UNIVERSITY OF SUNDERLAND

ASSIGNMENT COVERSHEET

Student ID: 219570044	**Student Name/ Names of all group members: Bisesh Shrestha**
Programme: Computer Systems Engineering	**Module Code and Name: CET343 Android Mobile Development Assignment (2022/23)**
Module Leader/ Module Tutor: Rishav Chudal	**Due Date: 06/01/2023 Hand Date:06/01/2023**
Assessment Title: BabyBuy	
Learning Outcomes Assessed: (number *as appropriate*)	

	Mark	
Areas for Commendation		
Areas for Improvement		
General Comments		

Assessor Signature:	**Overall mark (subject** to ratification by the assessment board**)**	**Moderator Signature**

I confirm that in submitting this assignment that I have read, understood and adhered to the University's Rules and procedures governing infringements of Assessment Regulations.

PRINT Student Name: Bisesh Shrestha **Faculty Stamp *(date/time)***

Student Signature:

Module Code and Name: CET343 Android Mobile Development Assignment (2022/23)

Name of Module Tutor : Rishav Chudal

Contents

Analysis

The development of smart phones in recent years has altered the meaning of mobile phones. Phones are no longer merely communication tools; they are an integral element of people's communication and everyday lives. Various applications made people's lives more enjoyable. The mobile terminal will undoubtedly be the network's future.

The Android operating system is becoming increasingly popular in the electronics sector, particularly in the smartphone market. Because of open source, certain development tools are free, resulting in a large number of apps. This substantially encouraged people to utilize the Android operating system. Furthermore, it provides a highly convenient hardware technology and infrastructure, allowing them to spend less time and effort realizing their ideas. This allows Android to be developed further. As mobile hardware development progresses, the performance index is substantially greater than the actual software configuration needs. The phone's functions are more dependent on software. The application built on the Android SDK is gaining popularity as the Android operating system grows in popularity. (Li, et al., 2014)

There are several types of app development, including native, hybrid, and web app development.

1. Native app development involves building apps for a specific platform, such as iOS for Apple devices or Android for devices that use the Android operating system. Native apps are usually built in the programming language specific to the platform they are being developed for, such as Swift for iOS apps or Java for Android apps.

2. Hybrid app development involves building apps that can be used on multiple platforms, such as both iOS and Android. These apps are built using web technologies such as HTML, CSS, and JavaScript, and are then wrapped in a native container, which allows them to be distributed through app stores and accessed on mobile devices.

3. Web app development involves building web-based apps that can be accessed through a web browser on a mobile device or desktop computer. These apps are built using web technologies such as HTML, CSS, and JavaScript, and do not need to be downloaded and installed like native apps.

Native apps are usually built in the programming language specific to the platform they are being developed for, such as Swift for iOS apps or Java for Android apps. App shops, such as the Google Play Store for Android applications and the Apple App Store for iOS apps, are used to distribute mobile software. Native applications are built directly on top of the services supplied by their mobile platform. These services are made available through a specific Application Programming Interface (API) that includes methods for communication and message, graphics, location, security, and so on (Fling, 2009) .Native mobile app programming languages and tools are platform-specific; for example, Android applications are written in Java using the Eclipse-based Android SDK, whereas Apple iOS apps are generated in either Objective-C or Swift using the XCode tool. Developers may design native mobile apps with rich user experiences, heavy complex graphics, and excellent performance thanks to platform-specific APIs and technologies. However, the usage of platform-specific technology causes the well-known problem of mobile platform fragmentation, because code produced for one mobile platform (for example, the Java code of an Android app) cannot be utilized on another (e.g., the Objective-C code of an Apple iOS app) (Malavolta, 2016). Native app development, in my opinion, is a preferable technique to develop my app for my assignment to develop app called BabyBuy. Due of the short time schedule and device limitations, I am adopting the strategy for this project. I am already using Android studio for developing apps so developing and testing my app will be lot easier for me.

Operating system

There are several operating systems available for smartphones. The following are the primary mobile operating systems (OS) utilized by current smart phones:

Google's Android, Apple's iOS, Nokia's Symbian, Rim's BlackBerry OS, and Microsoft's Windows Phone.

With terms of mobile operating systems, Symbian has long been the dominant technology; however, it appears that in the move to smartphones, other operating systems such as Android, iOS, Blackberry OS, and Windows Phone are now in the lead. Specifically, Google's Android aim of producing an OS that can work on all mobile devices has many people believing that Android is the most used and popular mobile operating system in the world.In terms of software platforms, Java ME has long been the dominant platform for mobile devices. It has, however, been heavy and slow to deal with. Qt has received more attention and interest in recent years because to its platform independence and lack of additional complexity (O, et al., 2014).

Design

It is hard to create a terrific customer experience without a proper visual hierarchy in your UI. I wholeheartedly agree that excellent designs are both unique and functional. Having a great Android wire framing tool in your designer toolset, such as Mock plus and wireframe, makes life much easier. Wireframes allow you to create an interactive high fidelity prototype in minutes and test it with real users.

Screen Hierarchy with Activity flow

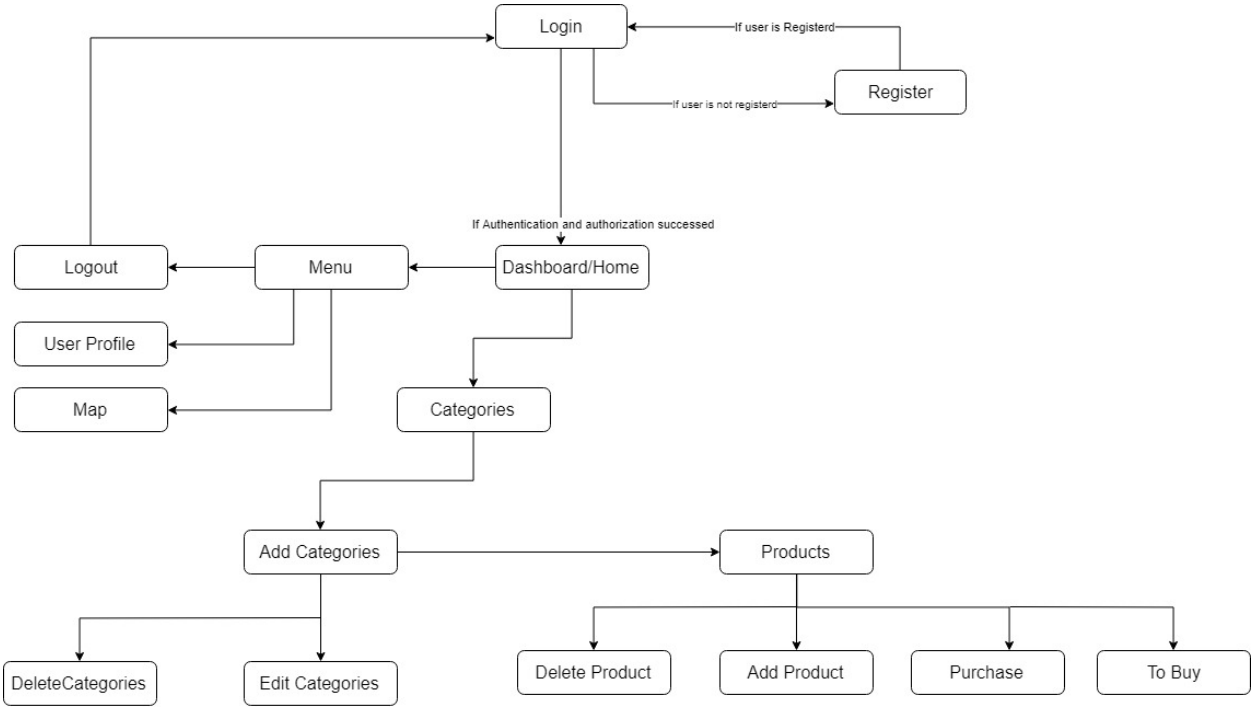


Fig:1 Screen Hierarchy with Activity flow

Wireframe

In my application, I used material design on several UI segments such as buttons, material icons, and app bar layouts, which assisted me in generating the most recent UX trends. Before designing an application, I used draw.io to create a wireframe for each layout and page of my application. Draw.io is a popular tool for creating an effective wireframe for apps. I've listed the wireframes I've produced below:

1. **Login**

Fig: Login

2. **Register**

Fig: Register

3. Menu with Logout

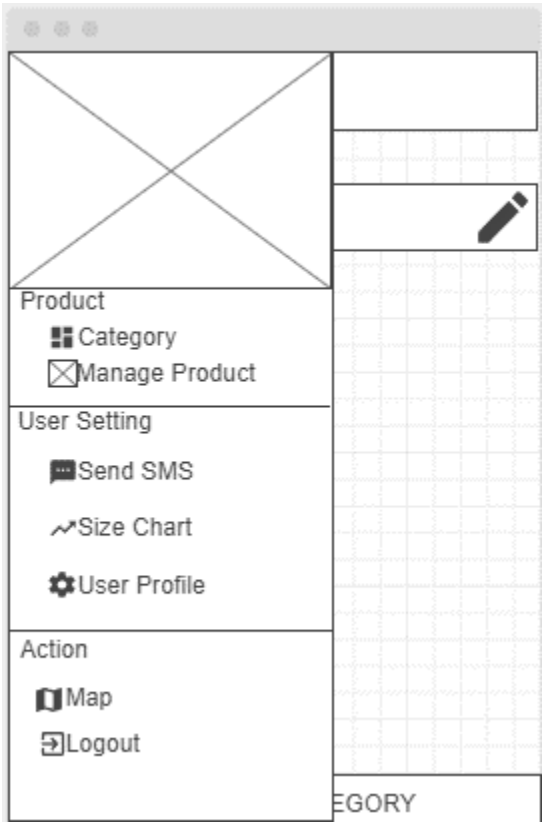


Fig: Menu with Logout

4. Categories List/Home page

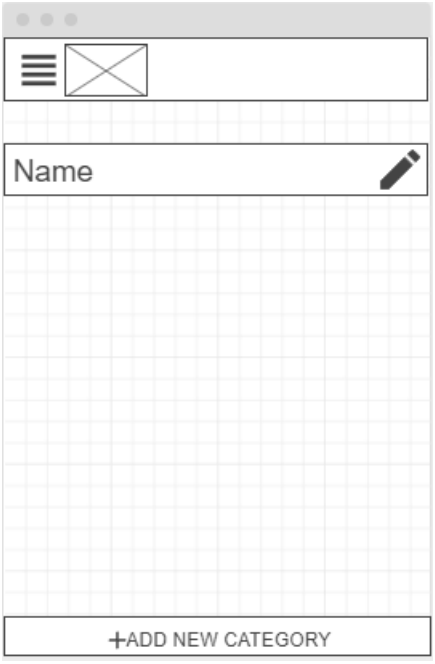


Fig: Categories List

5. **Add Categories**

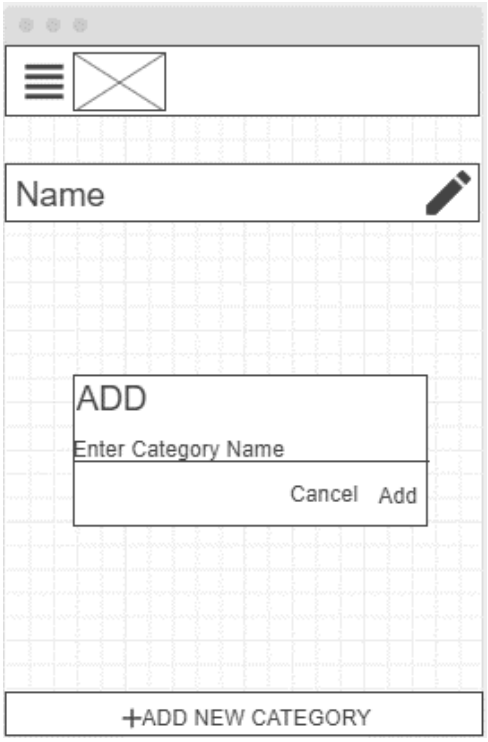


Fig: Add Categories

6. **Product**

7. **Add Product**

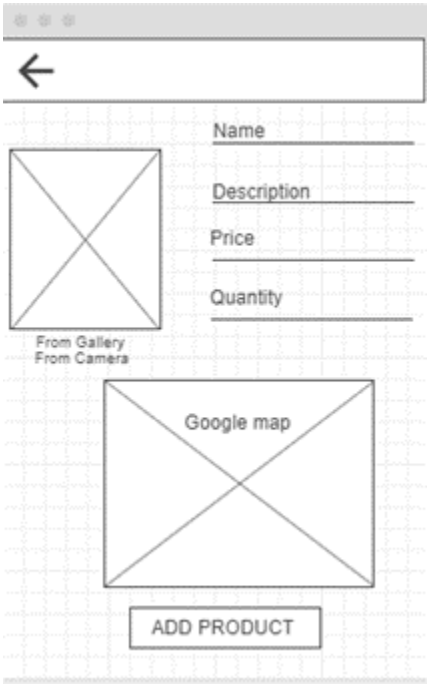


Fig: **Add Product**

Fig: product

8. **Manage Product**

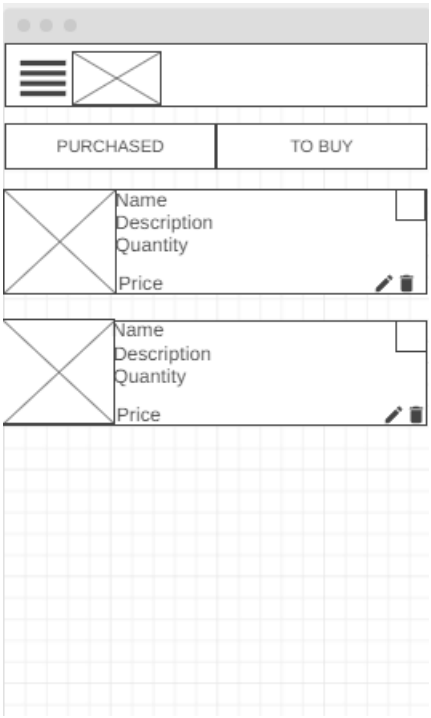


Fig: Manage page.

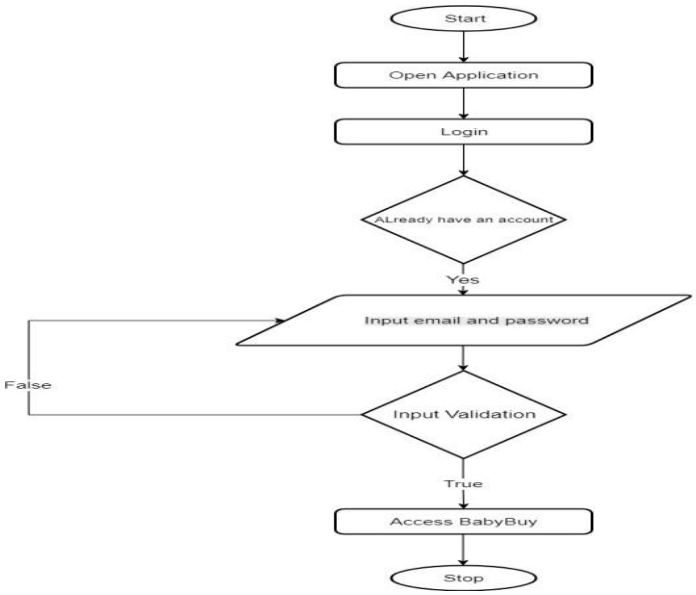
As you can see in the above wireframes, I have applied several approaches for designing an application employing material design techniques as well as numerous industry-based UX trends. Because I was constructing an application and came across numerous design concepts, there was some difference in the program such as the login and registration pages from wireframes. For example, optimizing a design for a range of displays is critical in application standards. To achieve this purpose, I modified the measurement unit from px to density-independent pixels (dp or dip). In keeping track of my design efforts, I was able to better understand how my design thinking progressed.

Functionality.

The "BabyBuy" app's basic functionality is to help parents create a list of items they need to buy before the birth of their baby and after. This app will store a list of items they need, as well as a description, price and picture of the item. Extra features will allow optional location tracking of the shop they need to visit as well as item delegation.

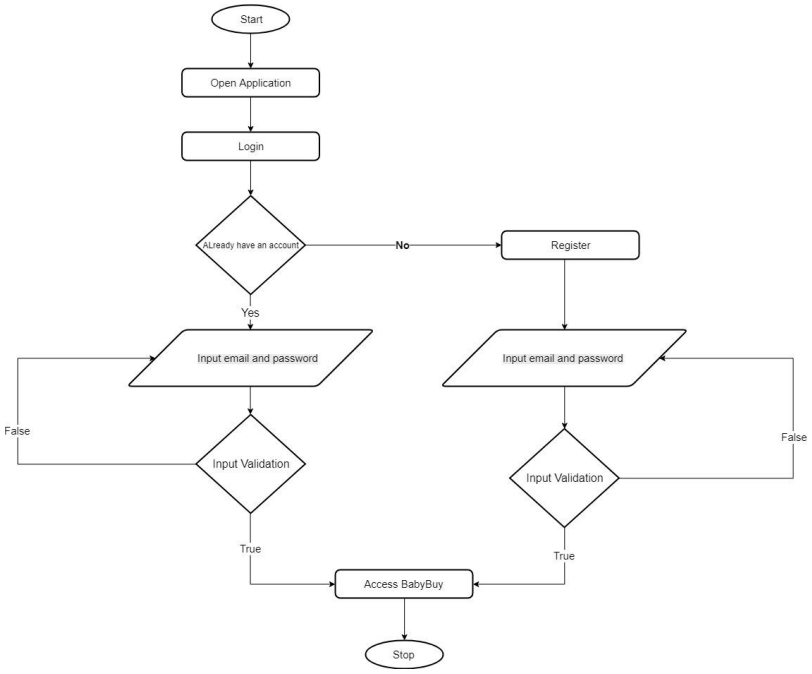
This application has many activities and pieces that work both independently and collaboratively to decide app functionality and system behavior. Some application features will be briefly detailed with code logic on how it will operate below:

1. **Login**



To login in Baby Buy app User have to use their registered email and password if thye they don't have registered email and password then they have to register their email through registration process by clicking register button on Login page.

2. Registration



To register user account user have to enter your valid information. User will have to enter your First and last name, user email in email format and your password which is required for login. User also have to enter the confirmation password so that password won't be mistakenly typed when user type the password and must be 8 digit long.

3. Menu with Logout

In menu screen user can see all the function such as managing categories managing Product, user can see whether the product is purchased or is in to buy list. In menu screen user can also see the product location using map function. User can also see your profile detail. User can also send SMS from menu screen where user can send detail about product. And mainly user can logout from the BabyBuy app using logout button.

4. Categories

In categories list user can see the list of product category. User can see different type of product categories and if they want to create the product category then they have to click on add new

category. To see the product of the category user will have to click on the product category and if user slide the category they can also delete it.

User can add different categories of product using add new category from product list screen. They can separate the product and can see which product fall on which category. User must have to provide name for the category to add it on categories list.

5. **Product**

After clicking product category user can see the product list screen where they can see the list of the product which is added from clicking add new product.

After creating category user have to add product using add new product button so that they can add as many product as they want inside those category. They must have to put product name, description, quantity, price, location and photo of the product. Photos can be uploaded using live photo capture or by selecting it from gallery.

If user want to see the product whether the product is purchased or they want to buy they can see the product from Manage product screen. They can see two different button purchased and to buy. If they see the product in purchased list then they have already purchased the product and if they see the product on to buy list then they are going to purchase the product.

6. **Delete category/product**

If user want to delete any category then they also have an authority to delete it. Thye simply have to swipe left to delete the product

7. **Geotag**

BabyBuy allows user to geotag the location of a product. The Google Map API is used to launch Google Maps and manually choose sites or search via a map, which later can see the product location via Google Maps.

8. **System requirements**

BabyBuy uses around 25 MB of your phone's RAM and runs well on approximately 98 percent of Android phones.

9. **Storage**

Some data required to install and utilize the BabyBuy app which would be saved locally on the phone, but all user data would be stored in the firebase, improving the overall performance and efficiency of the program and the device on which it is installed.

Test Strategy and Test Results

Testing is an important part of both the mobile and web app development processes. It might aid in testing software functionality, usability, compatibility, accessibility, and performance before releasing it into the wild. There are several testing techniques available, which can be automated or manual. Because there are multiple testing approaches, the technique or approach used to run those tests varies. Each step must be prepared in order for testing to be consistent; hence, the test strategy, i.e., a high level (static) testing document, must clearly explain the aim of testing. The following is a short testing plan document that clearly defines the goal of testing the BabyBuy app:

1. Overview of application

BabyBuy is a native Android application that offers users a secure means for gathering and saving memories. This application's functionality is straightforward but effective. Users may upload their product and detail from this app and Aside from that, users could geotag their product, which could then be viewed on Google Maps.

2. Purpose of the test strategy

The primary goal of this test method is to ensure that the BabyBuy application is completely functional and error-free, as well as to ensure that the produced application functions correctly in accordance with the specifications.

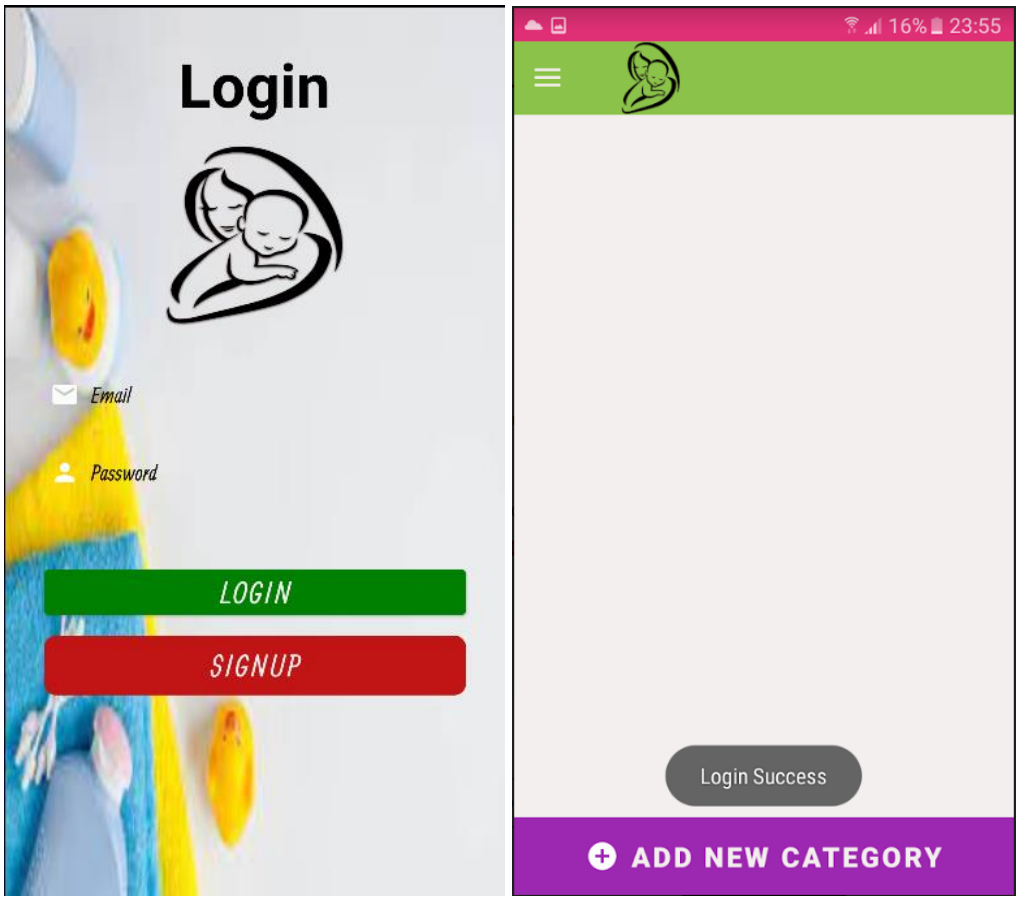
3. Select testing strategy

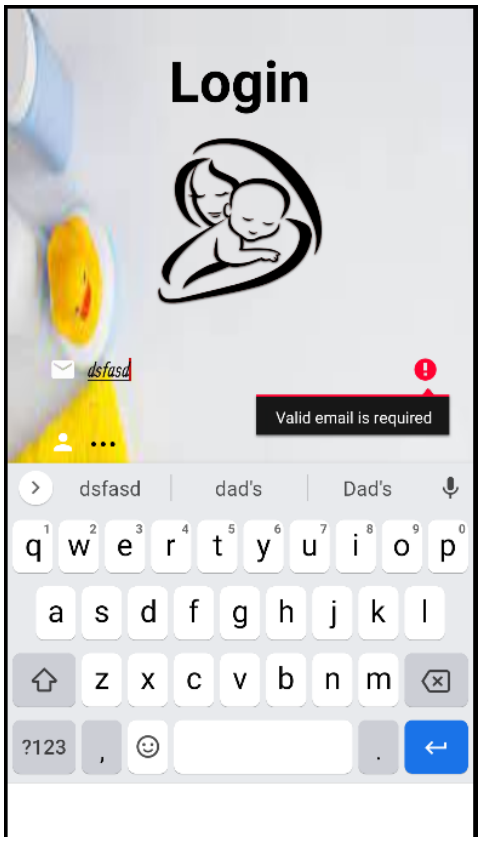
BabyBuy application is a tiny but powerful program that must be accessed logically, but the most significant testing that must be performed is User Interface Testing. Users interact with the application's UI, which must be thoroughly tested before release. As a result, UI testing appears to be better suited for this application.

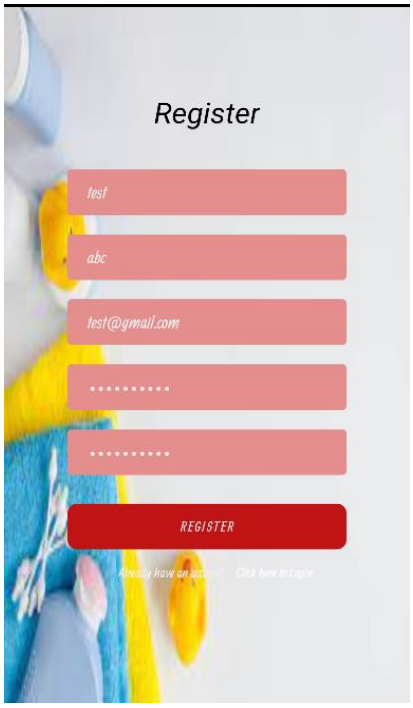
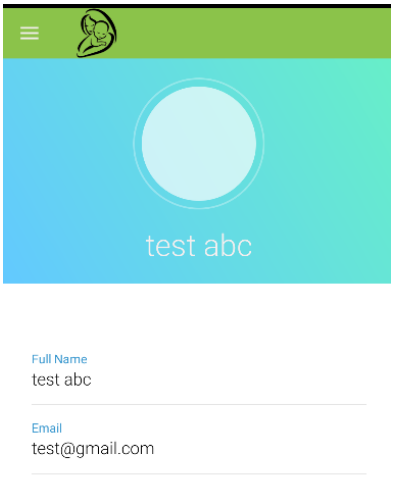
4. Test item
    a. Login
    b. Register
    c. Add category
    d. Edit category

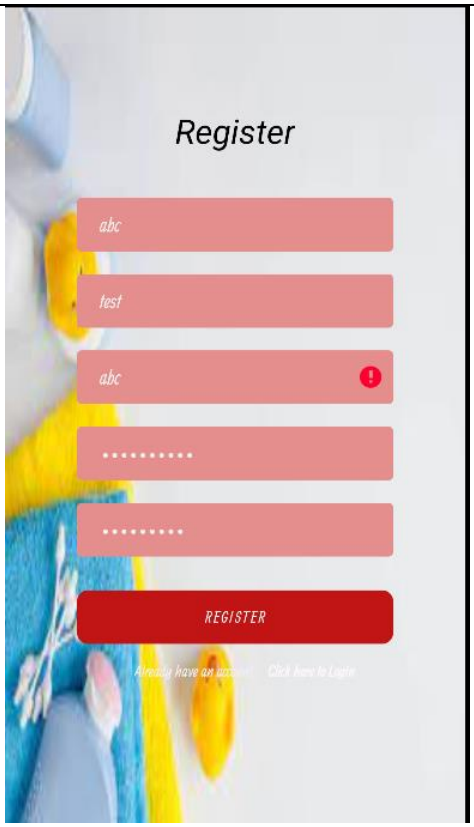
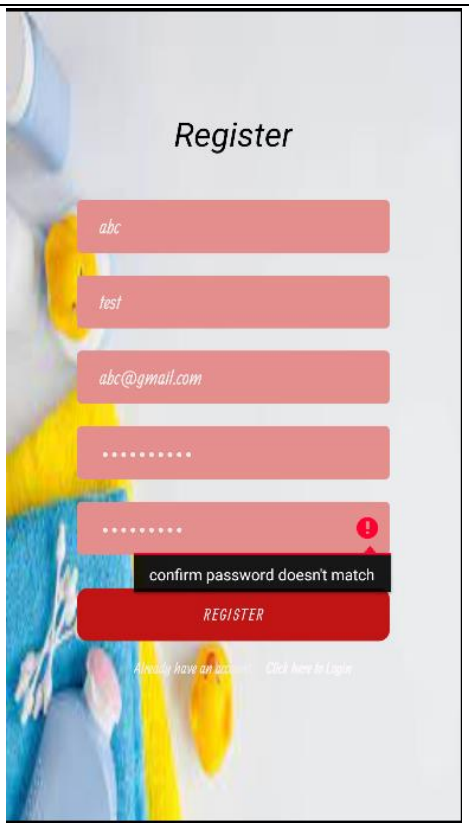
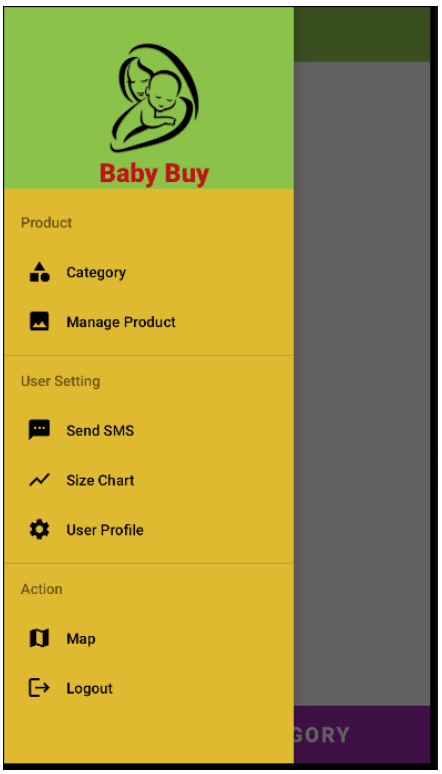
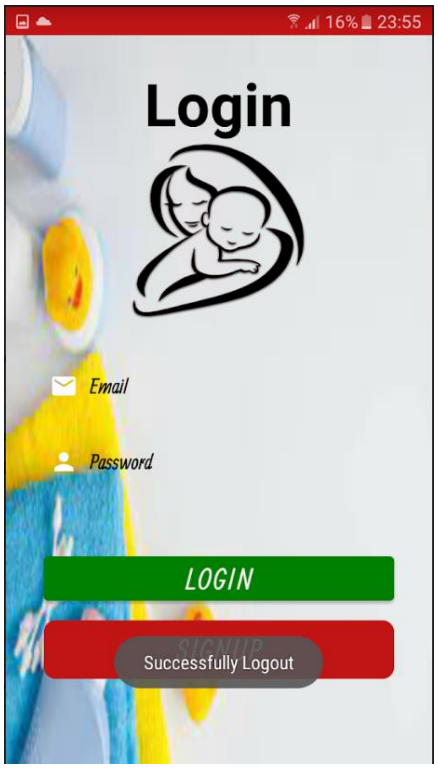
e. Delete category
f. Add product
g. Product location/ map
h. Size chart
i. Log out

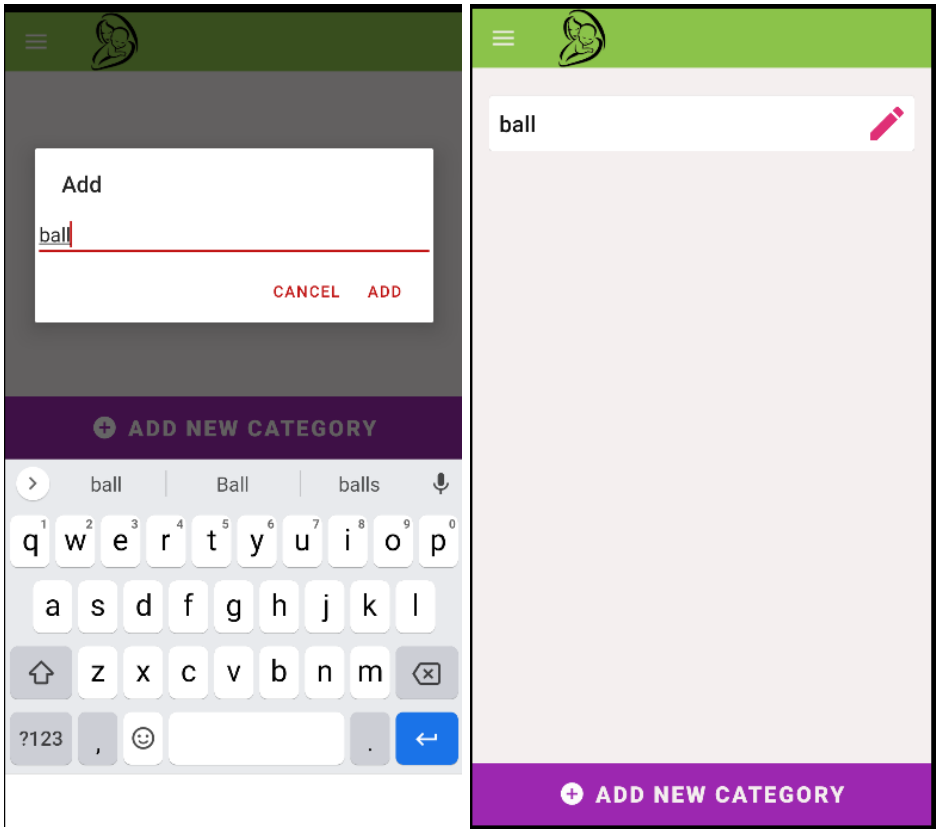
5. Testing table

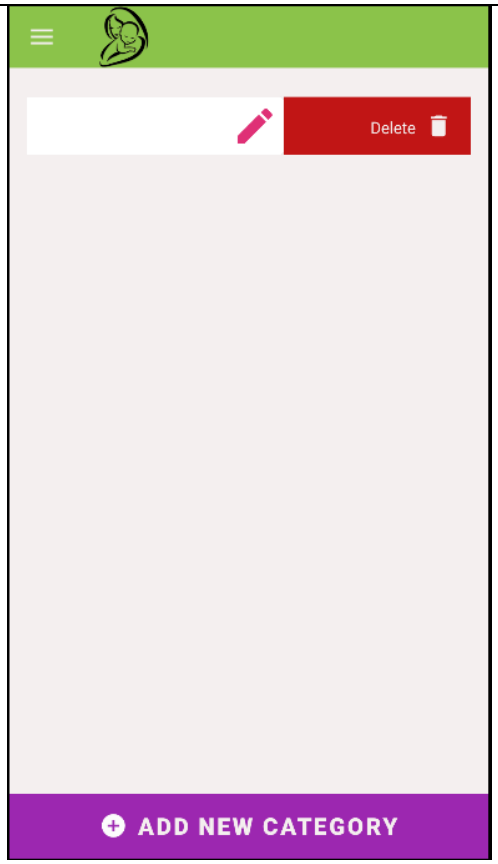
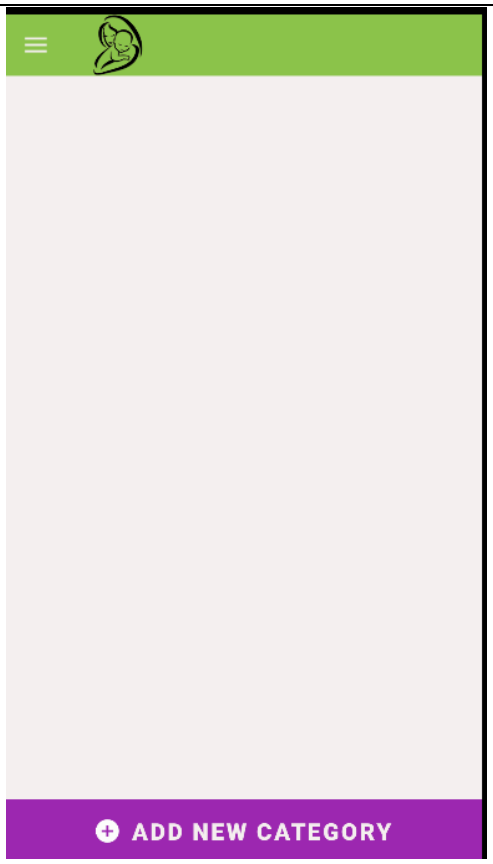
SN	Test case	Expected Output	Actual Output	Test Status
1	Login	User should be able to logged into the application	User was logged into the application and was redirected to home screen	Passed
	Screen shot: 			
2	Login if wrong password or	User should be able to logged into the application and error should display	Error is displayed	Passed

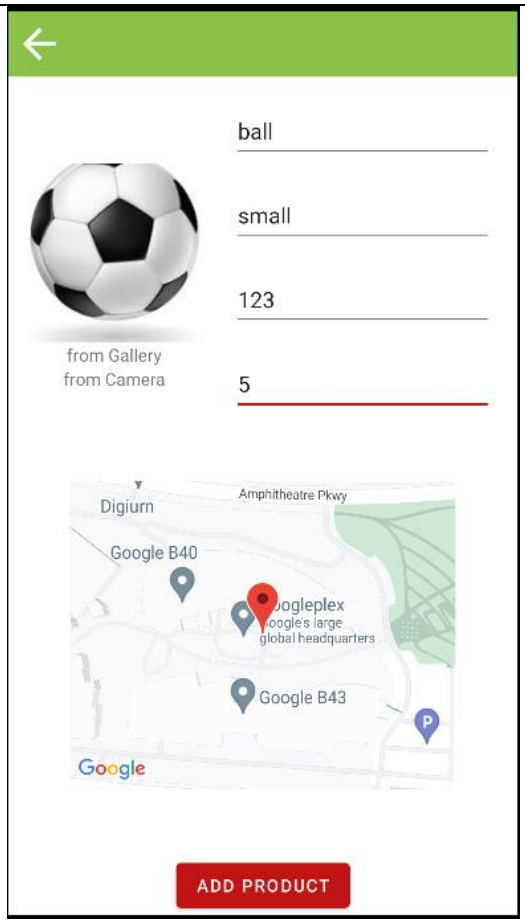
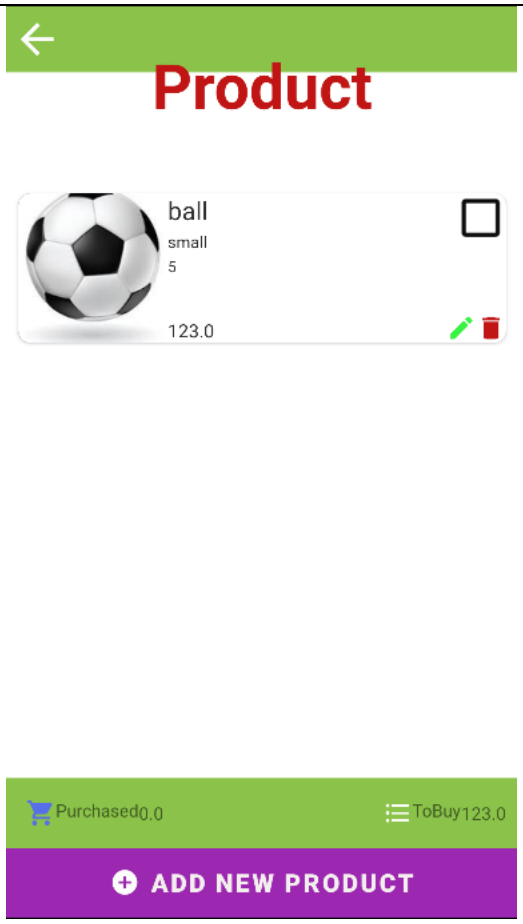
	email provided			
	Screen shot:   			
3	Register	User should be able to register their account	User was able to register their account	passed

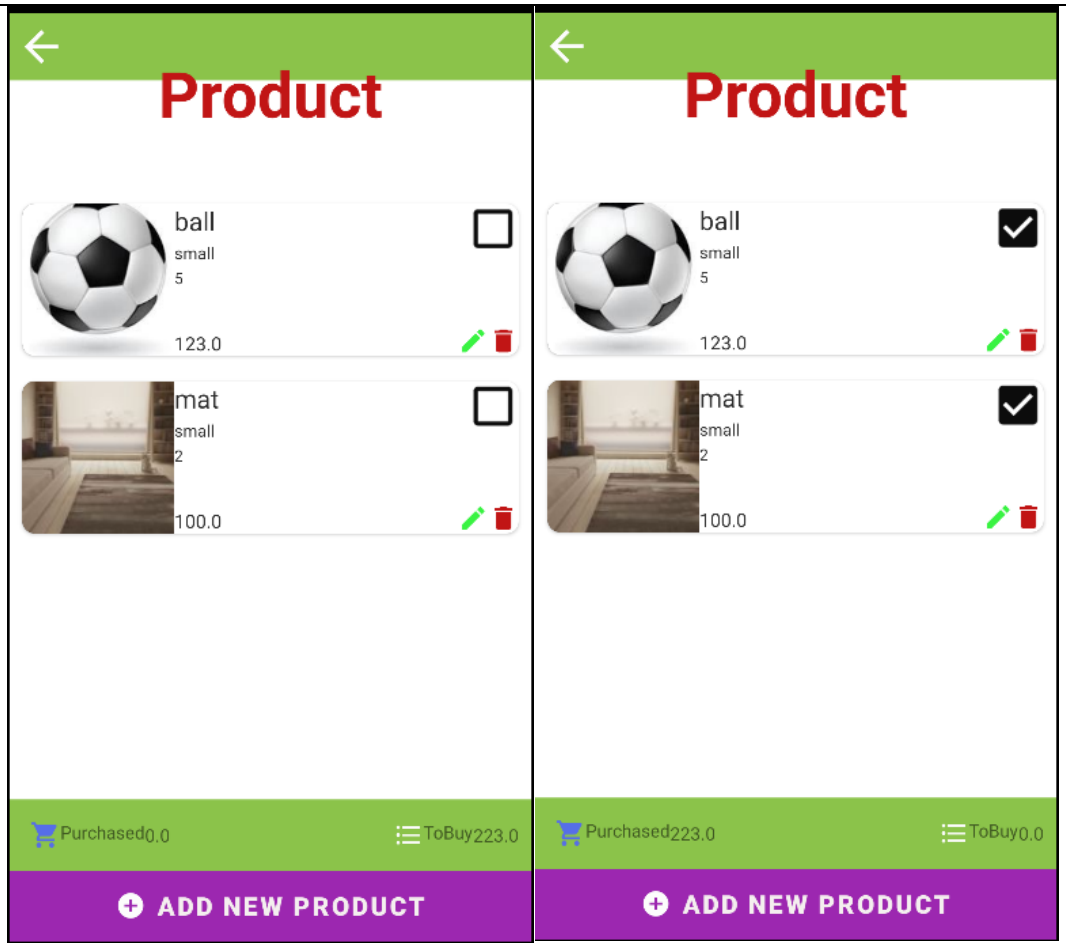
	Screenshot:   			
4	Register if valid credential is not provided	User shouldn't be able to register their account	User wasn't able to register their account	Passed
	Screenshot:			

	 			
5	Log out	User should be able to logout	User was logout	Passed
	Screenshot:  			
6	Add category	User should be able to add category	User was able to add category	passed

	Screenshot: 			
7	Delete category	User should be able to delete category after swiping left	User was able to delete category after swiping left	passed
	Screenshot:			

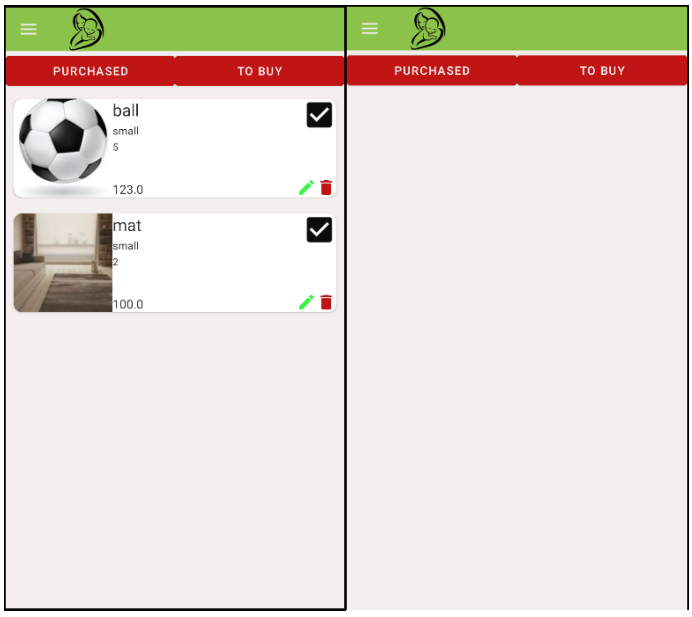
	 			
8	Add product	User should be able to add product	User was able to add product	passed
	Screenshot:			

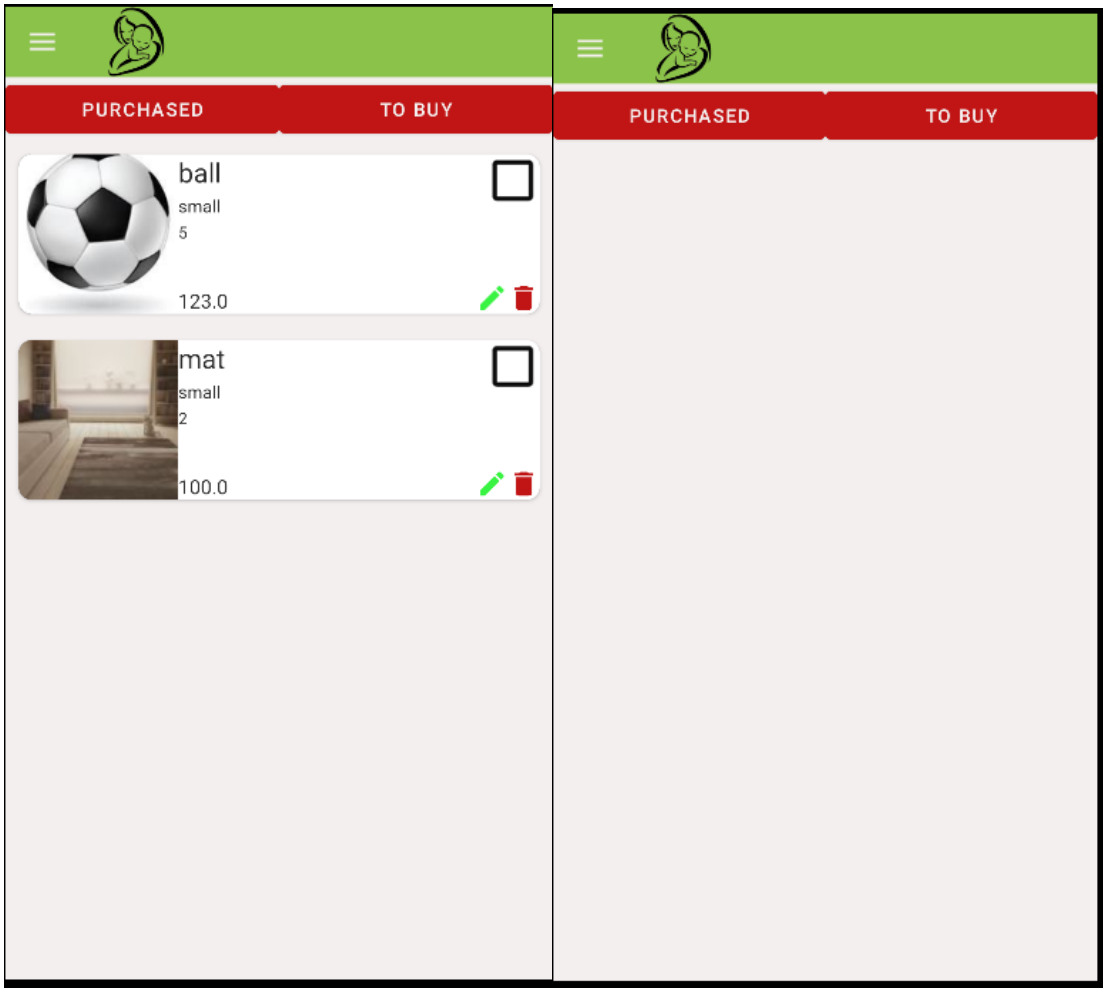
	 			
9	Price calculation	App must calculate the total price of product	App calculated the total price of product	pass
	Screenshot:			

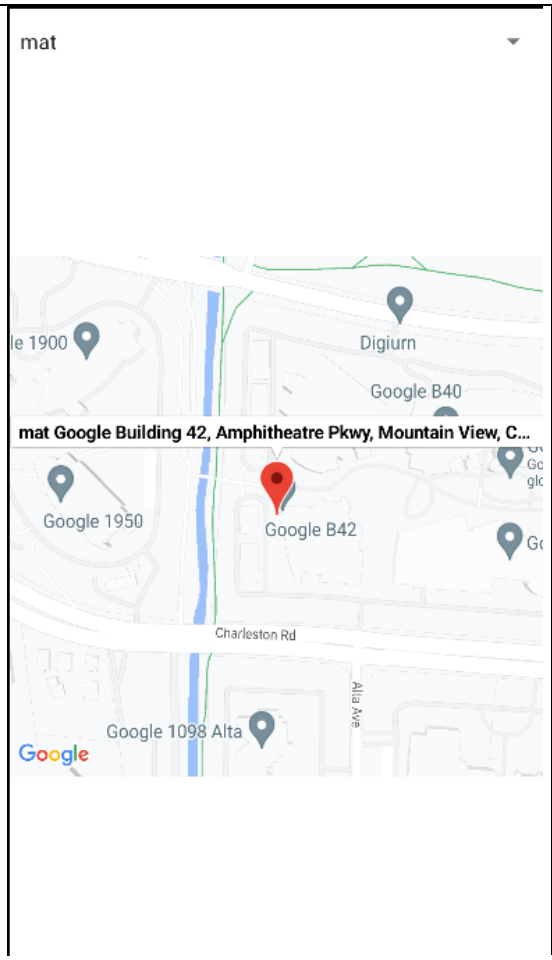
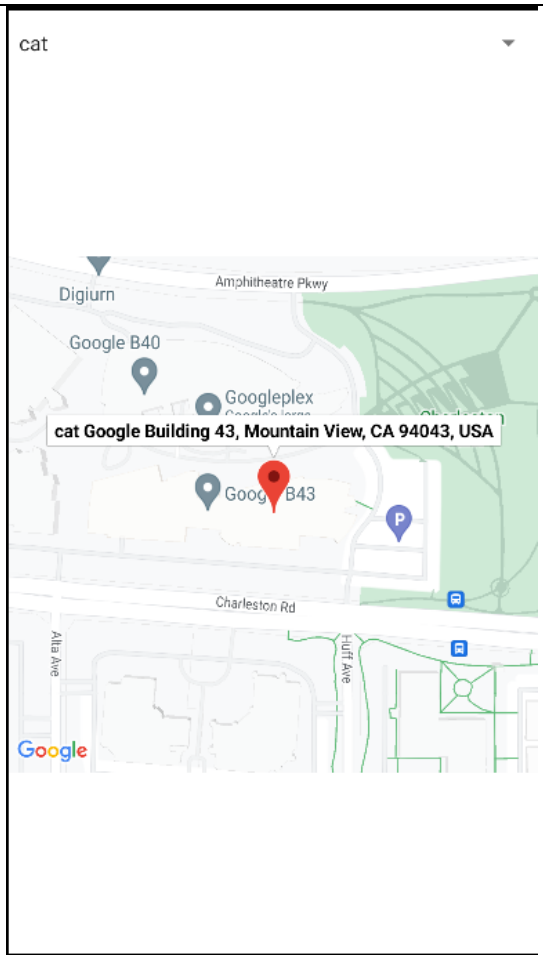


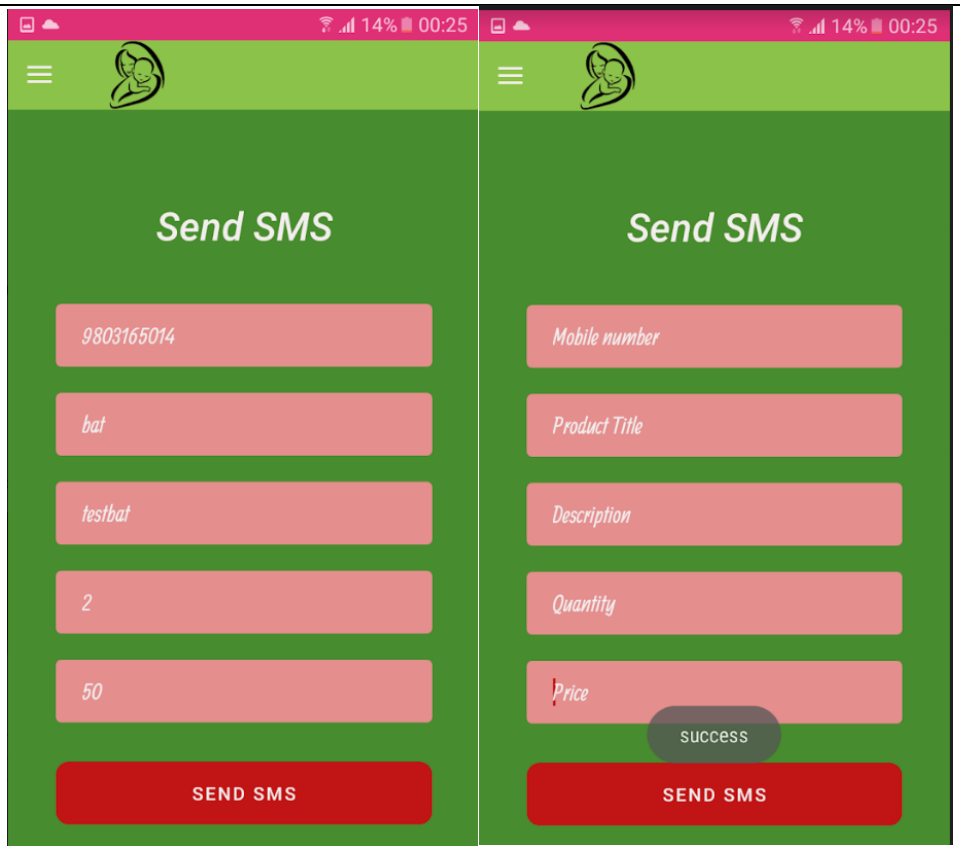
10	Purchase list	User must see the product in product list if they click the product	The product in product is shown on purchase list	Passed

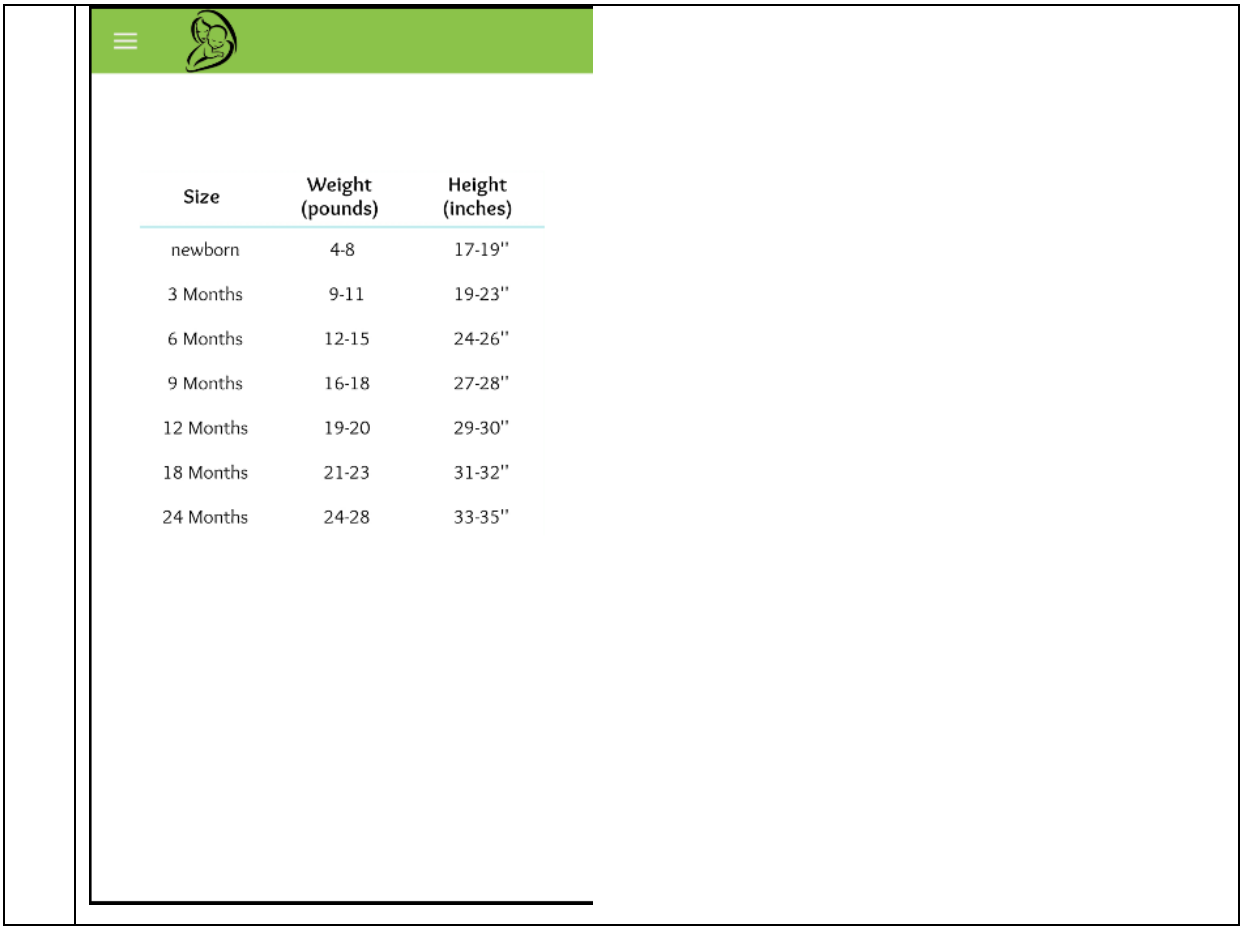
Screenshot:



11	To buy list	User must see the product in to buy list if they unclick the product	The product in product is shown on to buy list	passed
	Screenshot:  			
12	Geotag	User must be able to locate the location of product	User was able to locate the location of product	Passed
	Screenshot:			

	 			
13	SMS	User must be able to send the detail of the product	User was able to send the detail of the product	Passed
	Screenshot:			

				
14	Chart	User must be able to see size chart	User was able to see size chart	Passed
	Screenshot:			



Size	Weight (pounds)	Height (inches)
newborn	4-8	17-19"
3 Months	9-11	19-23"
6 Months	12-15	24-26"
9 Months	16-18	27-28"
12 Months	19-20	29-30"
18 Months	21-23	31-32"
24 Months	24-28	33-35"

Evaluation and Recommendation

The app's user interface is incredibly user-friendly and simple to use for users of all ages. The software has elements that any parent would require when shopping for their baby, such as the ability to add different products that they need for their baby as well as the place where they need to go to purchase those items. It is critical for the app to be user pleasant, simple to browse and use, and distinct from other existing apps.

The Baby Buy app is intended to assist expectant parents in preparing for the arrival of their kid by presenting a list of products that must be purchased and allowing them to seek the assistance of their contacts in making those purchases. It has a user-friendly design with a clear interface and a well-organized structure, demonstrating that the creators worked hard to create a user-friendly software. Nonfunctional elements include effective authentication and permission processes, easy-to-understand error messages, and a user-friendly interface using swiping movements rather than an abundance of buttons. Overall, it provides its consumers with a convenient and intuitive experience.

There are a few recommendations about how to enhance the Baby Buy app. One enhancement might be the inclusion of validation messages in the add/edit item modules and on the Signup page to assist users in determining whether their information is accurate. A profile management page where users can upload profile images and change their passwords might also be useful for password recovery and customization. Furthermore, allowing users to choose between bright and dark modes while adding or modifying things on pages might improve the design. Finally, introducing a search option that allows users to enter parameters such as a baby item's name, description, and price range and then populate results from other web sites may make it easier for customers to compare items and make informed purchase selections.

References

Fling, B., 2009. Mobile design and development: Practical concepts and techniques for creating mobile sites and Web apps. *O'Reilly Media,inc.*

Li, M., Lei, G. & JIn, W., 2014. Research and Development of Mobile Application for. *International Journal of Multimedia and Ubiquitous Engineering,* Volume 9, pp. 187-198.

Malavolta, I., 2016. Beyond Native Apps:Web Technologies to the Rescue!. *Vrije Universiteit Amsterdam, The Netherlands.*

O, O. O., T, A. O., A, G. R. & A, O. C., 2014. Mobile Operating Systems and Application Development Platforms: A Survey. *Int. J. Advanced Networking and Applications,* 6(1), pp. 2195-2201.