

Sodalicium: Reformed Rabbit Requisitions
Routine
Coding Standards

Oluwatosin Botti, Tlou Lebelo, Baven Pavaday and Ryan Hartley

May 2019

Coding Standards

The Following Coding Standards are put in place to ensure code written has a uniform style, clarity, flexibility and is reliable and efficient

Naming Conventions

- Variable and function names should use camelCase.
- The names of variables must be descriptive so it would be easy for anyone looking at the code to know what the variable contains.
e.g. var **textReceived** rather than var **x**
- The names of functions should be descriptive of what the function does.
e.g. function **processImage()** rather than function **func1()**

Formatting Conventions

- Code should be well indented so it will be easy to read, understand and maintain. Ensuring that the code is clear.
- Where brackets are needed, for example for functions or loops, the opening bracket should be on the same line as the function name and the closing bracket should be on its own line at the end of the function.

In-Code comments

- In-code comments should be included where the code may be complex and need further explanation.

Git Repository Structure

The current GIT structure consists of several branches. These Branches include the Master, which consist of the working prototype. The other branches are named after the different subsystems such as the GPS, OCR, Interface which are still in development and are being worked on in those specific branches. There are also folders for the different subsystems in each branch where integration testing can be performed between the different subsystems before the branches are merged with the master branch. Finally there is the readme.md document which gives an overview of the project.