## Testing

### Unit testing

Before we started developing our program we had set up an architectural model of the interaction between the different modules in our program. And set up unit tests for the functions that were did not seem redundant to test ( getting and setting fields, adding and removing items from lists etc. ).

We have also overridden .Equals() methods for many classes to make testing easier, as it allows for easy comparison our custom classes. This was not done for all classes though, as comparison of fields like DateTime proved impossible because of the way the objects are created. When created by the DateTime.Now property the objects “Kind” property is set to local, along with having detailed information about milliseconds etc. These properties are not able to be saved and read when we write and read files.

As the development continued, and we neared the deadline with every passing day, unit testing was overlooked and instead intensive debugging was done to test corner cases, program states and more.  
This was done for all functionality in our program, in all later sprints, but we have very little documentation on this, and as a consequence, we cannot provide a satisfactory amount of unit tests for our project.

