

The Blind Men and the Elephant: Views of Scenario-Based System Design

Scenario-based system design is an approach that employs scenarios as a central representation. The approach encourages user involvement in system design, envisions the uncertain future tasks of the system users, and enhances ease of developing instructional materials. It provides a good brainstorming tool for planning and allows the stakeholders to consider alternative choices in decision making. It addresses dynamic, multiple, parallel, and/or distributed factors in a manageable manner. This rich variety of roles is selectively used from different viewpoints in diverse communities including human-computer interaction, strategic planning, requirements engineering, and object oriented analysis/ design. The purpose of this paper is to help build a common language in software design for stakeholders from a wide range of disciplinary backgrounds providing an overview of scenario- based system design. Human-computer interaction can be more effective if it can cooperate with and leverage efforts in other fields. Likewise, human-computer interaction can more effectively contribute to the evolution of the other fields.

Scenarios in System Design

A scenario is a description of a person's interaction with a system. Scenarios help focus design efforts on the user's requirements, which are distinct from technical or business requirements. Scenarios may be related to 'use cases', which describe interactions at a technical level. Unlike use cases, however, scenarios can be understood by people who do not have any technical background. They are therefore suitable for use during participatory design activities.

Strategic Planning

Strategic planning is an organizational management activity that is used to set priorities, focus energy and resources, strengthen operations, ensure that employees and other stakeholders are working toward common goals, establish agreement around intended outcomes/results, and assess and adjust the organization's direction in response to a changing environment. It is a disciplined effort that produces fundamental decisions and actions that shape and guide what an organization is, who it serves, what it does, and why it does it, with a focus on the future. Effective strategic planning articulates not only where an organization is going and the actions needed to make progress, but also how it will know if it is successful.

Human-Computer Interaction

Human-Computer Interaction (HCI) is a multidisciplinary field of study focusing on the design of computer technology and, in particular, the interaction between humans (the users) and computers. While initially concerned with computers, HCI has since expanded to cover almost all forms of information technology design.

Requirements Engineering

Requirements engineering is a process of gathering and defining of what the services should be provided by the system. It focuses on assessing if the system is useful to the business , discovering requirements , converting these requirements into some standard format and

checking that the requirements define the system that the customer wants. In practice, requirements engineering isn't sequential process, it's an iterative process in which activities are interleaved.

Object-Oriented Analysis/Design

It's a structured method for analyzing, designing a system by applying the object-orientated concepts, and develop a set of graphical system models during the development life cycle of the software. There are three typical scenario-based approaches: Jacobson's use case approach, Wirfs-Brock's responsibility-driven approach, and Koskimies, Systa, Tuomi, and Mannisto's automated modeling support approach. According to Jacobson, a use-case depicts how a user or another system uses a system. A basic concept behind the use-case approach is that a system is well described from a black-box view. A use-case model consists of two elements: actors and use-cases. Another object-oriented approach relating to scenarios is the responsibility-driven approach by Wirfs-Brock. She states that the approach emphasizes informal methods for characterizing objects, their roles, responsibilities, and interactions. The approach uses cards as tools; they are called "CRC cards". Koskimies and his colleagues proposed automated support for modeling object-oriented software. They follow the definition of scenarios in the Object Modeling Technique (OMT), a commonly used object-oriented analysis/design method.

Relationship of the Four Communities

The four communities have a nested relation, in which the latter two communities can be categorized into software engineering. From inside to outside, there are three layers: software engineering, human-computer interaction, and strategic planning. The nested structure describes the focus of each field based on the degree of tangibility of the target content of scenarios. Software engineering scenarios focus on real world objects or physical artifacts; furthermore, they include scenarios of the existing system use. Software engineering scenarios are relatively small in scope. They typically include keystroke- and command-level scenarios. Human-computer interaction studied human aspects in traditional system design approach and the use of systems constructed from that approach. It grew out of software engineering research and has focused on practice.

Scenarios as Common Language

Scenario-based techniques contextualize and concretize design thinking about people and technologies. Requirements engineering scenarios help elicit and refine functional descriptions of future systems. Object-oriented analysis/design use-cases of the system identify the possible event sequences of the system to accurately model the domain objects, data structures, and behaviors. These different applications of scenarios emphasize different viewpoints and different resolutions of detail, and they address different purposes.