Goal Oriented Requirements Engineering for Web Applications: A Comparative Study

Sangeeta Srivastava¹, Shailey Chawla²

¹Bhaskaracharya College of Applied Sciences, DU, Delhi
²Department of Computer Science, University of Delhi

Email: sangeeta.srivastava@gmail.com
Email: shaileychawla@gmail.com

Abstract— The web applications need to be engineered methodologically as other information systems because now they have become pervasive and mushroomed very fast. The Goal oriented Requirement engineering is a very popular approach because it creates a product closer to the stakeholders demands. Incorporation of this approach in web applications will reap numerous benefits. The study here compares some approaches, mostly built for generic systems that can be applied to Web applications. The Web applications' unique concerns are also analyzed against these applications. The need for Goal oriented Web Requirement Engineering is highlighted.

Index terms- Goal oriented Requirement Engineering, Web applications, Web engineering

I. INTRODUCTION

For achieving a better understanding of web application development, we need to understand the unique characteristics of web systems and how these characteristics impact the development process. The web applications are unique in many aspects mainly the dynamic nature, vast audience, high volatility of client needs, rapidly changing technology. For creating a web system, like any system, it is critical that web application meets the requirements put up by the stakeholders and technically be efficient.

The web applications have been classified in [1] and the requirements have been studied against it[2]. The requirement analysis for web applications is different from generic applications in many aspects. The requirements catered by a web applications are also specific to the area. Combining the generic nature of software applications and special needs of web applications, the requirements have been classified as in [3] as below:

Functional requirements are capabilities that a system must exhibit in order to solve a problem. Other definitions are given by [4][5]. Functional requirements can be sub- classified as:-

- Data requirements
- Interface requirements
- Navigational requirements
- Personalization requirements
- Transactional requirements

Non functional Requirements are the expected qualities a system should possess. The paper [6] studies the impacts of non-functional requirements in web system projects. Examples of NFRs are portability requirements; reuse requirements, usability requirements, availability requirements, performance requirements, etc.

Requirements Engineering (RE) refers to the activities intended at assuring that a software system fulfills the goals, the needs and the expectations of all the relevant stakeholders [7][8]. In particular, RE involves a set of intertwined tasks [9][10]: a) requirements elicitation; b) analysis and modeling; c) negotiation and validation. Poor requirements augment the risk of missing the opportunity of meeting customers’ needs and enhancing the user experience [11]. In fact, in order to be successful on the market, a web application has to be stakeholder-centered.

II. GOAL ORIENTED REQUIREMENTS ENGINEERING

According to [10] Requirements engineering extends the ‘what is done by the system’ approach with the ‘why is the system like this’ view. This why question is answered in terms of organisational objectives and their impact on information systems supporting an organisation. The systems build are to support the functioning of the organization and make things easier and more organized for them. This has two implications (a) elicitation and validation of the requirements of a system are done with respect to their purpose in organisations and (b) only organisationally purposeful systems are conceptualised. RE addresses the problems associated with business goals, plans, processes, etc. and systems to be developed or evolved in order to achieve organisational objectives[12][8]. A goal is an objective the system under consideration should achieve. Goal formulations thus refer to intended properties to be ensured; they are optative statements as opposed to indicative ones, and bounded by the subject matter [13][14][15].

In particular, for business application systems, goals may be used to relate the software-to-be to organizational and business contexts. The benefit of goal modeling is to support heuristic, qualitative or formal reasoning schemes during requirements engineering. Goals are generally modelled by intrinsic features such as their type and attributes, and by their links to other goals and to other elements of a requirements model. Soft goals are goals that do not have a clear-cut criterion for their satisfaction [15]. It allows representation of ill defined, abstract and high level objectives of stakeholders. In paper [15] the meanings are different for satisfying a goal and satisfying a goal. While satisfying a goal implies a mechanic translation of a goal into an operative behaviour, satisfying a goal suggests the indefiniteness of the relationship goal-
requirement, and points out the risk of doing a strategic
decision that might contribute to fulfill an objective. 
Softgoals are satisfied when there is sufficient positive
evidence for their satisfaction; otherwise, they are not satisfied.

There has been a massive amount of work on linking
goals and scenarios together. The obvious reason is that
scenarios and goals have complementary characteristics;
the former are concrete, narrative, procedural, and leave
intended properties implicit; the latter are abstract,
declarative, and make intended properties explicit.
Scenarios and goals thus complement each other nicely for
requirements elicitation and validation.

III. GOAL ORIENTED APPROACHES

There has been a lot of work on Goal oriented
Requirement Engineering in the Information System
Domain. The Goal oriented approach is more successful
because it captures the intentions of the stakeholders
closely and various issues like feature interaction,
conflicting requirements are resolved in the Requirement
phase only. Though the work is mostly for generic systems,
the specific needs of Web applications are not catered
entirely by these approaches. Following are some
approaches that have been developed for Goal oriented
analysis and applied to Web Systems as well.

A. Tropos

Early requirements analysis focuses on the intentions of
stakeholders. These intentions are modeled as goals which,
through some form of a goal-oriented analysis, eventually
lead to the functional and non-functional requirements of
the system-to-be [16]. In i* (which stands for “distributed intentionality”),
stakeholders are represented as (social) actors who depend on each other for goals to be achieved,
tasks to be performed, and resources to be furnished. The i*
framework includes the strategic dependency model for
describing the network of relationships among actors, as
well as the strategic rationale model for describing and
supporting the reasoning that each actor goes through
concerning its relationships with other actors.

B. Scenarios

Scenarios are widely used in visualizing the usability of
the information systems. A scenario is description of the
world, in a context and for a purpose, focusing on task
interaction. It is intended as a means of communication
among stakeholders, and to constrain requirements
engineering from one or more viewpoints (usually not complete, not consistent, and not formal) [17]. Scenarios
are useful for design too, because they may surface
application features to be designed and help making design
decisions. Scenarios should be complemented by other
design techniques, that might reconcile scenarios to obtain
a consistent and complete set of requirements.

C. Aware

It is a goal oriented modeling approach for web
application development[18]. It uses i* to represent
functional goals and softgoals. The model studies
hypermedia intensive sites like cultural heritage websites,
educational websites, promotional and corporate websites.
The work inspires from TROPOS and GBREM. It uses
scenarios to exemplify goals and point out requirements
vividly and are often used to complement goal analysis
approaches. It also focuses on goals in context of
stakeholders that are not the users. I* model is used to
model actors and their dependencies with respect to goals
and tasks. The model integrates the hypermedia
requirements with i* model and then the model is
combined with W2000 web design model to bridge the gap
between specification of goals and design primitives.

D. E3 Value Modeling

According to this model [19] it is important to
understand the business model for creating economically
viable web applications. It involves goal and value
modeling in requirement engineering. First an i* strategic
dependency diagram is created stating the enterprises
participating in this e service and how they depend on each other.
Based on this Strategic dimension model, e3 value is
developed containing only actors and their exchanges,
thereby concentrating on relationship between enterprises.
Then an i* strategic rationale diagram is created focusing
on internal enterprise interests. The i* strategic model can
use the results from profitability analysis because e3 value
quantifies many goals as profitable goals. This model thus
effectively uses i* model for identifying the goals,
establishing relationships between various actors, doing the
profitability analysis and justifies the finalization of
functional and non functional requirements according to
the goal and value analysis. The model specifically studies
the business value of a web application.

E. VIP Business Modeling Framework

This[20] is a value based business model that is
integrated with web design model by providing appropriate
business modeling notations. The Business model here has
three core parts: BVM(Business Value model),
BIM(Business information model), BPM(Business Process
model). VIP model uses UML 2.0 profile for modeling for
ease of tool support and wide usage. After all the three
model the VIP framework follows the WebML web design
method for creating the web application. WebML expresses
the core feature of the application without expressing
system description.

F. Crews

This work done by [21] focuses on enterprise modeling.
It combines goal modeling with scenarios. Goal modeling
helps the domain expert getting over the fuzzy nature of
goals, helps discover goals and aids in goal reduction.
Scenarios are descriptive, explanatory and exploratory.
Coupling of goals and scenarios is used to operationalize
goals. The coupling ensures that system usage fulfills
the expected goals, and new goals can be discovered through
scenario analysis.

G. URN

URN refers to User Requirements Notation[22], it has been
standardized in ITU standards Z-series in 2008. It is a

© 2010 ACEEE
DOI: 01.IJRTET 04.02.135
97
combination of two notations GRL and UCM. User Requirements notation aims to capture goals and decision rationale that finally shape a system and model dynamic systems where behavior may change at run time. It describes it can capture the user requirements even when very little design detail is available. It also facilitates seamless transition from requirement specification to a high level design. The two components of URN- GRL and UCM focus on goals and scenarios both.

TABLE I:

VARIous APPROACHES AGAINST PARAMETERS

<table>
<thead>
<tr>
<th>Model</th>
<th>TROPoS</th>
<th>Aware</th>
<th>SCEN</th>
<th>E3</th>
<th>VIP CREWS</th>
<th>URN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functional Goals</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*Soft Goals</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Users Goals</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*Web specific Goals</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*Choice of alternatives</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Goal scenario coupling</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Goal refinement</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*Goal Prioritization</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*Transition to Web Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*Applicability to Web</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*Model Reusability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

IV. CONCLUSION

All the approaches listed above are Goal oriented approaches. The VIP framework, AWARE and E3ValueModel have been built specially for Web applications and others are generic approaches. The Web specific approaches do not cover many Goal oriented aspects. The generic approaches can be applied to web applications but the areas in Web Development that need special attention like navigation are surpassed. As depicted in Table I there are many areas marked with (*) specially concerned with web applications that need attention like models applicability to web, analysis of web requirements, transition to web design. Also, goal prioritization is another issue that can be explored for Web requirement engineering. The comparative study gives an insight to the GORE approaches and gives a fair idea of future research scope. Our future work will include development of a Requirement model that suffices the Web concerns and covers both functional and non-functional goals in the analysis phase.

REFERENCES


[10] Collette Rolland, Naveen prakash, From conceptual modeling to requirement engineering.


