Chapter II

Software Review
History and Overview

Abstract

The aim of this chapter is to review software review literature. The literature is drawn from Fagan’s software review and forms of review structures. Fagan’s software review includes six-step review processes — planning, overview, preparation, group meeting, re-review, and follow up. The forms of review structures can be classified into Active Design Review, Two-Person Review, Phased Review, and Use of Review Meeting. The literature review also provides an understanding of the IEEE Standard for software reviews and informal software reviews. The common informal reviews include Walkthroughs, Pair Programming, Peer Check, and Pass-Around. It also compares and contrasts bring a comparison these review methods.
Introduction

In the last thirty years, software reviews have been recommended as one of the most cost effective quality assurance techniques in software process improvements and are widely used in industrial practice (Ackerman, Buchwald, & Lewski, 1989; Boehm & Basili, 2001; Fagan, 1976; 1986; Gilb & Graham, 1993; Parnas & Lawford, 2003a, 2003b; Schumeyer & McManus, 1999; Tvedt & Gollofello, 1995; Weller, 1993). The primary goal of a software review is to find defects during the software development life cycle (Biffl & Grossmann, 2001; DeMarco, 1982; Gilb & Graham, 1993; Halling & Biffl, 2002). A defect is considered to be any deviation from predefined quality properties (Boehm, 1981; Fagan, 1986; Humphrey, 2002b; Mathiassen, 2000; Wallance & Fuji, 1989; Will & Whobrey, 2004). The current definition of a software review is broader in scope than the one originally provided by Fagan (1976). Each review variation will be discussed in detail in the following sections.

The software review approach involves a well-defined and disciplined process in which qualified reviewers analyse software for the purpose of finding defects (Parnas & Lawford, 2003b; Ciolkowski et al., 2002). Existing studies such as Fagan’s software review (1976), Freedman and Weinberg’s technical review (1990), and Yourdon’s structured walkthrough (1989) have segmented the analytical framework according to the aims and benefits of reviews (Gluch & Brockway, 1999), the review process, and the outputs of review (Chatters, 1991). Even though some forms of software review (input process and output standard) are covered in IEEE standards, no single clear and consolidated solution that should be used has yet been provided for the software industry (ANSI/IEEE, 1998; Biffl, 2000; IEEE Standard 830, 1993; Johnson, 1998).

Since Fagan’s incremental improvements to software review were first proposed and trailed at IBM in 1972 (Fagan, 1986), several variations of Fagan’s review have been put forward to improve performance, including new methodologies that promise to leverage and strengthen the benefits of software review (Kosman & Restivo, 1992; Miller, 2000; Parnas & Lawford, 2003a, 2003b). Some distinctive structural differences among the review approaches have developed from Fagan’s original proposal. These comprise changing activities or emphasizing different purposes at each stage (Bisant & Lyle, 1989; Knight & Myers, 1993; Martin & Tsai, 1990; Parnas & Weiss, 1985), changing the team number (single and multiple review teams) (Bisant & Lyle, 1989; Kelly, Sherif, & Hops, 1992; Owen, 1997; Porter, Siy, Toman, & Votta, 1997; Porter, Siy, & Votta, 1997), changing the use of review meetings (Biffl & Halling, 2003; Johnson & Tjahjono, 1998; Porter, Votta, & Basili, 1995; Votta, 1993), reducing the number of roles (D’Astous & Robillard, 2001; Porter & Votta, 1994; Russell, 1991), introducing other external supports such as reading techniques (Basili et
An Integrated Approach for Specification and Analysis of Functional and Performance Properties of Concurrent Systems