|  |  |
| --- | --- |
| **Assignment Case** | Diagram  Description automatically generated |
| COMP6122 | COMP6122001  Framework Layer Architecture |
| **Computer Science** | **O233-COMP6122-AW05-01** |
| ***Valid on*** *Odd Semester Year 2022/2023* | **Revision 00** |

1. Seluruh mahasiswa tidak diperkenankan untuk:

*All students are not allowed to:*

* + - Berdiskusi dan/atau bekerja sama dengan mahasiswa lainnya,

*Discuss and/or work together with other student participants,*

* + - Melihat sebagian atau seluruh jawaban mahasiswa lain,

*Seeing a part or the whole answer from another student,*

* + - Membuka dan menyalin dari **BUKU** atau **CATATAN**, **VIDEO** dari pengajar (recording kelas, VBL, Youtube, dsb) dan **REFERENSI** lainnya,

*Open and copy from any resources such as notes, videos (class recording, VBL, Youtube, etc) and other references,*

* + - Membuka dan menyalin jawaban dari internet (google, stackoverflow, dsb),

*Open and copy answer from the internet (google, stackoverflow, etc),*

* + - Mengerjakan soal yang tidak sesuai dengan tema yang ada di soal,

*Working with another theme which is not in accordance with the existing theme in the matter of the case,*

* + - Melakukan tindakan kecurangan lainnya,

*Committing other dishonest actions,*

* + - Secara sengaja maupun tidak sengaja melakukan segala tindakan kelalaian yang menyebabkan hasil karyanya berhasil dicontek oleh orang lain / kelompok lain.

*Accidentally or intentionally conduct any failure action that cause the results of the project was copied by someone else / other groups.*

1. Jika mahasiswa terbukti melakukan tindakan seperti yang dijelaskan butir 1 di atas, maka **nilai mahasiswa** yang melakukan kecurangan (menyontek maupun dicontek) akan di – **NOL** – kan.

*If the student is proved to the actions described in point 1 above, the score of the student which committed dishonest acts (cheating or being cheated) will be “Zero”.*

1. Perhatikan jadwal pengumpulan jawaban, segala jenis pengumpulan jawaban di luar jadwal tidak dilayani.

*Pay attention to the submission schedule, all kinds of submission outside the schedule will not be accepted.*

1. Bila Anda tidak membaca peraturan ini, maka Anda dianggap telah membaca dan menyetujuinya

*If you have missed to read these regulations, so you are considered to have read and agreed on it.*

1. Persentase penilaiaan untuk matakuliah ini adalah sebagai berikut:

*Marking percentage for this subject is described as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| 100% | - | - |

1. Software yang digunakan pada matakuliah ini adalah sebagai berikut:

*Software will be used in this subject are described as follows:*

|  |
| --- |
| **Software**  *Software* |
| Eclipse 2020.6  Java 11 |

1. Ekstensi file yang harus disertakan dalam pengumpulan tugas mandiri, proyek, dan uap untuk matakuliah ini adalah sebagai berikut:

*File extensions should be included in assignment, project, and final exam collection for this subject are described as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| JAVA, CLASS | - | - |

## Soal

*Case*

**stAWbucks**

**Criteria**:

1. Builder

First you need **Builder Pattern,** Builder is a pattern that allows you to construct a complex or complicated objects using the same construction code. This pattern allows user to create multiple types of an object without creating multiple sub classes.

1. Singleton

Next you need to create **Singleton Pattern,** Singleton is pattern that restrict the instantiation of a class to one object. this pattern makes it possible to user does not conflict when call the instance.

1. Adapter

Last you need **Adapter pattern**. Adapter pattern is structural design pattern for convert the interface of a class into another interface clients expect. Adapter lets classes work together that couldn’t otherwise because of incompatible interfaces.

**stAWbucks** is a well-known coffee company that have hundreds of stores throughout Indonesia. The company has decided to modernize their order management system and they have hired you to create a program which will makes the task easier for their barista. The program criteria will be created with the following criteria:

1. **Main Menu**

Chart

Description automatically generated with medium confidenceAt first, the program will show the main menu. In this menu, the player can choose whether they want to **Brew Coffee, View Brewed Coffee, and Exit.**

Figure 1. stAWbucks Menu

1. **Brew Coffee**

If the user chooses **Brew Coffee (Option 1)**, prompt the user to input following coffee details:

* + - **Cup Size**

A picture containing graphical user interface

Description automatically generatedThere are 3 cup sizes in **stAWbucks.** The size **Tall** is **12 ounces**, **Grande** is **16 ounces**, and **Venti** is **20 ounces**. Display the cup type and its corresponding size in ounces (oz.). Then, validate that the user can only input **tall, grande, or venti.** The validation should be **case insensitive**.

Figure 2. Cup Size Input

* + - **Milk Type**

Prompt the user to input the milk type. The user can only input between **whole, almond, or soy.** The validation should be **case insensitive.**

Figure 3. Milk Type Input

* + - **Coffee Shots**

A picture containing chart

Description automatically generatedPrompt the user to input the amount of coffee shots. Each coffee shots are **1.5 oz**. The customer can choose between **1 or 2 shots** only (**inclusive**).

Figure 4. Coffee Shots Input

* + - **Syrup**

The user has the option of adding syrup to the coffee. First, ask the user whether they want to add syrup or not. The input should only be between **Y** or **N (Case sensitive).**

Figure 5. Syrup Option Input

If the user chooses **N,** then proceed to the **confirmation.** Else, if the user chooses **Y,** then ask the user of which type of syrup to add. The user can choose between **Vanilla, Hazelnut,** or **Caramel.** The validation should be **case insensitive.**



Figure 6. Choose Syrup Type Input

Then, prompt the user to choose how many **pump(s)** of selected syrup. Each syrup is **1 ounce (oz) per pump**. The user can only choose between **1 and 3 (inclusive)**.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 7. Syrup Amount Input

* + - **Confirmation**

Last, ask the user for confirmation. The user can only input between **yes** or **no.** The validation should be **case insensitive.**



Figure 8. Confirmation Input

If the user chooses **Yes,** then **brew the coffee** and add the coffee to the **coffee storage.** If the user chooses **No,** then return to the **Main Menu.**

1. **View Brewed Coffee**

If the user chooses to **View Brewed Coffee,** then validate whether the **coffee storage** is not empty. If the **coffee storage** is empty, then show the following message.

Background pattern

Description automatically generated with low confidence

Figure 9. Coffee Storage Empty Message

If there are **brewed coffee** in the **coffee storage.** Then, prompt the user to choose between **Imperial (oz)** or **Metric (ml)** unit to view the brewed coffee. The validation should be **case insensitive.**

Figure 10. Unit input

Calculate the coffee detail using the following formula:

Oz

1 Oz = 29.574 Ml

Then, show the following details, such as coffee **cup size** in **fluid unit (oz / ml), shots amount** in **fluid unit (oz / ml), milk type, syrup type** and **amount** in **fluid unit** if available **(oz / ml).**

A picture containing background pattern

Description automatically generated

Figure 11. View Brewed Coffee (Imperial)

A picture containing graphical user interface

Description automatically generated

Figure 12. View Brewed Coffee (Metric)

1. Exit

If user choose **Exit**, then show the following message and terminate the program.

Background pattern

Description automatically generated with low confidence

Figure 13. Exit Menu Message

***If there are any problems, please ask your assistant!***

Here are the **rules** that you must follow to create your project:

1. Use **appropriate software** for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya.
2. Use the **techniques taught during practicum**
3. Collect **appropriate files** for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya.
4. Include the **other files** that can support your project, such as:
   * All files in your project.
   * Other files (image, audio, video, etc.) used in your project.

**If you do not understand, please ask your assistant! Do not make your own assumption!**