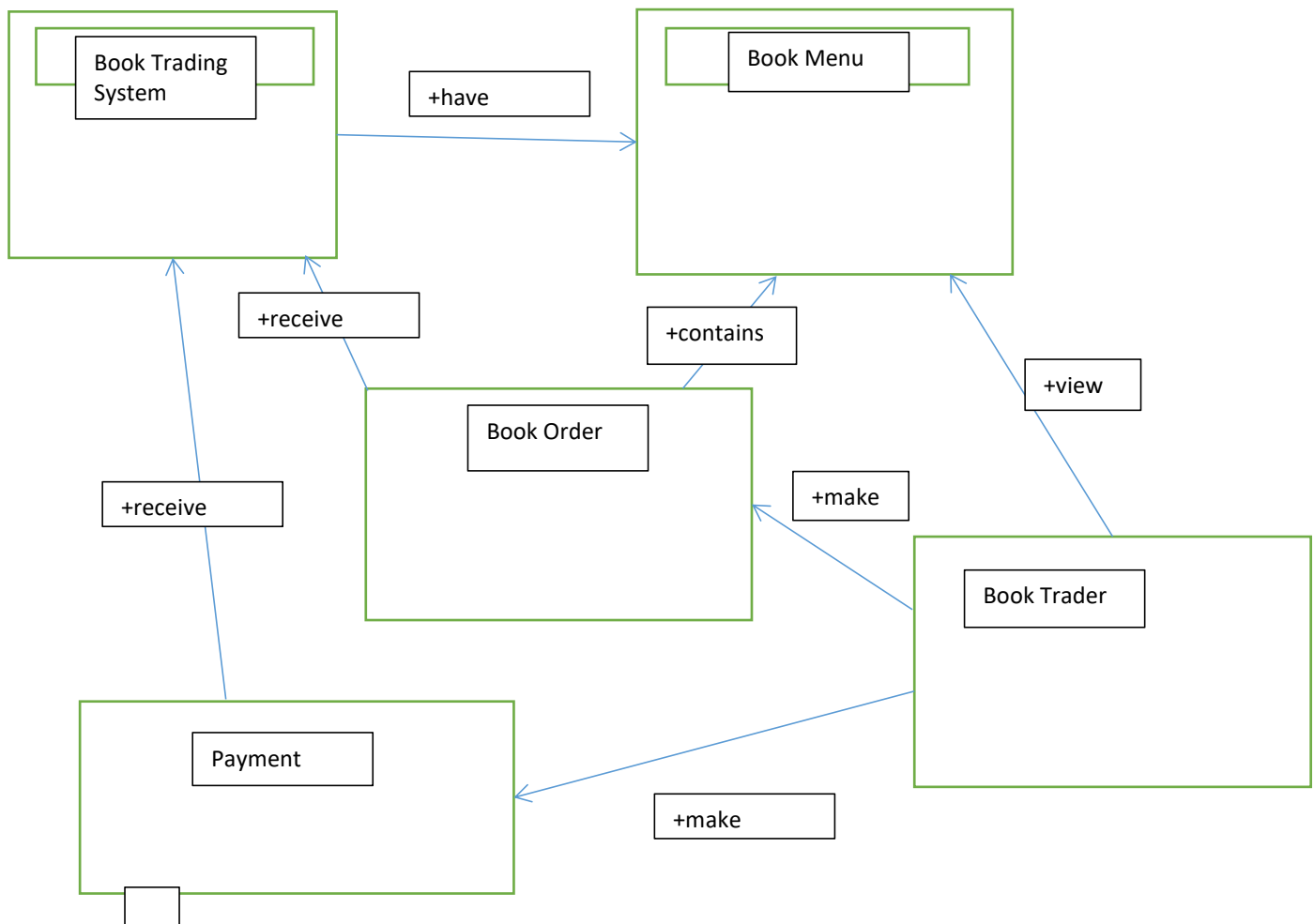


Name: Hong Khee Chee
ID: P21013167

Task 3 – Creating a Class diagram and design pattern selection – 30 marks

Simple class diagram



Class Responsibility

Book Trading System consist of a menu to let user search for book for trading
In the book menu user can sign up or login ,search for specific book
Upload books or buy books .popular and featured book will also displayed on the menu

Book menu display list of book for user to trade, after user confirm order book menu will direct user to book order and payment.

Book order in this page,user can cofirm the book order,or reject if the order incorrect

Book trader is the user of the website, they can make order ,make payment ,upload book ,view available book on the menu

In payment, user can see order information,price of book, they can choose payment method To pay for the book they wanted.

Design Pattern

The Factory Method design pattern is one of the "Gang of Four" design patterns that describe how to solve recurring design problems to design flexible and reusable object-oriented software, that is, objects that are easier to implement, change, test, and reuse.

The Factory Method design pattern is used instead of the regular class constructor for keeping within the SOLID principles of programming, decoupling the construction of objects from the objects themselves. This has the following advantages and is useful for the following cases, among others:

Allows construction of classes with a component of a type that has not been predetermined, but only defined in an "interface", or which is defined as a dynamic type. Thus, for example, a class Vehicle that has a member Motor of interface IMotor, but no concrete type of Motor defined in advance, can be constructed by telling the Vehicle constructor to use an ElectricMotor or a GasolineMotor. The Vehicle constructor code then calls a Motor factory method, to create the desired Motor that complies with the IMotor interface.

Allows construction of subclasses to a parent whose component type has not been predetermined, but only defined in an interface, or which is defined as a dynamic type. For example, a class Vehicle with a member Motor defined with a dynamic type, can have subclasses of type ElectricPlane and OldCar each constructed with a different type of Motor. This can be accomplished by constructing the subclasses with a Vehicle factory method, while supplying the motor type. In cases like this, the constructor may be hidden.

Uml
diagram

