**COURSEWORK SUBMISSION FORM**

| **STUDENT USE** | | **STAFF USE** | |
| --- | --- | --- | --- |
| Module Name | Web Applications Development | First Marker’s  (acts as signature) |  |
| Module Code | 5COSC017C | Second Marker’s  (acts as signature) |  |
| Lecturer Name | Isomiddin Abdunabiev | Agreed Mark |  |
| UoW Student IDs | 18150193/1 | **For Registrar’s office use only (hard copy submission)** | |
| WIUT Student IDs | 00010989 |
| Deadline date | 7/04/2023 |
| Assignment Type | 🗌Group🗌Individual |

**SUBMISSION INSTRUCTIONS**

**COURSEWORKS *must* be submitted in *both* HARD COPY (to the Registrar’s Office) *and* ELECTRONIC unless instructed otherwise.**

For hardcopy submission instructions refer to: <http://intranet.wiut.uz/Shared%20Documents/Forms/AllItems.aspx> - Coursework hard copy submission instructions.doc

For online submission instructions refer to: <http://intranet.wiut.uz/Shared%20Documents/Forms/AllItems.aspx> - Coursework online submission instructions.doc

| **MARKERS FEEDBACK (Continued on the next page)** |
| --- |
|  |

**Content page**

GIT Repo \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2

Development Plans \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 3

SRP, DRY \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 10

Design Patterns \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 13

Demo \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 17

Reflection \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 18

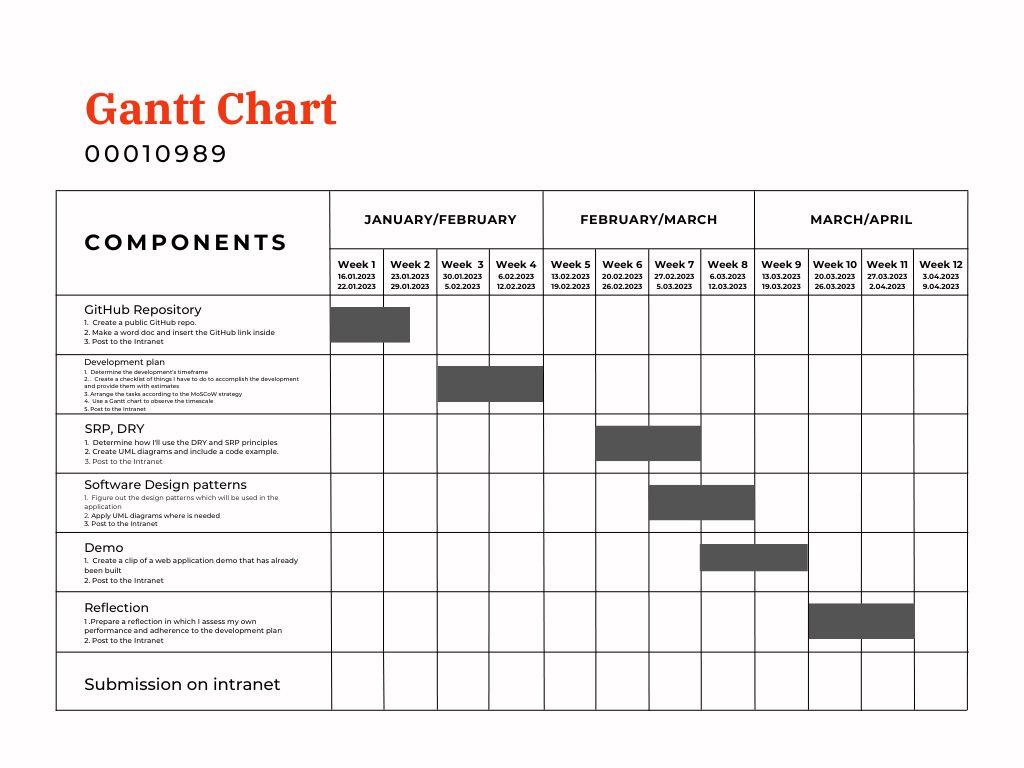
Final Submission References: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 24

GIT Repo

This work belongs to a student with the ID: 00010989. GitHub Repository was created for the successful completion of the course work on the subject: Web Applications Development. All further work will be done independently by the student himself in compliance with all the rules and requirements regarding the course work. Deadline of this component is 31/01/2023.

Link to the GitHub Repository: [*https://github.com/JSDeveloper545/WAD-00010989.git*](https://github.com/JSDeveloper545/WAD-00010989.git)

Development Plan

****

**If this image is not visible, please check the following link:**

[*https://www.canva.com/design/DAFaJt-PUpc/018\_wHrhGP403hbzfhzAnw/edit?utm\_content=DAFaJt-PUpc&utm\_campaign=designshare&utm\_medium=link2&utm\_source=sharebutton*](https://www.canva.com/design/DAFaJt-PUpc/018_wHrhGP403hbzfhzAnw/edit?utm_content=DAFaJt-PUpc&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)

**Task list**

* GitHub Repository

1. Create a public GitHub repo.

2. Make a word doc and insert the GitHub link inside

3. Post to the Intranet

* Development plan

1. Determine the development's timeframe

2. . Create a checklist of things I have to do to accomplish the development and provide them with estimates

3. Arrange the tasks according to the MoSCoW strategy

4. Use a Gantt chart to observe the timescale

5. Post to the Intranet

* SRP, DRY

1. Determine how I'll use the DRY and SRP principles

2. Create UML diagrams and include a code example

3. Post to the Intranet

* Software Design patterns

1. Figure out the design patterns which will be used in the application

2. Apply UML diagrams where is needed

3. Post to the Intranet

* Demo

1. Create a clip of a web application demo that has already been built

2. Post to the Intranet

* Reflection

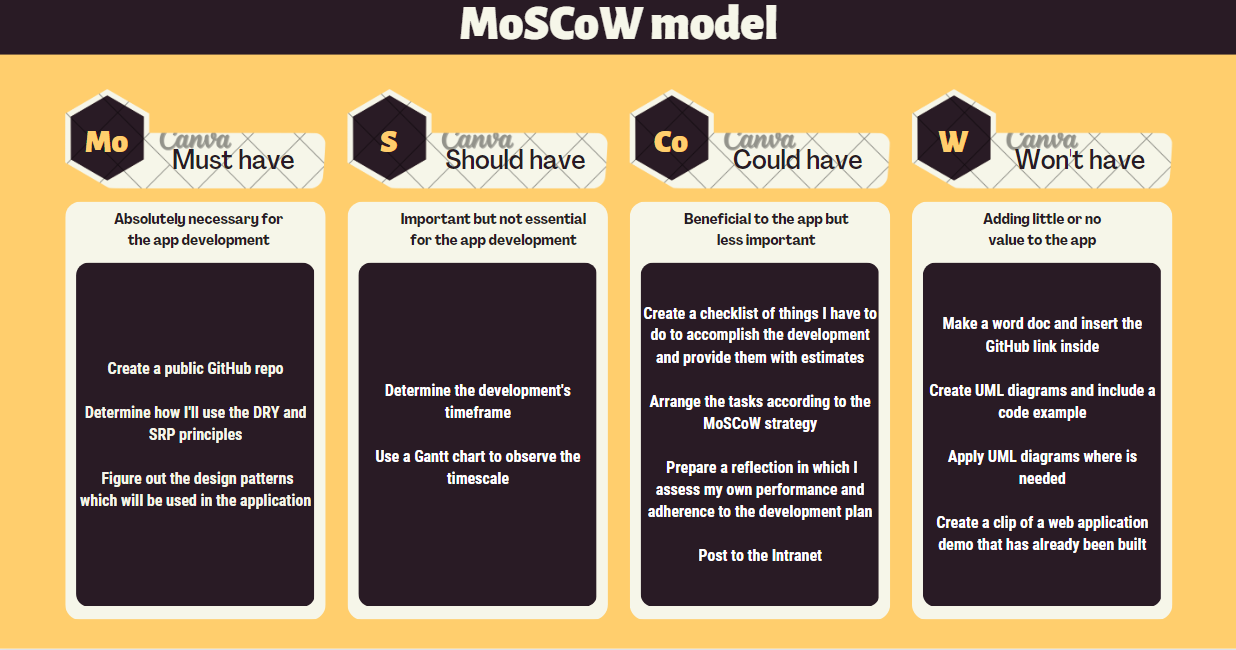
1. Prepare a reflection in which I assess my own performance and adherence to the development plan

2. Submit on Intranet

* Submission on Intranet

**Estimations**

| **Components** | **January** | **February** | **March** | **April** |
| --- | --- | --- | --- | --- |
| GitHub Repository | 16.01.2023  26.01.2023  **(10 days)** |  |  |  |
| Development plan |  | 5.02.2023  12.02.2023  **(7 days)** |  |  |
| SRP, DRY |  | 20.02.2023  27.02.2023  **(7 days)** |  |  |
| Software Design patterns |  |  | 5.03.2023  12.03.2023  **(7 days)** |  |
| Demo |  |  | 12.03.2023  19.03.2023  **(5 days)** |  |
| Reflection |  |  | 20.03.2023  27.03.2023  **(7 days)** |  |
| Submission on intranet |  |  |  |  |



### SRP, DRY

**//All classes are public (private is due to Turnitin). Please don`t pay attention to this (to the private naming)**

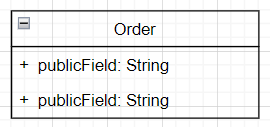
**SRP code example**

I will apply this principle by splitting the functionality of one class in to two classes if I`ll have 2 reasons to change for this class

**DRY code example**

I will apply the DRY Principle in my application in case if I'll have some piece of logic represented twice. I`ll remove this piece of logic and re-write the code immediately

**Without**

****

public class Order

{

private string Weight{ get; set; }

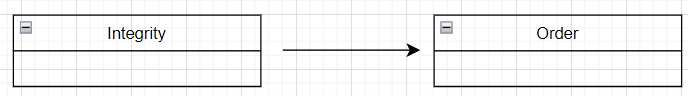
private override string ToString()

{

*/\*Some Logic\*/*

}

}

00

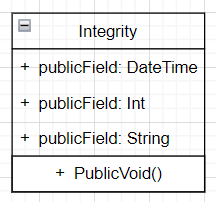
public class Integrity : Order

{

*/\*Some Logic\*/*

}

**With**



namespace Delivery

{

public class Integrity

{

public DateTime DeliveryTime { get; set; }

public int Price{ get; set; }

public string OrderWeight { get; set; }

public void SendOrderToCourier(TaxiDriver taxidriver)

{

*/\*Some Logic\*/*

}

}

}

Link to GitHub:

<https://github.com/JSDeveloper545/WAD-00010989.git>

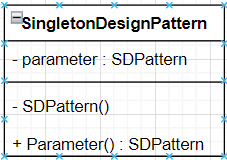
Design Patterns

**From Creational Design Pattern I will use:**

**Singleton**

**This is my own code example:**

<https://docs.google.com/document/d/1v1p596ox2aPhKWXn46uH1nga0xjNmXZMsJUdV6WShRo/edit?usp=sharing>

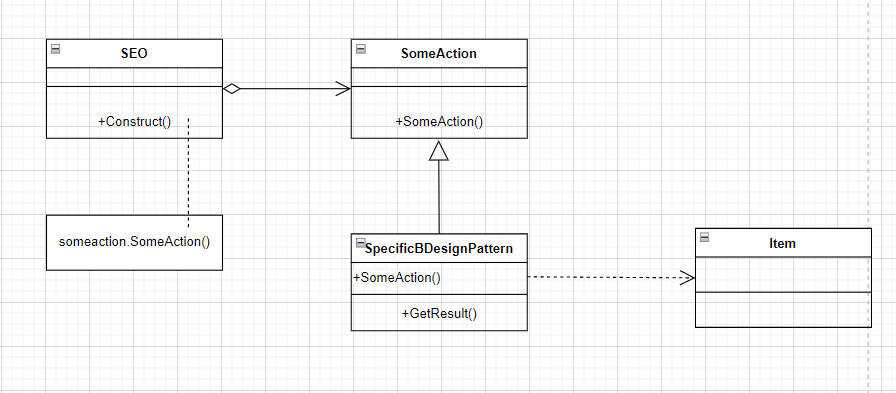
****

\*If image is not visible please make it bigger

**Builder**

**This is my own code example:**

<https://docs.google.com/document/d/165vK71Zlr2cVGc27mMExIhR7IsVrCzi-p2u15p8YTUE/edit?usp=sharing>



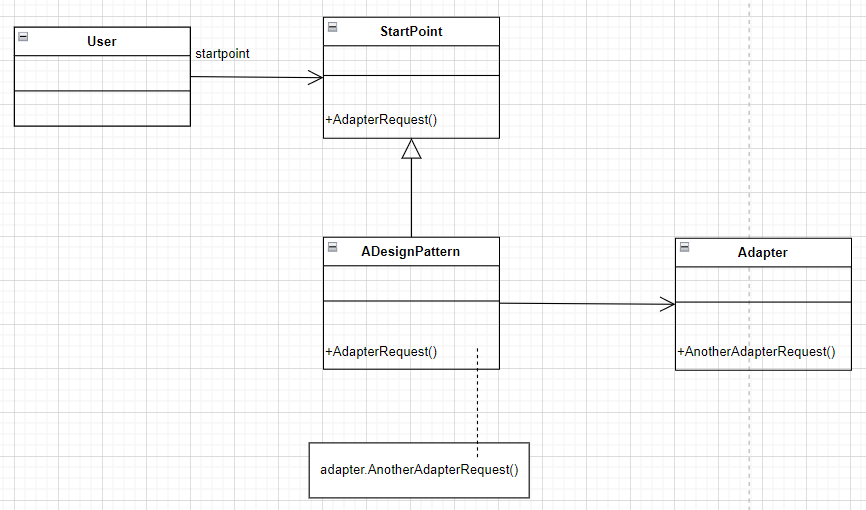
\*If image is not visible please make it bigger

**From Structural Design Pattern I will use:**

**Adapter**

**This is my own code example:**

<https://docs.google.com/document/d/1BM98ylOH8J5WV-5DBRYHxZ_lPFkzFme1wP4eWnow4Dk/edit?usp=sharing>

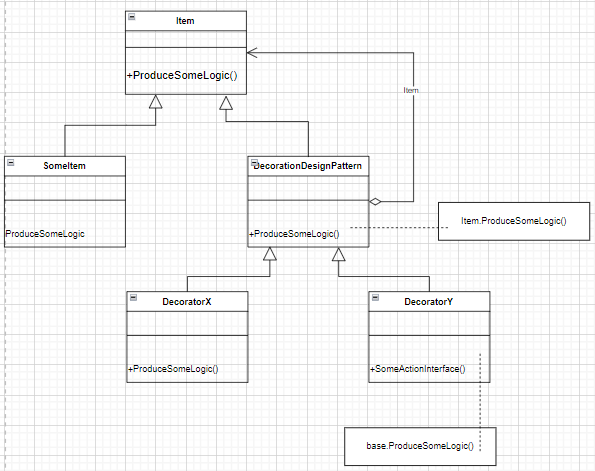
****

\*If image is not visible please make it bigger

**Decorator**

**This is my own code example:**

<https://docs.google.com/document/d/1GWSOwkq-hwZeN2w9OhvLXfjCQgQ1mMsbXnHV-BpPuCg/edit?usp=sharing>

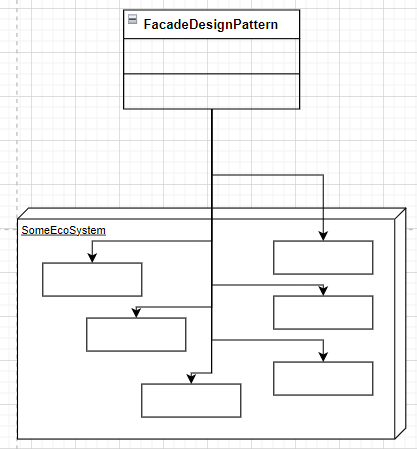
****

\*If image is not visible please make it bigger

**Facade**

**This is my own code example:**

<https://docs.google.com/document/d/1lDP_KYZyzfauQNsQR5ZejHYANiDlGDhCrOWLkk8FII8/edit?usp=sharing>

****

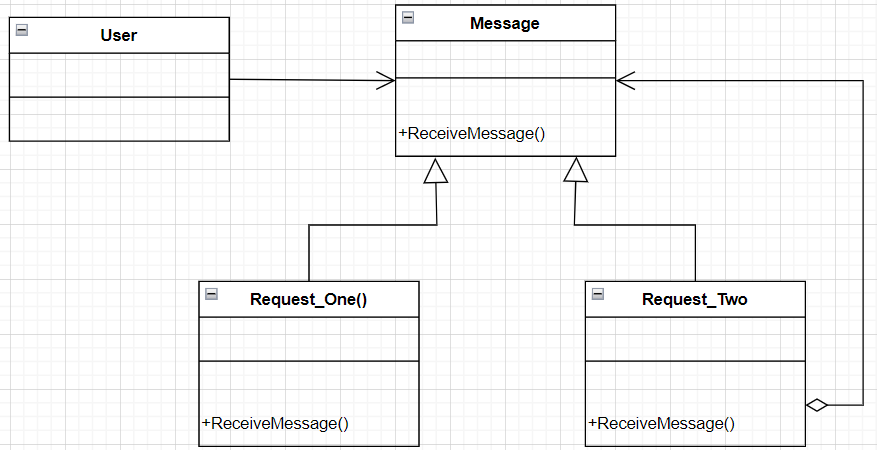
\*If image is not visible please make it bigger

**From Behavioural Design Pattern I will use:**

**Chain of Responsibility**

**This is my own code example:**

<https://docs.google.com/document/d/16iwmlnlecFWxNZahXw9oJa9qgAVjZLXmfE08jGljEDo/edit?usp=sharing>

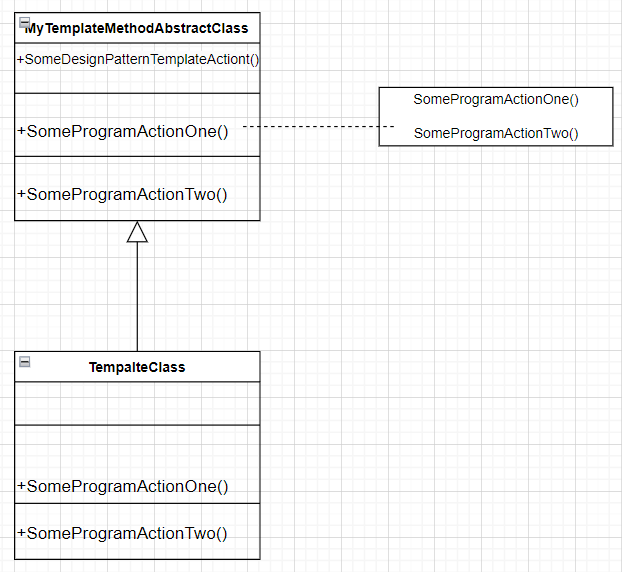
****

\*If image is not visible please make it bigger

**Template method**

**This is my own code example:**

[**https://docs.google.com/document/d/1Kf8e\_zyxNnfh05NXHB2IPwqQwpGtixDT5n0xyZI-R7Q/edit**?**usp=sharing**](https://docs.google.com/document/d/1Kf8e_zyxNnfh05NXHB2IPwqQwpGtixDT5n0xyZI-R7Q/edit?usp=sharing)

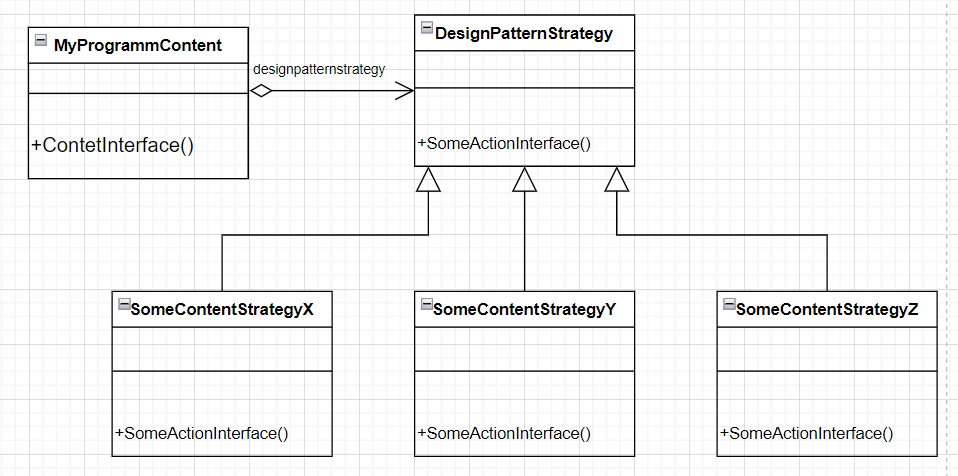
****

\*If image is not visible please make it bigger

**Strategy**

**This is my own code example:**

<https://docs.google.com/document/d/15MMURuNYEDY4tDh_s-2wDeimqpxj1_XkfnMKFk1uPhQ/edit?usp=sharing>

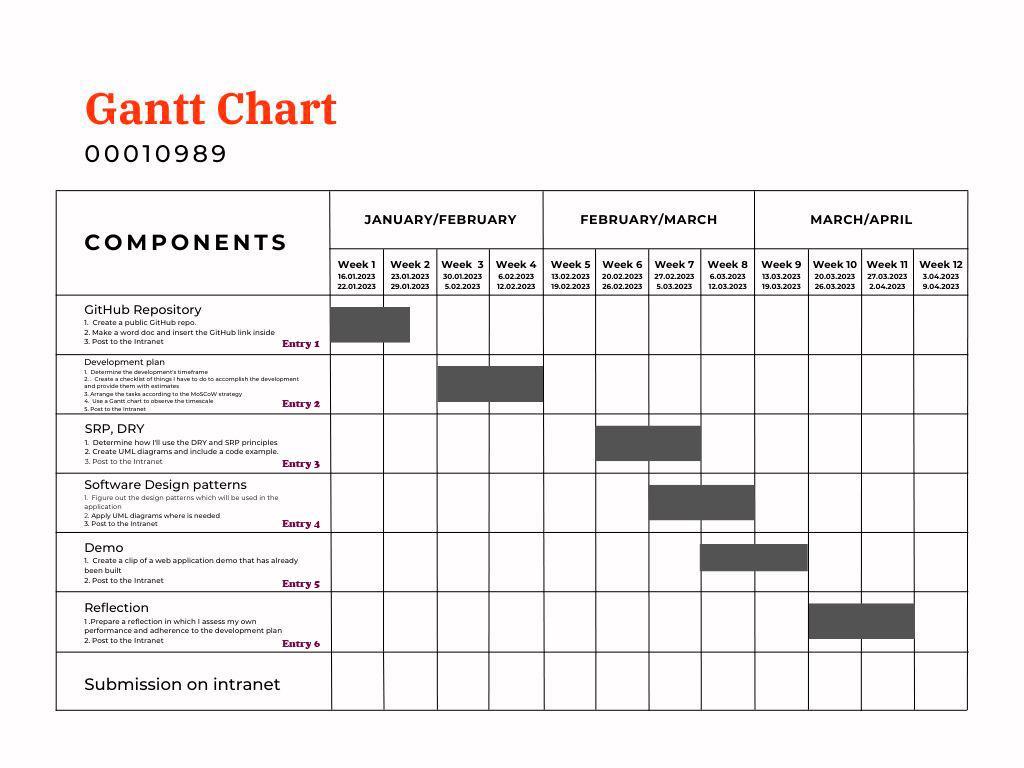
****

\*If image is not visible please make it bigger

### Demo

<https://drive.google.com/file/d/1224ceoobQCvzxQPfIjV5yUwjMMp6U3_b/view?usp=sharing>

Reflection

****

Entry 1. Deadline was 1/31/2023 11:59:59 PM.. My uploaded time was 1/30/2023 4:12:50 PM.

Referring to Gantt Chart, I can say that Entry 1 was my very first and one of the most basic parts of the whole portfolio. The most important aspect for me was to choose the right repository type based on the requirements of the course work and add the necessary files there. The most important problem for me was the password recovery from the GitHub account. Since after its creation I didn't write it down anywhere and later I completely forgot it. I had to restore it through my Google account several times, as the Internet was working very poorly and there was no way to restore the password right away. However, as for the delivery date, I think that I coped with it 100 percent and submitted my work one day before the deadline.

In conclusion, I would like to say that I fully managed this task and was able to deliver this part in the correct manner and at the right time.

Entry 2. Deadline was 2/14/2023 11:59:59 PM. My uploaded time was 2/12/2023 8:05:29 PM.

As for entry two, to be honest, I didn't want to start for a very long time, since the methods described in the requirements were not familiar to me at all. The main aspects for me were the study of the MoSCoW method structure and later the correct compilation of the Gantt Chart. As for the problems, I had two of them. The first is the drawing up of the Gantt Chart by setting tasks by deadlines and creating the MoSCoW method by distributing tasks into groups. As a solution for both cases, YouTube was chosen as the basis by which these problems were further solved. As for the delivery, I exceeded my expectations by handing over the work two days earlier. I think I handled this part even better than I expected.

Entry 3. Deadline was 2/28/2023 11:59:59 PM. My uploaded time was 2/28/2023 5:29:50 PM.

Continuing my reflection, I can note that entry three did not particularly cause me any great difficulties, since those parameters and queries that were listed in the description were familiar to me. We studied these methods at the seminar. The main aspect for me was how I would implement them into my code. As for the problems, I can name only one and this is the correct definition of the place where they need to be implemented. To solve this problem, StackOverflow forum helped me a lot. StackOverflow is a place where people share their tips and solutions to questions asked by other users. As for the day of delivery, I didn't keep up with my plans a little bit because I had other courseworks. However I uploaded my work on the deadline day at 5:29:50PM.

Entry 4. Deadline was 3/14/2023 11:59:59 PM. My uploaded time was 3/11/2023 4:05:48 AM.

Design patterns were very difficult for me and took almost more time than creating the application itself. The main aspect here is the correct choice of the design pattern itself for the necessary needs of the application. There were a lot of problems that I encountered, among which there were questions about how they can be applied, where they need to be applied, what kind and type they can be, where they can be applied, and so on. The solution came by using YouTube as the main service for answering my questions and studying these design patterns. As in the case above, this part of my portfolio again exceeded my expectations and was delivered three days before the deadline.

Entry 5. Deadline was 3/24/2023 11:59:59 PM. My uploaded time was 3/24/2023 7:01:50 PM.

To create a demo version of the application, I had to invest a lot of time and effort. The main aspect here is time, because the functionality of the application should be great, but for me, as a novice developer, it might not be enough. The main problems that I had while working were my inattention . Made a lot of mistakes in CRUD operations by incorrect syntax. I was able to solve these problems by testing the application and looking through the code several times. I have submitted the demo video in a completely different time frame namely on the last day due to problems with my laptop.

Entry 6. Deadline was 3/31/2023 11:59:59 PM. I will upload this entry on 30/03/2023.

The final part is a reflection on my work done so far, the main aspect of which is to evaluate my work by comparing it with the time of delivery and Gantt Chart. The main problem of this entry was to start writing in time, because due to the large volume of courseworks, it was not so easy to do. I solved this problem with a rather banal but very effective method. Having completed other works on time, I was able to work on WAD. Regarding the deadlines, everything is pretty great since I plan to hand over this work on March 29 just one day before the deadline.

Entry 7. Deadline 07.04.2023 11:59:59 PM.

Entry 7 is the head element, since the finished application must be submitted.. The main aspect here is time management, as I said before, because I need to finish everything before the deadline. One of the big problems that I may encounter can be caused due to the design patterns and their implementation into my application. I can solve this problem with the help of C# forums or YouTube. The scheduled time of delivery is March 6

Final Submission References: