COMP 3350 Group 3 Programming Standards

1 All files shall begin with a explanatory header

```
/**
* File name
*
* COMP3350 SECTION A02
*
* @author
* @date
*
* PURPOSE:
*
**/
```

2 If their purpose is not obvious, methods will have be preceded by an explanatory header

```
/**
 * Function name
 *
 * brief description of the function
 * @param {datatype} parameterName1 - parameter description
 * @param {datatype} parameterName2 - parameter description
 * @returns {datatype} Description of the returns value
 */
```

3 Non-obvious code chunks will be commentated

4 Contents of classes will be grouped consistently

The following order is recommended:
class Something {
// Instance variables
// Static variables
// Class-level constants
// Constructors
// Other instance methods
// Static methods: main comes first, if it exists
}

5 Classes will be in the package scheme that most corresponds to their purpose

6 All handlers and objects in logic / database layer will have interfaces that have names that begin with “I”

7 Consistent indentation to clarify control structures

8 No magic numbers, all constants will be assigned a variable

9 Variables have as small a scope as possible

10 float and strings not used with == or !=

11 for loop values not changed inside the loop

12 Best possible loop construct used

13 Avoid unnecessary duplication of code

14 Classes named in the form “ClassName”, constants are “ALL_UPPER_CASE” and all other identifiers are camelcase “otherNames”

15 meaningful variable names

16 Proper separation of responsibilities between layers

17 Access specifies as specific as possible

Punishment

Any group member caught violating these standards will have their name and code in question added to the wall of shame

🔔 Wall of Shame 🔔

Name: Jillian

Violation: Getting called out by Rob for having ugly declarations in the UI (if you're gonna have ugly code at least hide it better smh).

Code: