classobjects package adhere to Stable Dependencies Principle SDP. This is because there are no classes inside the "classObject" package that relies on the classes outside the package. We chose to do this because it is easier to make changes to the existing classes inside the package because other classes outside the package will not be affected. Also, we can easily extend the package by adding more classes inside the package without needing to make changes to classes outside the package.

In our application, the acyclic dependencies principle also applies here, because there is a total of less than 3 packages in our implementation, this means there is no chance for there to be a cyclic dependency between packages.

In our application, The common closure principle is also being used here. Because most of the classes inside the "classObjects" are subclasses of the User class.This means that changes that were made to one class inside the package, will only affect similar components inside the package.So it is easy to make changes inside the package. By doing this, the package will be highly cohesive and thus will obey the single responsibility principle.

Further more, in our application we used MVC architecture for each sub system.For example, ModifyBooking sub system is separated into: modifyBookingView, ModifyBookingModel and ModifyBookingController. One of the advantages of using MVC is easy code maintenance which makes it easier to extend and grow, improving the project stability. ForMoreover, each of the MVC components can be tested separately from the user. By separating each subsystem in a MVC architecture pattern, it allows development of project to be done parallelly.For example, if we need to do some update on a sub system, each person of this project can work on different subsystem such as login and updateBooking while not affecting each other. Another advantage of MVC architecture pattern is it divides the application into 3units: Model, view,controller, preventing us from adding complexity to the project. It also provides us with clean Separation of Concern (SoC) while being Search Engine Optimisation (SEO) friendly.

Although MVC provides us with lots of benefits, there are also disadvantages to it. One of the main concern is the framework navigation can be complex on some occasion as it introduce a new layer of abstraction which requires user to adapt to the decomposition criteria of MVC.

Facade design pattern is also used in this project for example: landingpageFacade and adminPageFacade. It helps us to simplify the interface while also decouple a client from a subsystem of components. By using facade in the system, we can easily structure a subsystem into layers and isolate our codes from the complex subsystem. Finally, it also helps to create a balance between CCP and CRP.

The singleton design pattern is also being used for scenario 1.1 and 1.2, The advantage for this is so that the singleton class can act as a database to store to previous configurations of the booking records, only then the system will be able to verify the change to previous booking by comparing the start time of the future booking and the previous booking with the same testing site being selected.The disadvantage of singleton classes is that they can cause the code to be more tightly coupled, and also it can be more harder to test the application because the constructor of the singleton only runs once, so you have to rerun the whole application again to test the singleton object.