Progress report

**01/08/2014**: Begun work on opening screen and splash screen

I considered using custom text and background color but found it too unpleasant and decided default white on black

**14/08/2014**: I fleshed out the basic patient module and begun work on the administrator section for resetting the ID File

Encountered a problem with the admin menu displaying an error message prematurely. After testing I have discovered that if works if the module is called straight away upon compilation but the error message appears if it is called through the menu. Couldn’t fix by the end of the lesson bears further investigation.

**15/08/2014:** Managed to develop a workaround for the issue, the problem still occurs and I don’t know why but I put a clear screen in the else of the log in and it removes the error from the screen before the user can see it. The fix isn’t pretty and causes slight issues with the invalid username else not appearing when the wrong username is entered but it will have to suffice.

Exiting to the main menu from a procedure resulted in messed up formatting but I fixed this by inserting gotos and labels into the main menu resulting in it.

Testing my patient module brought to my attention the fact that when I tried to add a new patient it crashed. After looking through the error messages I discovered it was due to the fact that a necessary file had not yet been created.

Following on from my previous error I discovered the admin subroutine responsible for recreating the ID file got stuck in an infinite loop. Adding a simple writeln to the if seemed to fix it.

Fixing the ID File stopped the patient module crash but the actual creation code doesn’t work. After comparing the terminal with the code I discovered it skips a readln for some reason. Adding another readln to the top of the add patient module seemed to fix the problem.

I entered a lot of comments explaining the add patient module in more detail

I have now fully filled in the add patient module with all the fields for patient details as outlined in the task sheet.

I fixed the add patient module so it now writes all the details just entered to the console so that the user can verify that everything is correct, however I still need to add a verification system to the module.

**19/08/2014:** I added the ability to re enter a patients information if there is an error during entry. I have now begun work on the function to list all patients filenames saved with the program.

The array seems to work now I just need to see if it can read from the array.

**20/08/2014:** I wrote some more comments and begun work on writing the array to a text file. For some reason it only writes the most recently used array to the text file so needs to be fixed.

I got the array to read from the file correctly by replacing the for loop when reading with a while not end of file but when writing it blanks out the file.

Managed to get it to not blank out the file but it doesn’t write the entered file name to the array. After investigation I discovered the new file name is written to the array correctly it just doesn’t write to the file properly. I did something and now it works so problem solved.

**~~This marks the completion of the patient module~~,** however at some point I need to go back and produce a GUI? Like that which is used in the rest of the program, for now I would like to get the rest of the programs functionality working though.

I added an admin routine to reset the File name File and added a directory within the program folder to store patient files so they don’t clutter up the main folder. Adding the directory proved to be a lot tougher than expected with the variable filename but adding a + sign made it work, I guess it joined the strings together or something?

**21/08/2014:** Today I added the list patient procedure which lists all the IDs contained within the File name File. I also begun work on the edit patient screen and have figured out a way to individually edit a field without having to get the user to reenter every field.

**22/08/2014:** I continued working on the edit patient module. I added the ability to edit another field at the end if the user wishes to. This means all the functionality of the edit patient module is there, I just have to figure out a way to stop the program from crashing if the user enters an invalid file name. I did some research and discovered the IOresult function which I used to make an error checking bit that stops the program crashing if the user enters an invalid file name. This means all the functionality of the edit patient module is now complete and I just need to add a GUI later.

I completed the remove patient module; it was easy because I was basically just recycling code I had already written. I also fixed all the double enter bugs but in the process discovered my list patient module still displayed deleted patients. It took a while to fix the list patient still displaying deleted patient file names problem because it was a lot more complicated then it seemed. However after some research I discovered T string lists which enabled me to solve my problem efficiently in a measly six lines of code, the discovery of T String Lists may enable me to clean up other areas of my code and is worth investigating.

I also added the view patient module which was the last of the patient module to be added.

**26/08/2014:** Today I begun work on the doctor module by adding the add new doctor procedure.

**27/08/2014:** I finished the doctor module by adding the edit doctor procedure which is just a copy of my edit patient module with limited functionality.

**28/08/2014:** I started work on the scheduling section of the program by adding a module in the admin menu to initialize the day that will reset all the files. I still need to finish the admin initialize module but I got it started.

**29/08/2014:** Worked some more on the initialization routine in the admin menu. I finished the admin routine today and continued work on the adding an appointment procedure. I finished the appointment procedure and just need to test it out. I also reformatted my admin routine for initializing the day to be more in line with the appointment procedure.

**2/09/2014:** I worked on the view appointment procedure. I cleaned up the add appointment module by making it create the new appointments in the appropriate folder. At some point I should go back and make all the created files .txts, Ill probably do it at the end. I converted all my appointment files to .txts. I fixed up some stuff in the add appointment module and added a error catching method in case of user error with entering the wrong time. I finished the view appointment module and now just need to do the print daily list, this may lead to me redoing parts of the appointment module depending on how I decide to do the print module.

**3/09/2014:** I formatted the view appointment module output so it was easier to read. I also fixed the reset ID file admin routine so it cleared the file name files otherwise there would be errors, realistically I could have just fixed the list all patients procedure so it didn’t rely on the ID file but the only reason you should be clearing the ID file is if you are deleting all patients anyway because with a reset ID file duplicates could happen that would result in lost data. When working on my print daily appointment list I discovered it crashed randomly. Something changed and now it doesn’t crash but it only reads the last entered appointment time and only writes that one to the console. I utilized some T string lists and fixed all the above issues and got my print daily list working which marks the completion of the functional side of the program I believe unless I discover some bugs later.

**4/09/2014:** After reviewing my code I discovered my view patient things were not working, need review

**5/09/2014:** After a bit of review I discovered the issue was a simple assign problem with me forgetting to add the .txt to the end of the assign line.

**6/09/2014:** Today I basically reformatted most of the program starting with the add patient module. Everything sorta works but there are irritating glitches with pascal and readlns that cause severe formatting issues. I managed to fix the formatting issues by sticking another clear screen after the readln necessary to stop line skips. I realized after I had re done my patient module that even though the actual files containing wrong information were deleted straight away there file names were still been recorded in the file name file which gave the impression when listing all patients that they were still there. This was an easy fix that involved simply moving the appropriate code to after the confirmation screen to make sure that the file was correct before saving it to the list of files.

The process of testing this led me to realize that I had not coded for a situation in which there were more then ten patients, this meant that with large numbers of patients not all the names fit on the list patient screen. This should be an easy fix however. I made a fix to the patient list problem that seems to work well so far. I also cleaned up the list patient module by adding a small GUI type thing. I adjusted all the delays in the screens because it was annoying me. I have been working on overhauling the edit patient screen so it has a GUI but have encountered a strange error where when you select to edit the patients title it instead reads the patients first name. After a bit of editing it now reads the title but doesn't want to read for a first name. I discovered the program will always read right on first load but if you try to edit another field after that without closing it down it doesn't work. Ok I discovered that the error was the result of pascal skipping a readln again, this bug is the bane of my existence. I fixed the problem by making all my reads in the edit case into readlns. This might be the reason for the bug, worth investigating if I have time. I finished the edit patient procedure

While working on fixing up the delete patient module I encountered yet another skip readln error, I tried fixing it by inserting another readln which worked but created other issues so I had to come up with another way, I managed to fix it by inserting a small if using a counter that only ran on the first go through of the module, meaning it fixed the initial skip line but doesn't cause other issues. This was the completion of the remove patient module. I redid the view patient module to be more GUI like, this should be the completion of all the patient modules but ya never know.

**7/09/2014:** Redid the add doctor module with a GUI, I did discover one slight issue with the formatting involving small ID’s messing up the spacing but I made some code to fix that if it recognises an ID less then 10 by inserting a space in the line. I redid the edit doctor field to have a GUI and the only problems I encountered were the same ones I had in the edit patient redo but I just redid what I did to solve those and it worked. I am not 99% sure all my readln skips are due to using read so I have been replacing all instances of read with readln as I go through which has solved most of the issues. Replacing the reads worked so well I had to go back and comment out my previous fixes to the line skip bug because it was messing up the flow, but that's a nice problem to have.

I finished the doctor modules. I formatted the appointment modules to be easier to read and use but didn't make a GUI due to time constraints.

**Program finished**

I wrote the user manual. I wrote supporting information on the program.