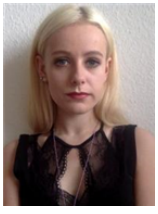
CDIO D2

Emily Skovgaard Rasmussen, s153374
Mathias Tværmose Gleerup, s153120,
Sofie Freja Christensen s153932
Morten V. Christensen s147300
Simon Lundorf s154008
Jonas Larsen, s136335,
Group nr. 15 Class: 02324

Deadline : March 17th 2017

March 17, 2017

Danmarks Tekniske Universitet






Contents

0.1 Git location

https://github.com/MathiasTG/CDIO2.git

1 Formalities

1.1 Abstract

A program has been compiled in java, including the use of the telnet client program Putty. The program is able to

1.2 Foreword

This report takes its foundation in the program (CDIO D2), drawn from the following courses Videregående programmering (02324) and Database (02327). These stretch over a period of 13 weeks, spring of 2017. All of which takes place at DTU campus Lyngby, spring of 2017. The project is made in collaboration between the members of group 15:

- Emily Skovgaard Rasmussen -SWT
- Jonas Larsen - SWT
- Mathias Tværmose Gleerup - SWT
- Morten Velin Christensen - SWT
- Simon Lundorf - SWT
- Sofie Freja Christensen - SWT

2 Process

3 Solution considerations

3.1 Un-implemented features

4 Systemrequirements

4.1 Functional

1. The system has to be executable on the machines in DTU's databars.
2. The system shall use a weight simulator.
   (a) the weight simulator has to appear as a server.
3. A java console program shall be compiled to act as the client.
4. The program shall be able to handle the following:
   (a) Tare
   (b) Brutto control.
   (c) Netto weighing.
5. The system shall be able to register and verify users, by using their ID.
6. One user shall be hardcoded as default user.
   (a) A batch number shall be tied to this user.
      i. Batch ID shall be between the numbers 1000-9999.
      ii. User ID shall be between the numbers 11-99.
7. The result may differ on a maximum of xx.
8. The following commands shall be included:
   (a) S.
   (b) T.
   (c) D.
   (d) DW.
   (e) P111.
   (f) RM20 8.

4.2 Non functional

1. The Weight simulator will use a graphical user interface (GUI).

5 Noun and verb analysis

5.1 Noun

Operatør data
Bruger ID
Batch ID
Klient
Vægt simulator
Vægt kommandoer
Afvejnings procedure
Server

5.2 Verb analysis

Brutto kontrol
Afvejning
Hardcode
Kommunikere
Tarering
Netto vejning
Indtastes
Kvittere
Instruere
Udskrive
Registrere

6 Design

The following section contains the design of the developed program. This is described by the use of a design class diagram (DCD).

6.1 Design Class Diagram (DCD)

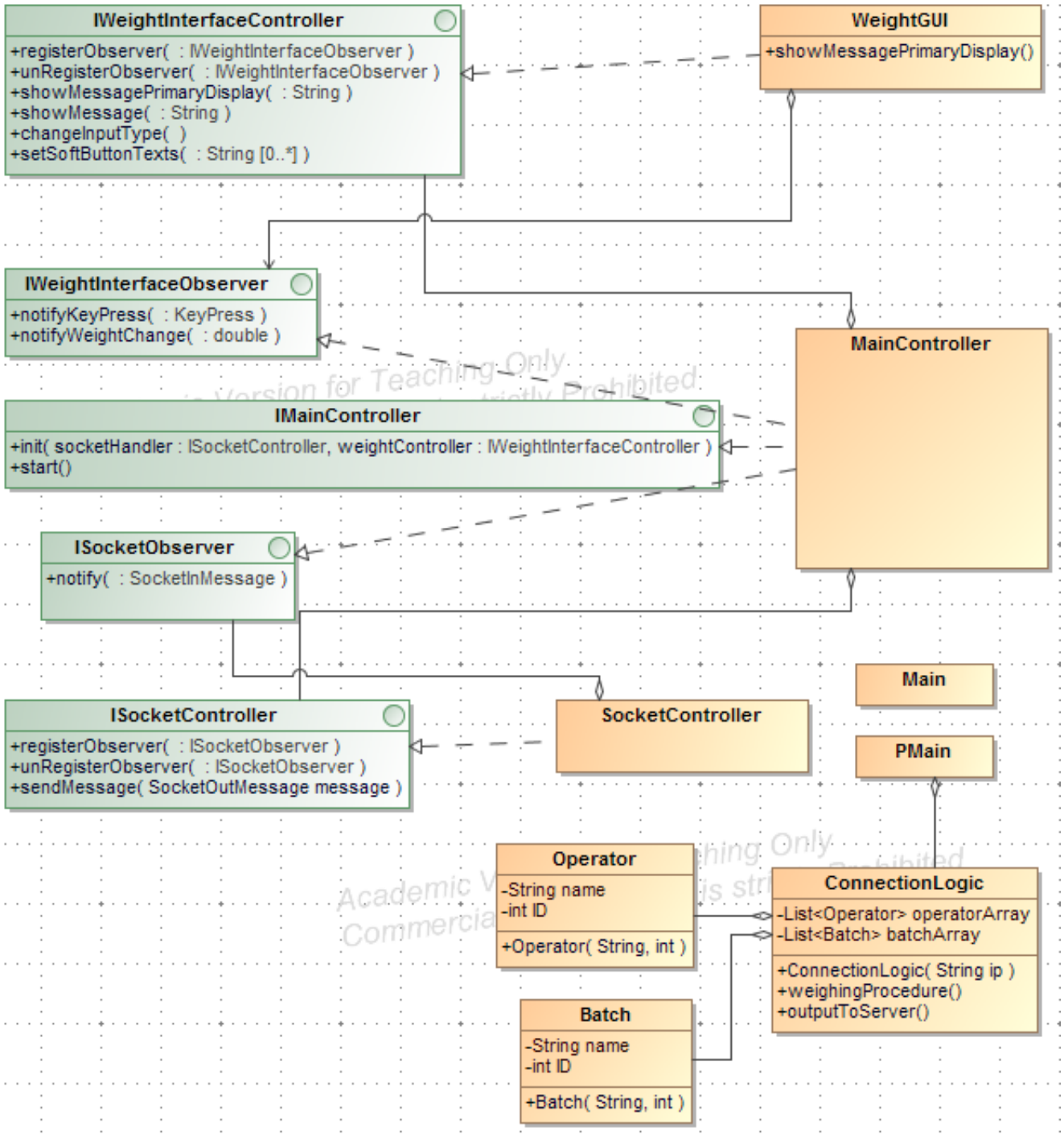


Figure 1: DCD

7 Implementation

The following section contains the implementation part of the project. During the process of development, certain choices have been made, to improve the experience and functionality of the program. Therefore distinctive methods will be explained and reasoned by the use of code examples and descriptive text.

In our process of weighing the matter, we have used socket to create a connection between our server (The weight) and our client (process of weighing the matter).

Our connection is made in The constructor in connectLogic, where we use a string for the IP address. The reason we do that, is to be able to switch between our simulated weight and the actual weight.

The way we made our connection in our constructor, is by making the objects:

Socket clientSocket = new Socket(ip, 8000);
PrintWriter outToServer = new PrintWriter(clientSocket.getOutputStream(), true);
BufferedReader inFromServer = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));

Our socket object is for the connection.

To be able to print text from our server, we use printWriter and to be able to read an input from the program, we use a bufferedReader.

To run our server and client on the same computer, we have to start them separately. That means, starting two different main controllers.

To do that, it is important to start the server first, and then start the client, otherwise it could cause an error.

On the server part, we implemented the following features in the code:

RM20, RM208, D, DW, P111, T, S, B, Q

F is a command we have created ourselves. It simulates that the weight is being removed. It sets the display to show the tare weight and sets curentLoad to tare. Answers with "F A"

Tare button, Softbutton, text: jumped, exit button, if there is an error on the commands, it answers "ES"

8 Test

The following section contains the tests executed throughout the development process. These are included to verify the functionality of the system. The system was tested by creating different scenarios and checking if it executed as expected. At the end of the development process, a user test was executed. This was to see if the system is intuitive and logical for the user to use. Each test will be described, and result noted.

8.1 User test

8.2 Test conclusion

9 Conclusion

We have successfully created a program that can control a weighing on a Mettler weight over its local network.

We did not however end up with a fully functioning weight simulator. The core functionality is almost complete; sending return messages and some buttons. We had a hard time deciphering the program-skeleton provided for us. Trying to understand something someone else had programmed, that was on another level of what we were used to programming wise, and where most of the missing parts were newly taught material that was not fully understood yet, was too big of a challenge. What we should have done better was to take advantage of the resources that was available to us. The person who wrote the code was there to ask, but we did not prioritize asking in time to come out of the CDIO with a fully functional weight simulator. To avoid this in the future, we already made measures in the process of how we begin a new project as a group.

10 Litterature

References

[1] Craig Larman, Applying UML and Patterns 2004.

[2] Lewis and Loftus Java Software solutions 7th ed.

[3] Jonas Larsen, Mathias Tværmose Gleerup, Emily Skovgaard Rasmussen, Morten V. Christensen, Simon Lundorf og Sofie Freja Christensen, CDIO final.

[4] Jonas Larsen, Mathias Tværmose Gleerup, Emily Skovgaard Rasmussen, Morten V. Christensen, Simon Lundorf og Sofie Freja Christensen, CDIO D1.