Chapter 4 System Architecture

An overview of the SIMPRAC data model is given in Figure 32 while a more detailed XML database schema can be found at Appendix A on page 225.

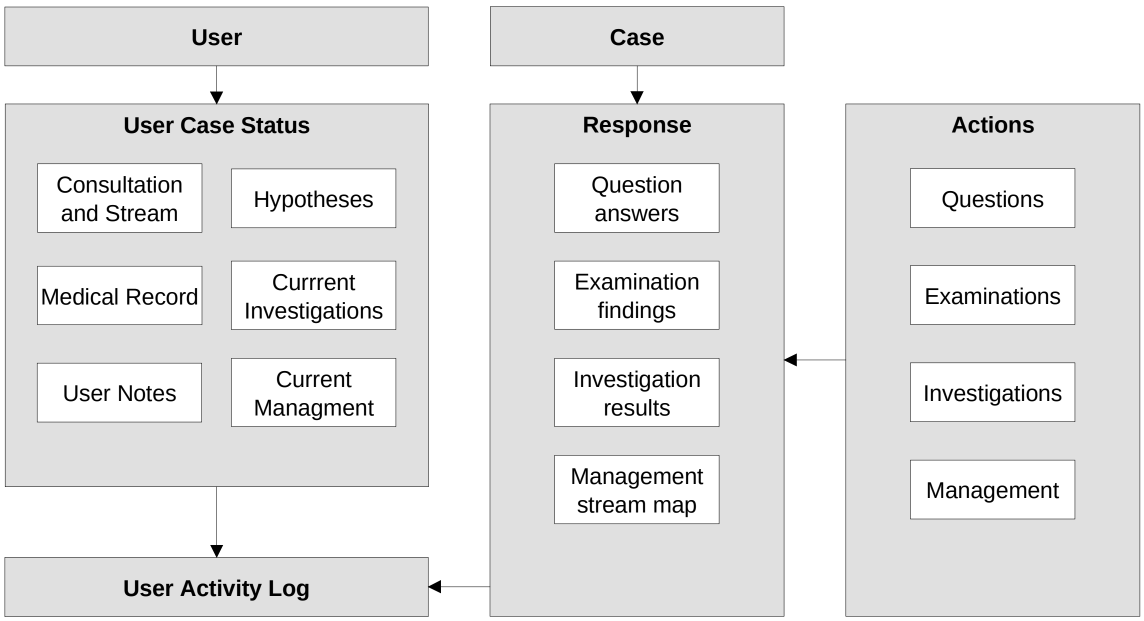

Figure 32: Overview of application data model.

The system can be viewed from three major perspectives. The first is the user model, the second is the disease model, and the third is the series of ancillary functions, including the library, medical record and user notes.

4.1 User Model

Figure 33 illustrates the structure of the user model, and the model for a single user is shown at Appendix B, on page 234. The user is uniquely identified by their username and each user is associated with a set of status records. The status records, for each user, are defined by:

1. The user's username
2. The case selected by the user, and
3. The professional background or peer group selected by the user.

Thus a user may attempt any case from different professional backgrounds.

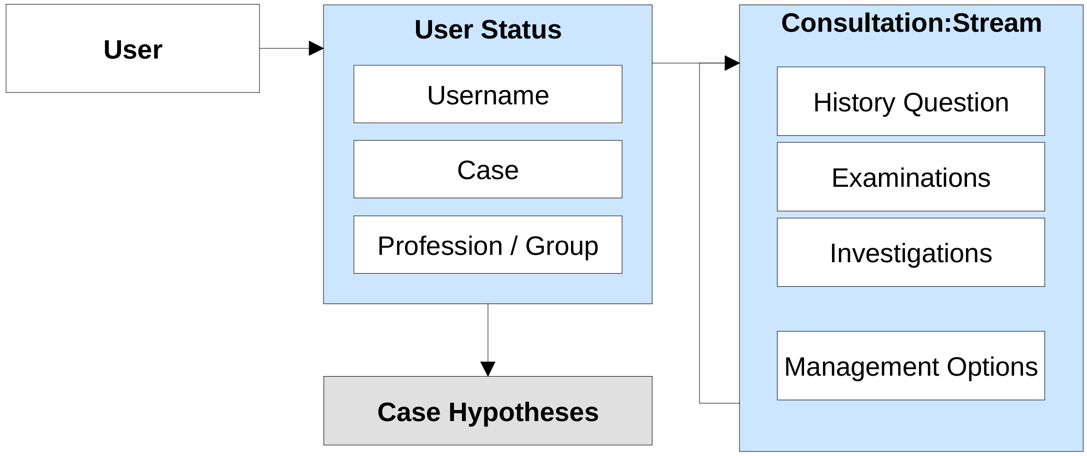

Figure 33: SIMPRAC User Model - Overview.

As illustrated in the diagram above, a case is undertaken by the user as a series of consultations, each of which is comprised of history questions, examinations, investigations, and management options. Information is also held on the user's hypotheses.

An abridged version of the model at Appendix B is given below (Figure 34). It can be seen that this user has registered for the first case as a medical student. The first consultation begins in Stream One. This user, Student04 has asked 8 questions beginning with, "Do you have a past history of hyperlipidaemia?" In the full interaction, not shown in Figure 34, Student04 also performed 59 examinations, nine investigations, and selected four management options in their first consultation. In the consultation two they were in Stream Four and asked two questions, did not perform any examinations, requested eight investigations, and selected the same four management options. In the third consultation this user was in Stream Four, asked five questions, again did not perform any examinations, requested nine investigations, and maintained the same four management options. Finally, in the fourth consultation, the user was in Stream Six, asked six questions, did not perform any examinations, requested 10 investigations, and changed from using a fibrate to uncooked cornstarch.

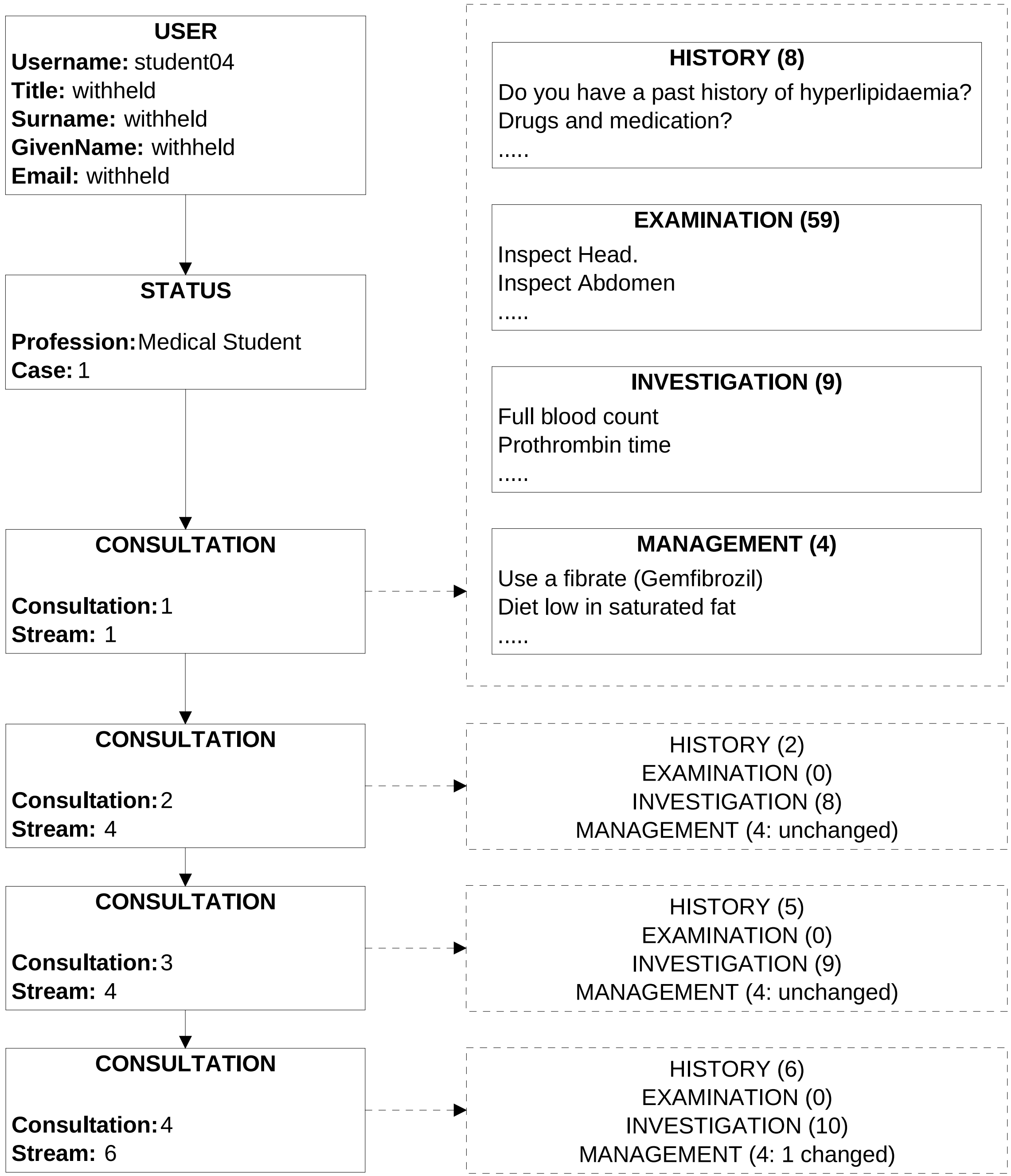

Figure 34: Abridged user model for Student04 (See Appendix B for full details).

4.2 Disease Model

The disease model is summarized in Figure 35. As can be seen from the illustration, the disease model is very simple. As the case proceeds, the patient moves along a series of states. These states are characterized by only two variables, the consultation and the stream. Each of these states is associated with three classes of diagnostic action response:

1. Answers to the questions,
2. Examination findings, and
3. Results of investigations.

As discussed in Section 3.3, each response is assigned a relevance score that is used in the reflective component of the review. The management actions are treated slightly differently to the diagnostic actions. In addition to having a relevance score for the patient's current state, the management actions also determine the state transitions of the patient. The management actions are mapped to a new stream in the next consultation, and these mappings are assigned weight values independent of the relevance scores. The responses, relevance scores, and state transitions are described in more detail below.

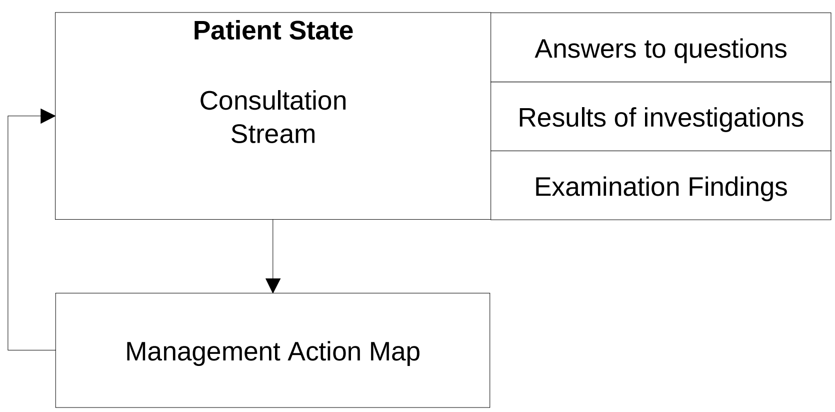

Figure 35: The disease model

4.2.1 Action Responses

The diagnostic actions have been described in chapter three, and include history questions, physical examination, and investigation requests. Each action response is specific to a particular stream and consultation. However, in practice most of the actions have a default response. If a consultation- and stream-specific response cannot be found, then the default response is returned. Furthermore, each action has been coded with a default response that returns a negative or normal value. As an example, in the first case during the first consultation, there is only one stream (see Figure 21 on page 60). If a user requests a serum triglyceride measurement, they will be given the information that the triglyceride concentration is 12.0 mmol/L. If another case had been undertaken on another topic, such as childhood food allergies, a consultation- and stream-specific response would not be available and the result would simply be returned as “normal”.

4.2.2 Relevance Scores

All actions, both diagnostic and management actions, are assigned a relevance score by the author of the case. The current values range from two to zero where; two is critical, one is relevant, and zero is not relevant. The use of just three levels was chosen to make the user task simple and quick. Critical items were regarded as those items that were critical to the diagnosis or management of the patient to produce the optimal outcome. Relevant items were considered to be those that were important to know about, but not essential to the patient’s outcome. Non-relevant items were those that were not necessary for the correct diagnosis and management of the patient. For example, ordering a skull X-ray is not usually relevant to someone presenting with a suspected myocardial infarction (heart attack). An electrocardiograph, on the other hand, is critical to the diagnosis. While three discrete categories were used, it was anticipated that there would be some uncertainty and disagreement over the classification of some items. This reflects the uncertainty inherent in the practice of medicine. Indeed, disagreement should be encouraged, and should be regarded as an indicator of the success of using such a component to encourage users to reflect on their actions.

The classification of each element of the history, examination, investigations, and management is determined by the author or authors of the case. Under normal circumstances the case author will be a domain expert and the decision of what is critical or relevant will be based on their opinion. However, this need not be the case. For example, a group of medical students could prepare a case for their colleagues. Under ideal circumstances, it would be useful to be able to classify actions based on published guidelines or best evidence. Unfortunately, such resources are not always available.

All default responses for diagnostic actions have been assigned a relevance score of zero. As a consequence of default responses having a relevance score of zero, any actions that return a normal response, yet are critical or relevant, must have a consultation- and stream-specific response.

As an example, a question that might be asked by a user during this hyperlipidaemia case is, "Do you have a past history of hyperlipidaemia?" In all circumstances, the patient will respond,

> *"I have had high cholesterol and triglyceride since I was a child but they have never been this high before. The triglycerides have usually been between 2 and 4 mmol/L while the cholesterol has generally been less than 6 mmol/L. They were last checked over 12 months ago."*

If this is asked in the first consultation it is assigned a relevance score of one. However, if it is asked in subsequent consultations, since this question should have been asked in consultation one, it is assigned a value of zero. Assignment of a score of zero is not meant to penalise the user, rather the score is indicative that the action was likely to have been non-contributory to the diagnosis and management of the patient.

4.2.3 Transitions between patient states

A summary of the state transitions is illustrated in Figure 21 on page 60. As indicated previously, the patient's state is determined by the consultation, and the value of the stream variable. The value of the new stream in the following consultation is calculated using a deterministic process based solely on the management options selected by the user and the stream values during the current and previous consultations (Figure 36). The mapping data required for this calculation is held in a database table and an extract is shown in Table 3. The dependence of the new stream on the value of the current and previous stream was included, so that users had an opportunity to change suboptimal management based on the feedback they received during the review sessions, at the end of each consultation. It was felt that users should be able to receive feedback, without their being an immediate adverse effect on the patient.

A very simple algorithm is used to calculate the stream value for the next consultation. Each relevant or critical management action maps to a specified stream. Thus if only one action is chosen, this maps to the relevant stream. However, if as is more likely, more than one action is chosen, then a decision must be made as to which stream represents the outcome for these actions together. In order to make this decision, a weighting, defined by an integer, is allocated to each action. Each time an action maps to the same stream, the weight values are incrementally summed. When all the actions have been chosen, the stream with the highest score is the one that is chosen. That is, the new stream is the one where $\sum Weight_{NewStream}$ has the greatest value. This approach is analogous to Multi-Attribute Utility Theory (MAUT), in which the overall evaluation of an object is defined as a weighted addition of its evaluation with respect to its value dimensions (Schafer, 2001). MAUT has principally been applied to decision analysis and estimating user preferences (Schafer, 2001, Von Winterfeldt, 1986).

Figure 36 is a diagrammatic representation of how a new stream is selected, based on the treatment chosen by the user. In this example, the user is in consultation two and stream Q. Suppose the user had been in stream P during the first consultation, then the stream sequence is “P:Q”. The weights given to treatments Rx-A to Rx-F are specific to the situation where the consultation is two and the stream sequence is “P:Q”. Suppose the user prescribed treatment Rx-A alone, then the Stream would be “A” in consultation three. If instead the user had chosen Rx-A, Rx-B, Rx-D, Rx-E, and Rx-F in combination, then the sum of the weights would be 0 (Rx-A) for Stream A, 5 (Rx-B, Rx-D, and Rx-F) for Stream B, and 4 (Rx-E) for Stream C. Therefore, in consultation three the stream would be B.

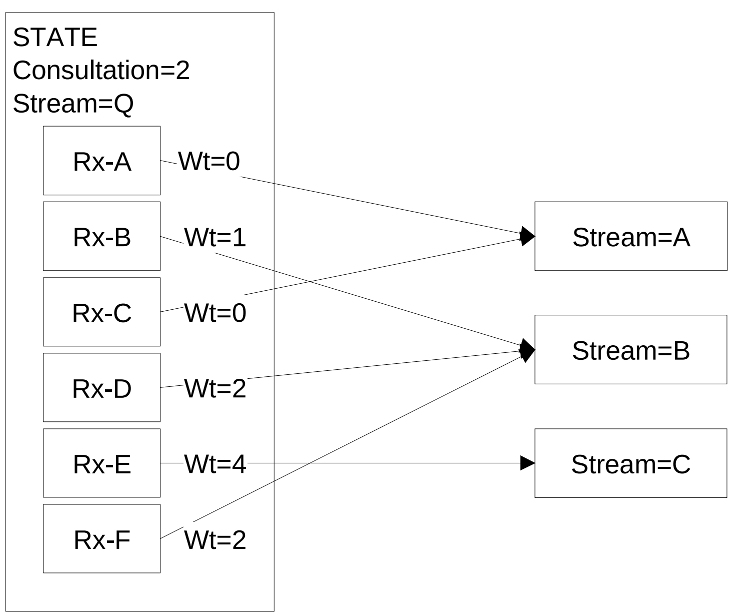

Figure 36: Mapping management actions to streams in the following consultation. Rx-A to Rx-F represent six different management options or treatments. Wt is the weight value given to each option. See text for examples.

Table 3 is an extract of the actual mapping table used SIMPRAC. In contrast to Figure 36, the streams within SIMPRAC are described by numeric values. This table shows the stream weights for the case where user is at the end of the second consultation, and the stream is 1. As there is only one possible stream in the first consultation, the stream sequence for the last two consultations is “1:1”. Suppose the user chose a diet low in saturated fat, 6 hourly cornstarch, fish oil, and a fibrate (rows one to four respectively). From the table we can see that a diet low in saturated fat is mapped to Stream 1 with a weight of zero, cornstarch is mapped to Stream 1 with weight zero, fish oil is mapped to Stream 2 with weight two, and fibrate is mapped to Stream 4 with weight four. In this instance the total weight value (column five) for Stream 1 is zero (zero plus zero), the total weight value for Stream 2 is two, and the total for Stream 4 is four. Thus the new stream with the greatest value is Stream 4. Therefore, at the third consultation, the user would be presented with the patient in Stream 4.

Table 3: Stream mapping data for the end of Consultation Two with a stream sequence of 1:1.

Action	Consultation	Stream Sequence	Weight	New Stream
Diet low in saturated fat	2	1:1	0	1
Six hourly cornstarch to maintain blood glucose	2	1:1	0	1
Use omega 3 fatty acids (fish oil)	2	1:1	2	2
Use a fibrate (Gemfibrozil)	2	1:1	4	4
Use a HMG CoA Reductase Inhibitor (Statin)	2	1:1	8	3
Allopurinol	2	1:1	0	1
Default	2	1:1	1	1

Using this simple algorithm has at least two benefits. First, it makes authoring cases relatively easy, provided the number of different patient states is constrained. The default mapping for all treatments is to Stream 1 with weight 0. Therefore if no effective treatment is provided, the case will enter Stream 1 at the following consultation. Using this system, the case author only needs to iterate through those treatments that they want to affect the stream selection. This must be done for each combination of consultation and stream sequence. The second benefit of the approach is that it enables the patient outcome to be directly related to the treatment modalities, thereby making explanations of patient outcomes relatively easy. An alternative to the deterministic approach would have been to enable probabilistic progression. While such an approach might have more realistically simulated actual patient progression, it would have lacked the beneficial characteristic of being able to directly relate treatment to outcomes, as noted above.

4.2.4 Alternative models and implications

One of the design goals for this application was for it to be able to be used as a template for a variety of chronic illnesses. While a more sophisticated disease model involving pathophysiological mechanisms and interactions that influenced the disease progression or control at each consultation could have been developed, it was unclear how well this would transfer to alternate clinical domains.

In line with the approach taken by the simulation software developed by the NBME (Clyman and Orr, 1990), careful consideration was given to the need to include a realistic time component. With such an approach, the review periods entered by users at the end of each consultation would influence the state of the patient. However, it quickly became apparent that this increased the number of potential patient states, and made determining what was relevant at a particular consultation unmanageable. For example, if a user wanted to review the patient after the first consultation in two months, the repeat measurement of triglycerides would be relevant or critical to the ongoing management of that patient. If, on the other hand, the user wanted to review the patient in one week, the repeat measurement of triglyceride would not be indicated. As a result of this type of complexity, it was decided not to incorporate strict measures of time within the model, other than that determined by the consultation number.

With the current model, the diagnostic action responses and relevance scores are dependant on the current state, while the management response is dependant on the current and immediately previous state. Having this dependency means that the disease model is simple but the number of responses and associated relevance scores for any particular case is potentially very large. This in turn raises questions regarding the scalability of the application and the ease with which additional cases can be authored. Despite the large number of diagnostic actions available to the user, the majority of responses for a particular case will use the default values. Nevertheless, the number of responses will generally increase proportionally with the number of possible patient states for a given case. Furthermore, the number of states is dependant on the anticipated effects of different management actions and the number of consultations. Therefore, when designing a case, care must be taken to limit the number of consultations, limit the number of outcomes for different management actions, or limit both.

The other limitation of the existing approach is apparent where combinations of treatment lead to different outcomes from any single treatment. For example, suppose treatment X leads to an outcome defined by stream A, and treatment Y leads to an outcome defined by stream B. With the current system, there is no method of mapping the combination of treatment X and treatment Y to stream C. Alternate outcomes from interactions between treatments a not overly frequent. However, they could be catered for by use of a rule-based system that considered these exceptions.

When developing the first case for the user evaluations, data was entered into the database using a very basic interface over a period of approximately one week. Development of an appropriate authoring interface would greatly simplify and accelerate the entry and configuration of additional cases.

4.3 Ancillary Functions

In addition to the two major models above, the application provides a number of additional functions. These include the medical record, user notes, current orders, current management, hypothesis lists, library resources, and free-text search engine for the history and management components. The medical record, user notes, current orders, current management, and hypothesis lists are associated with the user's status record (see Figure 32 on page 71), while the library pages are maintained as static hypertext pages.

4.3.1 Investigation orders and management options

The current orders and current management hold the investigations that have been requested and management options selected by the user. Each investigation order record is uniquely identified by the user status, and the consultation in which it was requested. Users are therefore able to review the results of investigations for the past and current consultations. This is in keeping with actual practice, where investigations are episodic and the results of investigations form part of the patient's medical record.

In contrast to the investigation orders and results, the management orders are only identified by the user status, and therefore only represent the current management of the patient. Historical management records for different consultations are generally not required by the user except during the review process, where they can be retrieved from the user activity log. Under normal circumstance, the clinician would record changes to the patient's therapy in the medical record. This can be done by the user, if they wish to do so.

4.3.2 Hypothesis log

The hypothesis log records the user's hypotheses for the current case and professional background. The log records the consultation, the stage of the consultation, and the stream in which the hypothesis was generated. In its current form, an audit trail of updates to the hypothesis statement, probability assigned to the hypothesis, or reason for holding the hypothesis is not maintained. On the other hand, records are never deleted. If the user requests that a hypothesis be deleted, it is merely inactivated and is no longer visible to the user. This was done so that this information would be available in the future, leaving open the opportunity to explore other ways of using this data to enhance learning.

4.3.3 Question and answer module

In both the history and management selection components of the application, the user has the option of entering free text questions or keywords that are then matched to those options held in the database. The free text searching is built around the Lucene free text indexing and searching package produced by the Apache Software Foundation (Apache Software Foundation, 2002). In the case of the questions database, the history_questions table contains a field for a full text statement of the question, and a second field containing keywords and alternate phrases that relate to the concept covered by that question. For example, the question, "Do you have a past history of heart disease?" is associated with the terms, "heart disease, ischaemic heart disease, coronary artery disease, myocardial ischaemia”. The Lucene package is used to index the text within these fields and is later used to search the index as users ask their questions. Information on the management options is maintained in an analogous manner, with a field for a text description of each option, and a second field to hold alternate descriptions and keywords.

4.4 Implementation

SIMPRAC was implemented using the following software.

1. MySQL database (MySQL AB, 2000)
2. Apache Turbine 2.1 framework (Apache Software Foundation)
3. Apache Tomcat 4.0.6 Servlet/JSP Container (Apache Software Foundation)
4. Lucene 1.2 (Apache Software Foundation)

SIMPRAC was originally implemented using version 3.23 of the MySQL database. This version was used for all the evaluations described in Chapter 5 and Chapter 6. SIMPRAC has since been migrated to MySQL Version 4.0, however, the database schema remains unchanged (see Appendix A).