[DRAFT] Product planning

Group PL4
28 April 2016

Arthur Breurkes	4398033
Ricardo Jongerius	4379500
Niek van der Laan	4296915
Daphne van Tetering	4375165
Niels Warnars	4372069
Ties Westerborg	4386817

Context coordinator:	Dr. T. Abeel
Teaching assistants:	Mr S. Huisman MSc
	Mr J. Linthorst MSc
	Mr J. Hommes

Table of Contents

1. Introduction

The discovery and manufacturing of dozens of new antibiotic medicines back in the 20th century has started a revolution in the treatment of patients and has drastically reduced the mortality rate for several bacterial infections that have now become treatable. This revolution did not come without much effort. Years of research have contributed to the development of these new antibiotics. But drug resistance is on the lure as bacterial gen mutations threaten to cause a rise in the number of treatment failures. [1] If drug resistant bacteria are spread humanity might lose its ability to treat conditions that are not considered easily treatable.

In an attempt to develop alternative treatment methods for infections caused by resistant bacteria scientists are looking more and more at research fields as microbial genomics for help in understanding the mutations that cause drug resistance. [2] This quest asks for large sets of samples genome data coming from multiple infected patients which causes a bottleneck. As the amount of research data grows so does the amount of time to analyse this data, therefore suitable visualisation and analysis software is needed that can process several hundreds or thousands of genome samples at a time.

2016’s Programming Life Context Project focusses on addressing this need by introducing tooling that can be used to interactively explore a sequence graphs of different sized sections of bacteria's genome. By putting the graph in the context of the evolutionary relationship of different samples mutations can be traced back till the point in history where it arose.

2. Product

The result of this project is a genome visualizer that can be used to interactively compare genomes of DNA samples coming from multiple organisms. A researcher working on mutations in the genome of multiple organisms can use the product to view the exact type of mutation that has occured in the DNA of a sample compared to the reference. The following chapter describes the product backlog and the roadmap of DART-N visualizer. The backlog provides an overview of basic milestones that have to be reach after each weekly delivery. The roadmap on its turn provides an overview of which functionality is expected to be included in each release version.

2.1. High-level product backlog

The product backlog is divided into sprints with a duration of one week. Per week a number of milestones has been defined that have to be reached at the end of each sprint.

Week	Milestone
1	Design decisions are made, the first graphical interface is designed and the first data set is analysed. At the end of the week, the first demo is finished. This means that the DNA researcher will be able to load a small genome and analyse it.
2	The DNA researcher will now be able to use a first concept of semantic zooming and view the phylogenetic tree of the used genome.
3	The DNA researcher will be able to scroll through the phylogenetic tree and load a larger data set of more than 10 genomes.
4	The DNA researcher will be able to load more genomes (100+) at once, and compare these by picking a reference genome.
5	The DNA researcher will be able to fully use semantic zooming.
6	The DNA researcher will be able to view the percentage of mutations, deletions or insertions in a pie chart and will be able to view the percentages of A, T, C, G bases in a pie chart.
7	The DNA researcher will be able to see a representation of meta-data that is associated with the samples, such as drug resistance, the location of the isolation, the isolation date, etc.
8	The DNA researcher will be able to see an indication for the convergent evolution of several variant.

2.2. Roadmap

Each milestone defined in the the product backlog is accompanied by a product release.

Week	Release (version)
1	V0.1: Initial demo of the DART-N visualizer, containing only the basics as mentioned in the week 1 milestone.
2	v0.2: Version introduces semantic zooming.

3	v0.3: Version introduces a navigatable phylogenetic tree.
4	v0.3.1: Version introduces the ability of loading 100+ genomes at once.
5	v0.3.2: Version incorporates finished semantic zooming.
6	v0.4: Version incorporates pie charts.
7	v0.4.1: Version incorporates meta-data of the given samples
8	v0.5: Final version.

3. User stories

In the provided user stories a high-level definition of features or characteristics of the DART-N visualizer is described. These user stories include both functional and non-functional specifications of the visualizer.

Data loading
As a user
I want to be able to manually load a Graphical Fragment Assembly (GFA) data file
So that any given set of genomes can be loaded and transformed into a graph

Large number of genomes
Given that a GFA file of 340 genomes is given
Then it should be loadable and processable on a low-end desktop machine running a dualcore processor with 4GBs of RAM within three seconds.

Data / graph navigation
As a user
I want to be able to explore the comparison graph
So that mutations and evolution between samples can be studied

Nucleotide-level comparison
Given that the graph visualisation is fully zoomed in close enough that individual nucleotides can be distinguished and individual mutations exist
Then the affected sequence of nucleotides should be shown.

Whole-genome comparison
Given that the graph visualisation is fully zoomed out
Then vertices should no longer be distinguishable and ribbons should be shown instead of a graph.

Evolutionary relationship between bacteria
Given that a section of genome of multiple strains is visualized and mutations exist in these sections of genome
Then given evolutionary information should be visualized in the graph / ribbons.

Mutation visualisation
As a user
I want to be able distinguish different types of mutation

Scenario 1: Bubble visualisation
Given that a genome contains a mutation and the graph is zoomed in to the level at which individual nucleotides can be distinguished
Then mutations should be shown as bubbles which distinguish insertion, deletion, duplication and inversion.

Scenario 2: Glyphs
Given that a genome contains a common mutation
Then this mutation should be visualized using a glyph if the graph is zoomed out too far to distinguish individual nucleotides.

Scenario 3: Mutation class filtering
Given that multiple classes of mutations exist within a set of genomes and I am only interested in a certain class of mutations
Then it should be possible to filter on this specific mutation class

4. Definition of Done

The definition of done is created so that everyone that is involved in the project knows what done means. This also means that if a feature is done, everyone is sure that there is nothing left to do and the feature can be closed.
We divided our DoD in three parts, all concerning a different phase of the project.

Features
- All user stories are met
- Functional tests are performed by all members of the team
- Unit tests are written and passed
- The code is commented
- The build is passing

Sprints
- All user stories and features indicated in the sprint are included
- Unit tests are written and passed
- The build is passing
- The code is commented and the documentation is written and updated

Release
- The demo is handed over to the customer
- The documentation is included with the demo

5. Bibliography

[1] *History of Antibiotics - heeve.com* [Web page] publication date unknown [Retrieved 28-04-2016] Available from: http://www.heeve.com/modern-history/history-of-antibiotics.html

[2] *Whole-genome sequencing targets drug-resistant bacterial infections - Punina et al. 2015* 5-08-2015 [Retrieved 28-04-2016] Available from: http://humgenomics.biomedcentral.com/articles/10.1186/s40246-015-0037-z