Towards Empirical Validation of Design Notations for Web Applications: An Experimental Framework

Paolo Tonella*, Filippo Ricca*, Massimiliano Di Penta**, Marco Torchiano***

*ITC-irst, Trento, Italy
**University of Sannio Benevento, Italy
***Politecnico di Torino, Italy

tonella@itc.it, ricca@itc.it, dipenta@unisannio.it, marco.torchiano@polito.it
Motivation

- Web application design requires to deal with several concerns
- Several web application modeling techniques exist
  - WebML [Ceri et al., 2002],
  - UWE [Knapp et al., 2004]
  - WSDM [Troyer et al., 1998]
  - OOHDM [Schwabe et al., 1998],
  - Conallen extension to UML [Conallen, 2000].
- Problems arise during maintenance
  - Documentation may not exist
  - Difficult to keep it aligned
  - Need for reverse engineered diagrams
  - Do these techniques really help?

NEED TO EXPERIMENT THEM!
Experimental template

- Follows the guidelines by Wohlin et al or Juristo and Moreno.

<table>
<thead>
<tr>
<th>Goal</th>
<th>Analyze the support given by Web design notations to the comprehension and modification activities during evolution.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null hypothesis</td>
<td>No significant effect on effectiveness of task execution and quality of the result.</td>
</tr>
<tr>
<td>Main factor</td>
<td>Design notations being validated.</td>
</tr>
<tr>
<td>Other factors</td>
<td>Systems, tasks, subjects and subject skills, training, tools.</td>
</tr>
<tr>
<td>Dependent variables</td>
<td>Knowledge acquired, capability to locate changes precisely, quality of the result.</td>
</tr>
</tbody>
</table>
# Design template

<table>
<thead>
<tr>
<th>Lab</th>
<th>Group 1</th>
<th>Group 2</th>
<th>Group 3</th>
<th>Group 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab 1</td>
<td>System 1 Treatment 1</td>
<td>System 1 Treatment 2</td>
<td>System 2 Treatment 1</td>
<td>System 2 Treatment 2</td>
</tr>
<tr>
<td>Lab 2</td>
<td>System 2 Treatment 2</td>
<td>System 2 Treatment 1</td>
<td>System 1 Treatment 2</td>
<td>System 1 Treatment 1</td>
</tr>
</tbody>
</table>
Subjects’ training

- Training of students on
  - the technique to be experimented
  - the tool

- Assessment of ability through questionnaires
  - Use of ability for assignment of subjects to groups and for blocking
Instantiating the template

- The **goal** of the study is to analyze the use of Conallen’s stereotyped UML diagrams with the **purpose** of evaluating their usefulness in Web application **comprehension**, **impact analysis** and **maintenance**.

- The **quality focus** is ensuring high comprehensibility and maintainability, while the **perspective** is multiple:
  - **Researcher**: evaluating how effective are the stereotyped diagrams during maintenance.
  - **Project manager**: evaluating the possibility of adopting a Web application design and reverse engineering tool in her/his organization.
Example: Conallen
Example: UML
Hypotheses and context

- The use of stereotyped reverse engineered class diagrams (versus non-stereotyped reverse engineered class diagrams) does not significantly affect:
  - $H_{01}$: The comprehension level
  - $H_{02}$: The effectiveness of impact analysis
  - $H_{03}$: The effectiveness of maintenance
Factors

◆ Main factor:
  - UML diagrams complemented with Conallen’s diagrams
    • Representing pages, links, etc.
  - UML (reverse engineered) diagrams

◆ Other factors:
  - System
  - Subjects
  - Training
  - Tools

◆ Dependent variables:
  - Comprehension level
  - Accuracy of impact analysis
  - Quality of modified code
Context

- **Objects:** different Web applications:
  - TuDu – web application for maintaining todo lists
  - DMS – web based document management System
  - Claros – webmail System
  - Web based workflow management System

- **Subjects:** different universities, different abilities
  - University of Trento – Master students
  - University of Sannio – Graduate students
  - Politecnico di Torino – Undergraduate students
## Experiment design

<table>
<thead>
<tr>
<th>Lab</th>
<th>Group 1</th>
<th>Group 2</th>
<th>Group 3</th>
<th>Group 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab 1-a (comprehension and impact analysis)</td>
<td>Claros/Conallen</td>
<td>Claros/UML</td>
<td>WMS/Conallen</td>
<td>WMS/UML</td>
</tr>
<tr>
<td>Lab 1-b (maintenance)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lab 2-a (comprehension and impact analysis)</td>
<td>WMS/UML</td>
<td>WMS/Conallen</td>
<td>Claros/UML</td>
<td>Claros/Conallen</td>
</tr>
<tr>
<td>Lab 2-b (maintenance)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Instrumentation

Material to be distributed:
- Experiment instructions
- Diagrams
- Source code
- Survey questionnaires

◆ Measurement instruments:
  - **Comprehension**: use of questionnaire - number of correctly answered questions and time needed to answer them
  - **Impact analysis**: use of questionnaire - number of correctly answered questions and time needed to answer them
  - **Maintenance**:  
    • Functional behavior of changed code (passed test cases).
    • Time required to implement the changes.
    • Flaws in new design (determined through inspections).
    • Code quality (determined through inspections).

◆ **Time measured through our experimentation framework**
Operation

- Introductory tutorial to the experiment
- Perform tasks:
  - Comprehension: answer questions
  - Impact analysis: answer questions
  - Maintenance: receive a maintenance request and enact it
    - On the diagrams
    - On the source code
- Finally, fill a survey questionnaire
  - How the tasks were clear
  - Enough time
  - Diagrams useful
  - Percentage of time spent on the diagrams
  - Ease of use of tools
  - Use of additional documents (manuals/etc)
Summarizing…

- Different Web application modeling and design techniques exist
- Pros and cons of each technique need to be experimented
- We are focusing on the usefulness of techniques in software comprehension and maintenance tasks
  - Conallen’s notation
  - Planning series of controlled experiments
- Future work: experiment other techniques / compare different ones when possible