

A Proposal of User Interface for a Distributed Asynchronous Remote Evaluation System: An Evolution of the QUESTOURnament Tool

Luisa M. Regueras, Elena Verdú, Juan P. de Castro, María Á. Pérez and María J. Verdú

Department of Signal Theory, Communications and Telematic Engineering
University of Valladolid
Valladolid, Spain

{luireg, elever, jpdecastro, mperez, marver}@tel.uva.es

Abstract—This paper presents a distributed asynchronous system which allows remote evaluation of students' submissions. This system is being developed in the context of the EduJudge project whose aim is to provide a greater pedagogic character for the UVa Online Judge, an on-line programming trainer. The UVa Online Judge has been integrated into the e-learning platform Moodle and a competitive tool called QUESTOURnament. More specifically, this paper focuses on the design of the user interface of the evolved QUESTOURnament tool. It has been adapted to be able to manage different competition strategies and to have a centralized system for the management of questionnaires.

Keywords—Competitive learning; Moodle; remote evaluation system; User Interface

I. INTRODUCTION

The EduJudge project came up from the necessity of improving mathematical, science and technological competences at European level as it is promoted by the European Commission [1]. The main objective of this European project is to integrate the UVa Online Judge into an effective educational environment.

The UVa Online Judge [2] is an on-line programming trainer created in 1995 at the University of Valladolid by a research team led by Revilla, with the aim of training users who participate in worldwide programming competitions.

The users of the UVa Online Judge can access an important number of programming problems, send their programs and automatically receive a feedback. Recently, these users are demanding a greater pedagogic character for this tool. On the one hand, teachers would like to use the UVa Online Judge to organize activities in their courses of secondary and higher education. This would require the possibility of managing courses and users, both students and teachers. It would also be interesting to allow teachers to organize programming contests in the course environment, since the competition increases students' motivation and performance [3] [4] [5]. Moreover, the motivation is increased when the students can work with problems of the appropriate level for their ability [6].

On the other hand, the set of problems is continuously being incremented with the contributions of the users. Therefore, it is necessary to give the problems an adequate and common structure, adding metadata and creating a

search engine so that the problems are more accessible for their potential users.

The integration of the UVa Online Judge into the e-learning platform Moodle and the competitive tool QUESTOURnament will provide some of these required pedagogical functionalities for the EduJudge system.

II. ARCHITECTURE OF THE EDUJUDGE SYSTEM

The architecture of the EduJudge system integrates three main component types (see Figure 1):

- A Learning Objects Repository (LOR) for managing programming exercises and retrieving them based on the student's profile.
- An Evaluation Server (ES) able to evaluate the quality of a solution submitted and to support problems with several formats.
- A Learning Management System (LMS) composed of Moodle, QUESTOURnament and the EduJudge User Interface (UI).

Each component is autonomous (it can be deployed in different hosts) and replaceable by one or multiple alternative components of the same type. Moreover, all services are standard-oriented and publicly exposed so that other systems can interact with the EduJudge components.

The need for externalizing the evaluation of different learning activities has arisen in order to make educational systems independent from the evaluation agents. Therefore, the system proposed for remote evaluation is distributed and asynchronous. It has a UI properly integrated into Moodle that is in charge of managing the general interaction with users.

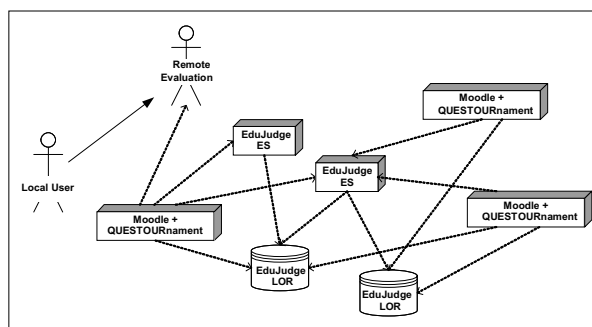


Figure 1. Diagram of the EduJudge system.

Although the UVa Online Judge is focused on programming learning, the EduJudge distributed system has been designed to be able to extend its use to other fields. Actually, the system could be used in different education levels and applied to any kind of problems whose evaluation can be outsourced. In fact, it should be possible to access not only to different ES like the UVa Online Judge, but also to human evaluators.

III. MANAGING CONTESTS WITH QUESTOURNAMENT

In formal learning contexts, the management of contests and their integration in the learning process is not an easy task and requires specific LMS. In order to satisfy this demand, the EduJudge system integrates the e-learning platform Moodle and the QUESTOURnament tool.

QUESTOURnament is a competitive e-learning tool, which is integrated into Moodle [7]. It allows teachers to configure both individual and group work environments where a set of challenges must be solved by students in a time-constrained way.

One of the most interesting characteristics of QUESTOURnament is the variable scoring system that intends to stimulate participation and to give an incentive to the effort. Since the time when a challenge is created until the end of the process, it goes through different phases. During an initial period of time, score remains constant, as proposed by the teacher, in order to allow students to understand and to take in the task. Then, score grows until the first correct answer is received. From this moment, score starts decreasing.

The QUESTOURnament system works in real-time, calculating and showing the current potential maximum score for each proposed challenge. In addition, a ranking with the highest five scores is permanently displayed on the screen.

IV. THE EDUJUDGE USER INTERFACE

The EduJudge User Interface consists of a set of plug-ins and modules for Moodle that allow applying different pedagogical approaches in combination with the automatic evaluation of programming problems provided by the ES.

Moodle allows developers to extend the functionalities of the platform in different ways. Some of these mechanisms have been implemented in the EduJudge UI:

- **Activity Module:** This module is the evolution of the QUESTOURnament module previously described. It enables Moodle with a way for organizing contests involving both programming problems and general purpose questions.
- **Question-Type plug-in:** This plug-in provides question-types for remote evaluation (by an ES) and remote storage (in a LOR). Hence Moodle can delegate the grading of some kind of answers to other external subsystems. The objective is to define a general purpose interface for remote grading, although it will be only available for programming problems in a short term.

A. Activity Module

The current EduJudge prototype allows teachers to use general questionnaires to define the challenges to be included in a contest. Questionnaires are based on the Moodle Question Engine, so that teachers can reuse and share repositories of questions all around the platform, and not only in QUESTOURnament. This approach also allows teachers to define challenges with automatic correction.

The activity module is based on the Model-View-Controller (MVC) architecture so that it can be adapted to various contest management strategies. The MVC architecture [8] splits the logic into three independent and modular parts:

- **Model:** It covers the abstract representation of the concepts (data) involved in the contest. It is implemented as a platform-independent object model with three logic layers: the first one to represent contest concepts, the second one to represent general e-learning platform concepts, and the third one as an abstraction layer to integrate the system into Moodle.
- **View:** It is a set of scripts to render the pages demanded by a user. This is a very specific part of the module heavily tied to Moodle.
- **Controller:** It holds all the logic of the module in terms of Forms and Actions. Forms represent the way of interacting through a web browser; hence it is easily portable to other platforms if needed. Actions process all decisions and control the flow of the application. This is completely independent of the platform.

Actions, Forms and Views are orchestrated through configuration files. So, it is possible to extend and customize the QUESTOURnament behavior by implementing only the new parts.

B. Question-Type Plug-in

The Question-Type plug-in allows users to have a centralized questionnaires management system for the QUESTOURnament module. Each challenge can be defined as a complete questionnaire with several questions from the repository. The plug-in has been implemented by using the support of the Question Engine and the Question Bank of Moodle. Thus, it has been necessary to adapt the following elements:

- The QUESTOURnament module, in order to use the questionnaire infrastructure of the QUIZ module of Moodle. Initially, the QUESTOURnament module only supported one type of answers: the Essay Question. Recently the tool has been adapted through the Question-Type plug-in to allow teachers to design and set contests consisting of a large variety of Question types of the QUIZ module (such as True/False, Multiple Choice...). Specifically, the Question Engine Application Programming Interface (API) of Moodle has been used in order to allow the use of

any form as a Question. The Question Engine is responsible for rendering the questions and processing students' answers.

- The way of adding challenges to the QUESTOURnament module, in order to take advantage of the support of the Question Bank (see Figure 2). It allows teachers to create questions of different types, to classify them in categories and to use them in different courses as well as to share them with other teachers.
- The assessment phases of the QUESTOURnament module have been adapted to the new types of questions, such as automatic correction, in order to speed up the assessment process.

V. CONCLUSIONS

Currently, the UVa Online Judge and the QUESTOURnament tool have already an important number of users worldwide, who will benefit from the integration and the improvement of the pedagogic character of both systems. In addition, the new QUESTOURnament tool allows teachers to define different contest strategies and to maintain and share a centralized system for the management of questionnaires with different question types.

Moreover, the development of the EduJudge project will contribute to extend the use of the UVa Online Judge to other scenarios such as secondary schools or Vocational and Education Training (VET) centres in order to improve and facilitate the lifelong learning, such as the European Commission proposes.

Through the distributed architecture of the system, many deployment scenarios can be possible. Multiple LMS, LOR and ES could be arranged and mixed easily, increasing the range of problems variety, evaluation methodologies and, hence, raising the possibilities and potential utility of this system. Additionally, resulting APIs and protocols would cover the general problem of remote evaluation, so that many grading tools and services, far from programming questions, can be connected to EduJudge-abled sites.

Besides, as a result of promoting and extending the use of the UVa Online Judge beyond the limits of programming subjects, teachers and students of other fields will be also able to benefit from the system.

ACKNOWLEDGMENT

The EduJudge project has been funded with support from the European Commission. This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- [1] Recommendation of the European Parliament and of the Council of 18 December 2006 on key competences for lifelong learning, 2006/962/EC, *Official Journal of the European Union*, L394, December 2006, pp. 10-18.
- [2] M. A. Revilla, S. Manzoor, and R. Liu, "Competitive Learning in Informatics: The UVa Online Judge Experience," *Olympiads in Informatics*, vol. 2, 2008, pp. 131-148.
- [3] R. Lawrence, "Teaching Data Structures Using Competitive Games", *IEEE Transactions on Education*, vol. 7, no. 4, 2004, pp. 459-466.
- [4] L. M. Regueras, E. Verdú, M. J. Verdú, M. Á. Pérez, J. P. de Castro, and M. F. Muñoz, "Motivating Students through On-line Competition: an Analysis of Satisfaction and Learning Styles", *Lecture Notes in Computer Sciences*, vol. 5145, 2008, pp. 167-177.
- [5] A. Siddiqui, M. Khan, and S. Katar, "Supply chain simulator: A scenario-based educational tool to enhance student learning," *Computers & Education*, vol. 51, no. 1, 2008, pp. 252-261.
- [6] W.M.C. Wu, H.N.H. Cheng, M-C. Chiang, Y-C. Deng, C-Y. Chou, C-C. Tsai, and T-W. Chan, "AnswerMatching: A Competitive Learning Game with Uneven Chance Tactic," *Proc. First IEEE International Workshop on Digital Game and Intelligent Toy Enhanced Learning (DIGITEL'07)*, 2007, pp. 89-98.
- [7] E. Verdú, L. M. Regueras, M. J. Verdú, M. A. Pérez, and J. P. de Castro, "Improving the Higher Education through Technology-based Active Methodologies: A Case Study", *WSEAS TRANSACTIONS on ADVANCES in ENGINEERING EDUCATION*, vol. 3, no. 7, 2006, pp. 649-656.
- [8] I. Singh, B. Stearns, and M. Johnson, *Designing Enterprise Applications with the J2EE Platform*, Second Edition. Addison-Wesley, 2001.

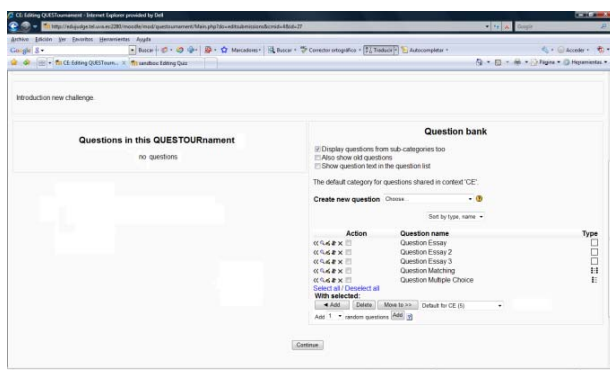


Figure 2. Adding a new Question to the QUESTOURnament module by using the Question Bank.