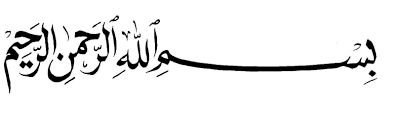
****

**Punjab University College of Information Technology**

**Pick and Drop Services**

**For Female Students**

**SUBMITTED TO**

**Ma’am Amina Mirza**

**SUBMITTED BY**

**BSEF19M006 – Muhammad Adil**

**BSEF19M043 – Abubakar Dar**

**Vision**

**Motive:-**

**“Your Child’s security is our priority”.**

**Description:-**

**We envision to create an environment that is safe for female students and provide them with a secure as well as affordable Pick & Drop service in Pakistan.**

**As parents are busy with their jobs and have no time to pick up and drop their children, we are providing them with this facility at economical cost.**

**Creational Design patterns:**

**Abstract Factory:-**

As the application’s frontend will have two themes dark theme and lite theme.

So the buttons ,background color ,labels, text colors ,animation colors,map colors, locations points colors,path color and many more are the products which will be made in a factory.So selection of these two factories will be decided on the day time base.

**Singleton pattern:-**

Only a single map object will be created and it will be used by everyone So i can assume this pattern will be used on map objects.

**Builder pattern:-**

A driver, Student ,Parents,Cab can have multiple properties and some of the objects can have do not contain these properties so complex objects on the basis of their properties can be made using constructor method.

**Structural Pattern:**

**Adapter:-**

I am assuming the adapter pattern will be used to pay challan as some will use credit cards. However payment is the same but method is different so adjusting the credit card payment as cash.

**Bridge pattern:-**

I am assuming the bridge pattern can be used to define the different color model and its plate number of the same van number, however the same category car can have other functions with slightly different variations.So making these types of cabs will be through bridge pattern.

**Composite pattern:-**

We will use employee classes in which it contains the subordinates which are working under this employee so there is 1 ceo then 2 managers and 1 is handling driver other is handling cashiers.

**Behavioral Design Patterns:**

**Observer :-**

A way of notifying change to a number of classes.

In our project, Manager class is observing changes in different classes like

IsStudentPickedUp() or IsDriverAvailable() etc so observer design pattern is applicable.

**Template :-**

Defer the exact steps of an algorithm to a subclass.

In our project, We can use the template design pattern on Challan class and implement the Pay() method in other classes.

**Mediator :-**

Defines simplified communication between classes.

In our project, We can use mediator pattern to handle communication between drivers because they are not communicating directly with each other but through a middleman i.e manager.