Healthcare Applications based on Networked Agents

Christian Mazzi[1], Pronab Ganguly[2] & Pradeep Ray[3]

[1]*Faculty of Medicine*
The University of Sydney
NSW 2006, Australia

[2,3]*School of Information Systems, Technology and Management,*
The University of New South Wales, Kensington,
NSW 2052, Australia

[1]*Tel: 61 2 404 149326, Fax: 61 2 9351 3196; Email: cmazzi@gmp.usyd.edu.au*
[2]*Tel: 61 2 9385 5803, Fax: 61 2 9662 4061; Email: p.ganguly@unsw.edu.au*
[3]*Tel: 61 2 9385 5890, Fax: 61 2 9662 4061; Email: p.ray@unsw.edu.au*

Abstract

Healthcare has been recognized as one of the most important areas for networked enterprise applications and services. Presently, information technologists and systems engineers all over the world are working towards achieving better efficiency and quality of service in various sectors of healthcare, such as telemedicine, hospital management, patient-care, and treatment. This paper addresses the issue of more effective and efficient handling of doctor-patient relationship. The idea is to use Software Agent technology to extend the patient-doctor relationship beyond the physical and logistical limitation of face-to-face consultations in the management of diabetes. Software agents can provide an extension of the doctor by interacting with the patient via a computer and the Internet. The agent is visualized as an anthropomorphic figure to further enhance the patient interaction.

Keywords: Healthcare, agents, doctor-patient relationship, UML use cases, workflow, chronic disease management, patient monitoring.

1. Introduction

During the past few years the enterprise application scenario in the health care industry has dramatically changed from a sparse use of mainframes to a massive use of low-cost PC and networking technologies. This resulted in a need for the decentralization of the computational capabilities and a deep modification of the systems architecture [LAN, 99]. These advances include sophisticated AI applications including agents to achieve the interoperability of diverse systems participating in the healthcare workflow.

The therapeutic value of the patient-doctor relationship has long been recognized. The creation of an integrated healthcare workflow must include the patient. Overwhelmingly, technology in medicine is Physician-focused, often overlooking the crucial patient interface. Patient involvement in the healthcare process is very important to any attempt to improve health care quality and patient satisfaction [KIM, 97]. This has been associated with several desirable outcomes, including greater satisfaction, increased adherence to treatment, and positive treatment outcomes [TEN, 00].

Unlike many other networked enterprise application areas, healthcare involves wide semantic differences due to the diversity of terminology used by different interacting disciplines (e.g. nursing and radiology). This has led to more than twenty different types of standards bodies in this area. Hence effective solution to healthcare IT problems needs innovative and sophisticated techniques. In recent years, a new computing paradigm has been developed, in which software programs can be seen as autonomous social and reactive components called Agents, which are able to act in a specific environment in order to fulfill their design specifications. The modules respond to external changes, and communicate when needed with other Agents using an agreed language [DEL, 99]. This approach helps design complex systems, where different intercommunicating functionalities are implemented using dynamic and distributed components. These modules can incorporate legacy systems and provide a common communication framework for multiple-system integration.

This paper proposes the application of agent technologies to the integration of the medical treatment process illustrated through a typical doctor-patient relationship. The paper starts with a workflow analysis of healthcare services. This is followed by a description of the role of an agent as a personal assistant for a doctor and a patient. Thereafter, the paper illustrates an agent- based patient treatment process, using a prototype developed by us using Microsoft agent technology. This is followed by a technical discussion of different types of agent functionalities for healthcare services, using UML use cases. Finally, we conclude with some observations on the future of Internet based healthcare incorporating agents.

2. The Healthcare Cycle

A careful examination of the healthcare process for non-chronic patients reveals that it follows a natural cycle (henceforth "Healthcare Cycle") as illustrated in Figure 1.

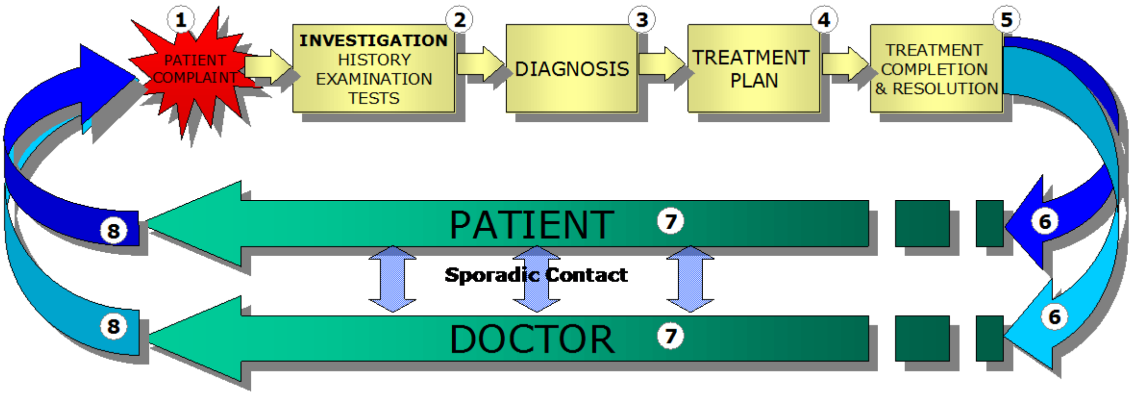


Figure 1: THE HEALTHCARE CYCLE

1. Patient and doctor are brought together as a result of a patient complaint
2. The doctor performs an investigation on the patient's condition. This may require physical examination, test and specialist consultations
3. The doctor diagnoses the condition. Specialist collaboration and negotiation may be required.
4. The doctor collaborates with specialists and the patient to design a treatment plan
5. The treatment is carried to completion and the patient condition returns to normal.
6. The interaction concludes. Patient and doctor separate
7. Occasional patient-doctor interaction:
   - Patient seeks information from doctor.
   - Doctor checks up on patient.
8. Until the next complaint when the cycle is repeated.

The examination of Healthcare Technology reveals that current software applications are dedicated to only one activity of the Healthcare Cycle and that this functionality is not integrated with other systems [LAN, 99]. This is most likely due to the inverse relationship between the performance of an Information System and the size of its domain space [LAN,99]. The result is a collection of isolated islands of excellence.

Further examination reveals that most Healthcare Technology is dedicated to the resolution of a patient's complaint (activities 1-5) of the Healthcare Cycle [COI, 97]. There is a lack of software that fulfills the patient's need for medical information and the doctor's desire to monitor patients (activities 6-8). The Agent paradigm is a potential solution to this issue.

This paper presents agent-based personal assistants that fulfill both the patient and doctor's needs. The next section provides a discussion of the agent-based personal assistants. This is followed by an illustration of doctor-patient relationship during a treatment process, using a prototype agent system in Section 5.

3. Agent-Based Personal Assistant

Abramovich and Shwartz propose a model for doctor-patient interaction that consists of three stages: 1. An initial Personal Meeting stage 2. An Examination stage 3. Integration through Dialogue stage. [ABR, 96]. Of particular interest is the third stage which involves the integration of medical information into the ongoing dialogue between doctor and patient.

Patients value most the information provided to them by their doctor and in today's society individuals will have to take greater responsibility for their healthcare. Patients will actively demand more information from their doctors [GRI, 96]. A study of 1276 Norwegian doctors indicated that the "informed patient' is becoming an integral part of their regular workday. Three out of four doctors had experience with patients bringing internet information to the consultation setting. Most of these doctors found this natural and unobtrusive [HJO, 99].

This exchange of information is a crucial element in consolidating the therapeutic doctor-patient relationship. Agent technology can be applied to facilitating this stage as an extension of the doctor. Psychological studies have determined that individuals attribute anthropomorphic characteristics to the machines they interact with [TEN, 99]. This natural behavior can be leveraged by doctors when dispatching software Agents to their patients.

The functionality encapsulated by the agent need to include:

1) The ability to discourse with the patient using natural language.
2) The ability to remind the patient to follow the treatment course prescribed by the doctor.
3) The ability to traverse a triage algorithm to determine if the patient needs immediate care.
4) The ability to answer the patient's questions by retrieving high quality information from the Internet and presenting it.
5) The ability to notify the doctor, and retrieve all relevant information about the pathology for the patient

This presents the following benefits:

- The patient-doctor relationship is extended
- Proactive healthcare with emphasis on prevention
- Patients have access to high quality medical information
- Patients are monitored at home or on the road.
- Patients often do not know if their symptoms are serious enough to see a doctor. Agents can help to identify those patients who really need medical attention from those that only need information
- Better treatment completion

3. Illustration of Functionality \based on a prototype agent system

Diabetes affects an estimated 13 million people in the United States and with an annual mortality rate of 54,000 , it is the seventh leading cause of death [COT, 99]. Type 2 diabetes may account for about 80% to 90% of all diagnosed cases of diabetes [PAR, 97]. Treatment of this type of diabetes includes [TAT, 90]:

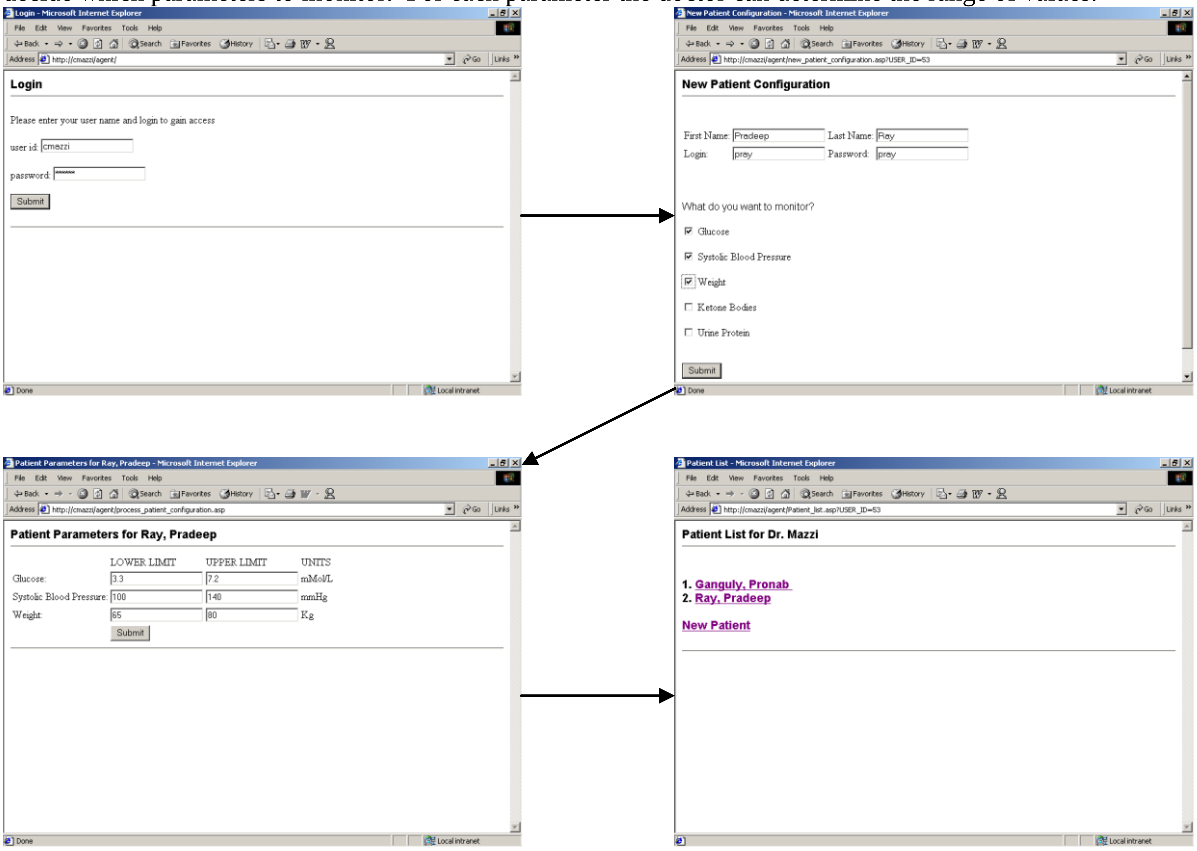
- Diet control
- Exercise
- Home blood glucose monitoring
- Oral medication/ insulin injections

It may be noted that there is presently no cure for diabetes – it needs to be managed. The care of diabetes is based on self-management by the patient, who is helped and advised by those with specialized knowledge [COT, 99].

We have constructed a functional prototype based on the diabetic patient to determine the functional requirements of such a system. Screen shots from the prototype will be used to illustrate a scenario.

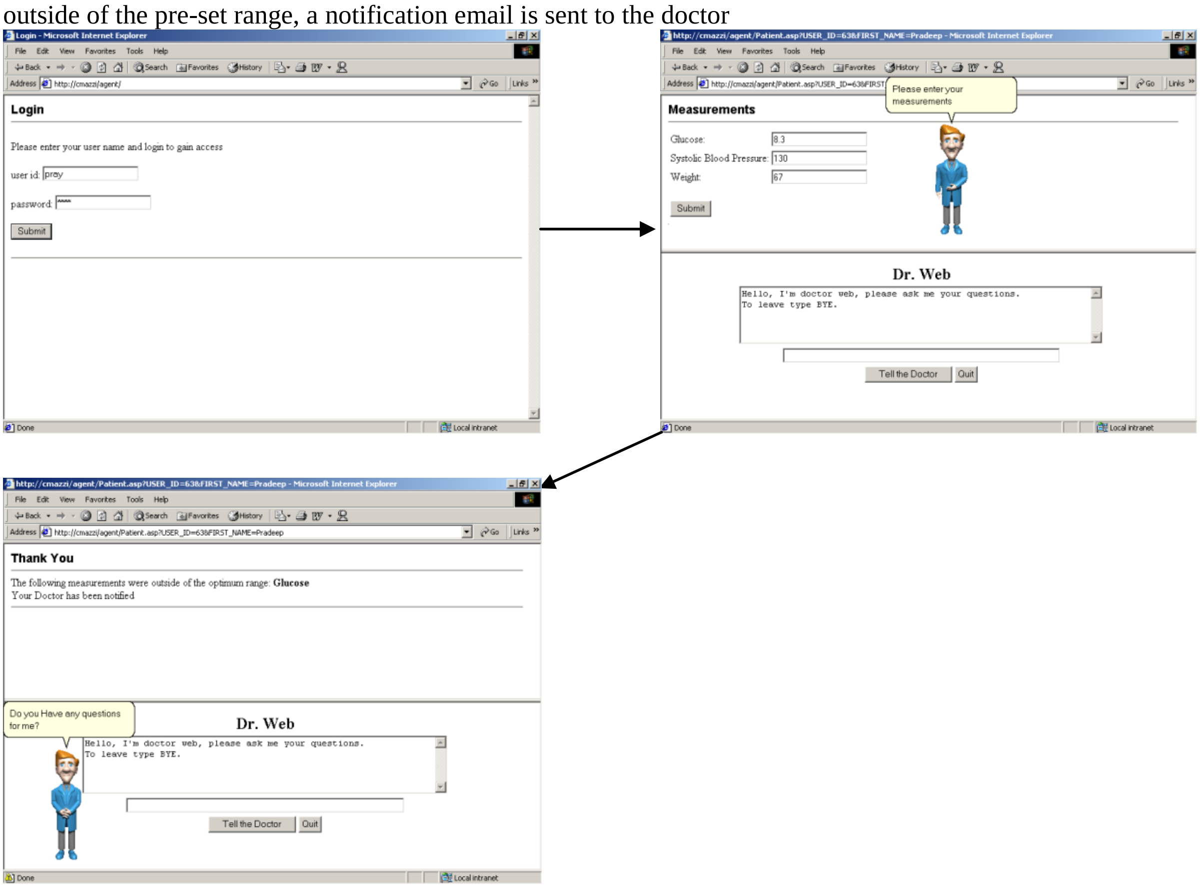
Scenario 1: New patient set-up by Doctor

The doctor has a security profile that allows him to add/modify/delete patients. For each patient the doctor can decide which parameters to monitor. For each parameter the doctor can determine the range of values.

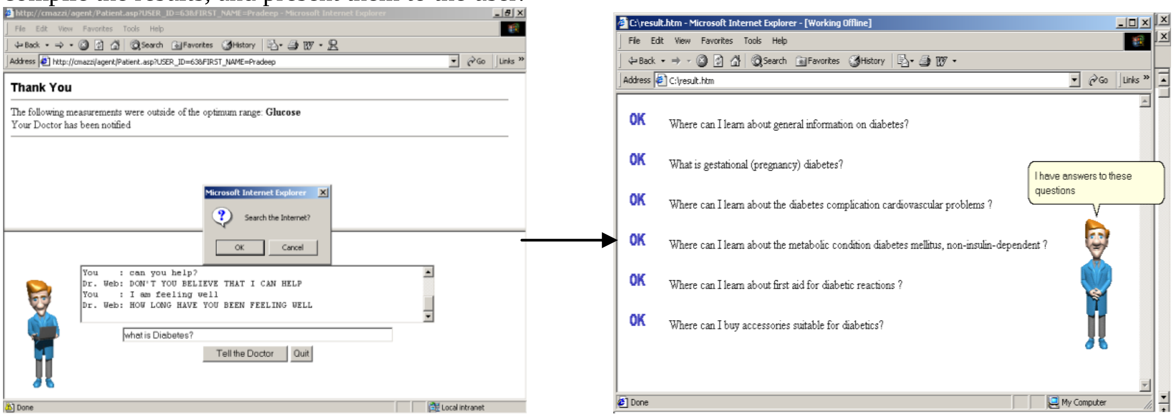


Scenario 2: Patient Data Entry

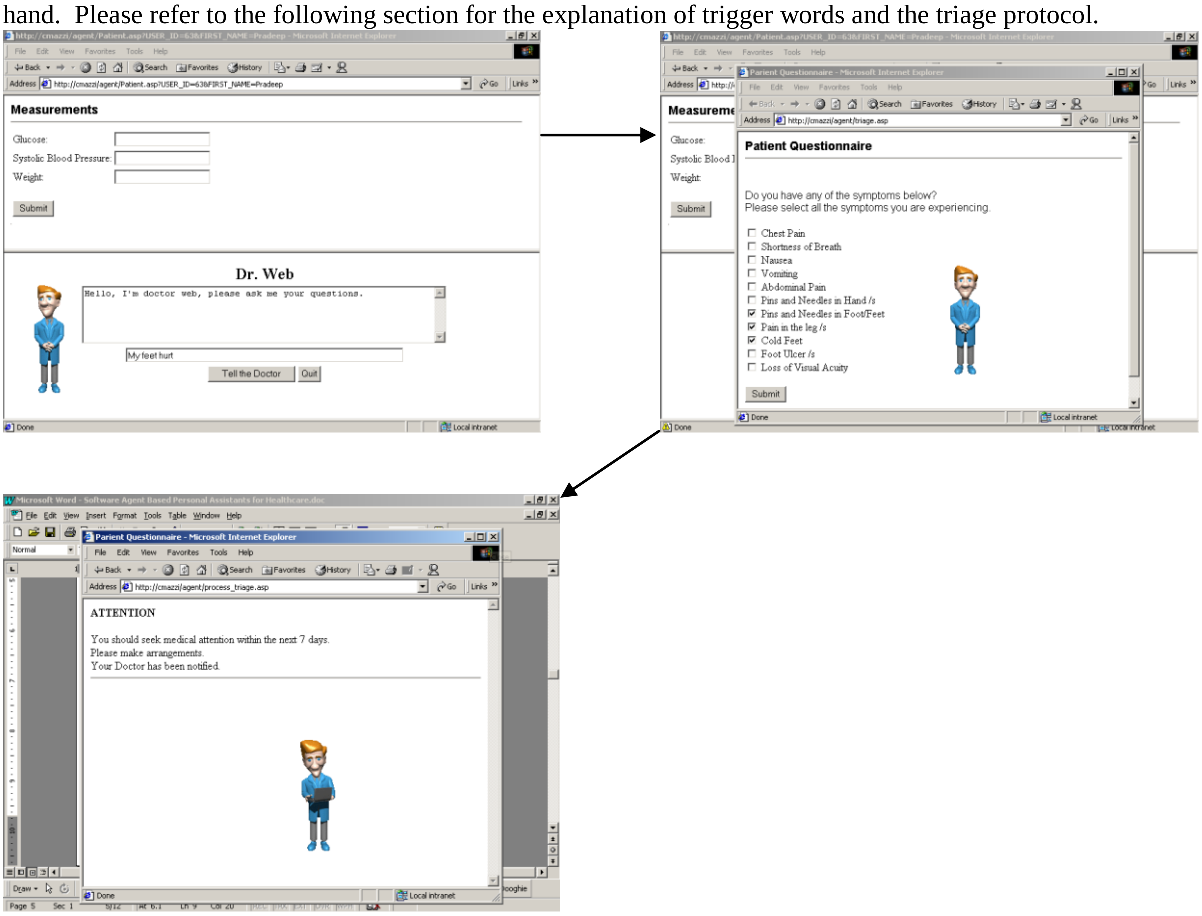
After the patient logs in, the agent will request for the measurements to be entered. If the measurements fall outside of the pre-set range, a notification email is sent to the doctor



Scenario 3. Agent-patient dialogue and information search by agent
The agent has the ability to dialogue with the patient using natural language. When a question is detected, the agent will enquire if the patient wishes to search for information on the internet. The agent will perform, compile the results, and present them to the user.



Scenario 4. The patient makes a statement with a trigger word (e.g."Hurt")
If the patient uses a trigger word, the agent will initiate a triage protocol to determine if there is an emergency at hand. Please refer to the following section for the explanation of trigger words and the triage protocol.



The scenario above has several implications for Agents in medicine. Firstly, the software Agent is introduced as part of the management. This facilitates its acceptance and use by the patient. Secondly, the Agent provides patients with the medical information they demand. This results in patient-Agent interaction that may extend beyond the initial treatment thus consolidating the doctor-patient relationship.

The patient-Agent interaction presents an important opportunity for the use of Artificial Intelligence (AI) encapsulated in the Software Agent. Patients will interact with their personal Agents through a natural language interface. This allows the patient to ask questions or present medical complaints as if they were interacting with the doctor.

During its interaction with the patient, the Agent will be vigilant for any suspicious patient behavior. A patient's input is considered suspicious if it is indicative of a potentially threatening medical condition. This is likely to be detected as trigger words such as "pain" or "weakness" in the patient's input. Trigger word detection is a simple but robust approach.

If a trigger word is detected, the Agent will initiate a triage procedure to determine to urgency of the patient's condition. The triage algorithm is based of a series of criteria that determine the urgency level from 1 to 5. Level 1 requires immediate attention (within minutes) and level 5 does not require any urgency. Triage algorithms are used in hospitals all over the world and this prototype uses the "Diabetes Triage Algorithm" as published in "Emergency Triage". 1999. BMJ. (Figure 3)

Figure 2 is a flowchart that illustrates the process for patient-agent interaction.

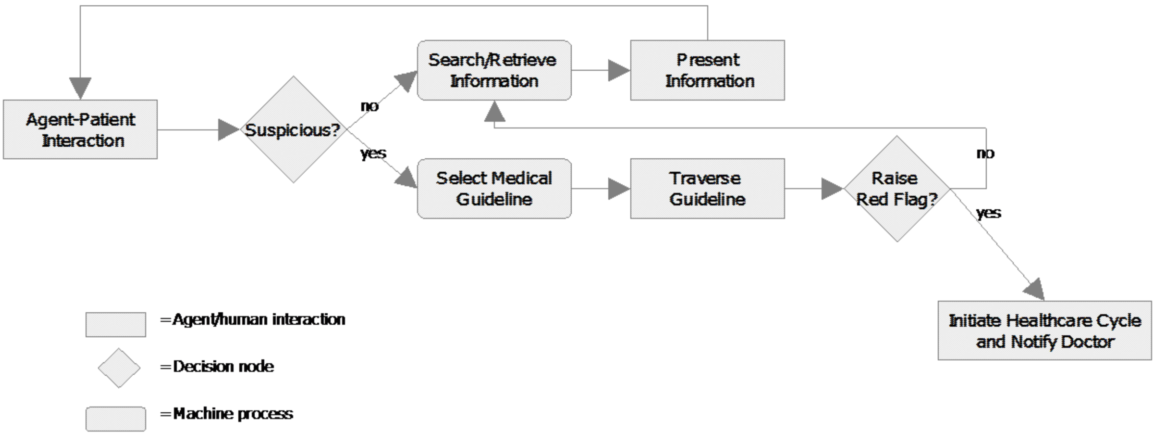


Figure 2. Patient-Agent interaction flowchart

5. A High-Level Technical Description

There are several methodologies that can be used for the design of an information system. These methodologies differ on the approach they take to the system's implementation: data –oriented (Information Engineering), function oriented (Structured System Analysis & Design Methodology (SSADM)) and object oriented (Unified Modeling Language (UML) techniques are available [GAN, 00]. For the purpose of designing Agents, UML is the ideal technique due to the closeness of the Agent paradigm to the object-oriented paradigm. Namely, the encapsulation of discrete behaviors and functionality into software components that can interact with other components via their interfaces. Not surprisingly, most Agent technologies are object-oriented. This Section presents (using UML use cases). The functionality of different types of agents required in a patient-treatment process.

The key components of an UML-based system design are:

- Use Case Model - Use Cases define the functionality requirements of the system. Each use case specifies a sequence of actions, including variants, that the system can perform, interacting with actors of the system. An actor is defined as the user that communicates with a use case to invoke its functionality.
- Conceptual Model - in UML a conceptual model is explained with a set of static structure diagrams. It does not show the details of operation.
- System Behaviour Model – Collaboration diagrams explains object interactions in a graph or network format whereas System sequence diagram illustrates events from actors to systems.
- Class Model – this involves design of class diagrams
- Design State Model – this involves state diagrams for classes.

The use case diagram in Figure 3 defines the relationship among actors and use cases.

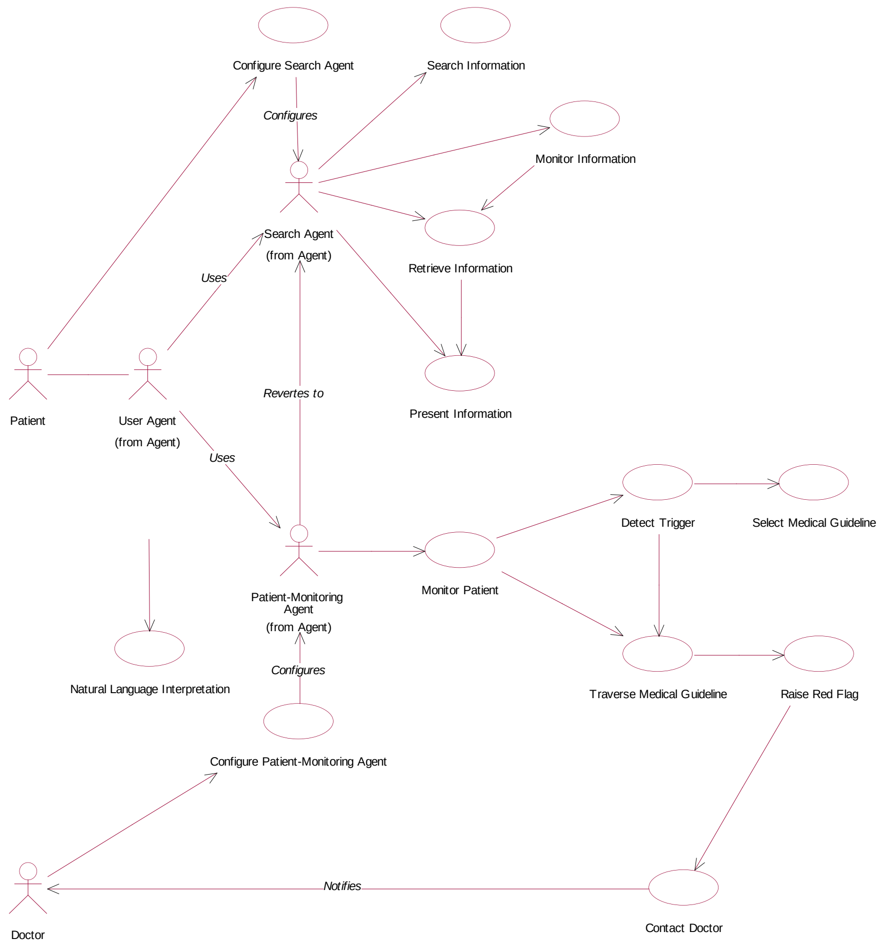


Figure 3. Use Case diagram

Three key Agents are identified in the design: Personal Assistant, Search Agent, and Patient-Monitoring Agent. Agents are represented as actors because they may autonomously invoke the functionality encapsulated in the use cases in the system. Each Agent is responsible for managing a subset of use cases as discussed below.

Personal Assistant
This is an interface Agent that interacts with users. It translates user input into functional requests. Which are forwarded to the appropriate Agent.

Natural Language Interpretation – The personal assistant Agent requires natural language interpretation in order to communicate with the users. This can be achieved by encapsulating existing natural language software and exposing a standard interface for use by the Agent. The prototype system was built using a modified version of the Eliza algorithm. We modified the algorithm to react selectively to different trigger words as will be discussed later. The algorithm was coded in Visual Basic 6 and the trigger repository was built using Microsoft Access as a Database. The patient-agent interaction was further enhanced by vocalizing the agent's responses via the Lernout & Hauspie Truvoice synthetizer.

Search Agent
Invokes the use cases involved in the search functionality. These use cases are discussed below.

Configure Search Agent – The configuration of a Search Agent allows the user to specify criteria such as keywords for the search, indicate amount of information desired, select search extent, choose presentation/report style [BUI, 00].

Search Information – The Search Information use case consists of four main phases: initialization, perception, action and effect [YOU, 00]. The initialization phase should set up all the variables – Target keywords, the goal (number of websites containing the target), a place to start, and a method of searching. These variables can be defined by the user via the Configure Search Agent use case. The perception phase is centered on using the knowledge provided to contact a site and retrieve the information from that location. The action phase takes the information from the perception phase and determines if the goal has been met. This is the intelligence of the Agent – it must decide if the retrievals satisfy the goal (quality, quantity, etc.). The effect phase is the listing of locations [YOU, 00]. We programmed the Search Information use case in Visual Basic 6. The program connects into the Ask Jeeves © and WebMD© search engines, retrieves and merges the search results. If the computer is offline at the time of a search request, it will be put in a queue.

Monitor Information – Patients may configure their Agents to notify them when new information about their condition (or any other condition) becomes available. To achieve this, initial search results will stored. The search will be repeated after a determined period of time. If the subsequent search yields new information, the user can be notified by invoking the Retrieve and Present Information use cases.

Retrieve information – This use case is defined separately from the Search Information use case because they may occur asynchronously. A Search Agent may perform a search while the user may be offline. Therefore, the search result will have to be stored until it can be presented to the user, hence the Retrieve Information use case.

Present Information – As mentioned in the description of the Configure Search Agent use case, users will require information to be presented in a specific format. This use case provides the functionality to format the information obtained from the Retrieve Information use case to fit the user's desires.

Patient-Monitoring Agent
Monitors the patient behavior for any suspicious conduct (triggers) in which case it can to divert the interaction. Ultimately, this Agent is responsible for invoking the relevant use cases to determine if a red flag should be raised and the doctor notified.

Configure Patient-Monitoring Agent – The doctor has the ability to configure the Agent to detect specific patient behaviors that he wants to monitor. These behaviors can be expressed as a specific set of trigger words. The doctor can add/modify the pre-existing set of generic trigger words. For example, a diabetic patient has a 50-fold increased risk of foot gangrene due to bad peripheral circulation [KUM, 98] therefore his/her doctor may want to add "cold feet" as a trigger. The doctor can associate the Diabetic Triage to this trigger.

Monitor Patient – Monitoring the patient requires an ongoing process that detects suspicious behavior, responds, and takes the correct action. The Monitor Patient use case provides this functionality by parsing patient input and invoking the use cases discussed below. This functionality was built using Client-Side JavaScript. The program checks for the existence of any trigger words in the patient's input.

Detect Trigger – An active list of trigger words or word sequences will be maintained and compared to the patient's inputs. If a match is detected, this could indicate suspicious behavior by the patient. This will trigger the Select Medical Guideline use case. We stored the active list of trigger words in the Access Database. Active Server Pages retrieve the list of trigger words.

Perform Triage – This use case encapsulates the Emergency Triage for Diabetics. The appropriate recommendation is made to the user (i.e. See you doctor NOW) and the doctor is notified using the Contact Doctor use case. We wrote the algorithm in Visual Basic 6. It accepts the patients input and returns the degree of emergency on a scale of 1 to 5 as specified by the triage protocol.

Contact Doctor – The Contact Doctor use case will provide the communication and reporting functionality. Doctors may select various methods of communication (email, pager etc.). The prototype uses Visual Basic 6 to access the MAPI interface and send an email stating the name of the patient and any potential problems (i.e. patient measurements outside of the range, triage emergency level above 1)

6. Discussion
Most efforts in healthcare technology are focused on assisting the doctor in diagnosing and treating a disease. This approach tends to omit a key component of the Healthcare Cycle: the Patient. The new trend in medicine favours the inclusion of the patient as an integral part of the healing process. A review of 22 studies by Stewart et Al [STE, 99] indicated a positive effect of communication on actual patient health outcome such as pain, recovery from symptom, anxiety, functional status, and physiologic measures of blood pressure and blood glucose.

Future developments in healthcare technology should focus on improving this positive trend of patient integration. To this purpose, Agents can play a key role in facilitating patient-doctor communication and collaboration. The benefits that arise from implementing such a system are discussed here.

- **Proactive healthcare with emphasis on prevention**

From the perspective of HealthCare services, the presence of a medically-aware Agent interacting with a patient is an opportunity for proactive healthcare. Patients will often experience symptoms long before they visit a doctor. In fact, studies indicate that patients with a history or a gradual progression of symptoms will present to the doctor later than patient with acute attacks [MUM, 99] and generally doctors perceive a greater need for investigation than patients [PEA, 98]. Furthermore, patients will often seek information about their symptoms to decide on whether to see a doctor. Agents can play a pivotal role in providing the information patients seek and proactively determining the importance of the symptoms. Arguably, patients with potentially serious medical conditions would present earlier, usually improving prognosis.

- **Enhanced Patient-Doctor interaction**

Agents can act as an extension of the doctor by providing patients with contextual information and monitoring the treatment to completion. Often patients do not have the opportunity to ask all their questions to the doctor. This is due to the limited time that doctor can spend in a consultation or because the patient may have reservations about asking too many questions. Either way, Agents can fill the gap by providing contextual information such as: why was a specific treatment selected or what other treatments are available. Therefore, Agents may mitigate the frustration patients often feel towards their doctor for not providing them with all the answers, but more importantly, maintain the continuity of the patient-doctor relationship.

- **Enhanced information exchange**

Neither diagnosis nor treatment can be completely successful unless the patient and the doctor understand each other and collaborate in an effort to gauge the other’s requests, needs and concerns. This is made even more difficult by the fact that there are often big differences between the doctors and patients in terms of expectations, vocabulary used, and other factors [BUC, 98]. A study at the University of Pittsburgh describes a model of asynchronous communication between doctors and patients that reduced some of the differences in communication [BUC, 98]. Agents can play this role by allowing patients to pace themselves with the relevant domain terms, some of the medical factors underlying the condition under question, and the justification and implication of the prescribed treatment plan. Also, Physicians can request more information of patients through the Agent.

- **Healthcare services to geographically remote patients**

The implementation of software Agents makes them accessible via the Internet. Thus patients and doctors will benefit from the ability to maintain a relationship while geographically remote.

6.1 An Example Scenario

For treatment of type 2 diabetes, tele-homecare is effective [ADA, 99]. Services like Webmd [WMD, 01] provides information and assistance to the patients about diabetes and its management. Whereas commercial products like The Diabetes Partner PC (Numedics) and Caregiver Desktop (Health Hero Network Inc) [FEE, 99] allow patients to take more active role in treatment and management of diabetes. However, the application program at both ends (patient and medical service provider) directly manipulates the collection and management of information related to treatment. Hence it suffers from following disadvantages [BRA, 97]:

- Very limited scalability
- Actions are initiated in response to immediate user interaction only
- Delegation cannot be achieved
- Less flexibility
- Function oriented rather than task oriented
- Lacks adaptability

The agent-based framework [GMR, 00] for diabetes management overcomes some of these limitations in addition to the provision of exhaustive managed information to the patient.

7. Conclusion

This paper has discussed a multi-agent based scenario for the implementation of networked applications and services in healthcare industry. The idea is that the interaction of human roles in healthcare can be enriched by the computational interactions of agents representing each human role. This is illustrated in the paper using a typical doctor-patient interaction process in a medical treatment.

Agent-based personal assistants have the potential to dramatically improve the patient-doctor relationship by providing continuity, facilitating communication and decision-making. To unleash this potential a system needs to be deployed and evaluated in a clinical environment. An evaluation strategy based on functional, technical, organizational, clinical and economic factors is under consideration. Though Agent-based personal assistants are increasingly being investigated, issues such as scalability, crash recovery, inconsistencies and security need to be properly addressed [GEN, 97].

The Internet will play a fundamental role in the delivery of healthcare services. One of the driving forces has been the desire for medical information by the consumer (the patient). Healthcare agents can satisfy this desire by providing medical information to patients by acting as an extension of the doctor. The accurate balance of Artificial Intelligence with human intervention will improve the quality of service. Software agent technology is ideally suited for this.

Negroponte argues that in a digital world, humans will team up with computers as co-workers to optimize execution of decisions [NEG, 95]. Agent-based personal assistants are a direct extension of this concept.

7. References.

[ABR, 96] Abramovich H; Schwartz E; Three stages of medical dialogue; Theoretical Medicine 1996 Jun; 17(2): 175-87

[ADA, 99] Adil A et al; Diabetes Home Monitoring Project; Telemedicine Journal; Vol 5 ; page 76; 1999

[BRA, 97] Bradshaw, M j; An Introduction to Software Agents; Software Agents; Jeffery M. Bradshaw; AAAI / MIT Press; 1997; page 3- 46;

[BUC, 98] Buchanan BG et al; Designing computer-based frameworks that facilitate doctor-patient collaboration.

[BUI, 00] Bui T; Building Agent-based corporate information systems: An application to telemedicine; European Journal of Operational Research 122 (2000) 243-257

[COI, 97] Coiera Enrico; The Guide to Medical Informatics, The Internet and Telemedicine; Oxford University Press, 1997

[COT, 99] R Cotran et al, Pathologic Basis of Disease sixth edition, Phildelphia: WB Sauders

[DEL, 99] Della Mea V et al; Internet Agents for telemedicine services; Medical Informatics (1999) Vol. 24, No. 3, 181-188

[EPS, 97] Epstein M: Diabetes and Hypertension: the bad companions. J Hypertens 15 (suppl 2):S55, 1997

[FAU, 98] Fauci A et al; Harrison's Principle of Internal Medicine 14th edition; McGraw-Hill, 1998, pg 1927

[FEE, 99] Feedback Research Services: Diabetes Care Markets: Revenue and Cost-Savings Forecasts; ISBN # 1-889713-20-1; 1998-99

[GAN, 00] Ganguly P, Ray P; A methodology for the development of software Agent based interoperable telemedicine systems: a telemedicine electrocardiography perspective; Telemedicine Journal 2000 Vol 6 Num 2

[GEN, 97] Genesreth R. M, An Agent-Based Framework for Interoperability, Software Agents; Jeffery M. Bradshaw; AAAI / MIT Press; 1997; page 317- 345;

[GMR, 00] Ganguly P, Mazzi C & Ray , Software Agent Based Personal Assistant for Tele- Homecare in Diabetes Treatment; Proceedings of MAMA 2000 (Paper #1574-367); December 11-15, 2000, Wollongong, Australia

[GRI, 96] Griffin JP. Griffin JR. Informing the patient [see comments]. [Journal Article] Journal of the Royal College of Physicians of London. 30(2):107-11, 1996 Mar-Apr

[HEA, 93] Heathfield H, Wyatt J; Philosophies for the design and development of clinical decision support systems; methods Inf Med 1993;32:1-8

[HJO, 99] Hjortdahl P et al; Internet and the physician-patient relationship – from "thank you' to "why"; Tidsskr Nor Laegeforen 1999 Nov 30; 119(29):4339-41

[KIM, 97] Kim J et al; An empirical study of the Health Status Questionnaire System for use in patient-computer interaction; Proceedings / AMIA Annual Fall Symposium: 86-90, 1997

[KUM, 98] Kumar P, Clark M; Clinical Medicine Fourth Edition; W.B. Sauders, London 1998 pg. 978

[LAN, 99] Lanzola G et al; A framework for building cooperative software Agents in medical applications; Artificial Intelligence in Medicine 16 (1999) 223-249

[MUM, 99] Mumford AD et al; Delays by patients in seeking treatment for acute chest pain: implications for achieving earlier thrombolisis; Postgrad Medicine Journal 1999 Feb; 75(880):90-5

[NEG, 95] Negroponte N; Being Digital; Alfred A. Knopf, New York, CT, 1995

[PAR 97] Parker SL, et al: Cancer Statistics, 1997. Ca Cancer J Clin 47:5, 1997

[PEA, 98] Peay MY; Peay ER; The evaluation of medical symptoms by patients and doctors; J Behav Med 1998 Feb; 21(1):57-81

[STE, 99] Stewart M et al; Evidence on patient-doctor communication; Cancer Prev Control 1999 Feb;3(10):25-30

[TAT, 90] Tattersall RB, Gale EA (eds) , Diabetes Management, Edimburgh: Churchill Livingstone, 1990

[TEN, 00] Tennstedt SL; Empowering older patients to communicate more effectively in the medical encounter; Clinics in Geriatric Medicine. 16(1):61-70, ix, 2000 Feb.

[YOU, 00] Youngblood M; Web Hunting: Design of a Sumple Intelligent Web Search Agent; ACM Crossroads 1999. http://www.acm.org/crossroads/xrds5-4/webhunting.html

[VON, 93] van Bemmel JH; Criteria for the acceptance of decision-support systems by clinicians; lessons from ECG interpretation system; Proceedings of the AIME'93 Conference, 1993: 7-10

[WMD, 01] WebMd; http://www.Webmd.com accessed on 26/2/01