Synchronized Shopping List

User Manual

© 2018, Jim Schrempp and Bob Glicksman, Team Practical Projects
Version 1.0; date: 4/07/18

NOTICE: Use of this document is subject to the terms of use described in the document "Terms_of_Use_License_and_Disclaimer" that is included in this release package. This document can also be found at:

https://github.com/TeamPracticalProjects/Synchronized_Shopping_List/blob/master/Terms_of_Use_License_and_Disclaimer.docx.pdf

Table of Contents

Table of Contents

1. Introduction.

This document provides instructions and suggestions for using the *Synchronized Shopping List* app.

The *Synchronized Shopping List* is an app that multiple people use to add items to a common Shopping List and remove items from the common Shopping List as the items are purchased. The idea is that all people who share shopping chores for a household, facility, club or organization will have the app installed on their personal mobile device(s). At any time, anyone who sees a need to purchase an item simply opens the app and adds the item to the Shopping List. Whenever anyone in the group of users is shopping for items, that user displays the Shopping List and either removes the item from the Shopping List as the item is placed in their shopping cart, or else updates notes about the item (such as the quantity that is still needed to be purchased).

The Shopping List that appears on each user's mobile device is synchronized with all other users' Shopping Lists because the Shopping List is stored in a Google Firebase Realtime Database in the "cloud" (on the Internet), as illustrated in figure 1. The Shopping List is always current on all users' mobile devices because all users share this cloud database. Every time anyone uses the app to change something in the Firebase cloud database, the change is reflected in the apps of all of the users that you share shopping chores with.
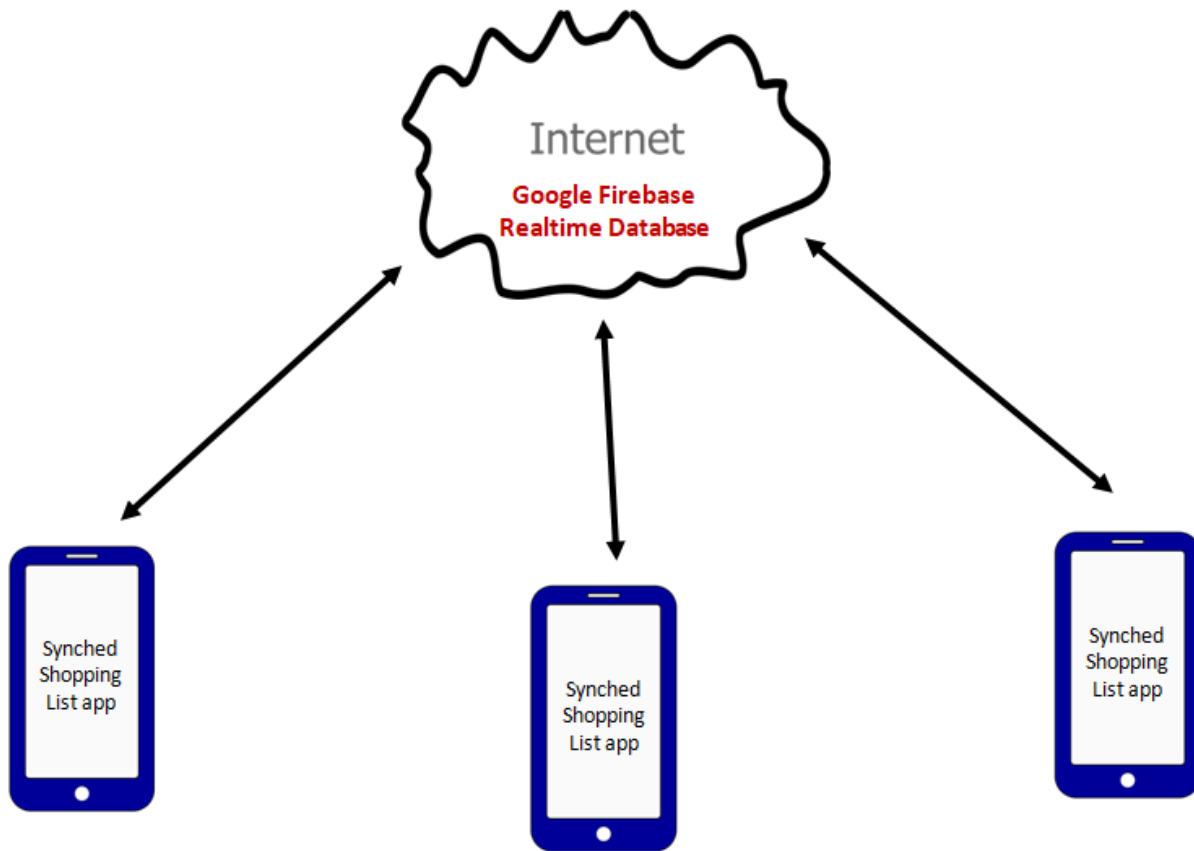
Figure 1. Synchronized Shopping List Overview.

The app itself is written using the MIT App Inventor 2 (AI2) integrated development environment (IDE).   At present, AI2 can only create apps for Android mobile devices. The MIT team has promised support for iOS (Apple) mobile devices later this year.

One person in the user group must create the Firebase database using their Google account. That person must then create the app installation file (.apk) that is used by each member of the group. The installation file contains credentials that allow the app to access the cloud database. Instructions for creating the cloud database and installing the app with the database access credentials are provided in the document: "Synchronized Shopping List Setup and Installation Instructions" which is located at:

https://github.com/TeamPracticalProjects/Synchronized_Shopping_List/blob/master/Documentation/Synchronized_Shopping_List_Setup_and_Installation_Instructions.pdf

2. <u>Concepts.</u>

It is important that you understand the basic concepts behind the *Synchronized Shopping List* app before you learn to use the app's features. The cloud database actually contains two lists: the *Item List* and the *Shopping List*.

<u>Shopping List:</u> The *Shopping List* is simply the list of items that need to be purchased. Each entry in the Shopping List contains two components: the *item* to be purchased and "*notes*". The item and notes are simply text fields and can contain anything that you (the users) wish. Section 5 of this document contains suggestions about how to use these two fields, but these are only our suggestions and you can use them in any manner that works best for your group. A Shopping List *item* and *notes* are always paired and always added, updated and removed as a unit.

<u>Item List:</u> The *Item List* is a cumulative list of all items that have ever been added to the Shopping List. The *Item List* is a convenience feature that makes it easy to add previously purchased items to the *Shopping List*. An item on the *Item List* can be tapped to add it to the *Shopping List* without the need to type the item name into the app. The idea is that items that are purchased once will likely end up on the Shopping List again.

The *Synchronized Shopping List* app does not provide any means to remove items from the Item List. This is by design. It is intended to avoid any confusion on the user's part about which list items are removed from. The REMOVE button on the app only removes items from the Shopping List. The administrator of the cloud database (the user who created the database in their Google account) can easily remove items from the *Item List* directly using their Google Firebase console.

In summary, users of the *Synchronized Shopping List app* add items to the shared *Shopping List* as they see that the items are needed to be purchased. Users of the *Synchronized Shopping List app* remove and/or update items on the shared *Shopping List* as they load items into their shopping cart for purchase, so that other users do not duplicate the purchase.

3. <u>Introducing the App.</u>

Figure 2 contains a photo of the *Synchronized Shopping List* app's screen. The app is very simple and has only this one screen.

Figure 2. Synchronized Shopping List app screen.

The following is a description of the components of the app:

View Shopping List: This is a BUTTON. When this button is tapped, the cloud database is accessed and a list of items on the *Shopping List* is displayed (nothing is displayed if there are no items on the cloud Shopping List). Tapping on an item in the Shopping List closes the Shopping List display and populates the *Item* field with the item from the Shopping List and also populates the *Notes* field with any notes associated with that item on the Shopping List. The Shopping List display can also be closed without adding an item to the *Item* and *Notes* fields by tapping the mobile device's back button. No other actions are taken at this time by the app.

Item field: The *Item* field is an editable text field. The app can display text in this field that is either typed into this field by the user or automatically added to this field by the user tapping an

item on the *Item List* or the *Shopping List* (or all of the above). No other actions are taken at this time by the app.

Notes field: The *Notes* field is an editable text field. The app can display text in this field that is either typed into this field by the user or automatically added to this field by the user tapping an item on the *Shopping List* (or both). No other actions are taken at this time by the app.

REMOVE: This is a BUTTON. When this button is tapped, the cloud database is accessed and the item in the *Item* field and the corresponding note in the *Notes* field are removed from the Shopping List (but not from the Item List).

ADD/UPDATE ITEM: This is a BUTTON. When this button is tapped, the following actions happen in the cloud database:

- The text in the *Item* field is added to the Shopping List in the cloud database unless it is already on the Shopping List. In other words, the item is never duplicated on the Shopping List but is added to the list if it is not currently present on that list.

- The text in the *Notes* field is updated on the Shopping List in the cloud database.

- The text in the *Item* field is added to the Item List in the cloud database unless it is already on the Item List. In other words, the item is never duplicated on the Item List but is added to the list if it is not currently present on that list.

CLEAR: This is a BUTTON. When this button is tapped, the *Item* field and the *Notes* field are cleared out.

Show Item List: This is a BUTTON. When this button is tapped, the cloud database is accessed and a list of items on the *Item List* is displayed (nothing is displayed if there are no items on the cloud Item List). Tapping on an item in the Item List closes the Item List display and populates the *Item* field with the item from the Item List. The *Notes* field is not populated. The Item List display can also be closed without adding an item to the *Item* field by tapping the mobile device's back button. No other actions are taken at this time by the app.

4. Using the App.

There are two basic use cases for the *Synchronized Shopping List* app: adding items to the Shopping List and removing/updating items on the Shopping List. Unlike traditional shopping apps, multiple users can perform either or both of these use cases at the same time.

Adding Items to the Shopping List: The benefit of using a mobile app for your Shopping List is that you can add items to it as you discover that you need them; for example, as you are

planning a meal or as you use up items when preparing the meal. As you notice that you need to place something on your Shopping List, you open the app and add that item to it.
The fastest and easiest way to add an item to your Shopping List is to open the Item List (tap "Show Item List"), locate the item that you need to purchase and tap on it. This will place the item in the *Item* field of your app. You may optionally add a note in the *Notes* field by simply typing in the note.

For example, you notice that you are getting low on orange juice while pouring a glass for breakfast. You take out your mobile device, open the *Synchronized Shopping List* app, tap "Show Item List", locate and tap "orange juice".   "Orange juice" will appear in the *Item* field. If you wish, you may add something to the *Notes* field, e.g. "2 quarts". You then tap on "Add/Update Item" and voila, the item "orange juice" and the note "2 quarts" are added to the Shopping List in the cloud database for all users to see.

If you discover that the item (e.g. "orange juice") is not found in the Item List, you simply type the item's name into the *Item* field on the app screen. See "Suggestions" below for some suggestions about naming items. Now when you tap "Add/Update Item", the item "orange juice" is added to both the Shopping List and to the Item List, the latter so that the next time that you need orange juice, you can find it on the Item List and not have to type it in again.

That's all there is to it!

Shopping for Items: Whenever you are out shopping, you open the *Synchronized Shopping List* app and you tap "View Shopping List". Head for the isle where one or more items on your Shopping List are found. As you remove an item from the shelf and place it into your shopping cart, you tap the item's name on the Shopping List and then tap either the "REMOVE" button or edit the *Notes* field and tap the "Add/Update Item" button. You tap "REMOVE" if you have fulfilled this item purchase. If you have only partially fulfilled the purchase of this item, e.g. if the required purchase was "2 quarts" but you are buying only 1 quart, you would edit the *Notes* field to change "2 quarts" to "1 quart" and tap the "Add/Update Item" button to update this on the Shopping List.

Next, you tap "View Shopping List" again and head for the next item on the Shopping List, repeating the procedure above until you have completed your shopping expedition. Note that each time that you tap "View Shopping List", you are accessing the shared cloud database. Therefore, if someone else in your group is shopping at the same time, your purchasing activities and their purchasing activities are off of the same Shopping List, avoiding duplication of purchases.

And, that's all there is to it!

5. Suggestions.

The *Synchronized Shopping List* app has intentionally been kept simple and flexible. The names of items and notes about items are simply text fields and you can type in whatever you wish. Here are some suggestions about how to name items for convenience. These are only suggestions, and you may operate the *Synchronized Shopping List* in whatever manner works best for you and your user group.

Naming Items: Whenever you tap the "Add/Update Item" button, the exact text that is in the *Item* field is added to both the Shopping List and to the Item List. You can, of course, type whatever you wish into the item field. However, we recommend that you type a simple but fully descriptive name the first time that you add an item to the Shopping List, so that you get the most benefit from the Item List in the future. For example, if you routinely purchase non-fat milk, we recommend that you type "milk, non-fat" into the *Item* field and then tap the "Add/Update Item" button. This will store the entry "milk, non-fat" on the Item list, so when you subsequently add non-fat milk to your Shopping List, the full description will come off of the Item List, saving you from typing "non-fat" into the Notes field each time.

Following this example, you might be expecting guests who drink whole milk and you want to add this item to your Shopping List. In this case, you might want to type "milk" into the *Item* field, and put "whole" in the *Notes* field (or perhaps, "whole, 2 quarts" in the *Notes* field to be fully descriptive). This way, you will hereafter have the item "milk" on your Item List, giving you the flexibility to note the type (as well as the quantity) of milk – "whole", "2%", "lactose free", etc. The idea here is that you don't expect to buy whole milk routinely, but you do expect to buy some type of milk other than your own non-fat milk occasionally.

We prefer the format "milk, non-fat" vs. the format "non-fat milk", but this is a matter of personal preference. We believe that the format "milk, non-fat" makes it easier to search the Item List for all types of milk thereon, as opposed to "non-fat milk". This way, if you have several types of milk on your Item List, it seems easier to find "milk, non-fat", "milk, 2%", and "milk, whole", as opposed to "non-fat milk", "2% milk", and "whole milk". However, this is a matter of preference and it is more important that your shopping group sets standards for itself than for individuals to do whatever each one thinks is best, making it harder for others to find items on the Item List.

Technical Note: technically speaking, item names are *tags* in the cloud database, while notes are *values* in the cloud database. Items in the cloud database are accessed by tags, and the tags are never duplicated. If you use a tag that is already used on one of the Lists, the value for the tag is updated but the tag is not duplicated. This feature is very handy in keeping your lists clean. But the cloud database can only do exact (and case sensitive) matching on tags. If the exact text for a tag is not found on a list when the "Add/Update Item" button is tapped, both the tag and the value will be added to the List(s). Therefore, consistency in the way that you name items is extremely important for keeping the Shopping List and Item List clean and easy to use!

Using Notes: The *Notes* field is intended to be just that – some clarification about an item on the Shopping List. The clarification can be anything that you want all users in your group to understand about each item named on the Shopping List. Usually, the *Notes* field will be used to indicate qualifiers on the item (e.g. if the item is ice cream, the Notes field might qualify the flavor: "chocolate", "strawberry", "vanilla", etc.), quantities of the item that need to be purchased (e.g. if the item is "milk, non-fat", the Notes field might specify "2 quarts"), and/or other data such as brand names (e.g. if the item is ice cream, the Notes field might qualify the brand: "Ben and Jerry's").

Notes are specific to a *shopping session* and not to an item! *Notes* are stored on the Shopping List but not on the Item List. This is why it is preferable to name an item "milk, non-fat" rather than name the item "milk" and make a note "non-fat", if you regularly purchase non-fat milk. Either way you do this will work, but you will save time and typing if you use the former method of naming items.

Technical Note: technically speaking, notes are *values* in the cloud database. These values are associated only with an item (the *tag*) on the Shopping List. The *values* associated with items (*tags*) on the Item List are always empty text strings. So it is advisable to make your item names fully descriptive of what you expect to purchase routinely and use notes to qualify and clarify a specific purchase.