Electronic Coursework Assessment and Feedback

Shun Ha Sylvia Wong, Alan J. Sellers and Anthony J. Beaumont
Computer Science, Aston University, Aston Triangle, Birmingham B4 7ET, U.K.
{s.h.s.wong, selleraj, a.j.beaumont}@aston.ac.uk

Abstract: Recent surveys reveal that many university students in the U.K. are not satisfied with the timeliness and usefulness of the feedback given by their tutors. Ensuring timeliness in marking can result in a reduction in the quality of feedback. Though suitable use of Information and Communication Technology should alleviate this problem, existing Virtual Learning Environments are inadequate to support detailed marking scheme creation and they provide little support for giving detailed feedback. This paper describes a unique new web-based tool called e-CAF for facilitating coursework assessment and feedback management directed by marking schemes. Using e-CAF, tutors can create or reuse detailed marking schemes efficiently without sacrificing the accuracy or thoroughness in marking. The flexibility in marking scheme design also makes it possible for tutors to modify a marking scheme during the marking process without having to reassess the students' submissions. The resulting marking process will become more transparent to students.

Introduction

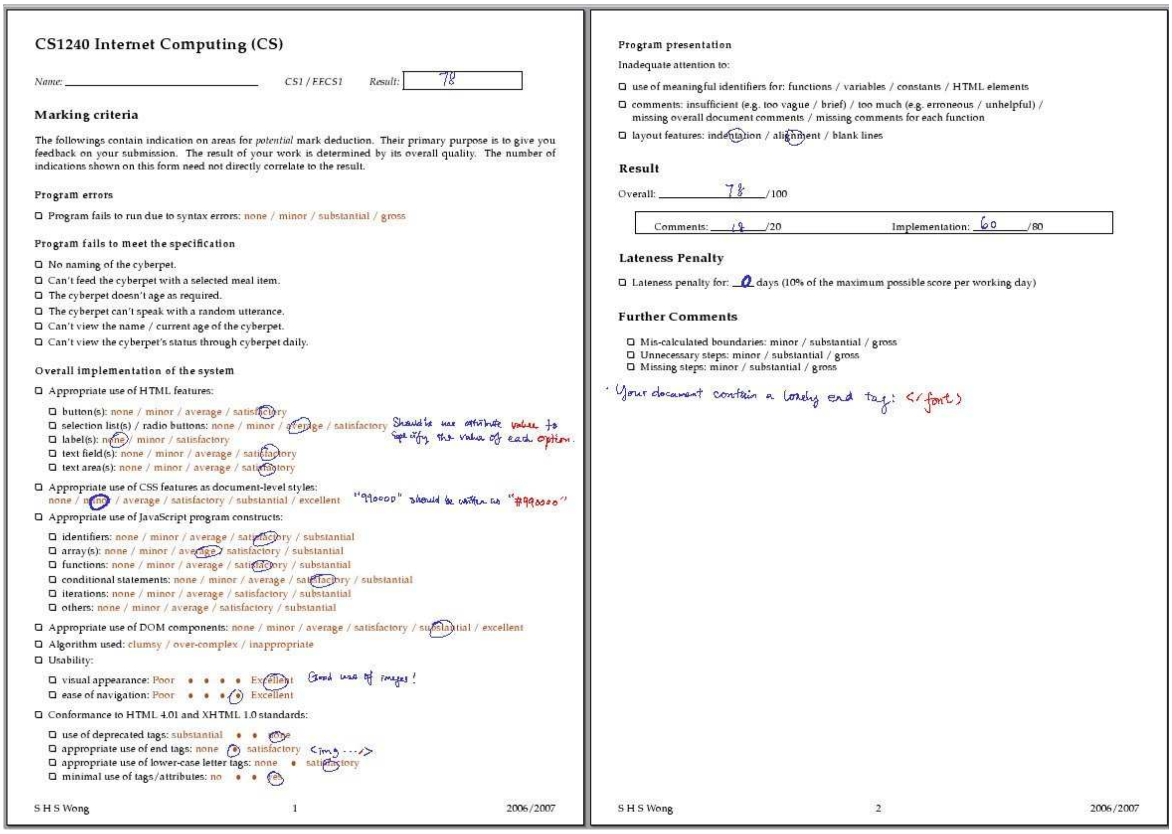
According to the 2006 National Student Survey (NSS), roughly 40% of UK university students are not satisfied with the feedback given by their tutors (Surridge, 2007, p. 20): 50% of them did not think that the feedback on their work had been prompt and 49% of them did not agree that the feedback on their work helped them clarify things that they did not understand. From students' point of view, lengthy turnaround time and insufficient feedback are generally considered to be the two major causes of concern in assessment feedback. Higher Education in the U.K. traditionally views assessment as a means of classifying the performance of students into categories. This view implies that emphasis is placed on the ability to produce an end of year grade for each student, rather than giving timely and detailed feedback on the student's work. In addition to ranking students' performance, we view assessment, especially coursework assessment, as a means of promoting students' learning and stimulating inquiring minds. This is in line with the UK national policy for assessment and testing (DES, 1988). However, as pointed out by Black (1995), that policy has not been adhered to in practice. Indeed, the results from the latest available NSS results (Surridge, 2007) also confirm this observation. Giving timely and detailed feedback to coursework is a laborious task. The increase in class size in Higher Education and the lack of effective electronic means to support the assessment process make it even more difficult for tutors to provide timely feedback.

In terms of student satisfaction in assessment and feedback, the NSS figures further revealed that Engineering, Science and Technology are amongst the worse performing disciplines, with the figure for Computer Science (CS) in the lower quartile (Surridge, 2007, p. 32). CS is a technical subject whose teaching ethos relies heavily on *learning by doing* (Ylijoki, 2000). Therefore, a substantial amount of individual coursework is required to adequately address the teaching and learning needs in CS. With an average class size of over 100, the marking load for a CS tutor can be very high. Suitable use of Information and Communication Technology (ICT) can help tutors to maintain consistency in marking by reducing the cognitive load. It can also enable tutors to produce more timely feedback by taking away the repetitious, tedious and mechanical tasks from them. For example, ICT can facilitate a flexible marking scheme design and management, better organisation of feedback banks and flexible annotation of submitted documents. ICT can also improve the accessibility of feedback for students and provide a better archiving facility for students' work. However, our experience in using conventional Virtual Learning Environments (VLEs) such as WebCT and Blackboard (Blackboard Inc., 2007b, 2007a) reveals that they do not adequately support our coursework assessment and feedback practise. Other e-assessment packages such as *OpenMark* (Ekins, 2006) supports instant assessment and release of relevant study advice. However, it is tailored for assessing online quizzes rather than programming coursework, reports or essays. To reduce this gap, we are developing a new web application to facilitate coursework marking and giving detailed formative feedback to students' work.

In this paper, we will give an overview of our new coursework marking and feedback management system *e-CAF*. This paper will focus on presenting the design and implementation of the flexible marking scheme features of e-CAF. It will conclude with the planned future work and our vision for e-CAF. First, we will expound on the motivation behind e-CAF by reviewing related work within computer science teaching.

Motivation and Related Work

Assessing programming coursework is a time-consuming task. While contemporary automatic computer program assessment systems such as *BOSS* (Joy, Griffiths, & Boyatt, 2005) and *analyse* (Zin & Foxley, 2007) can speed up the assessment process, they are designed for assessing program correctness automatically, rather than for giving detailed formative feedback to students. To promote learning, it is important to provide students with timely feedback that is clearly linked to the tasks and their performance (Black & Wiliam, 1998). Formative feedback that points out mistakes, corrects misconceptions and demonstrates good practise is particularly beneficial to learning practical skills such as programming. Existing automatic program assessment systems cannot support such a level of feedback. Hence, the involvement of human markers in the assessment process is paramount.


CS1240 Internet Computing (CS)

Name: ______________________ CS1 / EECS1 Result: 78

Marking criteria

The followings contain indication on areas for *potential* mark deduction. Their primary purpose is to give you feedback on your submission. The result of your work is determined by its overall quality. The number of indications shown on this form need not directly correlate to the result.

Program errors

- Program fails to run due to syntax errors: none / minor / substantial / gross

Program fails to meet the specification

- No naming of the cyberpet.
- Can't feed the cyberpet with a selected meal item.
- The cyberpet doesn't age as required.
- The cyberpet can't speak with a random utterance.
- Can't view the name / current age of the cyberpet.
- Can't view the cyberpet's status through cyberpet daily.

Overall implementation of the system

- Appropriate use of HTML features:
  - button(s): none / minor / average / satisfactory
  - selection list(s) / radio buttons: none / minor / average / satisfactory — Should use attribute value to specify the value of each option.
  - label(s): none / minor / satisfactory
  - text field(s): none / minor / average / satisfactory
  - text area(s): none / minor / average / satisfactory
- Appropriate use of CSS features as document-level styles: none / minor / average / satisfactory / substantial / excellent — "990000" should be written as "#990000"
- Appropriate use of JavaScript program constructs:
  - identifiers: none / minor / average / satisfactory / substantial
  - array(s): none / minor / average / satisfactory / substantial
  - functions: none / minor / average / satisfactory / substantial
  - conditional statements: none / minor / average / satisfactory / substantial
  - iterations: none / minor / average / satisfactory / substantial
  - others: none / minor / average / satisfactory / substantial
- Appropriate use of DOM components: none / minor / average / satisfactory / substantial / excellent
- Algorithm used: clumsy / over-complex / inappropriate
- Usability:
  - visual appearance: Poor • • • • Excellent — Good use of images!
  - ease of navigation: Poor • • • • Excellent
- Conformance to HTML 4.01 and XHTML 1.0 standards:
  - use of deprecated tags: substantial • • none
  - appropriate use of end tags: none • satisfactory
  - appropriate use of lower-case letter tags: none • satisfactory
  - minimal use of tags/attributes: no • • yes

S H S Wong 1 2006/2007

Program presentation

Inadequate attention to:

- use of meaningful identifiers for: functions / variables / constants / HTML elements
- comments: insufficient (e.g. too vague / brief) / too much (e.g. erroneous / unhelpful) / missing overall document comments / missing comments for each function
- layout features: indentation / alignment / blank lines

Result

Overall: 78 /100

Comments: 18 /20 Implementation: 60 /80

Lateness Penalty

- Lateness penalty for: 0 days (10% of the maximum possible score per working day)

Further Comments

- Mis-calculated boundaries: minor / substantial / gross
- Unnecessary steps: minor / substantial / gross
- Missing steps: minor / substantial / gross

· Your document contain a lonely end tag:

S H S Wong 2 2006/2007

Figure 1: Sample feedback form with marking criteria.

When assessing large numbers of students' work, Brown (2001) suggests delegating some of the marking to other tutors or mentors. One way to ensure consistency and efficiency in marking is to use a tailor-made detailed marking scheme for feedback and assessment. Such marking schemes may include a hierarchy of marking criteria. Each marking criterion can be further divided into a set of marking criteria or associated with a precise scale. For example, a criterion such as "*Appropriate use of HTML features...*" can be further divided into criteria such as "*button(s)*", "*label(s)*", etc. The criterion "*button(s)*" can be marked on a scale of four: "*none*", "*minor*", "*average*" and "*satisfactory*", with a mark value associated to each item (Fig. 1). When assessing a coursework submission, a tutor can simply select the appropriate descriptor for each marking criterion that best describes the work undertaken. In a marking scheme, two (sub-)sets of marking criteria can also be mutually exclusive, i.e. only one set can be applied in marking a coursework submission. The overall mark for each submitted piece of coursework will then be calculated from the score for all applicable marking criteria.

To facilitate such a complex and technical assessment and feedback process, a suitable computing tool is needed. Contemporary VLEs such as WebCT or Moodle (Cole, 2005) are inadequate to facilitate such a marking process. Typically, only a single final mark and an overall, general comment can be given to a coursework submission. *MarkTool* (Heinrich & Wang, 2003; Heinrich & Lawn, 2004; Zhang & Heinrich, 2005) was designed to combat this inadequacy. It provides a desktop solution for creating marking schemes and using them for onscreen formative assessment of essay-type assignments. The assessment results can then be uploaded to WebCT for returning the marked assignments and feedback to students (Heinrich, 2007).

While MarkTool provides good onscreen support to assessment and feedback, it does not provide sufficient support for creating an elaborate marking scheme such as the one described above (Fig. 1). Moreover, MarkTool is a desktop software application rather than a web-based application. This raises a portability issue. Markers are expected to install MarkTool on their computers and work offline until the assessment results are ready to be uploaded to WebCT. If more than one computer is used in the marking process, a marker will need to ensure that MarkTool is installed on each computer and the up-to-date assessment data is also kept in each computer. For markers who need to travel frequently, without their personal computer, MarkTool will become inaccessible.

MarkTool does not support modification or refinement of the marking scheme during the marking process. Hence, any mistake in the marking scheme identified during the marking process cannot be rectified. Even if a later version of MarkTool were to incorporate such a marking scheme modification feature, tutors involved in the group marking will need to be notified of the change and are required to take the initiative to update the marking scheme accordingly because MarkTool is a desktop application,

In the light of the above issues, we are developing a web-based software for online coursework assessment and feedback management known as e-CAF. Unlike WebCT, Moodle or MarkTool, e-CAF is equipped with a sophisticated, yet intuitive, marking scheme design and creation feature. Our main focus is to enhance students' learning experience by facilitating a better assessment and feedback process. e-CAF aims at making it significantly easier for tutors to:

1. assess large classes more consistently and transparently using detailed marking schemes;
2. include structured personal formative feedback, which is clearly linked with specific points within the submitted documents;
3. reduce the turnaround time of feedback to learners so that its effect is maximised.

e-CAF also aims at enabling the archiving of assignments and feedback for internal or government inspection and quality assurance purposes.

The e-CAF System: an Overview

One of the reasons for software developers to avoid using web-based solutions is that web-based software applications had a reputation for having an inflexible user interface and being inefficient. Recent advances in Internet Computing technologies such as Java Platform, Enterprise Edition (J2EE) (Sun Microsystems, Inc., 2007), AJAX (Garrett, 2005) and Google Web Toolkit (GWT) (Google, 2007) have alleviated these problems.

Our e-CAF system adopts a web-based solution to coursework assessment and feedback management. The development began in November 2006 and is still on-going. We have adopted an object-oriented approach to the design of e-CAF. The key data in e-CAF are coursework submissions, marking schemes and summative and formative feedback to each coursework submission. The implementation technologies used include J2EE version 5, AJAX and GWT version 1.4. e-CAF is backed by the SQL-style relational database Apache Derby which is part of the standard J2EE distribution. All assessment data, e.g. marking scheme details, coursework submissions, coursework assessment details and feedback, etc are persisted using the Java Persistence Architecture in J2EE. Hence, tutors can give detailed feedback to and assess coursework any time and anywhere so long as they have Internet access and they will not be troubled by the need to synchronise the data on their computers beforehand.

The e-CAF system allows tutors to:

- set elaborate hierarchical marking schemes with detailed guidelines for assessment and feedback,
- adapt and reuse existing marking criteria,

- assess coursework submissions based on set marking schemes,
- annotate coursework submissions with personalised feedback that ties in with specific points in the marking scheme,
- adapt and reuse feedback in the marking process, and
- view summary statistics on feedback given to students.

Within e-CAF, students can view detailed coursework feedback online. As soon as a tutor has completed the assessment process, students can access personalised feedback to their work any time and anywhere. The summary statistics feature also facilitates students to learn from other students' mistakes by enabling them to view a list of frequently-made mistakes (Fig. 2).

A guided marking process requires the display of relevant marking scheme details for tutor's reference. From the tutors' point of view, a good marking tool must support such a display instantly. Traditional interactive web-based applications are known to suffer from network traffic bottleneck. To prevent such a problem from occurring, e-CAF uses AJAX to ensure that data exchange between the server and the client computers are done asynchronously without affecting the perceived behaviour of the web page.

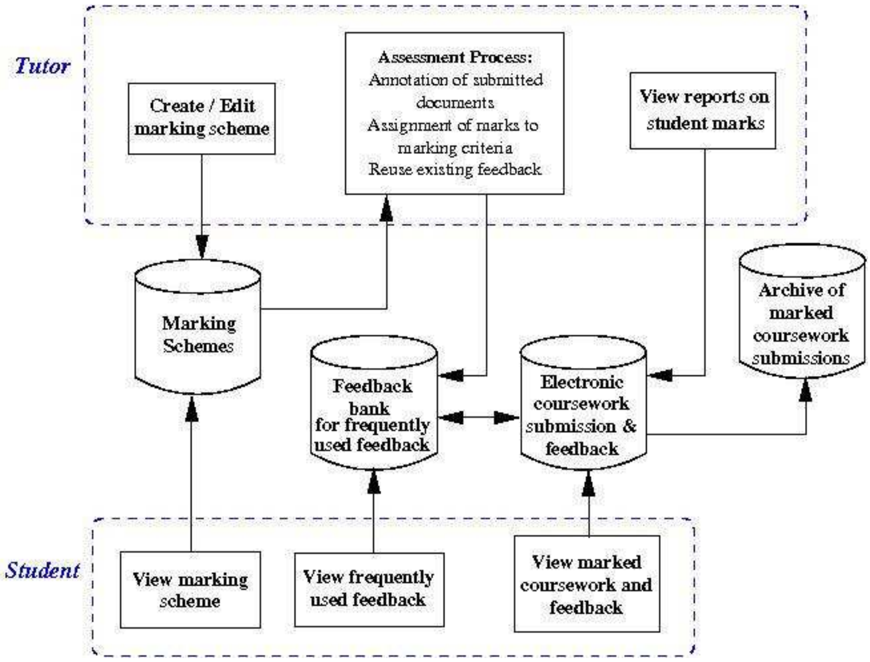


Figure 2: The e-CAF architecture.

Marking Scheme Design and Creation

Consistency in marking is very important because it affects the reliability of the marking process (Brown, 2001). It could be helped by a well-designed marking scheme that is as prescriptive as possible. In terms of efficiency, the smaller the number of criteria a marking scheme has, the better. However, with assignments that aim at assessing a number of learning outcomes and various practical skills, the marking schemes are likely to be complex.

It is generally regarded as a good practise to devise and even publish a marking scheme together with the coursework specification. Nevertheless, it is usually difficult to design a well-balanced detailed marking scheme before seeing students' solutions. For example, the set of marking criteria designed may not assess the students' work suitably, e.g. too specific, too harsh, or too lenient. The details of the marking scheme can often be worked out from a good understanding of the space of potential answers, of which the students' solutions give a good example. To facilitate a fair and consistent coursework assessment and feedback process, we need a tool that enables the design and creation of marking schemes flexibly.

The e-CAF system has been designed with flexibility in mind. Using e-CAF, tutors can add new marking criteria to, or alter the mark scale of a marking criterion in, a marking scheme during the marking process without having to manually recalculate the mark distribution for all marking criteria. In some cases, consistency in marking will be maintained even when a tutor alters a marking scheme after having marked a subset of coursework submissions. Hence, no remarking of those submissions will be required. e-CAF also ensures that all tutors involved in the group marking will use the most up-to-date version of the marking scheme to guide their marking process.

Hierarchical Marking Scheme

A structured marking scheme helps markers to focus on assessing and giving feedback to different aspects of the learning outcomes systematically. Our brief survey of marking scheme design revealed that tutors tend to structure the criteria in their marking scheme hierarchically. Typically, a marking scheme may contain two to three sections, each containing several marking criteria, with a possibility for each marking criterion to be further divided into sub-criteria (Fig. 1). To cater for the potential diversity in students' work, a marking scheme may contain several mutually exclusive sections. e-CAF ensures that a subset of sections is used to assess each submission. For example, with a programming coursework in which the syntactic correctness of the submission is paramount, different sets of marking criteria may be applied to assessing a compilable piece of code and a non-compilable piece of code.

The e-CAF system models a marking scheme hierarchically as a tree of marking criteria (Fig. 3). Each marking criterion can be subdivided into further marking criteria, hence enabling sectioning of marking criteria. e-CAF enables the inclusion of alternative marking criteria through special tree nodes named *AND* and *OR*. An AND node combines criteria to form a final mark and an OR node specifies mutually exclusive marking criteria (Fig. 4).

.

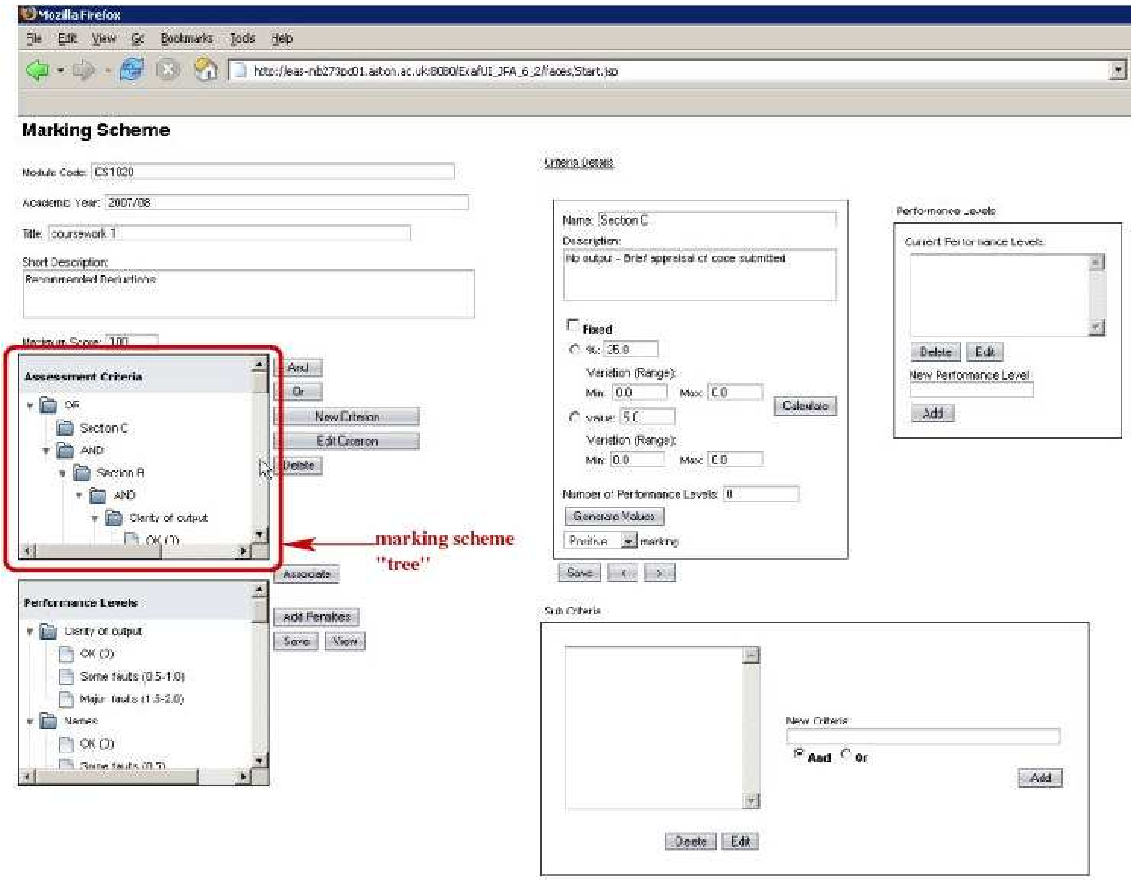


Figure 3: Sample screen of the e-CAF marking scheme creation feature.

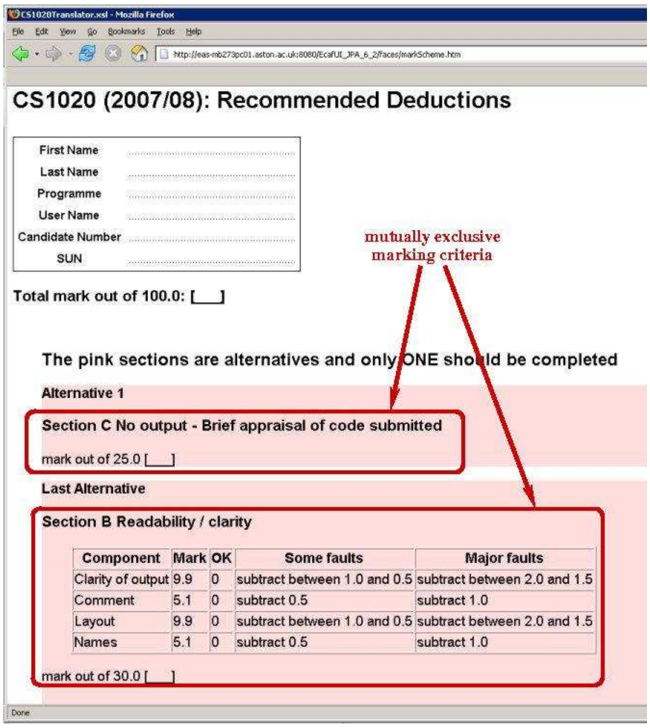


Figure 4: A sample marking scheme generated by e-CAF.

Tutors are encouraged to design a marking scheme in a top-down manner by identifying the broad assessment criteria and refine each criterion to the required level of detail. The complete marking scheme is viewed as a "pie" and each criterion at the top level AND node is viewed as a slice. Further refinement of a marking criterion is viewed as a process of further dividing a slice.

Automatic marking scheme validation

When designing a marking scheme, the more complex a marking scheme is, the more time-consuming and tedious the design process will become. To streamline the assessment process, tutors are often advised to avoid using detailed marking criteria. Our experience shows that well-designed detailed marking criteria helps to maintain consistency and increases the speed of marking. The marking scheme creation feature in e-CAF enables tutors to simply specify a set of marking criteria and how they are to be used in conjunction with each other in the marking process. For each marking criterion, tutors may specify the mark contribution as a fixed or a percentage value. The e-CAF system is equipped with a sophisticated validation routine to ensure agreement between the specified mark contributions and the maximum possible score for a coursework. As new marking criteria are being created and/or existing ones being modified or removed, e-CAF ensures the validity of the marking scheme by automatically recalculating all the actual mark contribution values based on the values specified by the tutor.

Recall that e-CAF models the criteria in a marking scheme hierarchically as a tree. To facilitate the marking scheme design process, the validation routine is implemented using a "pie-slicing" approach. The maximum possible mark for a coursework is viewed as a "pie" and each criterion at the top level AND node receives one slice of that pie. Further refinement of a marking criterion is viewed as a process of further dividing the slice. The validation routine simply ensures that all marking criteria beneath each AND node has their mark contributions summed to 100% and the actual mark contribution of a criterion at the bottom of the hierarchy is relative to that of its ancestor criteria. When designing a marking scheme, tutors will not need to manually calculate the exact mark contributions of each marking criterion. The validation routine will also notify tutors of impractical mark contribution assignments.

To illustrate these features, let us assume we are to design a marking scheme for a programming coursework which can be assessed based on two main criteria: (i) *how well the end-product meets the specification* and (ii) *the appropriateness of the language constructs used*. These two criteria constitute 40% and 60% of the maximum possible score, respectively. These two marking criteria will be children of an AND node in e-CAF and the validation routine will check if their associated mark contributions sum to 100%. The first criterion can be subdivided into a set of, say four, marking criteria, each addressing one requirement in the specification. If those criteria are not mutually exclusive, they will be grouped under another AND node. Again, the validation routine ensures that summing the mark contributions of those criteria must be 100%, e.g. 20%, 30%, 30%, and 20%. Assuming that the maximum possible score is 50 marks, e-CAF will work out that the exact mark contribution of the first of those criteria will be 40% × 20% × 50 = 4 marks.

If a new, but not mutually exclusive, marking criterion, e.g. *how well the end-product was designed* which is worth 20% of the maximum possible score, is to be introduced to the top level of the marking scheme, the validation routine in e-CAF will inform the tutor that the mark contributions of 40% and 60% of the existing two criteria are needed to be changed because the new criterion will lead to an incorrect overall mark allocation of 120%. The validation routine will then automatically reduce the mark contribution of the two existing criteria by a total of 20%, i.e. to 32% and 48% respectively, before introducing the new marking criterion.

Automatic Generation of Performance Levels for a Marking Criterion

To facilitate the marking and feedback process, each marking criterion modelled in e-CAF is associated with a set of performance levels. A performance level describes how well the submitted work has met the marking criterion. To assist markers in selecting the most appropriate performance level as well as to inform students of how a given mark was derived, e-CAF enables tutors to specify a description for each performance level. Each performance level can be defined as a mark range or a single mark.

We recognise that defining detailed performance levels for each marking criterion can be a tedious task. To facilitate this process, e-CAF is equipped with an automatic feature for generating performance levels for each marking criterion. For example, if a marking criterion constitutes 10 marks of the maximum possible score and it is expected to have five single-mark (rather than mark-range) performance levels, e-CAF will generate five performance levels, each named by their associated mark value, i.e. 0, 2.5, 5, 7.5 and 10. Tutors can then annotate them with a description and/or modify them to meet their need.

To further facilitate the marking scheme creation process, e-CAF enables the same set or subset of performance levels to be applied to two or more marking criteria. For example, consider two marking criteria "*clarity of output*" and "*layout*", each constituting 9.9 marks to the maximum possible score (Fig. 4). Both of them are expected to have the same set of performance levels: "*OK*", "*some faults*" and "*major faults*". Once the first set of such performance levels have been created for the first criterion "*clarity of output*", e-CAF enables the tutor to associate these performance levels with the second criterion "*layout*". If the tutor wishes to change the mark implication for one or more of the performance levels, the change(s) will be reflected on both marking criteria by default, but the tutor may also choose to override this default behaviour.

Conclusion and Future Work

Assessment and feedback is an important aspect of teaching and learning. This is particularly so when the discipline emphasises the acquisition of practical skills. A well-designed marking scheme is a key to maintain consistency and ensure timeliness in marking. In this paper, we have given an overview of a new web-based flexible coursework assessment and feedback management system named e-CAF. A main goal of e-CAF is to facilitate the marking scheme design process. The fundamental design principle of e-CAF is to enable maximum reuse of existing assessment effort. The marking scheme design and creation feature of e-CAF enables tutors to focus on the design of a marking scheme by hiding the tedious and repetitious mechanical tasks, such as calculating the mark contribution for each performance level and editing the same set of performance levels repetitively, from them. The flexibility in marking scheme design also makes it possible for tutors to modify a marking scheme during the marking process without having to reassess the students' submissions.

We have illustrated how e-CAF facilitates the design and creation of flexible marking schemes. Such marking schemes play an important role in ensuring consistency and timeliness of the marking and feedback process. They also promote reliability in marking. Using e-CAF to design marking schemes, simplicity in the assessment and feedback process will not become a trade-off for accuracy and thoroughness. The marking process will also become more transparent to students because they will be able to see clearly how well their work meets the tutors' requirements and how they could improve in future.

The development of e-CAF is an ongoing process. To date, the marking scheme design and creation feature has been completed. The development of other key features such as coursework assessment and feedback bank management are almost complete. A beta version of e-CAF is planned to be released in Spring 2008 for user evaluation.

At present, e-CAF is designed to facilitate individual coursework feedback and assessment. In future, we plan to extend e-CAF to address the assessment needs for group projects. Our vision for e-CAF is to develop it into an electronic coursework assessment framework which will support a component-based marking procedure across several coursework items. This will then enable a detailed profile of students' work to be recorded and analysed during the students' lifetime in an institution, and hence facilitating the detailed tracking of students' progress in learning.

References

Black, P. (1995). Assessment and feedback in science education. *Studies in Educational Evaluation*, 21(3), 257-279.

Black, P., & Wiliam, D. (1998). Assessment and classroom learning. *Assessment in Education*, 5(1), 7-74.

Blackboard Inc. (2007a). Blackboard. [Online]. Available at: http://www.blackboard.com/. Accessed on: 10/09/2007.

Blackboard Inc. (2007b). Blackboard & WebCT. [Online]. Available at: http://www.webct.com/. Accessed on: 10/09/2007.

Brown, G. (2001). *Assessment: A Guide for Lecturers*. No. 3 in Assessment Series. Learning and Teaching Support Network (LTSN) Generic Centre.

Cole, J. (2005). *Using Moodle: Teaching with the popular open source course management system*. O'Reilly, Sebastopol, CA.

DES (1988). *Task Group on Assessment and Testing: A report*. Department of Education and Science, London.

Ekins, J. (2006). The use of interactive quizzes in Maths / Science. Presented at a Workshop on E-assessment in the European Association of Distance Teaching Universities Conference (EADTU 2006). Available at: http://www.eadtu.nl/academic-networks/eltan/files/Ekins%20-%20The%20Use%20of%20Interactive%20Quizzes%20in%20Maths%20Science.pdf. Accessed on: 22/04/2008.

Garrett, J. J. (2005). AJAX: A new approach to web applications. Adaptive Path, San Francisco, CA. [Online]. Available at: http://www.adaptivepath.com/publications/essays/archives/000385.php. Accessed on: 10/09/2007.

Google (2007). Google Web Toolkit -- Faster AJAX than you can write by hand. [Online]. Available at: http://code.google.com/webtoolkit/. Accessed on: 22/04/2008.

Heinrich, E. (2007). *WebCTConnect 1.1 Manual*. Institute of Information Sciences and Technology, Massey University, Auckland.

Heinrich, E., & Lawn, A. (2004). Onscreen marking support for formative assessment. In Cantoni, L., & McLoughlin, C. (Eds.), In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications (ED-MEDIA 2004)*, pp. 1985-1992. Association for the Advancement of Computing in Education (AACE).

Heinrich, E., & Wang, Y. (2003). Online marking of essay-type assignments. In Lasser, D., & McNaught, C. (Eds.), In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA 2003)*, pp. 768-772. Association for the Advancement of Computing in Education (AACE).

Joy, M., Griffiths, N., & Boyatt, R. (2005). The BOSS online submission and assessment system. *ACM Journal on Educational Resources in Computing*, 5(3), 1531-4278.

Sun Microsystems, Inc. (2007). Java EE at a glance. [Online]. Available at: http://java.sun.com/javaee/. Accessed on: 10/09/2007.

Surridge, P. (2007). *The National Student Survey 2006*. HEFCE, Bristol.

Ylijoki, O.-H. (2000). Disciplinary cultures and the moral order of studying -- a case-study of four Finnish university departments. *Higher Education*, 39(3), 339-362.

Zhang, J., & Heinrich, E. (2005). Using computers to support formative assessment of assignments. In Kommers, P., & Richards, G. (Eds.), In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications (ED-MEDIA 2005)*, pp. 4510-4515. Association for the Advancement of Computing in Education (AACE).

Zin, A. M., & Foxley, E. (2007). Automatic program assessment system. [Online]. Available at: http://www.cs.nott.ac.uk/cmp/more info/html/ASQA.HTM. Accessed on: 05/09/2007.

Acknowledgements

This project is supported by the Teaching Quality Enhancement Fund (TQEF) from the Higher Education Funding Council for England (HEFCE) administered by the Flexible Learning Development Centre (FLDC) at Aston University.