The development and evolution of Web applications is viewed from an engineering perspective that relies on and accommodates the knowledge inherent in patterns. A methodology for pattern-oriented Web engineering (POWE) that deploys patterns as means for assuring the quality of Web applications is proposed. POWE consists of a sequence of steps that include the identification of stakeholder types, following a suitable development process model, identification of relevant quality attributes, and selection and use of suitable patterns. To support decision making and to place POWE in context, the feasibility issues involved in each step are highlighted. The use of patterns during macro and micro-architecture design of a Web application is illustrated. Finally, some directions for future research, including extensions to POWE, are discussed.

**Keywords:** design methodologies; explicit knowledge; information architecture; process model; quality; semiotics; stakeholder; Web applications

**INTRODUCTION**

The success of the Web in various sectors of society has come with its share of failures, many of which are related to quality (Pertet & Narasimhan, 2005; Ricca & Tonella, 2006). In order to elevate, restore, and sustain end-user confidence, it is incumbent upon the providers to ensure the quality of Web applications. In doing so, the providers also need to be realistic in their aims.

In a quality-centric approach, every relevant decision involves an explicit consideration of quality (as perceived by stakeholders or otherwise) as a ‘first-class’ member. There have been several approaches from different viewpoints for understanding and managing the issue of the quality of Web applications. However, they largely focus on: (1) quality as an afterthought rather than as an integral consideration to be embraced early and carried throughout the development process, and/or (2) curative rather than preventative means for addressing quality.

In this article, we address the quality of Web applications by considering these applications as end-products of a pattern-oriented approach to Web engineering. The purpose is to motivate the use of patterns (Buschmann, Henney, & Schmidt, 2007) in a systematic and feasible approach towards the development of
Web applications and to point out the benefits and challenges in doing so.

The organization of the rest of the article is as follows. We first outline the background and state-of-the-art necessary related to quality and patterns for the discussion that follows, and state our position in that regard. This is followed by the presentation of a methodology for pattern-oriented Web engineering (POWE) for systematically addressing the quality of Web Applications. POWE builds on previous work (Kamthan, 2008a) and consists of a non-linear sequence of steps including suitable development process model, relevant quality attributes, and systematic selection and application of suitable patterns, along with the feasibility of each of these steps. Next, we outline some challenges and directions for future research. Finally, we present the concluding remarks.

BACKGROUND AND RELATED WORK
In this section, we briefly present the background and previous work on the need for a systematic approach to the development of Web applications from the perspective of quality and the role of patterns.

For the sake of this article, a Web application is a Web site that behaves like an interactive software system specific to a domain. It typically has a large-size that requires a non-trivial infrastructure: a systematic development process, a team with high-level of knowledge and skills, deployment of additional software on the client and/or server-side, and a schedule comprising of several weeks or months from inception to completion.

Understanding and Addressing the Quality of Web Applications
A commitment to quality of a Web application is both an imperative and a challenge to all stakeholders involved. Indeed, empirical studies have shown (Sharkey, Scott, & Acton, 2006) that the quality of a Web application is directly related to its success.

There have been initiatives in the past for understanding and addressing the quality of Web applications. They contribute in various ways: suggest a quality model that consists of a set of quality attributes (Brajnik, 2001; Mich, Franch, & Gaio, 2003; Offutt, 2002); discuss the significance of a particular quality attribute (Kappel et al., 2006; Mendes & Mosley, 2006); highlight the impact of a quality attribute on a class of stakeholders for specific Web applications (Ricca & Tonella, 2003); recommend means for evaluating certain quality attributes (Hasan & Abuelrub, 2006; Ricca & Tonella, 2006); and so on.

However, these efforts are limited by one or more of the following issues: the view towards quality is, at times, not stated; the relationships, including trade-offs, among quality attributes relevant to Web applications are not always indicated; means of systematically addressing the quality attributes in early stages of development are often suggested informally, say, using guidelines, if at all; or the focus is less on assurance (prevention) and more on evaluation (cure), say, via inspections or testing.

Patterns for Quality of Web Applications
The reliance on past experience and expertise is critical to any development. A pattern is commonly defined as a proven solution to a recurring problem in a given context (Buschmann, Henney, & Schmidt, 2007).

There are several possible views of a pattern. From a knowledge perspective, patterns are a posteriori, explicit, and conceptually reusable form of knowledge. From a structural viewpoint, a pattern is typically described (Meszaros & Doble, 1998) using an ordered list of elements labeled as (pattern) name, author, context, problem, forces, solution, examples, and related patterns. The labels can vary in literature and optional elements may be included to enrich the description. It is this structure that makes patterns more practical in their applicability compared to other expert bodies of knowledge.

In the following, the elements of a pattern are highlighted in italics in order to distinguish them from the main text. There is, however, one
Related Content

Class Level Test Case Generation in Object Oriented Software Testing
[www.igi-global.com/article/class-level-test-case-generation/2644?camid=4v1a](www.igi-global.com/article/class-level-test-case-generation/2644?camid=4v1a)

Building a Semantic-Rich Service-Oriented Manufacturing Environment
[www.igi-global.com/article/building-semantic-rich-service-oriented/2632?camid=4v1a](www.igi-global.com/article/building-semantic-rich-service-oriented/2632?camid=4v1a)

Design and Implementation of Wireless Voltage Monitoring System Based on Zigbee
[www.igi-global.com/article/design-and-implementation-of-wireless-voltage-monitoring-system-based-on-zigbee/182267?camid=4v1a](www.igi-global.com/article/design-and-implementation-of-wireless-voltage-monitoring-system-based-on-zigbee/182267?camid=4v1a)