

Retrospective based on Data-Driven Persona Significance in B-to-B Software Development

Yasuhiro Watanabe, Hironori Washizaki,
Kiyoshi Honda, Yoshiaki Fukazawa

Waseda University
Shinjuku-ku, Tokyo

jellyfish44-time@akane.waseda.jp, washizaki@waseda.jp,
khonda@ruri.waseda.jp, fukazawa@waseda.jp

Masahiro Taga, Akira Matsuzaki, Takayoshi
Suzuki

e-Seikatsu Co.,Ltd.
Minato-ku, Tokyo

masahiro.taga@e-seikatsu.co.jp, akira.matsuzaki@
e-seikatsu.co.jp, takayoshi.suzuki@e-seikatsu.co.jp

ABSTRACT

A Business-to-Business (B-to-B) software development company develops services to satisfy their customers' requirements. Developers should prioritize customer satisfaction because customers greatly influence on agile software development. However, it is possible that a B-to-B software development company has following issues: 1) failure to understand actual users because the requirements are not often derived from actual users and 2) failure to satisfy the future customers' requirements when only satisfying current customers. Although many previous works proposed methods to elicit the requirements based on actual quantitative data, these works had not considered customers and end-users simultaneously. Herein we proposed Retrospective based on Data-Driven Persona Significance (ReD2PS) to help developers to plan future releases. ReD2PS includes Persona Significance Index (PerSil) to reflect the correspondence between target users, which developers assume based on requirements in releases, and end-users' personas. A case study involving a Japanese cloud application shows that PerSil reflects the relationship between target users and end-users to discuss about the validity and effectiveness of ReD2PS.

1 INTRODUCTION

In Business-to-Business (B-to-B) domain, customers greatly influence on agile software development. Continuously, developers often prioritize customer satisfaction. However, in B-to-B development situations, customers may differ from actual end-users and customers' requirements may not reflect on end-users' requirements. Customers' business goals typically depend on acquiring and maintaining end-users. Consequently, end-user satisfaction is important, and software does not provide value to customers when end-user dissatisfied.

Additionally, even in a B-to-B software development company, software development is an activity to achieve business goals and must consider the linkage between business goals and software development [6]. However, we assume that a B-to-B company often misses the improvement for the future customers because current customers are often reluctant towards changes in service [15].

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ICSE-NIER'18, May 27-June 3, 2018, Gothenburg, Sweden

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-5662-6/18/05.

<https://doi.org/10.1145/3183399.3183410>

To solve above issues, developers should clarify end-users' requirements considering customers' requirements. With regards to understanding end-users, some previous works proposed practices to elicit requirements based on quantitative data [9] (Data-driven requirement engineering). Moreover, many researchers have reported that personas, which are fictional characters to grasp users' requirements, constructed from actual users' data [13] [10] [14] (Data-Driven Personas). However, previous research had considered customers and users individually.

Herein we propose Retrospective based on Data-Driven Persona Significance (ReD2PS) to clarify the developers' implicit priorities of personas. ReD2PS's contributions are followings:

- Identify the implicit target users and end-users' personas to help developers to understand the current planning.
- Indicate the developers' implicit priorities of personas and help developers to plan future releases.

To validate ReD2PS, we employ a case study in E-bukken-one, which is a Japanese B-to-B cloud application.

The remainder of this paper is structured as follows. Section 2 introduces some basic concepts of ReD2PS and our motivating example. Section 3 explains the details of our approach, while Section 4 describes about the case study. Section 5 analyzes the case study results. Section 6 mentions threads to validity. Finally, Section 7 summarizes our contributions.

2 BACKGROUND

Data-Driven Requirement Engineering practices are requirement engineering practices based on quantitative data [9]. Additionally, many previous works reported requirement prioritization method based on quantitative information [3] [7]. However, these works had not considered customers and end-users simultaneously.

ReD2PS is a retrospective method with data-driven personas to consider both customers and end-users. A persona is a fictional character developed to understand users' requirements [13]. It has some attributes like a real person (e.g., name, gender, job, characteristic, goal for its service, etc.). To construct more reliable personas, many researchers have reported that personas can be based on actual users' data employing machine-learning approach [10] [14].

Some works reported issues when personas are used in actual development [12] [8] [5]. With regards to these issues, previous work provides the utilization of personas for business decision-making or requirement elicitation [11] [2] [4] [1]. These works do not consider both a B-to-B company's customers and its service's end-users.

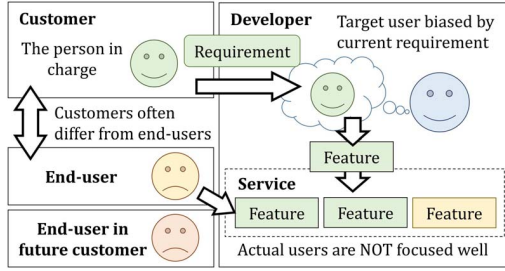


Figure 1 : Issues with a B-to-B software development

Herein we explain our motivating example. For B-to-B software development organizations to make a profit, developers must satisfy current customers' requirement. In this case, it is difficult for developers to satisfy actual end-user requirements when only considering the requirements of current customers because customers often differ from actual end-users (Figure 1). This prevents B-to-B companies from achieving customers' goals because the conclusive goal of customers is to benefit through satisfaction of the end-users. Moreover, this issue leads developers to missing the opportunities to acquire the new customers because future customers may differ from current customers with respect to system usage and the requirements.

3 PROPOSAL

3.1 Overview

ReD2PS is an iterative practice to verify the scope of a project based end-users of a service. Figure 2 shows three steps of applying ReD2PS: 1) Construct end-user personas. 2) Identify target users in each iteration. 3) Adapt target users to personas.

The relationships among the terms in ReD2PS are shown in Figure 3. In ReD2PS, end-users are represented as personas which are constructed from latest users' logs on a service when planning backlog. In our proposal, target user is defined implicitly to clarify the purpose of a release. We assume that developers plan each release considering target users. In ReD2PS, each persona's significance is calculated based on identifying target users of current backlog when planning backlog.

To apply our method, the following data sets should be prepared: D1) End-users' log history to construct personas, D2) Issues which include related feature in previous releases and D3) Issues that developers want to identify target users.

3.2 Construct end-user personas

To understand the requirements of end-users, personas are constructed based on D1 via a data-driven approach (end-users personas in Figure 2 step 1). In ReD2PS, we assume that actual user tasks in a service can be derived from end-users' usage through a topic model and personas can be constructed from frequent patterns of user tasks. In ReD2PS, personas are defined based on frequent task usage pattern of services (user task in Figure 3) and other characteristics (age, gender, goals etc.) are not defined. However, we assume that personas can be defined because previous work includes identification of personas based on users' task or intention patterns [10] [14].

Generally, a topic model is applied in Natural Language Processing to predict abstract topics in a collection of documents. An

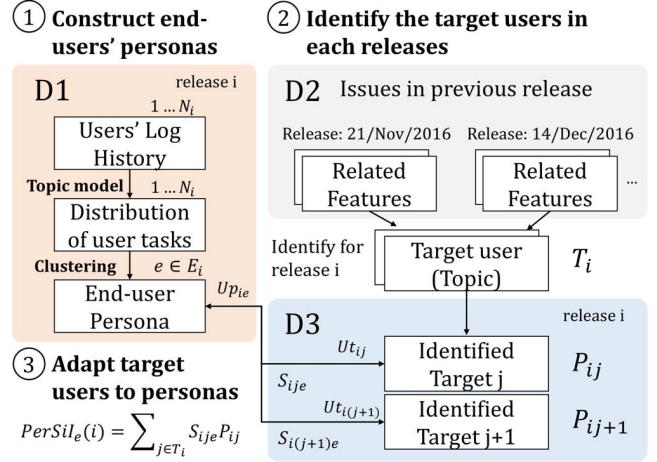


Figure 2 : Procedure to apply ReD2PS to a service

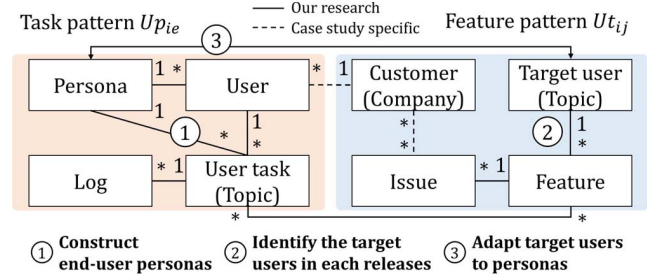


Figure 3 : UML class diagram based relations of steps and attributes in ReD2PS

end-user's usage log is the list of actions that the user took. In ReD2PS, actual user tasks in a web service can be derived as topics by applying a topic model when an end-users' usage log is considered as a collection of item list. For example, in the online shopping site, we may identify user tasks: "buy the items" when users took actions: "watch the shopping cart". We assume that we can define personas based on the frequent pattern of user tasks.

3.3 Identify the target users in each release

In ReD2PS, target users are extracted by applying a topic model to a collection of features in D2 (target users in Figure 2 step 2). We assume that each requirement is divided into small issues and issues can be categorized into related features based on user tasks (issue and feature in Figure 3). When developing a service, developers plan implementations based on actual users and the pattern of implemented features depends on target users (Figure 1). For example, in online shopping service, when developers' target user is a customer, "search the item" and "buy the items" feature may be improved to satisfy such customers. In this example, the feature pattern is "search the items" and "buy the items".

3.4 Adapt target users to personas

To analyze the current direction of a service, we propose the index which considers the developers' priorities of end-users in every release (PerSil). PerSil reflects the relationship between customers'

requirement and end-users, which is not always obvious (Case study specific in Figure 3).

In ReD2PS, we consider the importance of the target users as the appearance of target user when applying a topic model in each release. ReD2PS assumes that similarity between a persona and a target user is considered as the similarity between persona's user task pattern in step 1 and target user's feature pattern in step 2. In ReD2PS, the following information must be acquired:

- T_i is the set of estimated target users in release i
- N_i is the number of users to construct personas in release i
- E_i is the set of end-user personas in release i
- P_{ij} is the frequency of predicted target user j in release i
- S_{ije} is the similarity between target user j and persona e in iteration i

For example, in the case study we held, we used Jaccard similarity for S_{ije} . Let U_{tij} and U_{pie} be the set of related features of target user j and persona e . The Jaccard similarity between U_{tij} and U_{pie} is $S_{ije} = \frac{|U_{tij} \cap U_{pie}|}{|U_{tij} \cup U_{pie}|}$.

We propose the following Persona Significance Index for persona e in iteration i ($PerSiI_e(i)$):

$$PerSiI_e(i) = \sum_{j \in T_i} S_{ije} P_{ij} \quad (1)$$

PerSiI reflects how much developers consider actual end-users by focusing on target users. Statistical analysis of $PerSiI_e$ can help developers to plan releases through clarifying current direction of a service.

4 CASE STUDY

4.1 Overview of the case study

This study investigates the following research questions: RQ1) Do PerSiI reflect the correlation between target users and personas? RQ2) Do PerSiI help developers to plan releases considering end-users?

To answer the above research questions, we employ a case study involving E-seikatsu, which is a Japanese B-to-B company. E-seikatsu provides E-bukken-one which is a cloud application to assist with the tasks in the real estate domain. The tasks in real estate operation are divided into sales or rental. In E-bukken-one, 12 features are defined for each sales and rental.

In this case study, we obtained following data: D1) Users' log history from 1 Dec. 2016, D2) 896 tickets which are implemented in 68 releases from 2012 to Nov. 2016, and D3) 48 tickets which are released in 4 major releases (release 0: 14 Dec. 2016, 1: 1 Feb. 2017, 2: 22 Feb. 2017 and 3: 29 Mar. 2017).

4.2 Construct End-users' personas

To construct end-users' personas, we applied the Hierarchical Dirichlet Process (HDP), which is a topic model, to users in each release. We defined personas in release i are constructed from users who used E-bukken-one from two weeks prior to a date of release $i-1$ because these users are latest when planning release i . The log data was converted into a collection of each user's action list for release 1, 2 and 3. Afterwards, we applied HDP to it and estimated user tasks. HDP provides the frequencies of each user task (topics in

Table 1 : Target users (T_i)

| Release | Id | Description | P_{ij} |
|---------|-----------|--|----------|
| 1 | TB_{11} | Sales: Managing only contracts | 0.407 |
| | TB_{12} | Sales: Sales personnel | 0.305 |
| | TC_{11} | Rental: Sales personnel | 0.730 |
| 2 | TC_{21} | Rental: Sales personnel | 0.217 |
| | TC_{22} | Rental: Managing contracts & customers | 0.161 |
| | TC_{23} | Rental: Managing only contracts | 0.314 |
| 3 | TB_{31} | Sales: Managing only contracts | 0.835 |
| | TC_{31} | Rental: Customer support | 0.040 |
| | TC_{32} | Rental: Managing contracts & customers | 0.208 |
| | TC_{33} | Rental: Managing the owners | 0.103 |
| | TC_{34} | Rental: Sales personnel | 0.351 |

HDP) for each user. To summarize users' action and define personas, we applied Spectral Clustering, which is an extension of K-means clustering, to users' frequencies of each user task. Based on Calinski-Harabasz index, which is a clustering criterion, we identified eight clusters in each iteration and defined five common clusters as personas in every release (E_i). End-users personas are followings:

- EP_1 : Rental, Managing information of real estate
- EP_2 : Sales, Sale personnel
- EP_3 : Rental, Sale personnel
- EP_4 : Sales, Managing information of real estate
- EP_5 : Rental, Managing contract and customers

4.3 Identify the target users in each release

To estimate the target users in each release, we applied the Latent Dirichlet Allocation (LDA) to D2 and predicted target users in D3. For every 68 releases in D2, we acquired collections of lists for the sales and rental features for each iteration. As preprocessing, we converted tickets released on the same date into a list of related features. Then we applied LDA to every two collections of lists. When applying LDA, the number of topics equaled the number of combinations of actors for rental or sales. After training the LDA model, we applied it to D3, which is processed in the same way to estimate the topic distribution in each iteration (P_{ij}). We selected more frequent topics than a randomly selection and defined them as target users for each of sales (TB in Table 1) and rental (TC in Table 1).

4.4 Adapt target users to personas

To evaluate the target users in each release, we calculated PerSiI. To determine the similarity between target users and personas (S_{ije}), we identified U_{tij} and U_{pie} by relating the topics of personas and target user to features manually. After identifying the relationship, we calculated the $PerSiI_e$ for releases 1, 2 and 3. We show $PerSiI$ and average median of users' action time in each persona (Figure 4).

In this case study, E-seikatsu recorded the company reporting the requirement for each ticket and the company of each user. To validate PerSiI, we calculated Spearman's rank correlation coefficient between the series of $PerSiI_e$ and the series of the number of companies that users in each personas belong to and reported the tickets in each release (Figure 5). Correlations were 0.900 for release 1, 0.872 for release 2, and 0.616 for release 3.

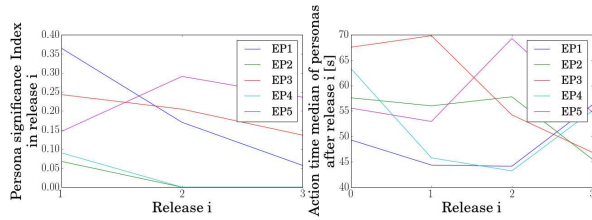


Figure 4 : Statistics of $PerSil_e$ and performance for five common personas

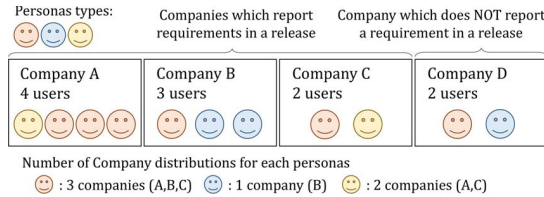


Figure 5 : Relation between companies and personas

5 DISCUSSION

In this section, the validity (RQ1) and effectiveness of ReD2PS (RQ2) are discussed based on the result of case study and the comments from a manager of software quality and CIO in E-seikatsu.

1) **As the answer to RQ1**, $PerSil$ reflects the correspondence between target users and personas. The manager said: “Performance of EP_2 and EP_4 was improved in release 1 while we just fixed some bugs in release 2 and 3. It seems that $PerSil_2$ and $PerSil_4$ can reflect the actual target users in releases and we can guess the improvement of their performance are grounded on the implementation.”

Additionally, in our case study, we assume that personas are prioritized based on how typical personas are in companies which reported requirements for each release. $PerSils$ in release 1 and 2 are highly correlated with the distribution of customer in personas. Although $PerSil$ in release 3 is not so correlated as other releases, we observed that a persona’s company reported requirements for other personas. This result suggests that $PerSil$ clarifies the developers’ priorities to end-users even if end-users’ companies are not tracked.

2) **As the answer to RQ2**, $PerSil$ helps developers to plan the release by identifying the current direction of a service. In this case study, the performances of EP_2 and EP_4 are improved after release 1 (right side in Figure 4). $PerSil_2$ and $PerSil_4$ indicates that EP_2 and EP_4 are focused in release 1. Although the performance of EP_2 and EP_4 changed in release 2 and 3, we can determine whether these performance changes are grounded on the release or not.

On the other hand, the CIO suggested that releases’ effectiveness to end users cannot be explained based only $PerSil$ and the CIO said: “When developers have clear target users in a release, sales personal in our company can explain the improvement of our service to customers successfully. It is possible that customers’ understanding contributes the improvement of performance.”

However, the CIO’s feedback also suggests that ReD2PS helps developers to communicate their implementations to sales personas through clarification of personas which they focus. Moreover, ReD2PS helps developers to plan releases with specific purposes. A manager also said: “In release 2 and 3, many target users are estimated and focused (TC in Table 1). It is possible that we implemented some defects due to the unclear purpose. We perceive that we just

implemented remaining reported bugs without purposes. ReD2PS may help to plan the releases for a specific purpose based on actual users.” Therefore, ReD2PS provides the correspondence between target users and personas and helpful analysis to revise the direction of a service.

6 THREATS TO VALIDITY

The results depend on the data analysis method. In this case study, Ut_{ij} and Up_{ie} are identified manually. Additionally, our analysis does not consider domain-specific factors which affect users’ task. Moreover, ReD2PS could not be applied to many releases because the users’ logs in the previous release date were unavailable.

7 CONCLUSION AND FUTURE WORK

We assume that satisfying only current customers’ requirements will not satisfy actual end-users’ requirements and developer company’s business goal. To help developers to consider stakeholders and end-users, we proposed Retrospective based on Data-Driven Persona Significance (ReD2PS). ReD2PS can confirm a service’s direction by analyzing Persona Significance Index ($PerSil$) to reflect the correlation between target users assumed by developers based on customers’ requirements and end-users’ personas. Through a case study involving a Japanese cloud application, Validity and effectiveness of ReD2PS are discussed. In the future, we will employ additional case study to verify and extend ReD2PS in terms of persona utilization or requirement elicitation techniques.

REFERENCES

- [1] M. Aoyama. 2007. Persona-Scenario-Goal Methodology for User-Centered Requirements Engineering. In *Proceedings of RE 2007*. 185–194.
- [2] C. Uchida et al. 2016. GO-MUC: A Strategy Design Method Considering Requirements of User and Business by Goal-Oriented Measurement. In *Proceedings of CHASE 2016*. ACM, 93–96.
- [3] D. Ameller et al. 2016. A Survey on Software Release Planning Models. In *Product-Focused Software Process Improvement*. Springer International Publishing, 48–65.
- [4] J. Cleland-Huang et al. 2013. A Persona-based Approach for Exploring Architecturally Significant Requirements in Agile Projects. In *Proceedings of REFSQ ’13*. Springer Berlin Heidelberg, 18–33.
- [5] K. Rönkkö et al. 2004. Personas is Not Applicable: Local Remedies Interpreted in a Wider Context. In *Proceedings of PDC 04*. ACM, 112–120.
- [6] L. Lehtola et al. 2009. Linking business and requirements engineering: Is solution planning a missing activity in software product companies? *Requirements Engineering* 14, 2 (2009), 113–128.
- [7] P. Achimugu et al. 2014. A systematic literature review of software requirements prioritization research. *Information and Software Technology* 56, 6 (2014), 568–585.
- [8] T. Matthews et al. 2012. How Do Designers and User Experience Professionals Actually Perceive and Use Personas?. In *Proceedings of CHI ’12*. ACM, 1219–1228.
- [9] W. Maalej et al. 2016. Toward Data-Driven Requirements Engineering. *IEEE Software* 33, 1 (2016), 48–54.
- [10] X. Zhang et al. 2016. Data-driven Personas: Constructing Archetypal Users with Clickstreams and User Telemetry. In *Proceedings of CHI ’16*. ACM, 5350–5359.
- [11] Y. Watanabe et al. 2017. ID3P: Iterative Data-driven Development of Persona Based on Quantitative Evaluation and Revision. In *Proceedings of CHASE 2017*. IEEE, 49–55.
- [12] E. Friess. 2012. Personas and decision making in the design process. In *Proceedings of CHI ’12*. ACM, 1209.
- [13] J. Pruitt and J. Grudin. 2003. Personas: Practice and Theory. In *Proceedings of DUX ’03*. ACM, 1.
- [14] M. Rahimi and J. Cleland-Huang. 2014. Personas in the Middle: Automated Support for Creating Personas As Focal Points in Feature Gathering Forums. In *Proceedings of ASE ’14*. ACM, 479–484.
- [15] O. Rissanen and J. Münch. 2015. Transitioning Towards Continuous Delivery in the B2B Domain: A Case Study. In *Agile Processes in Software Engineering and Extreme Programming*, Casper et al. Lassenius (Ed.). Springer International Publishing, 154–165.