Design patterns

* Change UML Class Diagrams

Factory method for the creation of users

A diagram of a diagram

Description automatically generated

Composite pattern for the house layout (house>room>light/window/door,…)

The composite design pattern is used for the implementation of the house layout since the intent of that design pattern is to structure objects into a hierarchy in a tree form. In the case of the house layout, it starts with 1 house, then it breaks down into a small number of rooms (e.i. bedrooms, kitchen garage, etc…) and in each room there will have a specific (according to the type of room) number of lights, doors, and windows. The number of objects increases the deeper down the tree. It started from one “House” object to a few dozen of light, door and window objects.

A diagram of a component

Description automatically generated

Singleton for the dashboard

The Singleton is the perfect design pattern for the implementation of the dashboard since it relies on a class having only one instance while giving a global access point to the instance. There should not be many dashboards, rather a single dashboard in the simulation that can be accessed by all existing users. The dashboard should include all the existing actions the user can perform, but depending on the type of user, some actions can be enabled or disabled in the dashboard, giving a level of restriction according to each user. Therefore, instead of creating a new instance of dashboard every time a new user appears, it would be easier for each user to access the dashboard.

A computer screen shot of a diagram

Description automatically generated