J. King Saud Univ., Vol. 19, *Comp. & Info. Sci*., pp. 17-37, Riyadh (1427H./2007)

An Empirical Process for Evaluating and Selecting AWEM Environments: "Evaluation Stage"

Abdullah Alghamdi
Department of Information Systems,
College of Computer & Information Sciences, King Saud University,
P.O. Box 87806, Riyadh 11652, Saudi Arabia
Email: ghamdi@ccis.ksu.edu.sa

(Received 13 June 2006; accepted for publication 10 March 2007)

Abstract. This paper elaborates on the evaluation stage of an empirical process whose purpose is to evaluate and select a best-of-breed AWEM, Automated Web Engineering Methodology environment. The process is basically comprised of four major stages, namely: characteristics identification, screening of available AWEM environments, evaluation, and selection.

During the evaluation stage, an “evaluation scheme”, which serves specific web application domains reflecting the organization's perspective towards the AWEM environment, is created. The idea is to make the evaluation process more user-centric. The actual evaluation is then conducted based on the results gained through the development of real pilot projects.

The paper contributes to the current research in web engineering area by proposing an evaluation mechanism through the introduction of a so-called "evaluation scheme", a sub-set of predefined essential criteria. Another major contribution is the introduction of an evaluation algorithm for weighing and rating various characteristics and alternatives that will eventually assist in making a final decision. Both, the process and the algorithm, were fully automated and evaluated on real world cases using the AWEM-ESS, which is an evaluation and selection system built specifically for this purpose.

Keywords: Web engineering, Automated Web Engineering Methodology environments, Evaluation and selection of Web Engineering Methodology environment.

1. Introduction

A crucial problem currently facing the developers of web applications is the absence of a sufficient automatable web engineering methodology. This has led, in most cases, to an ad hoc development of web applications, lacking systematic approach, quality control and assurance procedures, and eventually a failure of many large web application projects.

The recent introduction of Automated Web Engineering Methodology environments has a vision of equipping the web development community with expressive yet powerful techniques to increase the efficiency and quality of their applications. Although a handful number of those AWEM environments is available there, this only represents a light at the end of the tunnel. However, their maturity and sufficiency have not been widely or thoroughly verified yet.

One of our currently ongoing maxi projects is to build a process for evaluating, selecting and eventually customizing an AWEM environment capable of satisfying the needs of the web development community. Here, we focus on the most important part of this process, the evaluation stage, and its implied algorithm.

1.1. AWEM

The web engineering is a new emerging discipline that is still immature, and lacks a vertical and horizontal coverage with respect to both theoretical backgrounds and best practices.

Most of the research efforts in the WE discipline so far concentrate either on the process or on the automation tools. Only a handful of these efforts have dealt with both parts jointly [11]. This distinction has negatively affected the proper utilization of web engineering methodologies and processes by the functional stakeholders, i.e. developers, architects, designers, etc…

In this research, we introduce a new notion to overcome the inherited problem caused by the process/automation distinction and to increase the productive use of the web engineering methodology "WEM". The invented AWEM notion is primarily and exclusively concerned with those WEMs that are automated by CASE tools.

Consequently, AWEM environment can be defined as "*a collection of well integrated computerized tools aimed at supporting all the WEM activities within a web application development life cycle to create high-quality web applications*."

1.2. Requirements of AWEM evaluation method

Based on the definition above, we can identify a number of characteristics that make AWEM environment different from other currently available traditional software engineering technologies. Therefore, any well established evaluation method must take these unique characteristics into consideration and perform the selection and evaluation activities in a consistent manner.

- AWEM is not just a mere SW tool. It is rather a methodology that is automated by a SW tool. Thus, the focus is concentrated more on methodological rather than technical aspects.

- Web application development process generates a complex array of artifacts that vary based on the web application domain. Each web application domain, i.e. E-commerce, E-government, E-learning etc., has its own architecture, stakeholders, deliverables, standards and so on. Thus, in order to make the evaluation and selection process more realistic and reflect the actual user's field of interest, the domain of the web application must be taken into consideration. Consequently, AWEM environment is largely affected by this nature that necessitate the provision of a vast variety of models and techniques that support modeling such applications. This would yield a large number of characteristics that must be used for evaluating any AWEM environment. This is reflected on the requirements (or characteristics) structure of AWEM environment that is supposed to assist in building such web applications. Thus, the nature of these characteristics and the way they are structured are different from traditional software tools, as they include both qualitatively and quantitatively measured characteristics
- Also, the huge number of characteristics necessitates the existence of a customizable evaluation process that allow evaluators to narrow down the characteristics to suit the requirements of various spectrums of web applications, i.e. E-Gov, E-Biz, etc.. Thus, the domain of the application affects which criteria to use and the weight to assign to each.
- The evaluation process must also support the reuse of predefined characteristics which count in hundreds to save the evaluator's time.
- The process must provide a central characteristics repository to ease the transition from the characterization stage to the evaluation.
- The majority of characteristics (criteria) are qualitative, which requires a proven evaluation technique that reflects the actual weight of the criteria.
- The evaluator of qualitative characteristics requires special guidance to conduct the actual evaluation. Each atomic characteristic is measured according to a set of predefined measurement guidelines that must be defined based on the implication of each characteristic. AWEM evaluation process must define quality measures in order to facilitate and standardize the way each characteristic is evaluated.
- Qualitative criteria require a technique that measure the consistency ratio of the evaluator and reject inconsistent evaluations which would be the likely result of fatigue in evaluators.
- The AWEM evaluation process must be fully automated in a friendly multi-user environment.
- The evaluation process must allow for narrowing down the evaluation of alternative AWEMs against specific super or sub criteria.
- The AWEM evaluation process must allow for a hybrid type evaluation. There are three different types of evaluation that can be applied to the AWEM selection process as follows: quantitative that measure the benefits in a

quantitative way, qualitative assess features and characteristics of each AWEM and hybrid which is a combination of the two, quantitative and qualitative
- The definition, weighing and evaluation of characteristics (criteria) must be integrated within the same environment and very easy to use.
- The roles and responsibilities in the evaluation process must reflect the special nature of the AWEM

1.3. Related work

Due to the immaturity of the AWEM field, you can hardly identify research efforts that deal specifically with the evaluation and selection process of AWEM environments. This also implies the absence of any evaluation technique in the literature for WE methodologies and their tools.

However, generally speaking, the AWEM has its roots in the software engineering arena and the supporting CASE tools environments. Therefore, tracing back the literature in these areas reveals that some efforts have been conducted in this respect [10, 13, 15]. Most of these efforts are outdated and cannot be considered a potential reference for a continuously evolving field such as the web engineering. However, some of them suggested best-practice based techniques that can be slightly utilized in environments with similar characteristics [2, 4]. In her work, Koch illustrated a number of web engineering processes with some comparisons between their different functional constructs [5, 12].

There are also some international standards that have been designed to guide and standardize the work of acquiring and implementing software engineering tools [7-9].

1.3.1 Surveying software tools evaluation methods

A number of methods for the evaluation of software tools in general have been proposed in the literature. Each of these methods emphasizes one or more critical aspects of AWEM evaluation characteristics outlined above. This section will discuss highlights of these proposed techniques. The overriding goal is to identify those aspects of the methodologies that might meet the general requirements of AWEM evaluation process.

1.3.1.1. PORE method: Maiden and Ncube [18] propose a template-based approach to requirements definition that depends on evaluating software tools. The method is based in an iterative process of requirements acquisition and product evaluation. This method integrates some techniques, methods and tools, such as: knowledge engineering techniques, multi-criteria decision making methods, and requirements acquisition techniques. It also provides guidelines for designing product evaluation test cases.

Although the PORE method includes some requirements acquisition techniques, the templates only give a preliminary view of the steps necessary to perform a systematic evaluation. It is not clear how the requirements are specified in the evaluation process and how products are eliminated (i.e. do not capture the decision rationale).

1.3.1.2. CISD method: Tran and Liu [19] propose, within the CISD model, a two-stage software tools selection process. The first stage is product identification, where candidates are identified and classified. The data for this stage is gathered via vendor documentation, personal experience or other means. The results are a list of potential candidates. The second is evaluation, where the final candidates are chosen and unsuitable candidates are eliminated. In this stage, the authors depend on concrete techniques. They state that the software tool evaluation phase requires the extensive use of prototyping techniques. They argue that prototyping is the only way to practically evaluate a software tool candidate within the systems context.

They define three critical stages of the evaluation phase: functionality, interoperability, and performance. In the functionality phase, the candidates are tested in isolation to confirm that the functionality of the software product is applicable to the current application. In the interoperability stage, the candidates are evaluated to ensure their ability to co-exist with other components of the system. The performance evaluation stage consists of a quantitative analysis of the effect of the software tool component on the overall performance of the system.

The final aspect of the methodology is a management evaluation that considers the less tangible aspects of integrating the software tool. These include such things as training, cost, vendor capability, etc. At the end of this process, a final selection of software tools is made.

Note that this methodology depends on having a relatively complete predefined set of requirements since the product identification stage is dependent on software tool candidates meeting the requirements. The methodology in general is a waterfall-style process in that each stage depends on the results of its predecessor.

1.3.1.3. OTSO method: Kontio *et al.* [20] present a multi-phase approach to software tools selection. The phases are the search phase, the screening and evaluation phase and the analysis phase. The method provides specific techniques for defining evaluation criteria, comparing the costs and benefits of alternative products, and consolidating the evaluation results for decision-making. The definition of hierarchical evaluation criteria is the core task in this method, it identifies four different sub-processes: search criteria, definition of the baseline, detailed evaluation criteria definition and weighting of criteria.

The evaluations are always performed against a set of evaluation criteria which are established from a number of sources, including the requirements specification, the high level design specification, the project plan, etc. The final phase of the selection process is the analysis of the results of the evaluation. This leads to the final selection if software tool products for inclusion in the system.

Even though OTSO realizes that the key problem in software tool selection is the lack of attention to requirements, the method does not provide or suggest any specific solution. A central repository for characteristics would be one of possible solutions. The method assumes that the requirements already exist since it is based on the requirements specification for defining the evaluation criteria.

1.3.1.4. STACE framework: *STACE* [21] is an approach that emphasizes social and organizational issues related to software tool selection. The main limitation of this approach is the lack of a process of requirements gathering and specification. Moreover, the STACE does not provide an analysis of the evaluated SW tools using a decision-making technique.

1.3.1.5. CDSEM: Jeanrenaud and Romanazzi [22] present a methodology for evaluating software that employs checklists, which they use to determine a quality metric for each item in the checklist. The process is metric-based and provides a numerical result that describes the suitability of the component. This approach is very attractive because it quantifies the evaluation results. However, the authors base some of their discussion on the availability of source code and access to individual modules, neither of which are usually available in a software tool.

They also depend heavily on the vendor documentation and demonstrations for supporting data as opposed to in-context, practical evaluation. This may lead to the adoption of unsuitable candidates.

1.3.2. Discussion

In this section, we concentrate our discussion on the requirements of AWEM evaluation method that are not met in the surveyed evaluation methods presented above.

The main unique feature of AWEM evaluation is that we do not evaluate a tool or a method. We rather conduct an evaluation that simultaneously covers both. Thus, literature that deals merely with evaluating SW tools are not related and so is the literature dealing with the methodological aspects. We are evaluating a methodology thru its supporting tool. Based on this perspective, the main shortcomings of the software evaluation methods presented above can be summarized in the following points:

- They do not define a central repository meta-model for characteristics to keep the track of their evolution through out the evaluation life cycle. In general, these methods rely on the definition of pre-established and structured criteria based on fixed requirements. These approaches are not appropriate to handle with the impositions of a highly volatile and uncertain marketplace.
- A common approach found in all methods described above is the use of a single multi-criteria decision making (MCDM) technique. The most used approaches is the WSM (Weighted Scoring Method). The basic concepts of these approaches are establishing a list of criteria that products should meet, assigning scores to each criterion based on its relative importance in the decision and then ranking products based on their total scores. WSM technique has some limitations when applied in software assessment, for instance: (i) this approach produces real numbers as results, so they can easily be interpreted as the true differences between the alternatives rather than the relative ranking, and (ii) difficulty in assigning weights when the number of criteria is large.
- They mainly concentrate on technical aspects of the SW tool.
- Most of them assume a few and simply structured characteristics.
- Do not provide an evaluation scheme that allows for flexible customization of the targeted characteristics. This led to a very restricted use of the environment.
- Difficult to weigh and rate large numbers of criteria.
- Most of them supports quantitative-based evaluation except OTSO that slightly supports qualitative however without guidance.
- Use simple unguided WSM approach.
- Do not provide techniques to measure the consistency ratio of the evaluator performance.
- They do not support new evolving characteristics. The AWEM evaluation process must provide an open ended environment for defining new requirements. As with any modern system, the requirements evolve over time. Some of the proposed software tool evaluation methods have proven to be less than successful because they are based on traditional development paradigms. Many of these paradigms rely on a highly structured requirements definition and specification that sets the criteria for software tool selection. As such they are slow to react to the fast changing commercial marketplace.

Advantages of AWEM-ESP

In addition to avoiding the shortcomings and meeting the essential requirements outlined in the sections above, (not mentioned again in this section to avoid repetition) the AWEM-ESP is distinguished with a number of unique features.

- In the design and implementation of AWEM-ESP, we concentrated on having a customizable and extensible environment that is based on a highly accurate evaluation technique. Therefore, we have identified a class model for representing the generalization and specialization aspects of the characteristics based on the application domain. Each category inherits all characteristics from the category at the level above and contains common characteristics common to categories at the level below as illustrated in Fig. 1 below. As can be seen from the model, the lowest level contains characteristics pertaining to specific types of web applications such as E-commerce, E-government and E-learning.
- The AWEM-ESP has a number of well integrated and strict stages that can ensure a final coherent and consistent decision.
- AWEM-ESP is fully automated. The automation system, AWEM-ESS, is web-enabled providing a user-friendly interface as well as sophisticated graphical reports.

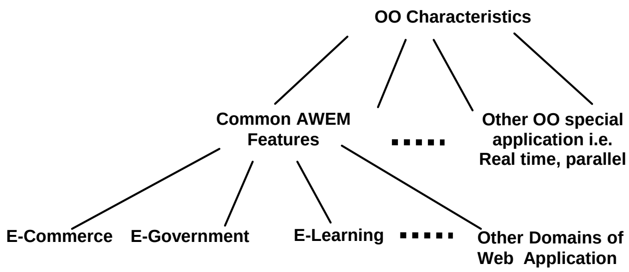

Fig. 1. Generalization and specialization of characteristics based on the application domain.

Consequently and based on the above outlined advantages, we can conclude that the AWEM evaluation and the selection process in general and the evaluation technique, presented here in specific, are both genuine and has not been identified and documented, nor made available for web engineers yet. The evaluation technique presented here has been designed in a way that avoids most of the critical failure factors of tools evaluation and selection process that have been discussed in [13]. One of the most important factors pointed out is the process itself, and more specifically the evaluation technique.

1.4. AWEM-ESP

The AWEM Evaluation and Selection Process, "AWEM-ESP", consists of four major stages. During the first stage "characteristics identification", the AWEM characteristics are captured, filtered, classified and categorized. The next stage is to screen academic and commercial sectors searching for AWEM environments that meet a list of preliminary requirements, e.g. UML-based, covering core SDLC stages, supported by tools, etc. AWEMs meeting such preliminary requirements will then be nominated as candidate AWEM environments. The nominated candidates are then passed to the next stage; the evaluation, in order to be assessed against a set of selected characteristics.

Actually, there isn't a best AWEM environment, which can serve all types of users for all purposes. The evaluation phase makes use of a so-called "evaluation scheme" which is a set of characteristics that serve specific types of users for specific domains. The idea is to make the evaluation process more user-centric, as each environment has its own agenda of priorities and there isn't a best AWEM environment, which can serve all of them. Thus, each evaluation reflects the organization's perspective towards the AWEM environment.

Finally is the selection stage, where the results of the evaluation phase are assessed and compared using an automated system built specifically for this purpose. Based on the comparison results, the selector may select the AWEM environment that was judged to be the best. The selected AWEM environment might need further customization efforts in order to meet the intended requirements perfectly. Finally, and after accomplishing all necessary customizations, the selected AWEM environment will be recommended and documented for further implementation.

2. Evaluation

The "Evaluation" stage comes after both "characteristics identification" and "AWEM environments screening" stages. During this stage, nominated AWEM environments are methodologically and technically assessed against the predefined characteristics. Prior to conducting the actual evaluation process, the "Evaluation Administrator" should create an "Evaluation Scheme", which is subsequently used to evaluate a specific AWEM environment against specific organizational needs for specific application domains. It is created by assigning predefined weights to the selected characteristics (from the main characteristics repository) in order to reflect their degree of importance with respect to the rest of the characteristics from the evaluation administrator's viewpoint. The importance degree is affected by two factors; the domain of the intended web application and the enterprise's requirements.

Furthermore, the "Evaluation" stage implies a development of a real pilot project. The pilot project must be selected very carefully so it can cover all significant characteristics identified by an "Evaluation Scheme". Once the evaluator is adequately familiar with the features of the AWEM under consideration, the evaluation scheme can be filled up with the final results of atomic characteristics.

The activity diagram illustrated in Fig. 2 below explains, in brief, the steps related to the evaluation. The swim-lanes represent areas where the activities of each role-player occur. The activities are represented with rounded rectangles, the artifacts with sharp ones.

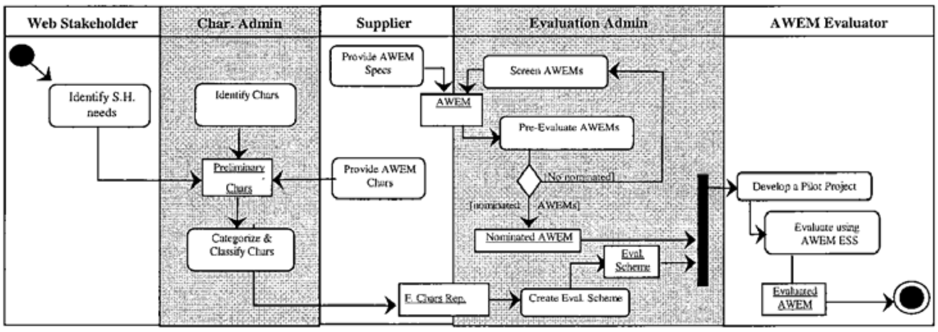

Fig. 2. Workflow for the activities performed during the evaluation stage.

2.1. Evaluation approaches

Two evaluation approaches have been identified and implemented in the process suggested here: quantitative and qualitative. Quantitative evaluation assumes that each AWEM can have some measurable characteristics that are expected to vary.

The quantitative evaluation is implemented for each characteristic through the use of a so-called "quality measure". This attribute guides the evaluator on how to assign a predefined “rate” to a specific characteristic. The rates assigned can take one of the predefined values, which are in descending order: *"Strong", "Sufficient", "Weak"* and "*Not Available*". The original quantitative value of the characteristic is still kept in the atomic characteristic for further traditional statistical analysis.

On the other hand, qualitative evaluation is based on identifying the user needs and mapping them to features that an AWEM aimed at supporting that need should possess. An evaluator then assesses how well the identified features are provided by a specific AWEM environment. Qualitative approach usually requires a subjective assessment of the relative importance of different features and how well a feature is implemented.

The qualitative approach is also implemented through the use of "quality measure" as a guideline based on which a “rate” is assigned to atomic-characteristic. Moreover, the qualitative measure is supported by capturing the user's feedback on certain characteristic using the atomic-characteristic's property "*Previous Tester's Comments*".

2.2. Ways for conducting the evaluation

In addition to the evaluation approaches introduced above, [10] has identified a very important dimension in the evaluation stage. That is the way in which the evaluation is organized. Three different ways of organizing an evaluation exercise have been identified. First, the evaluation may be conducted as a formal experiment where many subjects (i.e. software engineers) are asked to perform a task using the different methods/tools under investigation. A second way is the use of a case study where each method/tool under investigation is tried out on a real project using standard project development procedures. A third way is to conduct a survey where staff/organizations that have used specific methods or tools on past projects are asked to provide information about the method or tool.

The process suggested here has primarily considered the second option where a real pilot project is to be developed. The development process must pass through all the standard stages recommended by the AWEM under evaluation. The use of the first and third options is taken into consideration but as a supplementary source, in case part of the AWEM cannot be perfectly explored for some reason.

Finally, the AWEM environment will be examined against a variety of enterprise system development platforms in order to assess how sophisticated the evaluated AWEM environment is.

2.3. AWEM-ESP evaluation mechanism

As stated above, the evaluation technique proposed by the AWEM-ESP combines both quantitative with qualitative measures. The evaluation mechanism suggested has slight similarities with other techniques [3, 14, 16], with them all having almost the same eventual purpose which is weighing, rating and finally selecting the best alternative. Satty’s AHP is based on the use of massive number of tabular formats to capture information about the evaluation process, which we found less user friendly especially with the vast number of criteria identified. On the other side, Stanney suggested a very simple graphical model that cannot serve the purposes of qualitative intensive nature that cover significant part of the AWEM area. Consequently, the relatively high number of characteristics as well as their complex nature made the design of an easy-to-use and practical evaluation mechanism a real challenge.

The evaluation activity used in this process consists of two major steps: creating a so-called “evaluation scheme” and conducting the actual evaluation against a selected AWEM environment. The former is conducted by the “evaluation administrator”, while the latter is dealt with by the “evaluator”.

2.3.1. Creating “evaluation scheme”

The characteristics repository includes a massive number of atomic characteristics that have been collected from various application domains, classified and made available for the "evaluation administrators". The next task of the "evaluation administrator" is to narrow down these characteristics by aggregating them into "Evaluation Schemes". Creating an “evaluation scheme” involves the following steps that must be conducted consecutively:

1. Documenting the newly created “evaluation scheme” using a standard template provided by the AWEM-ESP that contains all necessary information which may facilitate future use of the scheme by prospective evaluators. Examples of information fields contained in the template include: evaluation scheme name, objectives, assumptions, category, characteristic weighing, e.g. "Not Related", "Normal", "Important", "Very Important" or "Highly Important".
2. Specifying the business domain of the intended scheme, i.e. E-commerce, E-learning and E-government. This will filter the characteristics structure and narrow it down to those characteristics related to the problem domain.
3. Selecting and deselecting characteristics at various levels of the characteristics model, i.e. super, sub-1, 2 and 3 and atomic. This step will further filter the characteristics model by trimming away those characteristics that are not related to the theme of the “evaluation scheme”.
4. Assigning a weight for each selected characteristic according to the extent to which the characteristic affects the intended “evaluation scheme”. The weight assignment goes top-down wise starting from the super characteristic, going through all the intermediate sub-characteristics and ending by the atomic ones as illustrated in the ellipses attached to the left of the characteristics in Fig. 3. If no weight is specified at a certain level then characteristics at that level will be assigned equal weights based on the value assigned to the parent. In the case of super characteristics, if no weight is assigned, then the default is the highest weight value, i.e. “Highly Important”. The weight value assigned to each characteristic should be selected according to the weighing scheme presented in Table 1 below. The verbal representation and values are flexible and fixed according to the “evaluation administrator’s” preferences.

Table 1. Characteristic weighing scheme

Weight ID	Value
Not Related	0
Normal	1

Important	3
Very Important	5
Highly Important	7

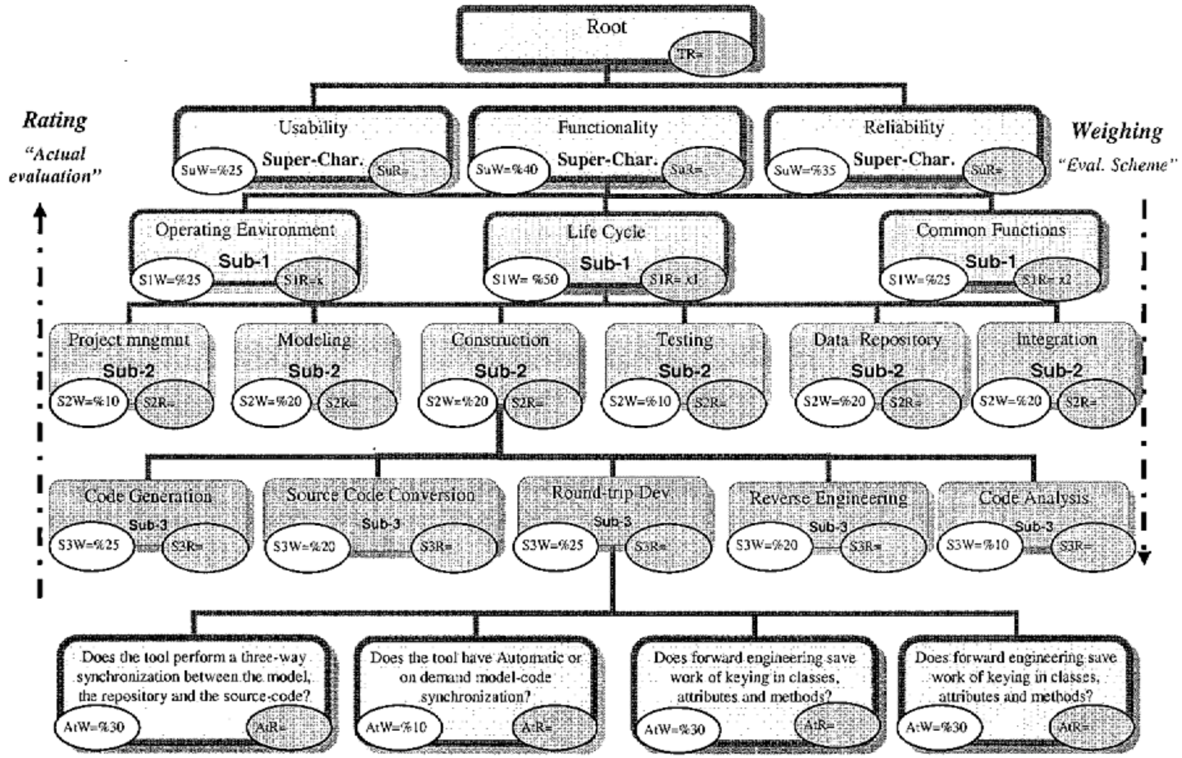

Fig. 3. Weight and rate assignments to characteristics at various levels.

2.3.2. Conducting the actual evaluation

Once the evaluation scheme is created, evaluators can start using it to evaluate and compare various AWEM environments based on the results of the evaluative pilot project. Conducting the actual evaluation consists of the following successive steps:

1. Selecting one of the previously approved AWEM environments.
2. Specifying the domain of the web application against which the intended AWEM environment is to be evaluated, i.e. E-commerce, E-learning and E-government, etc. This is in order to narrow down the search to those evaluation schemes related to the specified WE domain.
3. Selecting one of the previously created evaluation schemes to be the basis for the evaluation activities of the selected AWEM environment.
4. Using the "quality measure" guidelines as well as "previous user's comment" provided with each atomic characteristic, the "evaluator" can start assigning rate for each atomic characteristic. The rate assigned should reflect the degree of achievement by the AWEM environment against the specified characteristic, according to the rating scheme presented in Table 3. The total rate of the atomic-characteristic is calculated according to the series of equations listed below. The rate of the rest of the intermediate characteristics will be calculated starting from the atomic characteristic and going all the way up until the super characteristics. Finally, the root characteristic is assigned the final rate of the whole AWEM environment under evaluation as illustrated in Fig. 3 where rates are shown as gray ellipses attached to the characteristic rectangles.
5. Along with the quantitative rates used in the previous step, the "evaluator" may also add textual evaluative information to the atomic characteristic that elaborates more on how perfectly the AWEM environment achieved in this.

Table 2. Characteristic rating scheme

Rate ID	Value
Not applied	0
Weak	1
Sufficient	3
Strong	5

After giving each atomic-characteristic its own rate, the total score of the AWEM environment is calculated according to the series of equations presented in the following section.

2.3.3. Evaluation algorithm

The "evaluation administrator" manually assigns a weight "AtW" to each atomic-characteristic, based on its importance, by selecting one of the predefined values laid out in the weighing scheme (Table 1). The weight's corresponding value is then divided by the sum of all weight values given to the rest of the atomic-characteristics under the same sub-3 characteristic "TAtW(s3)" in order to calculate its respective average weight.

Based on the achievement of the AWEM environment, the “evaluator” then assigns a respective rate “AtGR” for each atomic-characteristic according the “Rating scheme” in Table 2 above. The rate’s corresponding value is then divided by the maximum rate value “MaxAtR”, i.e. “Strong”, in order to calculate its respective average rate. The average weight is then multiplied by the average rate to come up with the final rate of the atomic-characteristic “AtR” as shown in Eq. (1) below.

$$\rightarrow \textbf{AtR(at) = (AtW(at) / TAtW(s3)) * (AtGR(at)/MaxAtR)} \qquad (1)$$

where
AtR(at): The average rate of the atomic characteristic “at”.
AtW(at): Weight given to the atomic characteristic “at”.
TAtW(s3): Total weight of all atomic characteristics under the sub-3 characteristic “s3”.
AtGR(at): The rate given to the atomic characteristic “at”.
MaxAtR: The maximum rate value, i.e. “Strong”.

$$\rightarrow \textbf{TAtR(s3)} = \sum_{i=1}^{NOAt(s3)} AtRi \qquad (2)$$

where
TAtR(s3): Total rate of all atomic-characteristics under the sub-3 characteristic “a3”.
NOAt(s3): Number of atomic characteristics included in the sub-3 characteristic “a3”.
AtRi: The average rate of atomic-characteristic No. (i) under the sub-3 characteristic “s3”.

Then the rate of each sub-3 characteristic “S3R” is worked out. This is done by multiplying the total rate of all atomic-characteristics “TAtR” belonging to the sub-3 characteristic by the average weight of the sub-3 characteristic (S3W(s3) / TS3W(s2)) as presented in Eqs. (2) and (3).

$$\rightarrow \textbf{S3R(s3) = (S3W(s3) / TS3W(s2)) * (TAtR(s3))} \qquad (3)$$

where
S3R(s3): The average calculated rate for the sub-3 Characteristic “s3”.
S3W(s3): Weight assigned to the sub-3 Characteristic “s3”.
TS3W(s2): Total of weights assigned to all sub-3 characteristics under the same sub-2 characteristic “s2”.
TAtR(s3): Total rate of all atomic-characteristics under the sub-3 characteristic “a3”.

$$\rightarrow \textbf{TS3R(s2)} = \sum_{i=1}^{NOS3(s2)} S3R(i) \tag{4}$$

where

TS3R(s2): Total rate of all sub-3 Characteristics under the sub-2 characteristic “s2”.
NOS3(s2): Number of sub-3 characteristics under the sub-2 Characteristic “s2”.
S3Ri: The rate of the sub-3 characteristic No. “i” under the sub-2 Characteristic “s2”.

The next step towards calculating the total score is to calculate the average rate for each sub-2 characteristic S2R. To do this, the total rate of all sub-3 characteristics “TS3R” belonging to the same sub-2 characteristic is multiplied by the average weight of the sub-2 characteristic (S2W(s2) / TS2W(s1)) as presented in Eqs. (4) and (5).

$$\rightarrow \textbf{S2R(s2) = (S2W(s2) / TS2W(s1)) * (TS3R(s2))} \tag{5}$$

where

S2R(s2): The average calculated rate for the sub-2 characteristic “s2”.
SW(s2): Weight assigned to the sub-2 characteristic “s2”
TS2W(s1): Total of weights assigned to all sub-2 characteristics under the sub-1 characteristic “s1”.
TS3R(s2): Total rate of all sub-3 characteristics under the sub-2 characteristic “s2”.

$$\rightarrow \textbf{TS2R(s1)} = \sum_{i=1}^{NOS2(s1)} S2R(i) \tag{6}$$

where

TS2R(s1): Total rate of all sub-2 Characteristics under the sub-1 characteristic “s1”.
NOS2(s1): Number of sub-2 characteristics under the sub-1 Characteristic “s1”.
S2Ri: The rate of the sub-2 characteristic No. “i” under the sub-1 Characteristic“s1”.

Next is to calculate the average rate for each sub-1 characteristic S1R. To do this, the total rate of all sub-2 characteristics “TS2R” belonging to the same sub-1 characteristic is multiplied by the average weight of the sub-1 characteristic (S1W(s1) / TS1W(su)) as presented in Eqs. (6) and (7).

$$\rightarrow \textbf{S1R(s1) = (S1W(s1) / TS1W(su)) * (TS2R(s1))} \tag{7}$$

where

S1R(s1): The average calculated rate for the sub-1 characteristic “s1”.
S1W(s1): Weight assigned to the sub-1 Characteristic “s1”
TS1W(su): Total of weights assigned to all sub-1 characteristics under the super characteristic “su”.

TS2R(s1): Total rate of all sub-2 Characteristics under the for sub-1 characteristic "s1".

$$\rightarrow \textbf{TS1R(su)} = \sum_{i=1}^{NOS1(su)} S1R(i) \tag{8}$$

where
TS1R(su): Total rate of all sub-1 characteristics under the super characteristic "su".
NOS1(su): Number of sub-1 characteristics under the super characteristic "su".
S1Ri: The rate of the sub-1 characteristic No. "i" under the super characteristic "su".

The last step before determining the final AWEM rate is to work out the average rate for each super characteristic "SuR". This is obtained by multiplying the total rate of all sub-1 characteristics "TS2R" belonging to the same super characteristic by the average weight of the super characteristic (SuW(su) / TSuW) as shown in Eqs. (8) and (9).

$$\rightarrow \textbf{SuR(su) = (SuW(su) / TSuW) * (TS1R(su))} \tag{9}$$

where
SuR(su): The average calculated rate for the super characteristic "su".
SuW(su): Weight assigned to the super characteristic "su".
TSuW: Total of weights assigned to all super characteristics.
TS1R(su): Total rate of all sub-1 characteristics under the super characteristic "su".

The final rate of the whole AWEM environment under evaluation is then obtained by summing up the calculated rates of all selected super-characteristics using Eq. (10).

$$\rightarrow \textbf{TR(e)} = \sum_{i=1}^{NOSu} SuRi \tag{10}$$

where
TR(e): Total rate of the AWEM Environment "e".
NOSu: Total number of super characteristics selected in the evaluation scheme.
SuRi: The calculated rate value assigned to the super characteristic number "i".

2.4. AHP for super characteristics

As discussed above, AHP was a good choice for conducting evaluation on few numbers of criteria. Thus, we decided to incorporate AHP to assign weights to characteristics at the super level to fine tune the final results. Due to the space limitation, a concise example of a software development company is included here.

The company is in the process of selecting a sufficient AWEM environment for conducting their Web IS development projects. For simplicity, we have chosen a sub-tree that consists of a single sub-2 characteristic construction. This consists of three sub-3 characteristics (code generation CG, round-trip dev RT. and reverse engineering RE.).

The pair-wise comparisons for the three sub-3 characteristics out of the sub-2 characteristic, construction, are shown in Table 3. The last column of Table 3 shows the weights of these characteristics that are determined by the AHP. The weights determined in this table are the eigenvector associated with the maximum eigenvalue of the pair-wise comparison matrix.

Table 3. AHP pair-wise comparisons for the three sub-3 characteristics

	CG	RT	RE		CG	RT	RE.	Aver
CG	1	1/	5		0.313	0.111	0.789	**0.40**
RT.	2	1	1/		0.625	0.222	0.053	**0.30**
RE	1/5	3	1		0.063	0.667	0.158	**0.29**

3. Findings & Recommendations

Throughout the theoretical and technical work on this empirical process, we can point out some of our findings and recommendations as follows:

1. Introducing the concept of "evaluation scheme" made the process more customizable to many web application domains.
2. The evaluation technique has been designed in a way that makes it easier for both the evaluation administrator to create evaluation scheme and for the evaluator to conduct the actual evaluation by assigning rates to various characteristics. This simplicity also helped in building an interactive and friendly user interface that facilitated conducting the evaluation task that is always complicated.
3. The process has been fully automated by a web enabled application called AWEM-ESS which allows remote actors to participate in testing the viability of the process [17].
4. The process has been tested using a real project to compare two AWEM environments and the result was more than satisfactory.

5. The use of a combination of two well known evaluation techniques, i.e. WSM and AHP helped to have a much reflective, non-subjective and realistic results.

4. Conclusion

The evaluation mechanism and algorithm presented in this paper represent a very crucial part of the AWEM-ESP evaluation stage. Amongst four other major stages, the evaluation stage comes into view after the two core stages, "Characteristics Identification" and "Screening Available AWEM Environments" and before the "Selection" stage. Prior to conducting the actual evaluation task, the "*Evaluation Administrator*" should create an "*Evaluation Scheme*", which includes a set of characteristics for evaluating a specific AWEM environment against specific organizational needs for specific application domains. Then the actual evaluation task starts, which implies the development of real pilot projects. Finally, the evaluation results are fed into an automated system that analyzes the results and presents them in a form usable for both evaluators and selectors.

The evaluation mechanism and algorithm have been implemented and tested using a AWEM-ESS which is an automation package built specifically for this purpose. Moreover, the AWEM-ESS has been used for evaluating two AWEM environments, namely IBM Rational XDE for java and .Net against two evaluation scheme, namely, "Small E-commerce Project" and "E-government Project". The evaluation process went very perfectly and the results were more than satisfactory.

References

[1] Alghamdi, A. Common Characteristics Required for Automated Web Engineering Environments". *iiWAS, The 7th International Conference on Information Integration and Web-based Applications & Services*, 19-21 September 2005, Kuala Lumpur, Malaysia.

[2] Annette, L. *Techniques: A Method for CASE tool Evaluation, Information & Management 25*. North-Holland: 1993.

[3] Bodin, L. D.; Gordon, L. A. and Loeb, M. P. "Evaluating Information Security Investments Using the Analytic Hierarchy Process". *Communications of the ACM*, 48, No. 4 (2005), 78-83.

[4] Daneva, M. "A Best Practice Based Approach to CASE-tool Selection Software Engineering Standards". *Proceedings of the Fourth IEEE International Symposium and Forum*, 17-21 May 1999, pp. 100–110.

[5] Hennicker, R. and Koch, N. "Modeling the User Interface of Web Applications with UML". *Workshop of the UML-Group at the UML*, 2001.

[6] Hennicker, R.; Koch, N. and Kraus, A. "The Authoring Process of the UML-based Web Engineering Approach". *1st Workshop on Web-oriented Software Technology WWOST'01*, Valencia, 2001.

[7] IEEE. *IEEE Recommended Practice for the Evaluation and Selection of CASE Tools*. IEEE Std.1209-1992.

[8] IEEE. *Adoption of International Standard lSO/lEC 14102: 1995) Information Technology: Guideline for the Evaluation and Selection of CASE Tools*. IEEE Std 1462-1998

[9] ISO/IEC. *Information Technology, Guideline for the Evaluation and Selection of CASE Tools*. ISO/IEC 14102 First edition 1995-11-15.
[10] Kitchenham, B. "Evaluating Software Engineering Methods and Tool, Part 1: The Evaluation Context and Evaluation Methods". *Software Engineering Notes*, 21, No. 1 (1996), 11-15.
[11] Knapp, A.; Koch, N.; Moser, F. and Zhang, G. "Argo UWE: A Case Tool for Web Applications". *First Int. Workshop on Engineering Methods to Support Information Systems Evolution (EMSISE´03)*, Sept. 2003.
[12] Koch, N. "Software Engineering for Adaptive Hypermedia Systems". *PhD Thesis, Ludwig Maximilians University,* Munich, 2001.
[13] Prather, B. "Critical Failure Points of CASE Tool Evaluation and Selection". *Computer-aided Software Engineering, 1993 (CASE '93), Proceeding of the Sixth International Workshop*, 19-23 July 1993, pp. 60-63.
[14] Saaty, T. L. *Fundamentals of Decision Making and Priority Theory with the Analytic Hierarchy Process*. Vol. 6, Pittsburgh, PA: RWS Publications, 2000.
[15] Schneider, K. "A Systematic Approach to CASE Selection; Assessment of Quality Software Development Tools". *Proceedings of the Second Symposium*, 27-29 May 1992, pp. 42-54.
[16] Stanney, M.; Mollaghasemia, M.; Reevesa, L.; Breauxb, R. and Graeberc, D. "Usability Engineering of Virtual Environments (VEs): Identifying Multiple Criteria that Drive Effective VE System Design". *International Journal of Human-computer Studies*, 58, No. 4 (April 2003), 447-481.
[17] URL1: http://www.mcrlab.uottawa.ca/WebEng
[18] Ncube, C. and Maiden, N. A. M. "PORE: Procurement-oriented Requirements Engineering Method for the Component-based Systems Engineering Development Paradigm". *International Workshop on Component-based SE*, May 2002.
[19] Vu, Tran and Dar-Biau, Liu. "A Risk-mitigating Model for the Development of Reliable and Maintainable Large-scale Commercial-off-the-shelf Integrated Software Systems". *Proceedings of the 2001 Annual Reliability and Maintainability Symposium*.
[20] Kontio, J. "A COTS Selection Method and Experiences of Its Use". *Proceedings of the 20th Annual Software Engineering Workshop*, Maryland, November 1995.
[21] Kunda, D. and Brooks, L. "Applying Social-technical Approach for COTS Selection". *Proceedings of the 4th UKAIS Conference*, University of York, April 2003.
[22] Jeanrenaud, J. and Romanazzi, P. "Software Product Evaluation: A Methodological Approach". *Software Quality Management II: Building Software into Quality*, 2001, pp. 59-69.

عملية ربيبة لتقييم واختيار بيئة نهجيات هندسة بر يات الويب "مرحلة التقييم"

عبدا   الغامدي

قسم نظم المعلومات، كلية علوم الحاسب والمعلومات،

جامعة الملك سعود، ص.ب ٨٧٨٠٦، الرياض ١١٦٥٢،

البريد الإلكتروني: *ghamdi@ccis.ksu.edu.sa*

(قدّم للنشر في ١٣/٠٦/٢٠٠٦م؛ وقبل للنشر في ١١/٠٣/٢٠٠٧م)

ملخص البحث. تعرض هذه الورقة مرحلة التقييم من عملية تجريبية تهدف إلى تقييم واختبار أفضل بيئة لمنهجية هندسة برمجيات ويب بطريقة آلية. تتألف العملية من أربع مراحل رئيسة وهي كالتالي: تحديد السمات، ومسح بيئات هندسة البرمجيات المتواجدة، والتقييم، والاختبار. خلال مرحلة التقييم يتم إنشاء "مخطط تقييم" والذي يخدم نطاق محدد لتطوير أنظمة الويب والذي يعكس زاوية اهتمام المنظمة تجاه بيئات هندسة برمجيات AWEM. والفائدة الرئيسة من ذلك هو جعل عملية التقييم معتمدة على المستخدم. يتم إجراء التقييم الحقيقي بعد ذلك بناءً على النتائج المستقاة من خلال تطوير مشاريع تجريبية حقيقية.

تساهم هذه الورقة في إثراء الجهود البحثية في مجال هندسة الويب من خلال اقتراح آلية تقييم تستخدم فكرة "مخطط التقييم" وهو عبارة عن مجموعة جزئية للمعايير العامة للتقييم. بالإضافة لذلك هناك مساهمة أخرى مهمة للبحث تتمثل في تقديم خوارزميات لتقييم عدة سمات وبدائل والتي ستساهم في نهاية المطاف في عملية اتخاذ القرار. لقد تم تطوير نظام لدعم العملية والخوارزمية يسمى AWEM ESS وتم كذلك تقييمه من خلال استخدامه في حالات واقعية وكانت النتائج متميزة.