Object Oriented Programming in Java

Project Milestones

**Contents**

**1. Introduction 1**

**2. Group Members 1**

**3. Project Documentation**

**4. Milestones**

1. **Introduction**

In this era of technology, there have been several moments which made us question, “How could mankind come up with something like this”. The invention of airplanes was one of these moments, which did appear like the end of technological development but then technology made us struck with its next revolutionizing invention known as Unmanned Arial Vehicle (UAV) commonly known as drones.

As a part of **Java-OOP Project** we are delighted to get to know a little bit more about drones, and build an application which fetches data from Django based web-server and renders it in our app. Our aim is to get this app working using the fundamentals of object-oriented programming in Java and at the same time we are trying to make the app interactive, using buttons and several windows, ensuring that the app does not look cluttered. A Java based tool known as “Swing” is used for front-end development.

Interestingly, the rendered data also contains dynamic data which changes in regular intervals, this makes the application standout in terms of data-rendering. The written code deals with the server restriction, while also being able to authenticate it using basic authentication header. Our application is making sure to not send too many requests, which saves the server from getting abused.

After successfully completing the development, our Team would learn:

1. How to coordinate in a team
2. How to divide tasks ideally
3. How to solve problems together ensuring smooth development
4. How to think of a feature and research together in case we do not have the programming knowledge of this aspect.
5. **Group Members:**

|  |  |
| --- | --- |
| **Responsibilities** | **Name** |
| **Frontend Team** | Mohit, Bilal, Utkarsh |
| **Backend Team** | Andrej, Yun See |
| **Designing Team** | Whole Group |

**\*\*Besides the above listed tasks there used to be a group meeting at least once per week where opinions on the code were asked and improvements were done, this task was completed by each group member to the same extent\*\***

1. **Milestones:**
2. **Things that have been achieved so far:**
3. Visual representation (wire-frame) of the App using Balsamiq

(Discussed collectively in 2 meetings)

1. UML diagram of the back-end

(Andrej and Yun See)

1. Back-end: data can be successfully fetched from the API

(Andrej and Yun See)

1. Front-end: Basic Windows and Team decided to learn tools of swing first

(Mohit, Bilal, Utkarsh)

\*\*Apart from that, a version control of the application has been initialized using a private repository of GitHub and access to the repository is confined to group members only. Various branches are used to develop individual features and then merged to the master branch of the repository \*\*

1. **Thing We are planning to achieve for the next submission:**
2. Front-end: Successfully connecting front-end with back-end and achieving 70% of the designs of the wire-frame.
3. Accessing all the “Drone Dynamics” data and creating return buttons for all windows
4. Back-end: Making the code clean and reviewing it. Solving unwanted exceptions and bringing the back-end product to 90% of the desired product.

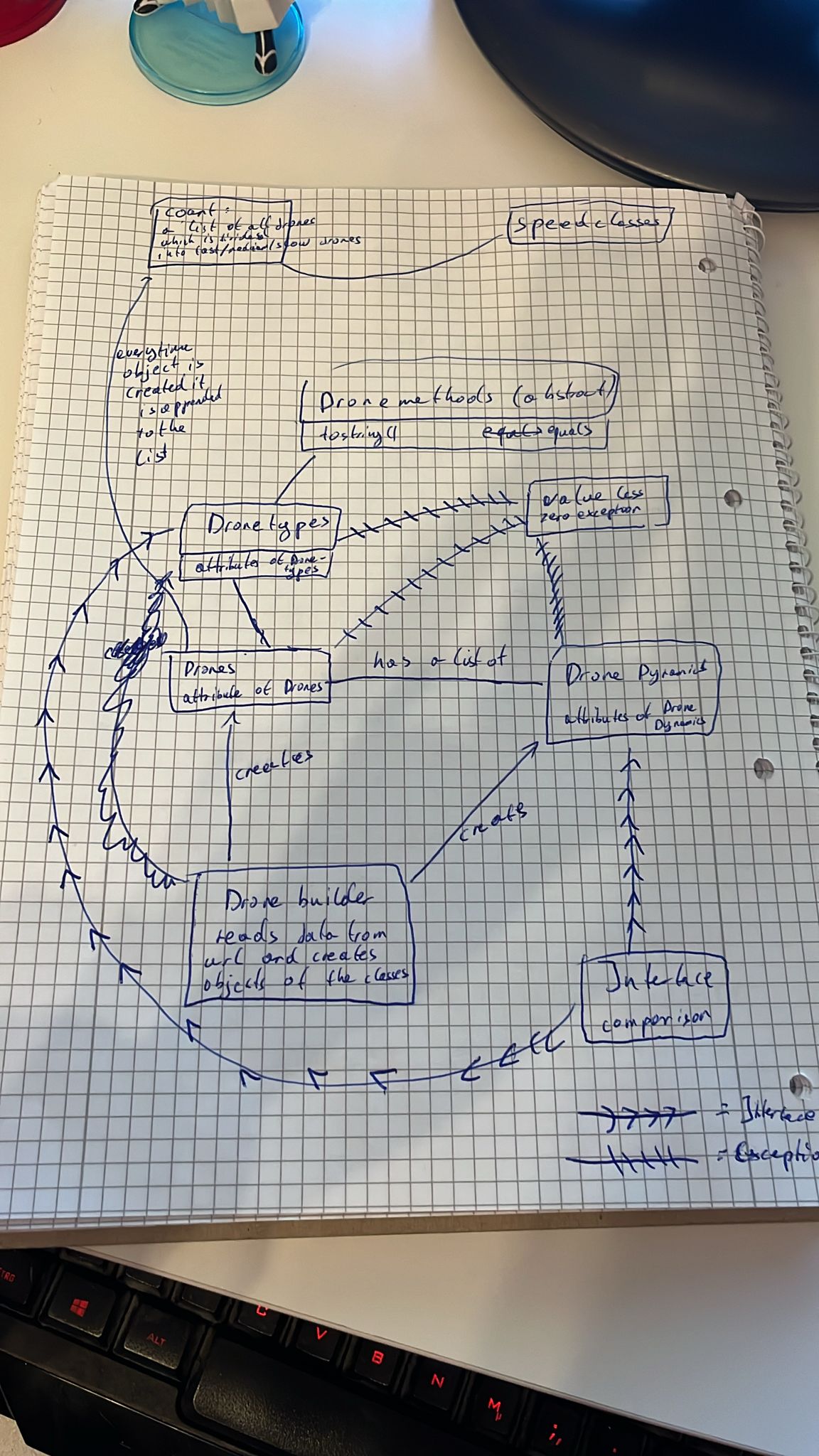
**3.Problems arise during the last milestone phase:**

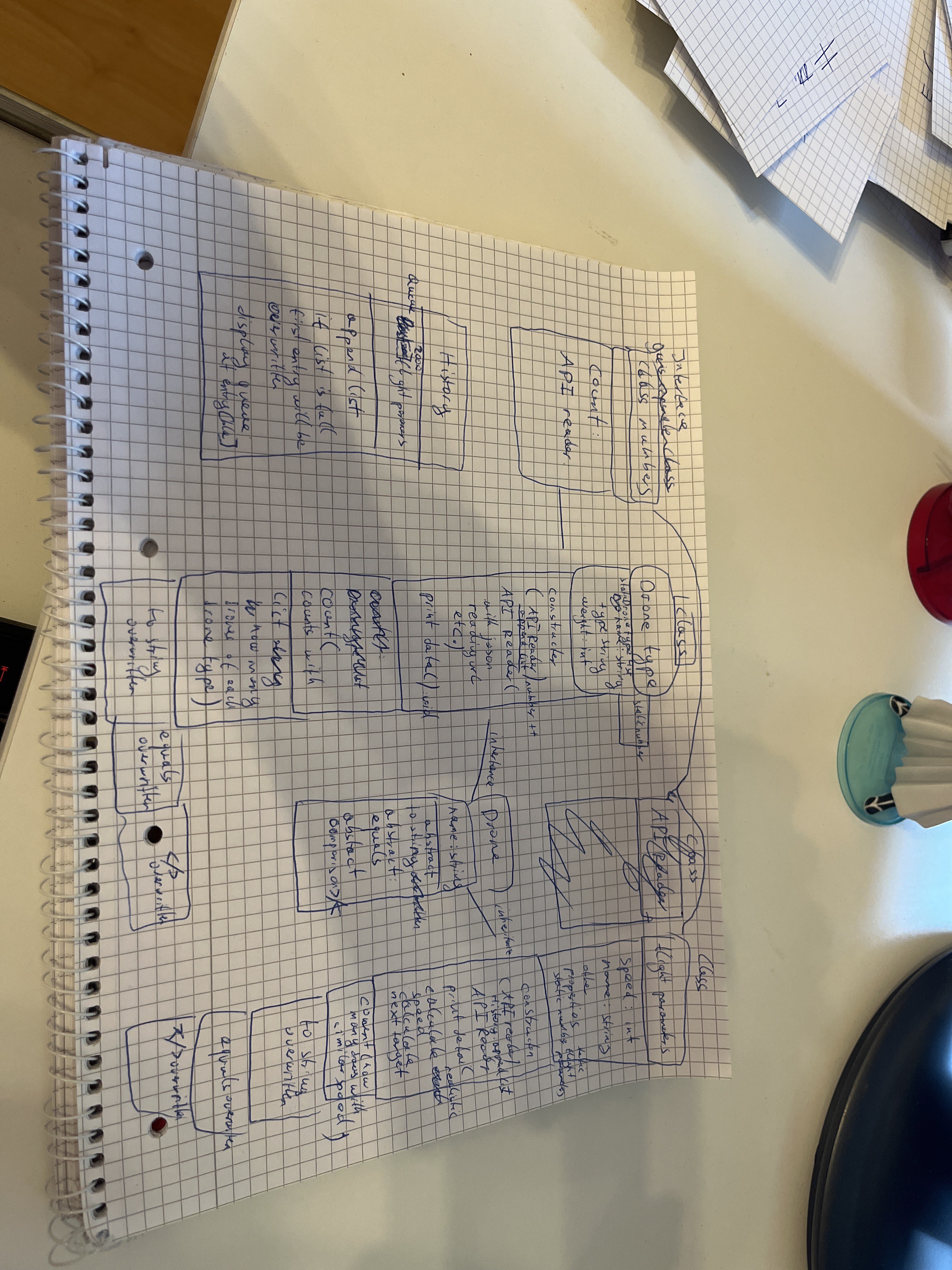
a. We encountered difficulties in gathering data from various web pages, primarily due to our limited knowledge of the correct format for writing query parameters in URLs. Our code consistently succeeded in accessing information from the first page but failed to retrieve data from subsequent pages. This challenge was ultimately overcome by relying on intuition and a deeper examination of the rules governing endpoints, as well as a more comprehensive understanding of the conventions for query parameters and variables within URLs. Unfortunately we did not overcome the problem of accessing all the data from the “Drone Dynamics”.

b. We had programmed buttons in our graphical user interface (GUIs) to facilitate the transition between different windows. However, despite our efforts, the buttons did not function as intended. Although the code was written in accordance with our existing knowledge and seemed correct, the buttons remained unresponsive. This problem was solved by setting the requested window as “visible”, when the button is pressed.

c. We had conflicts in merging Files that was solved manually. Furthermore, we had to grow accustomed to GitHub itself, as most of us had little to no experience with it. Consequently a lot of problems arised when using GitHub’s various commands and features.

\*\*Below are all the referenced diagrams used for development\*\*



****

**(Initial sketch for back-end)**

|  |  |
| --- | --- |
|  | |
|  |  |

A screenshot of a computer screen

Description automatically generated

A table of numbers with a number in the middle

Description automatically generated with medium confidence

