KID : AN INTEGRATED SYSTEM (PROTOTYPE 4.0)†

10.1. Introduction

As a matter of fact, the domain knowledge is vast and varied. For the unfolding of intimate areas of the knowledge, it was required to develop demonstration prototypes incorporating different issues using Turbo Prolog as a fast prototyping tool. These prototypes also helped understanding the requirements of domain users soliciting the development of an integrated, menu-driven user-friendly system. This chapter is meant for presenting such a system devloped named 'KID'. The system KID is totally object-oriented starting from the analysis to implementation.

In the next section, we shall present the analysis and design of the system. Section 10.3 contains the graphical user interface (GUI) issue needed for the domain. Section 10.4 deals with the implementation issue. In section 10.5, we have presented an excerpt of a consultation session with the system. In the section 10.6, we have considered the validation issue. At last, our conclusions and discussions are summarised.

10.2. Analysis and Design

The total paediatric range is divided into the following stages : foetal (conception - birth), neonates (birth - 4 weeks), infants (4 weeks - 1 year), toddler (1 year - 3 years), pre-school going (3 years - 5 years), school going (5 years - 10 years) and adolescence (10 years - 18 years); and hence it was decided to have seven modules of the system. These seven sub-domains were again managed with pull-down menus. The following typical steps are followed by doctors investigating a baby except 'foetal' module:

General identification and preliminary complaints (Phase I)
↓
General / Physical examination (Phase II)
↓
Complaints oriented examinations (Phase III)
↓
Differential diagnosis and advice
↓
Related investigations (Phase IV)
↓
Final diagnosis and treatment

---

† This is based on the publication [Proc. of the 3rd. Int. conf. on Modelling and Simulation. Victoria University of Technology, Melbourne, Australia, 29-31 Oct' 97, 151-156] of the author.

In Phase I, a doctor considers the general informations like name, age, sex, weight at birth etc along with the preliminary complaints from parents/guardians. Some of these informations may pilot the actions to be taken in later phases of investigations. In Phase II, usually, the examinations are done using the following four steps in order :

i) Inspection;
ii) Palpation;
iii) Percussion; and
iv) Auscultation.

The examination steps should be followed in every case so that it should be methodological. But, however, examination steps may have to be altered in some situations. For example, very often Auscultation is done before Palpation and Percussion when the baby is quite and co-operative before he turns hostile owing to various examination procedures. Phase III considers the complaints oriented examinations as well as it considers the knowledge relevant with complaints and symptoms. The successful end of this phase leads to differential diagnosis and advice. For final diagnosis and treatment, we consider Phase IV where related investigations are done. After having the reports of laboratory investigations, the system suggests the final diagnosis and treatment.

We consider each phase as a distinct class. A class represents general properties and structure of a group of objects. Each object has its associated attributes. For example, the class 'Physical Examinations' (**Appendix B**) has the typical objects like heart, abdomen, ear, eye, liver etc. A typical object, say ear, has the attributes like 'malformed pinna' or 'preauricular sinus'. Objects manifest themselves in one of the ways represented in fig.10.1[1].

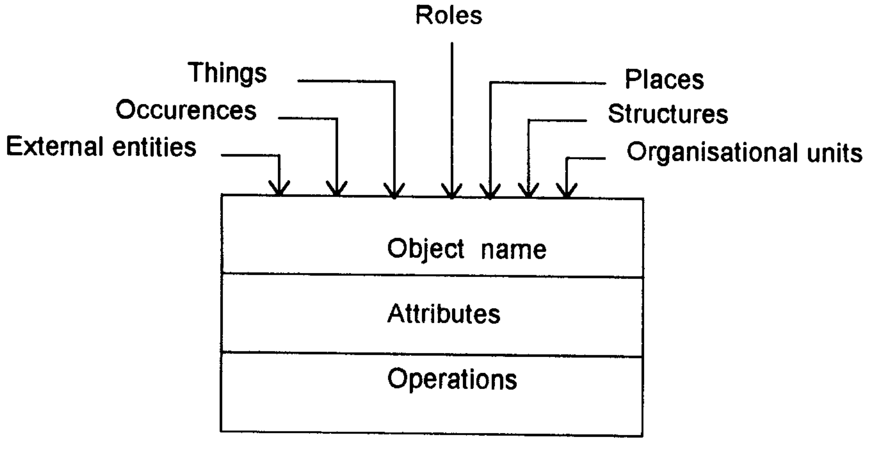


Fig.10.1 Objects

We have identified potential objects which are nothing but extraction of nouns. Coad and Yourdon [2] suggest following six selection characteristics for actual inclusion of objects in a project which are Retained information, Needed services, Multiple attributes, Common attributes, Common operations, and Essential requirements. They suggest that a potential object should satisfy all (or almost all) of the above characteristics for the legitimate inclusion of an object. In addition, we have identified different 'Rule Groups' for the identification of the cause(s) of a disease. For example, the cause of Jaundice in neonate may be Rh-Incompatibility or Septicemia or Neonatal Hepatitis or others. The Rule Group : Septicemia Diagnosis is as follows :

Rule Group : Septicemia Diagnosis(In a neonate presents with jaundice)

Rule 1

IF Appearence Day IS Third AND Lethergy AND Refused To Suck AND
Reflex IS Poor AND DIC & Shock AND Necrotising Enterocolitis
THEN Septicemia Diagnosis (CF = 85)

Rule 2

IF Appearence Day IS Fifth AND Lethergy AND Refused To Suck AND
Cry IS Poor AND Reflex IS Poor AND DIC & Shock
THEN Septicemia Diagnosis (CF = 70)

.
.
.

Rule n

IF Appearence Day IS Tenth AND Lethergy AND Refused To Suck AND
Cry IS Poor AND Reflex IS Poor AND Sclerema
THEN Septicemia Diagnosis (CF = 65)

Moreover, the links between different phases have been identified. For example, the attribute, 'Weight At Birth' of class 'Identification' (Phase I) will be useful in Phase III and Phase IV for suggesting differential diagnosis and advice; and final diagnosis and treatment. Currently, the system consists of 452 rules.

The structures and the number of items of each pull-down menus should certainly depend on the complexity of each module. For example, the foetal module may have the options for:

1) Prediction of gestational age from foetal parameters (sonograhically measured);
2) Prediction of foetal body weight from foetal parameters (sonograhically measured);
3) Determination of IUGR(Intra Uterine Growth Retardation).

Moreover, the issue of zonal variation has been considered in the above options. The neonatal module, as per present prototype, can fulfill the following objectives :

1) To find the general status of a neonate;
2) To find out the gestational age of neonates by date and/or examination;
3) To find out any problem related to its growth and development;
4) Complications due to prematurity and/or Low birth weight (LBW) and management;
5) Guideline of routine examination of neonates at delivery room;
6) Guidelines of management of new born at delivery room;
7) Common neonatal diseases affecting growth and development;
8) Guidelines of management of neonates requiring Level I and Level II care;
9) Guideline for directing the neonates for Level III care;
10) Finding the cause(s) of Jaundice and management.

The other modules are also planned as per the requirements of the sub-domains. For all these modules from foetal to adolescence, 'Phase I - Phase IV' scheme has been incorporated.

10.3. **Graphical User Interface (GUI)**

The better acceptance and popularly of a software depend not only on the internal technological content of the software but on the user interface to a large extent. A "user friendly" system is welcome. An interface may not satisfy all classes of users; this depends on the user's level of education, experience and expertise. It may be a formidable task to design a system with various levels of interface to satisfy all classes of users. But, there may have some common approaches of interface design along with some specific strategies for that particular domain and intended users. So, user interface design has as much to do with the study of people as it does with technology issues [1]. So, one has to fix up primarily the following : (i) Who will be the users ? (ii) How will they use it ? (iII) How will they learn to interact with the new computer-based system ? (iv) What will the user expect of the system ? (v) What sort of difficulties user might face and how to overcome those difficulties ? (vi) How the question-answer sequence is tailored ? Answering all these questions basically depends on the analysis of the problem domain and the requirements of the intended users (e. g. doctors or para-medical staffs etc. for medical domain). However, if a designer specifies a human-computer interface (HCI) that makes undue demands on STM(short-term memory) and / or LTM (long-term memory) of users, the performance of the human element of the system should degrade.

Four different models come into play when an HCI has to be designed : (i) design model, (ii) user model, (iii) system perception, and (iv) system image. A design model of the entire system incorporates data, architectural and procedural representations of the software. The user model depicts the profile of the end users of the system. To build an effective user interface, "all design should begin with an understanding of the intended users, including profiles of their age, sex, physical abilities, education, cultural or ethnic background, motivation, goals and personality" [3]. The system perception is the image of the system that an end user carries in his or her head. The system image combines the outward manifestation of the computer-based system with all the supporting information that describes system syntax and semantics. Each of these models may differ significantly. Reconciliation of these differences should be taken into consideration during the interface design. Users generally feel comfortable with the use of a system when the system image and system perception are coincident. For the purpose, the design model must have been developed to accommodate the information contained in the user model and the system image must accurately reflect syntactic and semantic information about the interface. In connection with the good coincidence of the system image and system perception, graphical user interface (GUI) methodologies might be a better choice. Moreover, GUIs are widely used for interaction with computers because of their transparencies, ease of use, user friendliness and robustness as they provide the user to interact with the system by manipulating the graphical objects [4,5]. For a diagnostic system where sign and symptom based input is necessary, graphical objects are highly suitable. In such domains, command driven, menu driven or simple textual user interface (TUI) might not be suitable simply because of the non-availability of displaying graphic objects / images. Medical diagnostics is such an application domain where GUIs are highly suitable for better decision making. For example, the Dubowitz [6] method of determining menstrual / gestational age of a new born uses different postures of the baby as depicted in fig. 10.2.

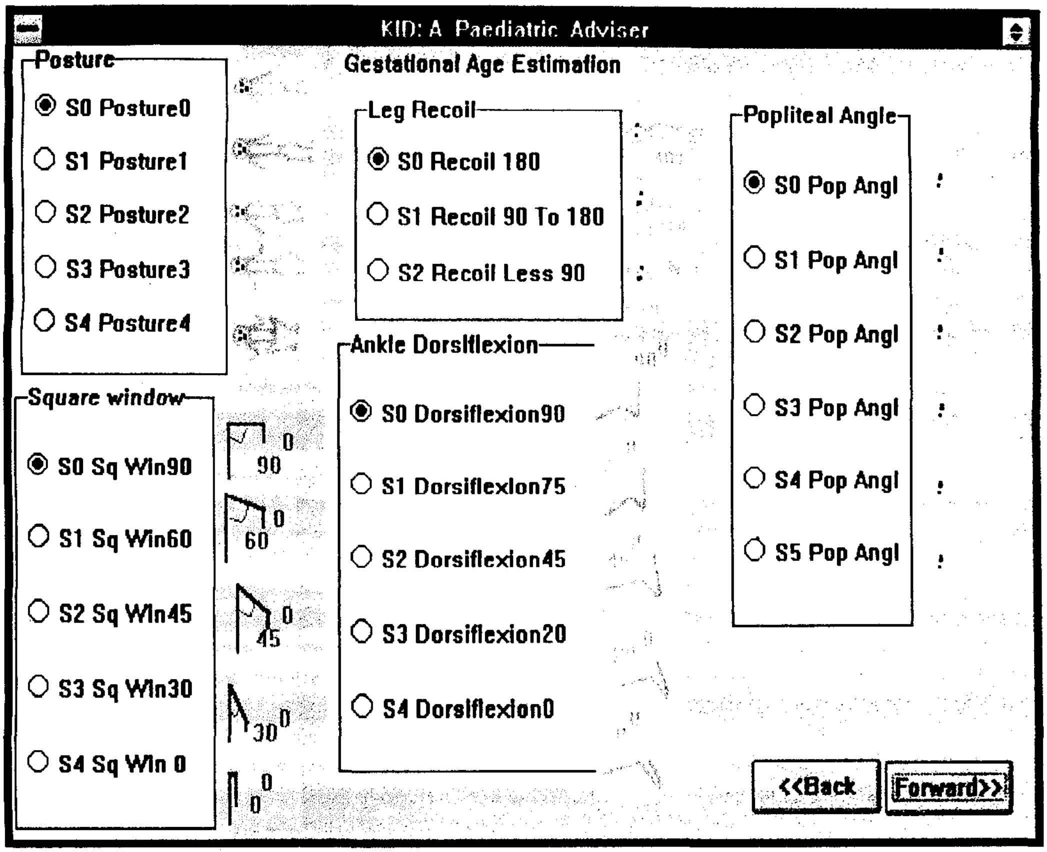


Fig. 10.2. Postures of neonates.

Our system provides GUI with the following characteristics :

(i) Having facility to select/deselect multiple options from a specified item-screen (e.g.,skin, fig.10.3).
(ii) Ability to take inputs at per user perceptions.
(iii) Ability to display pictures/images and texts as well as a combination of these.
(iv) Real-world situations are simulated through the system. For example, the 'postures' of a neonate (fig. 10.2) can be fed to the system as input as a doctor visualises a neonate.
(v) Lastly, user friendliness i.e. a menu-driven, GUI- based system needs no such rigorous training for its use.

10.4. **Implementation**

As a developmental tool one can use one of three : AI-languages, tool kits and shells. The selection of an appropriate tool should depend on the characteristics of the problem domain as well as on the architectural approaches with varying functionality and performance of candidate tools.

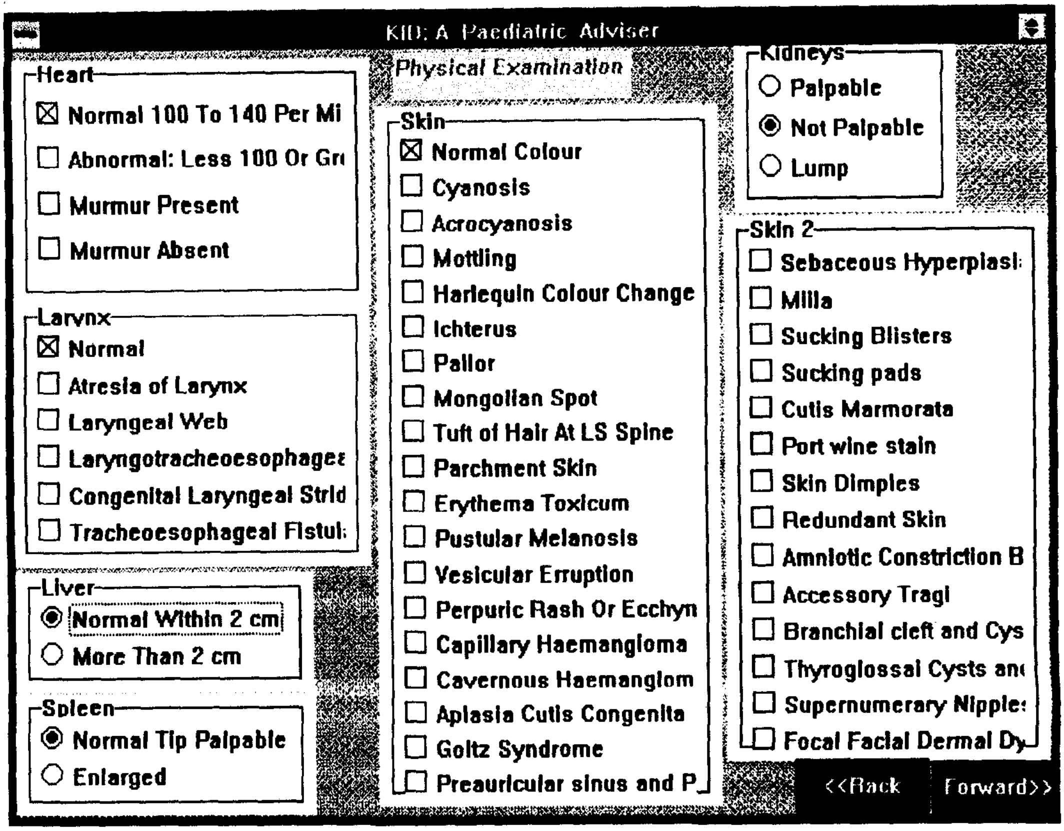


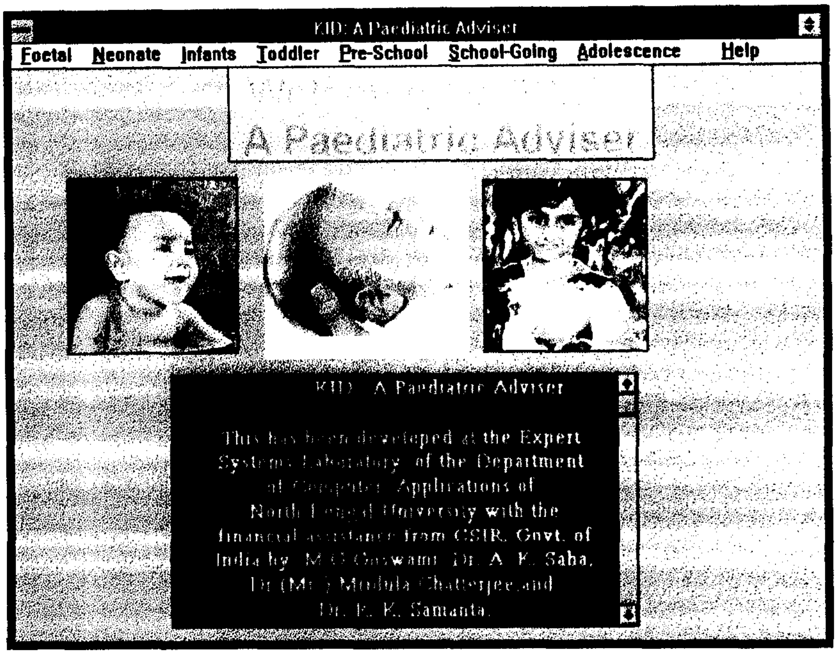
Fig. 10.3. Physical examination input screen.

This selection process should also be governed by whether the goal of the system development is to develop a system for actual use or to make major advances in the state-of-the-art of ES-technology. The goal of this system KID is to use it for practical situations. For the purpose, a large, hybrid object-oriented toolkit should be more appropriate as the problem is now well understood and formulated.

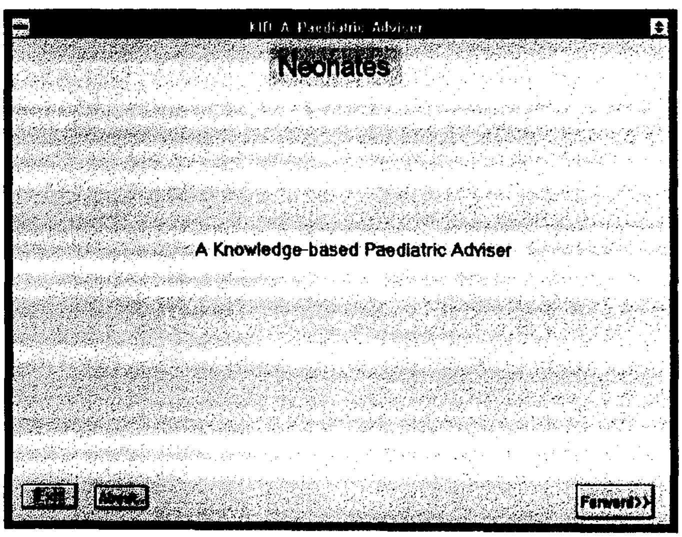
As an implementation tool, we have used Level5 Object running on PC and under Windows environment. Level5 is totally object-oriented starting with the design of the software itself. The editor, visual windowing system, displays, database interfaces, inference engines, knowledge bases, devices, files and timers are all built-in objects [7]. Level5 packages these objects as system classes that contain a rich array of built-in-logic and object tools, which give the developer ultimate control over application design and function. But, however, it can manage inexactness of uncertainty type only.

10.5. A Consultation Session : An Excerpt

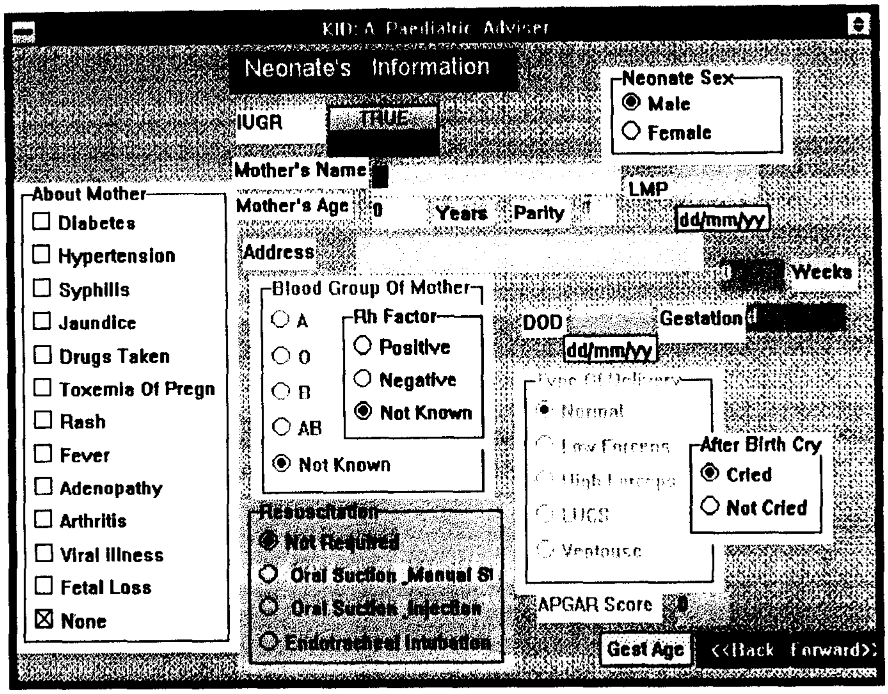
Let us now present an excerpt of a typical consultation session with the KID - expert. The following is the opening screen of the system :



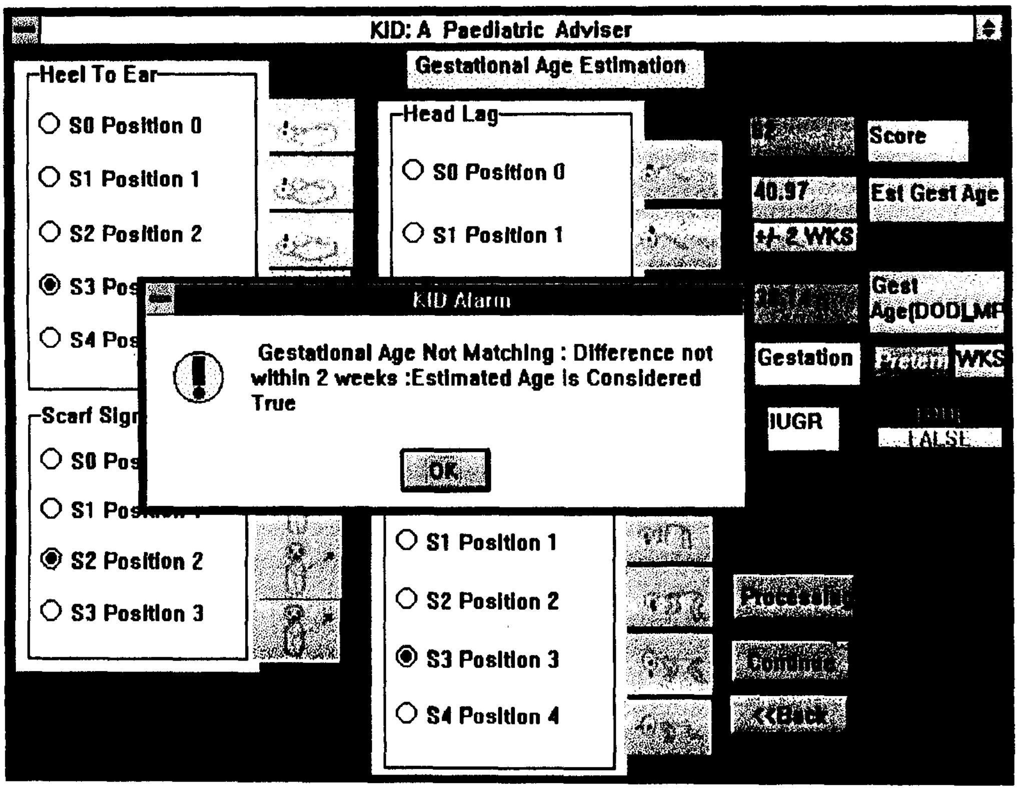
Using different menus and pull-down menus one can branch to different units of diagnosis and management. Let us select **Neonate ⇨ new case** as the following screen appears :



Click 'forward' you will get next :



Concentrating on the determination of menstrual / gestational age of a neonate using Dubowitz [6] method and comparing it with the LMP - DOB method we get the following output screen.



10.7. **Conclusions** and **Discussion**

In this chapter, an attempt has been made to present an integrated expert system (KID) within the confines of the state-of-the-art of ES-technology. Different developmental issues have been discussed. The findings of this system are compared with the findings of a domain expert. At per our initial observations, the agreement is satisfactory. But, for a system handling human life, more scrutiny is required before using it as a production/commercial system. More issues need to be added into the system to make it more useful. Our future efforts will obviously be towards such achievements.

References

1. R. S. Pressman. Software Engineering - A Practitioner's Approach. 3rd ed., McGraw-Hill, Inc; 1992.

2. P. Coad and E. Yourdon. Object-Oriented Analysis. Prentice-Hall; 1990.

3. B. Shneiderman. Designing the user Interface. Addison-Wesley; 1987.

4. G. Lee. GUI application development. Prentice-Hall; 1993.

5. D. Collins. Designing object-oriented user interfaces. Benjamin Cummings Pub. Co.; 1995.

6. L. Dubowitz and V. Dubowitz. Gestational Age of the Newborn. Addison-Wesley; 1977.

7. Level5 Object for Microsoft Windows Reference Guide (Rel 3.0). Information Builders; New York, USA; 1993.