

Design doc

System components

A. Product Catalog File Downloader

- This downloader will be operating in background. For demo purpose, we also display its activity and progress. It has background mode data download capability.
- A timestamp will be set once finished. The purpose is to update this catalog file once a new catalog is available. The timestamp will help to determine if the client app has latest data set. Repeat process can be performed to populate this database

B. Database service

- Use Sqlite3 to provide data storage, browse and search.
- Basic function includes – create db, product table, add entries, search. The most basic search will be no filter which is the enumeration of whole catalog.

C. Database importer

- Manage the process of converting CSV file entry to entity (Product – struct of Swift), combine K number of products and request database service to insert or update
- Report the progress and status to delegate (control logic)

- It can be cancelled using atomic boolean flag.
The loop of populating should respect this flag.
(Not included in the code)

D. The business logic of download and update DB.

- This is managed by timestamps of catalog file and database update.
- Observe the status of CSV file downloading job
- Start CSV to database populating process
- Notify the progress to UI
- Transition product catalog list view and search bar

E. View Model

- The View Model for whole product catalog and search results
- Manage the pagination process and model building

F. UI Views

- Activity view of the download and populating process.
- Product list view. This is a freely browsing view with pagination – automatically scroll. This list items count is available but not put into UI. Real app should also show this information
- Search bar: user can enter some string to apply as filter to search the whole catalog. The filter entered by the user is applied on product ID. The result set item count should also be displayed.

