Roderick Dunn
[redacted]

December 11, 2015

[redacted]
Algonquin College
1385 Woodroffe Ave.
Ottawa, ON
K2G 1V8

Dear [redacted],

Attached is my report on the research and design I have completed for a unique grocery list mobile app. The distinguishing feature of this app - *Speedy Shopper* – is that it automatically sequences grocery items according to their specific locations within the user's supermarket. Grocery shopping is a task that I personally prefer to finish as quickly as possible, yet I often find myself retracing my steps through the store, having overlooked items on my list. *Speedy Shopper* will allow customers to move through their supermarket on an efficient path, and ensure that they do not miss any items as a result of a disorganized, hand-written list.

This report is specifically intended to be read by mobile app developers and grocery store owners/managers, but will also likely be of interest to anyone who goes grocery shopping. It begins with a background on current methods of grocery shopping, an outline of some of their shortcomings, and an introduction to *Speedy Shopper* as a solution. Next, the results from a survey – used to determine levels of interest in the app – are explained, followed by a discussion of several aspects of the app's design, including the user interface and database integration. From my research and design efforts, I conclude that this project is both feasible and of legitimate practical interest to many consumers.

I would like to thank both [redacted] and [redacted] for helping me decide on professional-looking graphical layouts for the app.

I was thrilled to be given the opportunity to design this app as part of the curriculum here at Algonquin College. I actually had the idea for this app at the end of the summer (August 2015), but would not have had time to work on it this semester as a side project outside of school. I am also thankful to have carried out this project because I have gained a great deal of knowledge on Android app development, one of my desired fields of work. Finally, I am grateful for a first-hand experience with the analysis and design process that is so essential to developing software in the modern workforce.

If you have any questions or comments about this project, feel free to contact me by [redacted]

Sincerely,

Roderick Dunn

Speedy Shopper: The App for Efficient Grocery Shopping

By: Roderick Dunn

Algonquin College

School of Advanced Technology

Speedy Shopper: The App for Efficient Grocery Shopping

Prepared for [redacted] in partial fulfillment of the requirements of ENL2019T

By: Roderick Dunn

Algonquin College

School of Advanced Technology

Friday, December 11, 2015

Declaration of Sole Ownership

I, Roderick Dunn, confirm that this work submitted for assessment is my own and is expressed in my own words. Any uses made within it of the works of any other author, in any form (ideas, equations, figures, texts, tables, programs), are properly acknowledged at the point of use. A list of the references used is included.

Executive Summary

The task of grocery shopping is a weekly necessity for most people, yet few effective solutions exist that significantly reduce the time one must spend in a supermarket. Common methods of creating grocery lists, including writing them by hand or using a notes app on a smart phone, can increase the time spent shopping due to their tendency to produce disorganized lists. Certain apps have attempted to overcome this issue by categorizing items by product type, a feature which, although helpful, does not truly maximize the efficiency of one's grocery shopping experience. Another approach - online grocery shopping - is increasing in popularity, as consumers search for an easier means of carrying out this chore. However, these types of services can be unsatisfactory in terms of delivery time, order accuracy and the lack of control consumers have over the products they purchase.

This proposal discusses the research and design completed for a novel grocery list mobile app – termed *Speedy Shopper* – that sequences the user's grocery items according to their locations within a specified supermarket. A survey administered to 22 Algonquin College students demonstrated that there is significant interest in using this app. Users will be able to add multiple stores, and therefore create different lists for each one. Products will be sequenced by aisle number or store section as they are added to a list, making it easier for the user to collect each product efficiently. Furthermore, users will be able to cross off items as they are collected by tapping on each item.

Speedy Shopper will utilize a database of products and their respective locations within the store to carry out the sequencing task. As the final app will have numerous features, an agile development process will be used. This will allow an initial functional version of the app to be released quickly. Early versions will require the user to generate their own database – with the app's assistance – while later versions will include many built-in store databases. In future iterations users will also have access to a map interface, which will contain a graphical store layout screen overlaid with the locations of the user's grocery list items.

Development of the map interface feature will require several challenges to be overcome: creating store-layout maps for large number of supermarkets, modifying the database structure and content to increase the precision of product locations within stores, and developing a practical algorithm to calculate the shortest route around the supermarket while passing each item on the user's list. It was determined that adding sub-aisle location information would likely be best carried out using a crowd-sourced approach, in which existing users provide the data as they are shopping with the app. Moreover, the only known algorithm for calculating the true shortest path in this situation would be impractically slow, so an algorithm that yields an approximation of the shortest path will need to be implemented.

In conclusion, *Speedy Shopper* was found to be both feasible and practically interesting to a sample of consumers. The lack of currently available apps that can increase grocery shopping efficiency raises the chances that this app will be successful in the Android marketplace. Therefore, it is hoped that consumers will embrace this app as a valuable means of decreasing the time they must spend on tedious tasks, and provide them with more time for meaningful work (or play).

Table of contents

List of Figures

1.0 Introduction

The purpose of this report is to propose an idea for a mobile application (app) that will facilitate efficient grocery shopping. The task of grocery shopping is a weekly necessity for the vast majority of people in first-world countries [1], yet no innovative technological solutions currently exist that significantly reduce the time one must spend in a supermarket. I therefore believe that consumers will appreciate a novel and practical technology that will enable them to finish their shopping as quickly as possible.

People tend to use a hand-written list or a simple notes app on a smart phone to keep track of grocery items [2]. These methods often yield disorganized lists that result in an inefficient shopping experience, and may allow some items to be overlooked. Even dedicated grocery list apps, while better than simple lists in this regard, do not truly minimize the time a shopper must spend at the supermarket. Some consumers have begun to utilize online grocery services in order to avoid the inconvenience of conventional grocery shopping [1]. However, this type of service is expensive and decreases the user's control over what he/she buys, an issue which could result in receiving, for example, products that are nearly expired. The app that I am proposing, which will be referred to as *Speedy Shopper* within this report, will determine the most efficient path around a given grocery store, based on the user's specific list. This app will therefore enable consumers to carry out their grocery shopping in a highly efficient manner, and without resorting to an ordering service.

This report begins by discussing the deficiencies of current methods for creating and utilizing grocery lists. These methods include hand-written lists or list apps, as well as more sophisticated grocery list applications. The second section presents findings from a survey that was carried out on randomly-selected Algonquin students regarding their current means of making grocery lists, as well as their level of interest in the app. The survey results show that many students would consider using an app like *Speedy Shopper*. Next, the features that will make *Speedy Shopper* advantageous compared to similar apps are considered, followed by a discussion of the attractiveness of the app from a grocery store owner's perspective. The various aspects of design that will be involved in the production of *Speedy Shopper* are then discussed. This section consists of explanations of the primary components of the graphical user interface, the algorithm used to order the items on the user's grocery list, and the means for creating a database of grocery items to provide useful information for this app. Also presented are the findings from a related research effort, which sought to determine the feasibility of integrating a graphical "store-layout" screen (store map) within the app. It was found that although this feature is likely to be implemented in the future, it would be time-consuming and costly to carry out. The report concludes with a summary of the features of the proposed app, and outlines the next steps that will be taken to enhance its functionality.

The intended audience for this report includes grocery store owners/managers, app developers, and, to some extent, anyone who goes grocery shopping. Indeed, although the report contains some technical sections, other parts may be of interest to the average consumer. This report does not include a marketing plan or analysis, as these topics are better suited for an author with the relevant expertise. Also excluded is extensive research into the cost of development, since it is my hope that grocery stores would gladly cooperate with the app's need for database access, given the potential of the app to boost business. Instead, I will focus on explaining the overall design of the app and the advantages it will provide to its users.

Discussion

This section includes a background on popular methods of grocery shopping, before detailing the technical design of *Speedy Shopper.* First, several current methods of grocery shopping are discussed, including list apps and online grocery shopping, while highlighting some of their shortcomings. Next, results for a survey used to gauge interest in the app are discussed. The report then progresses to the app design, which includes sections on the graphical user interface, database integration and sequencing algorithm. Finally, the feasibility of a map interface within the app is considered.

2.0 Background

Modern day grocery shoppers tend to use either a hand-written list or a simple notes/list app on a smart phone to keep track of grocery items [2]. These methods tend to produce disorganized lists, thereby causing users to spend longer in the store than necessary. For instance, consider a list with several vegetables at the top, cereal and milk in the middle, and fruit toward the bottom. In this scenario a user might collect the vegetables and then advance to the dairy section, before realizing they must return to the produce section to collect the fruit at the bottom of the list. Furthermore, this type of disorganized list encourages overlooking of items that are not grouped with similar items, such as the fruit in the previous example. With longer or more complicated lists these problems would likely be magnified.

Various studies have been carried out that analyze paths of grocery shoppers within supermarkets (mainly for the purposes of market research into shoppers' behavior). This research, which employed shopping cart tracking systems, revealed that a significant portion of customers backtrack through the store, and return to a previously-visited area [3; 4]. Another study demonstrated that customer shopping paths tend to differ significantly from what is known as the *Travelling Salesperson Problem* (TSP) path – the shortest path possible based on his/her items [5]. (This term (TSP) originates from a long-standing computational challenge involving the calculation of the shortest route for someone required to visit X number of cities before returning to the starting point.) These studies demonstrate the inefficiency of grocery shopping using conventional list methods.

Certain specialized list apps are now available that may overcome some of these problems. Three of the most popular apps with the relevant functions are considered here. One such app, named *AnyList*, will categorize the grocery items entered (e.g. "produce", "dairy") [6]. One of the main goals of this function is to prevent users from having to return to the same section of a store more than once, since all items required in a given section are grouped together. However, the usefulness of this feature is limited by the fact that supermarkets vary in their physical placement of products: two related items may be found in the same section in one store, but may be far apart in another store. Another app, *Our Groceries*, allows the user to reorder his/her grocery list by manually dragging items into place within the list [7]. Similarly, an app named *Any.do To-Do List* provides the ability to add a comment associated with each list item (*AnyList* also has this feature) [6, 8], a function that could be used to note the aisle or section of each item on the list. Unfortunately, these features would only be useful for shoppers who know their supermarket's layout very well and are willing to perform this task each time he/she goes grocery shopping.

Some consumers have begun to cut out conventional grocery shopping altogether, opting instead for an online ordering system, followed by either pick-up or delivery of the groceries. In fact, a large global survey determined that more than 55% of consumers are open to online grocery shopping as an

alternative to conventional shopping [1], highlighting the need for a better way to carry out this chore. While online grocery shopping eliminates the need to physically shop within a supermarket, the service also reduces the user's control over purchases made. For instance, it would be difficult for a user to ensure that the produce he/she receives is as fresh as it would have been had he/she chosen it herself. Furthermore, most people have personal preferences that cannot be adequately described to another person. For example, a customer may only decide to buy mangos if they feel "perfect", or bananas if they are slightly under-ripe. These issues are echoed in several reviews of two online grocery services - Grocery Gateway and Loblaws' Click-and-Collect [9, 10]. Other shortcomings of online grocery shopping include the amount of time before groceries are ready to be picked up, and the lack of ability to bargain shop within the store.

3.0 Product Overview

To overcome the efficiency problems of grocery shopping with conventional lists, I propose an app that automatically sequences the user's grocery list according to the locations of the products within his/her specific supermarket. As a user enters grocery items into his/her mobile device, the app will reorder them in real time. Rather than being placed into product categories, as certain available apps already do, grocery items will be classified according to aisle number or section name (ie: "Produce"). Aisles and sections will be listed in the order they appear from the standard store entrance, with the appropriate products listed underneath, as shown in Figure 3-1 (a).

Ideally, the app will also feature a graphical store-layout screen, which shows a map of the user's supermarket (see Figure 3-1 (b)). In this screen, products on the user's list will be marked at their specific locations along an aisle or section. The app will then be able to calculate the shortest route connecting all of the user's grocery list items, and display the route on the map. This sequencing system will keep lists organized, thereby lowering the chances of needing to backtrack during a grocery shopping session. Another advantage of this app is that users will know where to look for products. Because aisle signs in grocery stores are often unclear about the aisle's precise content, *Speedy Shopper* would be highly useful for customers who don't know

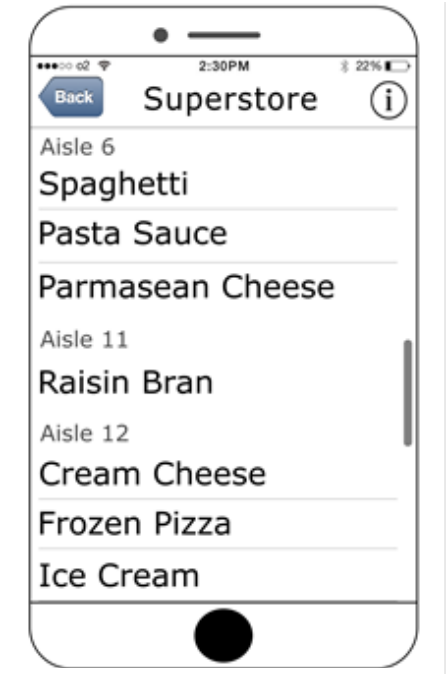


Figure 3-1 (a) *List View of Speedy Shopper.* Displays the user's grocery list organized into aisles of the supermarket.

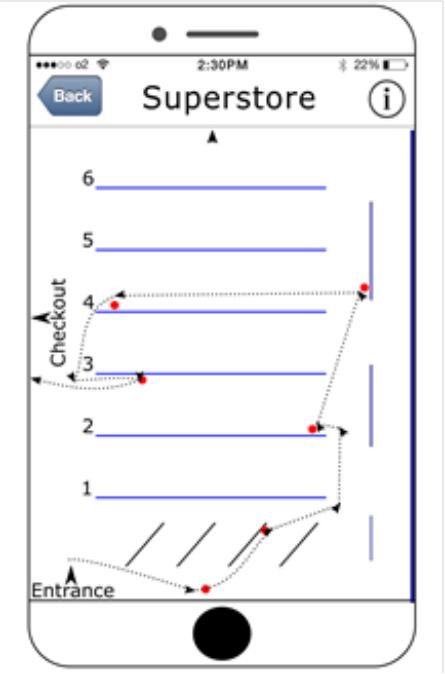


Figure 3-1 (b) *Map View of Speedy Shopper*. Displays the most efficient path around the supermarket on a graphical supermarket floor

the location of their desired product(s). These features will thereby decrease the overall amount of time a user would need to spend in a grocery store. Furthermore, items are less likely to be overlooked in this format, since they are laid out in a practical order.

This app is also advantageous from a supermarket manager's perspective. It is known that grocery shoppers generally attempt to follow the shortest route through the store, whilst collecting their items, although they do not usually accomplish this feat [6]. Therefore, customers may be more inclined to shop at supermarkets that have made their database of grocery items available for *Speedy Shopper*, since this app will provide customers with a more efficient, and therefore more pleasant, shopping experience.

3.1 Survey of Interest in App

In order to gauge interest in this app, I administered a short survey to 22 Algonquin students. I spent approximately 1 hour asking random students walking through the AC building if they could fill out a short survey for a school project. For those that agreed, I provided them with hard copies of the survey, which they filled out by hand on the spot. Based on this small sample, the majority of people would be interested in using an app with the features of *Speedy Shopper.* It was also found that most of these students already use an app to make their grocery lists, and admit that they sometimes have trouble finding items in the grocery store. Lastly, slightly more males indicated that they would be interested in using the app, compared to females. The results of this survey are summarized in Figure 3-2.

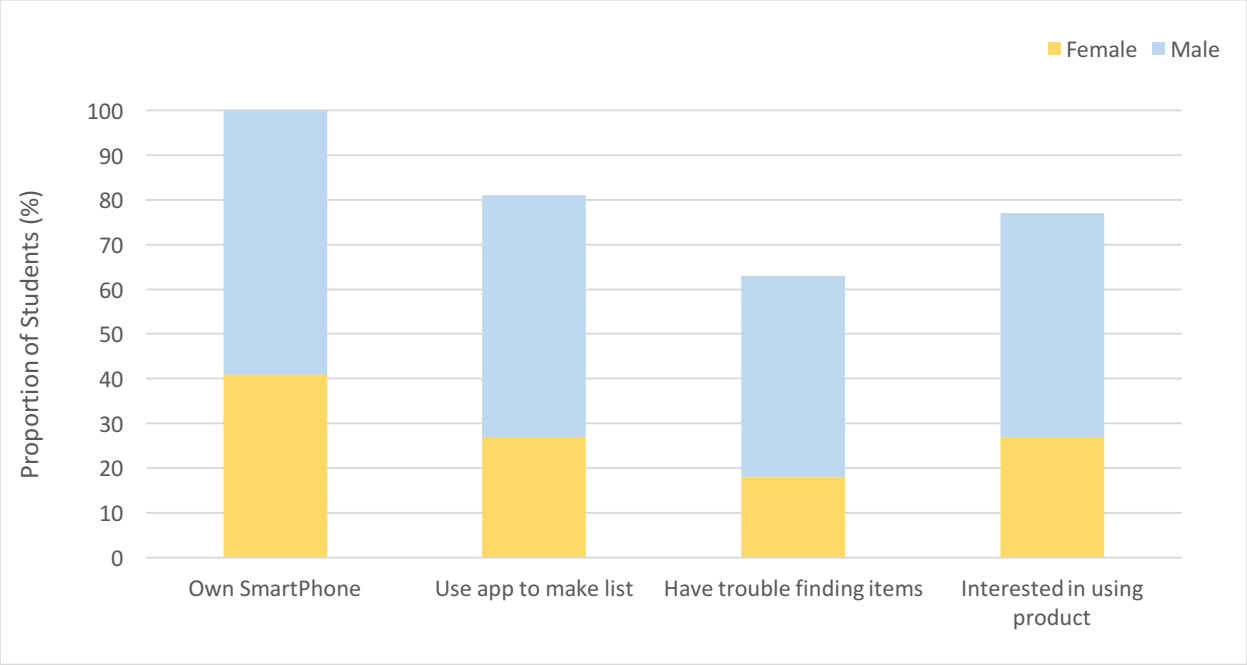


Figure 3-2 *Results of Survey to Gauge User Interest.* This figure shows the percentage of students who responded "Yes" on the survey for each question. The total number of students surveyed was 22 (n=22), and of these students 9 were female and 12 were male. The yellow portion of each bar indicates the percentage of female students who responded "Yes", while the blue indicates the percentage of female students who responded "Yes", while the blue indicates the percentage of male students.

4.0 Iterative Development Plan

The creation of this app will follow an agile development plan, during which several commercially-ready versions will be produced, each more feature-rich than the last. This section briefly describes the features of each of the three major releases.

Version 1

This version will concentrate on only the essential features of *Speedy Shopper*, the most important of which is the automatic sequencing of the user's grocery list items. *Version 1* will not include any built-in databases of supermarket product locations. Instead the app will help users build a personal database of grocery items and their corresponding locations within the supermarket (this process is detailed in the *Database Integration* section). Essentially, the app will "learn" the locations of each product as a user grocery shops. This process will be aided by automatic categorization of the user's grocery list (ie: "Produce", "Dairy"). As shown in Figure 3-1(a), *Speedy Shopper* will then be able to organize grocery lists by Aisle number and store section. In addition, users will be able to create lists for multiple stores and therefore build different databases for each store. These features will also be included in later iterations of *Speedy Shopper*, as some supermarkets may not allow 3rd-party access to their product databases.

Version 2

This version will include access to pre-made product-location databases for certain supermarkets. If a user goes to one of these included supermarkets, the app will automatically sequence his/her grocery list without any setup work on the user's part. Included databases will either be built by the development team, or be provided by supermarkets directly, depending on the willingness of supermarkets to allow access to their private databases. For those whose supermarkets are not included, the app will function the same way as in Version 1.

Version 3

The long-term vision for this app is to feature a "Map View", in which the user's grocery list is mapped to a graphic of the supermarket's layout (see Figure 3-1(b)). The app will then calculate the most efficient (shortest) route around the grocery store, while passing each item on the user's list, and display this path on the map. This feature is rather complex as it will require the product databases to include precise location information (within an aisle or section), and has therefore not been planned out like the first two versions. Instead, the feasibility of this feature is investigated in the section *Feasibility of a Map Interface.* The list view in this version will order items not only by aisle, but by location within each aisle. As such, the list view will reflect the same order specified by the route on the map view.

This version will also implement a feature to minimize the problems that will result from supermarkets changing the locations of products. If a user cannot find an item in the location specified by *Speedy Shopper*, he/she will be able to submit an update request to a central database that specifies the new location of the item. This update will then need to be verified by either the development team or a number of other users before the database is updated. For supermarkets that allow direct access to their database of products and locations, this will not be an issue. However, for developer- or user-created databases this feature will be necessary. Finally, Version 3 will include additional built-in supermarkets/databases, and subsequent updates will continue to add more.

In the following sections, the technical aspects of the short- and intermediate-term visions are considered. The feasibility of the long-term vision is discussed separately.

5.0 GUI Design

This section consists of images of each of the different screens within the app, as well as a contextual description of each one. The screens are presented in the order they would most likely appear when the app is opened. Graphics were created using a combination of the open-source program Inkscape and Android Studio.

5.1 Launching the App

In the Android operating system, a user generally launches an app by tapping on its home screen icon. In the case of *Speedy Shopper,* this icon is pictured in Figure 5-1 (a) as an isolated image, and in Figure 5-1 (b) in the context of an Android home screen. After tapping this icon, the app launches and a splash screen is presented, as displayed in Figure 5-1 (c).

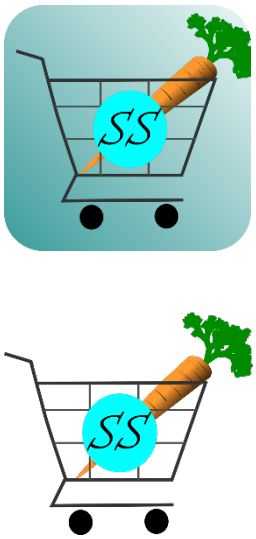


Figure 5-1 (a) *Speedy Shopper icons.* Icon is displayed as an iOS (top) and Android (bottom) icon.

Figure 5-1 (b) *Icon on home screen.* Speedy Shopper icon displayed in context of Android home screen

Figure 5-1 (c) *Splash Screen of Speedy Shopper.* Displayed during app start-up.

5.2 Adding a Store

The first interactive screen displayed to the user is the *Store List Screen*. This page displays a list of all the stores for which the user has made a shopping list (or is planning on making a list). There will also be an Add button (to add another store), and a Settings button to edit certain app preferences. When a user first opens the app this screen will be empty, apart from the Add and Settings buttons, as shown in Figure 5-2 (a). He/she will need to tap the Add button, which will open the *Edit Store Screen* (Figure 5-2 (b)). The user then types in the name of the new store, then enters any store details they wish to include (e.g. location, City, type). In versions 2 and above, starting to type in a store name will trigger an online search for stores that have pre-built databases, and results will be displayed in a drop-down menu (Figure 5-2 (c)). If the user's store is found he/she can tap its name, and the other information will be automatically filled in. At this point the pre-built database for this store will be downloaded. After tapping "Done" the user will be returned to the *Store List Screen*, where a new store will now be listed (Figure 5-2 (d)).

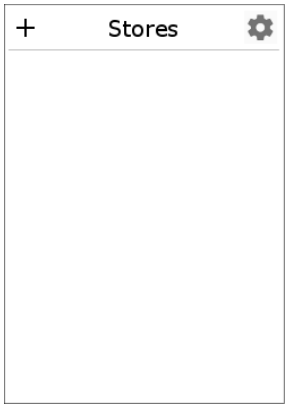


Figure 5-2 (a) *Store List Screen.*

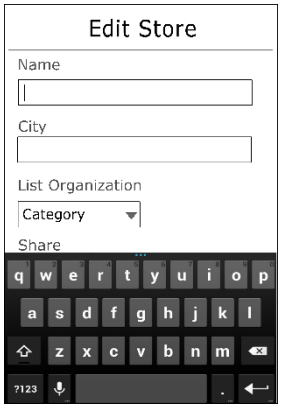


Figure 5-2 (b) *Edit Store Screen.*

Figure 5-2 (c) *Online Store Search.* Search for stores with premade databases

Figure 5-2 (d) *Store List Screen with Store.* Loblaw's has now been added.

5.3 Adding Items to a Shopping List

Tapping the name of a store in this view will take the user to the *Shopping List Screen* (Figure 5-3 (a)), where the user can add items to the shopping list for the store that was previously tapped. This screen consists of the name of the store, the grocery list, an Add button (for adding an item to the list), and a Settings button. For user-added stores (those without pre-built databases), this screen will initially only organize items based on category, not in-store location. For stores with pre-built

Figure 5-3 (a) *Empty Shopping List Screen*

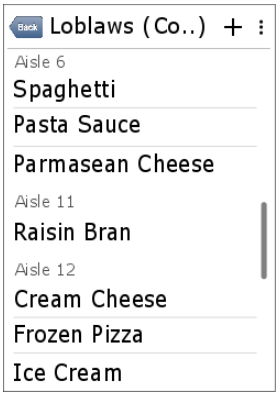


Figure 5-3 (b) *Filled Shopping List Screen*

databases (which will start to be added in Version 2), grocery items will be sequenced by in-store location as they are added to the list (Figure 5-3 (b)). Tapping the Settings button will allow the user to edit preferences for this specific list, such as whether to organize by category or location (if available). Tapping the name of the store will take the user to the *Edit Store Screen* described earlier. An item can be removed from the list by swiping on it, and tapping the Delete button that appears.

5.4 Shopping with Speedy Shopper

The location-based organization of the list will allow the user to finish his/her shopping efficiently, as he/she will clearly be able to see where the next item to collect is located. Users can tap a list item once it is collected at the store. This will add a strike-out line through the item, and therefore show the user the items that he/she has already collected. For lists that are fully sequenced by in-store location, this is the extent of the app's features while shopping. For lists that do not have a full database of items for the store available, there is some additional functionality that the user can choose to enable (within the *List Settings Screen*). This functionality is detailed in the Database Integration section and involves the generation of a database for such a store.

6.0 Database Integration

There will be several methods utilized for creating supermarket product databases. Users can create their own, the development team can build them, or they can be provided directly by supermarkets.

6.1 User-generated database (Versions 1+)

Since *Version 1* does not include integration with any existing supermarket product databases, sequencing can only be achieved via a user-generated database of product locations. (This will also be required in later versions for user-added stores - those without included databases.) Therefore, upon first use, the app will not rearrange the user's items by location, but rather, the user will need to go grocery shopping using the app as a conventional list app and tap the items on his/her list as they are collected in the store. Items that are not yet included in the database will be listed under the "Unsorted" section of the list. When an unknown item – "Carrots", for instance – is tapped, the app will prompt the user to enter the aisle or section he/she is currently in – "Produce". The app will then store this location information in a local database (on the mobile device). If carrots are added to the list any time after this, the list will show them under the "Produce" header. After several grocery trips, the app will have access to a fairly large database of grocery items and their corresponding locations, and will therefore be able to sequence the items that have already been purchased at the store. This process can theoretically be repeated until every item in the store is added to the database, but realistically it will only require a few trips to add most of the user's common items. After this, the user will only need to occasionally add location information.

6.2 Included Databases (Version 2+)

Beginning in Version 2, databases will be included for certain stores. These included databases will either be provided by grocery stores directly or built by the app development team. It is not known whether supermarket owners will be willing to allow access to their database of products for this purpose, so this generation process will be very important for development of Version 2 and higher. The team will use a very similar process to that described in the previous section to generate databases.

7.0 Sequencing Algorithm

In order for the app to correctly sequence the item on the list, it must first recognize that the item entered is present in a database. The app will need to utilize the Android Spell Checker Framework, since users will sometimes enter typos while typing in shopping list items. Also, because users may enter several different names for the same product (e.g. “Beans”, “Green Beans”, “Fresh Green Beans”), the database will need to include a “synonyms” column in the table.

Once an entered item is identified as a product within the database, the app will then place the item into the correct section of the list. In initial versions of the app, this will be accomplished by retrieving the “Section Name” attribute associated with the product from the database. The app will therefore be able to match the product entered with that in the database, using the appropriate SQL queries. The product will then be moved into the appropriate list section, which is created if it does not yet exist.

In subsequent versions of *Speedy Shopper*, this process will be more sophisticated, as the product database will include an attribute for “Subsection”. Practically, this will provide more precise location information for each product, resulting in a more organized list for the shopper. Based on this subsection information, the app will be able to sequence items *within* each section/aisle, which in turn will enable even more efficient shopping.

8.0 Feasibility of Map Interface

There are several challenges that would need to be overcome in order to implement the “Map Interface Screen”. Firstly, a floor map would need to be created or provided for each store. Given the sheer volume of stores, this process would require many people, significantly adding to the cost of development. Next, the database structure would need to be modified to include a “Sub-aisle” column, in order to specify the precise location of each product. Without such a database directly provided by supermarkets, this would also be a daunting task. A crowd-sourcing solution, in which owners of the app contribute to building a highly precise database (whilst using the app for its “Version 2” functionality), may be the best option for this part of development. Still, users will need to be provided with an accurate store map to begin with, so that they can specify where a product is located by tapping on a specific part of the map.

The final challenge involved in implementing the map interface screen is the calculation of the shortest route through the store. This calculation is an example of the Travelling Salesperson Problem (TSP) – what is the shortest path connecting a set of locations on a map? The only known way to ensure an accurate answer is a “brute-force” algorithm that calculates every possible path connecting all locations in order to identify the shortest one [5]. As the number of locations – grocery items in the case of *Speedy Shopper* – increases, the computational time required to solve the problem increases greatly. This solution is therefore completely impractical for most lists. Thus, the app would likely need to use one of several TSP estimation algorithms, the most popular of which guarantees a solution that is no more than 50% longer than the shortest possible path – Christophide’s Algorithm [11]. I believe that this level of efficiency would easily satisfy the requirements of this grocery shopping app.

9.0 Conclusion

Speedy Shopper is a mobile app that can decrease the time people spend grocery shopping by sequencing their grocery lists according to the locations of each item within the supermarket. This will be accomplished using a separate database for each supermarket, and an algorithm that sorts products by aisle number or sections. Future versions of the app will include a Map Interface Screen, which will display an overview of the store layout to the user. His/her grocery list items will be mapped onto this screen, and the app will calculate and display an efficient path that connects each of the product locations. I found that it was not practical to use an algorithm that is guaranteed to find the absolute shortest route around the supermarket, so I plan to use an algorithm that approximates this route. Given the limited computing power of mobile devices this is a much more sensible solution.

Much of the graphical design has been carried out during this project, so the next development step will be to build a prototype of the app, similar to Version 1, with which I will create a database for the grocery store where I shop most often. This process will allow me to test the essential functionality of Version 1, and therefore bring me closer to releasing a consumer-ready product. I will also need to create a separate database of product types so that users can organize their lists by category (rather than location) while they are building a database.

If you have any questions or comments about any report details, or my current progress on this project, feel free to contact me by email at [redacted] or by [redacted]

Works Cited

[1] A. McCaskill, 'More Than Half of Global Consumers Are Willing to Buy Groceries Online', *Nielsen.com*, 2015. [Online]. Available: http://www.nielsen.com/ca/en/press-room/2015/more-than-half-of-global-consumers-are-willing-to-buy-groceries-online.html. [Accessed: 24- Oct- 2015].

[2] Harvard Business Review, 'Scribd', Scribd.com, 2015. [Online]. Available: https://www.scribd.com/fullscreen/275945375?access_key=key-XsAfetSyCgHtn7yFG9yu&allow_share=true&escape=false&view_mode=scroll. [Accessed: 24- Oct- 2015].

[3] J. Larson, E. Bradlow and P. Fader, 'An exploratory look at supermarket shopping paths', *International Journal of Research in Marketing*, vol. 22, no. 4, pp. 395-414, 2005.

[4] M. Popa, L. Rothkrantz, C. Shan, T. Gritti and P. Wiggers, 'Semantic assessment of shopping behavior using trajectories, shopping related actions, and context information', *Pattern Recognition Letters*, vol. 34, no. 7, pp. 809-819, 2013.

[5] S. Hui, P. Fader and E. Bradlow, 'The Traveling Salesman Goes Shopping: The Systematic Deviations of Grocery Paths from TSP-Optimality', *SSRN Electronic Journal*.

[6] App Store, 'AnyList – Grocery Shopping List, Recipe Manager & Meal Planning Calendar on the App Store', 2015. [Online]. Available: https://itunes.apple.com/ca/app/anylist-grocery-shopping-list/id522167641?mt=8. [Accessed: 24- Oct- 2015].

[7] Play.google.com, 'Google Play Store - Our Groceries', 2015. [Online]. Available: https://play.google.com/store/apps/details?id=com.headcode.ourgroceries&hl=en. [Accessed: 03- Nov- 2015].

[8] Play.google.com, 'Google Play Store - Any Do. To-Do List', 2015. [Online]. Available: https://play.google.com/store/apps/details?id=com.anydo&hl=en. [Accessed: 03- Nov- 2015].

[9] Retail Category Consultants, 'Testing Loblaw's Click-And-Collect Service | Retail Category Consultants', 2014. [Online]. Available: http://retailconsultant.ca/testing-loblaws-click-collect-service/. [Accessed: 08- Nov- 2015].

[10] K. Yee, 'Is online grocery shopping worth the cost? | Toronto Star', *thestar.com*, 2015. [Online]. Available: http://www.thestar.com/business/personal_finance/spending_saving/2012/10/02/is_online_grocery_shopping_worth_the_cost.html. [Accessed: 08- Nov- 2015].

[11] E. Klarreich, 'Computer Scientists Find New Shortcuts for Infamous Traveling Salesman Problem', *WIRED*, 2013. [Online]. Available: http://www.wired.com/2013/01/traveling-salesman-problem/. [Accessed: 30- Nov- 2015].

Works Referenced

A. Gerber and C. Craig, *Learn Android Studio*. New York: Apress, 2015.

Developer.android.com, 'Design | Android Developers', 2015. [Online]. Available: http://developer.android.com/design/index.html. [Accessed: 01- Oct- 2015].

Lynda.com, 'Developing Android Apps Essential Training | Lynda.com Training', 2015. [Online]. Available: http://www.lynda.com/Android-tutorials/Developing-Android-Apps-Essential-Training-Revision-Q2-2015/369905-2.html. [Accessed: 10- Sep- 2015].

Glossary

Agile Development

An adaptive development methodology whereby projects are produced in an iterative manner. Each iteration is often market-ready, but as development continues more and more features are added.

Android

Popular mobile device operating system developed by Google and used by a large number of devices from various companies.

Database

In the context of this report this term refers to a *relational database*. This is a storage system with logical groupings of components and designed to provide efficient writing and reading operations.

Iteration

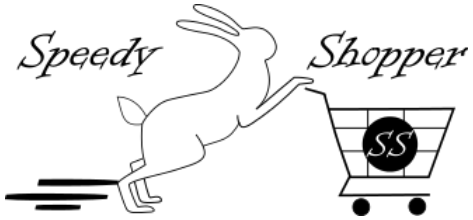
A version of a software product within an agile development project.

Traveling Salesperson Problem

A long-standing computational challenge involving the calculation of the shortest route for someone required to visit X number of cities before returning to the starting point.

Appendix A

Survey of User Interest



Please take a minute to fill out this survey on your grocery shopping habits.

Do you own a smart phone?

☐ Yes ☐ No

Do you currently use an app to make your grocery list (e.g. “Notes” app)?

☐ Yes ☐ No

Do you ever have trouble finding items in the grocery store?

☐ Yes ☐ No

Would you be interested in using an app that automatically organizes your grocery items based on where they are located in your store (e.g. condensed milk in aisle 8)?

☐ Yes ☐ No