

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON**

**SYSTEM REQUIREMENTS SPECIFICATION
CSE 4316: SENIOR DESIGN I
SUMMER 2019**



**NEPSTER
POCKET SHELF**

**UTSAB ACHARYA
SANDESH NAGA
SUDIP KANDEL
BHASKAR ACHARYA
TARA GURUNG**

REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	07.17.2019	SK, UA	document creation
0.2	07.19.2019	SK, UK, SN, BA, TG	complete draft
1.0	07.19.2019	SK, UK, SN, BA, TG	release candidate 1
1.1	07.20.2019	SK, UK	added customer change request
1.2	07.21.2019	SN, BA, TG	official release

CONTENTS

1	Product Concept	7
1.1	Purpose and Use	7
1.2	Intended Audience	7
2	Product Description	9
2.1	Features & Functions	9
2.2	External Inputs & Outputs	9
2.3	Product Interfaces	10
3	Customer Requirements	13
3.1	Registration	13
3.1.1	Description	13
3.1.2	Source	13
3.1.3	Constraints	13
3.1.4	Standards	13
3.1.5	Priority	13
3.2	Login	13
3.2.1	Description	13
3.2.2	Source	13
3.2.3	Constraints	13
3.2.4	Standards	13
3.2.5	Priority	14
3.3	Allocation of space	14
3.3.1	Description	14
3.3.2	Source	14
3.3.3	Constraints	14
3.3.4	Standards	14
3.3.5	Priority	14
3.4	Adding item to Inventory	14
3.4.1	Description	14
3.4.2	Source	15
3.4.3	Constraints	15
3.4.4	Standards	15
3.4.5	Priority	15
3.5	Modify Specification	15
3.5.1	Description	15
3.5.2	Source	15
3.5.3	Constraints	15
3.5.4	Standards	15
3.5.5	Priority	15
3.6	Search Using QR	15
3.6.1	Description	15
3.6.2	Source	15
3.6.3	Constraints	16
3.6.4	Standards	16
3.6.5	Priority	16

3.7	Manual Search	16
3.7.1	Description	16
3.7.2	Source	16
3.7.3	Constraints	16
3.7.4	Standards	16
3.7.5	Priority	16
3.8	Private Inventory	16
3.8.1	Description	16
3.8.2	Source	16
3.8.3	Constraints	17
3.8.4	Standards	17
3.8.5	Priority	17
3.9	Display result in Augmented Reality	17
3.9.1	Description	17
3.9.2	Source	17
3.9.3	Constraints	17
3.9.4	Standards	17
3.9.5	Priority	17
4	Packaging Requirements	18
4.1	Online Delivery	18
4.1.1	Description	18
4.1.2	Source	18
4.1.3	Constraints	18
4.1.4	Standards	18
4.1.5	Priority	18
5	Performance Requirements	19
5.1	The application should start within 1 to 2 seconds.	19
5.1.1	Description	19
5.1.2	Source	19
5.1.3	Constraints	19
5.1.4	Priority	19
5.2	The application should retrieve the data very fast and accurate.	19
5.2.1	Description	19
5.2.2	Source	19
5.2.3	Constraints	19
5.2.4	Priority	19
5.3	The application support the workload.	19
5.3.1	Description	19
5.3.2	Source	19
5.3.3	Constraints	19
5.3.4	Priority	20
6	Safety Requirements	21
6.1	Data Privacy	21
6.1.1	Description	21
6.1.2	Source	21

6.1.3	Constraints	21
6.1.4	Standards	21
6.1.5	Priority	21
7	Maintenance & Support Requirements	22
7.1	Compatibility	22
7.1.1	Description	22
7.1.2	Source	22
7.1.3	Standards	22
7.1.4	Priority	22
8	Other Requirements	23
8.1	Play store	23
8.1.1	Description	23
8.1.2	Source	23
8.2	standard	23
8.3	constraints	23
8.3.1	Priority	23
8.4	App Store	23
8.4.1	Description	23
8.4.2	Source	23
8.4.3	standard	23
8.4.4	constraints	23
8.4.5	Priority	23
8.5	Large Stores with items other than beverages	23
8.5.1	Description	23
8.5.2	Source	23
8.5.3	constraints	23
8.5.4	Priority	24
8.6	Peer Interaction	24
8.6.1	Description	24
8.6.2	Source	24
8.6.3	constraints	24
8.6.4	Priority	24
9	Future Items	25
9.1	Larger store	25
9.1.1	Description	25
9.1.2	Source	25
9.1.3	Constraints	25
9.1.4	Priority	25
9.2	Peer Interaction	25
9.2.1	Description	25
9.2.2	Source	25
9.2.3	constraints	25
9.2.4	Priority	25

LIST OF FIGURES

1	Basic UI of the application	8
2	Home Screen	10
3	Adding item	10
4	Product search using QR	11
5	Picking product once found	11
6	Adding Shelf	12

1 PRODUCT CONCEPT

We humans are surrounded by lots of items. Those items might be very important or might not be important at all. With having to interact with lots of items it is pretty hard for us to remember what items are needed and where it is located.

Pocket-self is the inventory tracking app that helps people to store and record the items that are important in a organized manner. With the help of the app users will be able to record the items in the database and later on retrieve it when needed. Users will be able to choose the particular self or the location that he/she wants to store the item. Then the app helps to enter the location and the item and saves the information in the database. Users will also be able to scan the item rather than having to enter the information manually. So, when the user needs the specific item then he can just search the app and it will tell where exactly is the item located.

1.1 PURPOSE AND USE

The purpose of the app is to make the life of user easier by organizing all the items as inventory in an organized fashion. The main goal of app is to store the information of the app in database and retrieve quickly when needed. When the user defines the location of the storage and provides the information about the item that they want to store, then the user can retrieve the information in the future just by doing the quick search. The app will also provide notification about the expiry date and vital information about the product in timely manner. Basically the app is used for storing items information and retrieve the location and information quickly when needed.

1.2 INTENDED AUDIENCE

This app is specifically intended for the people who want to organize the items that they have in their houses but, it can also be used for business to record the inventory if needed. This app is more useful for collectors. As for the collectors they have numerous amount of same type of the item. For example, the user might have collection of beverages or books. So, what he can do is he can enter the information and the location of where all the items he wants to store. Then, in future he can retrieve the item with ease.

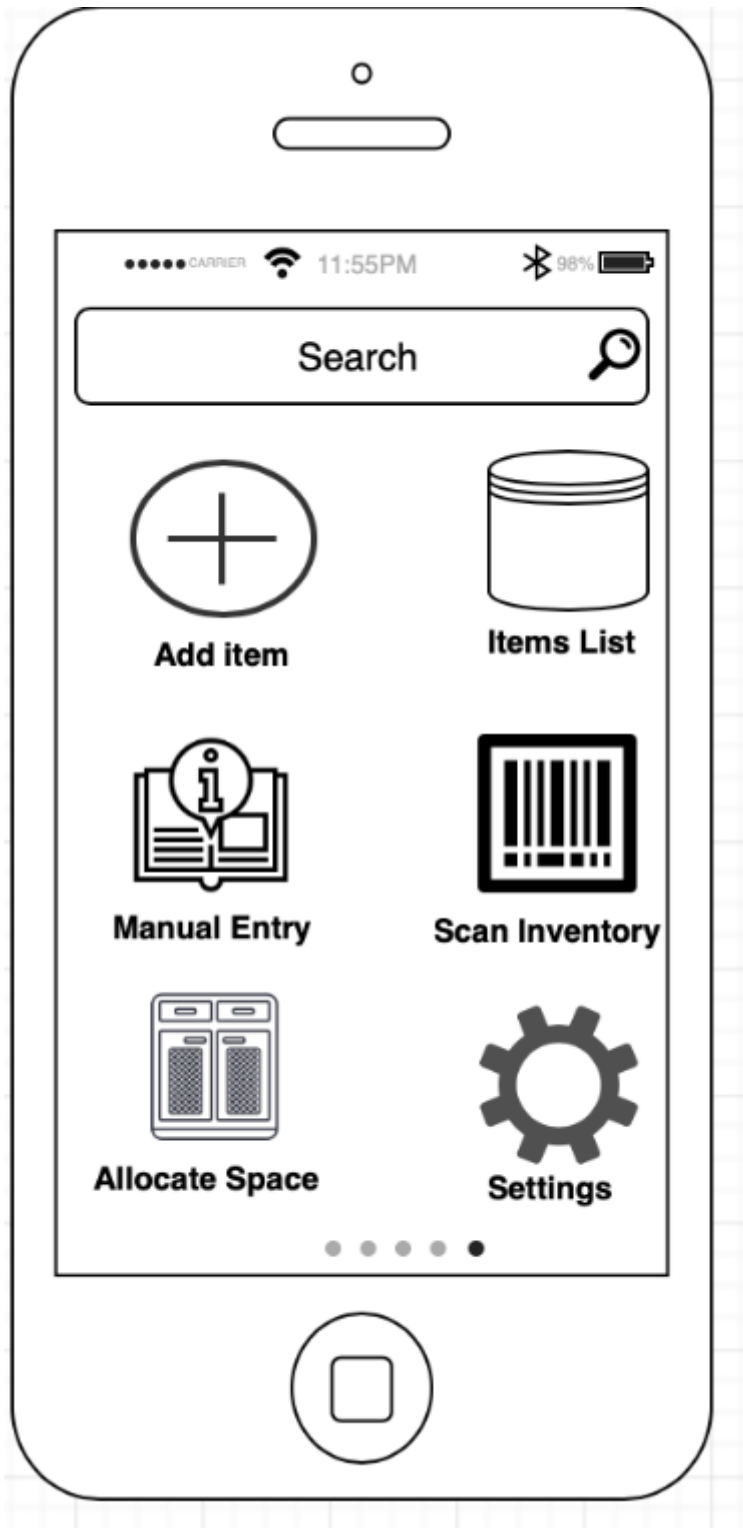


Figure 1: Basic UI of the application

2 PRODUCT DESCRIPTION

PocketShelf is an IOS/ Android based mobile application which allow user to keep their items in available space, which is previously created by user. We called it "PocketShelf" because, basically one is carrying all the shelf inside his/her pocket, like s/he can find his/her item just in a click. Not only that s/he can add an item in the app and app will decide where to keep that item, and it will ask some image of that item after which they can put that item on the given shelf and can forget where they kept it. All the remembering things will be done by the app, on top of it, our app will allow user to see the actual image of that item, so that the user can know that the item s/he is looking for is the exact item on the shelf before they grab it which will save their time and effort. Based on the items that can be added to the app, our application have different versions.

2.1 FEATURES & FUNCTIONS

"PocketShelf" is an inventory app, that can keep track of any item that user wants. We used Augmented reality as our key feature of the application. Where user can see the item before s/he approach to grab it. Which might be helpful in the sense that it will save the user effort and time spent, it s/he known that they are getting the right thing. Our app not only keep track of the item but able to sort the items based on the expiration dates and can notify the user if the item is expiring soon. Time frame to notify user can be pre- defined by the user at the time of its entry. Our app use the online database to save every entry by the user and had to be encrypted before it goes to the database, all the data had to flow through the server. So, user have to be online to be able to use the app. We had PIN protection as an extra feature on the app, where if the user want not to show the item location to other user using the same app on same device. Which we found cool feature if the user wants to secure his/her valuable items somewhere in the shelf.

2.2 EXTERNAL INPUTS & OUTPUTS

Initial version of PocketShelf is vary basic version of inventory app, which is fully based on the user input rather than machine learning's predictions. Each and every input should go through user. User can scan the bar code of the item to be stored(optional), user have to give unique name to the item to be store, user can add some notes about the item in item description field, user must dive the quantity of item to be stored in the shelf, it can be one or more. If the item have expiration date and user want to keep track of the expiration of any item on the shelf, during entry they can specify the expiration date and enable the notification before the specified date. If user does not want other user to see the location of the item, s/he can enable PIN protection for the location. User must add the picture of the item to be stored before adding to the database. Once user provide all valid input, user can press add button after which app will give the available location with a QR-code, user can save the QR-code for his record. Later if the user wants to retrieve some item, s/he can scan the QR, where the app will show the picture of the item up on the QR, to help user to able to identify the item. Once user pick the item S/he should input quantity picked from the location. If the location is PIN protected, during QR scan app will ask PIN to verify the user before reveling the item picture and item location.

2.3 PRODUCT INTERFACES

These are some of the user interface. Where Home screen is shown as soon as application is launched, if user wants to add item app will launch adding item screen and so on. We had implemented our interface in such a way that even a color blind person can easily use our app.

Home Screen

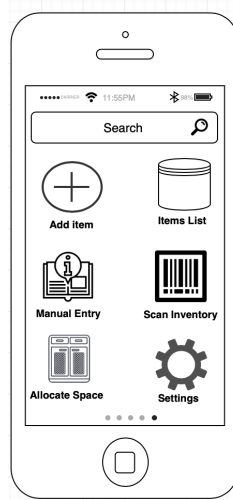


Figure 2: Home Screen

Adding Item.

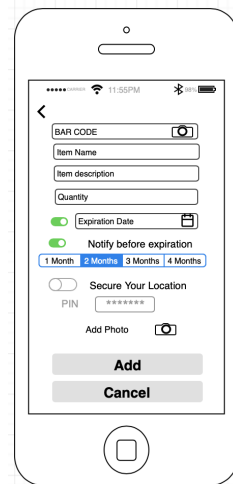


Figure 3: Adding item

Product Search.



Figure 4: Product search using QR

Product Found.

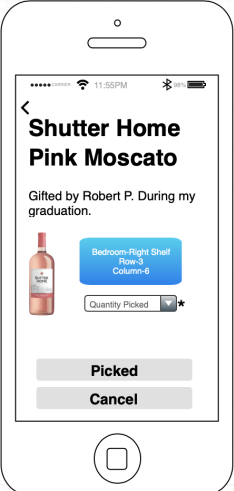


Figure 5: Picking product once found

Adding New Shelf.

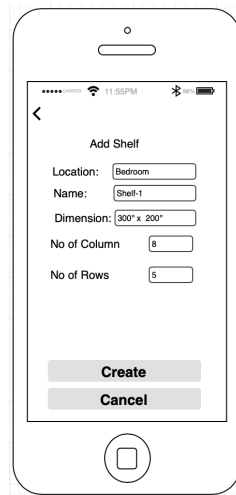


Figure 6: Adding Shelf

3 CUSTOMER REQUIREMENTS

The system aims at providing these features and functions in the app as per the requirements gathered from different sources. With the help of these core defined functionality users will be able to manage and store their inventory in easily reachable and secure manner. The main Customer Requirement for are app are:

3.1 REGISTRATION

3.1.1 DESCRIPTION

Application shall allow the first time user to register a account.

- User should provide valid Email
- User should provide unique username
- User should provide at least 6 alpha numeric password.
- User should provide three security questions.
- User should provide the account type(individual/business).

3.1.2 SOURCE

CSE Senior Design Project Specification.

3.1.3 CONSTRAINTS

- User must have a valid email.

3.1.4 STANDARDS

All the user information must be in encryption using *MD5 encryption*.

3.1.5 PRIORITY

- Critical

3.2 LOGIN

3.2.1 DESCRIPTION

Application shall allow the registered user to login their account with registered username and its password. In the login page application shall allow user some way to reset their password if forgotten.

3.2.2 SOURCE

CSE Senior Design Project Specification.

3.2.3 CONSTRAINTS

- User must be registered.

3.2.4 STANDARDS

All the input must be verified before using it to any queries to prevent SQL injection.

3.2.5 PRIORITY

- Critical

3.3 ALLOCATION OF SPACE

3.3.1 DESCRIPTION

Application shall allow the valid user to create customized shelf, based on the types of shelf s/he have, in order to store the items. For which:

- User should provide Name of the shelf
- User should provide the location for the shelf
- User should provide the numbers of rows and column in the shelf.

3.3.2 SOURCE

Customer

3.3.3 CONSTRAINTS

- Should be logged in to an account.

3.3.4 STANDARDS

N/A

3.3.5 PRIORITY

- Moderate

3.4 ADDING ITEM TO INVENTORY

3.4.1 DESCRIPTION

Application shall allow User to add item to their existing shelf.

- User must provide Item Name.
- user must provide Item Description. If item is Beverage
 - User may provide the name of Brewery.
 - User may provide Style of Beverage.
 - User may specify the format of the container.
- User Can provide the expiration date of the item if any.
- User must provide the quantity they want to store.
- User can select an option to be notified before the expiration of any item in the shelf.
- User can set up a pin to secure the position of the specific item for not letting other user to know its position if searched.

3.4.2 SOURCE

Chris Conly

3.4.3 CONSTRAINTS

- User Should have at least one empty position on the shelf.

3.4.4 STANDARDS

- System should generate some form of notification for successfully adding
- System should generate some thing to be able to locate same item in future.

3.4.5 PRIORITY

- Highly.

3.5 MODIFY SPECIFICATION

3.5.1 DESCRIPTION

System shall allow the authorized user to modify any thing from the added item.

- User can change the Item name or quantity added of so other description of the item even after it is added to the shelf.

3.5.2 SOURCE

Team member(Utsav Acharya).

3.5.3 CONSTRAINTS

- Should be authorized user for that item.
- Item should be in the shelf.

3.5.4 STANDARDS

N/A

3.5.5 PRIORITY

- LOW.

3.6 SEARCH USING QR

3.6.1 DESCRIPTION

System shall generate unique QR for every entered item and users shall be able to search for the item using specific QR code.

3.6.2 SOURCE

CSE Senior Design Project Specification

3.6.3 CONSTRAINTS

- User must be a valid registered user.
- User must have working camera in their mobile phone to scan the QR.
- User must have the valid unhampered QR code.

3.6.4 STANDARDS

N/A

3.6.5 PRIORITY

- HIGH.

3.7 MANUAL SEARCH

3.7.1 DESCRIPTION

Users shall be allowed to search the item just by using product name.

- User can just type the name of the item in the search bar to view result about their query.
- The search bar will provide the live search feature.

3.7.2 SOURCE

Cse Senior Design Project Specification

3.7.3 CONSTRAINTS

- User must be a valid registered user.
- User must have a working keyboard to provide input.

3.7.4 STANDARDS

N/A

3.7.5 PRIORITY

- Critical.

3.8 PRIVATE INVENTORY

3.8.1 DESCRIPTION

Users shall be able to keep the location of the item to private by providing pin to lock location of certain item.

- User should provide pin to make the items location private.
- User should provide pin after searching to reveal the location of inventory.

3.8.2 SOURCE

Team member(Sudip Kandel).

3.8.3 CONSTRAINTS

- Must have a valid working keyboard.
- Must be a valid registered member.

3.8.4 STANDARDS

- Pin number must have four digits.

3.8.5 PRIORITY

- Low.

3.9 DISPLAY RESULT IN AUGMENTED REALITY

3.9.1 DESCRIPTION

System shall display the result to the user in Augmented reality.

- User will be able to view the picture of the exact item in the Augmented reality.

3.9.2 SOURCE

Team member(Sudip Kandel).

3.9.3 CONSTRAINTS

- Item must be already entered in the inventory.
- User must be a valid registered user.
- User must have a working camera.

3.9.4 STANDARDS

- Size of the AR must not be bigger than the size of actual item.

3.9.5 PRIORITY

- LOW.

4 PACKAGING REQUIREMENTS

Since our product is a mobile application. It is not required to pack physically. When finished our product will be delivered as a mobile application. Our app will be uploaded on the ios and android application store and the customers will be able to download the app.

4.1 ONLINE DELIVERY

4.1.1 DESCRIPTION

Our PocketShelf App will be available online in the app store to download for the customers using both android and ios.

4.1.2 SOURCE

Customer

4.1.3 CONSTRAINTS

Wifi network is required to download the app.

4.1.4 STANDARDS

Files must be in .apk format in android and .ipa format in ios

4.1.5 PRIORITY

Critical

5 PERFORMANCE REQUIREMENTS

The performance of the app has significance important to make a beverage inventory fast, easy and accurate. Since the inventory app is finding the beverage products from shelves or suggesting the space in the shelves after scanning the QR Code, if performance is not good the app will mess up inventory list of beverage product, customers will not use it for inventory of beverage products. Inventory operation should perform fast and accurate. App will not drain the battery of the device and lasts as long as the battery of the mobile lasts and setup is not difficult, customers just have to install the application on their smart phones.

5.1 THE APPLICATION SHOULD START WITHIN 1 TO 2 SECONDS.

5.1.1 DESCRIPTION

Customer can open the application as fast as possible and able to make inventory manually, scanning bar-code or QR code. The application should be ready to inventory.

5.1.2 SOURCE

The Nepster

5.1.3 CONSTRAINTS

The application should open, scan bar-code or QR code very fast.

5.1.4 PRIORITY

High

5.2 THE APPLICATION SHOULD RETRIEVE THE DATA VERY FAST AND ACCURATE.

5.2.1 DESCRIPTION

System should retrieve the inventory data of beverage product accurate and fast as much as possible. Database should keep records of every update of the inventory.

5.2.2 SOURCE

The Nepster

5.2.3 CONSTRAINTS

Application should connected with wifi or internet. The application should retrieve inventory data accurate and update every records of inventory. The platform should be IOs and Android

5.2.4 PRIORITY

High

5.3 THE APPLICATION SUPPORT THE WORKLOAD.

5.3.1 DESCRIPTION

System should support free up to 5000 items and needs premium account for more than 5000 items.

5.3.2 SOURCE

The Nepster

5.3.3 CONSTRAINTS

Application should connected with wifi or internet and able to handle up to 1 millions users without any performance issues. The platform should be IOs and Android.

5.3.4 PRIORITY

High

6 SAFETY REQUIREMENTS

For the security of an individual's data privacy, a safety PIN will be required to locate an item if user had explicitly specified during entry. All the User data is required to be fully encrypted.

6.1 DATA PRIVACY

6.1.1 DESCRIPTION

PocketShelf will have a feature of creating a pin for items in the inventory upon specification of the customer. This safety requirement helps to protect the user's data from external invasion.

6.1.2 SOURCE

CSE Senior Design Project Specification

6.1.3 CONSTRAINTS

N/A

6.1.4 STANDARDS

Encryption

6.1.5 PRIORITY

Moderate

7 MAINTENANCE & SUPPORT REQUIREMENTS

Various elements are to be taken into consideration to keep the application functional all the time. The app should not only function in the latest versions of android and iOS but also should function without any glitch.

7.1 COMPATIBILITY

7.1.1 DESCRIPTION

-PocketShelf shall not crash, force close, freeze, or otherwise function abnormally on any targeted device.

-The app runs on the latest public version of the android and iOS.

7.1.2 SOURCE

Customer

7.1.3 STANDARDS

- 2.3: Android Gingerbread
- iOS version 5

7.1.4 PRIORITY

Moderate

8 OTHER REQUIREMENTS

PocketShelf, as customers download from app store or play store, the application should follow proper google or iOS terms and conditions. The application is growing and it is going to be used in lot more areas other than only beverage store.

8.1 PLAY STORE

8.1.1 DESCRIPTION

PocketShelf shall adhere to the terms of the Google Play Developer content policy.

8.1.2 SOURCE

Google Play Developer Content Policy

8.2 STANDARD

.apk

8.3 CONSTRAINTS

N/A

8.3.1 PRIORITY

Moderate

8.4 APP STORE

8.4.1 DESCRIPTION

PocketShelf shall adhere to the terms of the App Store Developer content policy.

8.4.2 SOURCE

App Store Review Guidelines

8.4.3 STANDARD

.ipa

8.4.4 CONSTRAINTS

N/A

8.4.5 PRIORITY

Moderate

8.5 LARGE STORES WITH ITEMS OTHER THAN BEVERAGES

8.5.1 DESCRIPTION

PocketShelf shall be able to locate, count, suggest items if necessary.

8.5.2 SOURCE

Customer

8.5.3 CONSTRAINTS

N/A

8.5.4 PRIORITY

Future

8.6 PEER INTERACTION

8.6.1 DESCRIPTION

PocketShelf shall be able to share with friends and family

8.6.2 SOURCE

Customer

8.6.3 CONSTRAINTS

Customer shall have andriod or apple smart devices

8.6.4 PRIORITY

Future

9 FUTURE ITEMS

This is only the initial version of the application. There will be a lot of changes in the future. Some are as listed below:

9.1 LARGER STORE

9.1.1 DESCRIPTION

PocketShelf shall be able to locate, count, suggest items if necessary in large stores such as Walmart, Target, etc.

9.1.2 SOURCE

Customer

9.1.3 CONSTRAINTS

The store shall have android or iOS device in use.

9.1.4 PRIORITY

low

9.2 PEER INTERACTION

9.2.1 DESCRIPTION

PocketShelf shall be able to share with friends and family

9.2.2 SOURCE

Customer

9.2.3 CONSTRAINTS

Customer shall have android or apple smart devices

9.2.4 PRIORITY

Future

REFERENCES