

# MUVES: A Virtual Environment System for Medical Case Based Learning

Samir M. Abd El-razek<sup>1</sup>, Waeil Fathi Abd El-wahd<sup>2</sup>, Hazem M. EL-bakry<sup>1</sup>

<sup>1</sup>Dept. of Information Systems, Faculty of Computer Science and Information Systems, Mansoura University, EGYPT.

<sup>2</sup>Dept. of Operation Research, Faculty of Computer Science and Information Systems, Menofia University, EGYPT.

## Summary

Virtual simulations of patients have been an increasingly important complement of medical learning. Virtual patients are interactive computer systems that simulate real life clinical scenarios, which provide an effective way for medicine teachers and students to not only learn clinical topics, but also to practice in a virtual and safe environment. This paper presents the design and results of MUVES, a virtual environment for medical e-learning in medicine college of Mansoura. The system is logically designed to simulate case-based learning methodology, and technically designed upon MedBiquitous specifications for virtual patient. In this paper we present the concept of case-based learning and the need for virtual environment to support this methodology of learning. Then we introduced the current researches of virtual patients and MedBiquitous specifications which became a major standard for building virtual patients. Finally we presented an overview of workflow, data structure, and functional modules of MUVES.

## Key words:

*Virtual Environment, Virtual Patient, Case Based Learning, Medical E-Learning.*

## 1. Introduction

Recent studies of medical learning methodologies have a major interest for case-based learning (CBL). This methodology of learning depends on putting students in a semi-real situation and the objective is to learn how to react in this particular situation [1]. Medical cases are based on real scenarios and supporting data to evaluate an open ended problem. Finding a real patient to demonstrate all medical case would be a difficult job for teachers. Here comes the value of virtualization which enables the teacher to create a virtual case based on a predefined problem and known answer. Many researchers developed frameworks for virtualizing medical cases using virtual reality technologies, which gained some success factors for simulating patients, surgeries, and clinical tools. Virtual reality systems have some rich graphical features and 3D engines containing great numbers of graphical elements and complex processing and rendering which are not suitable for the current trends for using the web technology in e-learning. In the last few years new web-based models and frameworks have been developed to make use of the accessibility, scalability, and low cost for using web. In

this paper we introduce Mansoura University Virtual Environment System (MUVES), a new web based virtual environment system for medical CBL based on Virtual Patients (VPs). The use of virtual patient is increasing in healthcare education, partly in response to increasing demands on health care professionals and education of students but also because they allow opportunity for students to practice in a safe environment.

## 2. Virtual Patient: Current Researches

The European Commission is funding a major 3-years project named eViP [2], the project started in 2007 and is due to finish by end of 2010. The project aims to explore the feasibility of repurposing and enriching VP examples in a variety of different ways, and for different purposes [3]. The project is developed with cooperation with seven universities and they could implement common technical standards for all VPs in collaboration with MedBiquitous. MedBiquitous Virtual Patients Working Group has developed XML standards and Web services requirements to enable interoperability, accessibility and reusability of Web-based virtual patient learning content [4]. In 2009 Inga Hege from the University of Muenchen-Germany has implemented the MedBiquitous standards into a learning system named CASUS [5]. Another project has been developed in the University of Maryland-USA, Marjorie and Nirenburg has been working on a simulation and a training system the name it Maryland Virtual Patient (MVP). The system gives the user the ability to act as a physician and treats virtual patients with or without a virtual tutor [6]. One of the major projects that are working on VPs is CAMPUS. The CAMPUS virtual patient system, developed at the University of Heidelberg-Germany consists of different modules for learning and assessment with VPs [7]. Some projects has a big focus of authoring VPs like VPSIM that has been developed in the University of Pittsburg-USA on 2008. VPSIM creates an authoring environment for medical educators, that requires minimal training and stimulated creative case writing [8]. Also researchers in the University of McGill-Canada have developed an authoring system to create VPs can be used

in the simulation of physical examination, taking history, and ordering the required laboratory tests [9].

### 3. MedBiquitous Virtual Patient (MVP) Specifications

The MedBiquitous Consortium developed the XML-based MedBiquitous Virtual Patient (MVP) data specification. This specification enables the exchange and reuse of VPs across different systems. The following are the main components of the MVP architecture [5], shown in fig 1: The Virtual Patient Data (VPD) provides the personal and clinical data that is related to the clinical scenario being simulated. The VPD contains data elements and some kind of structure that represents the medical history, physical and technical examination and therapy. The Data Availability Model (DAM) specifies the relation between the VPD and Media Resources (MR) elements to be used in the Activity Model. VPD and MR elements can be used multiple times in the DAM nodes depending on the logical relation between these elements, such as patient history and test results. The Activity Model (AM) encodes what the learner can do and how he may interact with the virtual patient. By creating available paths through content using interconnected nodes and controlling how the user can follow them using a simple rule system, a great variety of patient activities are possible. The AM provides the context in which they are exposed to the learner. Media Resources (MR) like images, animations, videos and audio files associated with the virtual patient are referenced in the DAM or in the VPD elements. IMS Content Packaging is used to structure media resources within the MVP specification and provide unique identifier for each media resource [10].

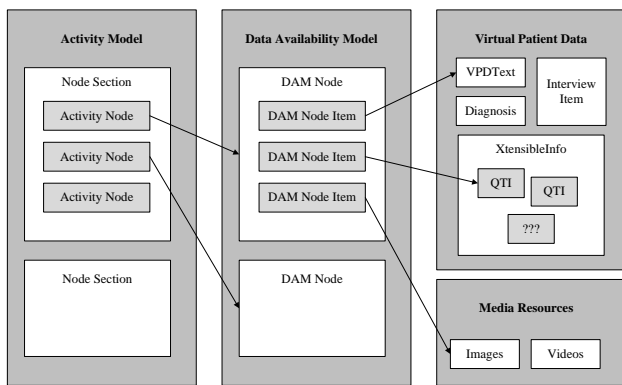


Fig. 1 MVP Specification Model

### 4. Overview of MUVES

MUVES is a medical e-learning system for authoring and delivering CBL to medicine students based on MVP standards. MUVES can work as helper tool for medicine teachers to demonstrate the diagnosis process for students over the web in a virtual lecture style presented in fig 2. Teachers can select or create a new VP template, these templates are categorized by teacher upon their relevance to subjects. Each VP template contains a complete VP data and media resources, the teacher can create an instance from this template to start performing the diagnosis process on this instance. The idea of creating templates is to “reuse” previously created VPs and make it easier for creating VPs rather than copying files. After creating the instance the teacher may need to change some VP data for more clarification on this case or to add new important information needed for diagnosis like up-normal values for some lab elements, or increase the measure of blood pressure, etc.

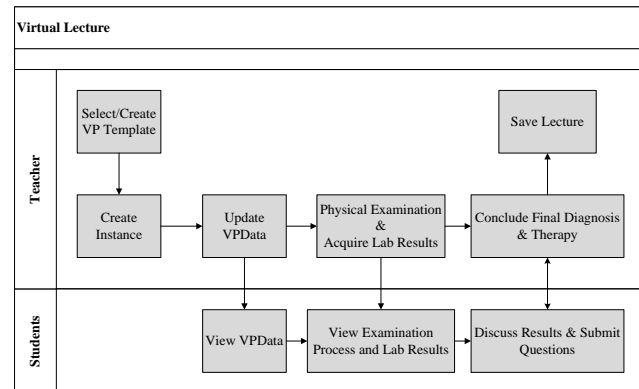


Fig. 2 Virtual Lecture Workflow

All the previous steps are performed “offline” or away of students. Once the teacher finishes his updates on the instance chosen, he can start the virtual lecture. The virtual lecture items are broadcasted to students over the web, starting with VP data and media resources. Teacher starts his lecture using collaboration tools like audio/video streaming, whiteboard, and performing visible physical examination on the VP instance. The physical examination is performed using a rich set of physician tools, and possibly to acquire some lab results for this VP instance. Each step of the diagnosis process can be opened for discussion by students using a simple chatting tool with the teacher until the teacher reach the conclusion of the diagnosis student can submit their questions and discuss the reasonability of diagnosis. The lecture can be saved to the Virtual Lecturers Library (VLL) and can be reviewed by student later. Another important feature of MUVES is it can be used as an assessment tool for students. Teacher can “setup” the instance as introduced previously, but he

will assign the diagnosis process to a group of students who need to collaborate together to conclude the possible diagnosis with guidance of the teacher.

### 5. MUVES Data Structure

In MUVES, VP’s XML data structure is based MedBiquitous VP specifications. This global data structure enables MUVES to integrate with any other VP authoring system by importing VP XML files. An XML package is a set of XML files that represents a single VP. This standardization required implementing two components for building VP XML packages, and another for validating XML files schema before loading in the virtual lecture player as shown in fig. 3.

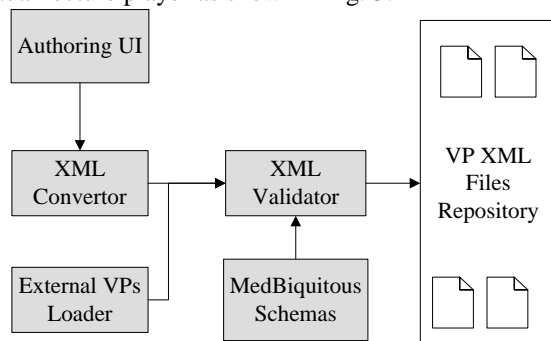


Fig. 3 VPs XML Handlers

VP data structure is defined by a root element named VirtualPatientData, which contains the clinical and demographic data for VPs in addition to the metadata. VirtualPatientData element must exist once in a VP package. VirtualPatientData has sub elements defining the data of a VP shown in fig 4, the existence of any sub element is optional [10].

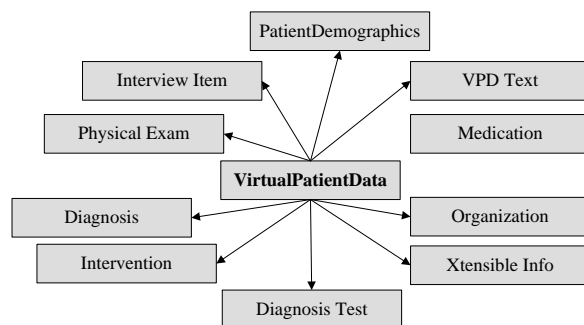


Fig. 4 VP data structure elements

### 5.1 Patient Demographics

The PatientDemographics element contains subelements that define the name, age, sex, and other demographic characteristics of a virtual patient and allows for the categorization or grouping of various demographic characteristics, see the example shown in fig. 5. It consists of two sub elements: Core Demographics which contains the regular demographics, and Demographics Characteristic which contains non-standard demographics.

```

    <PatientDemographics id="D009">
      <CoreDemographics id="C1003">
        <PatientID> P542 </PatientID>
        <Name> Adam Ali </Name>
        <Age> 50 </Age>
        <Gender> Male </Gender>
      </CoreDemographics>
      <DemographicCharacteristic id="154B">
        <Title> Marital Status </Title>
        <Description> Single </Description>
      </DemographicCharacteristic>
    </PatientDemographics>
  
```

Fig. 5 Patient Demographics Example

### 5.4 Interview Item

It describe a single question and response that is part of a clinical history, see the example shown in fig 6.

```

    <InterviewItem id="1">
      <Question>How are you feeling?</Question>
      <Response>Not very well.</Response>
    </InterviewItem>
  
```

Fig. 6 Interview Item Example

### 5.2 VPD Text

It provides narrative or other descriptive text that is part of the virtual patient data. It contains A unique identifier for this text that can be used by other virtual patient components to control how the text is used in an educational activity, also it contains an indicator of the function this text serves in the virtual patient. Valid values are: complaint, history, or problem. VPD Text could contain an xhtml div element that can be used to format how text will be presented to the user.

### 5.3 Medication

Medication describes a medication taken by the virtual patient in detail, including the medication name, dose, route, and frequency.

Medication name defines the name of the medication, possibly referencing a medical taxonomy or vocabulary. Dose defines the dosage of the medication, for example:

10 mg. Route defines the route of administration for the medication, for example: oral.

### 5.5 Physical Exam

It describes a single physical exam and the findings of that exam. It consists of sub elements like exam name, location on body, actions the virtual clinician takes to perform the exam, findings of the exam for example: normal heartbeat, and the description of the physical exam in more details.

Location on body is defined by sub elements like which part of body, whether it's proximal or distal, right or left, front or back, inferior or superior.

### 5.6 Diagnostic Test

Diagnostic test defines test result, it consists of test name, unit to measure this test, result, and the normal value of the test.

### 5.7 Diagnosis

Diagnosis defines either a single differential diagnosis or a single final diagnosis intended by the virtual patient author. It contains the likelihood of the differential diagnosis being correct. Valid values are: high, medium, low, none.

## 6. MUVES Modules

MUVES is composed of main four functional modules: configuration module, learning management module, VP authoring module, and virtual lecture module. MUVES supports three types of users, with different authorization levels: student, teacher, and administrator.

### 6.1 Configuration module

The administrator module manages users, permissions, and lookups. System administrator is responsible for managing users of the system and defining their permissions upon their types.

### 6.2 Learning management module

Teachers can access to VPs templates databases and they can create user specific virtual cases. Each teacher has his own set of virtual cases and then he can use these cases in virtual lectures or assessments. Teachers also can create new assessments and select students who will perform this virtual assessment. Through this module teachers can categorize virtual lectures into subjects and can view reports related to each subject like number of virtual lectures, or assessments reports.

### 6.3 VP authoring Module

This module provides a tool for building VPs with a simple user interface for teachers who does not need to know about the structure of VP, without help of multimedia developers or designers. Most of VPs created on MUVES are built using VP templates which were developed with assistance of Mansoura Medicine Faculty teachers who provided the initial information needed for each case. Based on the MVP specification model, templates contain VP data like text, media resources, diagnosis, some questions and answers. This information must lead to a specific medical conclusion about this VP. All templates are based on the normal template that presents the normal conditions and data for a normal human. Then teachers use these templates to define their own virtual cases by updating some information of the template that could makes it easier for students to perform the diagnosis or by making it harder in assessments. Teachers can update VP data from a usable web interface shown in fig. 7. VP data in authoring module are categorized into general information, patient history, physical examination, lab tests, diagnosis, and therapy.

Fig. 7 VP authoring Module

Teachers use these categories for submitting VP data, which consists mainly of text, boolean, images, and different multimedia formats. History information, physical examination, and lab tests are considered important to solve the case can be marked as required. The amount of time needed to create a single case varies and may depend on the complexity of the case.

### 6.4 Virtual lecture module

This module is main core of MUVES. It manages the CBL concept of putting users in a virtual situation, and starts to record and analyze their activities for solving this situation. The main components of this module are shown in fig. 8.

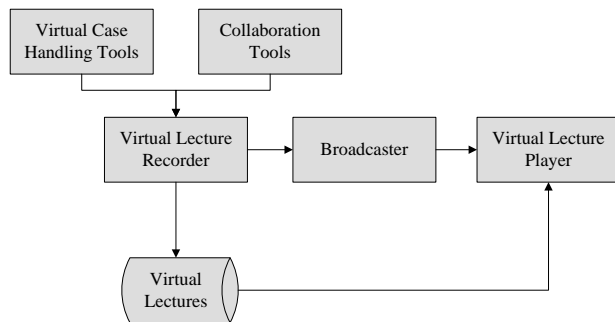


Fig. 8 Virtual lecture module components

The virtual case handling tools are an important component of this module, it manages the VP data and the virtual activities like diagnosis and lab tests. An example of the virtual activities is shown in fig. 9.

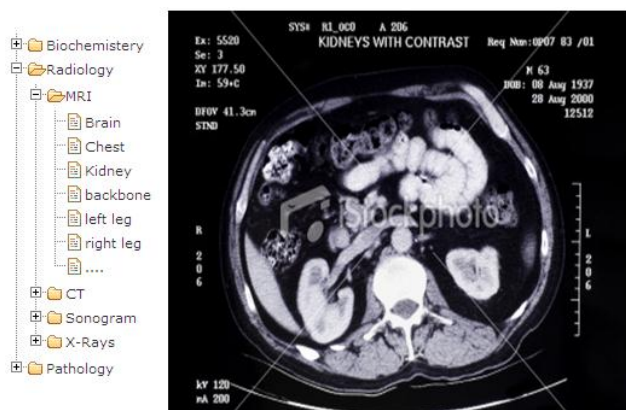


Fig. 9 Virtual Activities

Some VP data are visible to users of MUVES, others are hidden waiting for virtual activities to be performed to reveal these hidden information about the virtual case. In e-learning context collaboration tools are very important to help teachers and students to share knowledge and for open discussions related to virtual cases like discussion boards, whiteboard, and chatting.

Both virtual activities and collaboration activities are recorded using the virtual lecture recorder as a one unit like a video recording for a real lecture. Students can view any recorded virtual lectures from the virtual lectures database. Live virtual lectures are broadcasted to students and viewed using the virtual lecture player, which is a web-based interface that presents virtual case properties, virtual activities, and collaboration activities.

MUVES provides another feature which is virtual assessment. In virtual assessments students plays the physician role, and they conclude the final results and therapy plan. Then teacher evaluate these results like the real assessment scenario. Virtual assessment is built on the same components of this module.

## 7. Conclusion

MUVES consists of different modules for learning and assessment with virtual patients based on case-based learning methodology. An easy to use authoring system enables the user to create virtual patients based on a defined structure of virtual patient data like patient demographics, interviews, diagnosis, and medications. Building the system on MedBiquitous specifications enables MUVES to integrate with any other system built on the same specifications like eViP by importing and exporting virtual patients XML files. Medicine teachers can use MUVES to practice virtual diagnosis and therapy in a virtual lecture, or assess students online. Students can communicate with the lecturer by a rich collaboration tools like whiteboard, and discussion boards. Virtual lectures can be recorded containing virtual case, virtual activities, students' questions and answers.

## References

- [1] Samawal Hakim, "Learning methods and its application in medical education", 2007
- [2] <http://www.virtualpatients.eu/about/about-evip> , Last Accessed on July 20, 2010
- [3] eViP Project Team, "Annual Report" , 2009
- [4] <http://www.medbiq.org> , Last Accessed on July 29, 2010
- [5] Inga Hege, A. Kononowicz, M. Pfahler, "Implementation of the MedBiquitous Standards into The Learning System CASUS" , 2009
- [6] Marjorie McShane, Sergei Nirenburg, "Maryland Virtual Patient: A Knowledge-Based Language-Enabled Simulation and Training System", 2009
- [7] Jorn Heid, Benjamin Hanbeck, Frank Hess, "Demonstration of the CAMPUS Virtual Patient System", 2009
- [8] James Mcgee, "VPSIM – A Standards-Based Virtual Patient Authoring System", 2009
- [9] David Fleiszer, Nancy Posel, "Open Virtual Patient Authoring System From McGill University", 2010.
- [10] Valerie Smothers, Ben Azan, Rachel Ellaway, "MedBiquitous Virtual Patient Specifications and Description Document", 2010