SMA 2021

The 10th International Conference on Smart Media & Applications

Gunsan Saemangeum Convention Center
and Kunsan National University,
Gunsan-si, Jeollabuk-do, Republic of Korea

Smart Media
KOREAN INSTITUTE OF SMART MEDIA

In-Cooperation with
ACM SIGAPP

Pre-proceedings of

The 10th International Conference on Smart Media & Applications (SMA2021)

September 9-11, 2021 | Gunsan Saemangeum Convention Center and Kunsan National University, Gunsan, Republic of Korea

Sponsored by

Hosted by

KSNU KUNSAN NATIONAL UNIVERSITY

10th ANNIVERSARY Smart Media KOREAN INSTITUTE OF SMART MEDIA

In-cooperation with

SIGAPP

Hosted by KSNU

SPONSORS

Refined Code Extraction based on Creative Thinking

ChaeYun Seo
Dept. of Internet Information Communication, Korea International University in Ferghana, Uzbekistan
chaeyun@kiuf.uz

JangHwan Kim
SE Lab., Dept. of Software and Communication Engineering, Hongik University, Korea
janghwan@selab.hongik.ac.kr

Bo Kyoung Park
Dept. of Computer Education, ChinJu National University of Education, Korea
parkse@cue.ac.kr

R. Young Chul Kim
SE Lab., Dept. of Software and Communication Engineering, Hongik University, Korea
bob@hongik.ac.kr

ABSTRACT

In the new era of the Fourth Industrial Revolution, we need smart software at diverse area. Until now, software engineers deal with quality of software without considering creative thinking. For new smart software, we must attempt how to adapt software engineering with creative thinking to develop smart software. Not only development professionals but also non-major developers must engage in development with their creative ideas. Even non-professional developers with creative thinking, who do not have enough knowledges of software development, should easily develop software. To solve these issues, , we propose a refined code extraction method creative thinking. Through this method, we expect the contribution of software development with/without software engineering knowledge.

KEYWORDS

Software Engineering, Creative Thinking, Code Education Process, Refined Requirement.

1. Introduction

Throughout the Fourth Industrial Revolution, we are living in a flood of new software. As the paradigm of the industry is digitized and advanced, the software development environment is changing. Even experts in other fields, not developers, continue to develop and test the software product or prototype of software. As software development progresses through these methods, regular software development still requires difficult work and knowledge levels, making it difficult for those non-developers to enter. They also have a lot of ideas but are unable to begin development due to difficulties in defining and designing software functions, implementing them, and so on. In this paper, we propose ways to design a system design based on creative thinking mechanism. The design of software is a critical issue in software development that determines the completeness of the software.

This paper will be consisted of following order. In chapter 2, we will refer to traditional design thinking methodologies such as 5W1H, etc., as well as methodologies that incorporate design thinking and Abbot's proposed heuristic model as a related study. Chapter 3 describes the proposed method for the refined code extraction based on creative thinking. Chapter 4 describes how to generate the code structure from the result of proposed method. Finally, in chapter 5, we will remark the conclusion of the research and research that we plan for the future.

2. Related works

- STEP 1: 'Empathize' of Design Thinking

To identify the problem, we need to empathize with the problem we are trying to solve through user's problem.

Table 1: Abbot's heuristics for mapping parts of speech to model components [2]

Part of speech	Model component	example
Proper noun	Instance	Alice
Common noun	Class	Field officer
Doing verb	Operation	Creates, submits, selects
Being verb	Inheritance	Is a kind of, is one of either
Having verb	Aggregation	Has, consists of, includes
Modal verb	Constraints	Must be
Adjective	Attribute	Incident description

'Empathize' is important in human-centered design processes such as design thinking [1]. Table 1 shows Abbot's heuristic for mapping part-of-speech to model components [2].

Table 2: Abbot's heuristics for mapping parts of AEIOU

Part of speech	Door Lock System
Proper noun	User
Common noun	System Name
Doing verb	Activity
Being verb	Environment
Having verb	Interaction
Modal verb	Object

Table 2 shows a mapping of Abbott and AEIOU. Common maps to the system name, and Proper noun maps to the User of AEIOU. Because *'Doing verb'* represents system functionality, it maps to AEIOU's activity. Having verb represents a system component and maps it to AEIOU's Interaction. Adjective can be represented by system attribute values [3].

Methods for mapping AEIOUs to user requirements are as follows:

ACTIVITY- System Function
ENVIRONMENT- System Environment
INTERACTION- between system components
OBJECT- main functions of the system
USER – System User

We extract the necessary elements based on AEIOU methodology of Design Thinking Empathy Stage in Problem Statement [4].

- STEP 2: 'Define' of Design Thinking

In the second stage of design thinking, the 'Define' stage, the requirements, and problems of users are explained using the methodology of 5W1H. In the empathy stage, the information collected by the AEIOU method is used to define the problem through the 5W1H method and to define the user's requirements [5]. Among the 5W1H methods, the WHY item is not included as it is a concept that overlaps with the WHAT item. The method of mapping user requirements to problem definitions is as follows.

- WHAT-system function (with mapping on ACTIVITY)
- WHERE-system environment (with mapping on ENVIRONMENT)
- WHO-system user (with mapping on USER)
- WHEN-system objects with mapping on INTERACTION
- HOW-means of system function (with mapping on OBJECT)

In 5W1H, WHAT is a system function and is mapped to Activity among elements of AEIOU. WHERE is the system environment and is mapped with Environment among AEIOU elements. WHO is a system user and is mapped to User in the AEIOU element. WHEN is the interaction of system objects and is mapped with the interaction of AEIOU. HOW is a means of system function and is mapped to the object of AEIOU.

Table 2 shows the mapping relationship between elements of 4W1H and elements of AEIOU.

Table 3: 4W1H Method of 'Define' step

Empathy	Define
Activity	What
Environment	Where
Interaction	When
Object	How
User	Who

- STEP 3: 'Ideate' of Design Thinking

In the third stage of design thinking, the 'Ideate' stage, a system defined using mind map is designed [6]. When creating a mind map, put the name of the system to be developed in the middle part of the mind map. This is called the root node. A node extending to the right with respect to the root node can design a system component, that is, a system object. This system object is expressed as a rectangle, and the control object in the system object is expressed as a triangle on the rectangle. The control object can be expressed later as a message sequence diagram. A node extending to the right with respect to the root node can design a system component, that is, a system object. This system object is expressed as a rectangle, and the control object in the system object is expressed as a triangle on the rectangle. The control object can be expressed later as a message sequence diagram. A node extending to the left with respect to the root node is a system function. This system function is represented by a rectangle with rounded corners. You can design by extracting the functions used in the system through the mind map.

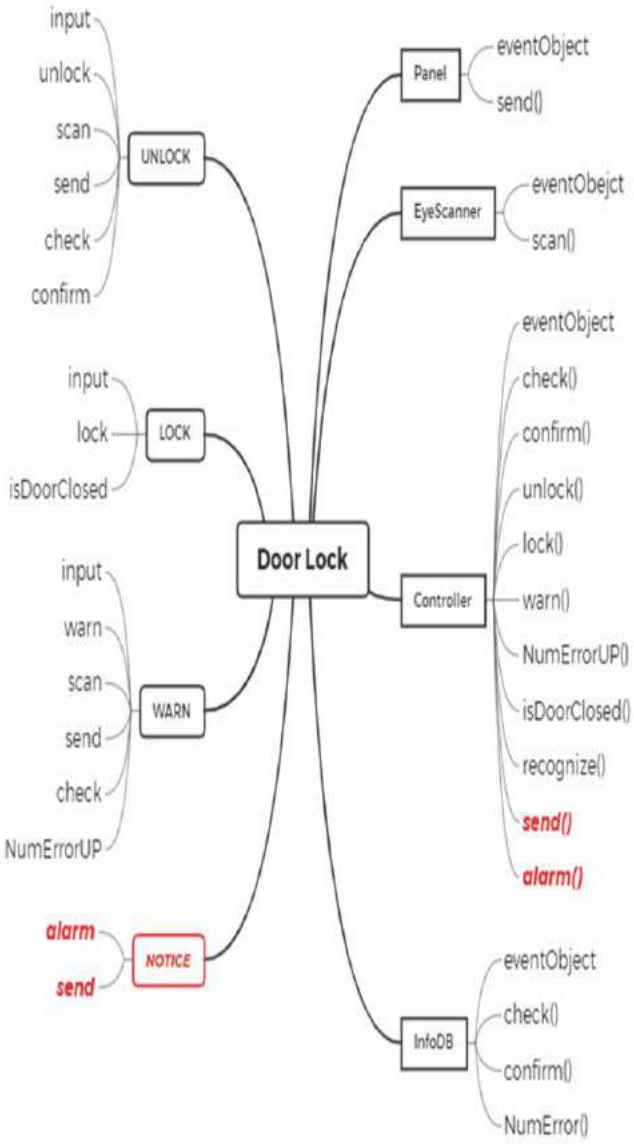

Figure 1: System functions and objects expressed as mind maps.

- STEP 4: 'Prototype' of Design Thinking

In the fourth stage of design thinking, the 'Prototype' stage creates each code template based on mind map [8]. The system name is set to class, and the ones represented on the right side of the mind map set to objects. The ones represented on the left side of the mind map are implemented by the method. Fig. 1 is a system object code template implemented on the right side of the mind map.

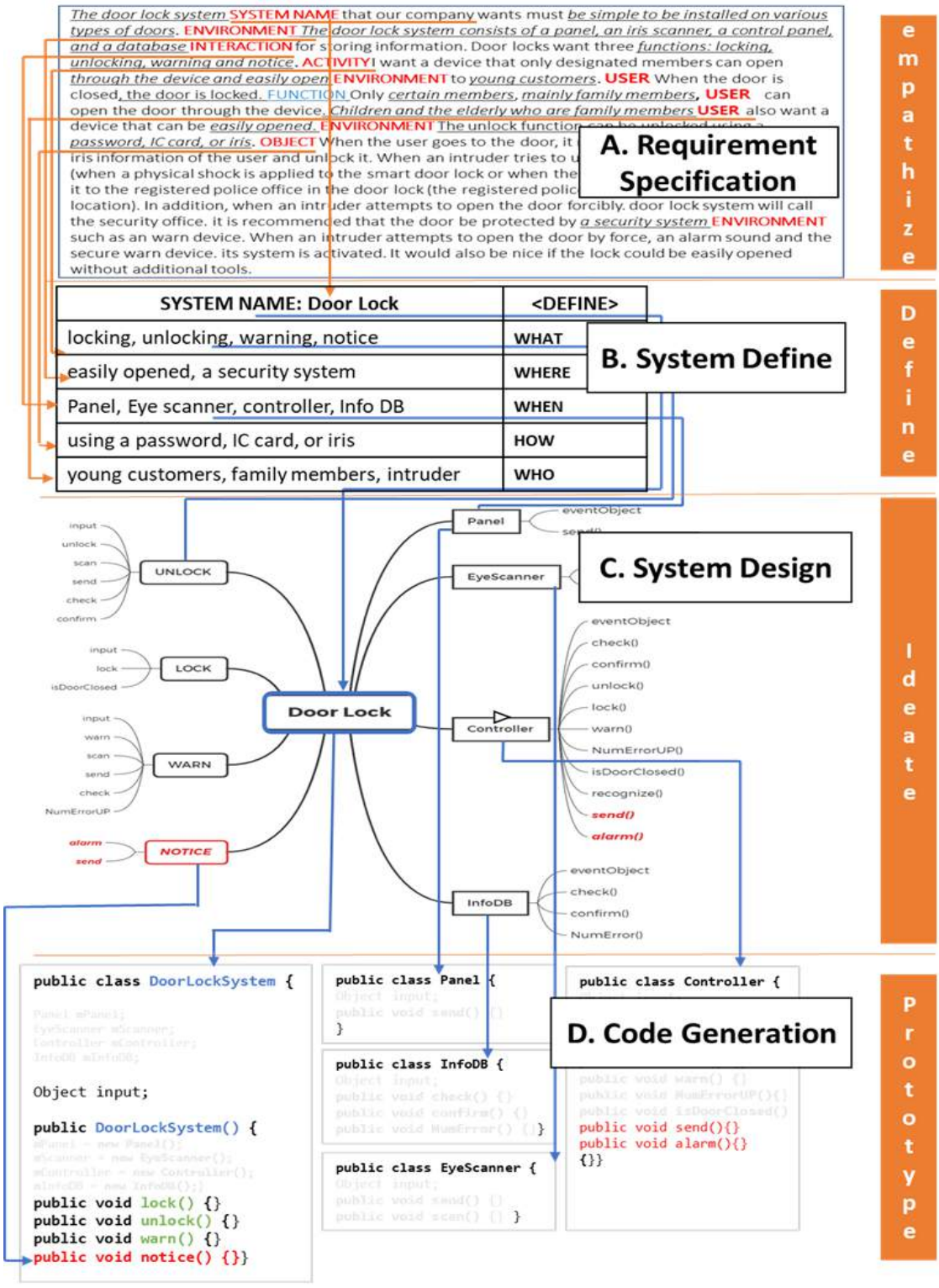

Figure 2: Refined Code Extraction based on Creative Thinking

Fig. 2 is a code template extracted from a system object. It is a system object code template designed to the right of the mind map.

3. Refined Code Extraction based on Creative Thinking

We look at a close architecture for designing systems by using creative thinking methodologies [7].

First, in the 'empathize' phase, we use AEIOU method to extract the elements that is necessary for the system.

A. Requirement Specification: Extract the AEIOU object of the empathy stage of design thinking.

The 'define' stage maps elements extracted from the previous stage and elements 4W1H to be used in this stage.

B. System Define: It defines the development of software systems 4W1H way.

These two phases establish the name of the development system and defines system functions, objects, methods, and interactions between objects.

In the 'ideate' phase, we design the system by utilizing the elements mapped in the define phase.

C. System Design: A mind map is designed for system objects and functions.

D. Code Generation: Objects and functions of systems designed with mind maps can extract skeleton codes.

4. Generating Code Structure from Requirement

System design using a mind map can be generated to code as follows:

```
public class DoorLockSystem {

                ……

Object input;

public DoorLockSystem() {
                ……
public void lock() {}
public void unlock() {}
public void warn() {}
public void notice() {}}
```

Figure 3: Code template representing the entire system

Fig. 3 shows a code template that is representing the entire system.

```
public class Panel {
                ……
}

public class InfoDB {
                ……
}

public class EyeScanner
{
                ……
 }
```

Figure 4: Code template extracted from System objects.

Fig. 4 describes a code template that is extracted from System object.

```
public class
Controller {

                ……


public void send(){}
public void alarm(){}
}
```

Figure 5: Code template for a control object

Fig. 5 is a code template for a control object. A code template for a control object on the right side of the system.

5. CONCLUSIONS

This paper is a study of the system design methodology based on creative thinking mechanism. This method can help those non-professionals develop on software developing by following this method to contribute the software development process. It includes the possibility of easier access to software development for the people through creative thinking and software engineering integration. In the future, the research about the automation of this methodology for more convenient will be conducted on methods using other creative thinking methodologies. This method may include the possibility of easier access to software development to the public through creative thinking and software engineering integration. In the future, automated research will be conducted on methods using creative thinking methodologies.

ACKNOWLEDGMENTS

The research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education(2021R1I1A305040711) and by the BK21 FOUR (Fostering Outstanding Universities for Research) funded by the Ministry of Education(MOE, Korea) and National Research Foundation of Korea(NRF)

REFERENCES

[1] https://www.interaction-design.org/literature/topics/design-thinking
[2] Bernd Bruegge, Allen H. Duttoit," Object-Oriented Software Engineering", PrenticeHall,2009.
[3] ChaeYun Seo, R. Young Chul Kim, "A study on mapping software engineering with design thinking mechanism", 2020KIPS Conference, 2020.5.
[4] H. E. Ju and J. S. Lee, "A Study of Open Collaboration Creative Thinking System Based on Design Thinking," Digital Design Studies, Vol.12, No.3, pp.179-190, 2012.
[5] H. J. Yoon, "Creative Thinking and Convergence Education for Innovative Design KCI," A Journal of Brand Design Association of Korea, Vol.11, No.4, pp.79-92, 2013.
[6] ChaeYun Seo, JangHwan Kim, R. Young Chul Kim, "Applied Practices on Codification Through Mapping Design Thinking Mechanism with Software Development Process", KTCCS. 2021.4.
[7]] Y. H. Shin, H. J. Jung, and J. S. Song, "Analysis of Learning Experience in Design Thinking-Based Coding Education for SW Non-Major College Students," Journal of Digital Contents Society, Vol.20, No.4, pp.759-768, 2019
[8] H. S. Shin, H. J. Jung, and J. S. Song, "Analysis of learning experience in design thinking-based coding education for SW non-major college students," The Journal of Contents Computing, Vol. 20, No.4, pp.759-769, 2019.