Name: Daniel Cyril Obon

Student ID: C2650218

Applications Development Report

1. The Advantages and Limitations of OO Design Principles

One of the main advantages that come with using Object Oriented (OO) design principles is the flexibility of using and modifying the code used. This is done by creating subclasses, which allows more functionality or methods to be added to the program. Additionally, these subclasses can also be inherited by making more subclasses (Child Class) that extends from it (Parent Class) and inherit its properties and method, this is known as polymorphism. Moreover, more properties and methods can also be added to the child class that are specific to it by using abstract classes.

Besides that, the code is also more maintainable when implementing OO design principles. Due the code being segmented into many parts and components, this makes it easily organisable such as encapsulating particular data or properties into their own subclasses by relationship. This is turn makes it easier to debug the code as errors only affect certain classes or objects of the code. Thus, with this maintainability it allows for the code to be reused for different projects in the future as the coder can pick the classes or objects that are only relevant to the current project and dismiss the rest.

However, there are some limitations that can come with implementing OO design principles. One such limitation is due to its complexity of implementation. As such, it is dependant on the users experience and knowledge in managing and planning the implementation of these principles. This complexity can stem from the knowledge needed in creating subclasses properly and the proper use of class inheritance. With this complexity, it may take more time when making a program as the planning and designing phase of the program before coding is essential to ensure the proper use of class inheritance.

2. Legal, Security and Ethical Issues Surrounding Application Development

A major factor to consider when developing an application is the usage and protection of personal data. Personal information such as credit card information, and a person's name and home address should be protected and used properly by the company using the implemented application as failing to do so could lead to fines. With the Data Protection Act 2018 implemented by the GDPR in the UK, it states that personal information that is collected can only be used lawfully and transparently. As such, users can request from how their data is used, and the party implementing the application in question must conform to the request.

Besides that, network security should also be kept in mind when developing an application. The party that is implementing the application should take steps in increasing that network security such setting up firewalls and monitoring the network regularly to ensure that it functions as intended and no security breaches have occurred. It is essential that security measures be implemented to lessen security risk which may violate the privacy of the users of said application.

Lastly, respecting intellectual properties should also be considered. This can be done respecting copyrights and trademarks as well as citing and giving credit to sources used during the development of the application. Failing to do so leads to the plagiarism of others work which infringes on copyrights, trademarks and patents, leading to legal action being taken.

3. Evaluation of Program Development

When approaching the problem, identifying the classes and subclasses needed for the program needs to be considered first. This can be done by making a class diagram of the relevant classes, in this case the Furniture class with its associated subclasses Chair, Table and Desk. After identifying the classes and subclasses needed, the properties between them should then be considered next. This is done by deciding whether to make the properties private or protected between the parent class and child class as well its data type (such as int, String, double etc.). Then, implement any methods such as the getters, setters file reading as well as abstract methods which are specific to each child class needed for the application, which in this case would be the 'calcPrice( )' abstract method as the formula used to calculate each furniture is different. Next, program these classes and subclasses in Java, following the class diagram created.

The next step is to design a GUI on paper and finalizing its design, such as button placement, panel and label placement, the number of interactable objects and prompt panel displayed (Option Pane used) for each input or when an error has occurred, before programming it in Java. This helps in giving a clearer picture of what to program in Java and saves the trouble of changing the GUI due to an unsatisfactory design choice. Then, program the GUI components in Java and edit the size of the components (frame, buttons and panels) accordingly.

After implementing the GUI in Java, the functions/methods for each interactable object should then be implemented according to instructions in ICA1. Before implementing the operations of each button, I have noticed that users may sometimes forget to click 'Review Order' after placing an order with 'Place Order', making their order not saved in the order database (orderlist.txt). Thus, I have decided to combine both operations of 'Place Order' and 'Review Order' into just 'Place Order' for a more user-friendly experience.

However due to time constraints, I was not able to implement many mouse events into the program that are requested in ICA1. These mouse events include interactions with the panel image allowing for order editing or displaying order data, depending on the type of click (right, left or centre click). Moreover, there are many lines of repeated code for appending the correct images into the nine panels of the GUI, this can easily be remedied by making an array list containing the nine panels and iterating accordingly, following the appropriate if statements, as well as implementing a method that can be called when the need to append images is needed.