

iVeLA – Internet Voice e-Learning Application

Leonardo O. Moreira¹, Jefferson C. Silva², José R. O. Damico³, Marcus A. H. L. Morais⁴, Rodrigo F. Almeida⁵ and Javam C. Machado⁶

¹ Federal University of Ceará, Brazil, leoomoreira@ufc.br

² Federal University of Ceará, Brazil, jeffersoncarvalho@ufc.br

³ IBM Industrial Machine and Services, Brazil, jdamico@br.ibm.com

⁴ Federal University of Ceará, Brazil, marcushlm@gmail.com

⁵ Federal University of Ceará, Brazil, rodrigofelix@lia.ufc.br

⁶ Federal University of Ceará, Brazil, javam@ufc.br

Abstract: e-Learning environments have as purpose to provide distance learning using technologies which support that benefit. English is widely adopted around the world, and the global market has been requiring the employees have, at least, basic English skills. There are many e-learning environments, but none of them provides a voice recognition mechanism, essential for foreign language learning. This article presents iVeLA, a Web-2.0-based e-learning environment built over an extensible architecture and with voice recognition. Experiments were developed and analyzed to measure the accuracy of the voice recognition strategy. The results of those experiments proved that the strategy could be used on iVeLA.

Keywords: e-Learning, Voice recognizer, Learning Management Systems

1. Introduction

Throughout the human history, since the beginning of the school, many different methodologies were applied aiming to improve the learning quality. The most common methodology has been that which the teacher keeps in direct contact with the students. Due to the Web increasing and spreading, e-learning environments have been developed to provide resources for distance learning [2]. This development project named iVeLA has as major objective to create an application for supporting teaching over the Web. That application enables to recognize and to analyze the user pronounce through a voice recognition module.

That Web application will enable a conventional interface for the most common e-learning application resources, such as chat, forum, downloads and uploads of teaching resources and communication tools among the users. iVeLA will be validated as an e-learning application supported by computer and tested on an English course for business (provided by a third-party company).

The system interface for interaction with the user was designed and implemented in such a way to be as much user-friendly as possible for all stakeholders (users, administrators, teachers, tutors and students). By the reason of that, Web-2.0 resources were chosen to create more powerful and nicer interfaces, enabling information quickly and easily.

2. Related Works

The Blackboard [1] is considered the e-learning platform leader, because of its easy way of using, large adoption in learning institutions, educational flexibility, wide huge of resources and intuitive characteristics for distance learning and for supporting the presential learning. It offers user-friendly environment for the course management, content development, exams, activities of collaboration and assignments developed in groups.

The WebCT (*Web Course Tools*) [3] was created in 1996 by University of British Columbia. This environment can be used to create complete on-line courses or as supporting interface for existing ordinary courses. It presents as main features: an user-friendly interface, ease of use, tools for course administration, communication among users and collaboration. WebCT provides four kinds of users: the administrator, professor, tutor and student. WebCT main advantage is related to the creation of an integrated environment of e-learning which allows a set of tools, such as video conference, chat, e-mail, student progress statistics, exercises, bulletin board, self-testing, calendars, automatic generation of search index, and so on. In addition, the WebCT provides a set of administration tools to help the high level user in the course management process.

Moddle [4][5] is an open-source software for e-learning purpose which runs over a virtual environment. This environment is targeted to programmers and academic personnel. It consists of an administration system for educational activities, which allows the creation of online communities. It allows, in an easy way, that a student or a teacher participates of an online course. Moddle has also a good performance if compared to other non-free e-learning environments. Moddle is the only open source software that can be compared to other commercial systems.

3. iVeLA

The iVeLA is software for distance learning assisted by computer in web base environment with voice recognition. This software is capable of reacting to user interaction through a voice capturing by a microphone. The Web application which supports the process of teaching and learning offers a textual interface for usual functionalities of distance education, like chats, forums and file upload and

download. Basically, this application has two main characteristics. The first one is a computer-assisted environment that allows the teaching and learning process, using also a voice recognition software for English language course. The second one is the composition of a set of tools that allows activity monitoring and helps interaction among students and the teacher or tutors.

The extensibility is one of the main requisites of iVeLA. Hence, the iVeLA environment has a SOA (*Service Oriented Architecture*) facilitating the inclusion of new third-party services. In addition, SOA permits that our services can be used by other systems. Using this functionality, new features can be added through iVeLA interoperability with other e-learning systems. The service architecture supposes that the XML language [8] is used for the service specification and data sharing among services interactions.

The iVeLA environment was developed with many different technologies as free tools such as: Java enterprise development frameworks, IDEs, relational database management systems and technologies for modular production of extensible interfaces. iVeLA will be released to the software community at the end of the project as free software system, open source (SL/CS), under GPL license. Therefore, content providers will be able to use the environment to create open courses to the community.

3.1 Architecture

The iVeLA environment is composed by three main components: The Back-end, the Front-end and the Voice Recognizer. This section describes these components, their objectives and technologies. Figure 1 shows the technologies used and layers of the iVeLA architecture. The nearest one from user layer is the vision layer. It provides the user interaction with the environment. This layer user Web 2.0 resources, provided by AJAX [9] libraries. The Web pages are written in JSP (*Java Server Pages*) which provides dynamic interaction. SiteMesh framework [10] provides easy interface modification and the modularization of the view layer, forcing a visualization pattern.

The control layer is responsible for orchestrating the process flux of the environment, this means, receiving a requisition from the user, processing the requisition or delegating it to business layer and returning the result. Besides, the layer control provides interoperability functions, allowing communication with other applications. The authentication service is provided by the ACEGI framework which is integrated to the environment. ACEGI is a free software component managed by Spring framework [11] and it provides that kind of authentication service in corporative applications written in Java. Struts 2 framework [12] is a powerful component that enforces the MVC pattern. At iVeLA, Struts 2 is basically used to flow control between the process and the business layer. The Spring framework is also a free software based in project pattern which provides dependency injection (DI) for Java based applications. DI

is very important because provides a low coupling level between classes of Object Oriented application. With DI objects have unique characteristics which allow easy integration between the frameworks and the related technologies of our project.

The business layer is composed by EJB technology 3.0 [13] and Hibernate 3.0 [14] for persistence. EJBs are components from the Java EE platform [13] suitable for business layer implementation and easy database integration. The last layer, the voice recognition layer, is responsible for capturing user voice signals and for dealing with them, associating each one with a phrase or word. This feature is possible thanks to Julius software [6], which is an engine for voice recognition. An applet interface is used to capture the user voice and send it to Julius. The results of the recognition are returned to the user after that.

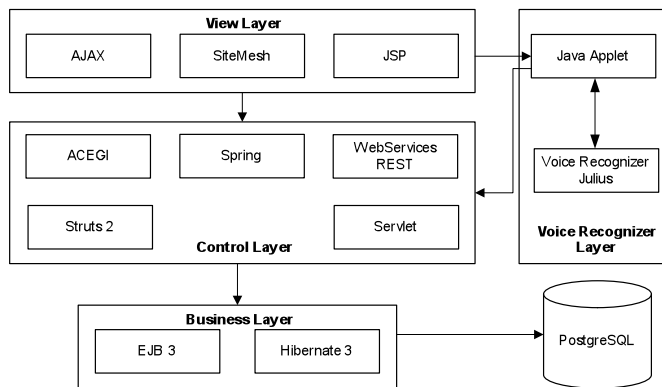


Figure 1 iVeLA architecture.

3.2 Back-end

The Back-end is the software administration component. It is responsible for the management of courses, students, enrollments, coordinators, teachers, tutors, content and activities related to distance education. From the using point of view, this component is manipulated by administrators, coordinator, teacher and tutors. The administrator user has granted access to the whole environment and all its features. He is also responsible for giving roles to other users of the environment. Hence, the administrator manages teachers, tutors, students and others administrators as well. The administrator manages grades, enrollments, course offerings and student marks. He has also a complete vision and full access to the courses.

The professor is responsible for managing course content and its progress. He is also responsible for being aware of his students' activities, for organizing

classes and for attributing marks. The tutor is an user delegated by the administrator to help the teacher in ordinary tasks. This kind of user has restrictive access to the administrative environment moderated by the professor of the current course. It is important to say that the administrator has full access to all of the coordinators' functionalities. The coordinators, instead, have full access of the professors' functionalities. Finally, the professors have full access to the tutor functionalities.

In addition to the user management functionality, the Back-end has also the feature of course creation, structure and content maintenance. The structure of a course deployed at iVeLA is composed by four levels. The first one is the course itself, where the name, description, target students and requisites are described. The second level is the discipline level. The third level is the discipline units, which presents the subjects or topics related to the discipline content. Finally, the fourth level is about the unit content, which specifies the content to be used by the students.



Figure 2 Initial screen from the Back-end.

In addition to the textual presentation of a class, iVeLA environment presents a set of tools for file upload and download. Uploaded files can help the students to get a better understanding of the class content. The class content is built in two flavors. The first one is the traditional one. The class creator (eg. Professor) uses an editor tool embedded at our environment to edit the text of a class. This text is after converted in a PDF (*Portable Document Format*) file. The second one was implemented as a way to speed up the class creation process. This is possible thanks to the upload feature of an entire package which includes the whole class

content. This package is built in such way that it contains a ready-to-use class, in HTML (*Hyper Text Markup Language*).

The iVeLA has a great numbers of maintenance functionalities for already created courses and grades. When is created a grade, the user specifies its course, coordinator, teachers, max number of students and the period which represents the life cycle of a grade. In addition, the user can maintain the attributes of a grade and its future removal. Another Back-end feature is the management of enrollments. In this feature, the responsible user can accept or reject enrollment requests from students. The user can also monitor the progress of its students and gather statistic information related to the grade. Helping in the learning process, the Back-end has modules for forum management and discussions related to grades. The newsflash is the module for viewing news or some interesting subject. The scheduler is the module responsible for organizing tasks and appointments at other user scheduler. Figure 2 shows a screen shot of the Back-end.

3.3 Front-end

The iVeLA Front-end is the module which the students interact with the courses that they are enrolled in and, this module is also used to request an enrollment. The Front-end offers to its users some tools to help them in the learning process and tools for professor-student interaction. Tools like the scheduler, the multimedia library, message passing system, newsflash, forum, chat and bulletin board are available for Front-end users. The scheduler is a tool that students use to organize their appointments and check reminders from administrative users. By using this tool, teachers can schedule tests, deadlines for homeworks and more. The multimedia library is a file repository for students. They can access this repository and download useful data, such as videos, text files or audio. The students can also upload their own files.

The message system is an interactive tool for message exchanging among users. Its main idea is to bound students and to generate documentation for communication. The newsflash is a tool which administrators can submit news for students. The incoming news can be viewed at the initial screen of the Front-end. The Forum tool is used for public discussion among users.

The Chat tool is a highly real time resource where teachers can create chat rooms and provide a communication among students. This tool is very useful for providing virtual classes and student assessment. Most of the available chat tools have poor technology in providing that functionality via web. At iVeLA, our chat tool is implemented using an well-known service, called IRC (*Internet Relay Chat*). This service is installed with iVeLA within its server and, at client's side, an applet (Java Software, for internet browsers) does the interaction with the server side. This approach leads to a very reliable chat server.

The Chat service, at client side, has an unique feature called "Blackboard". The Blackboard is a kind of a virtual draw panel which the teacher can "draw" pictures

to a better understanding of the class. The teacher uses the mouse pointer to draw in the Blackboard. After that, he updates the server iVeLA with his draw. Thus, the students can now update their local Blackboards and view the teacher's picture.

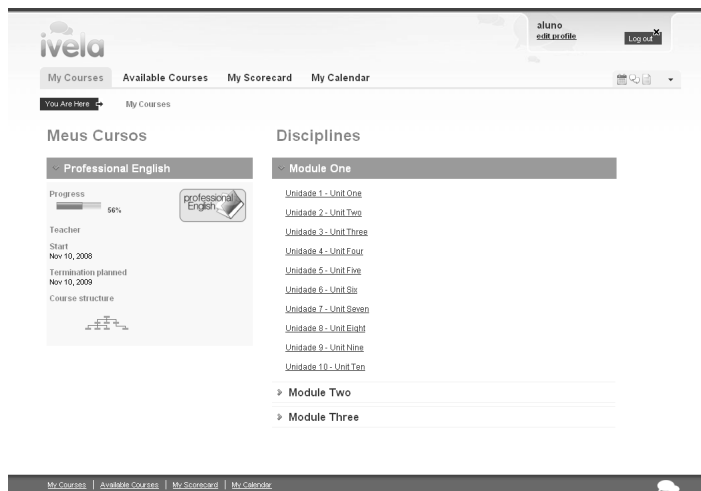


Figure 3 Initial screen of the Front-end.

The history is the tool used for students' progress visualization: their courses progress and bulletin board. The administrators (we mean, the professors and tutors) use the history tool as an information log of the functions and tasks done by the students. This is a very important tool to gather information about students' participation in a certain course or class. With this information, the administrators are able to build a profile for each student.

The Front-end initial screen shows the entire list of courses which the student who logged in is enrolled. This list is organized in a hierarchical way and the user can surf by clicking firstly in course, disciplines, units and unit contents (classes). When selected an unit content, a set of tools is showed (forums, chats and exercises or exams). Classes may have exams and/or exercises. Those kinds of exams have two sorts of question types: subjective and objective. As we said, the classes can be showed as a PDF file, using the features of this kind of file format, or they can be showed as HTML pages.

3.4 Voice Recognizer Component

The iVeLA environment offers as one of its course, the English Course. The English course and its exercises were used as our test case. An unique exercise is known as "speaking exercise". To implement the speaking exercise, we used two technologies: Java applets and *Julius* [6] voice recognizer. Java applets are well known by its capability of running in a web environment over the JVM (*Java*

Virtual Machine), previously installed at client side. The Applet Voice Recognizer (the name we gave to this mix, Java plus Julius) works as a *Façade* between the user (student) and Julius. The voice recognizer process assumes that the applet ran without any fault and correctly installs the Julius engine at client machine. After that, the client uses the high level microphone interface to submit his/her voice to Julius engine for further voice recognition.

Julius engine voice recognizer works as follows: given a certain grammar, a vocabulary and an acoustic model, a phrase or word (written in English) of the vocabulary can be recognized by the applet when it is spoken at the microphone. The grammar defines a set of rules that are used to build a phrase. The vocabulary is a set of words that are recognized by the acoustic model. The acoustic model is a very large sample of recordings, made in optimized environments. These recordings are used to gather mathematical information about the pronunciation of certain words in English (or any other language as well). These results are stored in a phonetic database, used by the recognizer.



Figure 4 Voice recognizer applet GUI

The acoustic model is supported by the North American group VoxForge [7], as free software, under GPL license. Recordings are submitted to specific tools that generate the model. The gathered phonemes are submitted to mathematical algorithms which compare the microphone audio spoken by the student with the actual phonetic database. This comparison generates a feedback about the accuracy level of recognition. This result provides how good, or bad, was the student's pronunciation.

Figure 5 shows us the applet graphical interface for voice recognition, with its interaction buttons.

4. Experiments

As we said, the iVeLA environment contemplates the main functionalities presented in almost all distance learning education systems. Our main and unique component is the voice recognizer. Therefore, in order to prove its scalability and

efficiency, we made a large number of tests, using many different levels of students who knew and did not know English. The tests took place at Federal University of Ceará (UFC, in Portuguese) during three days. The test consisted in five phrases in English, which were listened and repeated by the student. Table 1 shows us these phrases.

Table 1. Phrases used in our test case

Phrases
1. "My name is bob"
2. "I am from Canada"
3. "Here is my card"
4. "He is Jack"
5. "He is French"

The phrases are short because the recognition rate decreases with large phrases. Our solution was to break big phrases into small ones that could be easily detected by a person whose knowledge in English is medium. Therefore, the whole English Course includes small and some challenging phrases.

The phrases summed together were repeated 930 times. That means, 186 times per phrase. The application assessed the students by the following terms: (A) students who got OK at the first try; (B) students who failed the question and try number finished; (C) students who got OK at a question but only on second try.

The test case shows that only 57 samples were not recognized by the applet. This can be explained by the early stage of acoustic model development. Besides, not every voice in the world can be matched by a single program. There are a lot of accents and voices singularities, of course. The total score showed us that 873 samples (664 + 209), from 930 samples, were recognized by our applet. This proves that both the applet and Julius were accepted and they can be used as tool in iVeLA.

Table 2. Result of evaluation 1

	Phrase 1	Phrase 2	Phrase 3	Phrase 4	Phrase 5	Total
A	128	146	119	129	142	664
B	3	7	32	9	6	57
C	55	33	35	53	33	209
Total	186	186	186	186	186	930

5. Conclusion

This paper demonstrated the iVeLA environment and its functionalities, architecture and innovations. The iVeLA covers all the features needed by a good

distance learning software. Besides, iVeLA uses Web 2.0 which results in a beautiful and easy to hand interface.

The iVeLA has a modular, flexible, scalable architecture which results in a strong environment, reliable and extensible. Besides, the Web Service feature present in iVeLA enables the environment to interoperate with other systems. The experiments proved that the voice recognizer was well accepted. Finally, iVeLA was the best well suited for English language teaching among the analyzed environments.

Acknowledgement

The authors thank Emanuelle Vieira, Joe Kliment, Jonas Araújo, Maristella Myrian and Nelson Eldoro that contributed to iVeLA. We would also like to thank Flextronics Institute of Technology for the financial support of this research.

References

1. Yaskin, D., Everhart, D.: Blackboard Learning System. (Available via WEB, 2008), http://library.blackboard.com/docs/k12/Bb_Whitepaper_K12_Learning_System.pdf of subordinate document. Accessed 10 Dec 2008.
2. Martins, C., Azevedo, Isabel. Ribeiro, L.M., Carvalho, C.V.: Requisitos e avaliação da eficácia de ambientes de sistemas integrados de e-learning/informação no ensino superior. (Available via DIALOG, 2008), <http://www-gist.det.uvigo.es/ie2002/actas/paper-211.pdf> of subordinate document. Accessed 8 Dec 2008.
3. Pinto, M.F.P., Angelo, M.F., Silva P.G.T.: Estudo Comparativo das Plataformas WebCT, AulaNet e TelEduc. In: II Semana de Tecnologia da Fatec - I Jornada de Pesquisa (2006), São José do Rio Preto, Brasil.
4. Pulino Filho, A.R.: Moddle: Um Sistema de Gerenciamento de Cursos. (Available via WEB, 2005), http://www2.tce.sp.gov.br/ecp/pdf/manual_completo_moodle.pdf of subordinate document. Accessed 07 Jan 2009.
5. Tuparova, D., Tuparov, G.: Learning paths in open source e-learning environments. In: Current Developments in Technology-Assisted Education (2006), Badajoz, Spain.
6. Julius Voice Recognizer Home Page. <http://julius.sourceforge.jp>. Accessed 07 Jan 2009.
7. VoxForge Acoustic Model. <http://voxforge.org/home>. Accessed 07 Jan 2009.
8. Extensible Markup Language (XML). <http://www.w3.org/XML/>. Accessed 07 Jan 2009.
9. Ajax Tutorial. (Available via WEB, 2009), <http://www.w3schools.com/Ajax/Default.Asp> of subordinate document. Accessed 07 Jan 2009.
10. Sitemesh. <http://www.opensymphony.com/sitemesh/>. Accessed 07 Jan 2009.
11. Spring Framework. <http://www.springsource.org/>. Accessed 07 Jan 2009.
12. Struts 2. <http://struts.apache.org/2.x/>. Accessed 07 Jan 2009.
13. Java EE at a Glance. <http://java.sun.com/javaee/>. Accessed 07 Jan 2009.
14. Hibernate. <http://www.hibernate.org/>. Accessed 07 Jan 2009.