Requirements

- **Aesthetic & layout**
  - ease to use (desktop & mobile)
  - clean interface
  - leverages on convention - traditional calculator metaphor w/ latest enhancements considered (native mobile - iOS, Android)
  - logical grouping of operators, keypad/numbers, and clearing functions
  - cross-browser compatible
- **Functionality & Features**
  - Addition functionality
  - "Keypad" must operate as well as keyboard numberpad
  - Clearing controls (backspace and "clear all" functions)
  - Mobile compatible (set viewport data)
- **Storage**
  - Web server (for ease of viewing) – http://www.pixel180.com/calculator/calculator.html
  - Github – review - https://github.com/prestonshanepalmer/calculator.git

Users

- **EMC Interviewing team**
  - Developers/Designers/Manager

Design research (interfaces)

- **Google**
  - Legacy calculator images / interfaces
  - iOS6 & iOS7
  - Android interfaces
- **Android phones**
  - Latest HTC interface/layout
  - Latest LG3 interface/layout

Design

- **Photoshop comp**
  - Uploaded to Github
- **Sketches of possible layouts**

Development

- **Visual Studio code environment**
  1. HTML/CSS layout/design
  2. Javascript/jQuery functionality scripting
  3. Bug testing (along the way)
  4. Mobile compatibility check
  5. Cross-browser support
  6. User tests

(These were iterative in nature, so many times this was out of the order in which it is presented here, but this is the general pattern)