**Testing Plan and results**

For the entirety of our time building and updating the code, we have always tested to see if upon completion, the code works. For example for every method we completed we tested. And ultimately had a final examination of the code at the end. However, as most programs, we came across errors and had to fix or adapt the code to them.

First of all, after combining our code to make the finished program we tested to see if the files would ‘load’ correctly. Upon testing this we did not come across any initial errors and all files seemed to be entered correctly into the right arrays. We then quit the code immediately to test if the arrays correctly save even though unedited which was also successful. After relaunching the code we tested to see if the login systems were functioning correctly. This proved to be mostly successful however the dentist log in, if details that were entered were wrong, would come up with an error exception. This was quickly fixed and now functions properly.

Our next job was to see if all the admin methods worked correctly. Most of the methods worked up to standard and functioned correctly. For example the add/remove patient and dentist functions worked as it was supposed to. However the change working days for the dentist method was not functioning. We kept trying but could not solve the dilemma. So in the end we came up with a way to adapt the code in the calendar to solve this issue, resulting in us unfortunately having to leave out that method. Other than that there were no major obstructions.

Then we moved on to the next sub menu which was the dentists. The dentist’s sub menu had the least amount of methods used so we did this in fairly reasonable time. With this sub menu we ran into two errors while testing. One was that the add treatment function would not function correctly as it did not save to the treatments.txt file. We concluded that this was because of its array that could not be accessed by the save method and therefore would not work. We quickly corrected it and moved onto our next problem. We could not get the previous appointments method to work on its own, so we combined its code with the upcoming appointments method and managed to get it to function together. Other than that the change dentist details and other methods such as list treatments given worked perfectly.

Our final check was on the patient sub menu. This was done by accessing each one of the methods as we had done before with the other sub-menus. This had the biggest problem we had to overcome. We could not manage to get the request appointment working which in turn jeopardized the whole calendar code which is used by multiple methods. As stated this was the largest error we had to overcome. We eventually managed to come up with a solution. We completely reworked the calendar to use a hash map instead. This solved our problem even if it was a big job. After that there were only miniscule changes to do such as changes displayed text to make more sense and to guide the users.

The was one last thing that needed to be done was the encryption. We only managed to encrypt the hash map data which had private information on, however the others also included some like the details which we could not figure out in time.