

Java Development

a) Command Line Tool

There are cases where a bloated software with a GUI is not always needed. Rather, a software with a text based user interface (TUI) is required. A good java library capable of implementing a TUI similar to the famous command line tool midnight commander is Lanterna. It offers classes and interfaces for realizing both a TUI and a classical command line tool. A TUI might be suitable for a server application so that administrators can access the server or for client software which communicates with a server that runs only in the terminal.

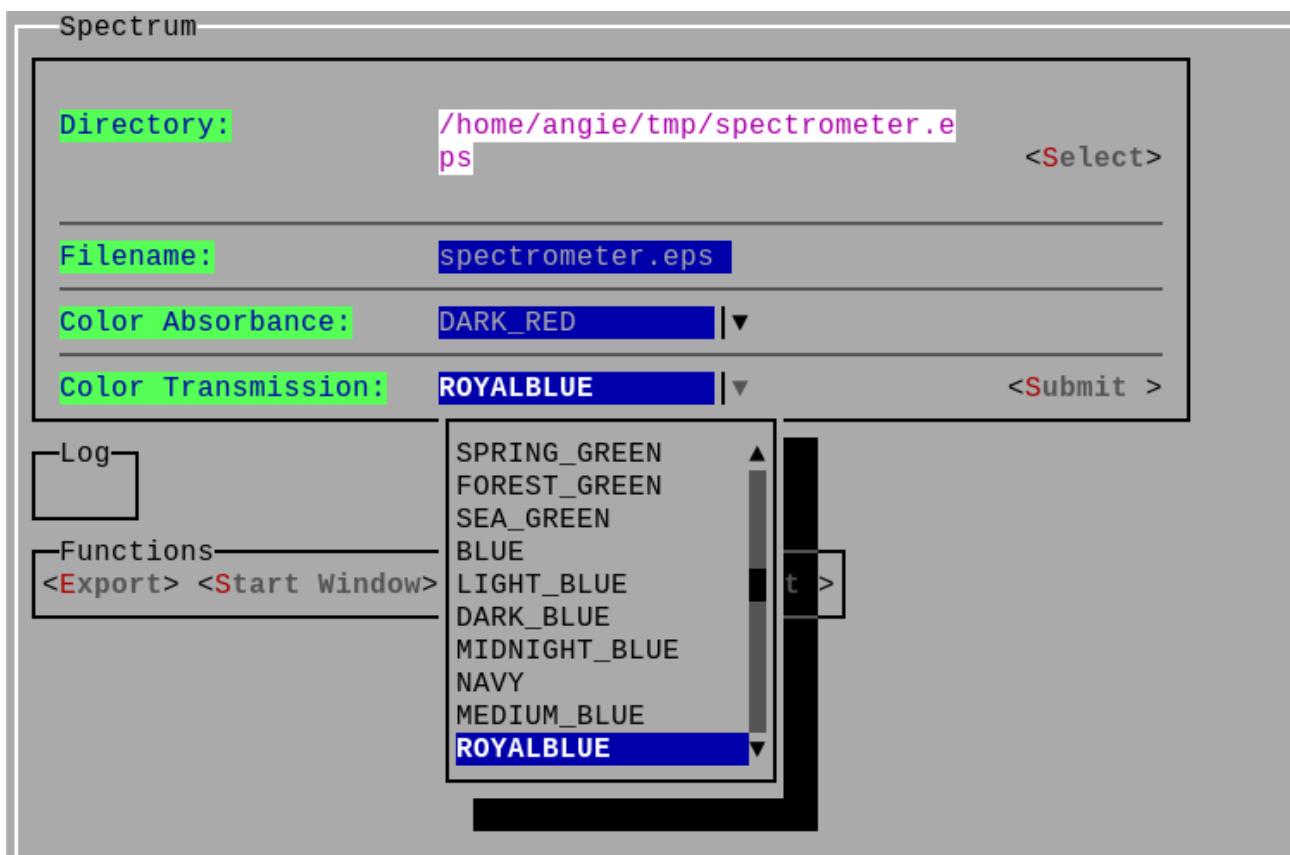


Figure 1: Implementing a plotting software with Lanterna as the frontend

As you can see in figure 1 Lanterna offers a wide range of ui elements such as buttons, text input fields, dropdown menus, labels, different colors, check boxes, grid panels, etc.

b) JavaFX

JavaFX should be taken into account if a software with a GUI is needed. Of all the java GUI frameworks out there, it is the one that is the most advanced. It allows for a liquid layout which can be created using the Grid Pane as the main container. Its styling is similar to a website as it works together with CSS scripts.

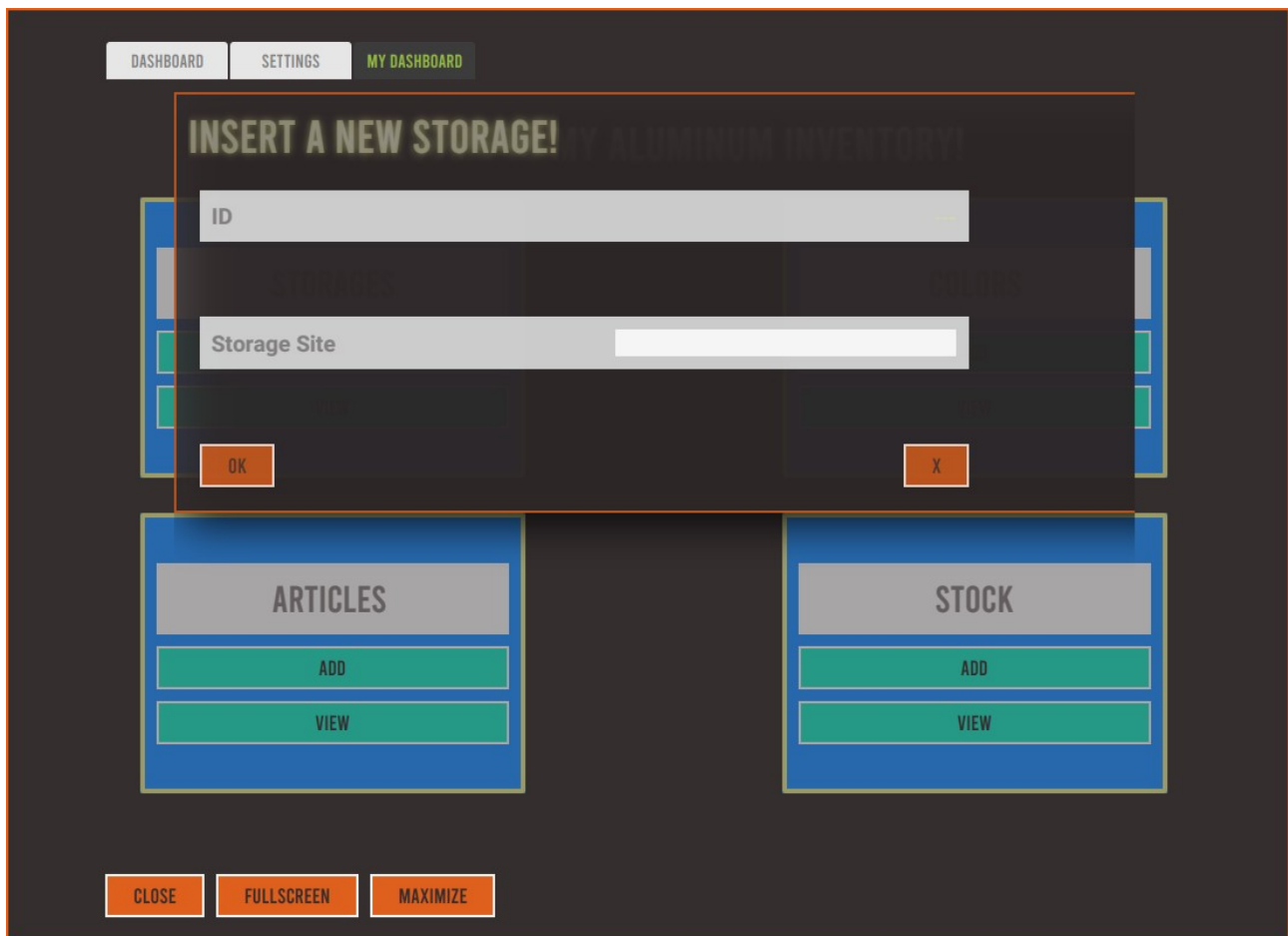


Figure II: Faded in popup window under JavaFX; Tiles on the dashboard

I implemented my own database application referred to as **MyDbApp**, which works with either an **SQLite, MySQL, MariaDB or PostgreSQL database**. Its main purpose is to allow for database editing in a convenient way by offering a modern GUI. It's however a customized application, because each database is different. As a result I'll need to adjust MyDbApp to your database. The view of a database entity such as an articles table can be reached by clicking on the corresponding link under your dashboard.

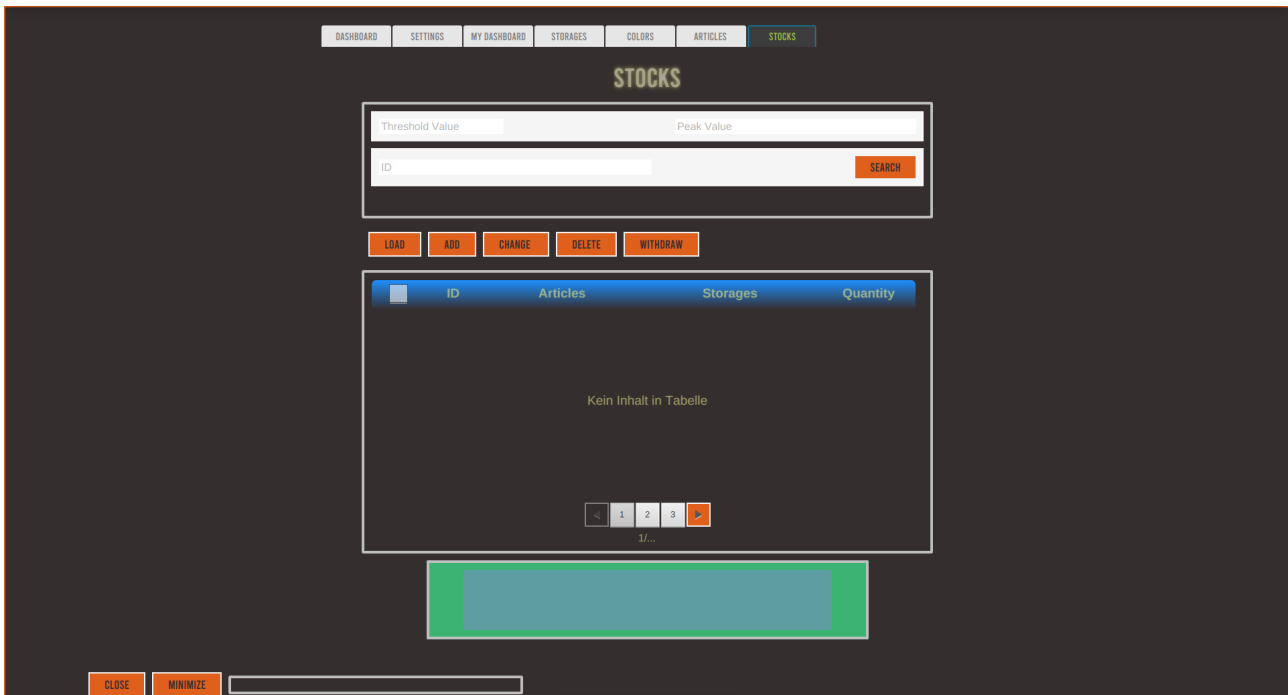


Figure III: Main View of a database table

Figure III depicts the view of a database table. If the table contains entries then you'll be able to click through the pages in which each entry has a checkbox at the beginning. By clicking on the checkbox you can delete several items at once. The progress of deletions can be observed by looking at the animated progress bar. I also provide a search form which lets you filter out specific entries of a whole table.

What's more, there are lots of JavaFX plugins out there which can be used for a JavaFX application such as implementing your own calendar view for managing the appointments of your patients, etc.

c) Features

A java application can become pretty outdated after a year or so. It's because newer versions of libraries have come out or functions and classes have become deprecated respectively. As a result, the java application needs to be recompiled.

What's more, I can add new features or extend a java library. I once extended a library used on a raspberry pi for measuring air pressure. So I added several columns to a database table in which my calculations were stored. The stored values could be seen on a dashboard of a website.

d) Libraries

d.i. Configuration

There are java applications which need a configuration file. This can be realized by using the Apache Commons Configuration Library. The configuration will be stored in

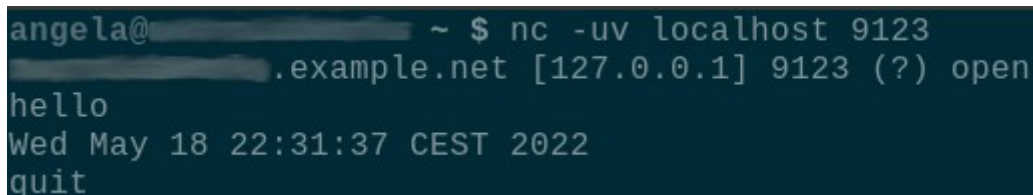
a text file which basically consists of key value pairs. This makes it quite easy to store or retrieve values from any configuration file.

d.ii. Networking

To implement a state of the art network application, i.e. client server application, it's better to use an advanced network library such as Apache Mina. Using Apache Mina lets you focus on the communication between the client and server while Apache Mina is responsible for the low level implementation of your client server application. Basically, for a client and a server you'll just need to override the methods provided by Apache Mina:

- 1) `exceptionCaught`
- 2) `messageReceived`
- 3) `sessionCreated`
- 4) `sessionOpened`
- 5) `sessionClosed`
- 6) `sessionIdle`

Apache Mina can deal with several protocols: TCP, UDP. Its area of expertise is the implementation of a non-blocking server. However, you can use Apache Mina for a blocking server, too.



```
angela@[REDACTED] ~ $ nc -uv localhost 9123
[REDACTED].example.net [127.0.0.1] 9123 (?) open
hello
Wed May 18 22:31:37 CEST 2022
quit
```

Figure IV: Communication with a UDP time server

Apart from different protocols, using Apache Mina lets you implement your very own protocol which is used for communicating between Client and Server. What's more, it offers several built-in protocols such as a text based protocol which can be used for example by realizing a chat server.

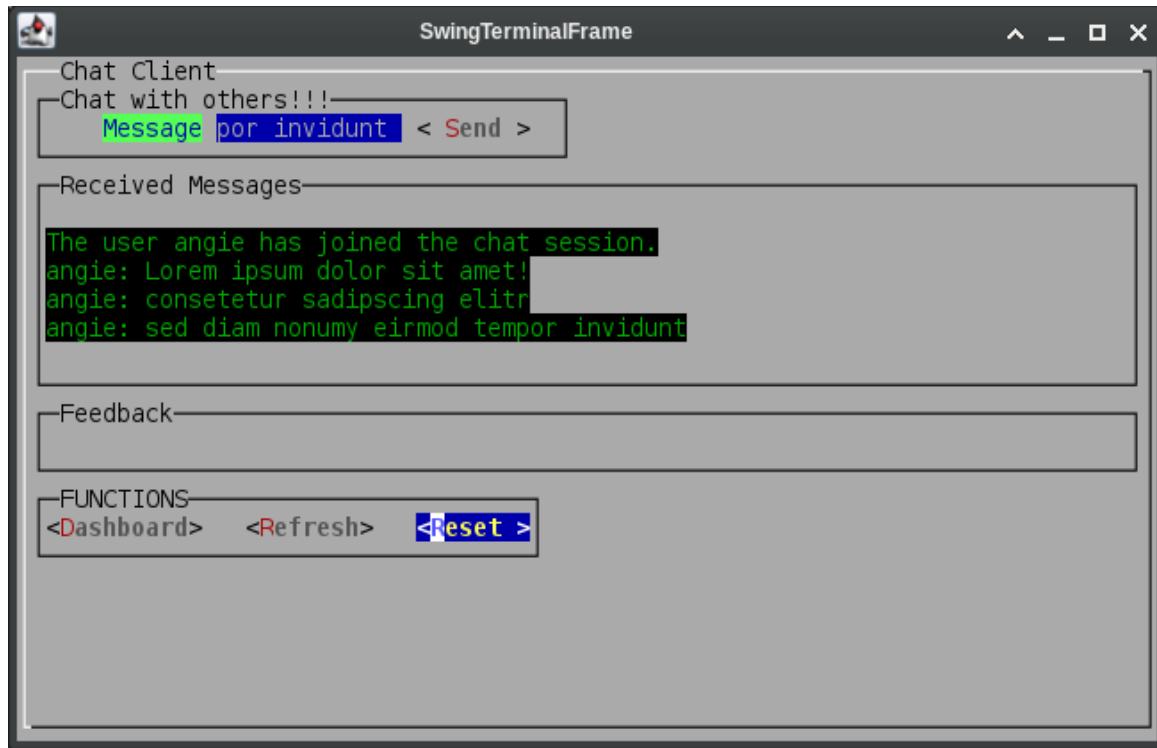
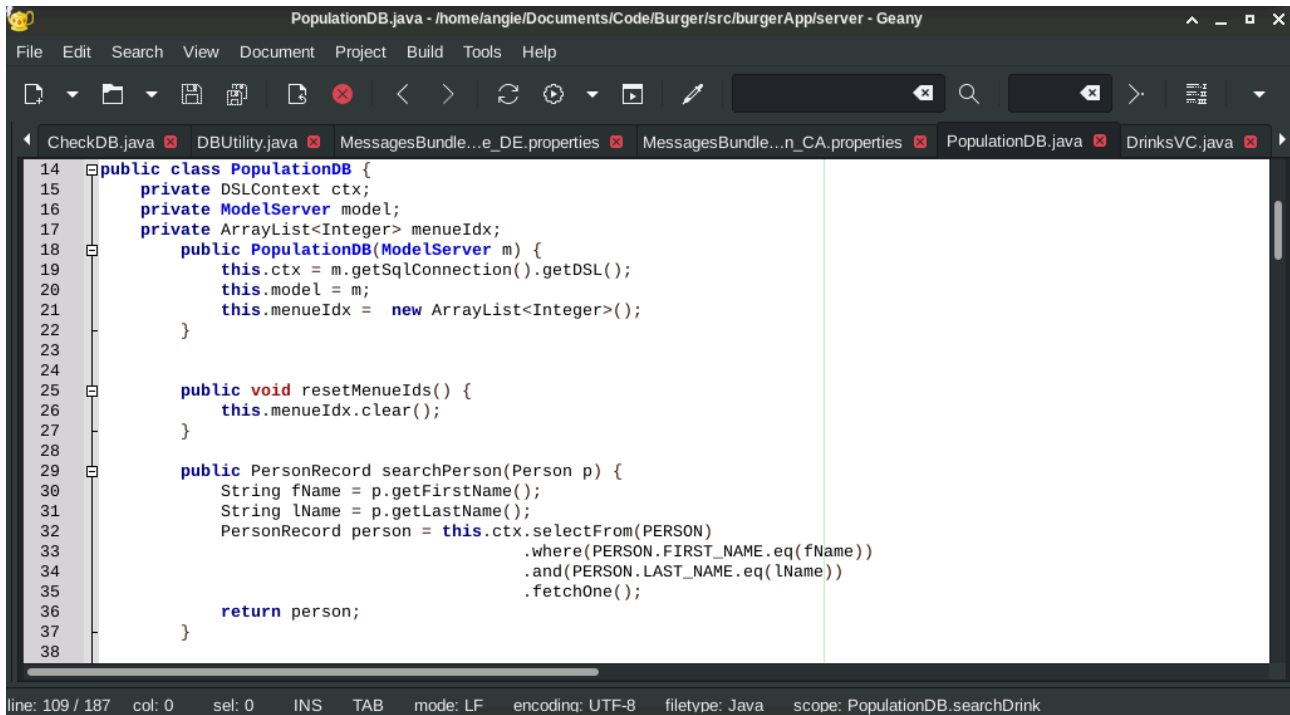


Figure V: Chat client receiving messages from chat server

d.iii. Databases

Editing a database by using string commands is error-prone, as you can save literally everything as a string. A better approach is to make use of the Jooq library which offers a good documentation and makes database editing a lot easier under Java. Jooq lets you connect to your database. Then you'll be able to update your database or view the results of your query.



```
14 public class PopulationDB {
15     private DSLContext ctx;
16     private ModelServer model;
17     private ArrayList<Integer> menuIdx;
18     public PopulationDB(ModelServer m) {
19         this.ctx = m.getSqlConnection().getDSL();
20         this.model = m;
21         this.menuIdx = new ArrayList<Integer>();
22     }
23
24
25     public void resetMenuIds() {
26         this.menuIdx.clear();
27     }
28
29     public PersonRecord searchPerson(Person p) {
30         String fName = p.getFirstName();
31         String lName = p.getLastName();
32         PersonRecord person = this.ctx.selectFrom(PERSON)
33                                     .where(PERSON.FIRST_NAME.eq(fName))
34                                     .and(PERSON.LAST_NAME.eq(lName))
35                                     .fetchOne();
36         return person;
37     }
38 }
```

line: 109 / 187 col: 0 sel: 0 INS TAB mode: LF encoding: UTF-8 filetype: Java scope: PopulationDB.searchDrink

Figure VI: Using a select query under Jooq

d.iv. Plotting / data analysis

There are several java libraries which can be used for plotting or for doing data analysis. For example, **Apache Commons CSV** lets you read in your csv file and store the data to a new csv file. The plotting of your data points can be done by using **JavaPlot** a free library based on gnuplot. JavaPlot can create a data plot, file data plot and a function plot. It lets you choose different colors for the curves and line styles such as points. You can adjust the lines by applying different kinds of algorithms to it such as Csplines. Additionally, you can label the axis, the lines and choose a title for your (multi)plot.

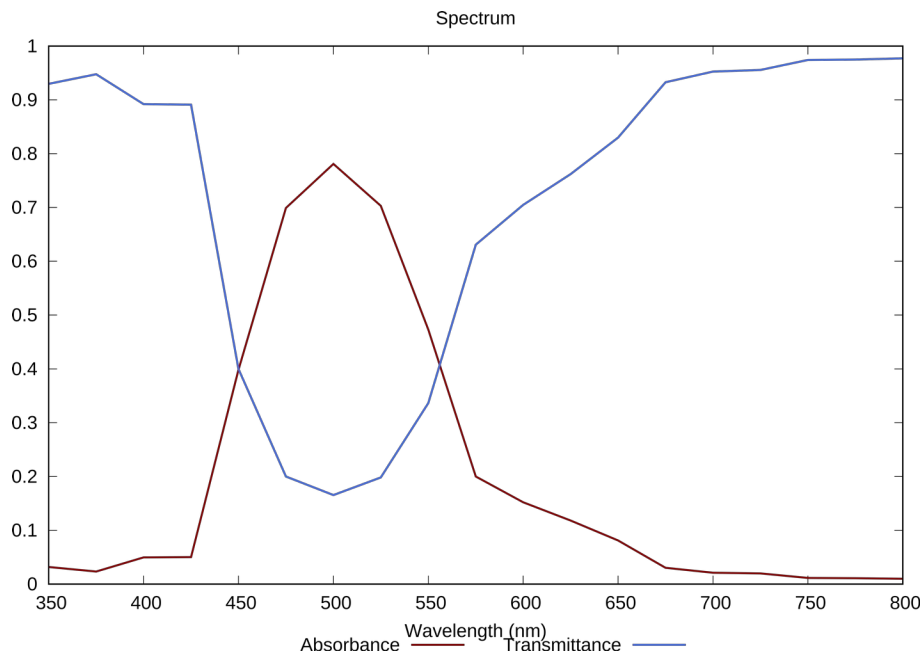


Figure VII: JavaPlot lets you customize your multiplots.

Even though JavaPlot does not offer you the possibility of creating a sigmoid curve, you can stick to the **Apache Commons Mathematics** Library which offers classes for recalculating your data points so that JavaPlot will plot a perfectly looking sigmoid curve. Not only that, Apache Commons Mathematics Library offers a statistics package which lets you calculate the regression line based on your data points or calculate the pearson correlation coefficient.