

Lessons learnt

1. Gained experience of developing in phases - Inception, elaboration, construction and transition
2. Importance of test driven development in software engineering. It not only helps write code with fewer defects, but also helps in code refactoring.
3. Identifying code smells and different ways of refactoring them.
4. Types of design patterns and why they are needed
5. The importance of low coupling leading to easier refactor within the system.

Challenges

1. Understanding the requirements and modeling the use cases and domain diagrams as per the scope of the project.
2. Deciding on a framework to be used for the project.
3. Deciding on what controllers are needed for the system and what responsibilities are should be assigned to them as per the GRASP principle
4. Applying the design patterns and refactoring methods in context of our project.