

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN



Data Management for Quantitative Biology

Project Paper

Project 1: Data Modelling and Visualization

SS 2015



Sebastian Goerges
Benjamin Schroeder
Nils-Oliver Schliebs

Abstract

....

1 Background

Data management in life sciences connects different fields and people with different scientific backgrounds. One challenge, especially when data needs to be collected and presented, is to keep track of the metadata that is important for analysis and reproducibility of an experiment, while exposing only informative and easy to understand data and metadata to users that are not concerned with computation or data management. Hence, for a given data set, the project task was to define what parts of data should be shown to end users (life scientists) and to implement a web-based Graphical User Interface, using Java, to visualize given data. Users should also be able to add some annotation on their own and download a summarizing report of the results. In addition, the user interface should be intuitive and well documented to guide even first-time users easily through the task of data annotation.

2 Material and Methods

2.1 Data

The main data structure is provided in form of TSV-files. The file *projects.tsv* contains multiple projects together with several attributes, describing the project. Projects themselves can contain multiple different experiments (*experiments.tsv*), which in turn contain two different types of samples (*QCOFF.tsv* and *QMOUSE.tsv*). *QCOFF* is storing samples of a coffee diversity project, whereas *QCOFF* consists of samples for a mouse knockout project. The set of samples can be structured hierarchically, with patients/organisms at the top (Entity samples) from which tissue/cell samples are derived. Experiments or samples can contain datasets, stored in *datasets.tsv*. Datasets link to one or more files. These files are for instance further descriptions in plain text, images, quality control HTML documents or FASTQ files. All structures of the system contain unique identifiers to connect the set of TSV-files together in a logical way.

2.2 Java and Vaadin

The graphical user interface (GUI) was developed using the open source web application framework Vaadin 7. Vaadin enables the possibility to build single page web apps in server-side Java. All of the browser-server communication and data transfer objects are automated by the framework. The app's state resides on the server, but the end-users use an HTML5 web app in their browsers. In addition, Vaadin's default component set can be extended with custom Google Webtoolkit Widgets (GWT) and designed over cascading style sheets (CSS) [1].

3 Results

3.1 Data Parser

... (Benni ?)

3.2 Project Visualizer

The resulting user interface tries to simplify the provided collection of data. Therefore, an clear navigation has been designed, to enable also new users handling all components of the web application in a fairly easy way. When the tool is started, all existing data is loaded in the background and available projects are displayed in a select-able list. If one choose a particular project, additional project information is shown in a extra table. In addition, a further select-able list appears, displaying all experiments under the project (Fig. 1).

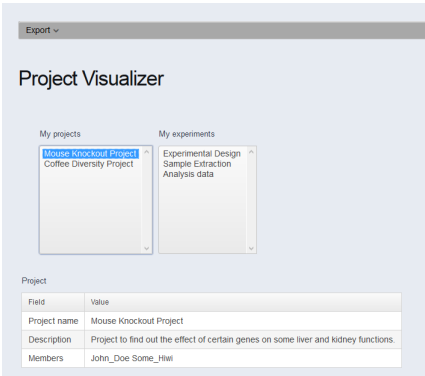


Figure 1: The GUI at the project level. Here, users can make a decision between various available projects and inspect additional information.

If the user clicks on a specific experiment, two additional tables are visualized (Fig. 2). One table showing available data files that are stored under a specific experiment and another one providing all existing experimental samples. The sample table has a multi selection feature, enabling the possibility to inspect the data sets that available for multiple samples. For instance, this can be seen in Fig. 2b. Here, two samples (QMOUS001A6 and QMOUS004AM) are selected, leading to a view at fastQ-files and quality control sites for each of the selected samples.

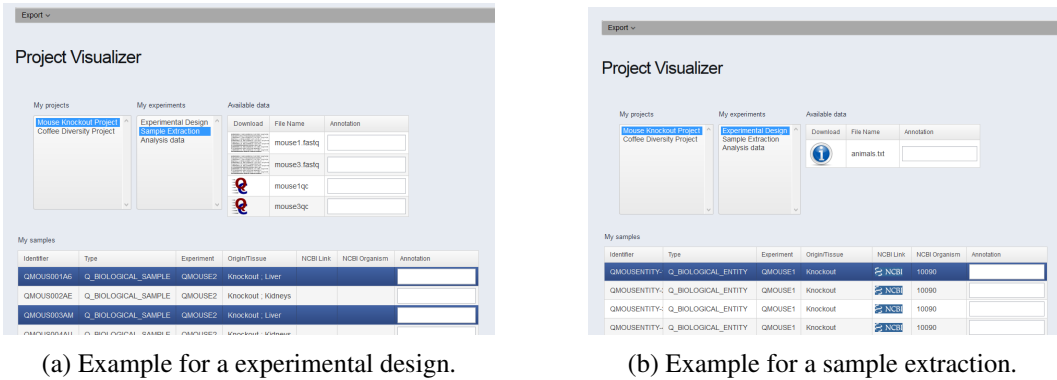


Figure 2: The GUI at the sample level. Here, users can inspect the collection of samples together with stored data sets.

... annotation, external link to ncbi, ...

3.3 Data Exporter

* Screen shot + description *

4 Discussion

Discussion of the columns, which are important for visualisation What did we additional need (IDs or stuff?) How do we want to visualize, what is expected at the experiment level. how should users navigate. Experiment subpoints

References

- [1] Vaadin Ltd. vaadin - user interface components for web apps, 2015.