Architecture

Our software is subdivided into four main packages, each with a specific responsibility. We will now briefly describe each package.

Control Module

The control module is effectively the driving package behind the application. It parses and interprets a script provided by the user to determine which actions should be performed. It then asks the input module to parse the files specified by the user. Then it will ask the analysis module to perform all analysis methods specified in the script on the parsed data. Finally it tells the output module to generate visualizations and formatted files as specified in the script for the analyzed data.

Input Module

The data is defined in the config.xml as:

```
<XML>
        <FILE name=”Creatine” format = ‘text’>
                <PATTERN regex = ‘regex hier’/>
                <COLUMN name=”crea” ignore=”true”/>
                <COLUMN name=”crea_lvl” type=”number”/>
                <COLUMN name=”unit” ignore=”true”/>
                <COLUMN name=”date” type=”date”/>
                alle collommen verder hier
        </FILE>
</XML>
```

For pattern recognition we will use regex defined in the config.xml, we will deliver the software with a custom regex for the data given for this specific assignment. If the user would like to use other data, it will be possible to declare this in the config.xml.

Analysis Module

The analysis module provides important transformations that are commonly needed in sequential data analysis.

Visualisation Module

The visualization module can generate certain visualizations based on the analyzed data and is able to output the data to a file in a format as specified by the user.

rough concept of internal datastructure for files:

```
class file {
	list<record> records;
}

class record {
	hashmap<string, value> columns;
}

string name = record[“Name2”].asString()
```

script