


Redefining the Use of Augmented Reality

Code Standards

Version 1.0
24 August 2015



Jason Gerbes
1274664

Joshua Son
1388288

Paul Lee
1264218

Sean Young
1302108

Contents

0.0 Version History	3
Version 1.0	3
1.0 Code Standards	4
2.1 Naming Conventions.....	4
2.2 Brackets { }	4
2.3 Indentation	4
2.4 Variable Names	4
2.5 Access Modifiers.....	4
2.6 Comments	4
2.7 Peer Programming.....	4

0.0 Version History

VERSION 1.0

Version 1.0 is the original version of the Development Standards document. This version of the document was created as part of Sprint Zero: Development Setup.

1.0 Code Standards

The code produced during the LuminAR project must adhere to the common code practices outlined in the sections below.

Sample code:

```
class SampleClass extends SuperClass {
    private int sampleVariable;

    public void sampleMethod () {
        // This is a sample comment
        System.out.println("Hello World");
    }
}
```

2.1 Naming Conventions

Fields and variables are to follow the camelCase naming convention. Classes are to follow the TitleCase naming convention.

2.2 Brackets { }

Brackets shall be placed on the same line at the end of the method declaration. The following code is to be indented, and a closing brace is to be placed on a new line, following the enclosed method.

2.3 Indentation

Indentation helps improve readability and ease of flow understanding. Code enclosed within methods and statements is to be indented beyond the method/statement declaration.

2.4 Variable Names

Variables must be named appropriately. Variable names must be clear, meaningful and understandable by someone unfamiliar with the code. It is essential that variables are named suitably within a collaborative project.

2.5 Access Modifiers

Variables and methods will always be given the most restrictive modifiers possible. It is important that public access is kept to an essential-only basis to ensure undesired modification.

2.6 Comments

Code is to be commented wherever its direct/implied functionality is not immediately clear. Comments are to be added to aid in the readability of the code and must be correctly formatted.

2.7 Peer Programming

Whenever possible, peer programming practices should be followed. Peer programming ensures code quality as the likelihood of error detection is greatly increased. When peer programming is not possible, the new code must be cross-checked by another developer before being added to the final product.