Contribution Guide

Code of conduct

Reporting Bugs

When reporting bugs, the incident should be reported in the #bugs channel within the slack workspace for this project. The person reporting the bug should give a general overview of what happened and what the leading steps to causing the bug to happen are. The version of the project should be posted which can be found through GitHub. If the bug is a visual problem, then screenshots should be uploaded along with any relevant annotations.

In order to track bugs, each bug should be given its own ID number which will simply be a count of how many bugs have been reported so far. When posting about a new bug the bug number should be worked out based on the previous messages posted. All team members should keep up to date with the bugs channel and should post a response to the bug if they are responsible for the code segment that has caused the bug.

When a bug is fixed, it should be pushed to GitHub with the bug ID being part of the commit message and a FIXED message should be posted in the slack channel.

New Enhancements

New enhancements, ideas and any feedback from members of university staff should be posted in the #improvements channel in slack. This channel should be dedicated to discussing creative ideas and monitoring which improvements need to be made after each deliverable. The person who presents each deliverable is responsible for posting and feedback given from their lab tutor. The project manager should be consulted on any major decisions that need to be made based on feedback, such as taking the project in a different direction.

Continuous Integration

For this project, we will be using continuous integration methods for merging changes to the project. It is encouraged to push changes to the master branch of the project when each separate feature is completed. We have chosen this method as all members of the team will be contributing to programming components in the project which need to be linked together as shown in the design diagrams. Having multiple small commits allows

When making a commit to the master branch, it is important to make sure the repository is pulled before the updated version is pushed, this eliminates the chance of data being overwritten as a merge conflict will arise. Merge conflicts are to be fixed by the member who is trying to push to the repository. When making a commit, the commit message must be relevant and describe exactly what has been achieved in the session. It is encouraged of members of the team to push at the end of each session of programming in order to update the team on progress, this also helps avoid merge conflicts as the cloud version is always up to date.