Requirements Document Changelog

Over the first course of our project's development lifecycle, we gradually came to the conclusion that many of our initial plans for the project - such as relational database integration and associated structures and interfaces - were no longer necessary. Additionally, as our client received word from associated parties and came to decisions, it became clear that some features were no longer needed.

Below is a comprehensive list of all changes made between our initially submitted requirements document and how it is today. Please note that the naming convention and formatting was changes on a document-wide level across versions. Ex. section II-A in the old document would read section 2.1 in the new version.

Scope

Updated paragraph 2 to specify more than one potential provider.

Added paragraph 3 to better outline and define export goals.

Product Overview - Product Functions

Added "Link drill log to drill site" to product functions.

Definitions

Added ArcGIS definition

Updated Kotlin definition to the more relevant Dart definition.

External Interfaces

Updated to reflect final technique for replicating gINT output. Initial plan was to potentially pipe data through gINT and collect output, the actual solution was to use a native third-party pdf generating extension for Flutter.

Usability Requirements

Removed automatic cloud uploading functionality. Restricted by limited flutter/dart functionality.

Performance Requirements

Removed third paragraph to reflect removal of database functionality.

Section II-E - Logical Database Requirements

Removed. Our initial plan for the project was to access data through a local SQLite database for retrieval and usage. However, as the project progressed and the structure of the data started coming together, we deemed it no longer necessary to implement relational database functionality in lieu of a functional file system.

- **Section 2.5 - Internal data storage**

Added. As our data is largely independent of each other, we opted to go for a more traditional file hierarchy rather than a SQL server. This is more straightforward for the purposes of our project, as we would have no need for the benefits of a SQL server, such as execution joins or creating views.

Design Constraints

Added second paragraph to require Dart/Flutter compatibility.

Software System Attributes

Updated to auto-saves to saves.

Removed third paragraph about SQL integration.

Maintainability

Updated references to external APIs to interfaces to better match the no internet connection requirement.

Supporting information

“And looks like” changed to “is shown below” to better reflect document formatting.