**Milestone 3**

**Group 3**

**Alshaykhah Alshammari**

|  |  |  |
| --- | --- | --- |
| **SID** | **STUDENT NAME** | **Effort given** |
| 202110519 | Maryam Bawazir | 25% |
| 202205676 | Hala Hamdoun | 25% |
| 202003441 | Alshaykhah Alshammari | 25% |
| 202109396 | Rogayah omar | 25% |

**CMPS310– B51**

**7, December 2024**

**L51**

**Task 1**

We chose the following design patterns:

1. **Singleton Pattern**

We applied this design pattern to the Database class. This class is responsible for managing the number of instances created in our system, such as Vehicle, Owner, InsurancePolicy, FitnessCertificate, RegistrationSticker, Invoice, and array lists for each of them.

After comparing all the available design patterns, we concluded that the Singleton design pattern works best for this class, as it should have only one instance and is responsible for controlling the number of instances created.

1. **Proxy Pattern**

We applied this design pattern to several classes, including Owner, AccidentReport, CreditCard, FitnessCertificate, and InsurancePolicy.

We used this design pattern for two main reasons:

1. The classes Owner, AccidentReport, and CreditCard contain sensitive information that needs to be protected.
2. Classes like InsurancePolicy and FitnessCertificate communicate with external systems (e.g., insurance companies and workshops). This communication involves remote retrieval of data from these systems, so these classes should not be directly exposed.

We updated all the mentioned classes accordingly with their respective design patterns

**Task 2**

**Task 3**

**Reflection Report**