### “Если вдруг вы ловите себя на том, что этот код вы уже писали/встречали раньше, остановитесь, подумайте, и не повторяйте себя*.*”

##### “Самый простой подход по уменьшению сложности — разделить систему на управляемые части.”

На пример компонент, отвечающий за управление пользователями. Этот компонент может быть разделен на другие компоненты, допустим, управление уровнями доступа, который сможет работать с другими компонентами системы безопасности.

##### “Каждая часть данных должна иметь четкое, надежное представление в системе.”

В идеальной системе каждая часть бизнес логики инкапсулирует свои данные в представление – переменную или свойство класса. Это свойство инкапсулирована в класс, отвечающий за часть архитектуры. Класс является частью компонента, который является реализацией части функционала системы.

Четко определите представления (св-ва), которые ваш код будет выдавать другим частям системы, и которые он должен скрывать (private/public).

Убедитесь, что ваш код слабо связан с другими представлениями системы – жесткие связки очень плохо сказываются на общей архитектуре. (Inversion of Control)