WEB ENGINEERING

Homework #3

Team name: Kilo

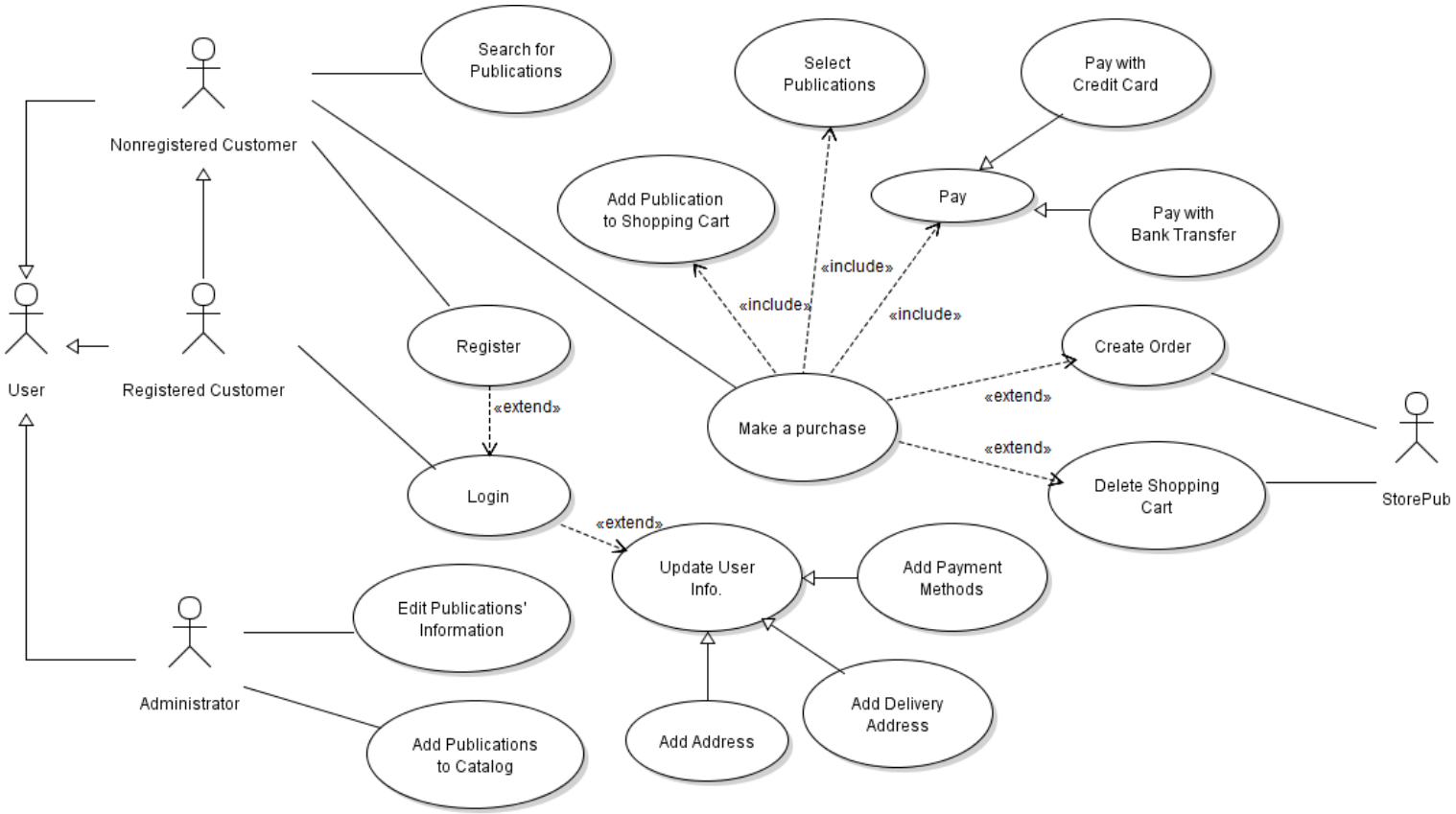
Hanadi Tamimi	215100961
Keya Kashem	215101009
Md Jakaria Nawaz	216203442
Mohammad Nizam Uddin	216101140

Exercise 1: Architectural Styles

a) Yes, it is possible to consider (MVC) as a layered architecture. Considering the (Controller) and the (View) within the (Presentation layer), and the (Model) containing the (Application layer) and the (Data layer). The idea is that (MVC) applications can transfer information between their components in a what could be considered as a layered architecture. The (Views) are used to present the outcome of the functions in the (Controllers) and/or the manipulation of the data in the (Models) as well as handling the user interface. The (Controllers) handle the requests from users and give back responses to (Views) in a way that confirms to the (Models). The (Models) define the database of the application and it's where all the application logic is defined.
b) The UWE layers (Content, Hypertext, Presentation): the (Content) would lay behind the (Models) of the MVC, the (Hypertext) behind the (Controllers) and the last layer (Presentation) lays behind the (Views)
c) High cohesion means a good correlation between the elements of the component. Low coupling means that although the constituents are highly related, each constituent depends on other few constituents which is a good engineering technique; since the less dependencies the less complexity.
d) The MVC uses the Observer Pattern; it is the pattern where there are objects in the system that watch over changes in the components of the system, and whenever there is a change a function is fired to notify the dependent components of that change to adapt with it. In MVC every model has dependencies and whenever an event is triggered to notify the other component of that change.

Exercise 2: Modelling

1. Use case diagram describing the functional requirements of StorePub system:



2. A State Machine diagram describing the lifecycle of a shopping cart in StorePub: