International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:4, No:8, 2010

A multilingual virtual simulated patient framework for training primary health care students

Juan L. Castro, María I. Navarro
Víctor López, Eduardo M. Eisman, José M. Zurita

***Abstract*—This paper describes the Multilingual Virtual Simulated Patient framework. It has been created to train the social skills and testing the knowledge of primary health care medical students. The framework generates conversational agents which perform in serveral languages as virtual simulated patients that help to improve the communication and diagnosis skills of the students complementing their training process.**

***Index Terms*—Medical Training, Conversational Agents, Patient Modeling**

I. INTRODUCTION

Colleges and medical foundations are very interested in methods for training the social abilities of primary health care medical students into the context of doctor-patient appointments. Usually this training consist in role playing sessions with actors performing the patientsŕole. An Embodied Conversational Agent (ECA) could be of great help for saving economical resources and improving the process of training, providing a tool that could be used anytime the students want to practice without additional cost.

The development of a ECA with a good performance for a specific domain is a hard process. There are some frameworks which could help in the development of the ECA [1][2] but they are primarily dedicated to the technical part of the ECA, i.e. specifying some arrangement of the modules that typically conform the architecture of an ECA and some general knowledge paradigm that can be employed to adapt it to the concrete domain, formalizing the knowledge by means of a knowledge engineering process. If the system needs to provide several kinds of ECAs with distinct behaviour to be useful, as in doctor-patient appointments where each patient has got different diseases and personality, this process becomes a bottleneck each time a new patient is going to be developed. To the best of our knowledge, there is not any other ECA framework which allows to simulate the performance of different kinds of patients with different diseases and personalities in a primary health care doctor-patient appointment without the need of an extensive knowledge engineering process to specify the knowledge for each of the new patients and diseases. This is the main contribution of our Virtual Simulated Patient (VSP), a knowledge based model of patient and diseases composed of symptoms, some features of them and localizations of these symptoms in the human body in order to make very easy the creation and combination of new diseases and patients. The patient model has got some features that allows the creation of new patients with distinct age, gender, personality mood and cultural level. The ECA part of our framework employs these models and simulates the corresponding patient. At the present, we are working with some european partners in the Multilingual Virtual Simulated Patient project (MVSP), which is aimed to increment the languages that the VSP is able to understand and talk to another six languages besides Spanish: Bulgarian, English, German, Hungarian, Italian, and Portuguese.

This last point will be developed in detail further in this paper (in section III-C). The VSP framework has been described in other papers [3], [4] so here we are going to give an sketch of the framework referring to the other papers where applicable.

The overview of the paper is as follows: in the next section we are going to see some related works in the field of patient simulation. In section III we are going to describe the patient model, the framework and the knowledge model used for the multilingual ability. After this, some tests will be presented in section IV and finally the conclusions will be drawn in section V.

University of Granada, Spain. (Corresponding author Juan L. Castro, email: castro@decsai.ugr.es). The VSP project was supported by Consejería de Innovación, Ciencia y Empresa of the Junta de Andalucía (TIC 1424) and the spanish government (TIN2007-67984). The MVSP Project is supported by the European Union inside the Transversal Programme KA3-ICT: Lifelong Learning Programme, Call for Proposals 2008 (EAC-30/07), Reference: 143423-2008-LLP-ES-KA3-KA3MP from January 2009 to December 2010.

II. RELATED WORK

An Embodied Conversational Agent technology is applicable to any situation where there is a human-to-human interaction, with the ECA playing the role of any of them. ECAs have been developed for very different fields like a museum tourist guide [5] or a virtual tutor for students of first courses in physics [6], but as far as we know there are few previous works in the field of patient simulation for training in medical environments that employ this technology. Dickerson, Johnsen, Raij, Lok, Hernandez and Stevens [7] present an ECA with its responses and behaviour written in a script based system similar to AIML [2]. This script captures in advance the typical questions and dialogue moves that a doctor uses for patients with a specific disease and the responses of the ECA. These scripts have descriptors, like sentences said by the doctor, or actions, like pointing, that are typical in this situation. The agent's behaviour in response to a user's input is done by selecting the script with the most similar descriptor to the previously normalized user's input. This solution needs writing scripts for all the specific situations of a patient's appointment.

The scripting effort is very high because it does not allow separately specifying the knowledge about the patients model, the symptoms, the cognitive and emotional behaviour and the linguistic knowledge. Kenny, Parsons, Gratch, Leuski and Rizzo describe a virtual patient for clinical therapist skills training [8] that has a separate model of non verbal behaviour and a statistical selection scheme of the agent's answer with some considerations about possible dialogue situations, like off-topics questions, orders given by the user to the agent, e.g. "Could you repeat that?", or alternative answers to question like "Do you have anything to add?". Although it uses rules to assign the non-verbal behaviour to the ECA depending on the selected answer, as before, this work lacks of a model to relate the symptoms of the diseases to the patient's behaviour.

III. The Framework

The framework is composed by two subsystems as depicted in figure 1 the Patient Creation Subsystem (PCS) which creates new patients and the Patient Simulation Subsystem (PSS) which simulates them. The PCS takes the physical patient model and combines it with the defined diseases following an aggregation process that uses a fuzzy operator. The PSS is an ECA built with a blackboard architecture that connects other four modules, each one doing the processes of translating the user input, creating the semantic patient answer, calculating the emotional state of the patient, and translating back the semantic answer into natural language.

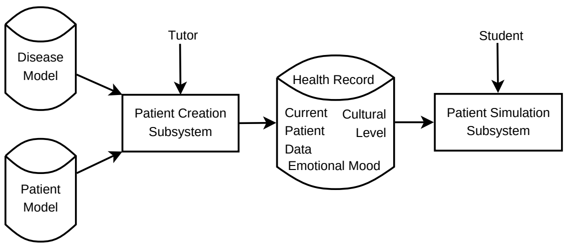

Fig. 1. MVSP Framework Architecture.

We are going to see each of these two subsystems in turn in the following subsections with a special remark in the multilingual ability of the natural generation module.

A. The Patient Creation Subsystem.

This system provides users with a graphical interface that lets them selecting the patient characteristics: age, gender, cultural level, initial mood at the start of the appointment, personality and the patient diseases. The Base Patient Module (BPM) gets this information and chooses a patient physical model based on the patient's features that the Patient Generation Module (PGM) combines with the models for the patient selected diseases using some integration function.

The patient physical model contains four entity types holding relationships between them:

- Symptom. The instances of this type represent symptoms, like temperature, pain or dizziness. They have some base, minimum and maximum intensity values for each one. A symptom is specified in its usual units, e.g. temperature is expressed in °C, and if it does not have any (like pain), it is expressed in a percentage.
- Location. The instances of this class represents the human body locations, like arms, chest, or left eye.
- MedicalTest. The instances of this class represents the results of some test that a doctor could make to a healthy patient. They have the same properties as the class Symptom, but their measurement units are the usual ones that real tests use.
- Features. Each patient model has got some features that serve to identify the kind of patient that a model is applicable to, like gender, personality, age, emotional state or cultural level.

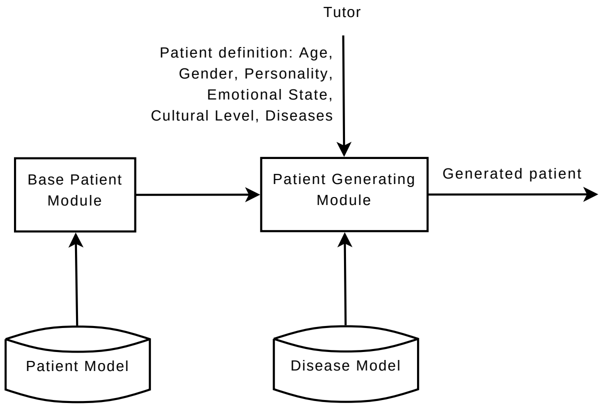

Fig. 2. The Patient Creation Subsystem.

TABLE I
An example of a disease's Model: Horton disease

Symptom	Location	Progression (Intensity)
pain at rest	right eye	60, 60, 60, 0, 0, 0, 0, 0, 0, 60, 60
pain at rest	jaw joint	60, 60, 60, 0, 0, 0, 0, 0, 0, 60, 60
pain at rest	right cheek	60, 60, 60, 0, 0, 0, 0, 0, 0, 60, 60
burning sensation	right eye	30, 30, 30, 0, 0, 0, 0, 0, 0, 30, 30
inflammation	right eye	60, 60, 60, 0, 0, 0, 0, 0, 0, 60, 60
congestion		70, 70, 70, 0, 0, 0, 0, 0, 0, 70, 70
unusual weeping		70, 70, 70, 0, 0, 0, 0, 0, 0, 70, 70

A disease is just a description of locations and the symptoms affecting them. The difference between the patient model and the disease model are the properties in the class Symptom. A disease produces a variation over the time of the intensity of a symptom above or below its base value so in the disease model the class Symptom has another property called *progression*. This holds pairs of (intensity, time) values that represent the progression of the symptom over the disease time. The same is applied to the medical tests that the doctor could make to a patient that who suffers from the disease. Our system contains several predefined diseases like anaemia, asthma, diabetes, and different kinds of migraine like the one in Table I, but new diseases could be easily added, just using the described model.

The patient model and the disease model together gives the

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:4, No:8, 2010

user the flexibility to create several kinds of patients without effort. However the user does not have a direct access to these models, instead the selection of the patient model to be used is done by the BPM using some of the parameters that the user chooses for the new patient. The BPM selects between them checking rules that takes into account the features selected for the patient like gender and age. When there are several patient models applicable to the same patient's features, one of them is selected according a ramdom selection schema. When the patient's model is selected, it is sent to the PGM to combine it with the model of the diseases selected in order to create the new sick patient using the Dombi's operator [9].

This method allows the creation of new patients with very little effort and a way to automatically integrate different diseases holding the restrictions specified for the symptoms. Our system provides the user with base patient models for both sexes and a small set of diseases, but the user is able to easily create new diseases and base patient models if required. In this section we have sketched what are the models of a patient and a disease, but if you want a deeper insight of how this module uses the Dombi's operator or what is the semantics of this operator applied to the diseases see [4]. The generated patients are later loaded on demand by the Patient Simulation System (PSS) and this module serves an adequate ECA that plays the role of the loaded patient. In the next section we are going to describe how the PSS interacts with the generated patient model.

B. The Patient Simulation Subsystem.

The modules of the PSS have been developed knowledge-oriented, using processes that can handle this knowledge, like Rule-Based Systems or Case-Based Reasoning. In our application, this knowledge is specified by the generated patient model together with other knowledge used by the modules of the PSS.

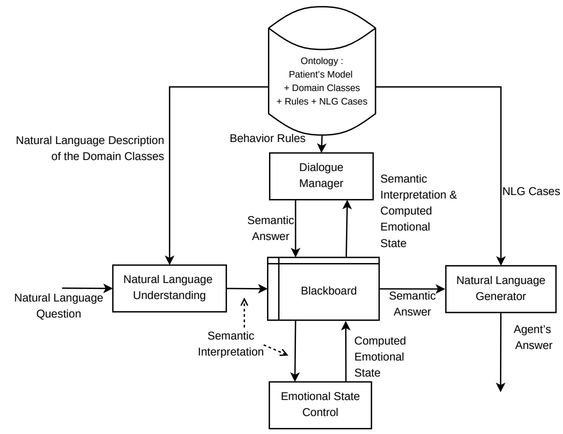

Fig. 3. PSS Architecture.

The Natural Language Understanding (NLU) module employs the words in natural language that stands for the entities in the medical domain to parse the user sentence and creates a semantic interpretation of that sentence. The Emotional State Controller (ESC) and the Dialogue Manager (DM) are two rule-based processes that need the knowledge related to the patient's state, i.e., symptoms, the previous emotional state, the semantic translation of the user's sentence and a context of previous patient model and the user's semantical interpretations to compute the new emotional state and the agent's semantic answer. The ESC uses fuzzy rules and the DM uses crisps rules. Finally, the Natural Language Generator (NLG) takes this semantic answer and uses a CBR process to translate it into natural language. This process retrieves a case from the semantic answer that contains a text template. Later, it adapts this case to the contents of the semantic answer and fills the template to make the natural language answer of the virtual patient. In this paper we are going to show a brief description of the operation of each module of the PSS. For a deeper understanding of these modules see [3].

TABLE II
EXAMPLE OF ACTIONS AND QUESTIONS CHARACTERIZED IN THE VSP

Actions*	**Example of characterized sentence**
QuestionIntensitySL	"How is the pain in your arm?"
QuestionFrequencySL	"How many times does your arm hurt?"
QuestionIntensityS	"Do you feel very tired?"
QuestionDurationS	"How long have you been feeling ill?"
QuestionReasonApp	"What's the matter?"
ActionGreet	"Hello"
ActionFarewell	"Bye"
ActionDiag	"You have migraine"

* Names of the actions stand for their semantic contents, S = Symptom, L = Location.

The NLU module is restricted to the language and the characteristics of a dialogue that is employed in a doctor-patient appointment. This is a sort of structured dialogue, with some introductory sentences, some questions posed by the doctor to gather information about the illnesses of the patient, a diagnosis and a farewell. The NLU module represents different types of user sentences using "communicative actions" (table II). With this set of actions the NLU parses the input sentences looking for correspondences between words and semantic categories and it assigns an interpretation to the entities discovered, e.g. a question like "How is the pain in your arm?", is translated into the following semantic entities: "how is" = QuestionIntensitySL, "arm" = Location arm, "pain" = Symptom pain.

This module sends the interpretations to the blackboard, and the ESC and DM get them. The ESC works to process the new emotional states of the patient based on two essential concepts, emotional state and personality. The emotional state is built on the basis of a set of basic emotions, which we call *emotional attributes*. These attributes represent the most relevant emotions that a real patient can feel. The value of every attribute is a real number between 0 (total absence) and 1 (total presence). Table III shows two examples of emotional state, which can belong to different patients or to the same one but in different times.

Personality is a set of static features defined for each of the eight emotional attributes described before. When assigning a personality to a particular patient, a little variation in the value of the emotional attributes associated to that personality is carried out so that two patients that share the same personality

TABLE III
SOME EMOTIONAL STATES

EMOTIONAL ATTRIBUTES							
JOY	DIS.	ANG.	FEAR	WOR.	SUR.	SAD.	EMB.
0.73	0.26	0.12	0.08	0.11	0.39	0.04	0.10
0.05	0.03	0.07	0.48	0.39	0.10	0.86	0.01

Legend: Dis(dain); Ang(er); Wor(ry); Surp(rise); Sad(ness); Emb(arrassment);

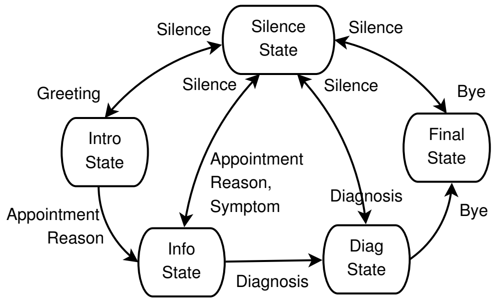

Fig. 4. Conversation states for the MVSP.

do not have to be exactly the same.

Each emotional attribute is updated independently using a fuzzy rule based system. The input variables of these systems refer to the personality type of the patient, his health (if he is ill or tired, he should be prone to get angry with others), the conversation type (the interpretation of the question that could be related to the appointment reason or a particular symptom, sexual content questions...), the features of the symptom the doctor asks about (intensity, frequency, duration, and weight), and so on. The output of these systems is a specific variation level to produce in every emotional attribute (high negative, medium negative, low negative, zero, low positive, medium positive, high positive, and personality).

Once the emotional state has been processed, it is sent to the blackboard. The DM gets this emotional state and the interpretation of the user sentence and tries to give the adequate behaviour to the patient. This process is a Rule-Based System that uses a finite state Markov process using two sets of rules in each state: transition rules and state rules. Transition rules control the next state to apply, and state rules decide the MVSP semantic answers. Each rule has a selection probability associated and the DM selects the rule to be applied from the set of fired rules taking into account this probability.

In figure 4, we can see five conversational states depicting the DM conversation model. The arcs are the transition rules with doctor's sentences as antecedents, and each state has rules giving an appropriate semantic answer at a given moment in the conversation that could be translated into natural language by the NLG. Each of these modules has been explained with more detail in [3] but the NLG module has suffered some changes to made it ready for the multilingual ability that will be explained next.

The NLG in our system is done by means of a Case Based Reasoning process. This process employs some linguistic knowledge at several levels: Paragraphs, Phrases and Lexical Descriptions of several kinds of entities. A lexical description of an entity is several ways of mentioning that entity in some context. In our doctor-patient appointment domain, the entities to be mentioned are symptoms (like fever or pain), human body locations (like arm, chest, left eye...) and intensities, frequencies and durations of symptoms (like "a lot", "a few times a day" or "for a week"). A Phrase is composed by a text template and the entities that this text template can talk about, besides a tag which classifies the phrases by their communicative intention, e.g. the tag "ExplainSymptom" classifies a Phrase in telling us that this phrase is trying to explain a Symptom. A Phrase could be a complete sentence or just a phrase with some syntactic structure. The last level of linguistic knowledge are Paragraphs. A Paragraph is composed by one or more Phrases, a text template which arranges them and like the Phrases a tag classifying the Paragraph by their communicative intention, e.g. the tag "Greeting" tell us that some paragraph could be employed to greet the doctor at the start of the appointment or the tag "Answer" could be employed to answer a question. A Paragraph has always got a complete syntactic structure of one or more sentences. All this knowledge is stored in a database and when a semantic answer arrives the NLG selects the adequate Paragraphs, Phrases and Lexical Description levels for translating the answer into natural language. The NLG has got the following stages in its algorithm:

1) Indexation. This stage is done once at the beginning of the process indexing the Paragraphs in descending order based on the number of instances of Phrase it contains and, for the Paragraphs with only one instance of Phrase, the action they have.
2) Retrieval. This stage uses the indexed case base and tries to match some of the restrictions in the semantic answer to the restrictions in a Paragraph and Phrases, i.e. it should try to match the tags signing the intention of a Paragraph and their Phrases with the tags in the semantic answer and some of the entities of the semantic answer with the entities associated to the Phrase. If all the Phrases in a Paragraph match, then the Paragraph is eligible. The process randomly selects one Paragraph between the eligible ones and saves it on a list. The retrieving stage is repeated until there are no restrictions left in the semantic answer.
3) Adaptation. This process fills in the gaps of each text template of the instances of Phrase in the Paragraphs previously selected with the words associated to the Lexical Description of the entities matched by the semantic restriction of the Phrase.

If the semantic answer AppointmentReasonObjective (Specify (high: Intensity, pain: Symptom, lefteye: Location), Specify (freqm: Frequency, numb: Symptom, tongue: Location, hands: Location)) is sent to the NLG, it retrieves the Paragraphs with the same tags and entities in their Phrases and selects one of them. If CaseMotive1 has been retrieved, it takes its Phrases' text templates: *"I've got a #Intensity# #Symptom# in my #Location#"*, *"#Frequency# my #Location# and #Location# become #Symptom#"*. Now, it assigns the entities matched in the semantic restriction of the semantic answer, i.e., high, pain, lefteye for the first Phrase and freqm, hands, tongue and numb for the second, to the lexical description level referenced in the template. This step specifies the exact entity that is going to be used in the template, resulting in the adapted templates: *"I've got a strong pain in my left eye"*, *"sometimes my hands and tongue become numb"*. Finally, it replaces the referenced phrases in the text template of the Paragraph, "#PhraseStrongPainInLoc# and #PhraseFreqSymptomIn2Locs#", with the obtained sentences of the phrase making the final answer *"I've got a strong pain in my left eye and sometimes my hands and tongue become numb"*.

In the next section we are going to show the improvements made to the described framework that primarily concerns to the NLU and NLG modules.

C. Multilingual Ability.

As stated in the introduction, the original VSP can only communicate in Spanish. Although the semantic knowledge, i.e. the concepts, used by the VSPs when they are communicating with the user are independent of the language that the VSPs should use to understand and talk about these concepts, in the original VSP, they were associated only with Spanish linguistic knowledge. To overcome this problem the VSP has been extended adding two databases for collecting the linguistic knowledge required for language understanding and generation. These databases gather for the NLU module, expressions in several languages referring to the question types (some of them have been shown in table II), the symptoms, the localizations, the properties (intensities, frequencies, durations), greetings, farewells and other expressions and for the NLG module, the linguistic levels of Paragraphs, Phrases, Symptoms, Localizations and so on, in those other languages.

The multilingual ability of the MVSP comes from the outcome of an European Project starting in 2009 and finishing in 2010 with the objective of increasing these language databases. Several partners from different countries have participated in this project filling the databases and in the present the MVSP is able to communicate in Spanish, English, Bulgarian, Hungarian, German, Portuguese, and Italian.

To enhance the work of these partners two tools have developed. The Natural Language Editor (NLE) allows the partners to introduce the Paragraphs, Phrases and Lexical Descriptions of several entities in their own language that are employed in the language generation task and other of our partners, the Andalusian Center for Innovation and Technologies of Information and Communication (CITIC), has developed two tools has developed the Question Comprehension Module Tool (QCMT) that gets the natural language description of question types and other entities like symptoms or locations in each of the languages for the language understanding task. Both tools are web applications and have let each partner to work independently from each other filling the MVSP language database with their own idiom. The NLE provides severals tabs representing each of the levels needed by the NLG module. Figure 5 and 6 are screenshots of the NLE and QCMT web

Fig. 5. Screenshot of the NLE for the Portuguese language.

Fig. 6. Screenshot of the QCMT.

applications, the first showing the phrases tab in Portuguese language and the last showing the symptoms tab. The NLE and the QCMT have proven to be very useful tools to gather the linguistic knowledge required by the MVSP.

In addition to the multilingual limitation, the original VSP tool presented a second problem. The patients generated by the system were all of them too ideal. It is true that it is very important that the patients reflect the main features of the typical inhabitants of each country because these are the kind of people who health professionals must usually treat. However, we must not forget that this does not always happen in real life. We live in a plural society, where people of different races, languages, and even accents, live together in the same city. For this reason, it is also very important that health professionals know how to face the specific communication problems that may arise during the clinical interview process. The MVSP framework allows the definition of non-native patients, representing a minority immigrant population of each country. These non-native patients could have some problems understanding the doctor if he or she uses a very technical vocabulary. They could also find it very difficult to express the reason of the appointment, using not grammatical sentences. The framework that we have presented in this paper not only facilitates health professionals' training but also breaks the language and culture barriers. The MVSP is able to speak and understand several languages becoming a valuable tool for training medical students of many nationalities.

IV. TESTS

One of our partners is the IAVANTE Foundation which is dedicated to professional medical training. In order to validate

TABLE IV
OVERVIEW OF THE SURVEY RESULTS

Question	Average score
Visual aspect (face, realism, etc.)	2.3571
Natural language generation (realistic, suitable, etc.)	2.7142
Interaction (interface, etc.)	3.2142
Question recognition (failed questions, etc.)	1.8571
About use (realistic conversation, etc.)	2.6428
Useful (training interviews, etc.)	2.7142
Mean	**2.6857**

the MVSP, we prepared a validation test that consist in a play with three patients each one with a different disease or combination of diseases. The IAVANTE foundation selected the three diseases to test, Smoking, Asthma and Respiratory tract infection and organized a pilot course called Integrated Assistance training for patients suffering from pulmonary disease. The course was proposed to six specialists in respiratory diseases who had to interview each one of the three patients as if they were in a patient-doctor appointment. The interviewers were not instructed about what questions they can do to the MVSPs, they only were told that they should release on their own experience to try to diagnose the illnesses of the MVSPs and that they were free to ask anything they want. After the session, each doctor had to fill out a document with several questions scoring the features of the MVSP in the interval [0, 5].

Table IV shows the results of the survey, The first column shows the sections that composed the survey and between brackets there are examples of the features that were asked for scoring this section. The last column shows the score obtained on that section calculated as an average score between the scores of the features that composed the section.

The model passed the tests except in the Question Recognition section. This lower result was due to questions that were outside the domain. As we said before the questions were free, without any restriction about the sentences that the doctors could ask. The best results were obtained in the Interaction section remarking its usability. The most important result obtained in this validation test was that all the doctors gave a correct diagnosis to each of the three patient's illnesses, meaning that the MVSPs performed reasonably well acting like patients and that their diseases were correctly modeled to be credible enough for the doctors. Apart from this, the preparation of this test confirmed us the enhancement that the MVSP framework provides in terms of the creation time for patients and diseases. Preparing all the three patients and diseases for the test did not take us more than 15 minutes.

V. CONCLUSION

We have developed a Multilingual Virtual Simulated Patient Framework which is useful to train the social skills of primary health care medical students as is drawn from the validation tests conducted. The knowledge based model of disease and patient makes very easy the creation and combination of both. The ECA part of our framework employs these models and simulates the corresponding patient which is able to perform in several languages behaving according to their diseases and personal characteristics, like personality, mood, cultural level and so on. The multilingual ability of the NLG module allows the ECA to perform in several languages and the conducted tests show us that although it needs a bit more tuning in its comprehension abilities to be ready for deployment on a medical training environment, it is a good tool for training medical students. In short, the MVSP framework presents a new methodology that complements and eases the health professionals training process, decreasing multilingual and multicultural communication problems and helping to improve the communication and diagnosis skills.

ACKNOWLEDGMENT

The authors of this paper thank to Fundación IAVANTE, Centro Andaluz de Innovación y Tecnologías de la Información y las Comunicaciones (CITIC), Friedrich-Alexander-Universität Erlangen, Politecnico di Torino - Dip. di Automatica e Informatica, Sheffield Hallam University Faculty of Health and Wellbeing, TrueBlue Consultancy, Scienter Soc. Cons. A.R.L, Katedra po farmakologia i toksikologia, Medicinski universitet of Sofia, Faculdade de Ciências da Saúde Universidade da Beira Interior, HCI Productions OY and Institute for Basic and Continuing Education of Health Workers their work filling the linguistic databases that give the MVSP framework its multilingual ability.

REFERENCES

[1] D. Bohus, A. Raux, T. K. Harris, M. Eskenazi, and A. I. Rudnicky, "Olympus: an open-source framework for conversational spoken language interface research," in *NAACL-HLT '07: Proceedings of the Workshop on Bridging the Gap*. Association for Computational Linguistics, 2007, pp. 32–39.

[2] R. A. F. Wallace, "Artificial intelligence markup language (aiml)," *http://www.alicebot.org/aiml.html*.

[3] V. Lopez, E. M. Eisman, and J. L. Castro, "A tool for training primary health care medical students: The virtual simulated patient," 2008, pp. 194–201.

[4] V. López, J. L. Castro, and J. Vázquez, "Reducing the effort in the creation of new patients using the virtual simulated patient framework," in *FUZZ-IEEE'09: Proceedings of the 18th international conference on Fuzzy Systems*. IEEE Press, 2009, pp. 764–769.

[5] S. Kopp, L. Gesellensetter, N. Kramer, and L. Wachsmuth, "A conversational agent as museum guide - design and evaluation of a real-world application," in *Intelligent Virtual Agents, Proceedings*, ser. Lecture Notes in Artificial Intelligence, vol. 3661. Springer-Verlag Berlin, 2005, pp. 329–343.

[6] A. Graesser, S. Lu, G. Jackson, H. Mitchell, M. Ventura, and M. Olney, A .and Louwerse, "Autotutor: A tutor with dialogue in natural language," *Behavior Research Methods Instruments & Computers*, vol. 36, no. 2, pp. 180–192, May 2004.

[7] R. Dickerson, K. Johnsen, A. Raij, B. Lok, J. Hernandez, and A. Stevens, "Evaluating a script-based approach for simulating patient-doctor interaction," in *Proceddings of International Conference on Human-Computer Interface Advances for Modeling and Simulating*, 2005, pp. 79–84.

[8] P. Kenny, T. D. Parsons, J. Gratch, A. Leuski, and A. A. Rizzo, "Virtual patients for clinical therapist skills training," in *IVA '07: Proceedings of the 7th international conference on Intelligent Virtual Agents*. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 197–210.

[9] J. Dombi, "Basic concepts for a theory of evaluaation - the aggregative operator," *European Journal of Operational Research*, vol. 10, no. 3, pp. 282–293, 1982.