

## CS 192 Software Engineering II

### General Programming Standards and Guidelines Code Review Checklist

<b>Reviewer's Name:</b>	Miguel Luis Posadas	<b>Peer Review Date:</b>	April 30, 2017
<b>Project Name:</b>	<b>Wi Finder</b>	<b>Project ID:</b>	
		(if applicable)	
<b>Developer's Group Name:</b>	2 Sabaw Guys	<b>Sprint Number</b>	5
<b>Review Files &amp; Source code</b>	home.html		
	results.html		
	views.py		
	wifi.html		
<b>Code Approved</b>			

This checklist is to be used to assess source code during a peer review. Items which represent the code being reviewed should be checked off.

Refer to the *Android and Java General Programming Standards and Guidelines* (<https://source.android.com/source/code-style.html>) document for more complete descriptions and examples of the items listed below.

Place a check mark if the code complies with the criteria. If not, place comments at the remarks column.

#### 1. License

Criteria	Comply		Remarks
	Yes	No	
License exists at the top of the code	●		
License contains the name of the author		●	
License contains the following: "This is a course requirement for CS 192 Software Engineering II under the supervision of Asst. Prof. Ma. Rowena C. Solamo of the Department of Computer Science, College of Engineering, University of the Philippines, Diliman for the AY 2015-2016".		●	

#### 2. Code History

Criteria	Comply		Remarks
	Yes	No	
Code History appears after the license of the source code.	●		Github
Code History has been updated accordingly based on name of programmer, change date and change description.	●		Github

### 3. Internal Documentation

Criteria	Comply		Remarks
	Yes	No	
A comment block exists after the Code History containing the following information: file creation date, development group, client group and a brief statement of the purpose of the software in the file.		•	Add: each variable would have a corresponding descriptions  variable_name; #description
Each method in the file is preceded by a comment block which provides the following information: method name, routine's creation date, purpose of the routine, a list of the calling arguments (their types and what they do), a list of required files and/or database tables, and return value.		•	Lack of documentation
Indentations are 5 spaces.	•		
Inline comments are found at the top of the statement using '/*' and '*/'.		•	N/A due to Python syntax
Bracketing of block statements follows the following format:  if () { statements; }	•		code uses python syntax instead of C syntax, indentation utilized was for python for the sake of the code
Exceptions are being handled appropriately.	•		

#### Reviewer's Comments:

No additional comments.