Proposal of In-house Development Model for Business System at Kagawa University

Satoru Yamada
Graduate school of Engineering
Kagawa University
Kagawa, Japan
email: yamada.satoru@kagawa-u.ac.jp

Takamasa Yatani
Graduate School of Science
for Creative Emergence
Kagawa University
Kagawa, Japan
email: s22g363@kagawa-u.ac.jp

Norifumi Suehiro
Graduate School of Science
for Creative Emergence
Kagawa University
Kagawa, Japan
email: suehiro.norifumi@kagawa-u.ac.jp

Horoki Asakimori
Information Technology
and Media Center
Kagawa University
Kagawa, Japan
email: asakimori.hiroki@kagawa-u.ac.jp

Yusuke Kometani
Information Technology
and Media Center
Kagawa University
Kagawa, Japan
email: kometani.yusuke@kagawa-u.ac.jp

Rihito Yaegashi
Information Technology
and Media Center
Kagawa University
Kagawa, Japan
email: yaegashi.rihito@kagawa-u.ac.jp

***Abstract*—The development of a new system or product service is not a sure thing. A new development method that identifies the Minimum Viable Product (MVP) and starts the development of a system or service is attracting attention. We propose a In-house Development Model to identify the MVP of a business system and develop the system in-house by the themselves.**

***Index Terms*—Digital Transformation, Agile Development, User-driven Development, In-house Development**

I. Introduction

The impact of COVID-19 is changing the global social structure. Digital transformation is necessary to adapt to change. The information system plays an important role in promoting DX of User Companies. However, user companies have the problem of ”starting the development of an information system with unclear requirements”. Agile development of the information system in user companies using ”Low-code/No-code tools” is attracting attention as a way to promote digital transformation.

”Design Thinking” [1] was proposed at the HASSO PLATTNER Institute of Design at Stanford. ”Design Thinking” consists of five steps: ”EMPATHIZE”, ”DEFINE”, ”IDEATE”, ”PROTOTYPE”, and ”TEST”. Fig. 1. is an example of the ”Design Thinking” process. ”Design Thinking” is a necessary concept for creating new value. Many companies are implementing initiatives based on ”Design Thinking”.

”Lean Startup” [2] was proposed as a methodology for launching a business under conditions of high uncertainty. A Minimum Viable Product(MVP) is developed in ”Lean Startup”. It provide users with MVPs based on hypotheses and define value by ”Verification” with them through the ”Build-Measure-Learn” cycle. ”Design Thinking” realizes ”Human-Centered” value delivery. However, ”Lean Startup” emphasizes the verification of business feasibility based on hypotheses.

Kagawa University defined ”Hypotheses” for business system requirements based on a ”Human-Centered” (Ex. faculty, staff, and students) approach. The ”Hypothesis” is ”verified” through co-creation with users (Ex. faculty, staff, and students). Kagawa University proposes the ”In-house development model for Business System at Kagawa University” in which ”Human-Centered” business system requirements are defined as ”Hypotheses”, and ”Verification” is conducted through co-creation with users. The ”In-house development model for Business System at Kagawa University” combines ”Design Thinking”, and ”Lean Startup”. Kagawa University is currently developing a business system using the ”In-house development model for Business System at Kagawa University”. This paper describes the ”In-house development model for Business System at Kagawa University”. The ”In-house development model for Business System at Kagawa University” is based on the iterative model of agile development. Development is done in phases. The iterative model of agile development in general aims to increase the product quality [4] of the system. However, the ”In-house development model for Business System at Kagawa University” defines a ”Hypothesis” that enhances the quality of usability. Then, the users (faculty, staff, and students) themselves develop the business system by repeating ”PROTOTYPE”, and ”TEST” through co-creation of the ”Hypothesis”. Therefore, the iterative model differs from the general agile development iterative model, which enhances ”product quality”. The ”In-house development model for Business System at Kagawa University” focuses on ”user value” rather than ”product quality”. Several methods for defining hypotheses, such as ”Design Thinking”, have been proposed. Therefore, this paper does not limit the methods for defining ”Hypotheses”.

Section II describes related research and related technolo-

 ISBN: 978-1-68558-014-8

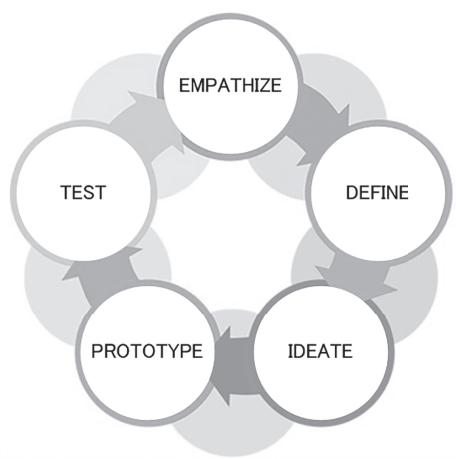

Fig. 1. Process of Design Thinking.

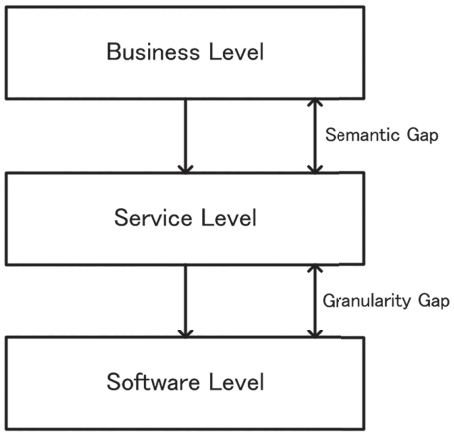

Fig. 2. User-driven development approach which Chusho [5] Proposes.

gies. Section III describes an ”In-house development model for Business System at Kagawa University”. Section IV provides a Results. Section V provides a Conclusion.

II. Related Research and Related Technologies

Chusho [5] says ”End users (defined as users in this paper) who have knowledge of the business develop systems and software on their own initiative. It is also important that users take the lead in maintenance”. The user-driven development proposed by Chusho is a three-tier architecture: ”Business Level”, ”Service Level”, and ”Software Level”. Fig. 2. shows the user-driven development approach proposed by Chusho. At the ”Business Level”, users with business knowledge create business models. At the ”Service Level”, create a domain model based on the ”Business Model”. Software is developed at the ”Software Level” from the created domain model. Chusho says, ”A semantic gap is created between the business level and the service level. Domain Knowledge Complementary Technology is a technology that complements the semantic gap”. Kato et al. [6] proposed a request acquisition method named THEOREE. The method proposed by Kato et al. systematizes domain knowledge by means of a thesaurus, which improves the efficiency of requirements analysis by providing a systematic thesaurus to analysts who lack sufficient domain knowledge. Kato et al.’s research falls under the category of domain knowledge completion technology. At the ”Software Level” systems and services are developed by utilizing components. Chusho says, A Software Unit Gap is created between the Service Level and the Software Level. ”Business Objects” [7] , ”Design Patterns” [8], and ”Frameworks” [9] are complementary technologies to the Software Unit Gap. The smaller the description unit of a program, the greater the scope of application because it can be expressed in a manner similar to a programming language. However, if the Software Unit is made larger and expressed in a business-like manner, it will be easier for users to use, but the scope of application will be limited.

With the development of information and communication technology, End-User Computing(EUC) [10] with ”Low-code/No-code tools” that enable system and software development without advanced programming knowledge is attracting attention. ”Low-code/No-code tools” have been introduced for use in DX promotion as a means to respond to the ”ambiguity of needs”, and ”rapidly changing requirements” for system and software development [11]. In addition, development using ”Low-code/No-code tools” is expected to significantly reduce development man-hours and shorten the time to ”Verification” of the MVP. Therefore, there is little Software Unit Gap between requirements and deliverables, and it has been reported that it is effective in developing systems and software with specific MVPs [12].

III. In-house development model for Business System at Kagawa University

Fig. 3. shows the ”In-house development model for Business System at Kagawa University” proposed in this paper. Chusho showed that business knowledge is important for users to develop systems and software that they themselves need, and proposed a three-tier architecture (”Business Level”, ”Service Level”, and ”Software Level”). Kagawa University integrated the ”Service Level” into the ”Software Level” by utilizing ”Low-code/No-code tools” based on the tree-tier architecture proposed by Chusho. In order to emphasize the definition of ”Hypothesis” for the realization of ”Human-Centered” value and the ”Verification” of MVP, we defined a three-step approach (”Business level”, ”Software level”, and ”Verification level”) with a ”Verification Level” to ”Evaluate” the developed system or software. By iteratively repeating this three-step approach multiple times, users themselves develop the systems and software they need. In this paper, the ”Low-code/No-code tool” was used to integrate the ”Service Level”, and ”Software Level”. However, if software can be developed without any granularity gap between the ”Service Level”, and the ”Software Level,” there is no need to use ”Low-code/No-code tools.

 ISBN: 978-1-68558-014-8

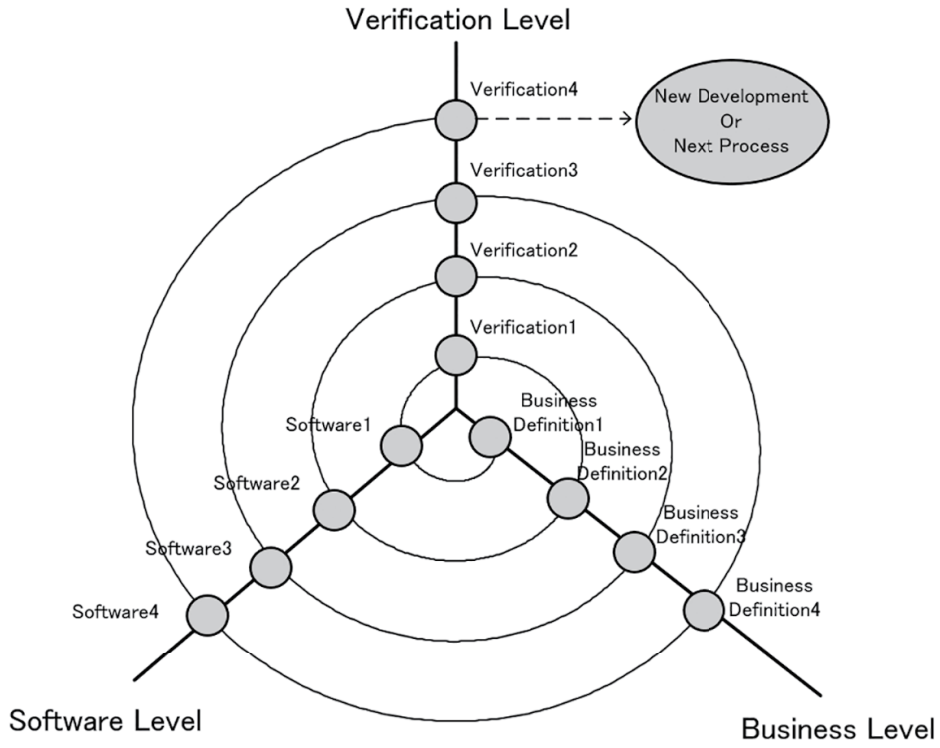

Fig. 3. In-house development model for Business System at Kagawa University.

In the ”In-house development model for Business System at Kagawa University” business is defined at the ”Business Level”. At the ”Business Level”, the ”Hypothesis” for the business system is defined and MVPs are identified through co-creation with users (faculty, staff, and students) who have business knowledge. At the ”Software Level”, systems and software are developed based on business definitions. At the ”Software Level”, MVPs are developed from hypotheses defined at the ”Business Level”, utilizing ”Low-code/No-code tools”. Developed systems and software are evaluated at the ”Verification Level”. At the ”Verification Level”, not only is the product quality of the system or software evaluated based on the business definition defined at the ”Business Level”, but also the validity of the ”Hypothesis” or MVP for the user value defined at the ”Business Level” is evaluated. At the ”Verification Level”, the continuation of development is also discussed. If the decision to continue development is made at the ”Verification Level”, the business definition is modified or added at the ”Business Level”. Develop improved systems and software at the ”Software Level” based on the reviewed business definitions. If a decision to terminate development is made at the ”Verification Level”, development is terminated. In the ”In-house development model for Business System at Kagawa University”, another system development or new needs may be discovered through system or software development. After the system development is completed, a new development project is launched. This paper does not define how to define a ”Business Model”, and how to generate a ”Software Level”, and how to evaluate a ”Validation Level”. In this paper, it is assumed that the method to be used is to select the necessary method according to the target business and the system or software to be developed.

IV. Results

Kagawa University established ”DX Promotion Division”, and ”DX Laboratory” in May 2021. In the ”DX Laboratory”, IT and business divisions collaborate to develop business systems in-house through co-creation. The ”DX Laboratory” works in the ”DX Project Team”. The ”DX Project Team” consists of users (faculty, staff, and students). Users with business knowledge from the business departments participate in the ”DX Project Team”. The ”DX Project Team” defined a ”Hypothesis” and identifies an MVP based on the ”In-house development model for Business System at Kagawa University”. In October 2021, there were six ”DX Project Teams”. The six ”DX Project Teams” have developed twenty-five projects business system in-house. Fifteen projects have already been completed. Ten business systems were developed in five months. At the ”Software Level”, the software was developed using the ”Microsoft Power Platform” [13], a ”Low-code/No-code tools”. The ”Microsoft Power Platform” includes four services: ”Microsoft Power Apps” [14], ”Microsoft Power Automate” [15], ”Microsoft Power BI” [16], and ”Microsoft Power Virtual Agents” [17].

Using the ”In-house development model for Business System at Kagawa University”, we interviewed the staff who

 ISBN: 978-1-68558-014-8

developed the business system. There are five questions. Question 1: Do you feel a ”Semantic Gap” from the ”Business Level” to the ”Software Level”? Question 2: Do you feel a ”Granularity Gap” at the ”Software Level”? Question 3: An impression of the use of ”In-house development model for Business System at Kagawa University”. Question 4: An impression of ”Design Thinking” and co-creation activities. Question 5: An overall impression.

All four respondents answered ”no Semantic Gap” for Question 1. The reason for this was that ”staff members who understand the work develop software at the ’Software Level’, so they do not feel a ’Semantic Gap’”. All four respondents answered ”no Granularity Gap” for Question 2. The business system is a flow definition using ”Microsoft Power Automate” with ”Low-code/No-code tools”. Therefore, I do not feel any ”Granularity Gap”. The respondents to Question 3 answered, ”Until now, we could not implement a system without ordering from a vendor, but now we can implement a system with a sense of speed”, ”We can implement a system that we really think is necessary”, and ”The larger the scale of the system, the more difficult it is for end users to develop”. The respondents to Question 4 answered, ”It was easier to share specific issues”, and ”The motivation of the business units made a difference in the results”. The respondents to Question 5 answered, ”the data obtained from the system is useful”, ”I want to improve the system based on the data”, and ”reviewing the operations gave me an opportunity to think about whether the operations were necessary”. The interview results indicate that the ”In-house development model for Business System at Kagawa University”, has the potential to solve the ”Semantic Gap”, and ”Granularity Gap”.

V. Conclusion

In this paper, we define a ”Hypothesis” for the realization of ”Human-Centered” value. The ”In-house development model for Business System at Kagawa University” in which business systems are developed by ”Verification” of the defined ”Hypothesis” through co-creation with users, was described. The ”In-house development model for Business System at Kagawa University” combines ”Design Thinking” and ”Lean Startup”. Through ”co-creation” between the IT and business divisions, ”Hypotheses”, and MVPs for the realization of ”Human-Centered” value can be identified, and business systems using EUC can be produced in-house using ”Low-code/No-code tools”. The ”In-house development model for Business System at Kagawa University” can define MVP by three steps: ”Business Level”, ”Software Level”, and ”Verification Level”. The ”In-house development model for Business System at Kagawa University” has the potential to solve the problem of ”starting development with unclear requirements” for user companies working to promote DX.

Using the ”In-house development model for Business System at Kagawa University”, an interview survey was conducted with university employees who have developed their business systems in-house. One comment was, ”It is difficult to judge what end-users can and cannot develop with EUC”. And, Some projects were terminated because the ”Hypothesis”, or ”MVP” could not be verified at the ”Validation Level”. The future work is to clarify the conditions under which end-users can participate in development and to establish guidelines.

References

[1] HASSO PLATTNER Institute of Design at Stanford, ”An Introduction to Design Thinking PROCESS GUIDE,” https://web.stanford.edu/~mshanks/MichaelShanks/files/509554.pdf [retrieved: July, 2022].

[2] Eric Ries, ”The Lean Startup.” Portfolio Penguin, 2011.

[3] National Strategy office for Information and Communications Technology Cabinet Secretariat, ”Guidebook of Agile Development Practice,” https://cio.go.jp/sites/default/files/uploads/documents/Agile-kaihatsu-jissen-guide_20210330.pdf [retrieved: July, 2022].

[4] Information-technology Promotion Agency, Japan, ”Guidebook of Software Quality in a Connected World,” https://www.ipa.go.jp/files/000055008.pdf [retrieved: July, 2022].

[5] Takeshi Chusho, ”Enduser-Initiative Application Development Methods with Business Knowledge,” http://www.isc.meiji.ac.jp/~chusho/paper/1611KBSEchu.pdf [retrieved: July, 2022].

[6] Junzo Kato, Motoshi Saeki, Atsushi Ohnishi, Haruhiko Kaiya, and Shuichiro Yamamoto, ”THEOREE: THEsaurus Oriented REquirements Elicitation,” Information Processing Society of Japan, pp.1-17, 2009.

[7] Rockford Lhotka, ”Visual Basic 6.0 Business Objects,” Apress, 1998.

[8] Gamma Erich, Helm Richard, Johnson Ralph, Vlissides John, and Grady Booch, ”Design Patterns: Elements of Reusable Object-Oriented Software,” Addison-Wesley Professional, 1994.

[9] Takeshi Chusho, ”Software engineering (3rd edition),” Asakura Shoten, 2014.

[10] Howie Goodell, ”End-user computing,” CHI EA ’97, Mar. 1997.

[11] Information-technology Promotion Agency Japan, ”DX White Book 2021,” https://www.ipa.go.jp/files/000093706.pdf [retrieved: July, 2022].

[12] Microsoft Corporation, ”Differences between Power Apps and traditional app development approaches,” https://docs.microsoft.com/en-us/power-apps/guidance/planning/app-development-approaches [retrieved: July, 2022].

[13] Microsoft Corporation, ”Microsoft Power Platform,” https://powerplatform.microsoft.com/ja-jp/[retrieved: July, 2022].

[14] Microsoft Corporation, ”Microsoft Power Apps,” https://powerapps.microsoft.com/ja-jp/[retrieved: July, 2022].

[15] Microsoft Corporation, ”Power Automate,” https://powerautomate.microsoft.com/ja-jp/[retrieved: July, 2022].

[16] Microsoft Corporation, ”Microsoft Power BI,” https://powerbi.microsoft.com/ja-jp/[retrieved: July, 2022].

[17] Microsoft Corporation, ”Microsoft Power Virtual Agents,” https://powervirtualagents.microsoft.com/ja-jp/[retrieved: July, 2022].

 ISBN: 978-1-68558-014-8