Code Documentation

Audiences:

- Management
- Operators
- Developers
- Researchers
- Technical Staff

Style Guides

- Effective Dart: Documentation: https://dart.dev/effective-dart/documentation
- Dart Comments: https://dart.dev/language/comments#documentation-comments
- Google Style Guides: https://google.github.io/styleguide/
- Rust Style Guide: https://doc.rust-lang.org/beta/style-guide/

Documentation Tools

- dart doc: https://dart.dev/tools/dart-doc
- Rust doc: https://doc.rust-lang.org/rustdoc/what-is-rustdoc.html

Code Visualization

Diagramming Code: (https://www.lucidchart.com/blog/visualize-code-documentation)

- UML Diagrams: https://www.lucidchart.com/blog/types-of-UML-diagrams
- Dependency Graphs: https://docs.github.com/en/code-security/supply-chain-security/understanding-your-software-supply-chain/about-the-dependency-graph
- Flow Chart

Naming Convention

NOTE: I have had a lot of difficulty in following the existing code in certain circumstances due to the fact that an on-screen field name is not used in code. For example, in trying to develop the D80 Add New Device Workbook, an Excel spreadsheet that provides the means to fill in all required values in preparation for actually adding a new device, I had a heck of a time figuring out which fields were read-only. Had the code used the same names for the variable as the field names shown on-screen, the effort would have been easier.

- Name variables to match the UI widget name seen by the user.
- Name variable to match the real-world terms used in the device's specification.
- Name variables to be descriptive of their utility