INTERFACE-DRIVEN SOFTWARE REQUIREMENTS ANALYSIS

Rais Aziz Ahmad

ABSTRACT

Software requirements are one of the root causes of failure for IT software development projects. Reasons for this may be that the requirements are high-level, many might simply be wishes, or frequently changed, or they might be unclear, missing, for example, goals, objectives, strategies, and so on. Another major reason for projects' failure may also be the use of improper techniques for software requirements specification. Currently, most IT software development projects utilise textual techniques like use cases, user stories, scenarios, and features for software requirements elicitation, analysis and specification. While IT software development processes can construct software in different programming languages, the primary focus here will be those IT projects using object-oriented programming languages. Object-oriented programming itself has several characteristics worth noting, such as its popularity, reusability, modularity, concurrency, abstraction and encapsulation. Object-oriented analysis and design transforms software requirements gathered with textual techniques into object-oriented programming. This transformation can cause complexity in identifying objects, classes and interfaces, which, in turn, complicates the object-oriented analysis and design. Because requirements can change over the course of a project and, likewise, software design can evolve during software construction, the traceability of software requirements with objects and components can become difficult. Apart from leading to project complexity, such a process can impact software quality and, in the worst-case scenario, cause the project to fail entirely. The goal of this article is to provide interface-driven techniques that will reduce ambiguity among software requirements, improve traceability and simplify software requirements modelling.

FULL TEXT:

PDF

DOI: http://dx.doi.org/10.19044/esj.2016.v12n30p40

DOI (PDF): http://dx.doi.org/10.19044/esj.2016.v12n30p40

European Scientific Journal (ESJ)

ISSN: 1857 - 7881 (Print)
ISSN: 1857 - 7431 (Online)

Contact: contact@eujournal.org

To make sure that you can receive messages from us, please add the 'eujournal.org' domain to your e-mail 'safe list'. If you do not receive e-mail in your 'inbox', check your 'bulk mail' or 'junk mail' folders.

Publisher: European Scientific Institute, ESI.
ESI cooperates with Universities and Academic Centres on 5 continents.
The software requirements are one of the root causes of failure for IT software development projects. Reasons for this may be that the requirements are high-level, many might simply be wishes, or frequently changed, or they might be unclear, missing, for example, goals, objectives, strategies, and so on. Another major reason for projects' failure may also be the use of improper techniques for software requirements specification. Currently, most IT software development projects utilise textual techniques like use cases, user stories, scenarios, and features for software requirements elicitation, an... In systems engineering and software engineering, requirements analysis encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking account of the possibly conflicting requirements of the various stakeholders, analyzing, documenting, validating and managing software or system requirements.