SOFTWARE DESIGN AND TESTING IN SOFTWARE PROJECTS: APPLICATION OF EVENT-BASED ANALYSIS AND TEST SCENARIOS

ABSTRACT

Software projects are inherently complex, risky and require careful planning. Inadequate planning and specifications, ill defined requirements, poor process of requirement analysis and testing, lack of metrics and measures to compute project’s sheer size and complexity, all together lead to numerous change requests, delays, significant added costs and increases the possibility of errors. But it is also a fact that software failures for the most part are predictable and avoidable by using a good requirement analysis method, proper management of software complexity and proper testing techniques. The entire work in this thesis focuses on these three factors to avoid software failures.

Currently, Object-Oriented Analysis is the most popular method for analysis of requirements. In Object-Oriented Analysis, various techniques have been proposed that either use Natural language processing approaches or employ Use cases to identify classes. Natural language based approaches have their own inherent limitations and several critical reviews have been done on Use case based approaches. It has been iterated that event thinking is equally important way of modeling a system. At the same time there are supporting facts that event modeling introduces rigor and discipline in the Use Case modeling by helping to determine list of Use Cases. So, work has been done in this thesis to use events to develop an alternate method for Object-Oriented Analysis of requirements, for measuring complexity of a system, to generate event sequence-based test scenarios and test cases and to specify and model changes in software requirements either in terms of addition and/or modification of events or event interdependencies. All these contributions are integrated under a single event-based framework. This framework addresses the requirements analysis, design, modeling, program construction, testing, and support activities of a typical software engineering process.

In this work, the principle for Event-based framework and an Event-Meta Model is proposed for Object-Oriented Analysis. Event templates are proposed to describe and document events from textual requirements. A methodology is defined to automatically build analysis level class diagram specification from requirements. A validation is done through a controlled experiment to compare the perceived ease of use and usefulness of the proposed event-based approach of requirements analysis with a more conventional and industry standard Use Case based approach. The findings of the controlled experiment have shown that the proposed
Event-based approach for Object-Oriented Analysis is more effective in terms of perceived ease of use and usefulness as compared to the conventional approach. An Event-based approach has brought a significant change in perception of users about using Object-Oriented Analysis technique. The metric proposed for measuring the complexity of a system from requirement specification uses event flows and event interdependencies. The metric has also been evaluated in terms of Weyuker’s properties. Results of evaluation show that it satisfies 8 out of 9 Weyuker’s properties.

Automation of the proposed event-based framework is done by developing a prototype tool that provides support to extract events from requirements, document extracted events, derive specification for an analysis-level class diagram, automatically compute complexity of the entire system at analysis level and generate test scenarios and test cases from events. The event-based framework finds is application in Event-based systems for establishing a systematic, disciplined, quantifiable software engineering approach to develop, operate, and maintain these systems. Domains for applications can range from control systems, safety critical systems and real time monitoring systems in production, logistics and networking to complex event processing in finance and security.