Criterion A: Planning

1. Client and scenario

1.1. Client

Client Mr. XX is a Master student of the grade of electro-mechanical engineering at the XX university. For his master's program, he is doing a master thesis project for the account of X, an engineering company located in X.

1.2. Scenario of the project

My client is working on his thesis project whose **aim** is: "comparing the performance of a conventional and a novel valve capable of generating bubbly flow". This comparison is used for applications such as precise liquid dosing.

Thus, my client is creating an experiment in which he measures several factors to answer his thesis. He is working with bottles with different valves that will be sprayed when pressed upon. His test setup is the following: a user initiates a push on the cap that jets a solution from the bottle into a tube that goes towards a beaker.

Hence, my client is in need of a digital solution to actuate a push on the cap for a certain amount of time; to measure the pressure of the liquid at one point in the tube while the cap is pushed; to measure the differential pressure between two points in the tube; to measure the volume of the solution in the beaker; and to access the data after *n* number of cycles[1]. The "consultation of client #1" in *appendix 1.1* proves this.

2. Proposed solution

2.1. The solution

- Connect a servo motor to an Arduino for the valve to be pressed down.
    - Create a mechanism for the cap to be pressed down
- Connect the necessary sensors to an Arduino and record the measurement.
    - Pressure at one point
    - Differential pressure between two points
    - Discharge rate
    - Volume of liquid
- The measurement would be transmitted to another Arduino by radio transmission.
- The Arduino on the receiving end would send the data received to a computer
- An algorithm using Java would store the measurements in a database using SQLite as the RDBMS
    - Store also information about the cycle and bottle(ID, Date, Cycle time)
- A Java application would be used to access the data from the database to create tables

[1] A cycle refers to the actuation of the cap for a certain amount of time. There are *n* numbers of cycles per bottle

- The application then stores this table on computer

⇒ Solution was finalised with my client in a zoom meeting ("Consultation of client #2" in *Appendix 1.2*).

2.2. Rationale

My first thought when my client told me that I needed to control and measure factors digitally, I immediately thought of working with an Arduino Uno. I could buy cheap and efficient sensors, plug them into the Arduino to record data, and transmit data, via the Serial connection towards a computer. Furthermore, an arduino would allow me to control a servo motor for the actuation mechanism on the valve of the bottle.

In addition, I would use Java to store and process the measurements because it is platform independent, as my client and I are working on computers with different operating systems. Also, Java, being an object-oriented programming language, includes the perks of programming in multiple classes with aspects such as class inheritance.

I will be using SQLite as a RDBMS because SQLite statements are intuitive to use with Java. Furthermore, SQLite transactional which is essential to query data that are ACID-compliant.

Lastly, the radio transmission is necessary in order to limit the wiring between my computer and the Arduino in the testing area.

3. Success criteria

1. Once the application starts, a database should be created with the needed tables. My client will be able to understand its structure if he needs to access it.
2. A user should be able to initiate a push on the valve with the desired time. The valve is successfully pressed down by a mechanism powered by a Servo Motor.
3. The time of the push for the cycle is recorded in a database.
4. The program needs to store the average pressure at a point of each cycle for a bottle in the database (every 0.1s).
5. The program needs to store the volume of the cup at the end of each cycle for a bottle in the database (every 0.1s).
6. The program needs to store the average differential pressure of each cycle for a bottle in the database (every 0.1s).
7. The radio connection between the two Arduinos is successful at transmitting the data from the cycle.
8. A Java algorithm is successful at reading the data from the COM ports
9. Provide an interface to create the needed tables with the correct data. The interphase will be accessed by my client, and he will know how it works.
10. The table can be saved by my client on his computer.

⇒ "Consultation of client #2" in *appendix* for client confirmation