In order to implement the Singleton design pattern, the GUI class was designed in a way that it would be responsible for creation, initialization, access, and enforcement of all aspects of the program. Instances were declared private static and public static methods were used to encapsulates all initialization code, and that provided access to the instance. Composite design pattern is a partitioning design pattern and describes a group of objects that is treated the same way as a single instance of the same type of object, for the farm program, the instance or the component was the Item class, and all other classes, Crops, Supplies, Drone, LiveStock, and Equipment, were extended from the Item class or they were the leaves. The purpose of composite design pattern is to achieve a tree structure of all the objects in different classes of the program, and allowing the nodes of the tree to perform different tasks.

