

Software Design And Analysis

Assignment #1

Deadline:7th October, 2020

Question #1:

Waiters on Call (WoC) is a restaurant meal-delivery service started in 2003 by Sue and Tom Bickford. The Bickford both worked for restaurants while in college and always dreamed of opening their own restaurant. But unfortunately, the initial investment was always out of reach. The Bickford notices that many restaurants offer takeout food, and some restaurant - primarily pizzerias - offer home delivery service. Many people they met, however, seemed to want home delivery service with a wider food selection.

Sue and Tom conceived Waiters on Call as the best of both worlds: a restaurant service without the high initial investment. The Bickford contracted with a variety of well-known restaurants in town to accept orders from customers and to deliver the complete meals. After preparing the meal to order, the restaurant charges Waiters on Call a wholesale price, and the customer pays retail plus a service charge and tip. Waiters on Call started modestly, with only two restaurants and one delivery driver working the dinner shift. Business rapidly expanded, and the Bickford realized they needed a custom computer system to support their operations. They hired a consultant, Sam Wells, to help them define what sort of a system they needed.

"What sort of events happen when you are running your business that make you want to reach for a computer?" asked Sam. "Tell me about what usually goes on."

"Well," answered Sue, "when a customer calls in wanting to order, I need to record it and get the information to the right restaurant. I need to know which driver to ask to pick up the order, so I need drivers to call in and tell me when they are free.

Sometimes customers call back wanting to change their orders, so I need to get my hands on the original order and notify the restaurant to make the change.

"Okay, how do you handle the money?" queried Sam.

Tom jumped in, "The drivers get a copy of the bill directly from the restaurant when they pick up the meal. The bill should agree with our calculations. The drivers collect that amount plus a service charge. When drivers report in at closing, we add up the money they have and compare it with the records we have. After all drivers report in, we need to create a deposit slip for the bank for the day's total receipts. At the end of each week, we calculate what we owe each restaurant at the agreed-to wholesale price and send each a statement and a cheque."

"What other information do you need to get from the system?" continued Sam.

"It would be great to have some information at the end of each week about orders by restaurant and orders by area of town – things like that," added Sue. "That would help us decide about advertising and contracts with restaurants. Then we need monthly statements for our accountant."

Sam made some notes and sketched some diagrams as Sue and Tom talked. Then after spending some time thinking about it, he summarized the situation for Waiters on Call.

"It sounds to me like you need a system that does some processing when these events occur:

- A customer calls in to place an order, so you need to *record an order*.
- A driver is finished with a delivery, so you need to *record delivery completion*.
- A customer calls back to change an order, so you need to *update an order*.
- A driver reports for work, so you need to *sign in the driver*.
- A driver submits the day's receipts, so you need to *reconcile driver receipts*.

"Then you need the system to produce information at specific points in time

- for example, when it is time to produce:

- an end-of-day deposit slip
- end-of-week restaurant payments
- weekly sales reports
- monthly financial reports

"Based on the way you have described your business operations, I am assuming you will need a database to store information about these types of things:

- Restaurant
- Menu items
- Customers
- Orders
- Order payment
- Drivers

"Then I suppose you are going to maintain the information in a database about restaurants and drivers. You will need to do some processing when you add a new restaurant, a restaurant changes the menu, you drop a restaurant, you hire a new driver

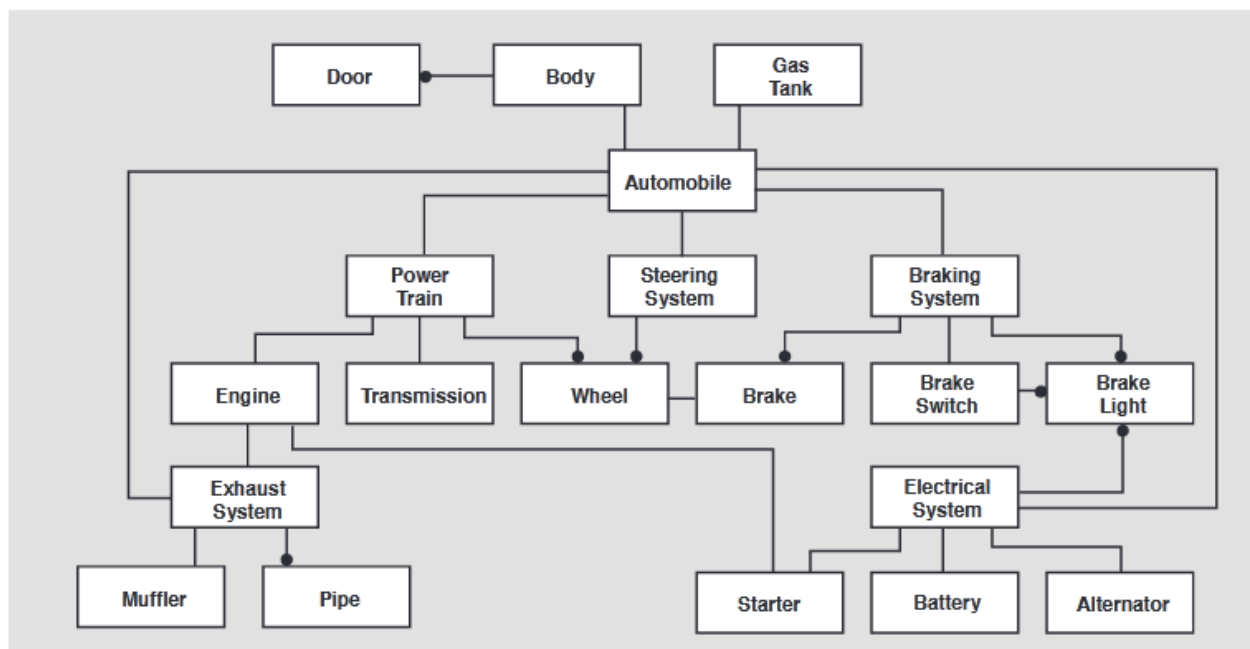
"Am I on the right track?"

Sue and Tom quickly agreed that Sam was talking about the system in a way they could understand. They were confident they had found the right consultant for the job.

So, requirements are finalized now.

Your task is to develop design.(UML class diagram for above requirements in any tool uml modelling tool).

Question #2:



The class diagram in Figure is a partial representation of the structure of an automobile. Improve it by changing some of the associations to aggregations.

Question #3:

Consider a simple electronic circuit. An electrical element in the circuit can be a resistor, a diode, or a complex element. A complex element may

contain any number of resistors and diodes. It may also contain one or more other complex elements. Each electrical element has a unique alphanumerical serial number and it consumes a certain amount of power. The power consumed by a resistor is equal to the product of its resistance and the square of its current. The power consumed by a diode is equal to the product of its voltage and its current. The power consumption of a complex element is equal to the sum of the power consumption of all the resistors, diodes, and complex elements contained in it.

Which design pattern is suitable for designing the electronic circuit described above? [Note: An invalid answer to this part will result in a zero in the entire question.]

Draw a UML 2 class diagram showing the design of the electronic circuit described above. This design must adhere to the structure of the design pattern chosen in part a above.

An incomplete implementation of the electronic system described above is given below. Complete this implementation by using the design pattern selected in part a above.

Missing code section A should contain the complete definition and implementation of all relevant classes.

Missing code section B should do the following in order:

Create two complex elements c1 and c2 with serial numbers "c1" and "c2" respectively.

Create resistor r1 with serial number "r1", resistance value 10, and current value 2.

Create resistor r2 with serial number "r2", resistance value 20, and current value 4.

Create diode d1 with serial number "d1", voltage value 100, and current value 2.

Create diode d2 with serial number "d2", voltage value 200, and current value 4.

Add r1 and d1 to c1.

Add r2 and d2 to c2.

Add c2 to c1.

Output (using cout) the power consumption of c1.

Your code must not have any compilation errors and it must produce the output given below.

```
#include <iostream>
#include <string>
#include <vector>
using namespace std;
```

```
//missing code section A
```

```
{
```

```
//missing code section B
```

```
}
```

Output

Power Consumption of c1 =1360