What automated Devops tools can help better understand clients needs throughout the Devops cycle?

Billy Bouchard, Nicolas Legros
billy.bouchard@polymtl.ca, nicolas.legros@polymtl.ca

***Abstract*—Understanding clients' needs and getting their feedback has always been a critical part of a software's success. However, the techniques and strategies used in the industry are less than optimal. For instance, research shows that up to half of all the features in products are never used [1].**

The goal of this paper is to aid development teams to improve their process in acquiring clients' feedback to better guide their decisions in future development cycles by providing an analysis on the different tools and strategies used in the industry and their characteristics.

I. Problem Statement & Link with Course

Since every piece of software exists to serve a client, wether internal or external, it's critical that the client's needs stays in the mind of its developers. However, once the software goes live, updates are always necessary to patch some small bugs and add interesting features. It can be challenging to evaluate what features would be the best or even which ones that were succesful. Moreover, developers sometimes work on features for a long time before validating their relevance to their client. It's important to keep the client at the center of decisions for new or existing features. Testing new features or confirming feature success by getting customers' feedback can require the creation of focus groups, surveys or other more complex systems. Using devops to create, receive and analyse customer feedback in a timely fashion would therefore help development teams in building a successful software. Indeed, automating the collection of customer feedback and automating the analysis of a software's metrics could substantially help development teams to be successful.

II. Research Questions & Motivation

Knowing that the focus of this research will be on the automation of different techniques, it should come to no suprise that the first part of the research will be to find what can actually be automated in the feedback collection process.

> RQ1. *What feedback techniques can be automatically operated throughout the Devops cycle?*.

Once there is a sufficient amount of information found on these strategies, we will focus on the different metrics leading to RQ2:

> RQ2. *What feedback metrics can be automatically gathered and processed through logs or other Devops tools and metrics?*.

Once the compendium of both the techniques and the metrics is found, we will be able to focus on their individual usages to see how they are currently used in the industry which lead to RQ3:

> RQ3. *How often are these techniques and metrics implemented in Devops projects?*.

All these considered, we should be able to better explain what techniques seem to be both easier to implement and that yield the best results.

III. Dataset & Analysis

Since the primary goal of the research is to characterize different approaches of collecting customer feedback, the dataset can't be composed of a single software development project. *Github API* allows to collect data from multiple open-source repositories easily. Furthermore, we can filter repositories with selected keywords with the *API* to narrow down the search to projects who collect and analyse user feedback. Some strategies will also be evaluated from literature and research papers we can find on the subject. This will help to get a good picture of the multiple techniques used in user feedback collection since some companies don't always open-source their code.

As for the analysis, existing literature will help on characterizing some approaches. The ones not mentionned in literature will need to be evaluated manually. The key principles mentionned in *Section IV* will serve as a baseline when evaluating different user feedback collection techniques.

IV. Two Related Papers

This research should be interesting due to the fact that there is not much research on the subject. However, a ton of corporate articles (e.g. Devops and development lifecycle in Google's Cloud Architecture Center) exist and explain some core concepts. One of the most interesting papers for this research is written by Google It explains common metrics that Google uses throughout their devops cycle [2]. It goes into the details to explain the importance of both speed and regularity when it comes to customer feedback. They present five key metrics to implement a good customer feedback cycle and process.

1) *Acquisition*: The percentage of users that come to your site who create an account.

2) *Activation*: The percentage of acquired users that activate their account and use the service.
3) *Retention*: The percentage of activated users that return to the service.
4) *Referral*: The percentage of retained users who refer other users to the service.
5) *Revenue*: The percentage of referring users who actually pay money for the service.

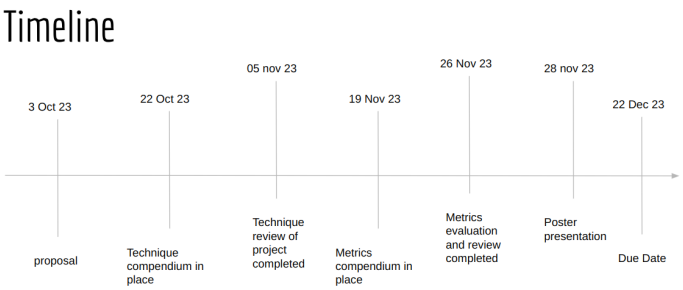
This corporate article is one of the numerous that explain well how to scale devops pratices to include customers into the process.

Another really interesting article is one from A. Fabijan et al on customer feedback and data collection techniques in software R&D [3]. This article goes in depth to find all the different techniques and metrics that the authors could find to qualify and quantitfy customer feedback. It gives an incredibly thorough start point to see what techniques and metrics this study should analyse and rank. It explains both the stages at which the techniques must be built, but also the limitation each strategy faces. Therefore, it will be easy to focus on the techniques that can be included in a devops cycle as both automated or semi-automated approaches.

V. Time Planning of Project

The complete paper should be done by December 22 and the poster by November 28. The research will be done in separate blocks of the complete techniques analysis followed by the complete analysis of the metrics, and then the combined analysis. The last month of the project will be allocated to write the report once the poster is completed. Therefore, the research should be completed by November 25 to keep a weekend as a buffer. As the metrics evaluation needs to be done once the techniques evaluation is completed, the metrics evaluation will start the week before (i.e. November 19). It would be reasonable to allocate about two weeks to collect all the required information on the metrics, which would bring to starting the metrics analysis on November 5. This means that the techniques analysis will need to be completed by then. It gives us a total of four weeks to complete the techniques analysis. The completed timeline can be seen at figure 1.

Figure 1: Projected timeline for the project



References

[1] H. H. Olsson and J. Bosch, "Towards continuous customer validation: A conceptual model for combining qualitative customer feedback with quantitative customer observation," vol. 210, 06 2015, pp. 154–166.

[2] "Devops capabilities: Customer feedback." [Online]. Available: https://dora.dev/devops-capabilities/process/customer-feedback/

[3] A. Fabijan, H. Olsson Holmström, and J. Bosch, "Customer feedback and data collection techniques in software r&d: A literature review," in *Software Business: 6th International Conference, ICSOB 2015, Braga, Portugal, June 10-12, 2015, Proceedings*. Springer, 2015, p. 139–153. [Online]. Available: https://urn.kb.se/resolve?urn=urn:nbn:se:mau:diva-12419