Survey justification

This survey was run in compliance with surveying procedures [1, 2] that validate its results. The key requirements were:

- Surveys should have no more than 30 questions, divided into at most five subject areas. More questions are likely to overwhelm or overtax the respondent’s attention. This survey has 24 questions on the EUD paradigm, plus six questions concerning personal details, like the name, identification, age, educational attainment, employment, previous programming experience, etc. The 24 questions are divided into five different groups or sections:
  - Real expected use of the EUD paradigm (5 questions)
  - EUD problem-solving validity (5 questions)
  - Usability (5 questions)
  - Functionality (5 questions)
  - Overall rating (4 questions)

- The survey contains a section for user characterization and another for general findings.

- A survey should not deal with over 15 different topics or categories. This survey has 12 topics, listed in Table 1. These topics are evaluated in the different sections listed above, without the respondent being explicitly aware of the topic underlying each question.

- All the surveyed topics should be supported by two or more questions that ask basically the same question to check respondent consistency. For this reason, the survey contains 24 questions surveying 12 topics, meaning that there are two questions within the survey evaluating each topic listed in Table 1.

- The survey uses both affirmative and negative question statements to rule out unmeditated responses based on respondent positiveness, and their tendency to give set responses. Therefore, as each topic has two associated questions, each statement will be reverse worded. This way each topic or category will be evaluated consistently, and any incompatibilities or inconsistencies will be detected.

This survey evaluates the following general topics:

<table>
<thead>
<tr>
<th>General topic</th>
<th>Number of Q</th>
<th>yes/no</th>
</tr>
</thead>
<tbody>
<tr>
<td>C0 Real expected use of the EUD paradigm by the respondent</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C1 EUD’s future, real use and success</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C2 Personal realization</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C3 Forecast network externality of EUD</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C4 EUD vs. traditional programming</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C5 EUD metaphor abstraction</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C6 Pre-/postconditions as an EUD composition technique</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C7 EUD solution testability and maintainability</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C8 EzWeb/FAST usability</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C9 EUD validity for programming illiterate users</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C10 Design element publication and catalogue</td>
<td>2</td>
<td>1/1</td>
</tr>
<tr>
<td>C11 Solution conformity to requirements using the EUD paradigm</td>
<td>2</td>
<td>1/1</td>
</tr>
</tbody>
</table>

Table 1. General topics of evaluation
As mentioned above, there are five major sections, combining the topics listed as reverse-worded statements, giving priority to topics related to the section heading.

Respondents should rate the 24 items using a five-point Likert scale, scored as:
1-I totally disagree
2-I disagree
3-Neither agree nor disagree
4-I agree
5-I totally agree

Given this premise, the 24 questions on the EUD programming paradigm are listed below, grouped as per the above five sections. Each question is followed by two notations, indicating: first, the topic to which the question refers (see Table 1) and, second, whether the question is stated affirmatively ("yes", where the best score is 5 and the worst is 1) or negatively ("no", where the best score is 1 and the worst is 5).

Note that the 24 questions are numbered as of 7 because the final survey will include six preliminary questions about the respondent’s personal particulars in order to categorize and characterize the sample under study.

---

**Section 1. Real expected use of the EUD paradigm**

Q7.- EzWeb/FAST is a satisfactory means for creating solutions to meet personal needs when it is not feasible to develop a traditional solution due to time and/or budget constraints. C0 “yes”

Q8.- It is rewarding to use tools like EzWeb/FAST and be able to rapidly and simply create mashups. C2 “yes”

Q9.- Domain experts, web programmers and service providers should consider the EUD paradigm as a design vision to be taken into account. C1 “yes”

Q10.- The more people that adopt the EUD paradigm the easier it will be to find useful design components and create end-user solutions. C3 “yes”

Q11.- The EUD paradigm enormously simplifies the stages of implementation, testing, debugging and any modifications to account for changing requirements of the EUD solution development process. C4 “yes”

---

**Section 2. EUD problem-solving validity**

Q12.- It was complicated to create a solution to the stated problem using EzWeb/FAST. C2 “no”

Q13.- The design components available in the EUD paradigm do not meet the needs of real-world problems: parts are either overly general or overly specific. C5 “no”
Q14.- The communication mechanism between the design elements is not suitable for solving the problems that end users are likely to have.  
C6 “no”

Q15.- The solution created using EzWeb/FAST can be straightforwardly evaluated in a stepwise manner to check that it is error free and be able to create increasingly complex solutions. C7 “yes”

Q16 Using EzWeb/FAST, a change in the end-user requirements leads to major rework to tailor the solution to the new problem. C7 “no”

Section 3. Usability

Q17.- The EzWeb/FAST EUD platform is easy to use even first time round. C8 “yes”

Q18.- Most people could learn to use EzWeb/FAST to develop end-user solutions. C9 “yes”

Q19.- I get the feeling that it is not easy to create real-world solutions using EzWeb/FAST. C8 “no”

Q20.- The development model interface and support built into EzWeb/FAST are too complex for end users to be able to create solutions. C1 “no”

Q21.- Users need a lot of additional training before they will be able to use EzWeb/FAST effectively to develop their own solutions. C1 “no”

Section 4. Functionality

Q22.- It is easy to link several components in the EzWeb/FAST using pre- and postcondition mechanisms. C6 “yes”

Q23.- Useful design components are easy to locate thanks to EzWeb/FAST catalogues. C5 “yes”

Q24.- It is hard to publish new design components as gadgets for use in composite applications. C10 “no”

Q25.- The composite system built did not respond as expected. C11 “no”

Q26.- It is hard to create a composite solution to a specific problem using EzWeb/FAST (considering that the catalogue is well enough populated with design components). C3 “no”

Section 5. Overall rating

Q27.- Which of the following do you think is the most realistic development time ratio considering two development options for a real problem: a) implement a solution from scratch and b) use the EUD paradigm?: C4 “no”
- The EUD paradigm can reduce development time/workload enormously (e.g., a EzWeb/FAST workload of 1 hour can save 100 hours of traditional development time)
- The EUD paradigm can reduce development time/workload appreciably (e.g., a EzWeb/FAST workload of 1 hour can save 10 hours of traditional development time)
- The workload for the EUD paradigm and for programming a solution from scratch is similar.
- The EUD paradigm takes more development time than traditional programming.
- The EUD paradigm does not always manage to produce a valid solution to a set problem, even if the catalogue contains the necessary components.

Q28.- Using the EUD paradigm and tools like EzWeb/FAST, any user (no matter how much programming knowledge they have) can create their own solution to a particular problem. C11 “yes”

Q29.- Users need to know how to program to create functional and stable solutions using EzWeb/FAST. C9 “no”

Q30.- Developing and tailoring new design components for EUD platforms like EzWeb/FAST will be key occupation of information technology enterprises in the future. C10 “yes”

References