Unit tests

Unit testing is a type of software testing in which individual components/units of a software are tested. This allows for validation of the functionality of the individual components. There are various benefits to unit testing;

* It increases confidence in changing/maintaining code. When well written unit tests are run on every system change, errors can be caught quickly preventing these changes from being added to version control.
* It makes code more reusable. This is as a result of the forced format which code is forced into in order to make unit testing possible.
* Generally, the cost of fixing an error is lessened in terms of time and effort for unit tests in comparison to normal debugging. The is because unit tests make the location of the error immediately apparent.
* It provides a basic documentation of the system. Reading unit tests gives developers a better understanding of the features being tested within the software system.

As a result of these benefits, using unit testing for the system made sense. Unit tests focused on the classes of the system due to their extensive use throughout it.

Unfortunately, as a result of the design of the DatabaseComms class, adding unit tests for its methods wasn’t possible thanks to the methods of the class requiring call-back functions. Likewise, unit testing the front-end forms wasn’t possible due to the format of them not being suitable for unit tests. During planning of the system, unit testing was somewhat overlooked, and as a result code coverage suffered.

<http://softwaretestingfundamentals.com/unit-testing/>