**SOEN 6441: RISKGAME BUILD 2 DOCUMENTATION:**

**CODING STANDARDS**

**GROUP MEMBERS:**  
**40087621 An Nguyen**  
**40104630 Naga Satish**

**Basic coding rules:**

This refers to the set of rules and guidelines to be followed when coding as well as other factors like comments, declarations etc. Not only is a good coding convention best practice it also ensures the readability and manageability of the code. As a team the follow coding conventions were agreed upon and used:

**Naming convention and Capitalization:**

* Use descriptive name for all variables, methods name, constants and any other if any
* Class names always start with an upper case and for every next word in the class name. For example: RiskBoardModel, RiskGameModel, etc…
* Class name must be the same as the file name.
* Variables name always start with a lower case and for every next word, start with an upper case (Camel case). Except for constant variables (final keyword) that are predefined in RiskGameConstants class, the name of those constants will be all upper cases.
* Methods name also start with a lower case and to separate the next word, that word will start with an upper case.
* No use of hyphens or underscore to separate multi-word variables.
* Use constant for the string data, if it is used more than once in the code.

**Javadoc commenting conventions:**

Javadoc comments will precede every class with descriptive comment about the purpose of a particular class. Include @author notation to indicate who is in charge of that class. One line for each @author notation if there are more than 1 team member working on it.

Javadoc comments will also precede every method with descriptive comment about the purpose of a particular method. The Javadoc comments should include the following:

* A description of the method
* The name and the description of each parameter (Will not present if the method does not have any parameter). One line for each parameter if there are more than 1 parameters.
* The name and the description of the return value (Will not present if the method does not return anything).
* Exception throwing if there is any. If the method has already included the catch exception, it is not necessary to include them in the Javadoc commenting.

An example:

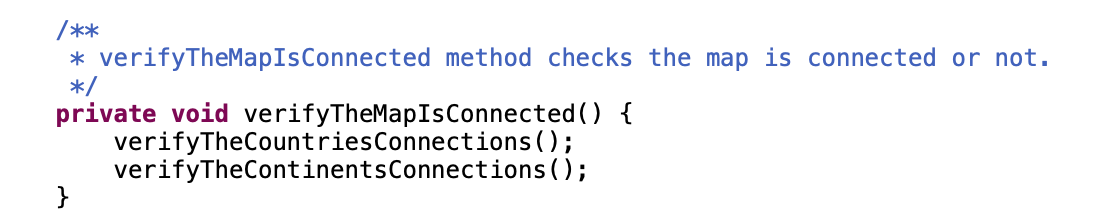
Une image contenant capture d’écran

Description générée automatiquement

Une image contenant capture d’écran

Description générée automatiquement**Spacing:**

Always put a line between two method within a class. For example:

Une image contenant capture d’écran

Description générée automatiquementAlways put a space before every “{“. For example:

**Indentation:**

* Indent a space equals to a tab key when beginning a new block.
* Open braces (“{“) will begin at the same line of the method name
* Close braces (“}”) will start at a new line.

For example:

Une image contenant capture d’écran

Description générée automatiquement