

Self Review

By AbdelRahman



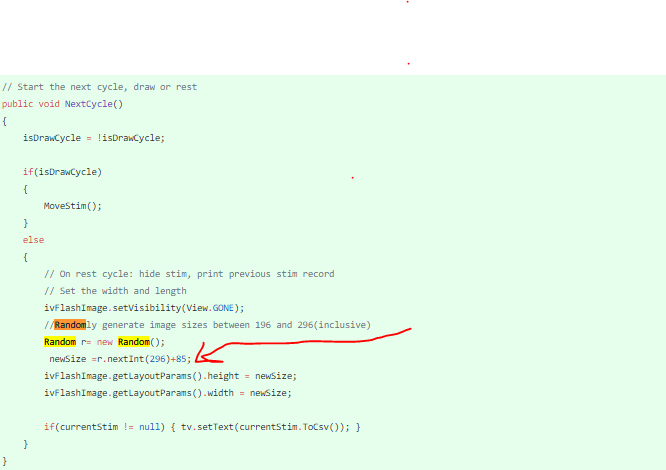
# Technical Proficiency

Project1 in the community group consisted of a lot of coding and being exposed to an array of different languages to which all posed a challenge and means to grow one vocabulary of languages. Moreover, the progressive nature of project1 was that it allowed us to gradually ease into the software development environment that involved much group work and intragroup communication. Coming from a somewhat simple

Programming background, this paper meant that there was so much more to learn. As Part of the client needs there 4 assigned tasks that are the following: the Visual Scan tool, Aya App, Internet of things Database API and a VR rendition of the Visual Scan tool. These tasks will form the bases for answering the self-review questions.

## Question: What is the overall quality of your code like?

### Task1: The visual Scan tool-FlashSizeImage.

As an initial transition to project1 paper, my task was to code the random generation of Image sizes that would appear on the screen in the visual scan flash tool. Integration of a new class flashSize fragment class with the required functionality needed another class /object called the FlashSizeTrial that defined its movement, and CSV file output. I injected a random generator and stored it in a variable called newSize. This value was used to set both the height and width of the image view drawn to the screen. This enabled the size of the image view to be controlled by changing the random upper and lower limit parameters.

<https://github.com/OtagoPolytechnic/CommSoftTasks/commit/e6353acb6070a6dd08533d57dbc559924d2950f1>

After the first initial meeting, the client needed information in the CSV to be more meaningful. This was remedied by adding functionality to display the size criteria of the imaged tapped. A simple method that checks the passed in random size integer and returns the corresponding size string. Following the preparation of the flashTrial constructor and its overriding toCSV toString to accept a size parameter, the categorize methods size property is seamlessly included in the class instances. The Simplicity of this code meant that it can be reusable and easily incorporated into additional classes with similar functionality.

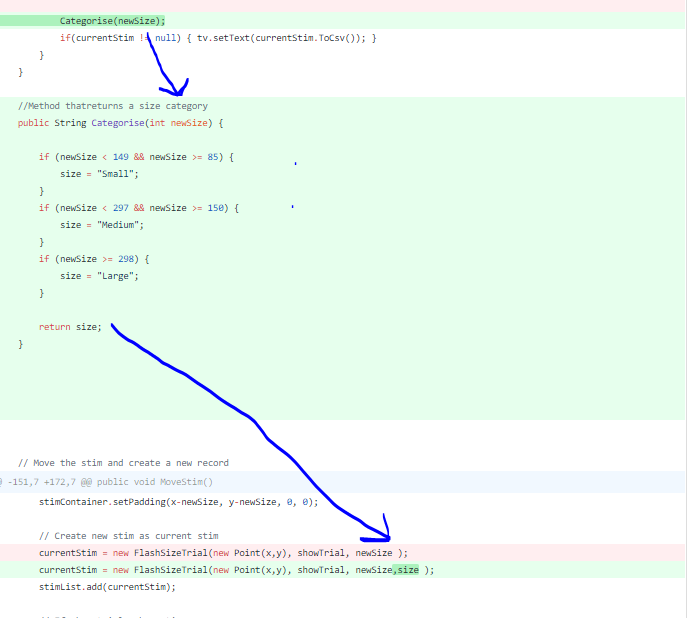


Figure 2: Size Category

### Task2: The Aya App.

### Task3: Internet of Things database API.

### Task4: Unity Virtual Scan Wander Tool.