An Enhancement of the Stakeholder Modeling Scheme for Requirements Engineering of Software Systems for Public

Jidtima Sunkhamani
Software Engineering Program, School of Informatics
Walailak University, Thailand
sjidtima@wu.ac.th

Abstract

Software systems for public are the systems initiated mainly for the benefits of a wide range of people. Such a system is initiated and its application domain and context should be focused early even before proposing it to development. Stakeholders are the main sources of software requirements and also domain knowledge. In order to include stakeholders in the requirements process judiciously, they have to be identified together with the reasons for their inclusion. This paper reports on an adaptation of Stakeholder Modeling Scheme to enhance it for the requirements engineering of software systems for public. The adapted scheme was applied to a case study, Krung-Shing Climate Monitoring system, a software system initiated for the benefits of the public. The results showed that the adapted scheme helped in stakeholder identification and requirements engineering of such a system.

Keywords: Stakeholder Modeling Scheme, Stakeholder Diagrams, Requirements Engineering, Software systems for public, Application domain study, Krung-Shing Climate Monitoring

1 Introduction

System stakeholders are parts of a system and its environment. They are the main source of domain knowledge and system requirements; thus, achieving good quality of requirements is impossible without involving stakeholders [1]. Many methods and techniques have been proposed to help requirements engineers in requirements elicitation. Applying the many existing methods and techniques does not warrant that the engineers will obtain requirements, which reflect the “real” needs unless the “right” stakeholders are identified and involved. However, including all or too many stakeholders is impractical because a project is always constrained by schedule and budget [2]. Hence, it is necessary that the engineers know why some specific stakeholders are more relevant or important to the system than the others. Without insightful information, the engineers may elicit requirements from the wrong stakeholders, approach the right ones but fail to elicit specific and clear requirements from each of them, and/or interpret what each stakeholder says or implies incorrectly.

There are cases that some systems are initiated, proposed, or recommended because of their potential for solving some specific problems of or bringing some opportunity to the public. Often they are technology- or market-driven systems. They may be initiated by engineers’ creativity or by someone’s interest in standard improvement of society. We then call such systems “software systems for public”.

The scope of a system for public can be unclear at the outset but it tends to be beneficial to a wide range of stakeholders rather than a particular organization. Hence, its stakeholders are likely to cross many organizations’ boundaries and may cover people in public. Before proposing such a system to a project sponsors, the initiator should study its application domain. Thus, its stakeholders must be identified as early as possible.

To date, many methods for stakeholder identification have been introduced. Sharp et al. suggested the engineers to explore the supplier, client, and network of 4 predefined classes of stakeholders: users, developers, legislators, and decision makers [3]. This method identifies stakeholders from the outset, so it works for systems having predefined and clear boundary. Gause and Lawrence proposed to use brainstorming to identify stakeholders [4]. Stakenet adopted Sharp’s method to acquire the initial set
of stakeholders’ roles and to have the identified stakeholders recommend other stakeholders similar to snowball technique [5]. Onion model defines a classification of stakeholders [6]. The method provides predefined stakeholder classes with no means to find who the real stakeholders are. When they are referred to as ‘users’ or ‘customers’ of a system, it is not clear who they really are in the subject domain of the system under study. All [3], [4], [5], and [6] were designed for studying the systems to serve the organizations that own them. Baléjos and Montagna presented an approach to stakeholder identification in inter-organizational environment. It first distinguished stakeholders to those internal and external to the firm and sub-classified them using 4 criteria, namely, function, geography, hierarchy, and knowledge before associated them with the roles they play [7]. Stakeholder roles were not provided in the method but general classes such as those defined in Onion model were suggested. Similarly, this approach was designed for the systems that have clear boundary and host organizations. Furthermore, all of these methods identify stakeholders without the information about why each stakeholder should be included in the process.

Sunkhamani [8] proposed Stakeholder Modeling Scheme for software systems engineering. The scheme also suggests that the proposed system belongs to a “focal organization”, the organization that owns the system as it hosts the development of the system. Nonetheless, unlike others, it offers a way to identify stakeholders alongside the reasons why they are the stakeholders of the system. It contains Stakeholder Modeling Language, enriched with the model elements enhancing the engineers to capture application domain knowledge while eliciting the requirements of stakeholders. However, the fact that the language comprises a rich set of model elements makes the modeling process time-consuming and the resulting model complex. To reduce the cost of applications for the study of a system at the early stage, a more compact process is needed.

This paper presents the enhancement of the Stakeholder Modeling Scheme for requirements engineering of software systems for public. It contains 6 sections. The Stakeholder Modeling Scheme is presented in Section 2 as background. Section 3 describes the adapted modeling language and process. Section 4 presents the result of the application to a case study. Discussion is given in Section 5, followed by conclusion in Section 6.

2 Background

2.1 Stakeholder Modeling Scheme

A stakeholder is “an organizational body, an individual, or a group of people who depend on the system, can be affected by the system, or who influence system requirements” [8].

Stakeholders are categorized by their concerns into three types: domain stakeholders, project/process stakeholders, and product stakeholders. Their concerns are in the subject domain, the development project/process, and the use of the system, respectively.

Stakeholders are modeled with Stakeholder Modeling Language. The model elements in the language are entities, relationships between entities, and attributes. Entities (or concepts) are grouped into core and supporting entities. The core entities are Stakeholder, Organization, FocalOrganization, Individual, Role, Task, and Resource. The two supporting entities added to the language to help in the modeling process are CollaborationSetting and Layer. Relationships link entities. Attributes capture the relevant details of entities and relationships. Most, if not all, of the language elements are included in the adapted scheme and their semantics are described in the next section.

The modeling elements can be represented in textual or graphical form. The textual model can present more information while the graphical model is more compact. As trade off for simplicity, the graphical form contains a subset of the language elements. In most cases, the information captured graphically is enough to help the engineers to understand the problem domain. Particularly for the early stage of the domain study, the graphical form is suitable. We adapted it in this work.

2.2 Stakeholder Diagrams

The graphical stakeholder model is presented as Stakeholder Diagrams. There are three different kinds, namely, Collaboration Structure Diagram (CSD), Task Exploration Diagram (TED), and Stakeholders Mapping Diagram (SMD).

CSD presents the structure of the collabora-
tions in tasks of all stakeholders, called the “collaboration structure”. It shows all collaboration settings and tasks included in each one, all tasks only at the highest level of detail, relevant stakeholders, and resources owned by each stakeholder.

TED presents the concepts related to the tasks displayed in CSD. It is the exploded version of CSD in a way that it presents tasks at all levels of detail including resources that each task uses and produces. However, as resources owned by each stakeholder are shown in CSD, they are not duplicated in TED.

SMD captures all revealed stakeholders and relationships between them. The number of stakeholders modeled in SMD may exceed that modeled in CSD because SMD contains all disclosed stakeholders, although they are not related to the collaboration structure.

For separation of concerns, each kind of diagrams can be modeled in three different worlds: subject, development, and usage worlds, focusing on elements in the problem domain, the development project, and the system in operation, respectively. Each type of stakeholders, namely, domain, project/process, and product stakeholder type is the subject to be modeled in each of the three worlds respectively. Moreover, the subject world is separated into two different sub-views, namely, view of the “system as is” and the view of “system to be”. So CSD can be drawn in 4 diagrams, i.e., CSD-S-AsIs, CSD-S-ToBe, CSD-D, and CSD-U to capture the collaboration structures in the subject world as it is, the subject world as it will become when the new system is in place, the development world, and the usage world, respectively. As well as CSD, TED and SMD each can be drawn in 4 different views.

3 The Adapted Stakeholder Modeling Scheme

This section describes the contribution of this work to make Stakeholder Modeling Scheme applicable for requirements engineering of software systems for public.

3.1 Modeling Elements in Stakeholder Diagrams

The entities, relationships, and attributes elements can be modeled in the stakeholders diagrams are described as followed. (For understandability, presenting each of them for the first time, we boldface the Entity names (and as Entity the following times), underlined the relationship names, and italic the attribute names. In addition, for clarity, the value of attribute in the language explanation is presented in single quotation marks whereas the value of some entities and attributes in examples are presented in double quotation marks.)

The entity at the center of the language is Stakeholder. It is classified into 3 subclasses, namely Role, Individual, and Organization. Stakeholder is an abstract type entity, meaning the type of each Stakeholder instance must be defined when modeled. The Role type Stakeholder refers to a set of stakeholders who have the same responsibility for behavioral characteristics, for example, “Student”. Individual is a type of Stakeholder referring to individual persons, “Daniel”, for example. The Organization type stakeholder can be any organization related to the system under study.

A Stakeholder of any type may play a Role type stakeholder. The stakeholder playing the role is called rolePlayer. For example, “Daniel”, an Individual type stakeholder, is a rolePlayer of “Student”, a Role type stakeholder. Each Role has the playerType indicating the number of the players of the role. Depending on the number of the players of the Role, the value of playerType can be ‘only one’, ‘one’, ‘many’, or ‘group’. They mean the role is played by one and only one, one, many, or a group of players, respectively. For the only one and one player Type, the name of the stakeholder who plays the Role must be specified. For the many and group playerType Roles, some, if not all, players should be specified unless the engineer knows that it is not useful to model them.

An Organization type stakeholder has some Stakeholder(s) as its representative.

Stakeholders may relate between themselves in 4 kinds of relationships, namely reports, consults, and liaises giving 3 attributes with the same names, i.e., reports, consults, and liaises. The reports relationship implies direct influencing power of the superior Stakeholder over the subordinate one. The consults relationship covers some degree of influencing power between them although less formal or informal. The engineers model these relationships when they see some influencing power between them from any spe-
cific circumstance. Liaises is the relationship which shows an informal communication link between Stakeholders where there is no influencing power found between them.

A Task represents an association between two groups of Stakeholders, namely the task owner and the task collaborator. The owner has the ultimate responsibility for completing the task whereas the collaborator participates and has responsibility for their participation. Many stakeholders may share responsibility in a Task. Some Task may not have a collaborator if the owner can complete it by oneself, a so-called ‘self-completing task’. If the owner and the collaborator of a task are specified with the same role, it means that different role players of the role collaborate in the task. A Task may have a regulator who defines a particular way to do the task.

A Task uses or produces some resources. A Task may be decomposed into smaller Tasks. In other word, a Task includes other Tasks.

Resources can be a system or some kind of information in any format. A Resource may include other Resources.

The owner of a Resource (called rOwner) is the Stakeholder who owns it. The privacy owner (called pOwner) is the Stakeholder whose privacy data are captured in the Resource. Some Resources do not contain privacy data, in which case, it has no privacy owner. For the system type Resource, its maintainer is the Stakeholder who maintains it.

The graphical notations of the language elements are presented in Figure 1.

### 3.2 The Adapted Modeling Process

As the intention of the initiator of a system for public is to solve some problems for or give some opportunities to the public, their main focus of study is at the situation in the subject domain. The engineer should identify domain stakeholders, with no concern at project/process/product stakeholders, to capture as much application domain knowledge as possible. Moreover, to reduce time and complexity, the as-is state and the to-be state of the subject world are not separated into different diagrams. Therefore, there are three diagrams to be modeled: CSD-S, TED-S, and SMD-S.

The modeling process is iterative so that more stakeholders are uncovered in each revision. Engineers model stakeholders concurrently with eliciting their requirements. The modeling process is as followed:

**Step 1:** From the problem statement, model outstanding domain stakeholders in SMD-S. In any cases, it should not be difficult to identify the very first stakeholders. They normally are obvious individuals or roles exist in the subject domain.

- For each individual, identify the roles they play, which are relevant to the problem.
- For each identified stakeholders, identify the tasks they relate to, i.e., the tasks each one owns, collaborates, and/or regulates. Model the tasks in TED-S and CSD-S. The tasks to identify are those relevant to the problem at hand. The statement of the problem is helpful for revealing them.

**Step 2:** Identify the collaboration setting(s) relevant to the problem domain. Model them in CSD-S.

- For each collaboration setting, reveal the tasks of which it is composed. Modify CSD-S accordingly. Moreover, all the revealed tasks must be modeled in TED-S.
- For each identified task, identify the task owner, collaborator, and regulator, (if known and/or exist). In addition, identify the resources used and/or produced by each task. Modify TED-S accordingly.
- For each revealed resource, identify stakeholders who are the owner, privacy owner, and maintainer (if exist) of the
resource in CSD-S. Model the resource used and produced by the respective tasks in TED-S.

- For each revealed stakeholder, add it in SMD-S. Depending on the type of the stakeholder, model the roles the individual type stakeholder plays or the player(s) of the role type stakeholder or the representative of the organizational type stakeholder. Identify and model relevant relationships between stakeholders in SMD-S.

**Step 3**: Identify the collaboration setting(s) relevant to the goal of the proposed system. Repeat Step 2 until all the elements in the collaboration setting are uncovered.

**Step 4**: Gathering domain knowledge and stakeholders’ requirements. Revise the diagrams by repeating Step 2 until enough domain knowledge is gained and all important requirements for the system are clear.

4 Results: An Application to a Case Study

4.1 The Case Study

The adapted modeling scheme was applied to a software system for public, Krung-Shing Climate Monitoring system. The problem statement was that Krung-Shing, a sub-district in Nakhon Si Thammarat, had not enough information to enhance the inhabitants to deal with a natural disaster, particularly heavy rainfall causing severe flood and landslides, in time for them to safely evacuate from the villages. From the last incident in 2009 when many areas were badly destroyed by flood and landslides, the inhabitants could not rely only on the authorities. During the disaster most, if not all, communication channels were disconnected. To better prepare for such incidents in the future, the system was initiated by a person living in the area.

The goal of the system was to provide basic and geographical information of the villages in Krung-Shing and information about the climate that helps the inhabitants and also the authorities to predict the likelihood of severe flood and landslides during a period of heavy rainfall. The system would be used mainly for monitoring rainfalls and supporting the evacuations if needed. The ultimate goal was to decrease loss of lives and properties from possible future incidents.

4.2 Data Collection and Modeling Process

The data were collected by interviews. The first stakeholder identified was the initiator of the system, Mr. Rangsan. From the first few pieces of data collected, the first version of SMD-S was drawn as illustrated in Figure 2.

![Figure 2. The first version of SMD-S](image)

The diagram presents that Mr. Rangsan was one of the “Inhabitants” and also “Unofficial Leaders”. During the disaster, many unofficial leaders played the “Volunteers” role. The keyword “MANY” means the role can be played by many players; whereas “SOME” means some of the players are presented in the model. One of them was Mr. Rangsan.

The first collaboration setting, “Evacuate”, was modeled in CSD-S. The stakeholder model led us asked the questions regarding the tasks, resources, and stakeholders existed in the collaboration setting. We found that it contained one task, i.e., to evacuate the inhabitants. Thus, the TED-S was appeared to be the same as CSD-S, as presented in Figure 3.

![Figure 3. The first version of CSD-S and TED-S](image)

In the figure, Evacuate is a task responsible by some government agencies, in collaboration with inhabitants and volunteers. At the same time, new pieces of data were added to the SMD-S, as shown in Figure 4.

![Figure 4. The second version of SMD-S](image)
The diagrams modeled so far partially described the subject domain as it was. Then the “to-be” state of the subject domain was analyzed. Together with the goal of the system, a collaboration setting “Monitoring and Evacuation”, a collaboration structure for the preparation of evacuation, alert and evacuation was identified. We then added the “Monitor” task into CSD-S. Mr. Rangsan added that, in doing so it was necessary to have information about rainfall and other basic data about the villages such as geographical aspects of the villages, data of inhabitants and households, pre-defined evacuation points. So we continued by asking about the sources of the data he mentioned. The information collected were used for revising CSC-S, as shown in Figure 5.

![Figure 5. The second version of CSD-S](image)

The data required for carrying out each task were identified and TED-S was updated, as depicted in Figure 6.

![Figure 6. The second version of TED-S](image)

More stakeholders identified during the modeling process were interviewed. Data were collected from Krung-Shing Community Head, a Community Development Officer, a Disaster Prevention and Mitigation Officer, and a Police Assistant Volunteer. The model was used as the guideline of the interview questions. The stakeholder diagrams of the subject world, namely, CSD-S, TED-S, and SMD-S were updated after each interview to capture new domain knowledge. The results after some more rounds are presented in Figure 7, Figure 8, and Figure 9, respectively.

![Figure 7. A refined version of CSD-S](image)

![Figure 8. The refined version of TED-S](image)

![Figure 9. A refined version of SMD-S](image)

While collecting the domain knowledge, the requirements of those stakeholders were elicited. Some examples are presented in table 1.

**5 Discussion**

During the course of study, many potential stakeholders were revealed and some of them were met for interviews. The stakeholders and their relationships are shown in SMD-S. The relationships between stakeholders and tasks including resources relevant to the problem are captured in CSD-S and TED-S. Applying the adapted modeling scheme to the case study, we found that the modified scheme had these advantages:
• The adapted model helped the engineer to focus the stakeholders in the problem domain, hence, it is suitable for studying a system at the very early stage, as this one for public. The separation of stakeholders into three different views in [8] helped in identifying domain stakeholders with no concern or question about who would be the project/process/product stakeholders.

• Developing the model of the subject world “to-be” from the subject world “as-is”, i.e., drawing different versions of one diagram, decreased errors, time, and complexity should have incurred from drawing many versions of two different diagrams. At the end, the view of the world when the system is in place is what the engineer must have. The engineer simply developed the changing view of the world from the current state through the process. For example, from Figure. 3 to Figure. 5 and to Figure. 7.

• The modified scheme was useful for studying the application domain of the case study because it could be applied without identifying the host organization or knowing the scope of the system. This means that engineers can get benefit from modeling the application domain of software systems for public even before the start of a development project.

• The adapted modeling process was done manually. This means that it was systematic and simple enough to use by a skilled modeler. In addition, the resulting diagrams are meaningful. However, a new adopter has to learn to be familiar with the modeling language. In any case, for the method to be adoptable in a more complex subject domain, a modeling tool is needed to help the process practical, less time consuming and less error prone.

The adapted scheme still offers its benefits to engineers, i.e., to systematically collect domain knowledge, no lesser than the full scheme does. At the same time, with the information suggested in the model, requirements can be elicited from identified stakeholders effectively. For example, with the essential information about their relationships to other stakeholders and other entities, such as Tasks and Resources from the model, we could ask more relevant questions, expect the scope of the answers, and so interpret requirements stated by the stakeholders more meaningfully. Furthermore, the diagrams gave the big picture of the application domain, which was helpful for making decision about the scope of the system. For example, in Evacuate task, if the geographical map of the sub-district was really needed, Geo-Informatics and Space Technology Development Agency, the owner of the map data, must have been consulted as a key stakeholder. Otherwise, the scope of the system must have been decreased or other options had to be found.

Table 1. Examples of stakeholders identified during the modeling process with some of their requirements

<table>
<thead>
<tr>
<th>Stakeholders</th>
<th>Domain knowledge gained from the model</th>
<th>Stakeholder’s requirements</th>
</tr>
</thead>
</table>
| Inhabitants                         | • They collaborate in the evacuation and monitoring tasks.  
• Their privacy data are kept in the populationDetials.  
• They usually respect the official and unofficial leaders.  
• They are closed to the health promotion volunteers. | The system shall..  
• alert the inhabitants in advance when the situation is becoming serious.  
• contain knowledge about geography, demography, evacuation process, including knowledge helpful for them to predict severe flood or landslide, e.g., quantity of rainfall, cloud observation, etc.  
• provide information about the persons who need special care during the incident such as elderly. |
| Unofficial Leaders                  | • They are a volunteer group.  
• They gather to help in evacuation during the incident. | The system shall..  
• provide geographical map with details enough for planning the evacuations when needed.  
• give precise details of the inhabitants in needed of special help. |
| Disaster Prevention and Mitigation   | • They are responsible for the monitoring | The system shall..  
• automatically warn the inhabitants when |
<table>
<thead>
<tr>
<th>Stakeholders</th>
<th>Domain knowledge gained from the model</th>
<th>Stakeholder’s requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Officers</td>
<td>and evacuation tasks.</td>
<td>the situation is risky.</td>
</tr>
<tr>
<td></td>
<td>• They report to</td>
<td>• record loss</td>
</tr>
<tr>
<td></td>
<td>Krung-Shing</td>
<td>information and</td>
</tr>
<tr>
<td></td>
<td>Community Head.</td>
<td>send it directly to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the office of the district.</td>
</tr>
</tbody>
</table>

### 6 Conclusion

Being the main source of domain knowledge and requirements, stakeholders must be correctly identified and properly involved. Stakeholder Modeling Scheme offers stakeholder identification alongside insightful information regarding each stakeholder which is very beneficial to the engineers because it shows the degree of relevance that each stakeholder has to the system under consideration. The model is both descriptive and prescriptive (i.e., it captured relevant details of stakeholders explored during the process and led to identifying them). These benefits are not offered in other approaches to stakeholder identification. However, the scheme is not applicable for requirements engineering of systems for public. Such a system has some special characteristics. It is initiated because of its potential to be useful to a wide range of stakeholders. The scope of such a system has not been decided and it tends to cross many organizational boundaries including people in public. Its application domain should be understood even before proposing to a development project sponsor, namely, before knowing the host organization.

We have presented the adaptation of Stakeholder Modeling Scheme to make it applicable for requirements engineering of systems for public. The adapted scheme applied to an initiative, Krung-Shing Climate Monitoring system, showed that it offered a systematic way to identify stakeholders alongside their relationships to other stakeholders and to other elements in the system and the model suggested more questions and requirements to ask each of them. In addition, the adapted version is more compact and can be applied before identifying the host organization. Hence, the adapted scheme is applicable for requirements engineering of systems for public.

### Acknowledgment

We would like to thank Walailak University for the research grant and the participants (who do not want to have their names presented) for their contribution as the sources of data for this study.

### References


