# Software Engineering (CSC 510) Mini project 2 report

## Design Pattern Description

**Gati Shah (gshah)**
**Priyanka Shankaran (priyan)**

### State design Pattern:

The State design Pattern is used to encapsulate varying behavior for the same routine based on an object's state object i.e. it provides the ability for an object to change its behaviour in response to internal state changes. We decided to include this design pattern in our architecture because the application demands different visibility states for different users. Since we have two users in our architecture, each user has some set of visibility components different, due to access restrictions for each. Hence the state of the interface changes with the type of user login.

### Mediator Design Pattern:

With the mediator pattern, communication between objects is encapsulated with a mediator object. Objects no longer communicate directly with each other, but instead communicate through the mediator. This reduces the dependencies between communicating objects, thereby lowering the coupling. The data access layer acts as a mediator and mediates between the database layer and the application layers of the software. It queries the data source for the data, maps the data from the database to a application layer, and persists changes in the application layer to the database. A mediator separates the actual logic from the interactions with the underlying database. The separation between the database and application provides the following benefits:

- It centralizes the data logic.
- It provides a flexible architecture that can be adapted as the overall design of the application evolves.

### Facade Design Pattern:

The Facade design pattern hides the complexities of the larger system and provides a simpler interface to the client. It typically involves a single wrapper class which contains a set of members required by client. The facade design pattern provides an outward appearance to conceal a less pleasant or creditable reality of the application layer. The client application helps user submit their requests via a more user friendly, easier to use interface as also allows them to understand and test the system since the facade has convenient methods for common tasks.

### References:

1. http://en.wikipedia.org/wiki/State_pattern
2. http://en.wikipedia.org/wiki/Facade_pattern
3. http://en.wikipedia.org/wiki/Mediator_pattern