# A Pattern Approach for Designing Application

Kohei Daimaru and Buntaro Kaji

IBM Japan
{daimaruru,cazy06}@gmail.com

**Abstract.** We propose to include a pattern approach in design-process for application. Pattern approach is typically used as reference of best practice. In this case, we use pattern of current operation as input to inspect requirement of next-new application. We tried case study and studied about those results.

**Keywords:** Design, Pattern Language, Application, Requirement.

## 1 Introduction

### 1.1 About Pattern Language

Originally, Pattern Language is theories about architectural design proposed by Christopher Alexander (born October 4, 1936 in Vienna, Austria).

Patterns are ways to describe best practices, explain good designs, and capture experience so that other people can reuse these solutions. When a designer is designing something, they must make many decisions about how to solve problems. A single problem, documented with its most common and recognized good solution seen in the wild, is a single design pattern. Each pattern has a name, a descriptive entry, and some cross-references, much like a dictionary entry. A documented pattern must also explain why that solution may be considered a good one for that problem, in the given context. [1],[2]

### 1.2 Patterns in Software

The concept of Pattern Language has been inherited in Software engineering. For example, “wiki” was made with Pattern Language and citation of architectural pattern.

Ward Cunningham developed first “wiki” (WikiWikiWeb in Portland, Oregon in 1994, and installed it on the Internet domain c2.com on March 25, 1995). He have adopted some pattern from Patterns in Urban Architecture. [3]

### 1.3 Leading Research about Patterns in Software Design

In papers, there is advanced research about Patterns in software design. [A Pattern Approach to Interaction Design] [4] shows outline of present of Patterns in perspective and example usage of patterns for interaction design.

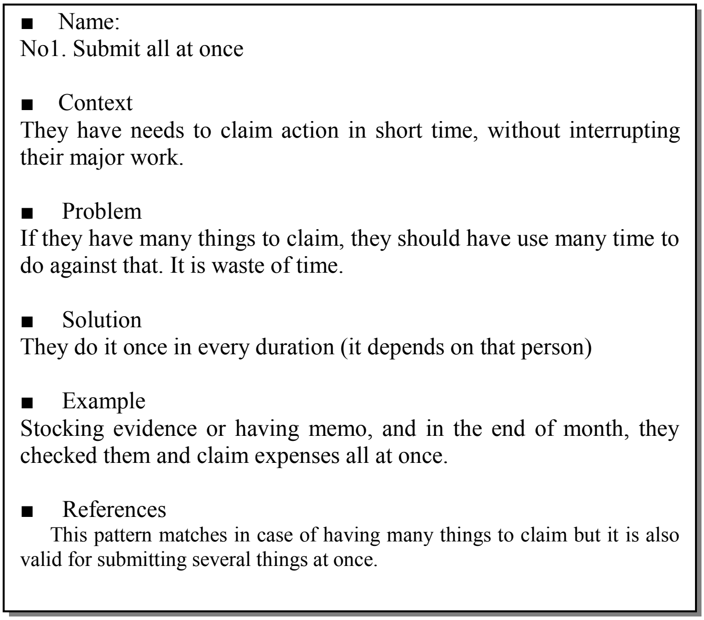
M. Kurosu (Ed.): Human Centered Design, HCII 2011, LNCS 6776, pp. 22–27, 2011.

## 2 Case Study

In this chapter, we present about case study. First, making Patterns of current operation about “Claim expenses” in 2.1. Second, extracting Application requirement from that patterns in 2.2. In the end, Showing proposal of Application form those requirement in 2.3.

### 2.1 Patterns of Current Operation

We have conducted hearing to current users who are using application system to claim expenses about that know-how, how to use it more efficiently. And we have made patterns of current usage from result of hearing.



■ Name:
No1. Submit all at once

■ Context
They have needs to claim action in short time, without interrupting their major work.

■ Problem
If they have many things to claim, they should have use many time to do against that. It is waste of time.

■ Solution
They do it once in every duration (it depends on that person)

■ Example
Stocking evidence or having memo, and in the end of month, they checked them and claim expenses all at once.

■ References
This pattern matches in case of having many things to claim but it is also valid for submitting several things at once.

**Fig. 1.** An Example of current operation’s Pattern (No1. Submit all at once)

And we classified Patterns into Category and User Roll. Category is property of Pattern. In this case, Category is composed of “Action” is know-how about operation and “Remind” is things to keep in mind. And There are two main User Roll those are “Applicant” and “Approver” as basic work-flow system.

**Table 1.** Mapping of current operation's Patterns

| Category＼User Roll | Applicant | Approver |
|---|---|---|
| Action: | 1Submit all at once | 6 Review for certain |
| | 2 Input same as evidence | - |
| | 3 Alert to rejection | - |
| | 4 Submit in fast | - |
| Remind: | 5 Alert to forget submitting | 7 Take care of pending things |

## 2.2 Outcome of Application Requirement from Current Requirement

We had an insight into Application Requirement from current operation's Patterns as a hypothesis, and listed them corresponding current requirement.

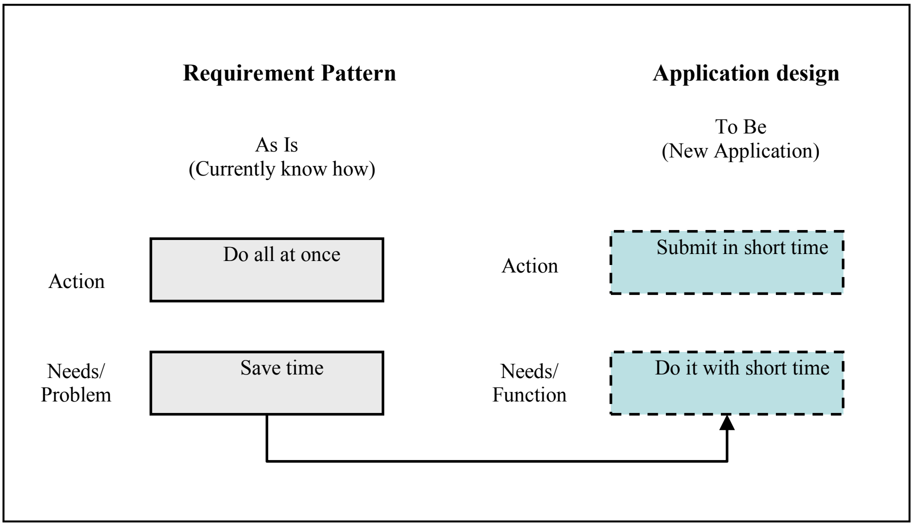


**Fig. 2.** Image of flow of deriving Current Requirement to Application Requirement

**Table 2.** Mapping of Current Requirement×Application Requirement

| Current Pattern | Application Requirement |
|---|---|
| 1 Submit all at once | 1. Submit plural things at a time |
| 2 Input same as evidence | 2. Automatic Validate function to check difference of Input form and evidence.<br>3. Automatic Input function same as evidence. |
| 3 Alert to rejection | 4. Display to alert about rejection |
| 4 Submit in fast | 5. Remainder or Alert for due date |
| 5 Alert to forget submitting | 6. Cross function with other relational system (For example, relation to )<br>7. Access by various platform and add and stock contents at once |
| 6 Review for certain | 8. Automatic Validate function to check difference of Input form and evidence and assist reviewing |
| 7 Take care of pending things | 9. Display to alert for pending things |

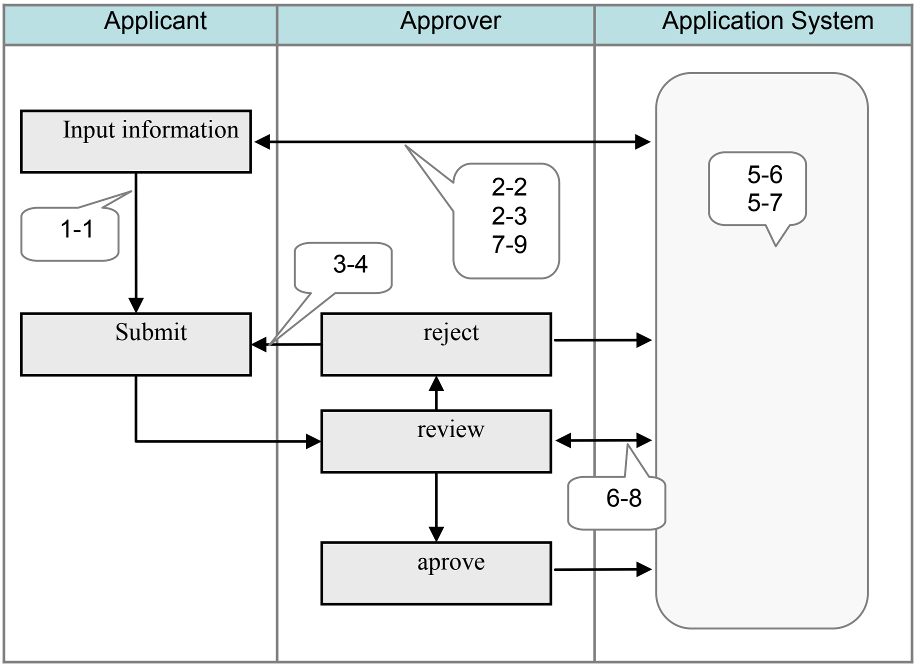
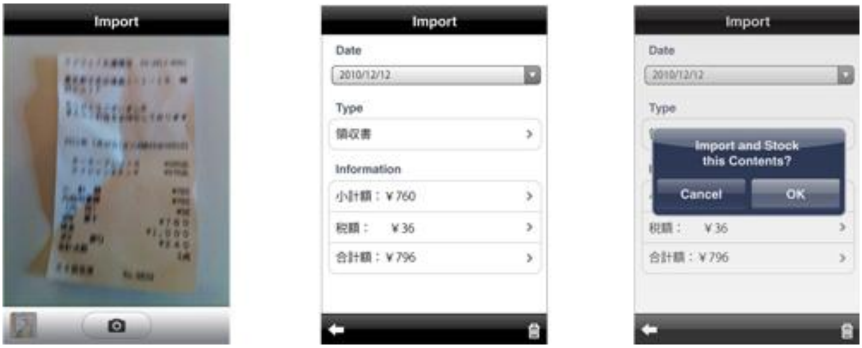


**Fig. 3.** Operation flow of Application with compatible Application Requirement

### 2.3 Designing of Application from Requirements

We designed application from Application Requirement. It is simple application that can access from web browser and local application.

Applicants can use the application with smart phone. And they can take a picture of till receipt and input automatically with OCR function. Approver can user widget of the Application and can operate of check things easily.



Date
2010/12/12
Type
領収書
Information
小計額：¥760
税額：　¥36
合計額：¥796

Import and Stock this Contents?

Fig. 4. Screen Image of Application for Applicant

## 3 Conclusions

### 3.1 Lookout in Result of Case Study

In Application system development project, we define Application requirement in first step (generally in requirement definition phase) by hearing or interview for clients or end users. We expect that we can discover more numerous requirements efficiently with pattern approach. Especially, potential user requirements are hard to discover from general hearing. Business analyst is usually familiar with current operation. If all project members share Pattern approach, it has possible to share knowledge of business side and development side and improve Application architecture and design.

### 3.2 Further Research

What scene can we adopt Pattern Approach efficiently? We expect it is effective when we can get voice of expert using current operation, lead user often add function or customize it for usage, often represents of potential needs. And typically in Business Application or own Application as services provided by themselves is adopted for that, we may have a tendency survey about users because we can figure out actual users. It also has potential for adopting to the scene when we plan new application by referring analog operation.

To improve Pattern approach for Applications, we should define details of how to extract patterns, how to derive Application requirements from patterns efficiently in short times. And in actual development project, it is essential to consider about cost and technical aspect so that we need to set priority of requirement from Pattern Approach and consider it with other entity. Application work flow is depends on type of operation. There is room to analyze operations of affairs and make framework of requirement by Application type.

## References

1. Alexander, C.: A Pattern Language. Kajima Institute Publishing Co., Ltd. (1977)
2. Alexander, C.: Timeless Way of Building. Kajima Institute Publishing Co., Ltd. (1979)
3. Eto, K.: Pattern, Wiki, XP ~ Timeless principles of creation (2009)
4. Borchers J.O.: A Pattern Approach to Interaction Design ( 2000)
5. Walkman 2.1 Development Background Wikipedia, (as of January 30, 2011), `http://ja.wikipedia.org/wiki/%E3%82%A6%E3%82%A9%E3%83%BC%E3%82%AF%E3%83%9E%E3%83%B3#.E9.96.8B.E7.99.BA.E7.B5.8C.E7.B7.AF`
6. Matsuo, H., Ogawa, S.: Innovating Innovation: The Case of Seven-Eleven Japan (2000)