

TI2806: CONTEXT PROJECT

TOOLS FOR SOFTWARE ENGINEERING

# Product Vision

**Octopeer Viewer**

Group: TSE1

*Borek Beker*

*Marco Boom*

*Leendert van Doorn*

*Ahmet Gudek*

*Daan van der Valk*

April 28, 2016

# 1 Introduction

Last couple of years, services for version control and continuous integration of software, like GitHub, have become popular tools to make software development easier. A cloud based codebase made it significantly easier for (large) teams to work on the same project, both commercially and non-profit. For both public and private projects, contributors suggest a major change by creating a pull request. These changes are then reviewed by the responsible project members or other contributors in so called *peer reviews*. The *Octopeer* project aims to collect data about these peer review sessions through a browser extension.

The goal of the **Octopeer Viewer** is to present data collected by the Octopeer extension in the most clear and useful way. It is a web application, accessible via the browser extension, and can be adjusted to the users needs. It should give insights in the reviews of individuals, as well as reviews on project or company scale.

## 2 Target audience

As part of the Octopeer project, the Octopeer Viewer mainly targets Software Engineers, Testers, Managers, Computer Science students, and others involved in software engineering where code reviews are a part of the work cycle. It is essential to the popularity and effectiveness of the project, to become part of as many software projects as possible. This will be achieved by making it a handy tool for both code reviewers and managers. The following personas could all be potential users of the Octopeer Viewer, having different backgrounds and goals.

### 2.1 Personas

#### 2.1.1 Lars

Lars (48) is a project manager of a software development company. The teams he is involved in produce a lot of code, aiming to produce a working deliverable every two weeks. To recognize and eliminate as much bugs as possible, Lars' company intensively uses peer reviews before branches are merged with the master. Lars would like to make this peer review process as efficient as possible, by measuring the code reviews of his team members. He aims to select his best code reviewers, in terms of speed and number of found bugs, to create a comprehensive protocol for peer reviews, getting the best out of his employees. Lars decides to use Octopeer, hosted on a company server, to use with his teams.

#### 2.1.2 Lisa

Lisa (19) is a first year Computer Science student. She has just learned to program in java, and wants to contribute to Open Source projects on Bitbucket. A project in which she is interested, demands her to install the Octopeer plugin for Chrome to monitor her code reviews. Lisa is not so much interested in this,

but agrees and uses the plugin anyways. After some months and a lot of both programming and code reviewing in various project, she opens the Octopeer Viewer and is curious to see whether her code review sessions have changed. See is able to see how her code review speed (per changed line) and quality (usefulness of comments, spotted bugs) have changed.

### **2.1.3 Eric**

Eric (37) is a software developer at an Open Source IT company. He works in a multidisciplinary team in which Eric both commits and reviews code. Eric would like to get insights on his code review sessions, aiming to do work faster, without missing any problematic parts. He installs the Octopeer browser plugin to get statistics helping him assess his GitHub peer review sessions, taking a look at the data using Octopeer Viewer.

### **2.1.4 Tim**

Tim (27) is a PhD student who researches peer reviewing procedures. He would like to get access to as much data as possible about different people looking at merge requests. Tim uses the public Octopeer database, containing the information of Octopeer users who agreed on sharing their anonymised data with the project. The various overviews and graphs of the Octopeer Viewer help Tim to analyse and understand the information gathered by the Octopeer project.

## **3 Customer needs**

There are a few different perspectives on the Octopeer Viewer product:

- Generally, software reviewers will mostly be interested in their own statistics, comparing their own development and changes over time, and might want to compare this to team/project averages;
- Project managers may want to see statistics on reviews of their project(s);
- Researchers will be interested in analysing as much data as possible.

## **4 Crucial product attributes**

The stated customer needs imply that there are several distinct pages required, corresponding to the scale the user is interested in. The most important product attributes are:

- Usefulness: by presenting useful data, the users will get practical and statistical insight in the peer review process. The Octopeer Viewer not only be fun to play with, but provide helpful analytic tools.

- Speed: as there is a lot of data being generated by the Octopeer extension, the Octopeer Reviewer should generate graphs and overviews fast.
- Adaptability: different users will focus on different elements of the data. This might not only vary from scale, but also to focus on specific parts of the data: mouse movements, coverage of all changes, comments and so on.

## 5 Comparison to existing products

Although there exist many tools for software reviewing, like Crucible (<https://www.atlassian.com/software/crucible>), there are no software review analysis tools available to the public. This provides a great opportunity for the Octopeer project.

## 6 Project time span

This project takes 10 weeks, and will contain 8 one-week sprints with a working deliverable. The project will be completed on June 17, 2016, and the final report will be handed in on June 23, 2016. After this, Octopeer Viewer should be merged with the Octopeer project.