1. How might you communicate your evaluations of different automation frameworks? (e.g. you’ve decided to use Ruby with Capybara; what made you choose this tool over the others?) Because you’re a ruby shop. ^\_^. Ruby + Selenium. Selenium is pretty much the standard for UI automation. Other options such as Appium or water are niche tools for specific application types. Ruby is straightforward, open source, and uses syntax close to plain English, making it easier to read and debug
2. The developers are aware of the issues resulting from flaky specs. As part of your implementation, how do you prevent (to the extent possible) flaky specs from occurring? What guidelines and/or processes can you put in place to ensure that flaky specs are dealt with properly? QA involvement earlier/at every stage of development process. Better grooming practices, standardizing definitions of what constitutes an acceptable story to bring into a sprint.
3. How might you reduce the time CI takes to complete? And how do you do so without making significant sacrifices to quality? Smaller targeted regression suites that can be triggered via tag, depending on what applications or modules are changed. Use prioritized test suites to mitigate most risk in the given time. Automate as much of the standard regression as possible and have procedures in place to add to t he regression library with each iteration.
4. How might you work to flip the testing pyramid so that developers can focus more on unit and integration level tests, while QA can focus on frontend testing? And how do you make the determination of what should be tested via unit/integration/api tests versus what needs to be tested via the frontend? Emphasize that the unit tests tend to have the most value to effort ratio of any testing. QA efforts on API or UI level tests free the dev to make max impact on improving quality.
5. How do you ensure that the changes described above have buy-in from the engineering team? Have a beta program, have a team follow new procedure to prove out the effectiveness.