# Testing

Testing is extremely important to the success of any software development project. A thorough testing strategy increases robustness and customer confidence within an application. As the application is used to deliver information that will be used by medical staff, it is vital that the application is well tested, as the consequences of poor testing could be fatal. Testing is one of the fundamental steps within the water model and therefore by choosing this methodology testing was enforced. This chapter will discus the approach to testing used throughout the project and also describe each test strategy used in detail.

## Approach to testing

Testing was used throughout the implementation process to help create a robust and effective application. Originally it was planned to use a test-driven development throughout the entire implementation process, but this was later found to be detrimental to the progress of the application. Although test-driven development was not used throughout the implementation progress it was used when developing classes that had could potentially be fatal, such as the calculation classes.

Throughout development the application was ran on a real device and any new features were thoroughly tested, using a variety of behaviours and carefully monitoring the applications state for any anomalies. This approach to implementation allowed for a well-tested application, before any test strategy had been complete.

After the implementation phase of the project had been complete, thorough testing of the application was executed. Unit tests were written for all classes that had not already had unit tests created. User interface tests were created and executed using android activity unit tests. The application was also stress tested by using Android’s exerciser’s monkey application.

One the tests had been written, several devices of varying setups using the Genymotion emulator, the full list of tests were then executed on each individual device. This ensured that the application runs well on an array of phones and tablets.

## Test database

When the testing process began an extra parameter was added to the database constructor, this parameter would allow the application to open a separate database that was identical to the actual database, this separate database could then be used during tests. Having a separate database whilst testing meant that the data could be manipulated without effecting the main application. This was needed, as downloading a new set of data after each test would take a long amount of time.

## Unit testing

The Android SDK provides classes for unit testing your application. These classes are based off the JUnit framework, they add extra feature such as the ability to access the applications context allowing you to test the database of the application.

Unit tests were written for every class of the application, testing each public and protected method. Most tests contained multiple assertions, testing that the expected output was returned when correct information is entered and that an error is raised when the incorrect data is entered.

As the unit tests have now been written for all classes, should a developer in the future make any changes to the application, they can execute the unit tests to ensure they have not broken anything. As the units tests will be provided to the NHS with the source code, they will be able to execute the unit tests for themselves, allowing theming to verify thorough testing was carried out, thus increasing their confidence within the application.

As Android runs on a large amount of devices, unit tests are useful for quickly testing that the application runs as expect across the Android platform.

## Bug that was found due to unit testing

Test data had been gathered for testing the calculator class. This data was then implemented into unit tests. To ensure the class was well tested a wide spectrum of values were used, including both small and large values.

Within the original implementation of the calculator class I had used floats to store all values and results of the calculation. After executing the unit tests, most tests were succeeding, but few were returning values slightly off the expected value. As the dosage given to patients must be correct to good level accuracy I began investigating the cause of the problem.

It was found that the issue lay with the data types used, the NHS’s website used doubles whereas I had used floats. Due to doubles being more accurate than floats, the data types were changed to doubles. The unit test were then executed again and all tests succeeded.