**Disclaimer:**

**This won’t make you an expert, but if you’re interested in doing something different than the classic Waterfall/monolithic approach that doesn’t really work, this should at least get you started.**

\*\*Note to Zac\*\* Docs to Find/Include:

* IV&V recommendations

Other important links to consider:

* USDS Playbook: <https://playbook.cio.gov/>. The United States Digital Services Playbook is a document that identifies the 13 high level “plays” an organization should be executing when creating modern software.
* 12 Factor Applications: <https://12factor.net/>. This is a list of 12 different “factors” that an organization should employ when building a software as a service application. Employing these 12 factors will help minimize time and cost for new developers joining the project, offer maximum portability between environments, be deployable on modern cloud platforms, enable continuous deployment for maximum agility, and can scale up without significant changes.
* Legacy Application Strangulation Case Studies: <http://paulhammant.com/2013/07/14/legacy-application-strangulation-case-studies/>. These are case studies from employing the strangulation, or encapsulation, pattern for replacing legacy systems.
* Agile BPA evaluation guide \*\* FINISH WRITING THIS SENTENCE\*\*
* Performance Based Quality Assurance Surveillance Plan: <https://github.com/18F/bpa-opm-eqip/blob/master/QASP.md>. This is the plan 18F uses to evaluate contractor's actions while implementing a statement of objectives or statement of work. It is designed to provide an effective surveillance method of monitoring contractor performance for each listed objective on the Performance Requirements Matrix in the call order. It also provides a systematic method to evaluate the services the contractor is required to furnish.