1. **Coupling**

The AIMS project is designed following MVC architecture. Each module only depends on each other via function parameters, which make each class more independent.

All classes use private variables so there is no direct access from one class to another, which prevent content coupling

The three classes CD, DVD and book have the same type of media with the same method and they extend general class Media, which prevent control coupling.

=> The project is data coupling

1. **Cohesion**

The classes Util, Config is global and can be used by many other classes. However, we provide static methods and variables for them so we prevent coincidental cohesion.

Media, Book, CD, DVD are separated so it is not logical cohesion.

Parts of modulus are grouped by when they are processed. Example: Class PlaceOrderController so it is ok for temporal cohesion

Loops, multiple conditions and steps in the codes which is procedural cohesion and we can clearly see them in the class PlaceOrderController

Class API has the output of one method is the input of another, reuse the related functions => which is good for functional cohesion

=> The project is really good for cohesion