My project was built to enable the current mentoring system for the Cherwell School's Year 11 scheme to use software for communication of meetings, targets and enable the external methods of contact from contact details. Through my project I have analysed the problem for which I designed, implemented, tested, documented and now evaluate a solution. In the analysis stage it is necessary to understand the aims of the project's solution. In the analysis I stated some objectives and the tasks which the solution was expected to perform (see *proposed system*, *analysis*). There were also other possibilities which were not implemented but were suggested in the analysis. In my objective evaluation I also comment on the effectiveness of the solution's component or how effective any absent features would have been. Further developments have been considered. These developments take into account the key needs of the system for its end users such as the readability and navigability of the system and how they could be improved. In the analysis I interviewed a mentor about the current system and asked for suggestions for the computer system. With my finished solution I consulted them once again for feedback on my solution (see the mentor's letter).

Objective evaluation

- 'The proposed system should use a facility for registering new users'

This feature would have been the most complex to include because there would be more tables to create, more pages to write and an additional user type would be required for adding new users of other types to StudentTable, ParentTable and MentorTable. Time constraints would have become unmanageable due to the increasing quantity of features to implement. This feature would have been highly beneficial to the system because the user may have been able to submit their details for the administrator to confirm that the user would have been eligible to be copied over to the mentoring scheme. Not every user would have needed copying as opposed to being added upon registration. Alternatively, the existing database with each student, parent and staff entry could have an additional field to flag the user as one participating in the mentoring scheme for Year 11.

- 'Each student may have at maximum one parent login allocated for a parent because only one parent for each student is sufficient in the process. However, there are instances where no parent is involved in the scheme.'

Unlike the original plan, there may be any number of parent login users between 1 and 999, but students may only have one assigned to them. Students may also not require a parent to use this system. However, the system will report to the users wherever a parent is not found. The targets are also designed to expect the parent to be present in addition to the student and mentor. Where there is no parent there could be a default configuration to disable any parent comments for targets where a parent is absent.

- Each end user may view only information which is relevant to them. Students may only view meetings which are directly connected with them; they are able to add comments to any of their meetings. Parents may only view their students' information and add parent comments to their students' meetings. Mentors may view only their own meetings with their students, but may also alter their meetings in case of mistakes with entering data. Mentors may also delete their meetings and

can add mentor comments to their meetings.

The SQL queries use the user's type and ID for viewing meetings, hence only those relevant to the user are returned and displayed. Meetings do not support the comment feature; the only opportunity is through commenting on targets. This explanation did not state that targets would support the comment feature, but implementing the feature to comment on targets enabled the users to understand the individual targets as opposed to the meeting as a whole, making target descriptions clearer.

- 'Each user may or may not use contact details: these could be an email address and/or telephone/mobile phone number.'

Contact details are indeed optional in the final system. However, there is no user interface built into the system to enable the user to alter their own details. In pages displaying the user's phone number and email address the contact details will be displayed if present. To improve, there could be a user interface to enable the user to add or alter their contact details.

- 'A feature which could be part of the proposed solution is the ability to enable the system to send automatically generated emails to only specified users, such as when a mentor fixes an error in their meeting, the student will receive an email regarding the alteration, while the parent will not.'

This was a feature which I decided not to implement since the design stage. This is because there was the concern that unnecessary emails would have been generated too frequently for notifying other users of changes. However, there could have been further features to disable this feature if permitted and to enable the user to permit the system to send emails to other users for certain actions, such as commenting on a target or adding a request for another user. This could have also included the use of functions and procedures because the sending of emails would use common properties for each purpose. Sending an email notification would use almost exactly the same code for each email, except the notifications would be about different processes which have happened, such as adding a request for someone else and deleting a comment.

- 'One objective this contact feature must meet is to be able to use readily available communication directly connected with the system to notify other end users about upcoming meetings/alterations and their updates.'

This objective I set was set to enable the users of the mentoring system to notify each other within the system through the contacts section. This would have been through the use of messages similar to that of commenting on a target. I did not meet this objective as I believed that there was no need to implement this facility. This was software designed to aid in communication within the mentoring system, not for discussing anything outside meeting-related topics. I also understand that contact between users may be done in various other ways, such as using email or telephone.

- 'The main objective is to provide an interface which the end users, students, parents and mentors, can use to directly communicate through to their target users

within an internal system.'

This objective is similar to the objective above, except that it does not involve events as opposed to the focus of direct communication. Here I have not implemented an absolutely direct method of communication. There is no internal communication facility within the contacts section of the program. There is however, the commenting facility for targets, although this is less accessible than it would be if implemented as a directly available area for communication within the contacts section.

- 'This enables each meeting to be localised within one location.'

This particular aim has been met because there is a database structure which stores data on the meetings and targets. This centralised system collects each individual meeting and target and stores it together in their respective tables. The user may access their meetings and targets, although they are required to be connected with either a student, mentor or both to be able to use the system. Students do not require a parent contact to use the system, but will receive notifications to add a request for one if absent. Students must be connected with their mentors in order to view their meetings. Given this condition is met they are able to view all their meetings and targets set by the mentor.

- 'So each user may review past meetings at any given time.'

This feature is also present in the system. The server I used (not the school server) for system testing enabled this. The school server, too, operates on a 24-hour basis, hence the data may easily be accessible from there. There may occasionally be slow responsiveness of the server or from within a particular network, but the system is reliable enough to be hosted normally.

- 'This eliminates the need to manually copy information and send to other users.'

This benefit may be subject to different circumstances. There is still the requirement to actually enter the data into the system, along with the possibility that there may not be a computer terminal within reach during the given meeting, so whatever is discussed and agreed may still require generating manual data, which is then transferred to the online mentoring system.

- 'Furthermore, meeting targets may be commented upon by any end user type and rectified in the case of error by the mentor.'

Having implemented a comments box for each end user type for any target I have achieved this objective. This is, however, almost the sole purpose of this feature. In contrast, this feature is not limited to notification of errors in the targets, but also perhaps to notify the other end users about any required materials for the next meeting. There are issues with communication in regard to other areas of the mentoring system because setting a meeting alone will not enable communication so no user may comment on a meeting if the wrong date has been set by mistake. Adding this communication interface in other sections would enable there to be more specific information for each purpose.

- 'This enables the student to focus on particular activities without the requirement for further meetings.'

This system does not aim to limit the number of meetings a student and mentor should have. The aim of the system is to enable the users to communicate to each other in regard to notifying each other about meeting-related issues. This is usually because the current system requires the student and the mentor to take a note of each action separately. The handwritten notes could then be lost, so there is less time required for the mentor and the student to share repeat information. The solution does enable the mentor to easily add a meeting and any targets within the meeting. However, the solution implementation does not allow just meetings to be commented on. A target must be set in order to comment on. So if a student and a mentor mutually agree on actions to take for the current meeting, the mentor must set a target for the meeting, otherwise the system cannot facilitate direct communication between end users.

Certain other features I attempted to implement were not included in the solution. Such features as automatic email generation and file uploading were proposed but not implemented as part of the solution. Email generation was not a high priority for this system because there would be the issue of unnecessary emails being sent to each user's email address. Unable to successfully implement a feature for uploading a file to the server I decided to not progress with this feature and continued through the project without it. With planning the file uploading system there were concerns about disk space consumption. Using email attachments students, mentors and parents may exchange files, hence the file upload feature would not have been a significant advantage.

Evaluation of computer system and further developments

Readability of the computer system was designed with a **sans** font in mind. Using a **sans** font is ideal in website readability because the user needs to be able to view certain details accurately, for example, the date of an upcoming meeting. I feel I have achieved this, although the system itself did not implement advanced text display enhancement, such as the enlargement of font size. Another feature which may have eased readability is the option to increase table border size so that meetings and targets are further separated from each other, increasing the space one can read in.

There was no use of metaphors in the system to enable a more visual representation of the system. There could have been the use of icons to supplement the user interface with the option to disable them. Including icons would have improved user appeal. There could have been images assigned with hyperlinks for each page. This would have enabled the user to select their option amongst a greater space as opposed to just a small portion of text.

Usability is intended for being as easy to use as possible. The end users were not expected to have advanced knowledge or ability to use a computer. The input pages were designed to restrict the user to only entering relevant information, such as AddMeetingPage, AddTargetPage and SearchContactPage. In each page there have been measures to control the method of input, allowing the user to understand the system quickly. No learning is particularly required of the system, except for the knowledge of the

structure of meetings. One area I feel have not emphasised sufficiently is the adding of requests. I did not clarify that once a request is added the requested end user must accept the request for the users to become connected with each other.

Navigability is not complex enough to require the use of an undo function or storage of content recently added/edited/deleted. Almost all areas are undoable with the use of cancellation of an action. However, when deleting data there is no undo function and when deleting a contact there is no confirmation. Clicking the 'Delete contact' hyperlink immediately deletes the user connection from the current user. There is no confirmation of this due to the time constraints induced from building the project components at such a late stage. This was partially contributed to by the contingency time used for addressing errors, such as the attempts to implement a feature to upload files.

Latency of the system sometimes causes issues in browsing the system because the server/networks could have slow response times. In some instances hyperlink requests fail and may require halting page loading/refreshing/re-requesting the page. If the system were to implement a user interface specifically for loading other pages there could be a loading page which uses a reference table for each page the user entered from and which page the user requests for. A user interface must be included for troubleshooting purposes, such as the page failing to load or if the user wishes to cancel their page request. This would enable the user to return to the previous page if further page requests fail. This loading page may also display loading time in seconds each time this page is called. A significant drawback to this feature is that page requests would increase, using more bandwidth and the use of browser buttons such as the 'Go back' feature would become unusable.

When recent meetings/targets have been added there is no notification about this to other users. As opposed to implementing the email feature there is also the use of a notification system built within the mentoring scheme software. As each meeting and target is added there could be notifications which are displayed to inform the user of these events. This could be displayed in another column. ViewContactPage and ViewMeetingPage could, in theory, be merged together, or remain separate, but using them together in one page, such as the main page. This would result in fewer pages, reducing the loading time required during overall use of the software. However, those intending to view only meetings or contacts in one session may have a small rise in loading times each time the main page (LoggedInPage) is loaded.

Another further development which would improve the programmed solution is that it includes a communication interface between different end user types within the contacts section. While this option was considered, the reference to external methods of contact was used above the direct communication interface. If the direct communication could also be built communication between users would be possible from within the system and not through commenting on targets. This could be useful if the mentor has not decided a date for an upcoming meeting. When upcoming meetings occur at pre-defined intervals there could be a feature to add meetings corresponding to particular days, such as on Mondays every two weeks.

Some existing features could also be improved to become more useful such as allowing

meetings and targets to only be on weekdays, and to set targets to be due in only after the meeting has happened, such as when a meeting is planned a week in advance but its target is due in 3 days before the meeting. This would require resolving, but due to the project being in such a late stage it is beyond possible to rectify this issue now. This would have potentially overridden the validation where the date must not be earlier than the current day, but not before the meeting has happened.

Using SQL queries for verifying if a user is viewing the correct page with the correct details loaded I found that the system performed less efficiently than otherwise possible. The issue with this is that each time a user browses a page with automatic forwarding of variables there is an SQL statement used for security purposes. There may be the potential to have separate tables for each mentor's/student's meetings and targets where there is no such requirement as the database automatically searches a table. This cannot be easily tampered with as GET requests are. The POST method was inadequate in forwarding information between pages. This results in the lack of efficiency with using GET and POST methods in forwarding data, together with searching a universal meeting table for all meetings, along with the table for targets.

A feature which may be extremely useful which I overlooked during completion of the project was to display availability dates and times which students and mentors would have used. This could involve one-off occurrences or be based on timetables. This would enable mentors and students to have schedulable meetings.

My solution to the mentoring scheme is too primitive to remain a complete solution during a long-term period. If the mentoring scheme develops and expands, the computer program would require modification in future. Such issues as having small mentor groups as well as one-to-one mentoring may be introduced. The system would require some significant changes to allow there to be group meetings as well as one-to-one meetings. There would then be the necessity to set targets for individual users within the meeting.

Feedback from other users

I introduced a current Year 11 mentor to the software and demonstrated it to them. Testing the system in the presence of the mentor was beneficial because they understood what tasks the user is able to perform within the system. In testing I was required to log in as other users because I demonstrated all the capabilities of each user. After the demonstration I asked the mentor to give me some feedback. This is given to me as an emailed letter.

The mentor began with a suggestion to add a visual theme and some formatting of the look and feel of the system due to its lack of theme. A margin could be added for browsing other pages so that pages higher in the hierarchy can be browsed directly from any other page from the derived path, not just advancing page-by-page. So after viewing a target the user may wish to immediately view the homepage; a hyperlink to each earlier page is available. A background could be added for each web page, icons could be used as hyperlinks and text formatting could be more sophisticated so that certain words are displayed as bold, italicised and underlined, as well as altering the font and size.

Their second suggestion was to replace the display of ID numbers for the student and mentor with their names in ViewMeetingPage. This would convey meaningful information to the user and this would also enable the viewer to browse more efficiently, particularly mentors, who may be required to use various areas of the system with the need to access all their information quickly and efficiently. Parents and mentors with several students would also benefit from this because they would see their student's name and the mentor's name for each meeting.