ISSN No. 0976-5697

Volume 4, No. 2, Jan-Feb 2013

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Building a Medical Diagnosis Intelligent System Using Simple Tools

Nael Hirzalla
Applied Science UniversityAmman, Jordan
hirzallah@asu.edu.jo

Abstract: Artificial Intelligence covers many topics including Intelligent or Expert Systems. This paper focuses on one of the fields that Intelligent Systems could be applied to, the field of Medicine. A prototype was implemented to show how a rule-based system could benefit families to diagnose their children for common diseases and save visits to paediatricians. The proposed system presented in this paper is very simple. It allows users who are not considered software developers or experts in the field of IT to develop their own system if provided with the needed Medical data.

Keywords: Expert System; Artificial Intelligence, Medical; Tools

I. INTRODUCTION

Intelligent system or knowledge-based system (KBS) refers to a computer program that simulates the professional capabilities of a human expert [1]. As stated by Edward Feigenbaum“It can advise, analyze, categorize, communicate, consult, design, diagnose, explain, explore, forecast, form concepts, identify, interpret, justify, learn, manage, monitor, plan, present, retrieve, schedule, test or tutor. They address problems normally thought to require human specialist for their solution”.

When people want to find an answer or a solution to a certain problem, the person uses the knowledge he or she has gained from various sources such as life, books, and teachers. If that problem is new, the person will need to search for the solution. One way to do that is to dig through the books and manuals that are rich with tremendous amount of knowledge. Computer programs may assist in such a task by using conventional decision-making logic. These programs are referred to as Expert systems, or more formally, Knowledge-Based Expert Systems. They mainly use human knowledge to solve problems that normally would require human intelligence. These systems represent the knowledge as data or rules within the computer. These rules and data can be called upon when needed to solve problems. Different problems, within the same domain of the knowledge-base, can be solved using the same program without the need for any reprogramming.

Expert systems, like human experts, are experts only in their field and as such are highly domain specific. Unlike human experts, once set up, they need not be expensive, they can be easily and cheaply replicated and they do not grow old and start making errors. The main components of an Expert System are as follows:

a. The knowledge base is the collection of facts and rules which describe all the knowledge about the problem domain
b. The inference engine is the part of the system that chooses which facts and rules to apply when trying to solve the user’s query
c. The user interface is the part of the system which takes in the user’s query in a readable form and passes it to the inference engine. It then displays the results to the user.

The user interface represent some graphical or text based system that allows communication between the user and the system. The interface should allow the system to question the user in an easy to understand fashion and to allow the user to enter responses in a similarly simple fashion. At any point, the user should be able to query the system as to why it is asking a particular question and the system respond with a reasoned account of its actions.

Once the system has concluded its enquiries, it should be able to present its results in a meaningful way. If the information it is dealing with is not certain but probabilistic it should present a range of possible answers with the likelihood of each alongside.

This proposal focuses on the domain of medicine. It is believed that people do suffer in the traditional diagnosing process from the following issues:

a. Frequent unnecessary visits to physicians and pediatrics due to false alarming or uncritical symptoms
b. Unavailability or long queues at physicians and pediatrics during peak hours
c. Mistakes in the initial diagnosis that may complicate the problem or increase the cost.

Thus, there is a need for an expert system which rationalizes its diagnosis and treatment (i.e gives in-depth analysis), and is available with its peak performance round the clock. Unlike humans, such system should provide unemotional, steady and infallible nature help that would increase efficiency and prevent inadvertent mistakes in the initial diagnosis of a medical problem. It may also serve as a learning tool for students

II. LITERATURE SURVEY

Today, expert systems are used extensively in finance, manufacturing, scheduling, customer service, …etc. For instance, American Express uses an Expert System to automatically approve purchases. TaxCut uses an Expert System to give tax advice. Phoenix Police Dept uses an Expert System to help identify suspects using M.O. Thus, many expert systems exited in the market that addresses few problems in various fields. In the field of medicine, one may also find a number of such systems that exist in market as well as discussed in the literature. In this section we will address few of them. MYCIN was an early expert system designed to identify bacteria causing severe infections, such as bacteremia and meningitis, and to recommend antibiotics, with the dosage adjusted for patient's body weight. The name derived from the antibiotics themselves, as many antibiotics have the suffix "-mycin". The Mycin system was also used for the diagnosis of blood clotting diseases.

A. MYCIN:

MYCIN: Medical system for diagnosing blood disorders. First used in 1979. MYCIN was developed over five or six years in the early 1970s at Stanford University [2,3]. It was written in Lisp as the doctoral dissertation of Edward Shortliffe under the direction of Bruce Buchanan, Stanley N. Cohen and others. It arose in the laboratory that had created the earlier Dendral expert system.

MYCIN was never actually used in practice but research indicated that it proposed an acceptable therapy in about 69% of cases, which was better than the performance of infectious disease experts who were judged using the same criteria.

Clinical decision support system (CDSS or CDS) is an interactive decision support system (DSS) ComputerSoftware, which is designed to assist physicians and other health professionals with decision making tasks, as determining diagnosis of patient data. A working definition has been proposed by Dr. Robert Hayward of the Centre for Health Evidence; "Clinical Decision Support systems link health observations with health knowledge to influence health choices by clinicians for improved health care". This definition has the advantage of simplifying Clinical Decision Support to a functional concept.

B. CADUCEUS:

CADUCEUS was a medical expert system finished in the mid-1980s (first begun in the 1970s- it took a long time to build the knowledge base) by Harry Pople (of the University of Pittsburgh) [4], building on Pople's years of interviews with Dr. Jack Meyers, one of the top internal medicine diagnosticians and a professor at the University of Pittsburgh. Their motivation was an intent to improve on MYCIN – which focused on blood-borne infectious bacteria - to focus on more comprehensive issues than a narrow field like blood poisoning (though it would do it in a similar manner); instead embracing all internal medicine. CADUCEUS eventually could diagnose up to 1000 different diseases.

While CADUCEUS worked using an inference engine similar to MYCIN's, it made a number of changes (like incorporating adductive reasoning) to deal with the additional complexity of internal disease- there can be a number of simultaneous diseases, and data is generally flawed and scarce.

C. ONCOCIN:

ONCOCIN is an expert system, a clinical decision support system that was developed in 1979 at Stanford by Shortliffe's group and used primarily from 1981 to 1987 at the Stanford Oncology Clinic as well as various other locations [5,6]. ONCOCIN, arose from an effort to increase the explanation producing power of an existing expert system.This system was the successor of MYCIN and predecessor of the Protégé and Eon systems. ONCOCIN uses artificial intelligence techniques to offer advice to the physician on medicines, dosages, and testing; in this process it integrated medical record keeping with decision support. It can determine these drug doses on the basis of time schedule, toxicity, and blood counts.5 It was designed to aid the physician in decision-making by combining clinical data with chemotherapy protocol guidelines and knowledge provided by expert oncologists. To be clear on terminology used within ONCOCIN's definition, oncology is the branch of medicine dealing with the physical, chemical, and biological properties of tumors, including study of their development, diagnosis, treatment, and prevention; and a protocol in the sense used here is a plan for a course of medical treatment. The typical users of ONCOCIN were residents and clinical assistants rather than certified physicians. When the system was being implemented there were high hopes for its use because in oncology the knowledge was already formally documented. Unfortunately, the system did not live up to expectations after the system was put to use. In some cases the situations at hand did not fit into the rules known by the system, also it took about six weeks to enter the rules for a new protocol and to test them.1 Although the knowledge was documented it was not all-inclusive; new protocols were being found and used all the time, there was no way of ever getting a complete set of the protocols. The advice provided by ONCOCIN was approved by experts in only 79% of the cases. ONCOCIN used the same rule-based approach as MYCIN.

D. PUFF:

PUFF can diagnose the presence and severity of lung disease and produce reports for the patient's file [7]. PUFF was the first system developed using EMYCIN (Essential MYCIN, van Melle, 1979). It included the domain-independent features of MYCIN:

a. rule interpreter
b. explanation
c. knowledge acquisition.

This provided a mechanism for representing domain-specific knowledge in the form of production rules, and for performing consultations in that domain. PUFF was originally written in Interlisp using the SUMEX-AIM computer facility and had to be rewritten in BASIC before it could be installed at PMC. PUFF does not require direct interaction with a physician, avoiding the human engineering problem.

PUFF is primarily a goal-directed, backward chaining system employing some 400 production rules. If it cannot conclude the value for a parameter, it asks the user that a pulmonary physiologist reviews the PUFF report, and if necessary modifies it on-line before printing it out. The report was found not to require modification in 85% of cases. Modifications, where made, often consisted of comments such as "This is consistent with last visit". The PUFF basic knowledge base was incorporated into the commercial "Pulmonary Consult" product. Several hundred copies were sold in the 1980s and used around the world.

E. HDP:

The Heart Disease Program (HDP) [8] is a large diagnostic program covering most areas of heart disease. The physician can enter patient information about the history, physical examination, and laboratory tests, and then the program generates detailed explanations of differential diagnoses indicating the clinical data items which support each diagnosis. Its' Purpose is to assist physicians in the diagnosis of patients with heart disease. The Heart Disease Program (HDP) can be divided into 3 main components:

a. A user interface.
b. The knowledge base and inference mechanisms.
c. Mechanisms to summarize and explain diagnoses

III. PROPOSAL METHODOLOGY

We are proposing to implement the expert system by adopting the form of IF-THEN or IF-THEN-ELSE rules, where the IF part is the antecedent, the THEN part is the conclusion, and the ELSE part, if exists, is the alternative conclusion. Candidate hypotheses are derived through some pattern-matching system. A reasoning “inference engine” carries out the manipulation specified to obtain an answer. The inference engine is no more than a program, the function of which is to decide what to do at any given moment, that is, it recognizes and activates the appropriate rules. Generally, an inference engine should include an interpreter, which activates the relevant rules at any given moment, taking into account the current state.

The overall functioning of the inference engine occurs in cycles called basic production system cycles. The nature of these basic cycles is very different depending on whether the search process is directed by the data or by the objectives. Given that, the system is essentially based on rules. It will be necessary to define how the propagation of knowledge within the system can be affected.

There are two basic propagation methods: Forward chaining of rules, and Backward chaining. We will adopt in building our expert system on the Forward chaining approach. Developing our Expert System may be split into four phases as follows:

A. Phase 1: Problem Assessment:

Most organizations when considering any new technology will ask the very practical questions‘

a. Will it work? and
b. Why should we try it?

Expert system can be written much faster than a conventional program, by users or experts, bypassing professional developers and avoiding the need to explain the subject. It’s structured according to the following tasks:

a. Task 1: Determine motivation
b. Task 2: Identify the problems
c. Task 3: Performs feasibility study
d. Task 4: Perform cost/ benefit analysis
e. Task 5: Select the best methodology
f. Task 6: Write the project proposal

B. Phase 2: Knowledge Acquisition & Analysis:

This task is the most difficult challenge in the development of an expert system. Knowledge acquisition is inherently a cyclical process. An expert system gains its power from the knowledge it contains [9]. We are proposing to collect the knowledge required from a book entitled “American Medical Association Guide to Your Family's Symptoms “ [10].

C. Phase 3: Design and Implementation:

This phase begin with the selection of the knowledge representation technique and control strategy. This is followed with the selection of a software tool that best meets the needs of the problem. The programming language we plan to use is Visual Basic.Net.

This process is structured according to the following task:

a. Task 1: Select rules as Knowledge representation technique
b. Task 2: Select Programming Technique
c. Task 3: Develop the prototype, interface, and product primarily using IF/THEN type statements.

D. Phase 4: Testing:

The developed system will need to be periodically tested and evaluated to assure that its performance is converging toward established goals. It is important that these decisions be made early, at a time when the original project goals are established.

a) Stage 1:

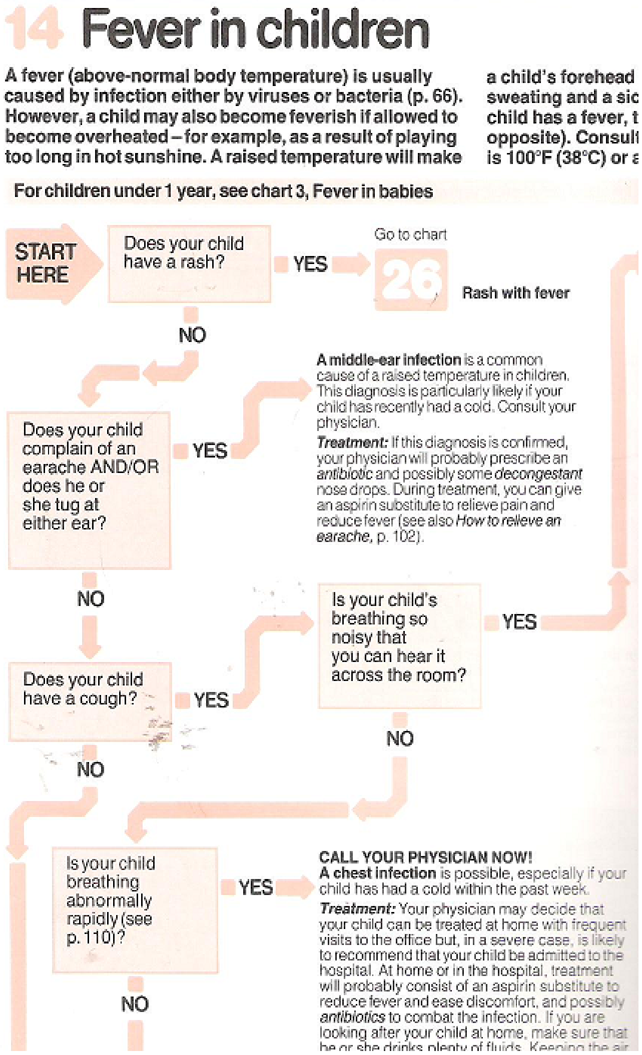
a. Preliminary Testing
b. Study the complete knowledge base
c. Uncover deficiencies in the knowledge and reasoning strategies
d. Validate knowledge representation and inference approach

b) Stage 2:

a. Demonstration Testing
b. Choose a problem of limited scope
c. Use demonstration to validate the expert system approach
d. Design interface to accommodate of the user

c) Stage 3:

a. Informal Validation Testing


14 Fever in children

A fever (above-normal body temperature) is usually caused by infection either by viruses or bacteria (p. 66). However, a child may also become feverish if allowed to become overheated – for example, as a result of playing too long in hot sunshine. A raised temperature will make

a child's forehead
sweating and a sic
child has a fever, t
opposite). Consult
is 100°F (38°C) or a

For children under 1 year, see chart 3, Fever in babies

A middle-ear infection is a common cause of a raised temperature in children. This diagnosis is particularly likely if your child has recently had a cold. Consult your physician.

Treatment: If this diagnosis is confirmed, your physician will probably prescribe an *antibiotic* and possibly some *decongestant* nose drops. During treatment, you can give an aspirin substitute to relieve pain and reduce fever (see also *How to relieve an earache*, p. 102).

CALL YOUR PHYSICIAN NOW!
A chest infection is possible, especially if your child has had a cold within the past week.

Treatment: Your physician may decide that your child can be treated at home with frequent visits to the office but, in a severe case, is likely to recommend that your child be admitted to the hospital. At home or in the hospital, treatment will probably consist of an aspirin substitute to reduce fever and ease discomfort, and possibly *antibiotics* to combat the infection. If you are looking after your child at home, make sure that he or she drinks plenty of fluids. Keeping the air

Figure 1 Chart image for an example of a major symptom Ref [10]

E. The general Life Cycle for Developing an Expert Systems:

The following represent the basic steps that compose a full cycle for developing an Expert System:

a. Problem Definition
b. Knowledge Acquisition
c. Knowledge Representation
d. Prototype system
e. Operational system
f. Knowledge base maintenance

IV. SYSTEM DESIGN

The prototype we are implementing is using Forward Chaining Systems. The system starts with initial facts and keeps applying the rules to draw the conclusion or the Result of the diagnosis. An example of a major symptom that we will study in the system is shown in chart image below:

Using flow chart to explain Figure 1, see Figure 2 depicts If-Then-Else usages to reach to the Result sought.

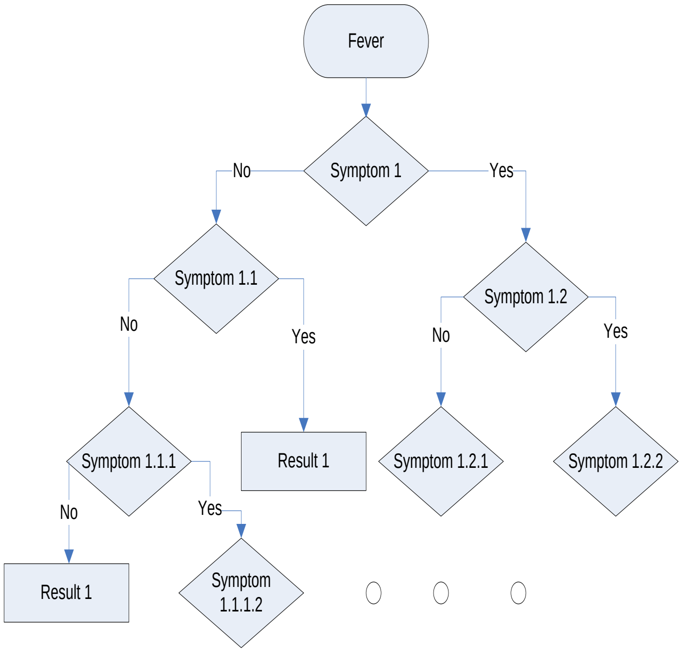


Figure 2 If-Then-Else

To generalize the implementation, since the IF-THEN-ELSE is common and the only things that are changing are the Rules applied and the Results drawn, we used one complete case, as depicted partially in Figure 1.

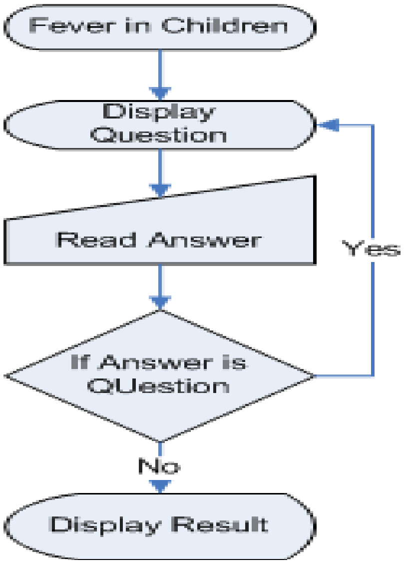
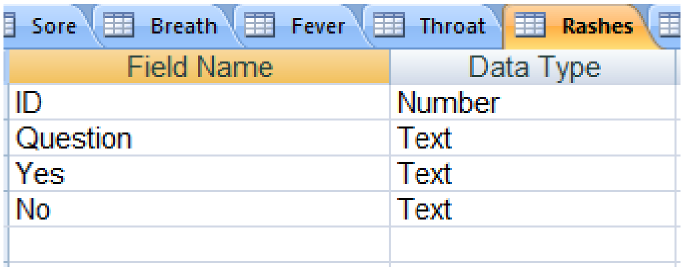


Figure 3Common IF-Then-Else

The following is the Pseudocodefor the system used:

```
Display Question
If Answer is Question then
    Go to step 1
Else
    Display result
```

This was implemented in VBA while the rules and associated results were stored in Access database. A table is made to represent a Major Symptom Name. Each record of these tables includes the following attributes:



Field Name	Data Type
ID	Number
Question	Text
Yes	Text
No	Text

Figure 4Table Fields

a. ID: Question Number
b. Question: The Question Text
c. Behavior upon Yes
d. Behavior upon No

The behavior mentioned above could be either

a) Another Question: Represented by “Q: TablenameQuestionNumber”. Example: “Q:Fever 3” Which means, it is a question number 3 from table named “Fever”.
b) Result: Represented by “Result:ResultText”. Example: “Result:Your child is suffering from chickenpox”

To split the string from, say the No field, into Question or Result, we used the following SQL Function:

Table name or ResultText =
(Mid([No],1,2)='Q:',Mid([No],3,InStr(3,[No]," ",1)-3),Mid([No],8,Len([No])-7))
QuestionNumber =
(Mid([No],1,2)='Q:',Mid([No],InStr(4,[No]," ",1)+1,Len([No])-InStr(4,[No]," ",1)),"")

The code for the common question approach is depicted in Table 1.

Table 1 “Question Form” VBA code

```
Option Compare Database
Dim index As Integer
Dim tables(1 To 100) As String
Dim questions(1 To 100) As String

Private Sub Command11_Click()
    If (index > 1) Then
        index = index - 1
        t = tables(index)
        q = questions(index)
        Form.FilterOn = False
        Form.RecordSource = t
        DoCmd.OpenForm "Question", , , "[ID] = " + q
    End If
End Sub

Private Sub Command9_Click() ' No
 Dim t As String
 Dim q As String
 If index = 1 Then
    tables(index) = Me.RecordSource
    questions(index) = "1"
 End If
 If [NoQue] = "" Then
        DoCmd.OpenForm "Result"
        If [NoTableOrRes] <> "" Then
          [Forms]![Result]![Text4].Text = [NoTableOrRes]
        End If
     Else
    t = [NoTableOrRes]
    q = [NoQue]
    index = index + 1
    tables(index) = t
    questions(index) = q
    Form.FilterOn = False
    Form.RecordSource = t
    DoCmd.OpenForm "Question", , , "[ID] = " + q
   End If
End Sub

Private Sub Form_Load()
  index = 1
End Sub

Private Sub Form_Unload(Cancel As Integer)
 'Me.Recordset = Null
End Sub

Private Sub YesBut_Click()
   Dim t As String
   Dim q As String

   If index = 1 Then
     tables(index) = Me.RecordSource
     questions(index) = "1"
   End If
   If [YesQue] = "" Then
        DoCmd.OpenForm "Result"
        If [YesTableOrRes] <> "" Then
          [Forms]![Result]![Text4].Text = [YesTableOrRes]
        End If
   Else
     t = [YesTableOrRes]
     q = [YesQue]
     index = index + 1
     tables(index) = t
     questions(index) = q
     Form.FilterOn = False
     Form.RecordSource = t
     DoCmd.OpenForm "Question", acNormal, , "ID = " + q
   End If
End Sub
```

V. EXPERIMENTAL RESULTS

Few major Symptom names were selected from the reference book [3] to include in our prototype. Each of these major symptom names is represented by a separate table. These major Symptom names are depicted in Figure 4.

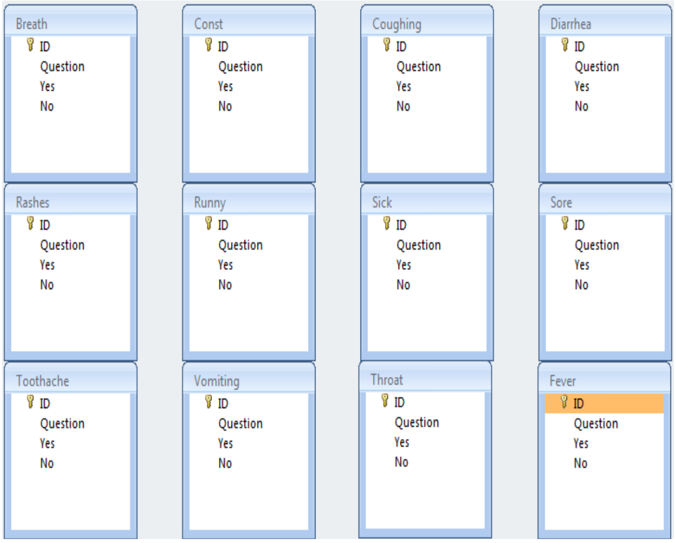
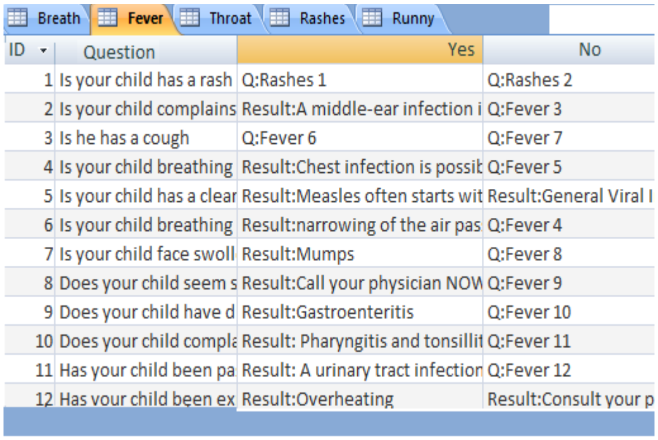


Figure 5Selected Major Symptom Names

Figure 5 shows records/rows sample of the table named Fever.



ID	Question	Yes	No
1	Is your child has a rash	Q:Rashes 1	Q:Rashes 2
2	Is your child complains	Result:A middle-ear infection i	Q:Fever 3
3	Is he has a cough	Q:Fever 6	Q:Fever 7
4	Is your child breathing	Result:Chest infection is possik	Q:Fever 5
5	Is your child has a clear	Result:Measles often starts wit	Result:General Viral I
6	Is your child breathing	Result:narrowing of the air pas	Q:Fever 4
7	Is your child face swoll	Result:Mumps	Q:Fever 8
8	Does your child seem s	Result:Call your physician NOW	Q:Fever 9
9	Does your child have d	Result:Gastroenteritis	Q:Fever 10
10	Does your child compla	Result: Pharyngitis and tonsillit	Q:Fever 11
11	Has your child been pa	Result: A urinary tract infection	Q:Fever 12
12	Has your child been ex	Result:Overheating	Result:Consult your p

Figure 6 Rows Sample of Fever Table

Figure 7 shows a screen shot of the main screen of the system.

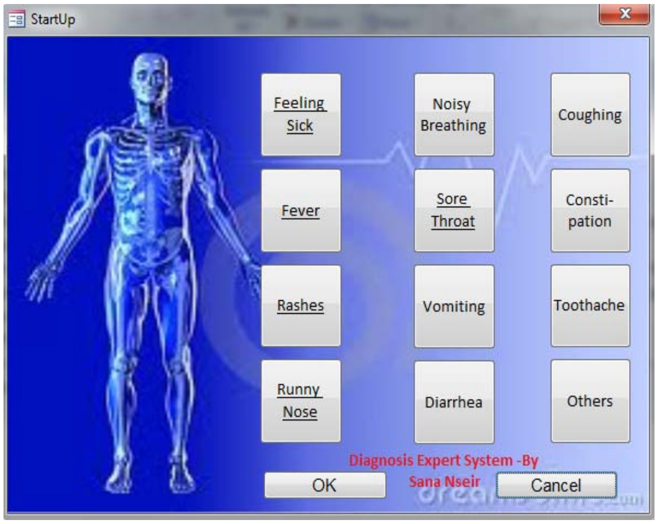
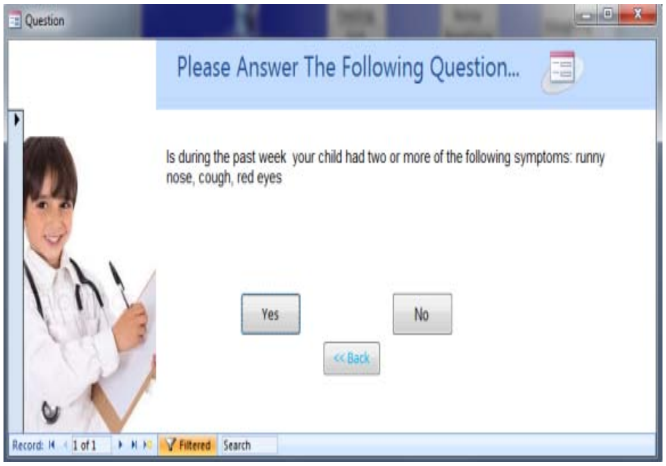


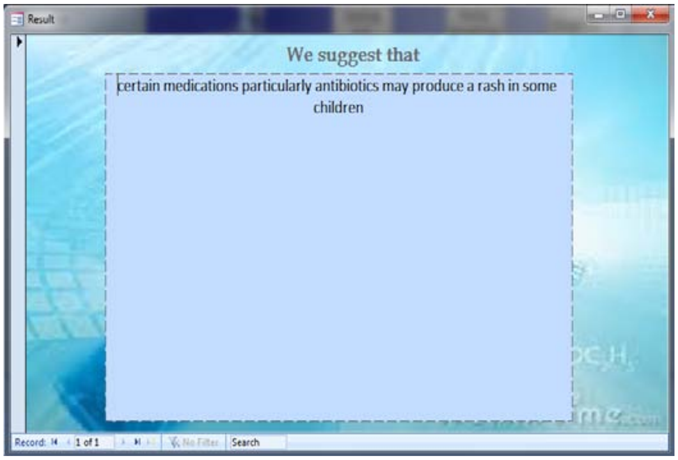
Figure 7 Screenshot of main screen

Figure 8 shows a sample question being displayed, while Figure 9 shows a sample Result.



Is during the past week your child had two or more of the following symptoms: runny nose, cough, red eyes

Figure 8 A Sample Question



certain medications particularly antibiotics may produce a rash in some children

Figure 9 A Result Sample

VI. CONCLUSION

This paper introduced a prototype that has been implemented to help families diagnose their children upon the appearance of symptoms. Generally speaking, Expert systems should be able to overcome the shortcomings of conventional human decision-making processes such as:Humans get tired from physical or mental workload., or Humans forget crucial details of a problem.

This paper proposed a simple system that could be developed by users who are considered as non-experts in the field of IT. The system offers many advantages for users when compared to traditional programs because they operate like a human brain. The advantages of such systems include:Being Conversational, Quick availability and opportunity to program itself, Ability to exploit a considerable amount of knowledge, and Reliability.

VII. ACKNOWLEDGMENT

The author wishes to acknowledge the financial support received from Applied Science University that helped in accomplishing the work of this article.

VIII. REFERENCES

[1] Peter L. M. Kerkhof, Amparo Alonso-Betanzos, Vicente Moret-Bonillo "Medical Expert Systems " Published on Willey Online: 27 DEC 1999 DOI: 10.1002/047134608X.W1416

[2] Edward H. Shortliffe "Mycin: A Knowledge-Based Computer Program Applied to Infectious Disease," Annual Meeting of the Society for Computer Medicine, Las Vegas, Nevada, 10 November 1977

[3] E. H. Shortliffe. Computer-Based Medical Consultations: MYCIN. Elsevier/North Holland, New York NY, 1976.

[4] Banks, G (1986). "Artificial intelligence in medical diagnosis: the INTERNIST/CADUCEUS approach". Critical reviews in medical informatics 1 (1): 23–54. PMID 3331578.

[5] E. H. Shortliffe. Artificial Intelligence in Management Decisions: ONCOCIN. Reprinted in: Frontiers of Medical Information Sciences (R.L. Kuhn, ed.), pp. 173-185. New York: Praeger Publishers, 1988. KSL-86-39.

[6] http://neamh.cns.uni.edu/MedInfo/oncocin.html

[7] Aikins JS, Kunz JC, Shortliffe EH, Fallat RJ. Comput Biomed Res. 1983 Jun;16(3):199-208.PUFF: an expert system for interpretation of pulmonary function data.

[8] Startsman TS, Robinson RE. The attitudes of medical and paramedical personnel toward computers. Comput Biomed Res. 1972 Jun;5(3):218–227.

[9] John Durkin Expert Systems: Design and Development ISBN-13: 9780023309700
Publisher: Macmillan Coll Div Facsimile

[10] Guide to Your Family's Symptoms by The American Medical Association

[11] Gorry GA. Computer-assisted clinical decision-making. Methods Inf Med. 1973 Jan;12(1):45–51