Chapter XII

Tool to Support the Design Methodology: A Configurable Editor

INTRODUCTION

In order to provide support to the designer in the design of the whole complex Web information system, it is essential to provide a suitable tool that hides the intrinsic complexity of the methodology supporting the designer in the application of the same that is often complex, and the tool has to be able to translate the design made up in a machine readable format to be able to use this design in the following automatic code generation of the Web application according to a model-driven approach.

In this chapter, we introduce the design and implementation of the editor made up mainly of the architecture presented (and based on Eclipse™ Platform as illustrated in the preceding chapter) and on the methodological steps of integration among the several editors for the design and implementation of these guidelines.

REQUIREMENTS

We now describe in detail the requirements on which the following design and implementation of the editor is founded.
First of all, we highlight that the tool that we want to provide is not a simple editor that supports the proposed methodology but a customization tool in which to make all of the editors meet allowing realization of the intermediary designs provided by the methodology and that provides the possibility of the designer to work with more designs at the same time. Besides, the tool has to be easily extensible in the sense that has to be simple adding new editors to support possible extensions of the methodology.

For this reason, we do not simply deal with an editor, but we speak about a base framework inside which to make the different editors meet.

Starting from the framework, and through well-defined configurations, it has to be possible to get the desired standalone editor: it will integrate, according to the particular requirements, one or more editors in order to support the methodology. The final product, starting from the framework opportunely shaped, will take the name of CASE tool. We want to guarantee an easy and elevated customization of the final product: the CASE tool has to be able to be built on the framework with simple passages. In this way, it will be possible to easily get the CASE tool that differs, for instance, for the number of integrated editors or activated features. It must be simple to add or to modify languages of modeling or methodological guidelines for the CASE tool.

The CASE tool must also allow design of a Web application applying in everything or partly the methodologies previously exposed. Although a project plans, as principle, the use of different methodologies of design, the design of the application must also have been made possible with only some of these. For instance, the designer has to be free, if he/she desires to directly create a P-IDM or P-IDM Process diagram.

The CASE tool will have to allow the creation of a project, correspondent to the design of the Web application. Within the project so made, it will be possible to draw different categories of diagrams and more diagrams for category. It is fundamental, therefore, that the usability of the final application has to be guaranteed to be an intuitive and easy to use environment to the designer; particularly, the integration among the different modeling must be easily, allowing the passage from modeling the other (when possible) in a simple and fast way.

A very important requisite is the implementation of the methodological guidelines of integration that allow the transition from a methodology of design to the other; this requisite involves the communication among the editors of the CASE tool. The complexity of the methodological guidelines must, nevertheless, be completely hidden to the final user.

The designer has to be able to work on more editors contemporarily and has to have the possibility of realizing or completing a well-defined diagram eventually using elements of other open diagrams; in this case, the application has to perform,
An Experimental Study for the Effect of Stop Words Elimination for Arabic Text Classification Algorithms

Bassam Al-Shargabi, Fekry Olayah and Waseem AL Romimah (2013). *Network and Communication Technology Innovations for Web and IT Advancement* (pp. 184-190). www.igi-global.com/chapter/experimental-study-effect-stop-words/72761?camid=4v1a