Report: Software Project Management

Antje Oldenburg

Department of Computer Science, Free University of Berlin, Takustr. 9, 14195 Berlin, Germany
E-mail: *antje.oldenburg@fu-berlin.de*

1 INTRODUCTION

In summer term 2013, the author attened the course "Software Project Management". The objective was to create an application "Mini-FASTQC: NGS Quality Control". This paper reports on project management aspects of creating this application.

2 OBJECTIVES

Using seqan C++ library, an application was to be created to report statistics on NGS data.

Established basic quality measures for the analysis of fastq files are base call quality and nucleotide distribution. In addition to those, checks for overrepresented and duplicated sequences, k-mer distribution where chosen to be implementation features.

The project was to encompass 4 weeks of software development by two student developers each. A plan was proposed, by which the effort was to be divided into 4 one-week iterations. One of the goals in this plan was to support development in parallel. Both students where encouraged to work on individual features by themselves

2.1 Week 1

2.1.1 Goal Create a minimal application

2.1.2 Tasks Achieved Project setup and source file structure was introduced and thus made possible a continuous code-build-test cycle. A central, singular ReadStats object was introduced to hold statistical information. The app main functionality was created with the support for a FASTQ input stream and the statistics object was populated with readlength information.

This week was also used to research doxygen and start commenting code.

2.1.3 Problems Encountered Interaction with the file system was researched and restricted to POSIX functionality. We decided to concentrate on creating a practical and not too time consuming way to create output folders and graphical components.

2.1.4 Review The interaction of the devlelopers was not sufficient to support TSV files as contract for the expected output.

2.2 Week 2

2.2.1 Goal Implement all general statistical features.

2.2.2 Tasks Achieved This weeks goal was behind target by 1 day. The ReadStats object was extended to collect nucleotide distribution data and raw score value distribution data (both per read and per position in read) that can be used to create TSV files. A class TSVWriter was created to support a separation of model and view in the application. Acceptance tests where set up using a recursive diff call to compare application output to the expected output. The task of adding argument parsing to support customization of features was not achieved.

2.2.3 Problems Encountered Test and Fail critera for statistical tests were unclear and the decision was made to use fastqc's criteria.

2.2.4 Review At this point, the developers were able to implementation their assigned features without consulting each other. It resulted in unclear expectations and missed teamwork.

2.3 Week 3

2.3.1 Goal Implement sequence duplication statistics

2.3.2 Tasks Achieved The main goal, the implementation of the sequence duplication logic was achieved. The ReadStats object was extended and refactored to handle variable read lengths.

2.3.3 Problems Encountered At this point comparison of fastqc results with those of the newly implemented application showed differences whose research took away from implementation time.

2.3.4 Review Because of the time it took to research different results in different applications, the argument parser functionality was once again not implemented. Also, the adaptation of the diff mechanism for acceptance testing was not performed.

2.4 Week 4

2.4.1 Goal Cleanup, Test and Document

2.4.2 Tasks Achieved When going through the application after the main functionality was implemented, a large number of small tasks, corrections and refactorings took up most of the time this week.

2.4.3 Problems Encountered

2.4.4 Review The lacking consultation of the developers showed issues too late and served to show that expectations not presented clearly would not be met on the last day. Despite all efforts, the development team was not able to work together enough to produce a graphical output that can accompany this report.