

# **XRPWeb: Extensible Requirements Patterns for Web Application Development**

Romi Satria Wahono

Indonesian Institute of Science (LIPI)  
Department of Information and Computer Sciences, Saitama University

**Keywords:** web application, extensible requirements pattern, web engineering

**Abstract.** This paper proposes an approach for developing web applications by using extensible requirements pattern. This work is based on the facts where a lot of redundant works actually occurred in the web application development. Therefore, developing web applications from scratch is a time-consuming work, because actually we can reuse the similar basic requirements to develop web applications. By reusing the extensible requirements patterns where the general characteristic of web application requirements are stored, the redundancy problem of web application development can be solved and the web developer productivity can be increased.

## **Introduction**

There are a lot of redundant works actually occurred in web application development. We can investigate that a lot of web applications actually have similar basic requirements, although have very different in design. For example: registration form, online auction, homepage, guest book, online shopping, online newspaper, etc. Therefore, developing web applications from scratch is a time-consuming work, because actually we can reuse the similar basic requirements to develop web applications. If there are the requirements pattern catalogs where the general characteristic of the web application requirements are stored, we can decrease redundancy problem of web application development and increase developer productivity by reusing these patterns. The next problem is how to solve the validation problem in requirements pattern. However, it is also demanded the strategy to validate requirements patterns according to the customer feedback become possible.

The objective of this research is to solve the above-mentioned problems in web application development. We propose the extensible requirements patterns of web applications (*XRPWeb*) [5] for accelerating time to market, for solving redundancy problem and for supporting the standard practices of requirements engineering process in the web application development. In short, we propose an approach for more efficient web application development by using an extensible requirements pattern. However, requirements pattern should be designed to be extensible, in order to make the validation of requirements patterns according to the customer feedback become possible.

## **The Concepts and Foundations of *XRPWeb***

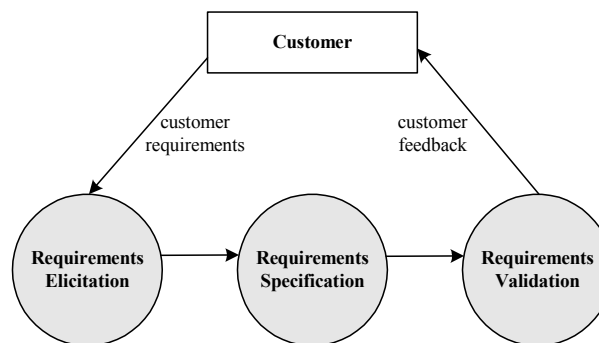
Nowadays, patterns are one of the latest hot topics in software development. The current use of the term pattern is derived from the writings of the architect Christopher Alexander who has written several books on the topics as it related to urban planning and building architecture [2][4]. Although Christopher Alexander was talking about architecture and urban planning, the patterns

that he talked about are applicable to many other disciplines, including software development.

The goal of patterns within the software community is to create a body of literature to help software developers resolve recurring problems encountered throughout all of software development. Patterns help create a shared language for communicating insight and experience about these problems and their solutions. Patterns readily available include [3]:

- Analysis Pattern
- Data Pattern
- Design Pattern
- Fundamental Pattern
- Message Pattern
- Transaction Pattern
- Requirements Pattern

Until now, no extensible requirements pattern existed in software engineering field, specifically for web application development. As shown in Figure 1, requirements engineering is an iteration process, which contains requirements elicitation, requirements specification, and requirements validation processes. Therefore, we think requirements patterns should be designed to be extensible, in order to support the system analysts to improve and to validate requirements specification according to the customer feedback when requirements engineering process is done.



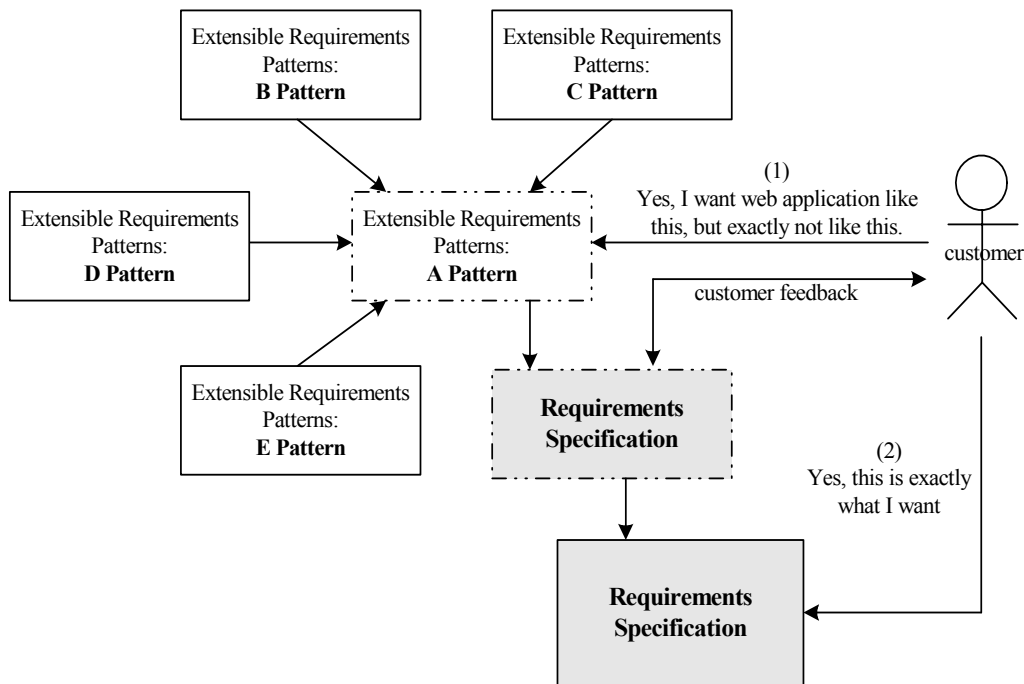
**Figure 1:** Requirements Engineering Process

In the web application development, customer needs are mainly to develop a web applications like “something” that have already known, but not exactly like it. We can say that because, the boom of Internet technology spread out to the lowest scale of the social community in the world, and it brings web applications become more popular application for us. The new ideas in the web applications commonly come from other already well-implemented systems, that we can reach them easily when surf in the web. In the requirements engineering, these phenomenon are known as the “Yes, But” syndrome (*That is what I meant, but not exactly what I meant*) and the “Undiscovered Ruins” syndrome (*Now that I see it, I have another requirement to add*).

As also shown in figure 2, the core idea of our research to propose the extensible requirements patterns of web applications are based on this phenomenon. By reusing, extending, revalidating and reconstructing the extensible requirements patterns, system analysts can create a requirements specification that match with what are exactly the customer needs.

We can conclude that the definition of extensible requirements pattern is as follows. An extensible requirement pattern is a reusable and an extensible framework for requirements that supports a system analyst to validate the customer feedback (and needs), by reusing and reconstructing requirements patterns. An extensible requirements pattern has some features related to the extensibility characteristic. Not like other existing patterns, an extensible requirements

pattern has two kinds of pattern elements: constraint elements and extensible elements.



**Figure 2:** The Concepts of Extensible Requirements Patterns and “Yes, But” Syndrome in Requirements Engineering

## Pattern Elements

Every pattern we define must be formulated in the form of a rule, which establishes a relationship between a context, a system of forces which arises in that context, and a configuration, which allows these forces to resolve themselves in that context [2]. There are many stylistic variants for describing pattern elements. Despite the use of these differing pattern formats, it is generally agreed that a pattern should contain certain essential elements. Some existing patterns books agreed that the essential elements of pattern are as follows: pattern name, problem, context, forces, solution, examples, resulting context, rationale, related patterns, known uses.

We argue that the extensible requirements pattern must contains the above mentioned essential elements that have constraint characteristic, and also contains the original elements for supporting extensibility characteristics. We designed the elements of extensible requirements pattern are as follows:

- Constraint Elements:  
pattern name, problem, context, examples, forces, related patterns,
- Extensible Elements:  
solution, user behavior, page behavior, prototype, data behavior, architecture, security

In the constraint element, the validation and modification are not enabled. And the other hand, the extensible elements are designed to have the validation and the reconstruction functionalities, in order to support a system analyst for eliciting the customer feedbacks (needs).

We can interpret the elements of extensible requirements patterns easily by using algorithm shown in Figure 3. The detailed explanation about the elements of extensible requirements patterns is as follows. Some definitions of the common pattern elements are extracted from [4].

### *Pattern Name*

Pattern names are extracted from the conceptual abstractions of the basic requirements of web

applications. Pattern names also provide a classification of the pattern in addition to its name.

#### *Problem*

A statement of the problem which describes its intent: the goals and objectives it wants to reach within the given context and forces. Often the forces oppose these objectives as well as each other.

#### *Context*

The preconditions under which the problem and its solution seem to recur, and for which the solution is desirable. This tells us the pattern's applicability. It can be thought of as the initial configuration of the system before the pattern is applied to it.

#### *Forces*

A description of the relevant forces and constraints and how they interact with one another and with goals we wish to achieve. Forces reveal the intricacies of a problem and define the kinds of trade-offs that must be considered in the presence of the tension or dissonance they create.

#### *Solution*

The solution describes static structure and dynamic behavior. The static structure tells us the form and organization of the pattern, but often it is the behavioral dynamics that make the pattern come alive. The description of the pattern's solution may indicate guidelines to keep in mind when attempting a concrete implementation of the solution.

#### *Examples*

Examples help the reader understand the pattern's use and applicability. An example may be supplemented by a sample implementation to show one way the solution might be realized.

#### *Related Patterns*

Related patterns share common forces. They also frequently have an initial or resulting context that is compatible with the resulting or initial context of another pattern.

#### *User Behavior*

User behavior element contains information about who are the users of web applications. It can be user, customer, administrator, etc.

#### *Page Behavior*

Page behavior element contains information about the behaviors of users related and defined at the user behavior element. Some browse functionalities and expected user navigation are also included in this element.

#### *Prototype*

Prototype is described using use-case model. The use-case model describes the proposed functionality of the new system. A use-case represents a discrete unit of interaction between a user (actor) and the system (use case). So, the prototype element consists of actor (the user that defined at the user behavior element) and use case (the user task that defined at the page behavior element).

#### *Data Behavior*

Data behavior element contains information about what kinds of database we should prepare.

#### *Architecture*

The information about system architecture is provided in the architecture element. What servers are necessary to built the application is also included in this element.

#### *Security*

Security element contains information about security level of web applications.

## **Pattern Catalogs**

We proposed some extensible requirements patterns based on the general characteristics and behaviors of web applications. Recently, the 30 extensible requirements patterns are available. These patterns are organized into the following categories.

### *Informational Patterns*

Informational Patterns are concerned with how an application can provide information to user about the contents or services. These informational patterns are at the core of all web applications, because actually the basic role and characteristic of web applications is as information provider for user. There are no functionalities for user action and input. Therefore, we can say that these patterns only provide informational functionalities from web applications to user. Online eBooks Pattern, Customer Support Pattern, Product Catalog Pattern, Personal Page Pattern, etc are included in this category.

### *Interactive Patterns*

Interactive Patterns are concerned with how an application can provide information to user, and also provide the functionalities for getting feedback from user. So, we can say that these patterns do not only provide informational functionalities from web applications to user, but also have interactive functionalities (user inputs, user actions, etc) from a user to web applications. Registration Forms Pattern, Guest book Pattern, Online Games Pattern, and Web Portals Pattern are included in the Interactive Patterns.

### *Transactional Patterns*

Transactional Patterns are concerned with an application that supports transactional actions between user and web applications. These patterns are particularly useful for developing an online shopping and an electronic banking service.

### *Workflow Patterns*

Workflow Patterns are concerned with an application that provides workflow functionalities. Online Planning and Scheduling Pattern, Inventory Management Pattern, and Status Monitoring Pattern are included in the Workflow Patterns.

### *Communities Patterns*

Communities Patterns are concerned with an application that supports the collaborations and the communications among people in the online communities. Communities Patterns contain Chat Groups Pattern, Recommender Systems Pattern, Online Marketplaces Pattern, Online Auctions Pattern, Bulletin Board System Pattern, and Online Organizer Pattern.

## **How to Use the Extensible Requirements Patterns**

We have already discussed the pattern elements in detail. In this Section, we will focus on how to use these patterns. However, we can easily interpret and use the extensible requirements patterns of web applications by using algorithm shown in Figure 3.

First of all, a system analyst captures requirements from customer. Then he goes to the pattern catalogs for investigating the categories and the extensible requirements pattern of web applications that near to the customer needs. System analyst can begin with finding the context of problem in context element. He can analyzes problem in problem element, and also refers to the example of the problem, in example element. If he thinks that all of the problem, context, examples, and forces are near with the customer needs, he has already prepared to go to the next step.

The next step is, to construct the solution of the problems. In this step, system analyst does an iteration process in order to construct the solutions, according to the customer feedback, until the final solution is decided. System analyst can applies and modifies the extensible elements of patterns including user behavior element, page behavior element, prototype element, data behavior element, architecture element and security element. He can also refer to other patterns, related patterns. The final results of this iteration process are the solution (the requirements) that matches with the customer needs.

```

IF {
    you find yourself in CONTEXT,
    for example EXAMPLES,
    with PROBLEM,
    entailing FORCES
}

THEN {
    DO (until the final SOLUTION is
    decided) {
        apply and modify {
            USER BEHAVIOR,
            PAGE BEHAVIOR,
            PROTOTYPE,
            DATA BEHAVIOR,
            ARCHITECTURE,
            SECURITY
        }
        refer to {
            new CONTEXT,
            RELATED PATTERNS,
            other PATTERNS,
        }
    }
}

```

**Figure 3:** How to Use and Interpret the Extensible Requirements Pattern of Web Applications

## Conclusions

In web applications development, reducing time to market and solving redundancy problem are a critical issues. We proposed the solutions for these problems using the extensible requirements patterns of web application. By reusing and reconstructing these extensible requirements patterns, we can accelerate time to market and solve redundancy problems in web applications development. In this paper, the concepts, the foundations, the characteristics and available catalogs of the extensible requirements patterns of web application are described. How to use and implement these extensible requirements patterns are also presented.

## References

- [1] Brad Appleton, "Patterns and Software: Essential Concepts and Terminology", Object Magazine Online, Vol. 3, No. 5, pp. 20-25, May 1997.
- [2] Christopher Alexander, "The Timeless Way of Building", Oxford University Press, 1979.
- [3] Patricia L. Ferdinandi, "A Requirements Pattern: Succeeding in the Internet Economy", Addison Wesley, 2002.
- [4] Christopher Alexander, Shara Ishikawa, Murray Silverstein, Max Jacobson, Ingrid Fiksdahl-King, and Shlomo Angel, "A Pattern Language: Towns, Bulding, Construction", Oxford University Press, 1977.
- [5] Romi Satria Wahono and Jingde Cheng, "Extensible Requirements Patterns of Web Application for Efficient Web Application Development", Proceedings of The First IEEE International Symposium on Cyber Worlds (CW 2002), pp. 412-418, IEEE Computer Society Press, Tokyo, Japan, November 2002.