TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Informaatikainstituut
Tarkvaratehnika õppetool

Arendusprotsessi ja selle tulemite kvaliteedi parandamise võimalikkuse analüüs Swedbank ASi projekti „Makselink“ andmestiku põhjal

Bakalaureusetöö

Üliõpilane: Ingrid Puusemp
Üliõpilaskood: 121097IAPB
Juhendaja: Tarvo Treier

Tallinn
2015

1. Kokkuvõte

Autor täitis antud bakalaureusetöö alguses seatud eesmärgi, analüüsides projekti käigus tekkinud vigu ning andes soovitusi parandamaks arendusprotsessi ning selle tulemi kvaliteeti. Selle eesmärgi saavutamiseks täitis autor alguses seatud alameesmärgid, korrastades raporteeritud vigade andmestiku, leides vigade analüüsiks sobilikud parameetrid, analüüsides projekti käigus raporteeritud vigu, leides meeskonnaliikmete parandamist vajavad tegevused, pakkudes lahendusi arendusprotsessi ning selle tulemite kvaliteedi parandamiseks, tuues välja vigade raporteerimis- ja haldamisprotsessis parandamist vajavad kohad ning tuues välja, kuidas on tulevikus võimalik antud analüüsi uuesti läbi viia.

Lahendused seatud alameesmärkide täitmiseks:

- Autor sorteeris kõik JIRAs ja Bugzillas raporteeritud vead ning kasutuslood ja valis parameetrid, mille alusel vigade analüüsi läbi viia. JIRAst vigade kätte saamiseks oli vaja käia läbi kõik vead ning kasutuslood, mis olid vähemalt korra arenduse käigus lahti tehtud. Bugzillast olid vead imporditavad. Parameetrite väärtuste kätte saamiseks oli vaja lugeda vigade ning kasutuslugude kommentaare. Autor analüüsis kokku 132 viga.

- Autor koostas andmestiku põhjal diagrammid, et näha, millises arendusetapis tekib kõige rohkem vigu ning mis neid vigu põhjustab. Lisaks sellele kaua võtab keskmiselt aega erineva tõsidusega vigade parandamine ning millistel tarkvara komponentidel viga tekib. Analüüsi põhjal selgus, et kõige rohkem tekib koodi vigu ning kõige kriitilisemad vead tekivad Androidi mobiilirakendustes. Kriitilisi vigu parandatakse ning testitakse kokku keskmiselt umbes 3 päeva. Kõige põhilisemaks probleemiks oli ajanappus ning riistvaralised ning tarkvaralised eripärad, millega ei jõuta ennast alati kurssi viia.

- Autor leidis kahe vigade raporteerimise keskkonna vahelised erinevused ning otsustas, mis on kasulik ning mis mitte, et tulevikus oleks võimalik analoogset analüüsi läbi viia lühema ajaga ja efektiivsemalt. Üheks peamiseks soovituseks oli see, et vigade raporteerimiseks peab olema üks keskkond ning vead peavad olema kõikide

parameetritega, mille järgi analüüsi teha, imporditavad Excelisse. Autor jõudis järeldusele, et kuna kasutuslood luuakse JIRAsse, siis oleks mõistlikum ainult seda kasutada. Vea tekkimise põhjuse väli peaks olema lisatud vormile, et testija saaks kohe vea raporteerimisel selle väärtuse lisada.

Sellist analüüsi on plaanis teha tulevikus sarnastes projektides kui ka mahult suuremates projektides. Seejärel tuleb tulemusi võrrelda antud lõputöö koostamise käigus läbi viidud analüüsiga. Tulemusena on näha, kas tehtud soovituste põhjal muutub arendusprotsess kvaliteetsemaks ning optimaalsemaks. Kui jah, võib antud töö tulemusi Swedbanki protsessides ka edaspidi rakendada. Kui ei, tuleks soovituses teha kohandused, mis viivad soovitud tulemuseni.

Summary

The author fulfilled the aim posed at the beginning of this thesis, to analyse the bugs found in the course of the project and make suggestions for improving the development process and the quality of its outcomes. To reach this aim, the author fulfilled her initial sub-aims, organising the dataset of reported bugs, finding suitable parameters for analysing the bugs, analysing the bugs reported during the project, finding the actions the team members needed to improve, suggesting solutions for improving the development process and the quality of its outcomes, bringing out specific places in the reporting and management processes and the possibilities of carrying out the same analysis again in the future.

Solutions for reaching the aims posed:

- The author sorted all bugs reported in JIRA and Bugzilla, and all user stories, and chose parameters for analyzing the bugs. In order to reach the data from JIRA, all bugs and user stories that had been reopened at least once in the development process had to be looked at separately. From Bugzilla, bugs could be imported. To get the parameter values, the comments for bugs and user stories had to be read. The author analysed 132 bugs altogether.

- On the basis of the dataset, the author compiled diagrams to see in which stage of development the most bugs appear and what causes them, how long on average fixing bugs with different severity takes and on which component. The analysis revealed that code bugs are the most common and the most critical ones appear in Android mobile applications. Critical bugs are fixed and tested for three days on average. The main problems are lack of time and specificities of hard- and software that cannot always be studied in detail due to this.

- The author found differences between the two bug reporting environments, and decided what is useful and what is not in the light of carrying out a similar analysis more effectively and more quickly in the future. One of the main suggestions is that there should be only one environment for reporting bugs, and the bugs need to be importable to Excel spreadsheets together with all the parameters used for analysis. The author reached the decision that since user stories are created in JIRA, it would be

more sensible to use this environment only. A field for “the reason for the bug” should be added to the form so that the tester can add the value while reporting the bug.

In the future there is a plan to make the same analysis in similar as well as in bigger projects. After analyzing the projects, the results have to be compared with analysis of the bachelor thesis. In the result, it can be seen whether the recommendations have increased the development processes quality and optimality. If the answer is yes, then the results can be used in Swedbank’s processes in the future. If not, changes should be made in the recommentations that will lead to the expected result.