

To output a library that enables responsively coded p5 sketches, I'd first identify the *ideal* experience from a p5 sketcher's perspective (~5-10 hours):

- Variables + defaults: what would the coder want control over? (E.g. max and min canvas widths and heights, possibly specifying breakpoints, lock canvas aspect ratio or not, etc)
- Which aligns more with the Processing mission: a bias toward simplicity or flexibility? Might it be better to simply enable "turning on" proportional sizing for all drawn elements and the canvas as a holistic system (simpler), or to allow the coder to choose individual elements that they could apply proportional sizing to (more flexible)?
- Error handling

It may not be feasible to build out the entire *ideal* p5 proportion-aware experience within the timeframe of a fellowship, so it'd be important to align on scope: the aspects to start with that'd provide the most value on their own, without hindering future enhancements (~2-5h).

Next I'd determine the p5-relevant programming patterns and syntax by which the user would employ the responsive attributes to their sketch code. For example, if we decide the user should be able to set canvas max width and height, should those values be folded into the existing `createCanvas()` element's value declaration list or set separately (could be something like `canvasMax(642, windowHeight);`)? It'd be imperative to get feedback from Processing/p5 advisors for these decisions before I built the library (~65-80h).

I'd finish by documenting the new capabilities through explanations and comprehensive examples, and record ideas for further future development, etc (~15-20h).

Daniel Shiffman comes to mind as a possible mentor due to his enthusiastic videos demonstrating a deep technical understanding of p5, as well as the philosophy behind Processing's goals and how they're achieved through tactical coding decisions within the p5.js library. Any mentor with expertise in these two areas would be great.