SWOT ANALYSIS OF REQUIREMENTS ENGINEERING FOR WEB APPLICATIONS

Akshi Kumar[1], Abhilasha Sharma[2]

Dept. of Computer Engineering, Delhi Technological University, (India)

ABSTRACT

Given the complexity of Web Applications, Requirements Engineering is a critical initial stage, but often poorly executed & handled with less relevance. The major consequences are inadequate software architectures, "Unforeseen" problems, viz., budget overruns & production delay, and low user acceptance. Web Applications (Web Apps) require a more extensive and detailed Requirements Engineering (RE) process due to the number of stakeholders involved, the diversity of the requirements, rapidly changing technology and environment (user needs), lack of communication with users of application & the high risk possibility. This study primarily highlights the SWOT Analysis of RE for Web Apps for evaluating the internal potential and limitations and the probable/likely opportunities and threats from the external environment. It provides a basis of solutions to the meta-problem of RE for Web Apps giving a better perspective on the tools and techniques that will be needed to develop a more efficient and cohesive application.

Keywords: Web Application, Requirements Engineering, SWOT

1.   INTRODUCTION

In the development of conventional (pre-Web) applications both practitioners and process experts regard requirements engineering as an imperative phase in the development process since the most common and time-consuming errors as well as the most pricey ones to repair, are errors that result from the inadequate engineering of requirements. An expected alliance of the two notable and active areas of research, requirements engineering & Web, is the emerging area of *Web Requirements Engineering* (WRE). WRE is concerned with addressing the technological challenges of requirements engineering activities specific for web applications. It has been the focus of several recent research projects and papers. The last decade has witnessed numerous comprehensive notations, models, and methodologies that have been introduced in the requirements engineering (RE) field; however, little attention has been paid to methodologies coping with requirements elicitation and finalization specific to web applications [1]. Consequently, WRE entails new demands.

Is RE for the Web really that different than RE for conventional software?

The answer is "YES". And what makes it "Special" are the following key discerning characteristics, viz., Multidisciplinary teams, Unavailability of stakeholders, Rapidly changing requirements & constraints, Unpredictable operational environment, No manual for the user interface, Content Management [2,3].

But, there isn't one single "right way" to do RE among the many methods, techniques, tools, etc. available. For a Web application project, the following questions needs to be addressed:

- What are the critical requirements?
- How should requirements be documented?
- What tools should be used, if any?

As a step to provide an insight to the RE adaptation to Web App, we do a SWOT Analysis that views all positive and negative factors inside and outside the area that affect the growth/scope. The paper is organised as follows: Section 2 gives a brief introduction to a definition & types of Web Applications, followed by a discussion on the prominent area of Requirements Engineering in section 3. Section 4 examines the potential convergence of the RE process in Web application development. Section 5 details the analytical technique of SWOT Analysis to address & urge researchers to make significant improvements to understand and work in the Web RE domain. The paper confers all preliminary information for developers, researchers, academics and students, required to understand, work & evolve the convergence of requirements engineering for Web Applications.
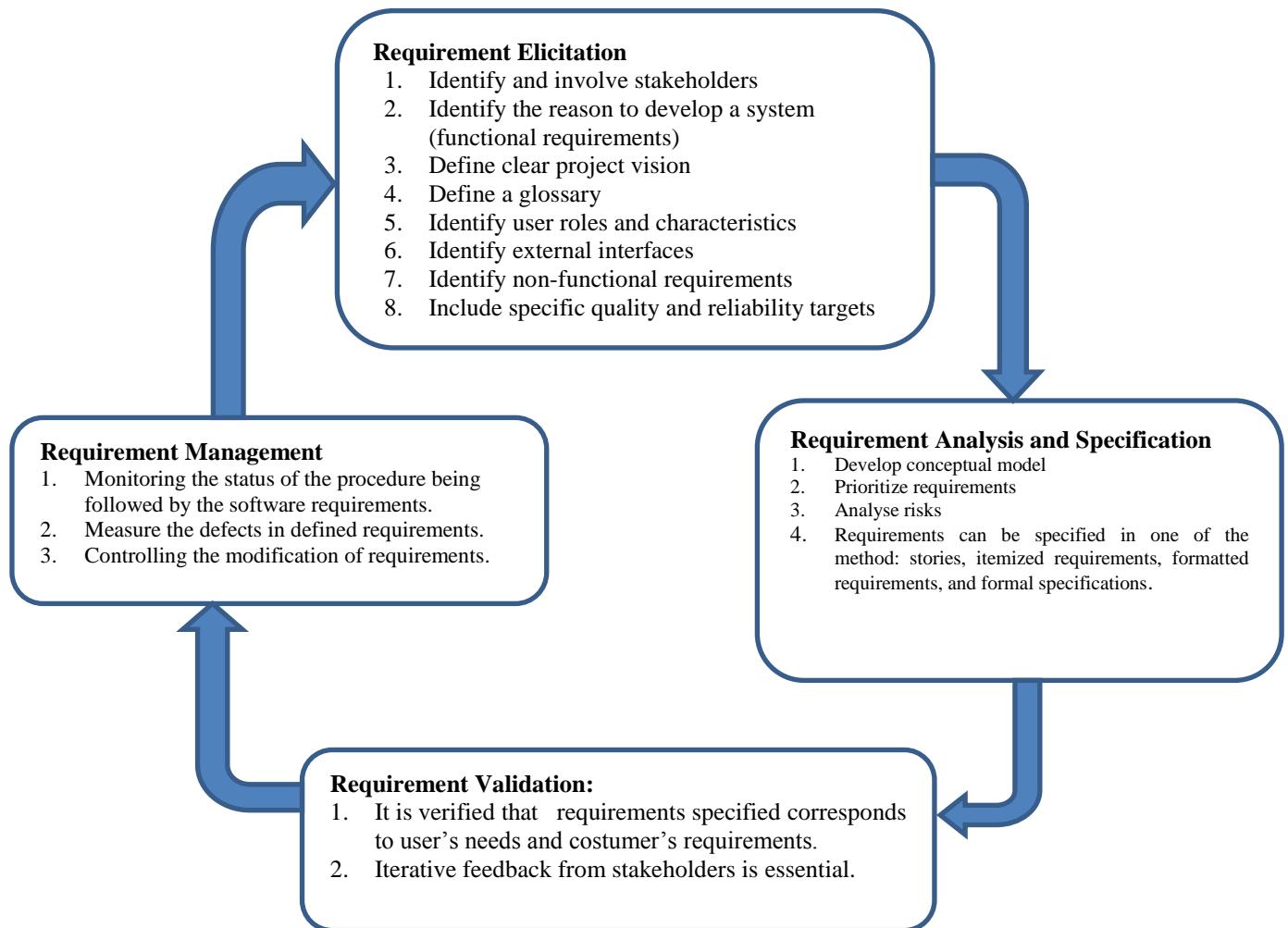
II BACKGROUND WORK

2.1 Requirement Engineering

Requirement Engineering (RE) is defined as the principles, methods, & tools for eliciting, describing, validating, and managing project goals and needs. Requirement engineering is a sub-area of software engineering in which a complete requirement set is developed, specified, and validated against customer requirement set for a software to be developed. A *requirement* is a necessary, quantifiable, and verifiable capability, function, property, characteristic, or behaviour that a product must exhibit to solve a real-world problem, or a constraint that it must satisfy or be satisfied during the development of a product [4]. According to IEEE standard 610.12-1990 requirement is defined as:

- Condition needed to solve a user's problem.
- Condition to be met or possessed by the system to satisfy a formal agreement.
- Documented representation of conditions as in 1 and 2.

Requirement engineering comprises of four processes/phases, namely, Requirement Elicitation, Requirement Analysis and Specification, Requirement Validation, and Requirement Management. Fig. 1 describes the activities involved.

Requirement Elicitation
1. Identify and involve stakeholders
2. Identify the reason to develop a system (functional requirements)
3. Define clear project vision
4. Define a glossary
5. Identify user roles and characteristics
6. Identify external interfaces
7. Identify non-functional requirements
8. Include specific quality and reliability targets

Requirement Management
1. Monitoring the status of the procedure being followed by the software requirements.
2. Measure the defects in defined requirements.
3. Controlling the modification of requirements.

Requirement Analysis and Specification
1. Develop conceptual model
2. Prioritize requirements
3. Analyse risks
4. Requirements can be specified in one of the method: stories, itemized requirements, formatted requirements, and formal specifications.

Requirement Validation:
1. It is verified that requirements specified corresponds to user's needs and costumer's requirements.
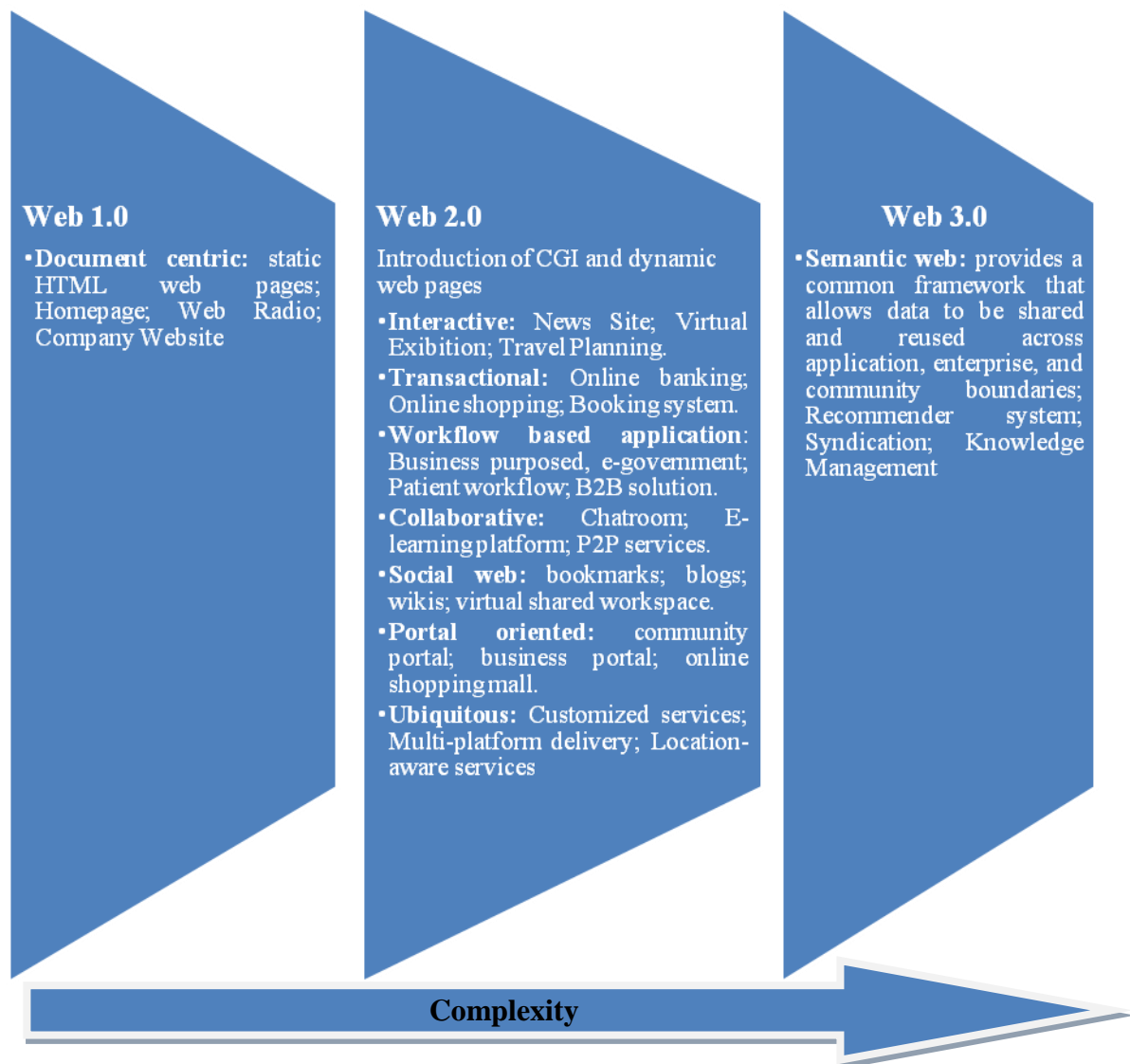2. Iterative feedback from stakeholders is essential.

Fig. 1: Activities of Requirement Engineering Phases

RE is important for a complete & proper understanding (projection) of the system; reduced cost of detecting & correcting errors and to significantly reduce the probability of project failure. Consequences of not applying Requirement Engineering properly are inadequate software architecture, unexpected problems like budget overrun, delay in production & "that's not what I asked for" and low user acceptance.

2.2 Web Application

A Web Application is customization software developed for a network, based on technologies and standards of the World Wide Web Consortium (W3C) that provide a specific purpose and functionality. These are programs which are written for the internet and to be hosted via a browser [5]. Fundamentally, till date, the Web-applications have been divided into nine categories with the development timeline depicting more sophistication and high complexity of new Web Apps. The Web Evolution effectively fits in all the categories of web applications that exist till date. Fig. 2 illustrates the various categories of Web Apps effectively fitted in the abstraction of Web evolution, namely the Web 1.0; Web 2.0; Web 3.0.

Web 1.0
- **Document centric:** static HTML web pages; Homepage; Web Radio; Company Website

Web 2.0

Introduction of CGI and dynamic web pages
- **Interactive:** News Site; Virtual Exibition; Travel Planning.
- **Transactional:** Online banking; Online shopping; Booking system.
- **Workflow based application:** Business purposed, e-government; Patient workflow; B2B solution.
- **Collaborative:** Chatroom; E-learning platform; P2P services.
- **Social web:** bookmarks; blogs; wikis; virtual shared workspace.
- **Portal oriented:** community portal; business portal; online shopping mall.
- **Ubiquitous:** Customized services; Multi-platform delivery; Location-aware services

Web 3.0
- **Semantic web:** provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries; Recommender system; Syndication; Knowledge Management

Complexity

Fig. 2: Web App Categories

Currently, the Web-based application development approaches are largely ad-hoc and personalized lacking rigor and a systematic approach. A legitimate need for disciplined approaches for the very important RE activity is to be addressed to achieve greater success in development and applications of complex Web-based systems.

III ADAPTING REQUIREMENT ENGINEERING TO WEB APPLICATION

Traditional requirement engineering methods are applicable for the system level programming but while developing web applications, there are some issues regarding web which make web application different from traditionally developed system applications. Firstly, the number of web users, i.e., the increased number of stakeholders, customers, users, graphical designers, security experts, etc., apparently. Secondly, issue is diversity of requirements: large number of users fosters new requirements satisfying their own needs in their own way and hence satisfying each user in its own navigational and usability perspective is not an easy task. Other challenges include unavailable stakeholders, volatile requirements and constraints, unpredictable

operational environments, inexperience with Web technologies, particular importance of quality aspects such as usability, or performance [3, 6]. For these major reasons there is a strong need of some special requirement engineering techniques which could handle these issues efficiently.

Adapting RE methods to Web application development needs to address the following questions:

- Which types of requirements are important for the Web application?
- How shall requirements for the Web application be described and documented? What are useful degrees of detail and formality?
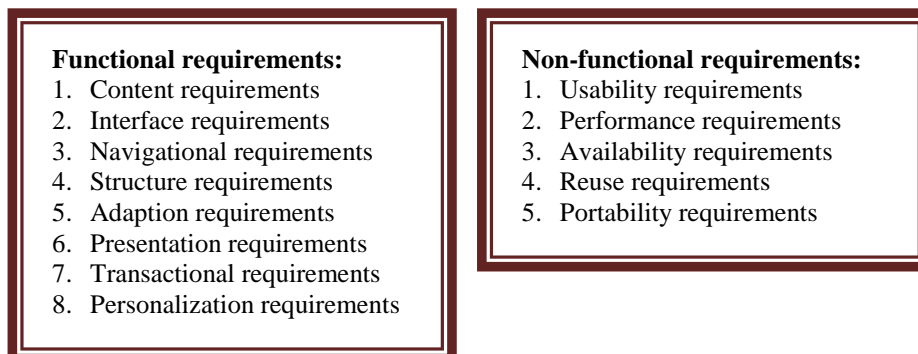- Shall the use of tools be considered? Which tools are suited for the particular project needs?

Functional requirements:
1. Content requirements
2. Interface requirements
3. Navigational requirements
4. Structure requirements
5. Adaption requirements
6. Presentation requirements
7. Transactional requirements
8. Personalization requirements

Non-functional requirements:
1. Usability requirements
2. Performance requirements
3. Availability requirements
4. Reuse requirements
5. Portability requirements

Fig.   3: Web App Requirement classification

From the origin of web applications we are dealing with a lot of requirement engineering techniques being derived very frequently, mend specially for web applications due to their multi-disciplinary nature. Requirements for web applications can be classified in following manner:

IV SWOT ANALYSIS OF RE FOR WEB APPS

SWOT is an acronym for **S**trengths, **W**eaknesses, **O**pportunities, and **T**hreats. A SWOT Analysis (SWOT Matrix) is a structured planning method used to evaluate strengths, weaknesses, opportunities, and threats involved in a project [7]. It is an analytical technique which is commonly used as part of strategic planning as a foundation for evaluating the internal potential and limitations and the probable/likely opportunities and threats from the external environment. It views all positive and negative factors inside and outside the area/field that affect the growth/scope. It looks at the following scenarios, namely, Internal strengths, Internal weaknesses, Opportunities in the external environment, Threats in the external environment.

Table 1: SWOT Factors

SWOT analysis	HELPFUL (to achieve the objective)	HARMFUL (to achieve the objective)
Internal factors (attributes of the organization)	Strengths	Weaknesses
External factors (attributes of the environment)	Opportunities	Threats

The analytical technique of SWOT presented here, addresses & urges researchers to make significant improvements to understand and work in the Web Requirement Engineering (WRE) domain. The following sub-sections expound the details:

4.1 Strengths

4.1.1 Effective classification of diverse users: Problem of large numbers of stakeholders in case of web application is solved by classifying them in different groups for the purpose of requirement elicitation and specification. As web application may be accessed by a large number of diverse users distributed over the world, thus it is necessary to recognise each type of user community so that each may be benefited by purpose of application and WRE techniques fulfil this necessity.

4.1.2 Various types of elicitation approaches*:* Various requirement elicitation techniques are available for web applications out of which anyone could be selected for the required purpose. They may be classified into different categories based on their approach. These may include navigational approach, conceptual approach, interface based approach, prototype-based approach etc. *Navigational* approaches focus on possible patterns of web page accesses by different categories of already classified users. It helps in resetting the navigational pattern of links in best suited way thus satisfying each user community's needs*. Prototype based* approaches reduces time by performing requirement engineering along with design phase thus handling and managing requirements at the same time. It is an iterative process in which requirements are recognised and defined on basis of the prototype developed. *Interface based* approaches along with traditional types of requirements focuses on user interface requirements also. In web applications, user interface plays a very vital role which cannot be neglected.

4.1.3 Powerful Specifications techniques: User interaction diagrams (UID) [8] are very strong tools for aiding web requirement gathering as well as specification process. They serve as a communication bridge between user and designer. Use-case diagrams and UIDs are used in combination to express requirements, which can also be used in validation of requirements. Also use of sketches, mock ups, and more importantly use of natural languages in specification of requirements in web application development makes the validation task effective for both users and analysts, and requirement gathering task easy for users and customers.

4.1.4 Emphasise on non-functional requirements: In terms of web application, non - functional requirements include fast data exchange rate between client and server, high data and network security, user satisfaction, data quality and representation etc. That is why; web requirement engineering techniques gives equal weightage to non - functional requirements. In validation phase of requirements, mainly non - functional requirements are validated.

4.1.5 Content Driven*:* Web Applications generally manage content and not just data & many web applications are document- centric versus data-centric. Content has its own value when talking about web applications. That's why requirements are taken into consideration for producing, gathering, storing, and modifying content.

4.1.6 Social feedback and prioritization of requirements: Social feedback means inviting feedbacks by web users about any issue. This mechanism is very helpful in gathering and analysing requirements. Social feedbacks are of three types: comments, rating, and voting, Comments are open arguments and discussion about requirements to help improve it, Rating is about judging quality on a scale (generally 5 point scale), and voting shows acceptance by user. These social feedback mechanisms might be very helpful for authors of requirements

as they give some hints where a requirement needs to be improved or more precisely defined also these can furthermore provide a valuable starting point for prioritization of requirements [9].

4.2 Weaknesses

4.2.1 Difficult to design architecture: In order to make requirement specification process easier and effective, from the customer's perspective, the task of software developers to design the architecture becomes difficult. In web application, techniques used for requirement specification like sketching, mock ups, use of natural language, and others no doubt make requirement specification process much easier but make the process of designing little bit complicated for designers.

4.2.2 Difficulty in convincing of method: A methodology is primarily created by accumulating the experience of failures or overruns encountered in previous projects and it is difficult to convince that methods or model can be applied in another environment.

4.2.3 Compressed development schedules: The Web Apps needs to be deployed quickly & require design choices to be made accurately within a very compressed development life cycle as the time pressure is heavy. Nearly every project faces shortened product release cycles & escalating pressures for timely delivery, but the Web redefines the notion of accelerated development & deployment [10]. The Web Apps as have short Timeto-Market (TTM) requirement & are Rush to marked. The average project time for Web Projects is under 3 months [11].

4.2.4 Small team with diverse background: Web Apps are often developed by a small team of people with diverse background, skill, knowledge [12]. Moreover, has to accommodate non-developers, especially management, when designing or recommending architecture and policies. As compared to a team of software developers their perception of web & quality of web based systems is considerably different causing confusion & misguided priorities.

4.3 Opportunities

4.3.1 General elicitation technique: All approaches for requirement elicitation for web applications can be combined to develop a single powerful requirement elicitation technique covering all advantages of each distinct approach. Thus a generalized approach can be developed, powerful enough for beating any specific technique in terms of their individual advantages of using them.

4.3.2 Requirement traceability: At current time, requirement traceability is supported by only some of the available requirement elicitation techniques for web application like NDT [13], and OOWS [14]. This is an indivisible feature of any web requirement engineering process as this is the only way to handle newly inserted requirements during application development at some later time.

4.3.3 Early threat detection and avoidance: Current environment enacts on problem once it is detected rather than enacting after detection thus just mitigating the effect of the problem. Avoidance of problem should be more adoptable technique. The key is detecting threats before they result in a failure and proactively switching to an alternative configuration—involving different commitments—to avoid the failure [15]. For this purpose, risk assessment is carried out and risk is associated with each requirement specified. But when dealing with web application development, it seems to be much difficult task to find criticality of the risk due to a huge diversity

in types of users using the application. Thus there is a scope in field of risk assessment and attachment to the requirement.

4.4 Threats

4.4.1 Lack of effective communication: Web is full of diversities, and in such an environment, setting up a communication link between all stakeholders and then executing it effectively, is such a difficult task and any little deviation from planned activity for communication may born possibilities for unclear and redundant requirement extraction. For example, we have a very powerful technique, *Social feedback and prioritization of requirements*, for gathering requirements from diverse users but what if the users themselves do not participate in feedback. This may create severe problem in task of web requirement engineering which in turn will severely affect further development of the web application.

4.4.2 Requirement traceability problem: The inability to trace the human sources of actual requirements and their related information is identified as the crux of the requirements traceability problem [16]. As application development process continues, some new requirements could be introduced, and then in order to check that these newly introduced requirements do not conflict with present requirements, a communication is needed to be established with human sources of the requirements. To solve this problem, link with the individual sources rather than with the groups or community, should be established, but in case of web applications this task may be a threatening one cause of diversity and unavailability of the application users.

4.4.3 Effective classification of users (strength may turn into threat if done incautiously): While creating application for web users, it is mandatory to classify users in proper groups so that actual and complete set of requirements could be gathered. This task must be implemented very cautiously assigning weightage to requirement gathering process and requirements gathered for each user group properly otherwise the strength may turn into threat.

4.4.4 Rapidly evolving implementation environment, encompassing various hardware platform*:* Web Apps are built in a constantly changing environment, i.e., the requirements are unstable & continuously evolve. So, web app development is not a one-time event rather it is a process with an iterative lifecycle to cope with a variety of supporting hardware, software & network [17].

4.4.5 Social, Legal, Ethical issues*:* Web Applications serve global audience & norms, laws may be different where Web Apps are defined as opposed to where they are utilized. Further, Web Apps as discussed are content-driven, so there may be information ownership issues as well. Thus, cultural or regional considerations; privacy, moral, legal obligations & requirements pose a serious threat.

Fig. 4 summarizes the SWOT Analysis of Web Requirement Engineering. The analysis clearly indicates the need of extensive and detailed requirements engineering process for Web Apps.
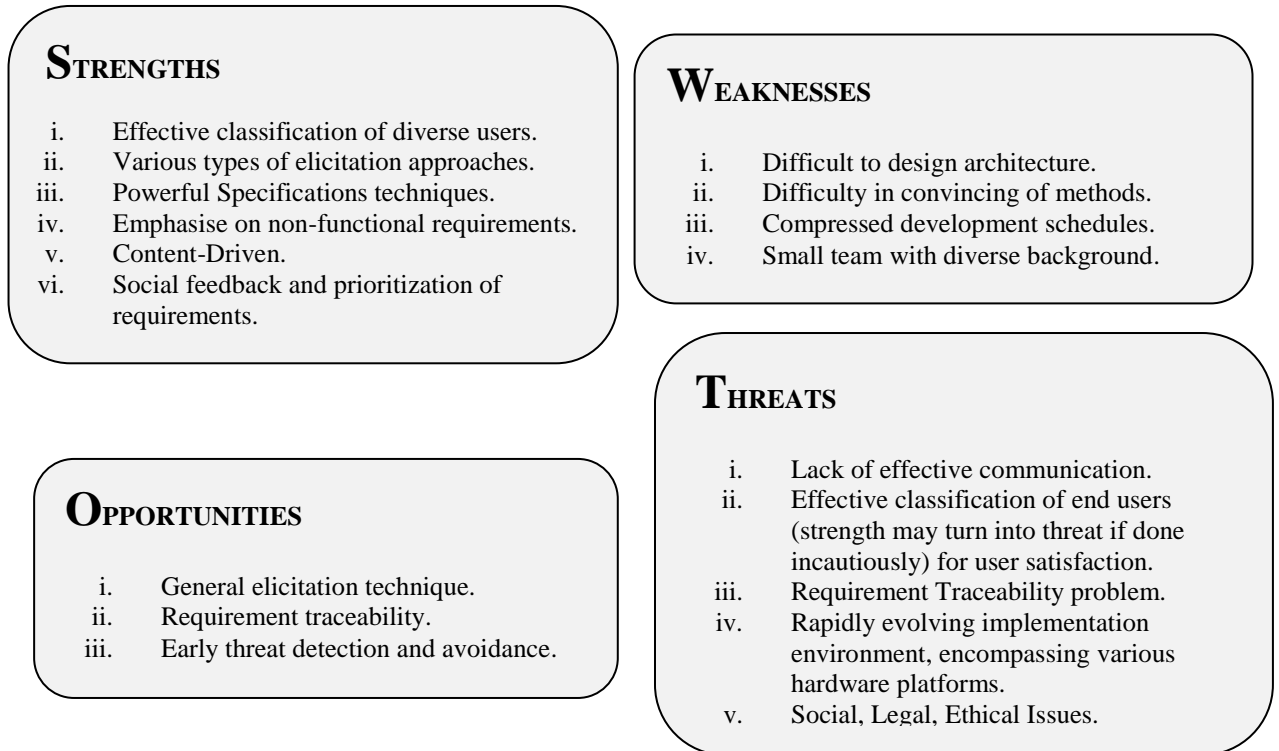
STRENGTHS

  i.   Effective classification of diverse users.
 ii.   Various types of elicitation approaches.
iii.   Powerful Specifications techniques.
 iv.   Emphasise on non-functional requirements.
  v.   Content-Driven.
 vi.   Social feedback and prioritization of requirements.

WEAKNESSES

  i.   Difficult to design architecture.
 ii.   Difficulty in convincing of methods.
iii.   Compressed development schedules.
 iv.   Small team with diverse background.

OPPORTUNITIES

  i.   General elicitation technique.
 ii.   Requirement traceability.
iii.   Early threat detection and avoidance.

THREATS

  i.   Lack of effective communication.
 ii.   Effective classification of end users (strength may turn into threat if done incautiously) for user satisfaction.
iii.   Requirement Traceability problem.
 iv.   Rapidly evolving implementation environment, encompassing various hardware platforms.
  v.   Social, Legal, Ethical Issues.

Fig 4. SWOT Analysis of WRE

V CONCLUSION

Our dependence & reliance on web has increased dramatically over the years. As a result, the development of Web Applications has become more challenging. Software development is generally regarded as the province of computing professionals. Web development affects the entire organization, including its interfaces with the world, and has to accommodate non-developers, especially management, when designing or recommending architecture and policies & thus requires a more extensive and detailed Requirements Engineering (RE) process. SWOT Analysis presented illustrates the positive and negative factors inside and outside the area that affect the growth/scope and necessitates a holistic & proactive approach to Web Requirement Engineering, which will bring the current chaos in web application development under control, minimize risk, and enhance its maintainability & quality.

REFERENCES

[1] D. Bolchini and P. Paolini, Capturing Web Application Requirements through Goal-Oriented Analysis, The 5th Workshop on Requirements Engineering, Valencia, Spain, Nov. 11-12, 2002.

[2] M.J. Escalona and N. Koch, Requirements Engineering for Web Applications – A Comparative Study, *Journal of Web Engineering, Vol.2, N. 3.*2004, 193–212

[3] A. Ginige and S. Murugesan, The Essence of Web Engineering – Managing the Diversity and Complexity of Web Application Development, *IEEE Multimedia*, 2001, 22-25.

[4] R.K. Kandt, Software Requirements Engineering: Practices and Techniques, tech. report document D-24994, Jet Propulsion Laboratory, Nov. 2003.

[5] Acunetix, Web Application Security, 2012. Web Applications: What are they? What of them? Accessed 20th April 2012 from http://www.acunetix.com/websitesecurity/web-applications.htm

[6] Deshpande, Y. and Hansen, S, Web Engineering: Creating a Discipline among Disciplines, *IEEE Multimedia*, Apr.-Jun. 2001, pp. 82-87.

[7] http://en.wikipedia.org/wiki/SWOT_analysis

[8] Vilain, P., Schwabe, D., Sieckenius, C. A diagrammatic Tool for Representing User Interaction in UML. Lecture Notes in Computer Science. Proc. UML'2000. York, England.

[9] Lohmann,S, Dietzold,S., Heim,P., Heino, N, A Web Platform for Social Requirements Engineering. Software Engineering (Workshops) 2009: 309-315.

[10] L.L. Constantine and L.A.D. Lockwood, Usage-centered Engineering for Web Application, *IEEE Software*, Mar./Apr. 2002, 42-50.

[11] Ziemer, S., Web Engineering, http://csgsc.idi.ntnu.no/2004/data/svenz/WE_csgsc.pdf, 02.09.2010.

[12] Y. Deshpande, S. Murugesan, A. Ginige, S. Hansen, D. Schwabe, M. Gaedke, and B. White. Web engineering. *Journal of Web Engineering*, Vol 1., No. 1, October 2002. 3–17.

[13] Escalona, M., Reina, A., Torres, J., Mej´ıas, M., NDT a methodology to deal with the navigation aspect at the requirements   In: OOPSLA Workshop: Aspect-Oriented Requirements Engineering and Architecture Design (2004).

[14] V. P. O. Pastor, J. Fons. OOWS: A method to develop web applications from weboriented conceptual models. In Web Oriented Software Technology (IWWOST'03), pages 65–70, 2003.

[15] FabianoDalpiaz. Social Threats and the New Challenges for Requirements Engineering. RESC 2011: 22-25

[16] Gotel, O. and Finkelstein, A. (1994). *An Analysis of the Requirements Traceability Problem*., Proceedings of the First IEEE International Conference on Requirements Engineering, Colorado springs, 18-22 April. pp.94-101.

[17] Daniel M. Brandon, Software Engineering for Modern Web Applications: Methodologies and. Technologies (Information Science Reference, 2008)