**Definition of Done**

**User Stories**

* The high level behaviour should match what was defined by the user story
* Acceptance criteria should be defined in the format GIVEN .. WHEN .. THEN ..
* The system should handle possible common edge cases with helpful error messages
* The completed user story implements a self contained feature
* The completed user story has been peer reviewed
* Major refactoring changes should be justified and documented
* All tasks for the user story must be completed
* Code must be merged with the develop branch

**Sprint Tasks**

* All unit test cases must pass
* Manual testing must have been carried out and must pass
* Completed task must be peer reviewed
* All the code must be peer reviewed
* All the code must adhere to the specified coding style
* All the code should have a reasonable amount of documentation
* All dependent tasks may begin after completion of the “prerequisite”
* Obscure pieces of code should be documented properly (or avoided)
* Other team members should be aware of the changes made
* New tasks should not break existing features
* The task must be merged and pushed to the Github repo after completion
* Excessive commented code should be removed
* Android Studio should not raise errors or warnings
* Gradle must build and sync the project with no errors or warnings

**Unit testing**

* Unit tests should be written to cover all Java files
* Each unit test should be able to be run independently
* Each unit test should not rely on other unit tests or the order which it is run in
* Each unit test should have the necessary documentation describing what it tests
* Each unit test should only test one thing
* Edge cases must be tested
* Multiple normal cases must be tested
* Failing cases which may throw exceptions must be tested
* Unit tests should be written and run using JUnit on Android Studio
* Manual testing should be carried out and peer reviewed
* All Android specific code (not pure Java) should be manually tested