## Development Standards

|  |  |
| --- | --- |
| **Guideline** | **Reason** |
| **Development** | |
| Code is well commented and each block of functionality is identified by start and end comment | Enable better and faster collaboration.  <!-- MESSAGING SECTION START-->  [code] <!-- MESSAGING SECTION END --> |
| Code is tested to work and looks as expected while testing on personal I drive (stu-iis server) before committing to GitHub | Helps reduce issues, mainly with front end and php errors - check for these, not thoroughly |
| No unnecessary id's and classes are added to HTML elements if not used in scripts | Deleting these while refactoring can cause issues and delays for checks if the id and class is actually going to break functionality |
| Work on a task for a specific story branch is to be completed in the branch directly - do not duplicate folders that need to be merged back into one | This almost certainly leads to merge conflicts that need to be manually resolved or work could get lost |
| Restrain from using unnecessary CSS externally in the style.css file, inline or internally | The framework we use is bassed on ready to use classes and keeps design consistent |
| If CSS is needed for custom elements it is only added to external file css/styles.css - this should be shared for all pages | This applies to element styles not available from the framework |
| Only commit high-quality code and closest resemblance of front-end to designs unless discussed with team to change | Not half done, refactoring takes longer than implementation from scratch; front-end looks exactly as per design prototypes or any limitations are discussed before approval for changing design |
| Branches for new stories are created from the master repo only when the story work starts | Keeps the branch up to date with latest completed feature code; ensures developers work on the latest code |
| Sync before committing changes to GitHub | Helps avoid change conflicts between existing and updated files |
| All team members are familiar with Google Material Design's principles |  |
| Team members have read and know how to apply the MaterializeCSS framework | Makes front-end implementation faster |
| Do not commit code straight into master in development repository |  |
| Do not commit bulk of features at once - commit small bits and sync regularly before continuing work | holds off other members' work, creates conflicts due to many days without syncing |
| Collective sync at the beginning/end of each successful sprint so everyone's files stay up to date | Fixing errors on obsolete files loses time |
| Inform the group whatsapp before commiting | Allow everyone to sync if necessary keep files up to date |
|  | |
| **Documentation** | |
| Before sending in documents for review or submission - read through them and ensure they are error-free |  |
| Front-end designs are kept up to date |  |
| Database schema docs are kept up to date with implementation |  |

Available at: <https://docs.google.com/a/greenwich.ac.uk/spreadsheets/d/1DoK7uvpsASj3ydNKW4Aw8kBGHvYd6CNVG57Ap81v8dU/edit#gid=0>