

GOREWEB Framework for Goal Oriented Requirements Engineering of Web Applications

Shailey Chawla¹, Sangeeta Srivastava², and Punam Bedi¹

¹ Department of Computer Science, University of Delhi, North Campus, Delhi - 110007, India

² Department of Computer Science, Bhaskaracharya College, Sec-2, Dwarka, Delhi, India
shaileychawla@gmail.com, sangeeta.srivastava@gmail.com,
pbedi@cs.du.ac.in

Abstract. In this paper, we propose a framework for modeling goal driven requirements of web applications. Web engineers mostly focus on design aspects only overlooking the real goals and expectations of the user. Goal oriented Requirement Engineering is a popular approach for Information system development but has not been explored much for Web applications. However, in today's times Web is dominating in every business making it imperative that its requirements are analyzed carefully and in profundity. Goal driven requirements analysis helps in capturing stakeholders' goals very finely, by choosing between alternatives and resolving conflicts. Detailed classification of both functional and non-functional requirements specific to web applications is discussed in the presented work. A framework, GOREWEB (Goal oriented Requirements Engineering for Web Applications) is proposed for analyzing goals and translating them into functional and non-functional web requirements.

Keywords: Goal oriented Requirements Engineering, Web engineering, Goals, Requirements, URN.

1 Introduction

Although web applications have mushroomed a great deal but they have still not received much attention from the requirements engineering community. Like the traditional information systems, where Requirements analysis is given utmost importance amongst all the phases, with web applications the focus is usually more on the presentation. Web applications involve multiple stakeholders, and the size and purpose of the applications are also varied [1]. Gaus et al in [2] defined Requirements Engineering (RE) as the set of activities intended at assuring that a software system fulfills the goals, the needs and the expectations of all the relevant stakeholders. In the requirements engineering community, the requirements have been divided into functional [3] and non-functional requirements[4]. There has been a lot of emphasis on the functionality; however the functionality is not useful or usable without the necessary non-functional characteristics [5]. According to Rolland et al [6] Requirements engineering extends the 'what is done by the system' approach with the 'why is the system like this' view. This 'why' is answered in terms of organizational

objectives and their impact on information systems supporting an organization. Poor requirements augment the risk of missing the opportunity of meeting customers' needs and enhancing the user experience [7].

The rest of the paper is organized as follows. In section 2, we describe the related work in the area of Goal oriented Requirements Engineering. Also, we give a brief overview of User Requirements Notation (URN) that we would be using for Goal oriented Requirements Analysis. In section 3, we propose a framework for incorporating goal oriented requirements analysis for engineering web application. For integrating goals with web specific requirements in the framework, we feel that a proper categorization of web specific functional and non-functional requirements needs to be done. We have provided a web specific functional and non-functional requirement categorization in section 4. Further, the framework has been explained using a case study on web based education in section 5. Finally, section 6 summarizes our work and concludes the paper.

2 Goal Oriented Requirements Engineering

In recent times, Goal oriented Requirements Engineering proposed by Mylopoulos [4] has become very popular for analyzing the requirements. A goal describes the objectives that the system should achieve through the cooperation of agents in the software-to-be in a given environment as defined by Liu et al in paper [8]. According to Lamsweerde, Goal-oriented requirements engineering (GORE) is concerned with the use of goals for eliciting, elaborating, structuring, specifying, analyzing, negotiating, documenting, and modifying requirements[9]. The goal based analysis helps to explore the alternatives, resolving conflicts, and relate them to the organizational objectives [4]. It has been also established in [10] that stakeholders pay more attention to goal models compared to the UML models because they can relate to the concepts more closely. There has been a massive amount of work on linking goals and scenarios together [11],[12], [13], [14]. The obvious reason for this linking 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. Based on a bidirectional coupling between scenarios and goal, Rolland et al [16] propose heuristic rules for finding out alternative goals covering a scenario, missing companion goals, or sub goals of the goal under consideration.

Many approaches have been developed for Goal oriented Requirements Engineering for generic systems [16],[17], [18]. However, the notations and models developed for generic applications do not address very important issues of web applications like navigation, adaptation etc. Some work has been done by researchers [19], [20], [21],[22] on web engineering approaches taking into account the Goal driven analysis, but many concepts of goal driven analysis like design rationale, conflict resolution, goal prioritization have been surpassed and not taken in totality.

URN [23],[24] refers to User Requirements Notation. It is currently the only standard that combines goals and scenarios in one notation. It is a combination of two notations GRL (Goal Requirements Language) and UCM (Use Case Maps). 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. GRL is *Goal Requirements Language* that focuses on Goal analysis. It help in defining the goals including the non-functional requirements, evaluating them, resolving conflicts etc. UCM stands for *Use Case Maps* that are the visual notation for scenarios. UCM notation employs scenario paths to illustrate causal relationships among responsibilities. The combination of GRL and UCM as depicted in Fig. 1 helps to improve the definition of new goals and satisfy them. GRL as described by Amyot in [23] supports five kinds of intentional elements explained below:

- **Goal:** Quantifiable high-level (functional) requirement (rounded cornered rectangle).
- **Soft goal:** Qualifiable but unquantifiable requirement, essentially non-functional (irregular curvilinear shape).
- **Task:** Operationalized solution that achieves a goal or that *satisfices* a soft goal which can never be fully achieved due to its fuzzy nature hexagon).
- **Resource:** Entity whose importance is described in terms of its availability (rectangle).
- **Belief:** Rationale or argumentation associated to a contribution or a relation (ellipse).

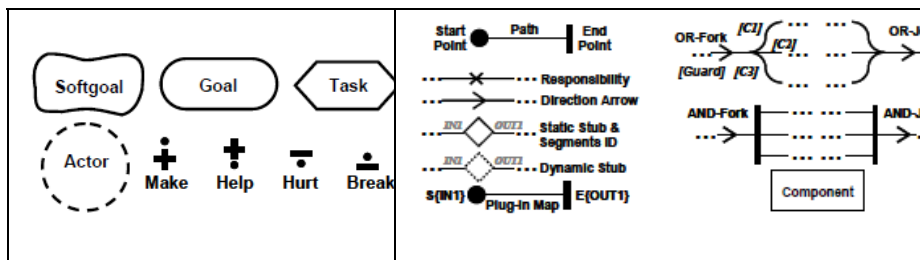


Fig. 1. Subset of GRL &UCM notation

There are also five categories of intentional relations, which connect elements:

- **Contribution:** Describes how soft goals, tasks, beliefs, and relations contribute to each other. Each contribution can be qualified by a degree: equal, break, hurt, some-, undetermined, some+, help, or make.
- **Correlation:** Contribution that indicates side-effects on other intentional elements (dashed line).
- **Means-end:** Link for tasks achieving goals. Different alternatives are allowed.
- **Decomposition:** Defines what is needed for a task to be performed (refinement), always AND.
- **Dependency:** Link between two actors depending on each other (half-circle).

UCMs have following basic concepts according to Amyot [23].

- **Start point:** Captures preconditions and triggering events (filled circle).
- **Responsibilities:** locations where computation (procedure, activity, function, etc.) is necessary (cross).
- **End point:** Represents resulting events and post-conditions (bar).

- **Paths:** Connects start points to end points and can link responsibilities in a causal way.
- **Component** represents an abstract entity (object, server, database etc.)(rectangle).

There has been work on Goal oriented Requirement analysis like i*, NFR framework, URN [17], [5], and [24] but these are for generic systems. The specific needs of web applications like heterogenous user group, specific emphasis on navigation and presentation need a special focus. The framework described in the next section overcomes the gaps in web engineering approaches.

3 GOREWEB Framework

For enhancing the requirements engineering activities involved in web application development, GOREWEB: **Goal Oriented Requirements Engineering for Web** applications framework offers goal oriented requirement analysis of web applications. GOREWEB model extends the concepts of User Requirements Notation (URN) for comprehensive study of web application requirements. Amongst numerous differences, the concepts of navigation, adaptation, presentation are distinctive to web applications. Bolchini has provided a metamodel for integrating goals with web engineering approach in [19] where he uses i* for goals modelling and maps the goals to web requirements and later to a web design approach WebML. However, we have enhanced the metamodel in our framework by taking URN as the backbone of Goal analysis that couples functional and soft goals with the scenario modelling. Also, as described in the next section web application requirements have been enhanced and redefined. The framework is shown in Fig. 2. The framework created using the standard UML class diagram depicts the relation between the raw goals captured from the stakeholders and the requirements to be used by the web designers. For realization of web specific requirements, we first need to categorize web application requirements, so that we know how goals are mapped to specific class of requirements. In the next section we categorize web application requirements so that they can be appropriately mapped with the sub goals and tasks in the GOREWEB framework.

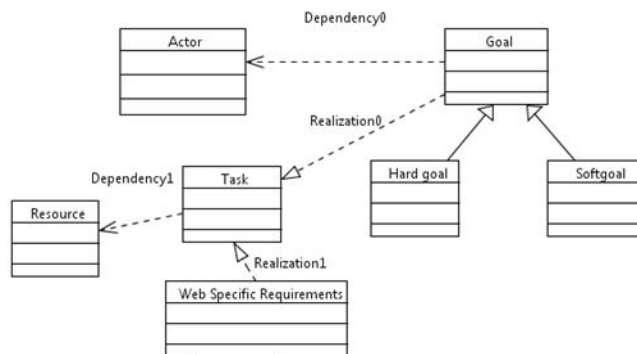


Fig. 2. GOREWEB Framework

4 Web Application Requirements

As with generic systems, web applications requirements are classified as functional and non-functional requirements. Although many other researchers classified the web requirements (mostly functional), a summarized unanimity is presented in [25]. However, we found that the focus is still more on functional requirements. We hereby extend and redefine web application requirements. The categorization is depicted in Fig. 4 below.

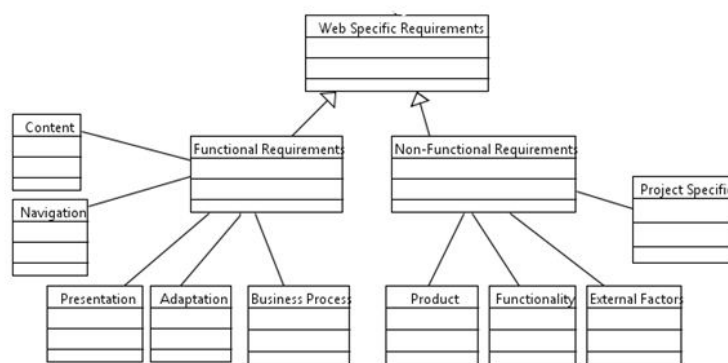


Fig. 3. Web Application Requirements Classification

4.1 Functional Web Requirements

The functional requirements are defined as requirements referring to the functionalities and behaviour of the system. They are classified in context of web as:

- a) *Content requirements*: These requirements specify what information or ideas are to be communicated to the user through web application. The content includes text, graphics, images, audio and video data that has to be provided by web application. The content requirements for a e-bookshop can be 'provide image of the book cover and information about the author', 'provide information about the organization's history on a web page'
- b) *Navigation Requirements* Navigation is defined as finding your way around a website. The requirements state the navigation structure and navigable elements. It will define what all web pages are linked with each other and how are the links provided. For example, 'connect a book with other books by the same author, and books on the same subject'.
- c) *Presentation Requirements*: These cover the visual elements and interface layouts. The stakeholders can give some insight on the aesthetics of web applications, the physical positioning of the graphics, the colour scheme or style.
- d) *Adaptation Requirements*: The website has to adapt itself depending upon user or environmental profile. The personalization can be done by profiling of users, regions etc. and suitably changing the content/ presentation for it.

e) *Business Process Requirements*: These are the requirements required to accomplish the structured activities or tasks required to serve a particular goal, and focuses on the operational purpose for which web application is being created. It can be refined into User or system operation. *User operation requirements* are the tasks that user will be doing like pressing buttons, browsing, searching etc. These operations are observable by the end-user. These operations are directly initiated by the user like ‘posting comments on a blog’, ‘add items to shopping cart’. *System operations* are the tasks done by the system at the back end like database operations so that user operations can be completed. The system operations are like ‘authenticating the user’, ‘validating a financial transaction’, ‘track web usage of the user for personalization’.

4.2 Non-functional Web Requirements

The term “non-functional requirements” is used to delineate requirements focusing on “how good” software does something as opposed to the functional requirements, which focus on “what” the software does.” [26]. However, we simply state that Non-functional requirements are the requirements that specifies the criteria used to adjudge the system. This should be contrasted with functional requirements that focus only on the operational aspects of the system, i.e. which are needed for a function to operate. It is imperative that unless the non-functional requirements are satisfied, the product is of no use for example, if the information presented in the web application can’t be comprehended by the users, it doesn’t serve its purpose or the web application doesn’t fulfil the soft goals of the organization like increase the profitability; the entire design, presentation of the Web application goes waste. Much work has been done for classification of quality and non-functional requirements[27],[28],[29],[30],[31]. However, in view of web applications no concise categorization of important non-functional requirements exists that would help the engineers create an eminent product. In this paper, we classify and explain the non-functional requirements based on the concerns- product, functionality, external factors and project specific concerns. The categorization is summarized in Table 1.

Table 1. Non-functional Requirements for Web applications

Concern	NFRs	
Product	Usability, Conformance, Security, Efficiency	
Functionality	Content	Credibility, Readability, completeness, Communicativeness, up-to-date
	Navigation	Accessibility, Consistency, Predictability, Relevance, Convenience
	Presentation	Attractiveness, Relevance, Clarity, Consistency
	Adaptation	Customizability, Suitability
	Business Process	Responsiveness, Simplicity, Unambiguity
External Factors	Organizational	Objectives
	Actor	User friendliness, Empathetic, Understandability
	Legal	Conformance to standards, Legal issues
	Environmental	Compatibility, Sustainability
Project Specific	Resource Constraints, Cost, Human Proficiency	

1. **Product:** The non-functional requirements enumerated in this category describe the quality expectations from the final product i.e. web application. We propose that a web application in totality is judged for its usability, conformance and performance. *Usability* The International Standards Organization's (ISO) defines usability as: "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" [32]. The usability for web application has similar meaning and it is used to measure the satisfaction of the user.

Conformance is whether the product delivered suffices all the initially set targets entirely. For example, if it was decided in initial meetings with participants/contributors/ that web application will provide songs playback that can only be availed online. In the final product, song can be played online but with a right click it can be saved in the user's disk, it means the requirement is not met properly.

Security is protection of the product's sensitive content and provide secure mode of data transmission and guard it against external threats. This is a very important requirement for web applications, because the data has to be transmitted through the network. The transactions should take place on secure mode, and encryption techniques need to be applied for transmitting and receiving data.

Efficiency of web application is how fast it loads the pages, graphics and responsiveness.

2. **Functionality** In this category, we explain the qualitative characteristics expected out of the operations in web applications that captures "how good" a system should function. We describe below how for each functional requirement, its qualitative expectations can be captured.

a) Content

Credibility according to the dictionary means whether the information provided is from a trusted source and correct and whether it can be relied upon. Similarly, we state that credibility of the information presented in the website is its reliability, correctness and a surety that it is from a trusted source.

Readability is defined as a measure of the ease of reading and understanding the information from web page, comprehensibility or understandability of written text.

Communicativeness is defined as the ability to communicate the intended information or idea effectively.

Up-to-date means the content should be updated as per the nature of information it is exhibiting. For example, for a news website the latest news should be reported in a matter of hours. Likewise a website for selling mobile phones should have the images, price and reviews for all the latest mobile phone models.

Completeness refers to the totality of the information posted on the web page.

b) Navigation

Accessibility is defined as approachability of hyperlinks when user want to navigate away from a page to a desired or unexplored information.

Consistency is defined as uniformity in positioning of the link on every web page, like we can say that the list of links always appear on left side of the web pages and in top row expandable list of links can be seen.

Predictability is defined as after accessing one page of web application user's can easily guess the placement and content of the hyperlinks of other web pages.

Relevance means the navigation links appropriate to the messages in the web page are displayed. The links have to be semantically related to complete a cognitive or operational task. For example, in an educational website the hyperlink to previous year question papers should be placed in the Web page related to examinations.

Convenience: We define the convenience of navigation links as the ease of reach and the prominence of links on a web page.

c) Presentation

Attractiveness is defined as web application's power of pleasing or appealing the intended users with the look and feel of the web pages.

Relevance is defined as pertinence or suitability of the visual interface of web application with its purpose. For example, application for kids can have bold and cheerful appearance but a professional application should have an corporate appeal.

Clarity is stated as clearness and comprehensibility of appearance of web application.

Consistency is defined as the logical coherence of appearance of various parts of a web application. It means that different web pages of the same website should have similar look and feel like company's logo placement at a uniform place, the color schemes used for menu's, the font etc.

d) Adaptation

Customizability according to dictionary means the ability to be modified dynamically to meet the individual requirements. In context of web we define it as the ability of web application to modify its contents/presentation/process dynamically according to the user profile and other factors. For example, in an online book store application depending upon the navigational usage of the user, he can be given suggestions of similar books using web mining techniques.

Suitability of a web application is defined as the quality of having the properties that are right for a specific purpose. i.e. After the application has personalized to individual needs, the outcome's suitability to the purpose or the user's profile is also significant. The customization or personalization should be such that it appropriate for the needs of the user's to maximum possible extent.

e) Business process

Responsiveness is defined as the quality of readily reacting to any stimulus like pressing buttons, playing videos, performing tasks etc. in web application by the user/ system.

Simplicity means that the business processes that involve both system and user operations must not be complex and be uncompounded to make interaction with the user easy. The complex tasks should be broken into straightforward stepwise parts to ease the procedure.

Unambiguity is defined as complete lack of confusion or uncertainty in the business process. The process should be clear and concise.

3. External Factors In this category, we describe non-functional requirements that are outside the system but greatly affect the ability of web application. The organizational factors, actor's expectations, legal requirements and environmental factors come under this category. They are described in detail below.

a) Organizational

Objectives: Web application is expected to meet organizational long term and short term objectives like profitability, expansion of business, attract new clients etc.

b) Actor:

User friendly: The ease of learning and memorability of web application. This means that the usage of web application should preferably not require special training and its operations should be memorable so that repeated visits are more easy to use.

Empathetic means understanding other's situation in dictionary. In context of Actor's requirement from web application it means that the creation of the web application should be done through identification with and understanding of users' situation and motives. It is a very significant non-functional requirement because in case of web applications, due to vast audience, the web designer should have some understanding of user's needs to make the application acceptable.

Understandability means ability of the users to comprehend the functioning of web application.

c) Legal

Conformance to standards: Besides completing the operational requirements, web application's abide by the relevant legal standards.

Legal issues: Various other legal issues like patenting, copyright etc. have to be looked into while making web applications.

d) Environmental:

Compatibility: Web applications interface and interaction with existing software / hardware shouldn't change the intended behaviour of the application or disrupt normal functioning of interacting items.

Sustainability: Web application should be capable of being sustainable or maintainable. This property involves web applications maintainability like change of the content, repair of some business process, change of technology etc. Also, web application should be able to work with the changes in the environment, like a new browser, a different platform.

4. Project Specific: In this category the non-functional requirements related to the project are listed. The state in which the project has to operate, with kind of resources, budget and human expert skills. They have been categorized as Resource Constraints, cost and human proficiency.

Resource Constraints:- The engineering of web application is dependent on many resources like hardware, software, time limit etc.

Cost- The budget is also a limitation and can affect the creation of web application like in choice of alternatives.

Human Proficiency- The quality of web application is principally dependent on the knowledge and experience of the engineers and designers creating it.

5 Case Study: Web Based Education System

There are many kinds of web applications with different set of requirements and demands [33]. For exhibiting the model and its benefits we take an example of Web application for providing Education. The organization's primary goal is to provide web based distance education. Also, it's the aspiration of the management of the

organization that the web application increases the enrollment of the students and runs in minimal cost. The analysis should also consider specific issues like electricity, connectivity and local language. Web application must provide tutorials, take assignments, conduct on line exams and also have facility for doubt clarification.

After eliciting the goals of the organization, the goals are analyzed and modeled using User Requirements Notation. Fig. 4 shows the GRL diagram. The main goal of the organization is to provide education with the following objectives in mind: Provide subject tutorials, Provide Assignments, Clarify Doubts, Conduct online Examination, Increase enrollment, Increase profit, Minimize cost and expand the reach. Amongst the above-said objectives, the last four objectives- increase enrollment, increase profit, minimize cost, expand the reach are the softgoals. In the GRL model, the softgoal increase usability is explored in detail. The target audience of the website will be students that may have diverse backgrounds, so it would be beneficial to give an online demo on usage of the website Also, the application can also overcome the language barrier and can give options of translating the contents to local language. The cost can be minimized by choosing alternatives that are pocket friendly. The goal of clarifying the doubts can be done in various ways depending upon various factors. The clarify doubts subgoal can be done by using email, discussion forum, chat or video conferencing. The email and discussion forum are easy on cost as they don't require the faculty to be present online and can be done on convenience of time. However, to improve the students' experience, i.e. to increase usability of the application, live chat or video conferencing are better options. Various alternatives can be weighed and chosen in conformance with the stakeholders.

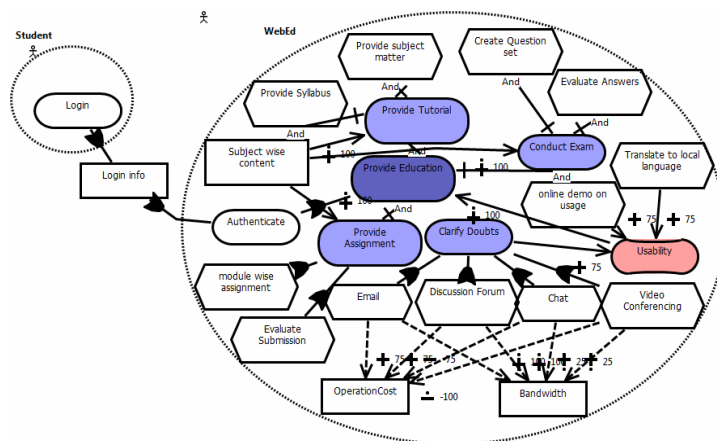


Fig. 4. GRL model for WebEd system

For analyzing this further, the study of this tasks scenario can be done and its operations can be visualized (Fig. 5). A combination of GRL and UCM helps in making decisions, e.g., it can be dynamically decided whether a registered student will be allowed to chat or do video conferencing. The scenarios help in visualizing the

situation and UCMs provide a clear vision of the same. We have modeled the example using jUCMNAV tool [34] that supports both GRL and UCM. It is an eclipse based tool for modeling User Requirements Notation.

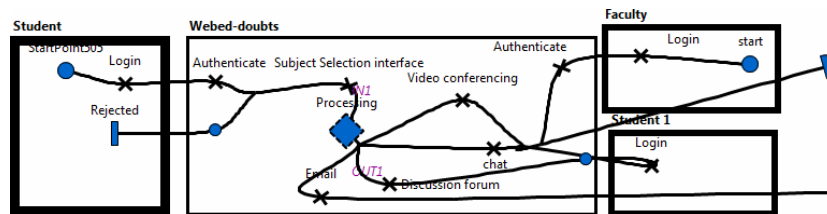


Fig. 5. UCM diagram showing scenarios to Clarify doubts goal

Using Goal analysis, many conflicts amongst goals are resolved and there is reasoning of goals as explored by researchers in papers[35][36]. The non-functional or the quality factors also need reasoning as discussed by Bedi et al in [37] and Jureta et al in paper [38]. We would cover this aspect in our future work to ease the incorporation the Goal oriented Requirements Engineering in the existing Web Engineering methodologies.

6 Conclusion

A framework GOREWEB for analyzing web application requirements from a goal driven perspective is presented in this paper. The GOREWEB framework uses User Requirements Notation (URN) for analysis of goals and scenarios. The stakeholders' expectations and goals are captured and modeled using Goal Requirements Language (GRL). GRL models both functional goals and softgoals. The goals are operationalized by tasks, which are then modeled by creating user scenarios using Use Case Maps (UCM). After careful analysis of both GRL diagrams and UCM, the goals are mapped to requirements of web application. The user goals can be both hard goals and soft goals; hence it is needed to map them to functional and non-functional requirements. A classification of functional and non functional requirements in context of web applications has also been provided in the paper. The framework is described in detail with the help of a case study on Web based education. As Goal driven approaches are closer to the stakeholder's thoughts, the presented framework will help the designers to have clarity of requirements of web applications knowing the goals from the early stages of development.

References

1. Srivastava, S., Chawla, S.: Multifaceted classification of websites for goal oriented requirement engineering. In: Ranka, S., Banerjee, A., Biswas, K.K., Dua, S., Mishra, P., Moona, R., Poon, S.-H., Wang, C.-L. (eds.) IC3 2010. CCIS, vol. 94, pp. 479–485. Springer, Heidelberg (2010)

2. Gause, D.C., Weinberg, G.M.: Exploring requirements: quality before design. Dorset House, New York (1989)
3. Somerville, I.: Software Engineering, 7th edn., ch. 6 (2004)
4. Mylopoulos, J., Chung, L., Yu, E.: 'From Object-Oriented to Goal-Oriented Requirements Analysis'. Communications of the ACM 42(1) (1999)
5. Chung, L., do Prado Leite, J.C.S.: On non-functional requirements in software engineering. In: Borgida, A.T., Chaudhri, V.K., Giorgini, P., Yu, E.S. (eds.) Conceptual Modeling: Foundations and Applications. LNCS, vol. 5600, pp. 363–379. Springer, Heidelberg (2009)
6. Rolland, C., Prakash, N.: From Conceptual Modeling to Requirements Engineering
7. Brinck, T., Gergle, D., Wood, S.D.: Usability for the Web: Designing Web Sites that Work. Morgan-Kaufmann, San Francisco (2002)
8. Liu, L., Yu, E.: From Requirements to Architectural Design—Using Goals and Scenarios
9. van Lamsweerde, A.: Goal-Oriented Requirements Engineering: A Guided Tour. In: 5th Intl. Symp. Req. Eng. (2001)
10. van Lamsweerde, A.: GORE: From Research to practice. In: 12th IEEE International Requirements Engineering Conference, Kyoto (2004)
11. Keller, S.E., Kahn, L.G., Panara, R.B.: Specifying Software Quality Requirements with Metrics. In: Thayer, R.H., Dorfman, M. (eds.) Tutorial: System and Software Requirements Engineering, pp. 145–163. IEEE Computer Society Press, Los Alamitos (1990)
12. Yu, E.S.K.: Modelling Organizations for Information Systems Requirements Engineering. In: 1st Intl Symp. on Requirements Engineering, vol. 0, IEEE, Los Alamitos (1993)
13. Letier, E., van Lamsweerde, A.: Deriving Operational Software Specifications from System Goals. In: 10th ACM Symp. On the Foundations of Software Engineering, Charleston (2002)
14. van Lamsweerde, A., Willemet, L.: Inferring Declarative Requirements Specifications from Operational Scenarios. IEEE Trans. on Software Engineering (1998)
15. Letier, E., van Lamsweerde, A.: Reasoning about Partial Goal Satisfaction for Requirements and Design Engineering. In: 12th ACM Symp. on the Foundations of Software Eng. (2004)
16. Rolland, C., Grosz, G., Kla, R.: Experience With Goal-Scenario Couplin. In: Requirements Engineering. In: IEEE International Symposium on Requirements Engineering, Limerick, Ireland (1998)
17. Castro, J., Kolp, M., Mylopoulos, J.: Towards Requirements-driven Information Systems Engineering: the Tropos Project. Information Systems 27, 365–389 (2002)
18. Antoñ, A.: Goal identification and refinement in the specification of software-based information systems. Dissertation, Georgia Institute of Technology, Atlanta, USA (1997)
19. Bolchini, D., Paolini, P.: Goal-Driven Requirements Analysis for Hypermedia-intensive Web Applications. Requirements Engineering Journal 9, 85–103 (2004); RE 2003 Special Issue
20. Jaap, et al.: e-Service design using i* and e3 value modeling. IEEE software 23(3) (2006)
21. Azam, et al.: Integrating value based requirements engineering models to WebML using VIP business modeling framework (2007)
22. Shailey, C., Sangeeta, S.: Goal driven Requirements engineering: A comparative study. In: CEE 2010 (2010) (accepted)
23. Amyot, D.: Introduction to the User Requirements Notation: Learning by Example. Computer Networks 42(3), 285–301 (2003)

24. ITU-T, Recommendation Z.151 (11/08): User Requirements Notation (URN) – Language Definition
25. Escalona, M.J., Koch, N.: Requirements Engineering for Web Applications: A Comparative Study. *Journal on Web Engineering* 2(3), 193–212 (2004)
26. Paech, B., Kerkow, D.: Non-Functional Requirements Engineering - Quality is Essential. In: 10th Anniversary International Workshop on Requirements Engineering: Foundation for Software Quality, REFSQ 2004 (2004),
<http://www.sse.uni-essen.de/refsq/downloads/toc-refsq04.pdf>
27. Boehm: Characteristics of a software quality. North Holland, New York (1978)
28. Gillies, A.C.: Modelling software quality in the commercial environment. *Software Quality Journal* 1, 175–191 (1992)
29. McCall, J.A., et al.: Concepts and definitions of software quality, Factors in software quality. In: NTIS, vol. 1 (1977)
30. Roman, G.-C.: A Taxonomy of Current Issues in Requirements Engineering. *IEEE Computer*, 14–21 (April 1985)
31. Grady, R., Caswell, D.: *Software Metrics: Establishing a Company-wide Program*. Prentice-Hall, Englewood Cliffs (1987)
32. Karat, J.: Evolving the Scope of User-centered Design. *Communications of the ACM* 40(7), 33–38 (1997)
33. Sangeeta, S., Shailey, C.: Goal oriented requirements analysis for Web applications. In: ICCSM 2010, Manila, December 4-5. IEEE, Los Alamitos (2010)
34. Roy, J.-F., Kealey, J., Amyot, D.: Towards integrated tool support for the user requirements notation. In: Gotzhein, R., Reed, R. (eds.) SAM 2006. LNCS, vol. 4320, pp. 198–215. Springer, Heidelberg (2006),
<http://jucmnav.softwareengineering.ca/>
35. van Lamsweerde, A., Letier, E.: *Handling Obstacles in Goal-Oriented Requirements Engineering*
36. Jureta, I.J., Mylopoulos, J., Faulkner, S.: Analysis of Multi-Party Agreement in Requirements Validation. In: ACE 2009 (2009)
37. Bedi, P., Gaur, V.: Prioritizing quality specification of Multiagent systems. In: Proceedings of World Congress on Engineering, WCE 2007, vol. 1 (2007)
38. Jureta, I.J., Faulkner, S., Schobbens, P.-Y.: Clear justification of modeling decisions for goal-oriented requirements engineering. *Requir. Eng.* 13(2), 7–115 (2008)