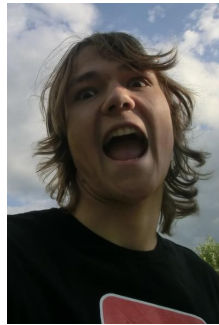


CONTEXTPROJECT PROGRAMMING LIFE
GROUP 2 - GEVATT
FINAL REPORT
TU DELFT



Ruben Bes
rbes
4227492



Mathijs Hoogland
mhhoogland
4237676



Jasper Denkers
jdenkers
4212584



Robbert van Staveren
rhvanstaveren
1527118



Willem Jan Glerum
wglerum
4141040

June 16, 2014

Abstract

This is the final report for the Programming Life Contextproject, a second year course from Computer Science at TU Delft. The Contextproject course is about applying all learned skills in a particular context at developing an piece of software. In this case the context was bioinformatics and we worked 10 weeks with a team of five people.

This document contains the main information about development, implementation and validation of the product. Main features of the product will be presented and it will be discussed how they satisfy the user needs. Furthermore, this document will contain an HCI module about the interaction of users with the product. Finally, an outlook will be given to show what possible improvements could be implemented if this project will continue in the future.

Besides this document several other documents are made covering other parts of the project. This is the final document consisting of the most information about the project.

Contents

1	Introduction	3
2	Overview	3
3	Description	4
4	Human Computer Interaction (HCI)	4
5	Evaluation	4
6	Outlook	4
7	Glossary	4

1 Introduction

The application developed is called GEVATT, which stands for GEnetic Variations Analyzer Through Triodata, which is an application used by doctors to browse genetic information of patients. Such applications are also called genome browsers. It is a secured web application based on the Play Framework. Doctors need to login from any device with a browser in order to use the application.

The application makes it possible for a doctor to upload a VCF file and let the application analyse it. A VCF (Variant Call Format) file is a file used in bioinformatics for storing gene sequence variations. This storing of genetic information of patients is based on variations between the patient and a reference genome. This is done because storing all information would be taking too much space. The

After uploading the file the user of the application, a doctor, waits till the file is processed. In the meanwhile the doctor could browse other patients he uploaded information about earlier. After uploading, the first important part of the application is executed: analysing the data. The outcome of the analysis consists of mutations found in the genome of the patient and the relations between these mutations.

Secondly, the main focus of the application is about visualising the found mutations. This is firstly done by giving a main overview of the whole patient. An overview of all chromosomes is given and per chromosome is indicated if and how many mutations it contains. Besides a visual overview of the chromosomes there's a tabular overview given with all mutations and some extra information per mutation. This makes it easier to estimate which mutations are more harmful than others.

Most information is shown on the overview pages per mutation. The pages with these visualisations have two visualisations which both give another insight in the mutation. The first part shows the position of the mutation relatively to a gene. The second visualisation is a graph based interactive visualisation showing proteins related to the mutation and the connections between these proteins. Here we also have a tabular overview per protein with per protein information about which diseases they could cause and to which other mutations of the patient they are related.

2 Overview

The developed product, a secured web application, is built on the Play Framework. It contains several parts that work together to deliver a user friendly environment for the users to explore genetic information.

Starting with the basis, it's a secured web application where a user needs to login. After logging in some secured pages become accessible. Some other pages already are accessible before logging in, e.g. the documentation and about pages.

The web framework and its builtin securing module isn't developed by ourselves so we could focus on developing the context specific parts. This was mainly separable in three parts: analysing data, retrieval of relevant data of databases and visualising data.

Analysing the data is done by processing the VCF file and detecting mutations. The outcomes of the processing is used in the visualisation. At both of these parts there's information used given by some databases. The information was used to get more information about mutations and find relations between several mutations.

3 Description

This chapter covers a detailed description of the developed functionalities in the product. The description is based on the overview of functionalities given in the previous chapter.

4 Human Computer Interaction (HCI)

Blaaat

5 Evaluation

Blaaat

6 Outlook

GEVATT is developed in quite a short time. The course Contextproject lasts about ten weeks and in this ten weeks the team had to become skilled in the context, make accompanying documents and present the final product. So only a limited part of the time is actually used to develop the application. This means we've focussed on the main parts of the application and there are more functionalities that could be added later if development would continue. This chapter is about what functionalities could be added.

7 Glossary

Blaaat