MEDINFO 98
B. Cesnik et al. (Eds)
Amsterdam: IOS Press

Virtual Patients for a Virtual Hospital

E Loke, KC Lun

Medical Informatics Programme
National University of Singapore
Republic of Singapore

Abstract:

With the introduction of VBscript & Active-X it is now possible to construct interactive websites with ease. This paper discusses the author's experience in setting up TWO different 'VIRTUAL PATIENT' websites. One such site, named VIRTUAL PATIENT 97, is based upon a goal based scenario, in this case the clinical management of common clinical problems like upper gastro-intestinal bleeding in a surgical ward. The second website, CLICK 'n' LEARN, is essentially a health education website providing material on common chronic illnesses like diabetes, hypertension and asthma. The first module deals with aspects of self care in diabetes. Included are textual as well as video clips which show various elements of diabetes self care: like the administration of insulin or the usage of a glucometer. An added twist is a target oriented interface borrowed from the TAMAGOTCHI. If the user dutifully looks after his virtual diabetic, it will thrive & the disease will be well controlled, else the virtual patient deteriorates in accelerated time. This allows for safe experimentation & promotes awareness of the importance of self care in the total management of chronic illnesses.

Keywords

VBscript, Microsoft Internet Explorer, Active-X, Goal based scenario, Bleeding GIT, TAMAGOTCHI, Self care, Health education

Introduction

It is probably safe to say that the Internet , in particular, the World Wide Web (WWW), is the light bulb of our time. With it's exponential rise in popularity over the last half decade, the WWW has touched almost all aspects of society as we know it. Not to be left out, the ancient art & science of clinical medicine has also, albeit slowly & cautiously, accepted its presence [1]. As literacy rates rise throughout the world, there is a tendency now, to not only share knowledge between medical professionals [2], but also to provide members of the public & patients with clear, concise & well written health related information on the WWW. Certainly, our patients now would ALSO like to know more about their disease states.

With this in mind, we set out to construct TWO virtual patient modules: the first: named Kkanjya – is targeted at the general public & specifically diabetics who want up to date health information, the second called Virtual Patient 97 is a goal based module which presents the medical undergraduate with clinical scenarios found on the general ward for him to manage – the first being a case of peptic ulceration with haematemsis.

Also in the last year or so, the Tamagotchi virtual pet from the Japanese toy maker giant - Bandai, has been a tremendous hit throughout the developed world, with people flocking to vendors to purchase their copy of the toy. The concept was simple: feed the pet, play with it, see to its other needs & it will thrive in its electronic home. While the psychosocial analysis of its success can easily fill a thesis, it is enough to say that users probably liked it because it had a feedback loop which allowed it to respond to the actions, or 'in-actions' of the 'owner'. We quickly realised that this model could very well be applied to teaching patients or care givers the importance of self care in chronic ailments like asthma & diabetes mellitus. With this in mind, we set out to create a user friendly, informative & current multimedia health education website on diabetes based upon the Tamagotchi concept.

Objectives And Aims

Formerly the objectives for Kkanjya are:

1. To construct a multimedia website containing health information on self care methods for chronic disease ailments - starting off with diabetes mellitus, which already has a large database on the WWW [5]
2. To provide concise, well written & current information,
3. To construct a module based on the Tamagotchi concept which would allow the user to safely experiment with neglecting the virtual patient.

The objectives for Virtual Patient 97 are:

1. To provide the medical undergraduate with, as accurate as possible, a goal based scenario module on common clinical scenarios found on the general ward
2. To construct a consistent, user friendly interface yet preserving as much of the look & feel of the original documents & forms used on the general wards for procedures & investigations

3. To ensure the logic engine contains up to date data on clinical management of these problems while staying clear of controversial points

Methods

The first project was named "Kkanjya - Click & Learn" the abbreviated form of "Kasou Kanjya" which is Japanese for "Virtual Patient" and also to highlight an important rule in developing the interface which relied as far as possible on mouse inputs only. Thus giving it a point & click feel consistent with most popular Graphics User Interfaces (GUI's).

Virtual Patient 97 was designed with the medical undergraduate in mind [3,4], & the aim was to recreate as accurately as possible, the feel & where possible the look of the documents used on the general ward when clerking real life patients.

Both Kkanjya & Virtual Patient 97 were developed on an Intel Pentium-166 PC running Windows 95. The files however were stored on a server platform conforming to the popular UNIX standard. Due to the fact that Kkanjya will be implemented as an Internet/Intranet browser application, it inherently is cross-platform, but browser specific.

Recent Internet Developments

In recent years, there has been a shift from the traditional static web-client-server model (FIG 1) to a more dynamic & interactive one. The server had to do advanced functions like validating forms or information if there was a need to as the client was unable to do so. With the introduction of client side plug-ins, software developers were able to supplement the deficiencies within HTML with their 'applets' running directly on top of the basic HTML structure (FIG 2). Netscape was one of the first few companies to implement this in as early as version 2 of Navigator.

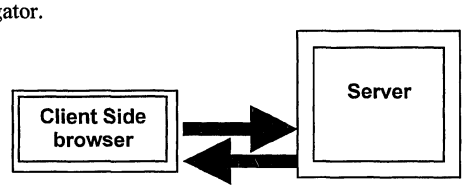


Figure 1 - Static model of client side browser making a HTTP request from a server, with the server sending the required packets back to the browser for display.

But credit has to go to Microsoft, who was dormant for quite a while on the Internet scene, for introducing a complete, comprehensive & bold strategy to revolutionalise interactive application development on the WWW. They introduced Active X objects with client side scripting language acting as 'glue' between these new 'plug-in's & HTML.

Microsoft's Internet Explorer (MSIE) 3.0 was a tremendous improvement over their initial offering of MSIE 1.0 & 2.0. The major improvements were the implementation of the above mentioned support for Active-X, & two client side scripting laguanges: namely JAVAScript & VBScript (Visual Basic Scripting Edition). The latter is a down sized version ('lite' version as it were) of their popular Visual Basic.

This was immediately appealing as there are numerous Visual Basic Programmers lurking out their, not to mention the fact that VB is definitely easier to learn, interpret & program in than JAVA or even JAVAScript which still retains some of the nuances of its big brother- JAVA.

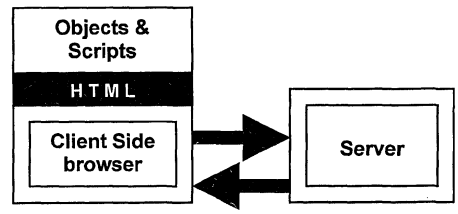
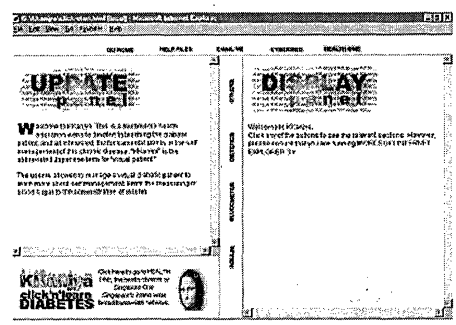


Figure 2 - Showing the improved model, with the basic request still being made via HTTP, but client side now has more tools to perform tasks to free the server & hence prevent cluttering of band width. Note that scripts & objects sit over HTML.

Kkanjya Design

The screen shot shows the main page of the module: with a general toolbar across the top frame, two text areas on the left & right flanking a menu which contains click-able animated links to sub modules within Kkanjya. A info cum status bar is included in the bottom left-hand corner.



It is immediately apparent that Kkanjya runs with something else in the background rather than just plain HTML, as you would note that the various element on the web page responds to mouse events, such as: positioning the mouse over the click-able areas will result in that particular link 'lighting-up' & a short summary of the target page is given in the status box in the bottom left hand corner. This is the result of integration between Active X button objects, VBScript & HTML.

It is this level of interactivity & programming that allows for the existance of Kkanjya's logic engine. You would also notice that there is a mini-picture of the Mona-Lisa within the status box. That is used as an indicator of the virtual diabetic patient's well-being: A smiling Mona-Lisa indicates all is well & the user is doing fine, A non-commital look from her indicates just that, A sour look means the user is doing something grossly wrong with a potential for disaster. The logic engine scores the user based on a few categories:

1. Insulin: the module will intermittently prompt the user when it is time to administer insulin to the virtual diabetic. (running on accelerated time: such that a typical day is covered within 15 minutes) The user then has to give not only the right type but the right amount corresponding to the regime the virtual patient is on.
2. Glucometer readings: depending on the dietary caloric intake, the adherence to or the non-adherence to the prescribed insulin regime, & other factors: like stress of an acute illness. It is to be noted that there is no hard & fast way to calculate the exact glucometer reading based on the categories mentioned & indeed, the logic engine in no way attempts to simulate a real life scenario, but rather, this is attempt to provide an approximate representation of the likely glucometer readings in the above mentioned setting.
3. Dietary caloric intake: the user is allowed to choose from a fixed menu with the corresponding amount of calories being shown directly under the picture of the dish. Attempts have been made to include a wide spread of culinary experiences to cater to the lcoal as well as cross-cultural tastes of the potential users

Figure 3 - Screen shot showing video playing back. Controls are located below the screen.

The other big attraction of Kkanjya is the audio visual section of the health education section. There are videos showing common issues in the self care management of diabetes, like how to use common glucometers, how to inject insulin & how to use some of the latest aids in therapy: like Novo Nordisk's NOVOPEN controlled insulin injection device. This was made possible with the introduction of video streaming technology, which removed the biggest hinderance to using videos on WWW pages – THE LONG WAIT FOR THE VIDEO TO DOWNLOAD first. Streaming ensures nearly immediate playback of the video as the rest of it streams in while the earlier segments are played back from the buffer. We used the VIVOACTIVE video streaming technology.Virtual Patient 97 Design

Again the screen shots show a "Frames" approach. The toolbar at the top of the page hold key links to important or relevant medical sites for further reference, The right sided frame is used to display information or test results, the upper window on the left side shows the virtual patient's up to date progress or clinical condition (based on the commonly used SOAP frame work: SUBJECTIVE, OBJECTIVE, ASSESSMENT, PLAN). It also contains Active-X buttons to allow the medical student to either order investigations & procedures or to fine tune the patient's management. There is also a prominent statusbar at the bottom of the screen to display the patient's vital signs & other text based information relevant to the simulation.

Figure 4 - Screen shot showing main page of Virtual Patient 97 (See Text for description)

Figure 5 - Screen shot showing the INVESTIGATIONS page

The respective test is ordered by clicking

On the buttons on the left frame, the friendly laboratory assistant will then display the test

Results on the right side. An added feature to aid fast data interpretation is a graphical

representation of the value obtained for a given parameter, plotted against the accepted normogram. (Can be seen on right side of screen shot: * indicate the test value in relation to the normogram marked by the '[' and ']' brackets indicating the lower & upper limits respectively.)

The logic engine also makes sure that basic investigations are done before allowing the student to proceed further. For example, in the first case, of a man with haematemesis, a full blood count, looking specifically at the haemoglobin & haematocrit would be mandatory. The student would be prompted to do the test if he had not ordered it in the first instance. However, where there is a point to be learnt from making omissions, the student would be allowed to see the end result of his decision. An example is: the same patient should have been kept NIL PER ORAL & an intravenous access with a drip secured with appropriate intravenous empiral anti-ulcer therapy instituted as bleed-

ing ulcers would be the most likely cause given the clinical scenario. However, if he chooses to not restrict the patient's oral intake & not secure good intravenous access & not institute appropriate therapy, needless to say, if the patient bleeds again, the outcome could very well be dire. The logic engine has allowances like these to enable it to be as accurate a simulation as possible.

Extensive literature reviews were done on the subject of bleeding ulcers & all the information was carefully perused with a specialist surgeon before the actual logic engine was coded. However, due the constraints of programming flexibility, there were instances where accuracy & dynamic simulation had to be replaced by hard coding.

Future plans

With the introduction of server side Active objects & scripting, more interactivity can be implemented. It would be possible for instance, to keep track of a user's profile & have a more true to life simulation, even over real time, by storing or updating the user's progress within a file on the server. This is an achievement because plain HTML is stateless. There are also plans to introduce a more time critical approach to Virtual Patient 97, by keeping track of how quickly (or slowly) the medical student responds to a critical situation & to then model this in the overall outcome of the simulation. All in all, with new technologies streaming in, WWW applications might one day have the feel & look of a desktop application in terms of interactivity & the ability to process & control user responses. Gone are the days where interactivity with HTML WWW documents is limited to the <A HREF => command.

References

[1] E Loke; *KKanjya -Click 'n Learn Diabetes* http://ch.nus.sg/cybermed/kkanjya/click-start.html

[2] *Virtual Patient '97* E Loke; http://www.health1.nus.edu.sg/

[3] Kruper JA; Lavenant MG; Maskay MH; Jones TM: Building Internet accessible medical education software using the World Wide Web. *Proc Annu Symp Comput Appl Med Care*, ():32-6 1994

[4] Chi-Lum BI; Lundberg GD; Silberg WM: Physicians accessing the Internet, the PAI Project. An educational initiative [editorial] ; *JAMA*, 275(17):1361-2 1996 May 1

[5] Dowd SB; Laws R : The Internet's role in education. *Radiol Technol*, 67(1):83-6 1995 Sep-Oct

[6] Chodorow S; Educators must take the electronic revolution seriously. *Acad Med*, 71(3):221-6 1996 Mar

[7] Lewis D; Komondor K Diabetes Educ, Diabetes on the Internet: a useful resource for educators. 22(5):503-4, 506 1996 Sep-Oct

Address for correpondance

eugene@loke.net

coflunkc@nus.sg