Cairo University
Faculty of Computers and Artificial intelligence
Department of Computer Sciences

Corner

Supervised by

Dr. Khaled Wassif

TA. Ashraf Mohey

Implemented by

20180246	*Mohamed Yosry Abdelaal*
20180239	*Mohamed Mahmoud Mansour*
20180218	*Mohamed Gamal Diab*
20180111	*Zyad Mohamed Amin*
20180290	*Manar Atef sayed*

Graduation Project
Academic Year 2021-2022
Final Documentation

Project Documentation

Version 1.0

Contents

Chapter 1 : Introduction

1.1. Abstract

One of the main problems is that everyone is buying new furniture not knowing if it will fit or suit the place. Most of the applications need persons who have knowledge in decoration or it costs too much. So that we want to make the process of decoration easier for people and make the application free to the user. We will use flutter, Unity (AR foundation) and firebase to develop our application.

The main idea of the application is based on Augmented Reality (AR). AR is an interactive experience of a real-world environment and it is a combination of real and virtual worlds. In simple words we can add objects don't exist in reality.

1.2. Background

Corner is a mobile application that can make the process of decoration easier for people who wants to decorate their own place without having background about decorating. And help to choose the most suitable color for the item to the place.

By applying AR users will be able to see their vision of their own place in the real world and ML for matching colors of furniture

1.3. Problem Definition

Lot of people face problem of buying new furniture to their own place. If they want to buy new furniture they must consider many factors like color of the furniture, length, width, height and where to be placed but most importance that if the furniture like the people who want to buy. Not to mention time and money which are wasting on finding the appropriate furniture with perfect color, size and most importantly if it is like them or not.

These problems almost face everyone who wants to decorate by themselves with no background of how to do that. Our project tries to solve these problems and make it easier to decorate any place with the appropriate pieces of furniture and color by using AR to help users to see their vision of their place in closest thing to reality and help to modify it. Also it saves time and money instead of go to different stores and search for the appropriate furniture that suits their taste.

1.4. Stakeholders

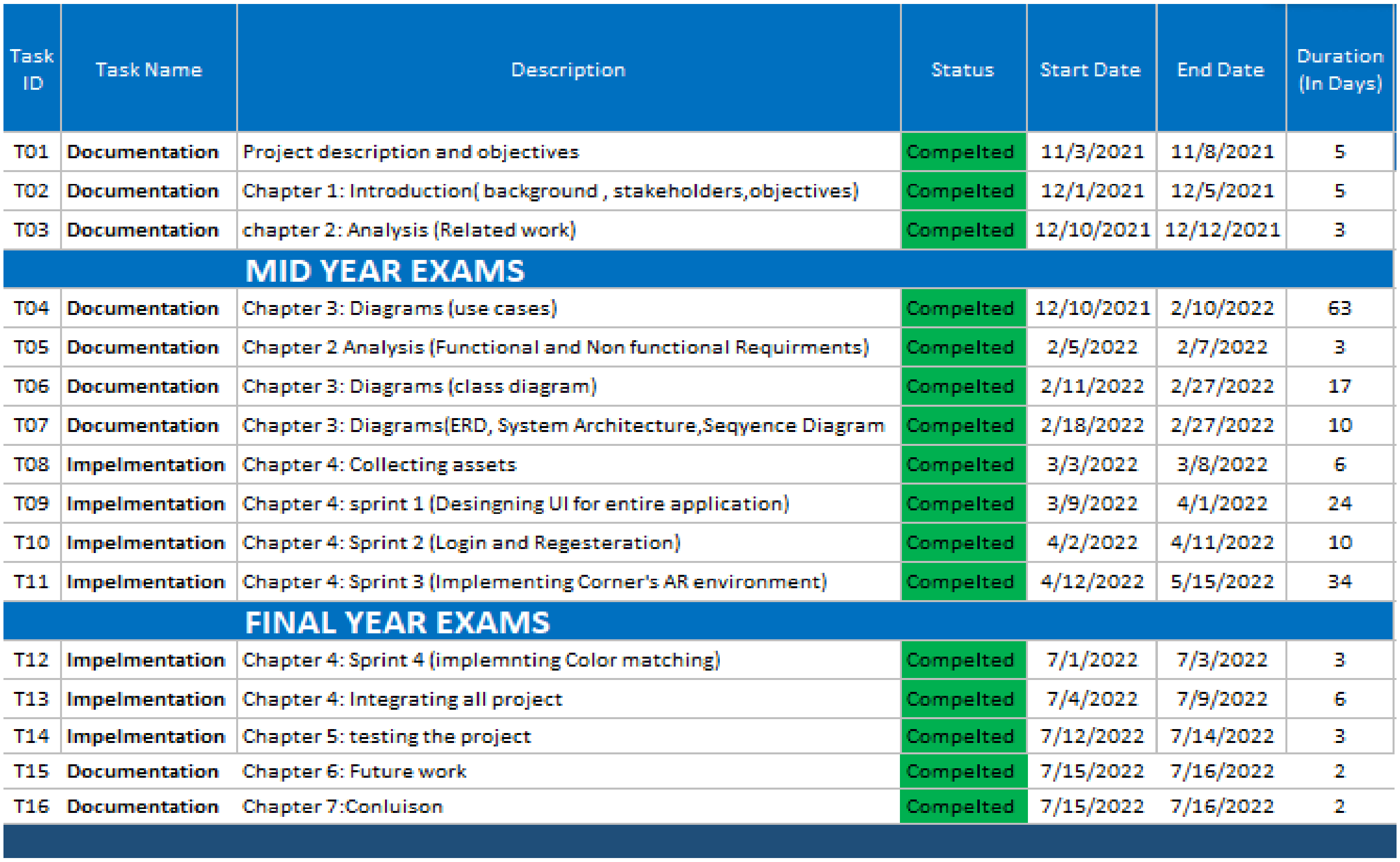
Target Users:

-Normal Users & Decorators

Owners:

- Project team members and supervisors.

1.5. Work plan



Task ID	Task Name	Description	Status	Start Date	End Date	Duration (In Days)
T01	**Documentation**	Project description and objectives	Compelted	11/3/2021	11/8/2021	5
T02	**Documentation**	Chapter 1: Introduction(background , stakeholders,objectives)	Compelted	12/1/2021	12/5/2021	5
T03	**Documentation**	chapter 2: Analysis (Related work)	Compelted	12/10/2021	12/12/2021	3
		MID YEAR EXAMS				
T04	**Documentation**	Chapter 3: Diagrams (use cases)	Compelted	12/10/2021	2/10/2022	63
T05	**Documentation**	Chapter 2 Analysis (Functional and Non functional Requirments)	Compelted	2/5/2022	2/7/2022	3
T06	**Documentation**	Chapter 3: Diagrams (class diagram)	Compelted	2/11/2022	2/27/2022	17
T07	**Documentation**	Chapter 3: Diagrams(ERD, System Architecture,Seqyence Diagram	Compelted	2/18/2022	2/27/2022	10
T08	**Impelmentation**	Chapter 4: Collecting assets	Compelted	3/3/2022	3/8/2022	6
T09	**Impelmentation**	Chapter 4: sprint 1 (Desingning UI for entire application)	Compelted	3/9/2022	4/1/2022	24
T10	**Impelmentation**	Chapter 4: Sprint 2 (Login and Regesteration)	Compelted	4/2/2022	4/11/2022	10
T11	**Impelmentation**	Chapter 4: Sprint 3 (Implementing Corner's AR environment)	Compelted	4/12/2022	5/15/2022	34
		FINAL YEAR EXAMS				
T12	**Impelmentation**	Chapter 4: Sprint 4 (implemnting Color matching)	Compelted	7/1/2022	7/3/2022	3
T13	**Impelmentation**	Chapter 4: Integrating all project	Compelted	7/4/2022	7/9/2022	6
T14	**Impelmentation**	Chapter 5: testing the project	Compelted	7/12/2022	7/14/2022	3
T15	**Documentation**	Chapter 6: Future work	Compelted	7/15/2022	7/16/2022	2
T16	**Documentation**	Chapter 7:Conluison	Compelted	7/15/2022	7/16/2022	2

1.6. The used tools

We used Android Studio to make Flutter application and the graphical user interface, Firebase was used to register users and store furniture objects by linking the application with Firebase, Unity engine to create the Augmented Reality (AR) environment and interactions, and Visual Studio & Visual Studio Code to write the C# scripts used by Unity AR.

Chapter 2 : Related work

Application	Platform	features	Similarity
Myty	android & IOS	Try different furniture products in your own home through AR	✓
		Create virtual rooms with different walls and floor textures. Add windows, doors or furniture.	✗
		Search through thousands of inspirational interior photos.	✗
		Find furniture pieces from world's most famous brands and access portfolios of the best furniture designers.	✗
		Save ideas you like so you can access them anytime.	✓
		Sign up as a furniture brand or a designer and share your furniture collections and 3D models.	✗
Agmenty-AR	android & IOS	Provide buyers and designers with a unique opportunity to "try" your furniture in Augmented Reality.	✓
		Filter furniture by room type or by furniture style	✓
		Many more advance filtering options (color, price and dimensions)	✓
5D Planner	Android, IOS, macOS, windows & web	Tips and ideas from interior design layouts.	✗
		Use ready-made projects and add your design, furniture, decor, floors, etc. to them.	✗
		Use the 2D mode to create floor plans and design layouts with furniture and other home items, or switch to 3D to explore and edit your design from any angle.	✗
		Edit colors, patterns and materials to create unique furniture, walls, floors and more - even adjust item sizes to find the perfect fit.	✓
		Use the Snapshots feature to capture your design as a realistic image - this adds shadows, lighting and rich colors to make your work look like a photograph	✗
		You can apply hundreds of textures and colors in different combinations, Drag and drop items to any place on your room layout and change the size of any item	✓
		You can view your work and move around in 3D mode (using joystick), or with VR mode	✗

Chapter 3 : System Analysis

3.1. Project specification

3.1.1. Functional Requirements

- New users have to create accounts by email and password.
- Existing users can login in the application by their existing accounts.
- Users can log out to change the account or create a new account.
- User needs to open his phone’s camera to detect the surface and colors in the room wanted to decorate.
- The type of room must be decided by the user to list suitable furniture to the type of chosen room
- User can drag and drop items in empty space and can remove, change color and change the place of the item in the room
- The system suggests a list of colors for dropped items based on the colors in the room. Users have option to change it to the color they want
- System allows user to store chosen items with its aspects
- Admin can add admin and delete admin.
- Admin can upload new furniture to specific category, delete and update existing furniture.
- Admin can add and delete category.

3.1.2. Non-functional Requirements

1. **Performance:**
   - To detect the place's surface and dimensions at a maximum of 1 minute.
   - Rendering the item into the room shall be done in real time.
   - The impact of adding too many items while using the main features shall not affect the performance too much.
   - The layout should allow users to reach the saved item within 3 clicks.

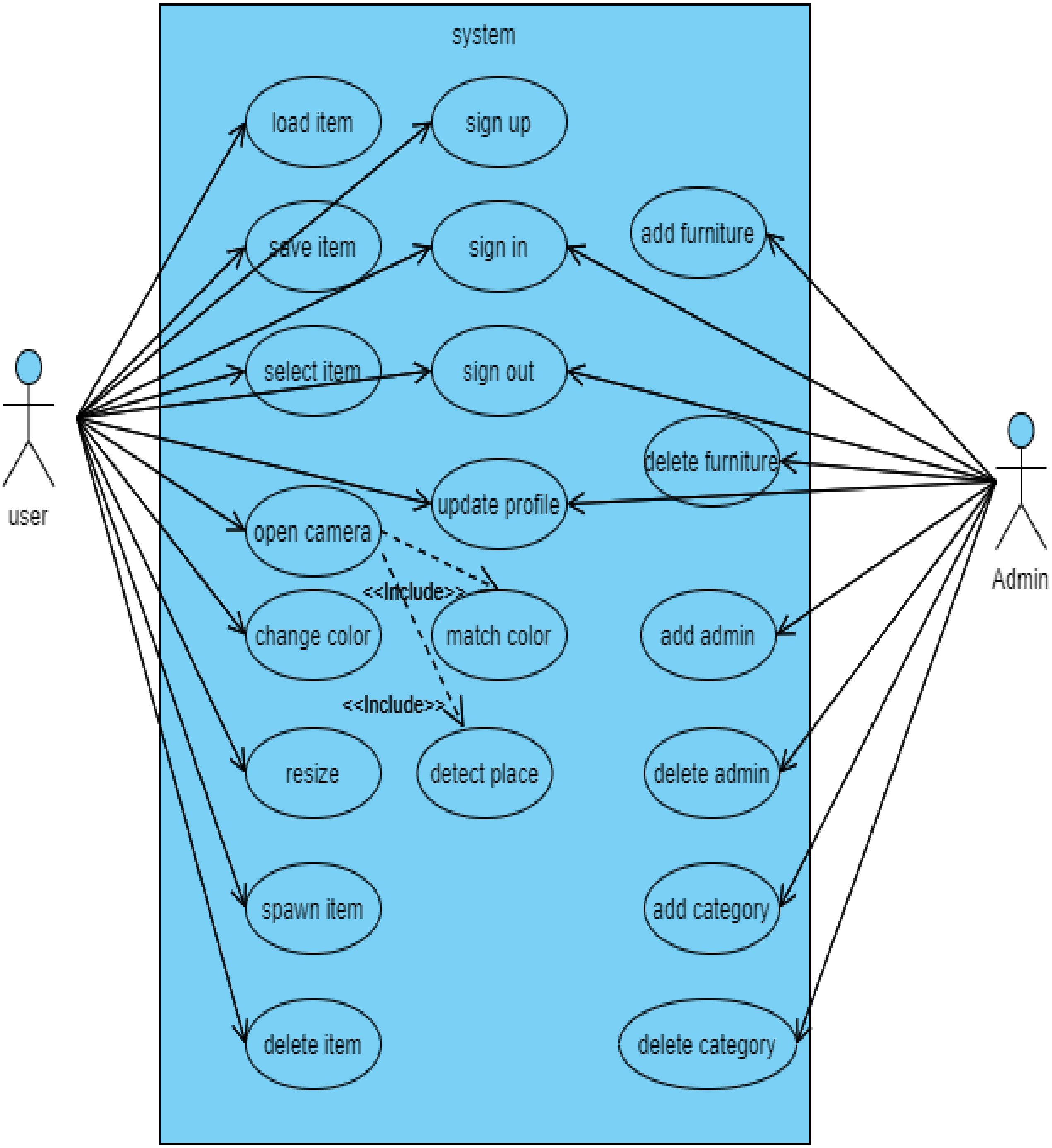
2. **Reliability:** the application could reach a 95 percent chance that the system won’t experience critical failure.

3. **Portability and compatibility:** the application can run on mobile phones with android operating system and hardware support needed to run AR technology.

4. **Usability**:
    - The application should be easy to learn.
    - With one or two clicks the user will start interacting with the application's main features.
    - The users can have their added and chose items saved and can get to them later.
    - The design must be direct, easy and satisfiable.

5. **Availability:** the application will be available 24/7.

3.2. Use Case


system
user
Admin
load item
sign up
save item
sign in
add furniture
select item
sign out
delete furniture
update profile
open camera
<<Include>>
change color
match color
add admin
<<Include>>
resize
detect place
delete admin
spawn item
add category
delete item
delete category

ID	UC_1
Title	Registration
Description	New user have to register in application by entering email and password, a message verification will be sent to the email that the user wrote to verify the user,(password shouldn’t be less than 8 characters)
Primary Actor	User and system
Post conditions	The user is successfully registered
Main success scenario	1. User enter his details and submit it 2. System verifies user 3. After verification user is added to the database
Extensions	1. Error message appears if the user writes a password less than 8 characters 2. The event is not successful if the email is not verified 3. All information needed must be entered by the user to register him
Frequency of use	Only once
Priority	1

ID	UC_2
Title	Log-in
Description	user log in to the app by their existing account, login will be by email and password.
Primary Actor	User and system
Preconditions	The user must be registered in the application
Post conditions	The user is successfully Logged in to the application
Main success scenario	1. The user enters his email and password 2. The system verifies user from the database 3. User can access all system functionalities
Extensions	1. “Try again” message appears if the password or email is incorrect 2. “User is not available” message appears if the user is not registered in the application
Frequency of use	Every time the application is opened
Priority	1

ID	UC_3
Title	Open camera
Description	User need it to use application and start user AR
Primary Actor	User and system
Preconditions	User’s phone supported AR
Post conditions	The camera opened and start to detect place
Main success scenario	1. After user login click to open the camera. 2. System opens the camera and starts to detect the place.
Extensions	1. If the phone does not support AR, send message.
Frequency of use	Every time the application is opened
Priority	1

ID	UC_4
Title	Detect place
Description	When camera opened the app need to detect surface to work
Primary Actor	System
Preconditions	User opened camera
Post conditions	Detect the surface successfully
Main success scenario	1. After the user opened the camera. 2. System starts to detect the surface and show users who hold the phone.
Extensions	1. If phone not supported AR send massage.
Frequency of use	Every time the camera opened

Priority	1

ID	**UC_5**
Title	select item
Description	After detecting the place, the user selects the item from the list of items.
Primary Actor	User.
Preconditions	Successful detect place.
Post conditions	The selected item appear in camera
Main success scenario	1. After the system detects the place, show a list of items. 2. Users select maximum one item.
Extensions	1. User detects the place again. 2. If the user selects the wrong item, cancel this selection and select again.
Frequency of use	More than one
Priority	1

ID	**UC_6**
Title	spawn item
Description	After selecting the item, the user drops the item in the detected place.
Primary Actor	User
Preconditions	Users select one item and appear in camera.
Post conditions	The item dropped in the correct place, correct size and correct position.
Main success scenario	1. Users choose a specific wanted place and drop it. 2. System drop item like the user wanted.

Extensions	1. If an item is dropped in the wrong place, the user can delete it or drop it again.
Frequency of use	Every item selected.
Priority	1

ID	**UC_7**
Title	Change color
Description	Change color of selected item
Primary Actor	User
Preconditions	Users select items and appear in camera.
Post conditions	The item changed the color to user's selected color
Main success scenario	1. User choose change color 2. System show small screen contain colors 3. User choose color
Extensions	1. If the user selected or wanted to change color.
Frequency of use	Every item changed color.
Priority	2

ID	**UC_8**
Title	match color
Description	System detect color of place and choose specific color
Primary Actor	System
Preconditions	System detect color of place right
Post conditions	The system do algorithm correct

Main success scenario	1. System takes time to detect the color of the place. 2. System do algorithm to suggest match colors.
Extensions	1. If the system can't detect the color of the place, do not suggest color.
Frequency of use	Only once
Priority	2

ID	**UC_9**
Title	delete item
Description	User can delete dropped item from the place
Primary Actor	User and item
Preconditions	System can detect the dropped item
Post conditions	The system delete the selected item
Main success scenario	1. System can detect dropped item in place 2. User select item to delete it 3. System delete selected item from the place
Extensions	2. If the system can't detect the dropped item in the place
Frequency of use	Only once
Priority	1

ID	**UC_10**
Title	Resize
Description	Resize of selected item to fit place and user's desire
Primary Actor	User
Preconditions	Users select items and appear in camera.

Post conditions	The item resize to user's desire
Main success scenario	1. User choose resize item 2. System show small screen 3. User start to resize
Extensions	1. If the user wants to resize the item again.
Frequency of use	Many time
Priority	2

ID	**UC_11**
Title	Save items
Description	User select item to save it to return it again
Primary Actor	User
Preconditions	User selected the item
Post conditions	System save item
Main success scenario	1. User select item and save it
Frequency of use	Many times
Priority	2

ID	**UC_12**
Title	Load items
Description	User select item to save it to return it again
Primary Actor	User
Preconditions	User selected the item
Post conditions	System save item

Main success scenario	1. User choose load item 2. System show saved items 3. User selects the item and starts to drop, changing color, etc.
Frequency of use	Many times
Priority	2

ID	**UC_13**
Title	Add category
Description	Admin can add new category to system.
Primary Actor	Admin
Preconditions	
Post conditions	The added category appears to users in categories.
Main success scenario	1. Admin named new category. 2. System adds the category to the categories.
Extensions	1. If the system doesn't add the category. 2. Admin adds the category again.
Frequency of use	Many time
Priority	1

ID	**UC_14**
Title	Delete category
Description	Admin can delete category from the categories list.
Primary Actor	Admin
Preconditions	The category should be available to user.
Postconditions	The deleted category doesn’t appear to users in the categories list.
Main success scenario	1. Admin select category and delete it. 2. System deletes the category from the categories list.
Extensions	1. If the system doesn't delete the category. 2. Admin adds the model again.
Frequency of use	Many time

Priority	1

ID	**UC_15**
Title	Add furniture
Description	Admin can add new furniture to the category.
Primary Actor	Admin
Preconditions	
Postconditions	The added furniture appears to users in the correct category.
Main success scenario	3. Admin after creating or choosing a 3D model. 4. Admin chooses the category of this model. 5. System adds the model to the category.
Extensions	3. If the system doesn't add the model. 4. Admin adds the model again.
Frequency of use	Many time
Priority	1

ID	**UC_16**
Title	Delete furniture
Description	Admin can delete furniture from the category.
Primary Actor	Admin
Preconditions	The item should be available to user.
Postconditions	The deleted furniture doesn’t appear to users in the category.
Main success scenario	3. Admin select model and delete it. 4. System deletes the model from the category.
Extensions	3. If the system doesn't delete the model. 4. Admin adds the model again.
Frequency of use	Many time
Priority	1

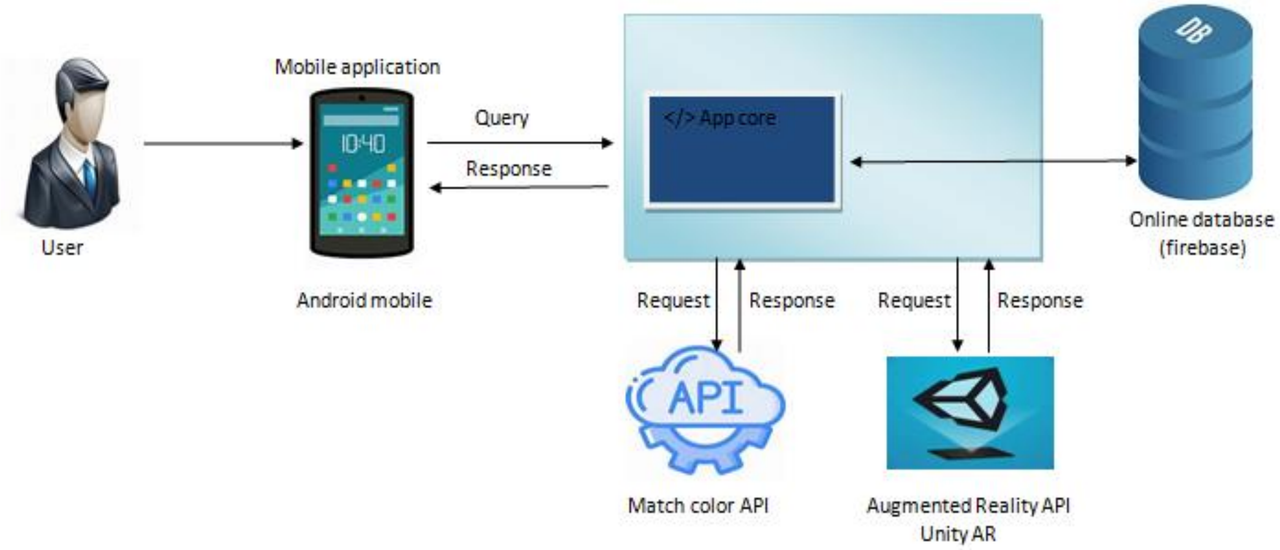
ID	**UC_17**

Title	Add Admin
Description	Admin can add new admin to the system.
Primary Actor	Admin
Post conditions	The added admin has the admin permissions.
Main success scenario	1. Admin adds new admin and fill his data. 2. System adds the new admin to the system.
Extensions	1. If the system doesn't add the new admin. 2. Admin adds the new admin again again.
Frequency of use	Many time
Priority	1

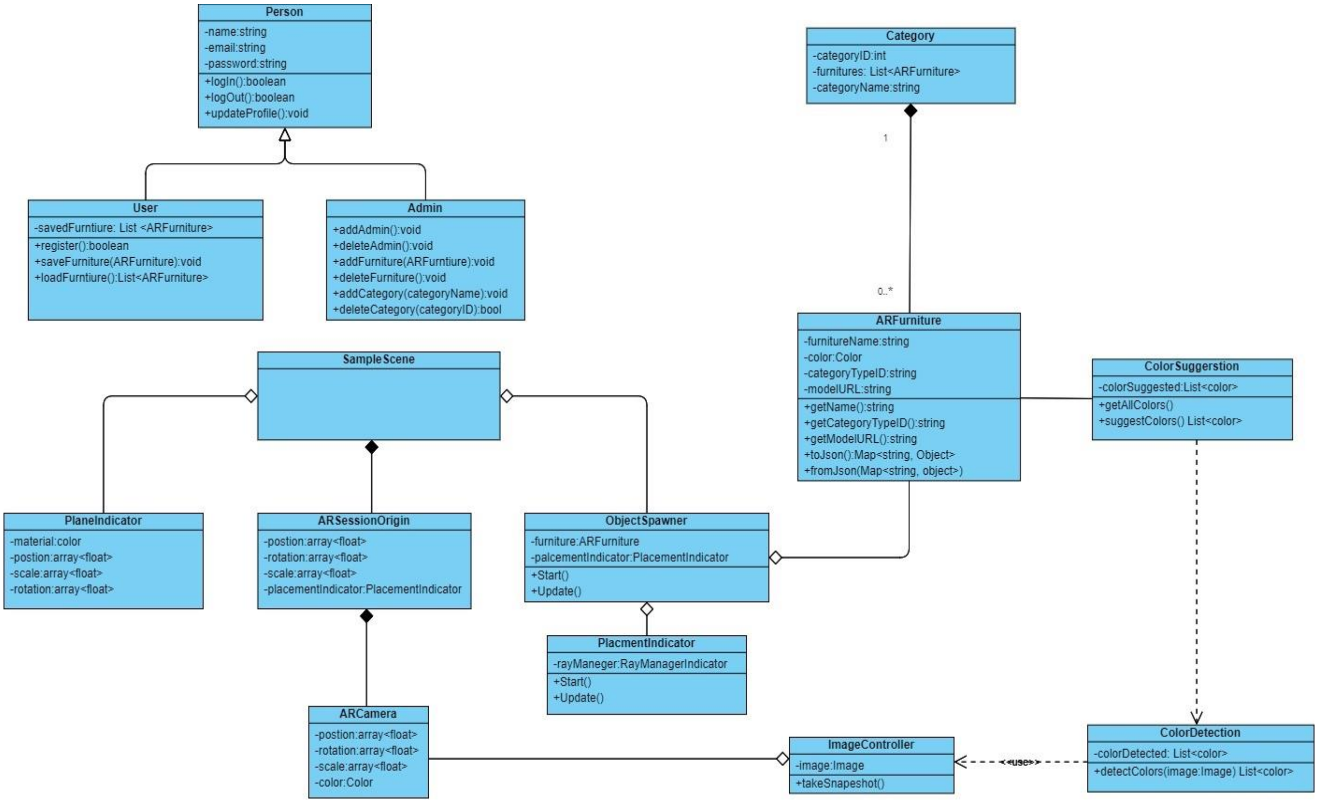
ID	**UC_18**
Title	Delete admin
Description	Admin can delete another admin from the system..
Primary Actor	Admin
Preconditions	The system should have at least two admins.
Postconditions	The deleted admin removed from the system.
Main success scenario	Admin select another admin and delete it. System delete the selected admin from the system.
Extensions	If the system doesn't delete the admin. Admin delete the admin again.
Frequency of use	Many time
Priority	1

Chapter 4 : System Design

4.1. System Architecture

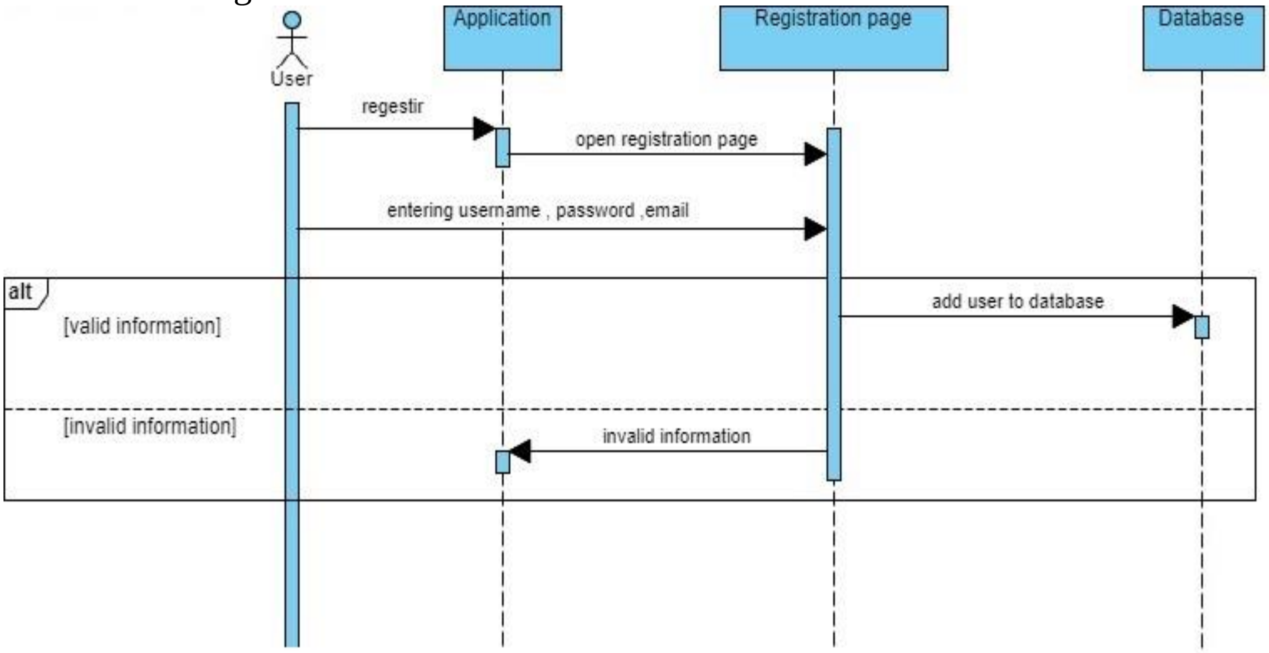


4.2. Class Diagram

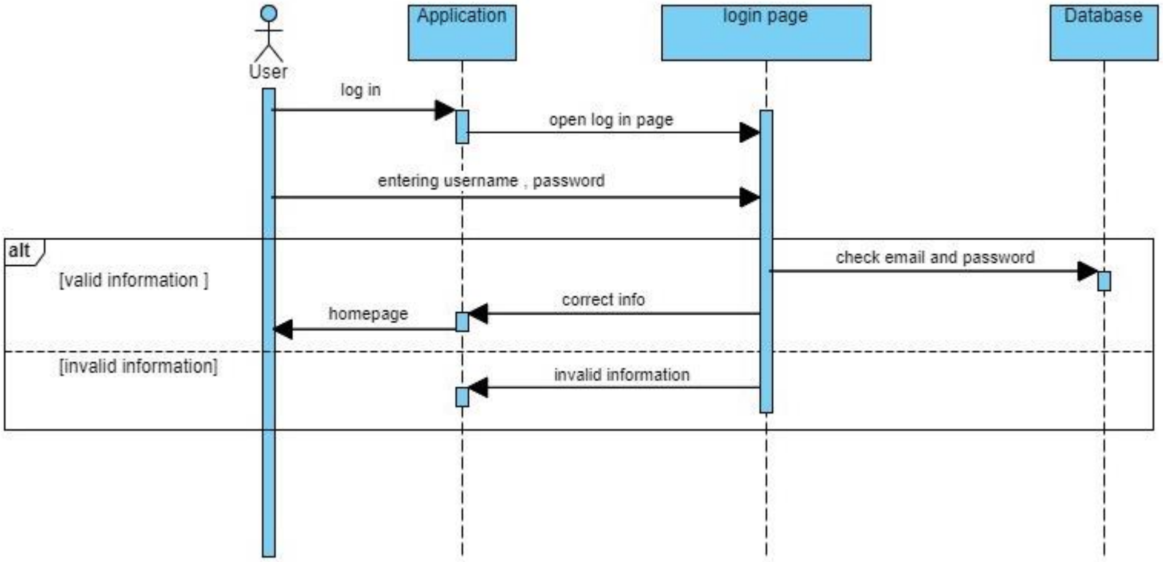


4.3. Sequence Diagram

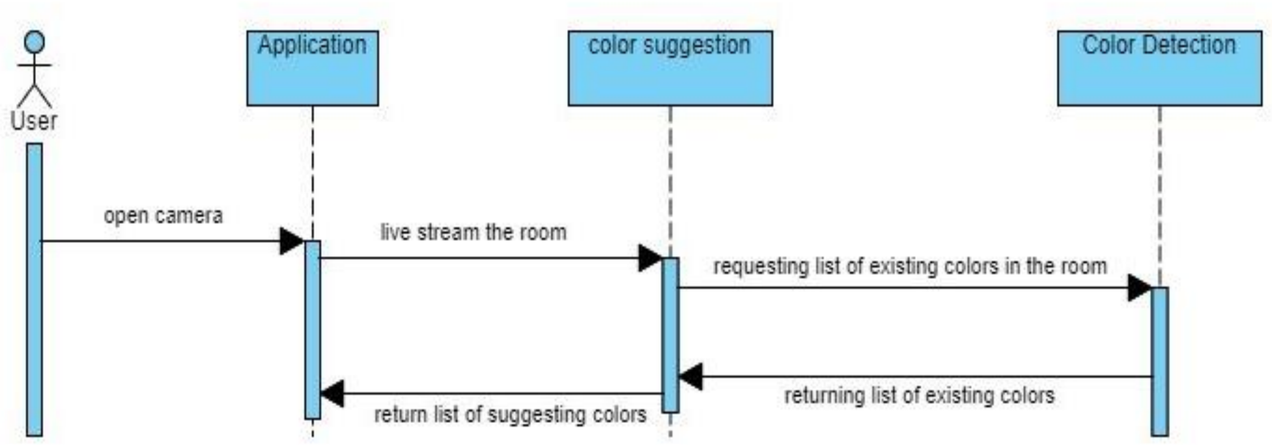
4.3.1. Registration



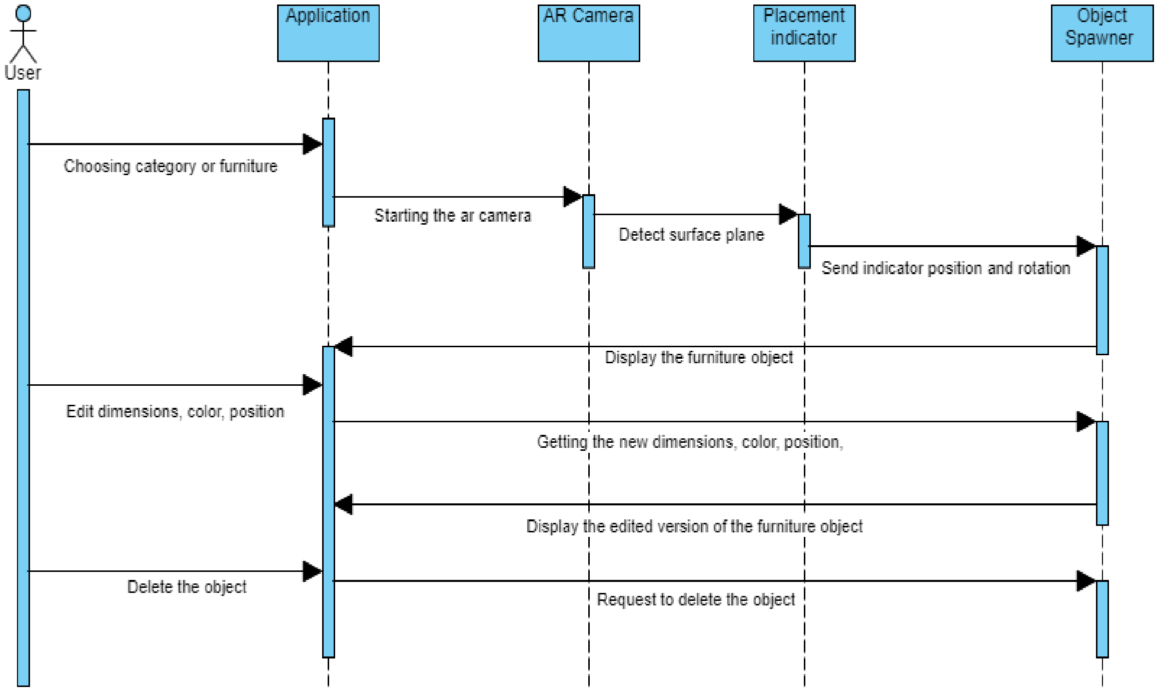
4.3.2. Log in



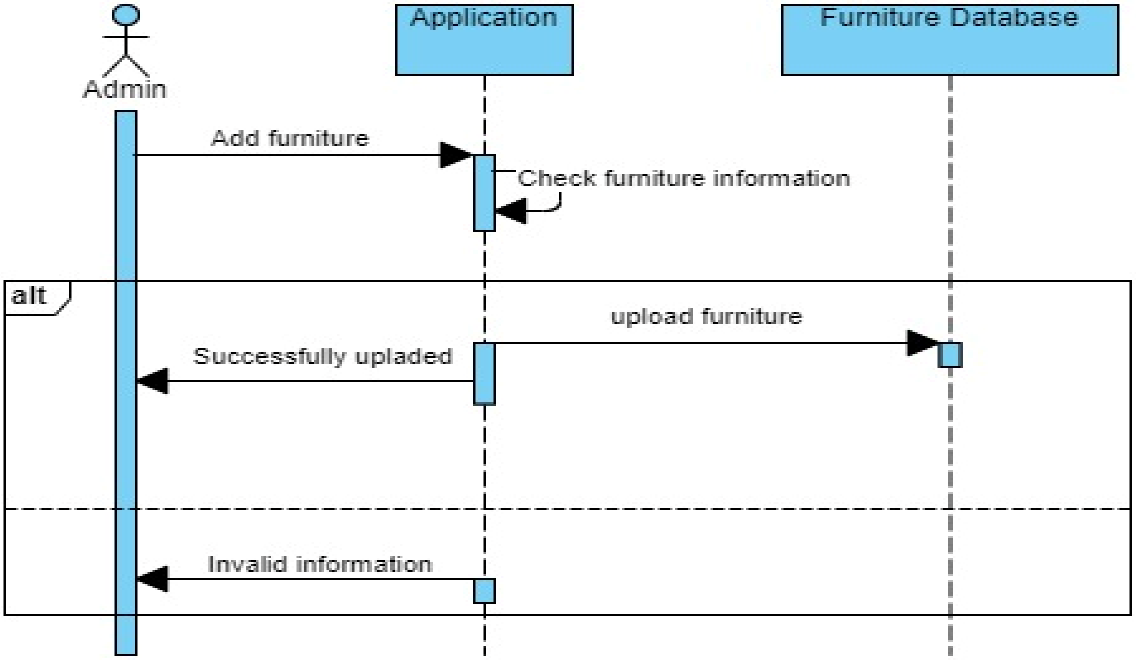
4.3.3. Color suggestion


User
Application
color suggestion
Color Detection
open camera
live stream the room
requesting list of existing colors in the room
returning list of existing colors
return list of suggesting colors

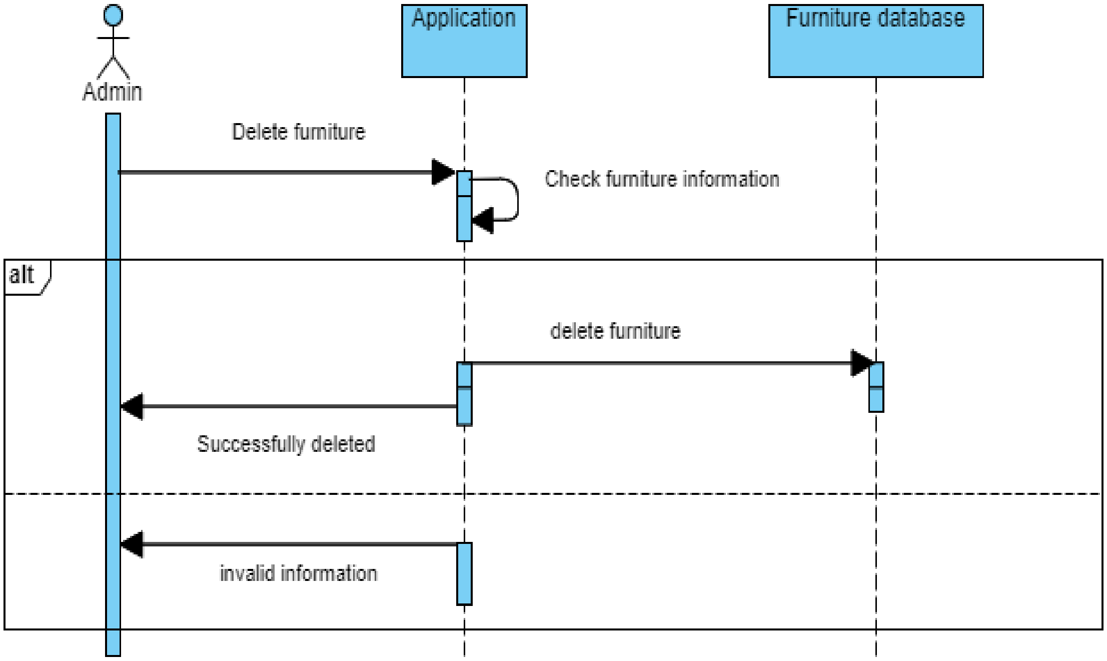
4.3.4. Object spawning


User
Application
AR Camera
Placement indicator
Object Spawner
Choosing category or furniture
Starting the ar camera
Detect surface plane
Send indicator position and rotation
Display the furniture object
Edit dimensions, color, position
Getting the new dimensions, color, position,
Display the edited version of the furniture object
Delete the object
Request to delete the object

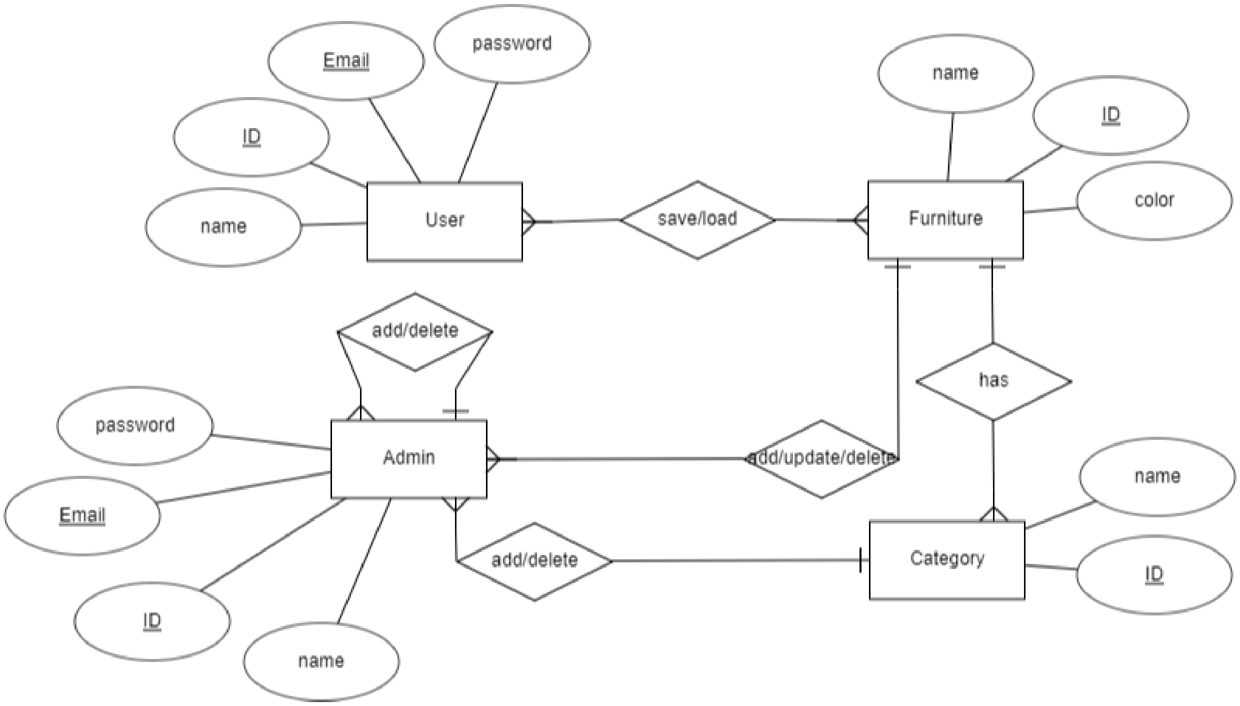
4.3.5. Add furniture



4.3.6. Delete furniture



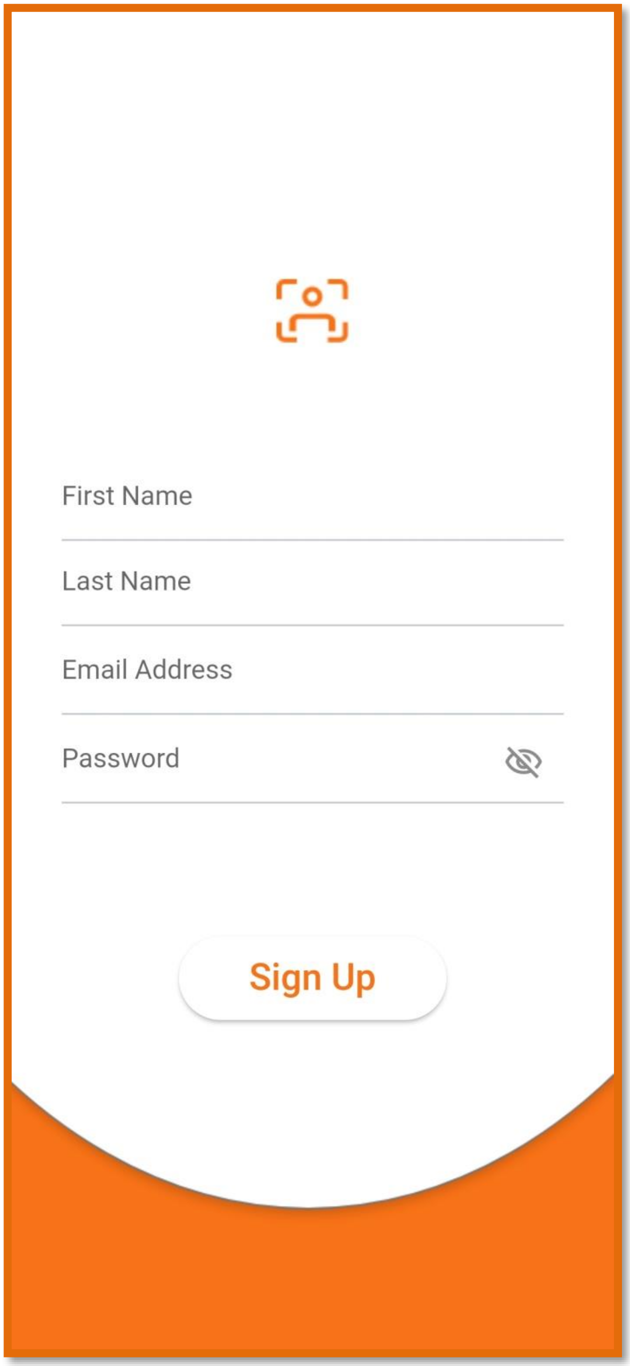
4.4. ERD


Email
password
ID
name
User
save/load
Furniture
name
ID
color
add/delete
has
password
Admin
add/update/delete
Email
name
ID
add/delete
Category
ID
name

4.5. System GUI Design

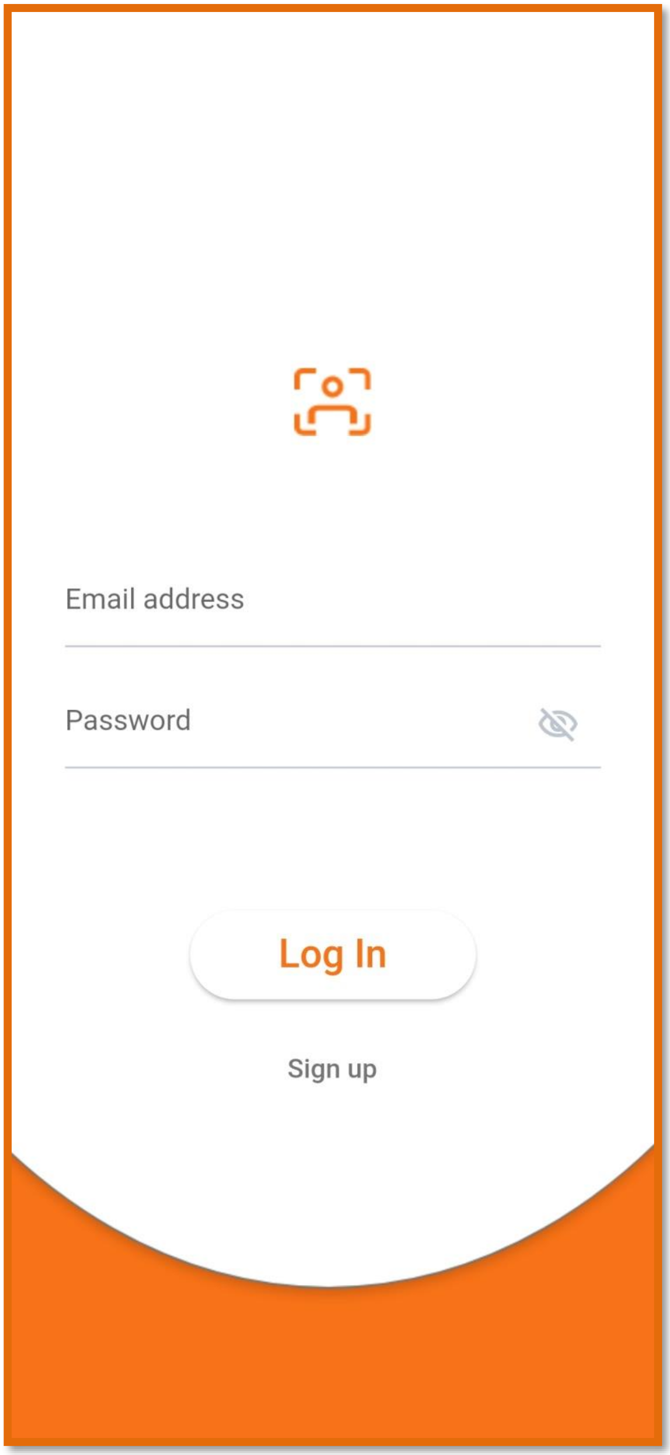
4.5.1. Signup

This is the registration page that the user will use if he doesn’t already have an account.



4.5.2. Login

This is the page that the user will use if he already has an account.

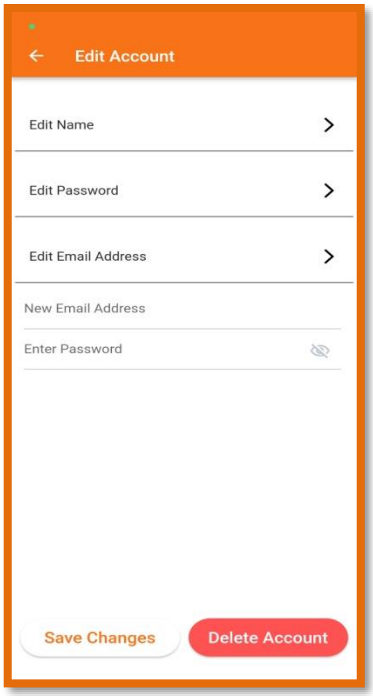


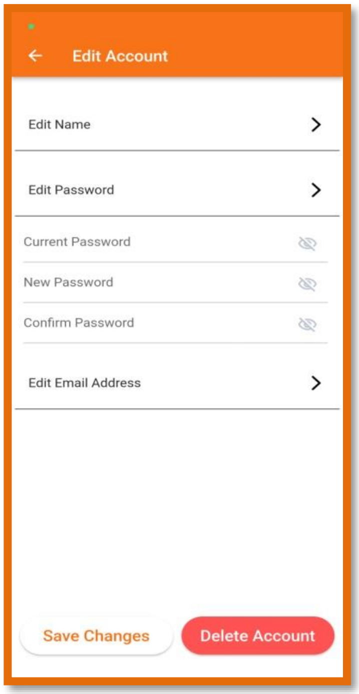
Email address

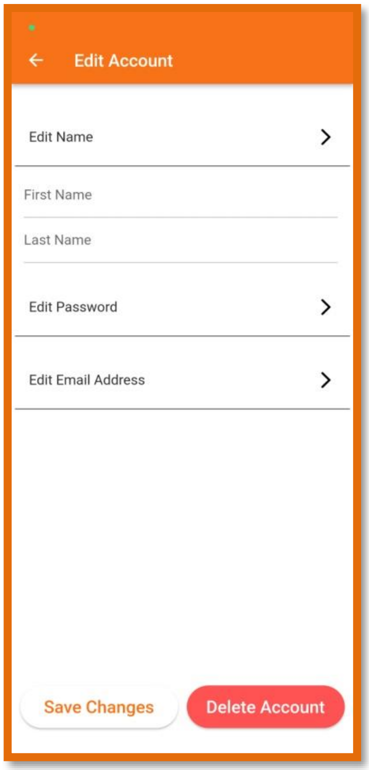
Password

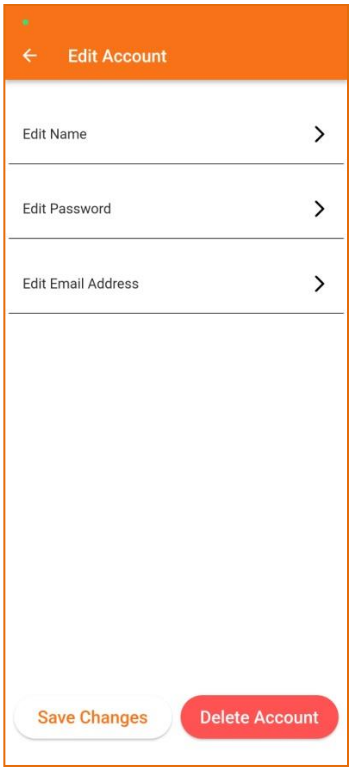
4.5.3. Edit Profile

This is the page that has the user information to edit (first name, last name, password, email).



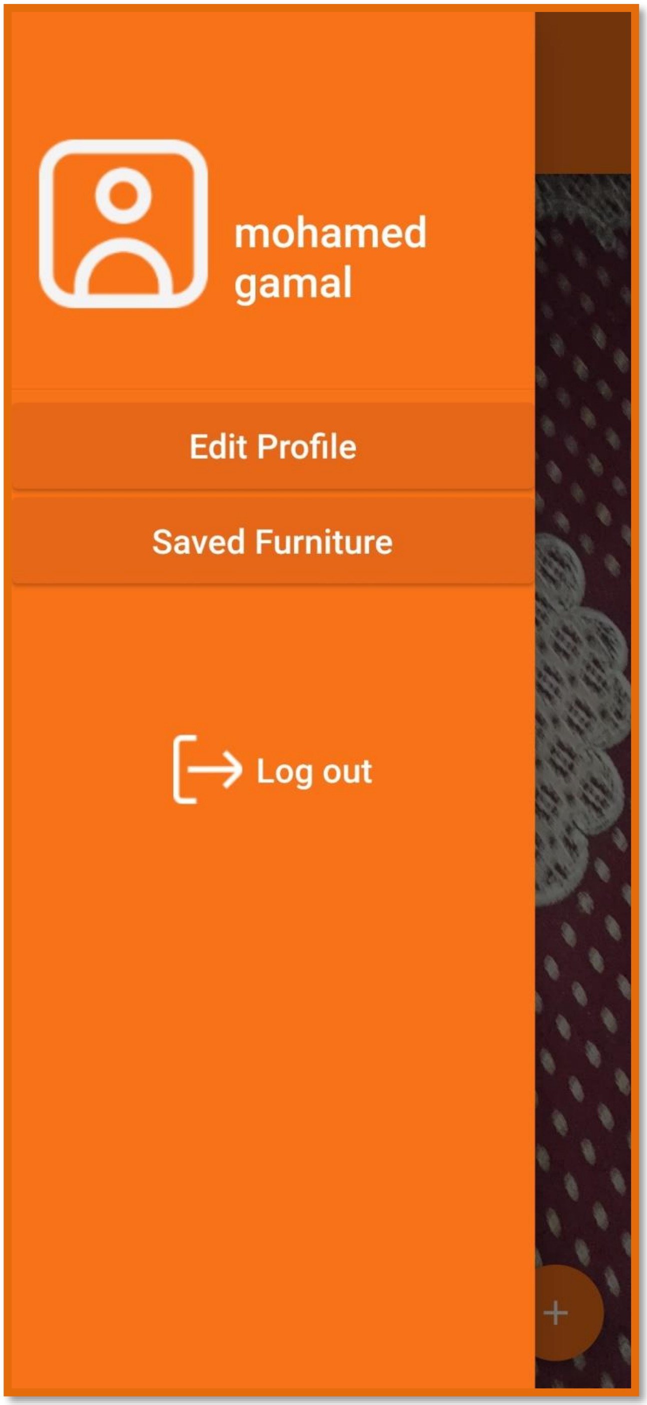






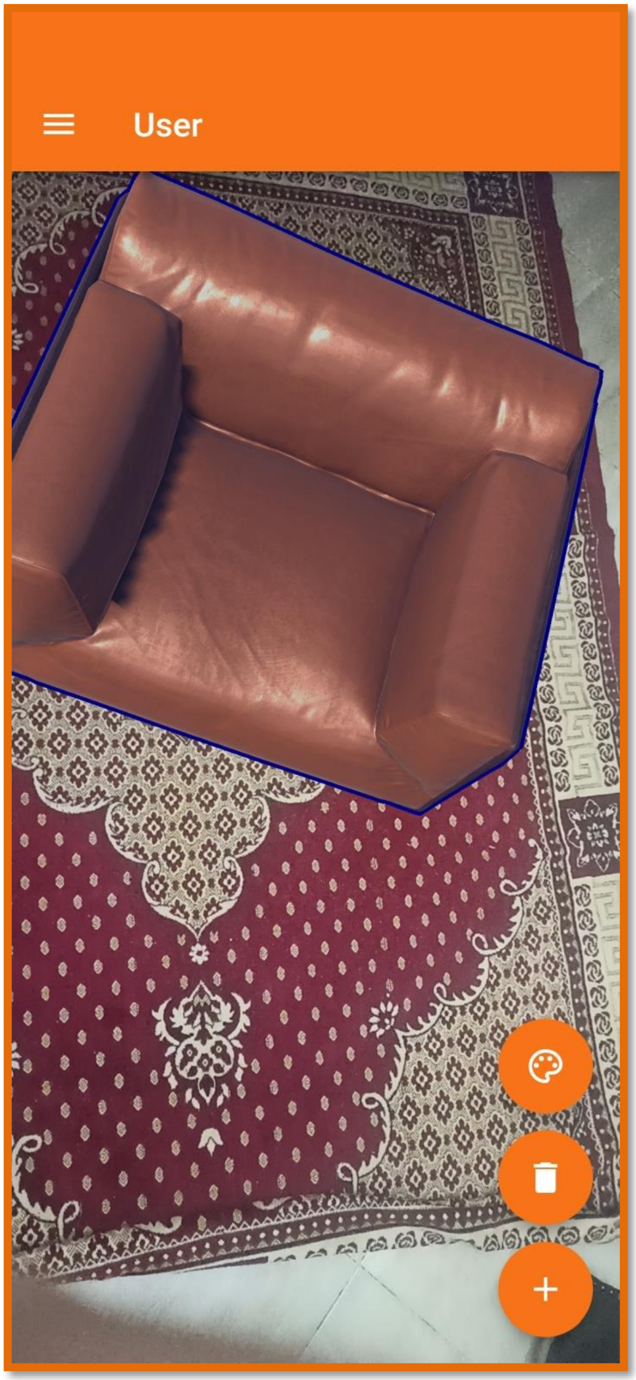
4.5.4. User side menu

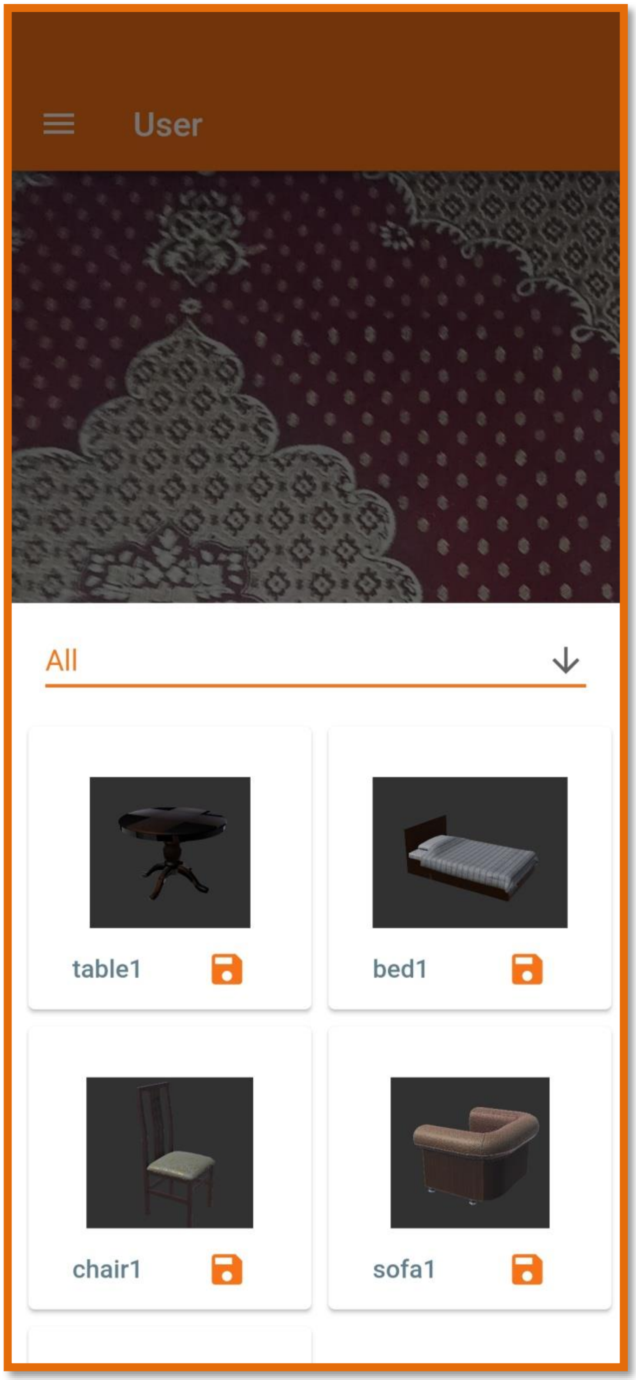
This is the page that has the user name, edit profile, saved furniture and logout.



4.5.5. Spawn furniture

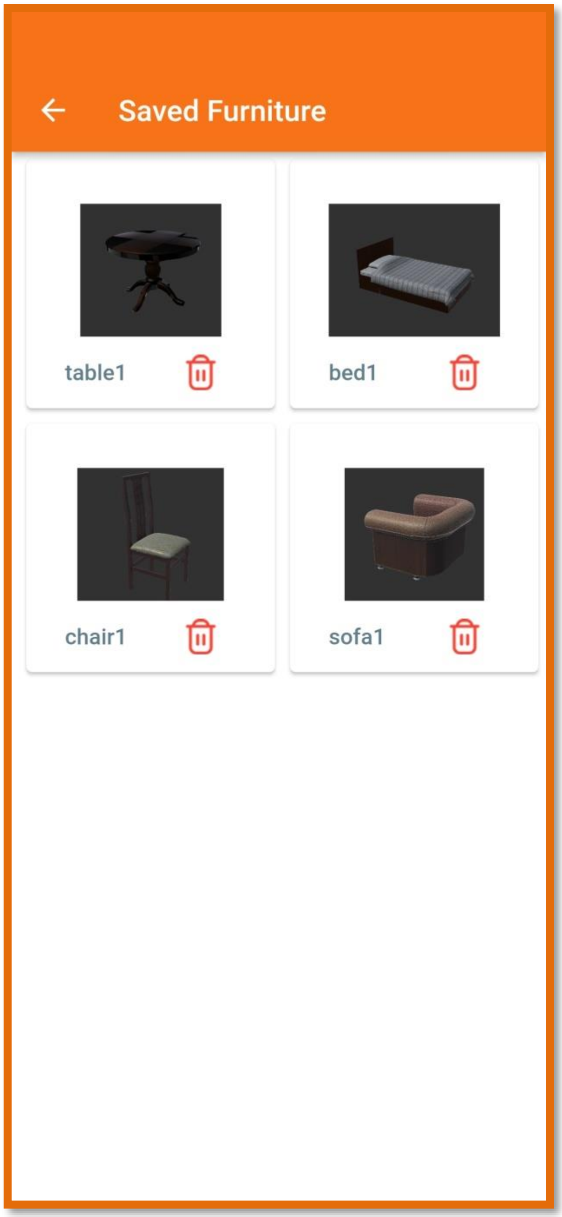
This is the page that has the furniture can user spawn it.





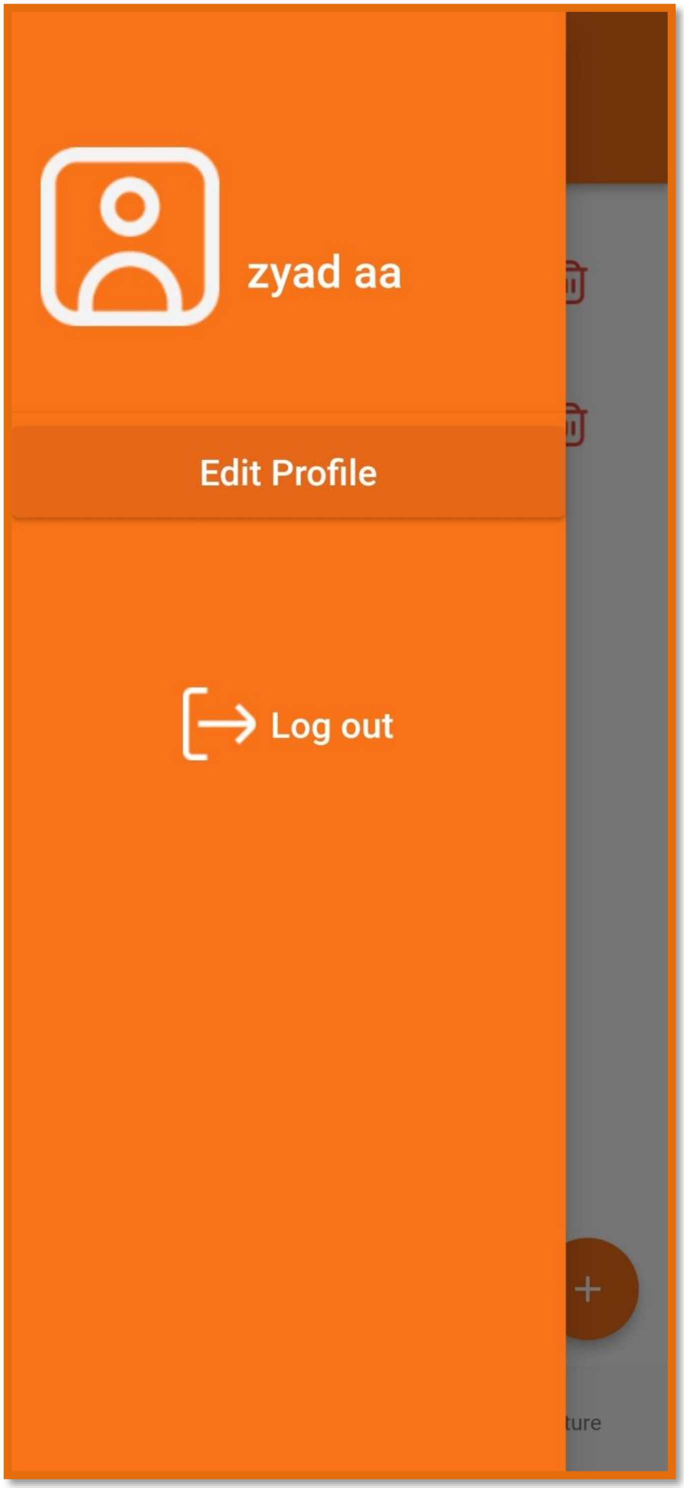
4.5.6. Saved furniture

This is the page that has the saved furniture by user.



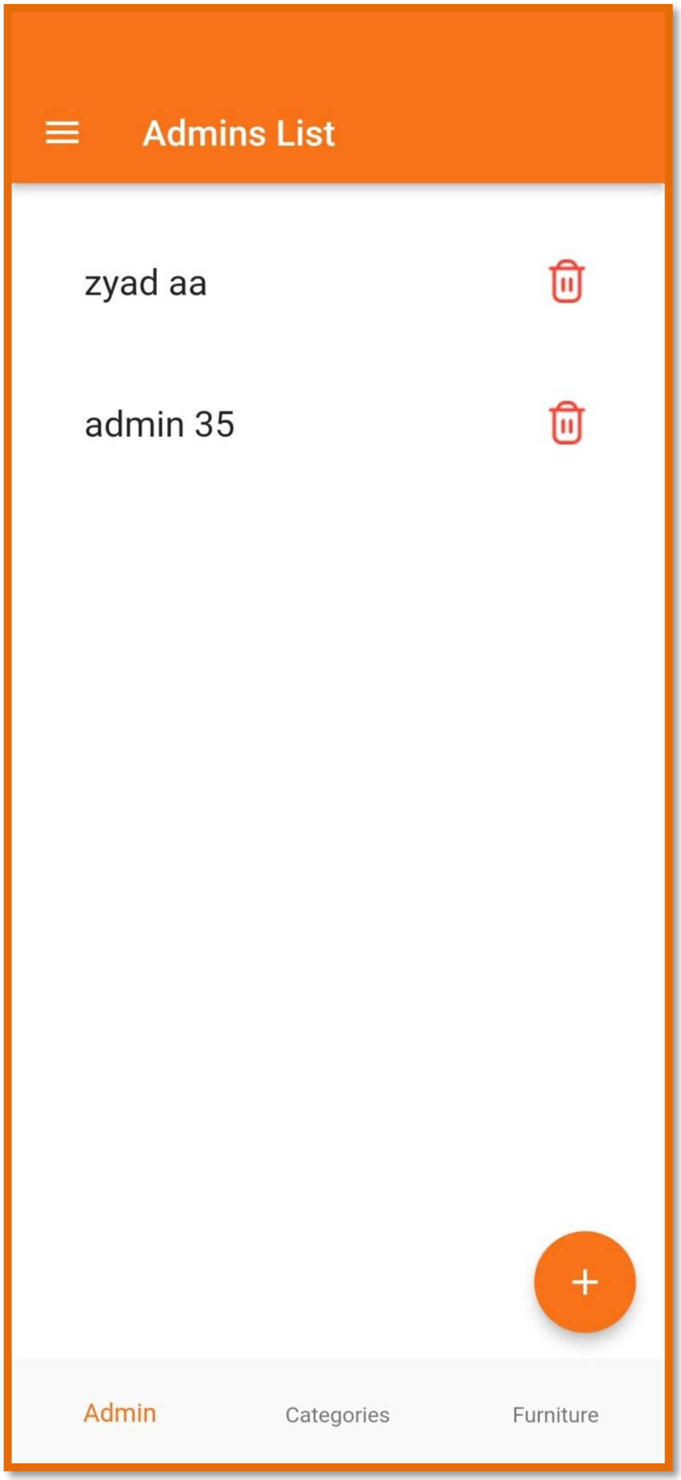
4.5.7. Admin side menu

This is the page that has the admin name, edit profile and logout.



4.5.8. Admin list

This is the page that views all admin in app.



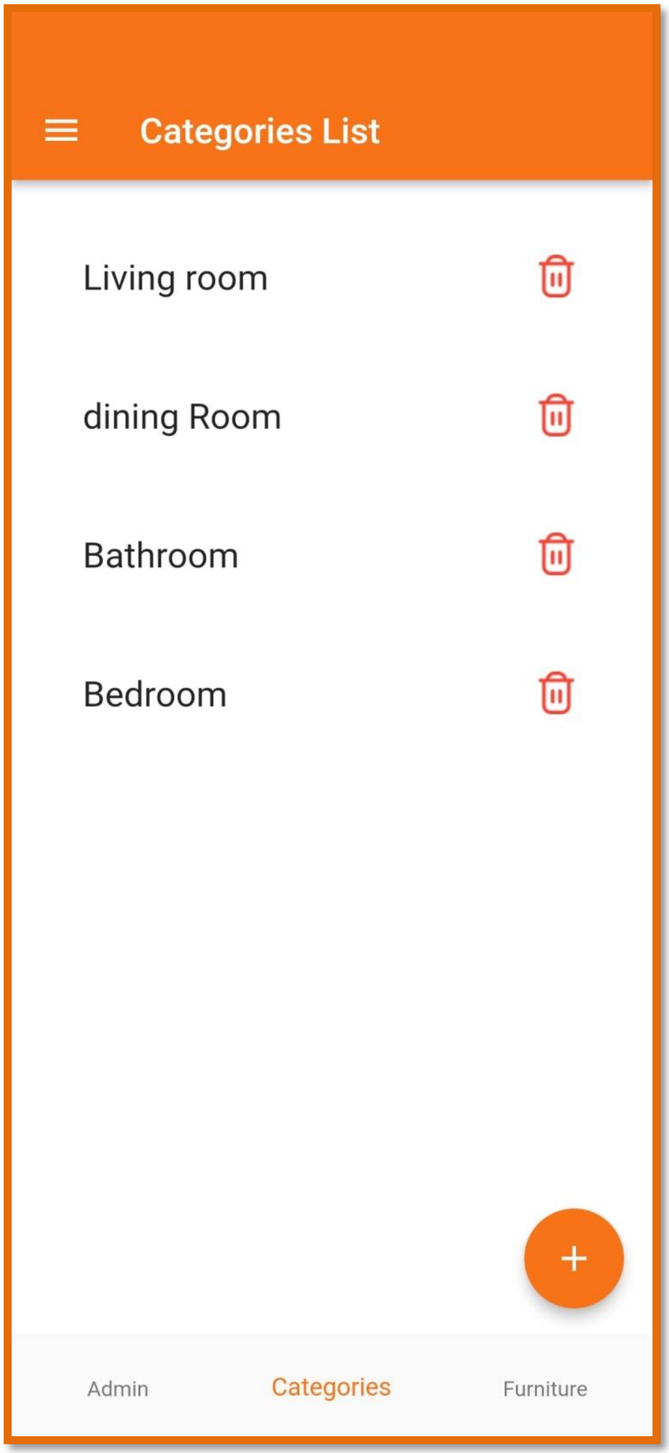
Admins List

zyad aa

admin 35

4.5.9. Category list

This is the page that admin can add new category.



Categories List

Living room

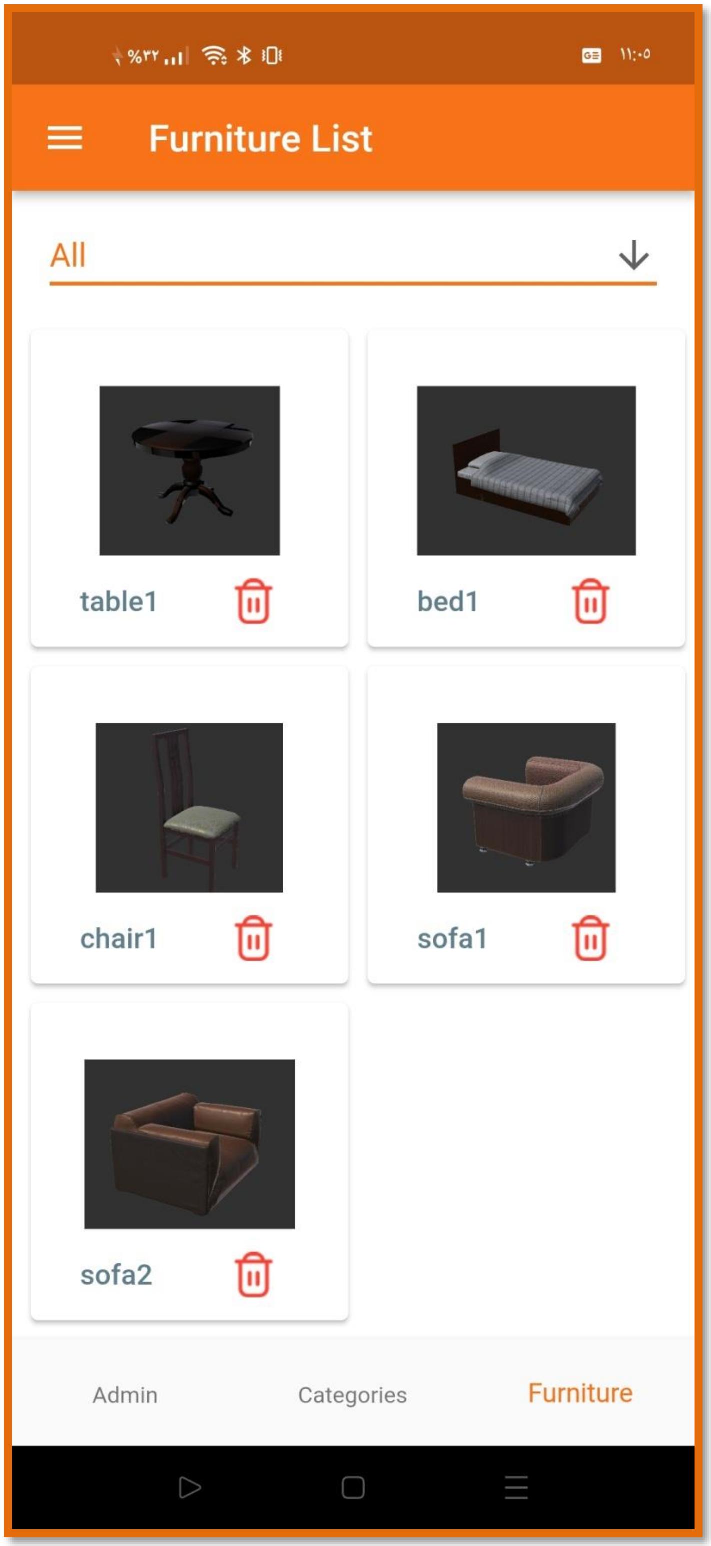
dining Room

Bathroom

Bedroom

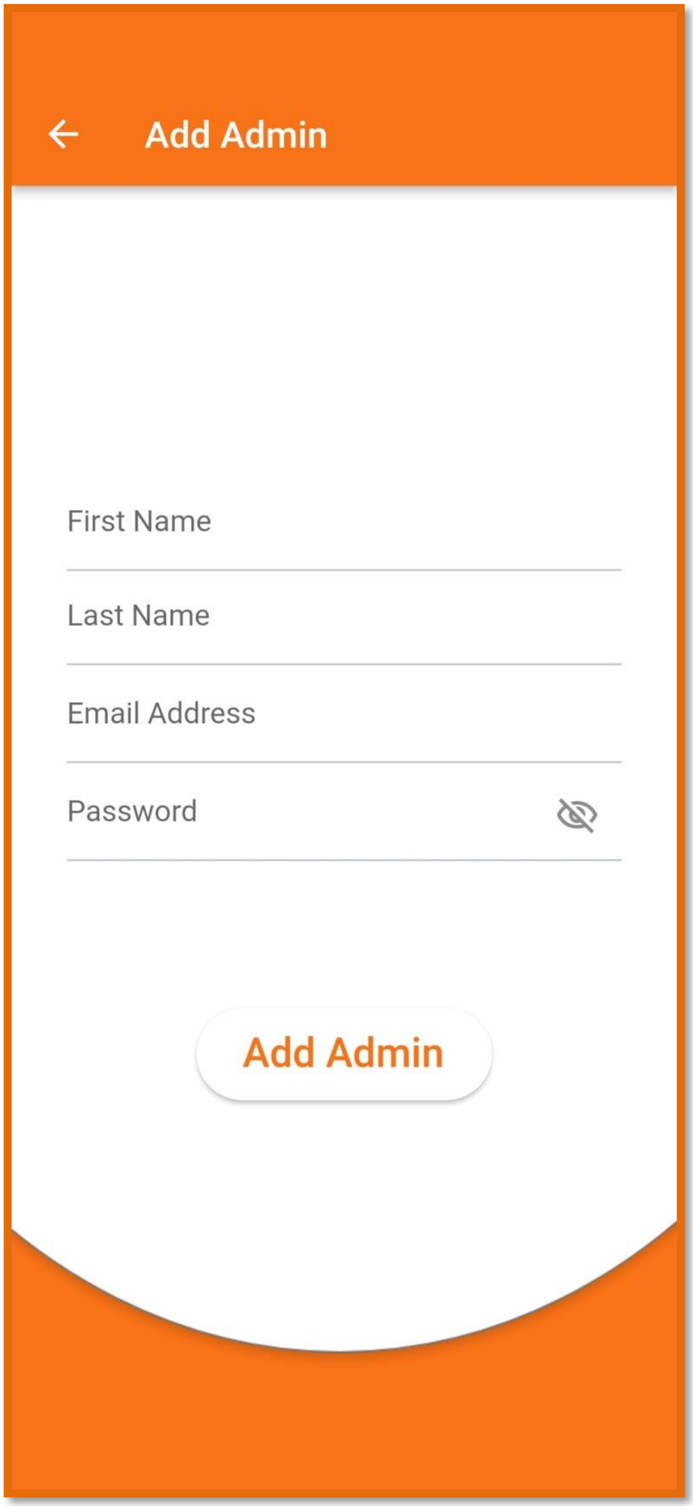
4.5.10. Furniture list

This is the page that admin can add new furniture to specific category.



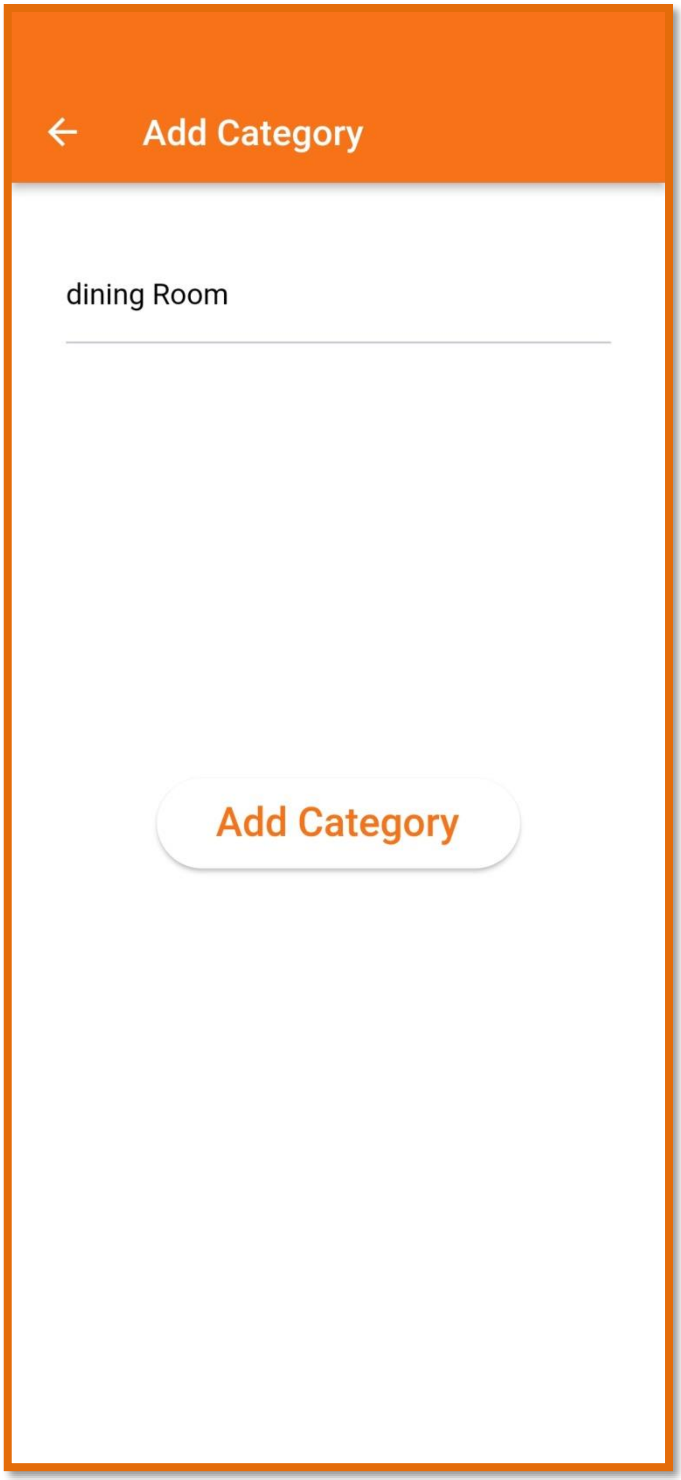
4.5.11. Add furniture

This is the page that admin can add new admin.



4.5.12. Add category

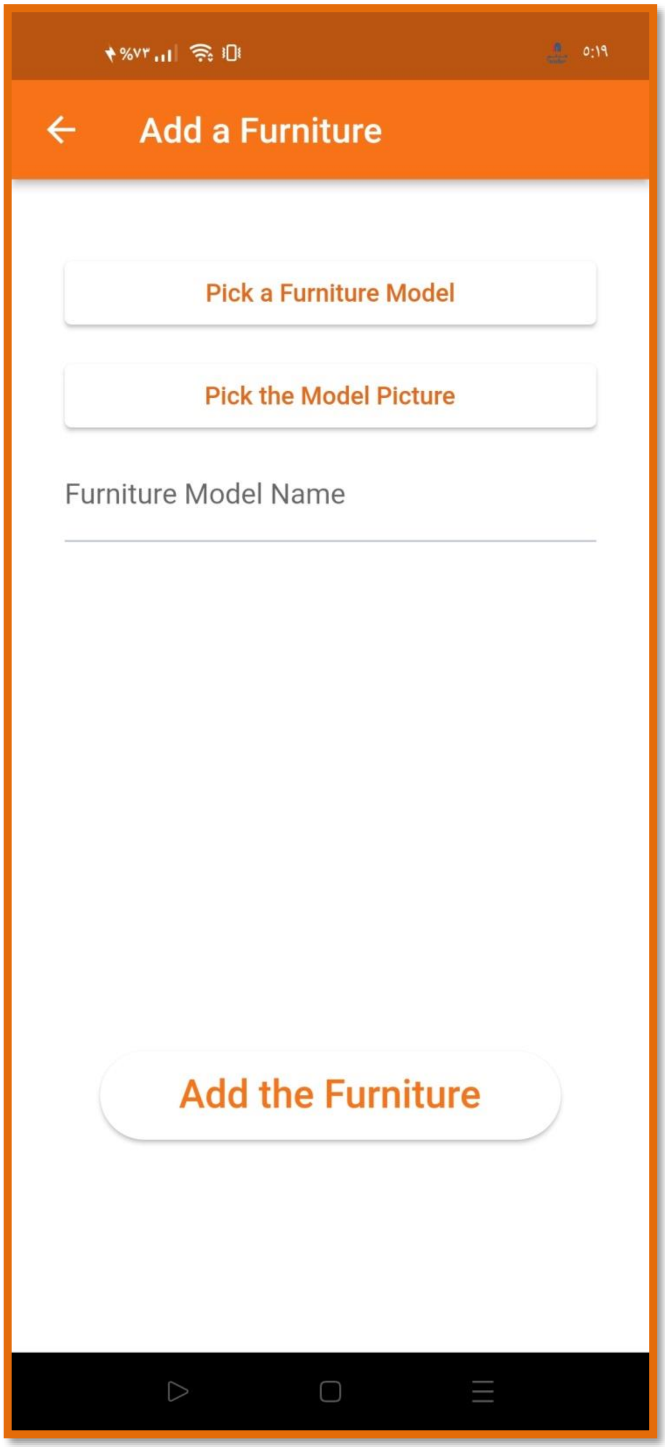
This is the page that views all categories in app.



dining Room

4.5.13. Add furniture

This is the page that views all furniture in app and admin can delete it.



Chapter 5 : System testing & Evaluation

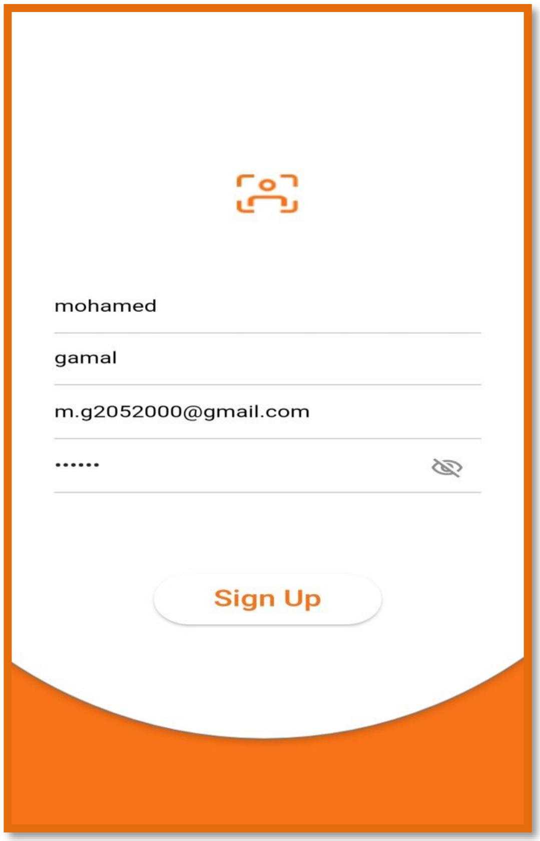
- System Testing:

Registration test cases:

Test case ID	**TC01**
Test scenario	Check Registration with valid email and password
Test steps	1. Open application 2. Click on Registration 3. Enter your email and password 4. Click sign up

Test Data	First Name=mohamed
	Last Name=gamal
	Email= m.g2052000@gmail.com
	password=000000

Expected Results	User Registration successfully and entry page is appeared to him
Actual Results	Entry page appeared and user Registration successfully
Pass/Fail	Pass



mohamed

gamal

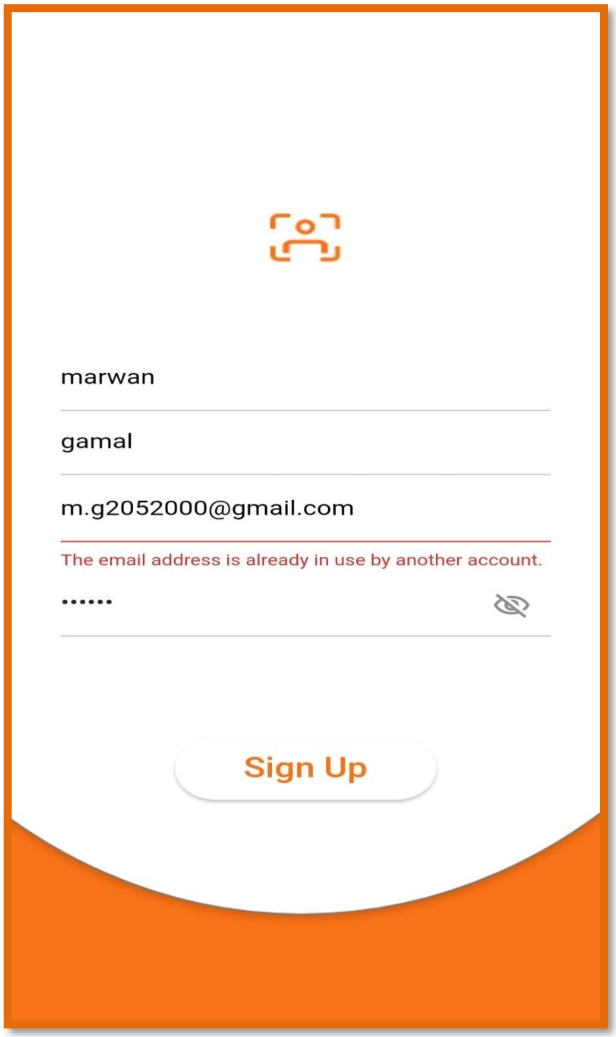
m.g2052000@gmail.com

••••••

Test case ID	**TC02**
Test scenario	Check Registration with valid email and password
Test steps	1. Open application 2. Click on Registration 3. Enter your used email and password 4. Click sign up

Test Data	First Name=marwan
	Last Name=gamal
	Email= m.g2052000@gmail.com
	password=000000

Expected Results	User Registration failed and send message the email is used.
Actual Results	Don't entry page appeared and user Registration failed and send message
Pass/Fail	Pass



marwan

gamal

m.g2052000@gmail.com

The email address is already in use by another account.

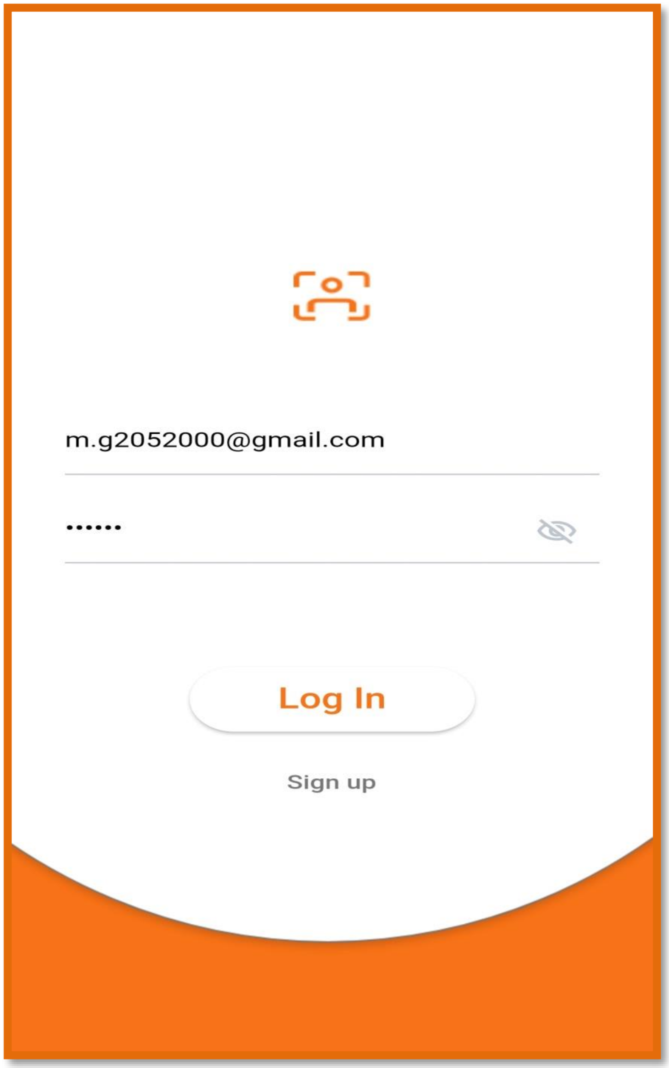
••••••

Login test case:

Test case ID	**TC03**
Test scenario	Check login with valid email and password
Test steps	1. Open application 2. Click on login 3. Enter your email and password 4. Click login

Test Data	Email= m.g2052000@gmail.com
	password=000000

Expected Results	User login successfully and entry page is appeared to him
Actual Results	Entry page appeared and user login successfully
Pass/Fail	Pass



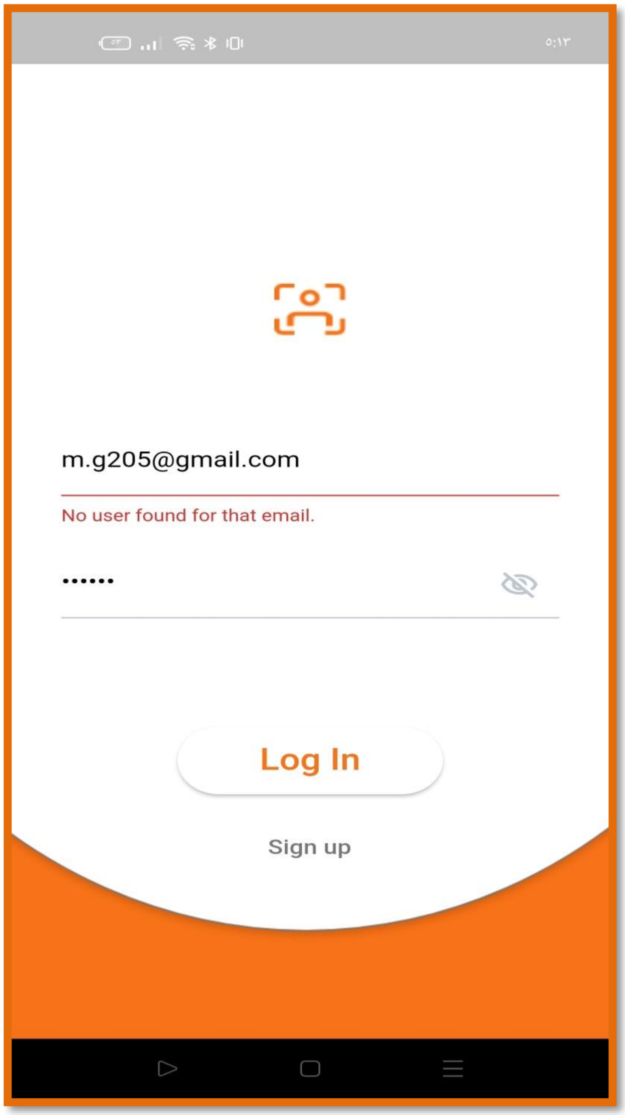
m.g2052000@gmail.com

••••••

Test case ID	**TC04**
Test scenario	Check login with invalid email and password
Test steps	5. Open application 6. Click on login 7. Enter your email and password 8. Click login

Test Data	Email= m.g205@gmail.com
	password=000000

Expected Results	User login fail and send message no user found for this email.
Actual Results	Login failed and send message no user found for this email.
Pass/Fail	Pass



m.g205@gmail.com

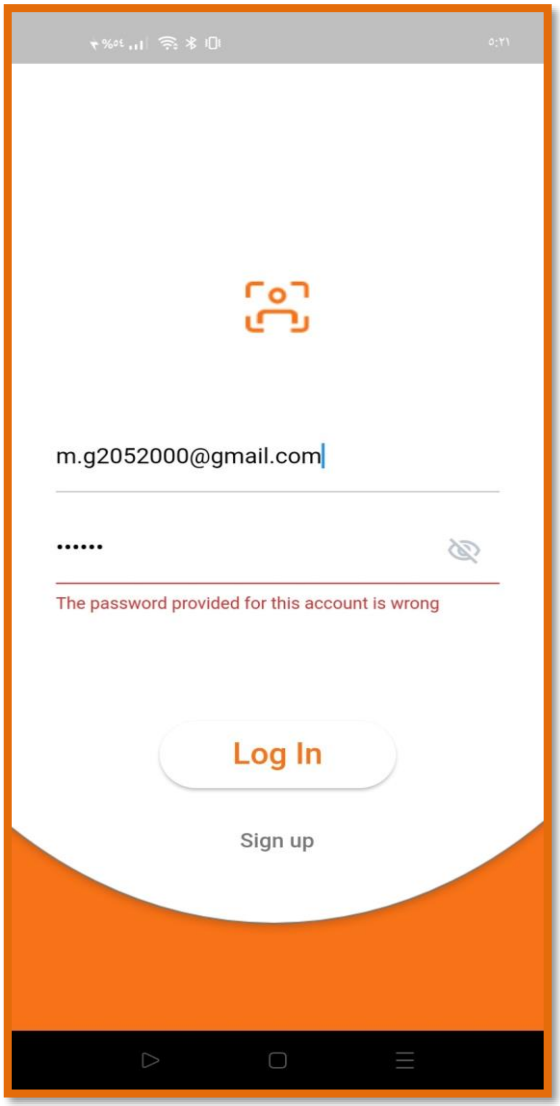
No user found for that email.

••••••

Test case ID	**TC05**
Test scenario	Check login with invalid email and password
Test steps	1. Open application 2. Click on login 3. Enter your email and password 4. Click login

Test Data	Email= m.g2052000@gmail.com
	password=111111

Expected Results	User login fail and send message password is wrong.
Actual Results	Login failed and send message no user found for this email.
Pass/Fail	Pass


m.g2052000@gmail.com

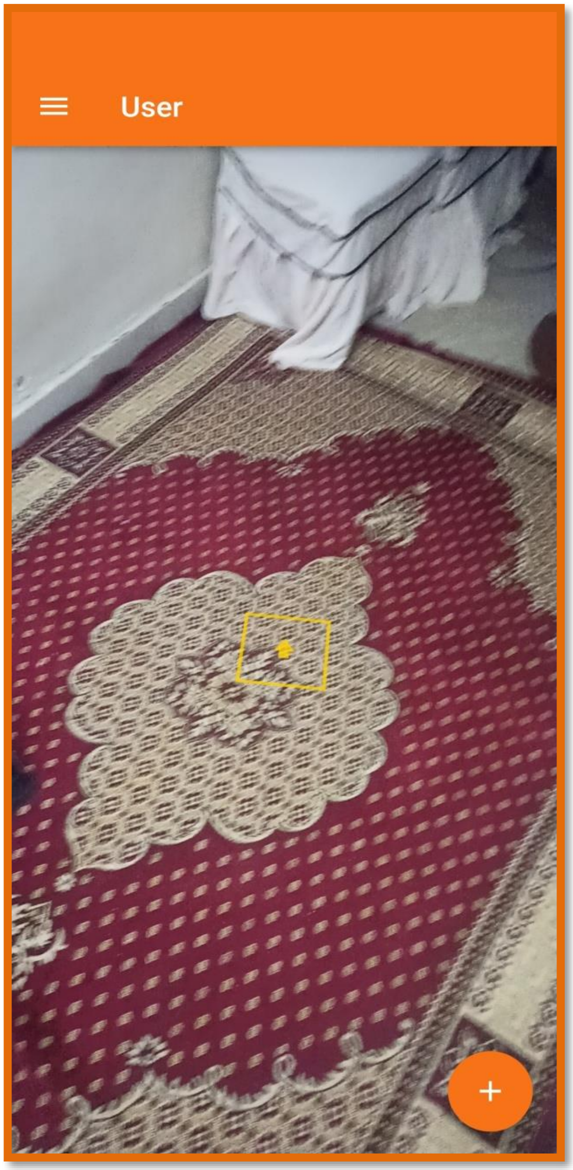
••••••

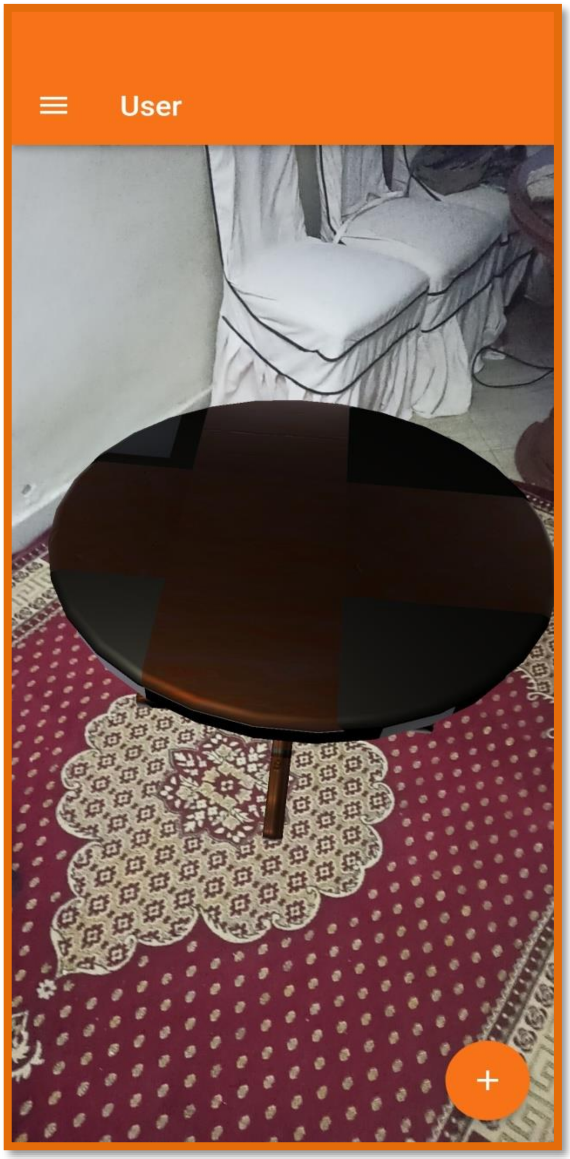
The password provided for this account is wrong

Adding furniture test cases:

Test case ID	**TC06**
Test scenario	Spawn furniture
Test steps	1. Open application. 2. Choose specific furniture and spawn it.
Test Data	Furniture = table
Expected Results	Furniture spawns in specific place.
Actual Results	Furniture spawned in specific place.
Pass/Fail	Pass

The place before add furniture	The place after added furniture

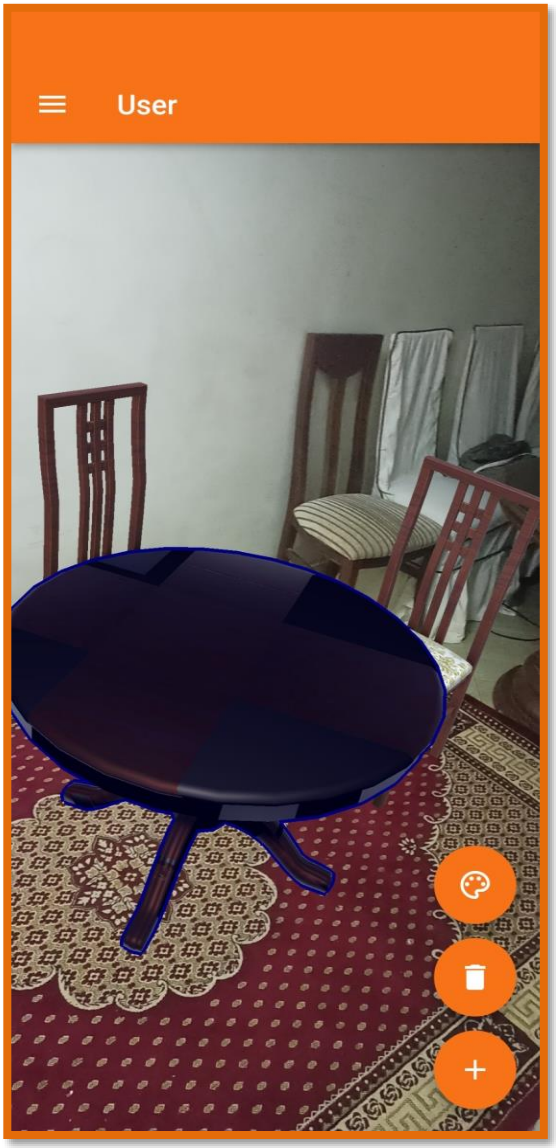




Test case ID	**TC07**
Test scenario	Spawn more furniture.
Test steps	1. Choose specific of spawned furniture and spawn it. 2. Try step 1 again.

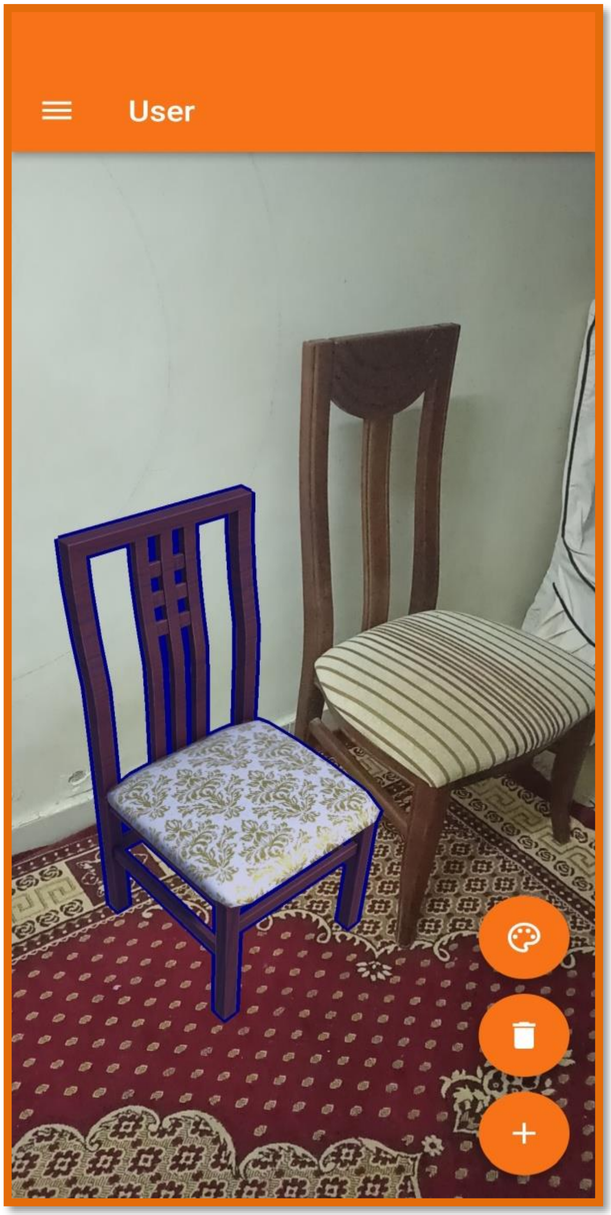
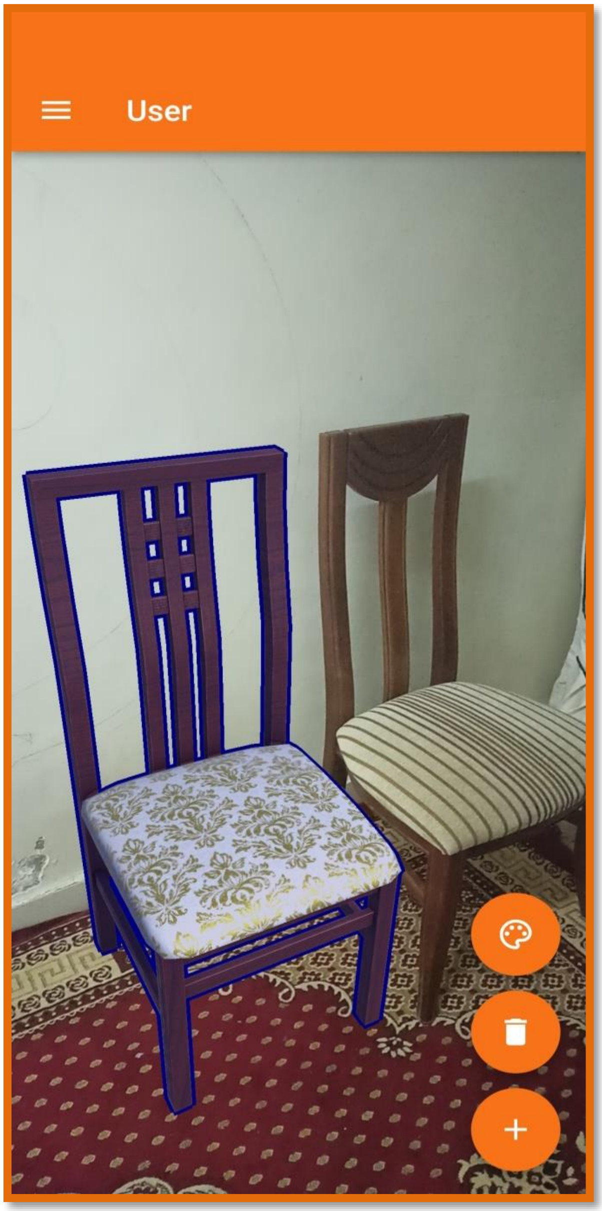
Test Data	furniture = table
	Furniture = chair
	Furniture = chair

Expected Results	Selected furniture spawned furniture in place.
Actual Results	Spawned furniture in specific place.
Pass/Fail	Pass



Resize spawned furniture test cases:

Test case ID	**TC08**
Test scenario	resize furniture
Test steps	1. Choose specific of spawned furniture and resize it by multi touching.
Test Data	furniture = chair
Expected Results	Selected spawned furniture in specific place can resize.
Actual Results	Selected spawned furniture is resized.
Pass/Fail	Pass

The select furniture before resize	**The select furniture after resize**
	

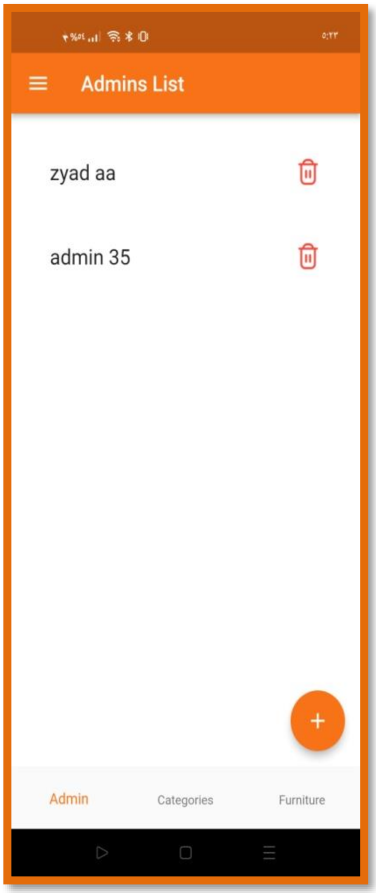
Adding system admin test cases:

Test case ID	**TC09**
Test scenario	Add admin
Test steps	1. Open application as a admin. 2. Admin data entry (name, email, password).

Test Data	First Name=mohamed
	Last Name=mansour
	Email= m.m@gmail.com
	password=000000

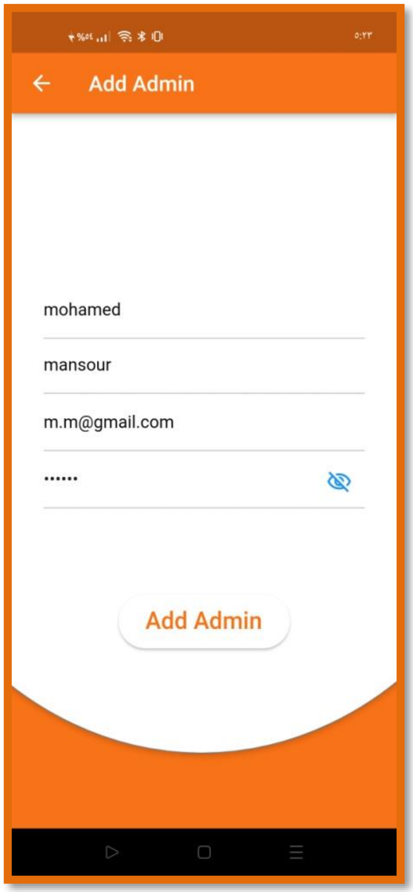
Expected Results	Admin add to admin list.
Actual Results	Admin added to admin list.
Pass/Fail	Pass

Admin list before add new admin	**Page of add admin and data**	**Admin list after added new admin**



zyad aa

admin 35

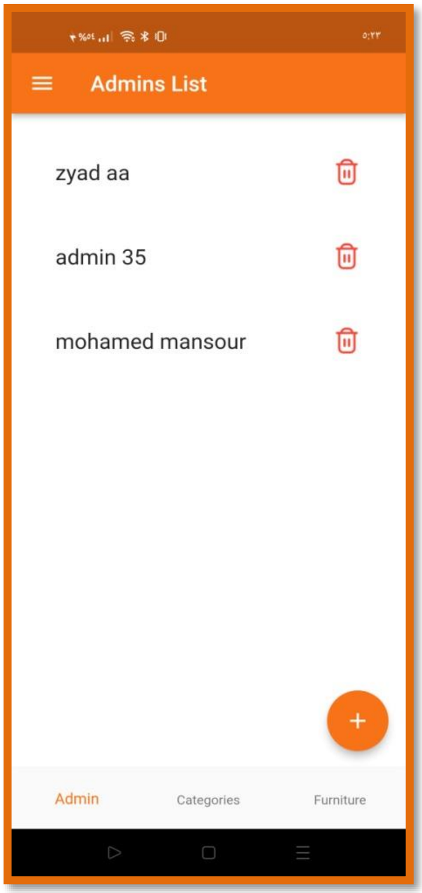


mohamed

mansour

m.m@gmail.com

••••••



zyad aa

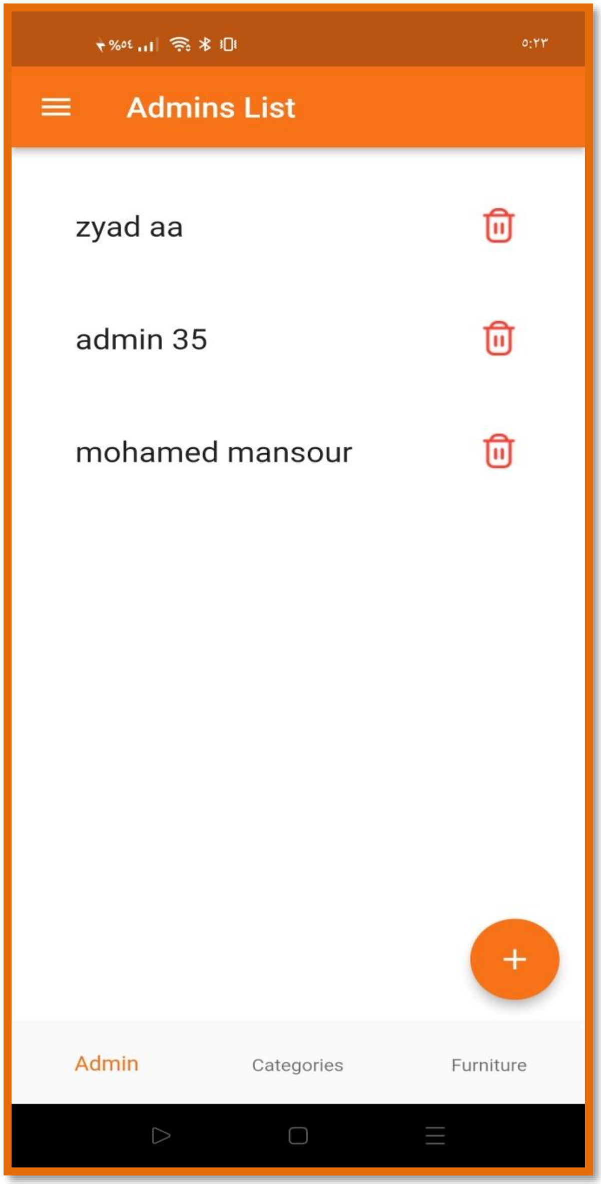
admin 35

mohamed mansour

Deleting system admin test cases:

Test case ID	**TC10**
Test scenario	Delete admin
Test steps	1. Open application as a admin. 2. Choose specific admin and delete it.
Test Data	admin = mohamed mansour
Expected Results	Chosen admin delete from category.
Actual Results	Admin was deleted.
Pass/Fail	Pass

Admin list before delete admin	**Admin list after deleted admin**

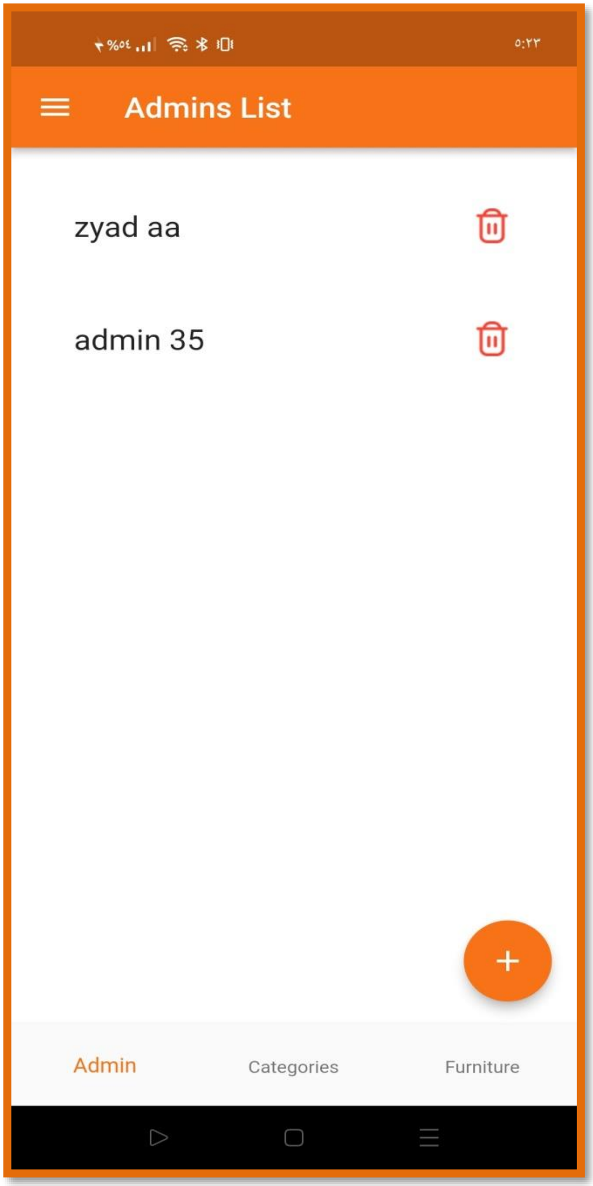


Admins List

zyad aa

admin 35

mohamed mansour



Admins List

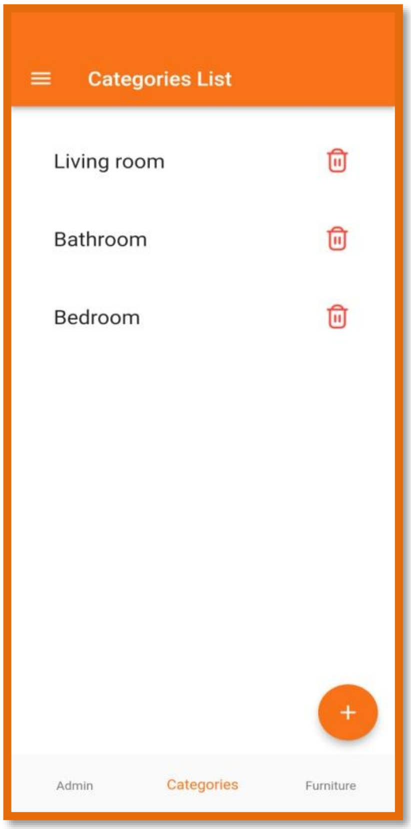
zyad aa

admin 35

Adding new furniture categories test cases:

Test case ID	**TC11**
Test scenario	Add category
Test steps	1. Open application as an admin. 2. Choose category and its name.
Test Data	Category = dining Room.
Expected Results	Chosen category adds to categories list.
Actual Results	Category was added to the categories list.
Pass/Fail	Pass

Category list before add new category	**Page of add category and data**	**Category list after added new category**

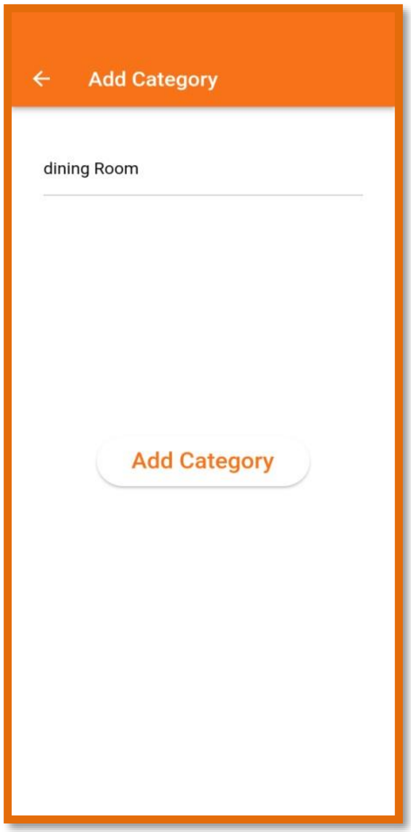


Categories List

Living room

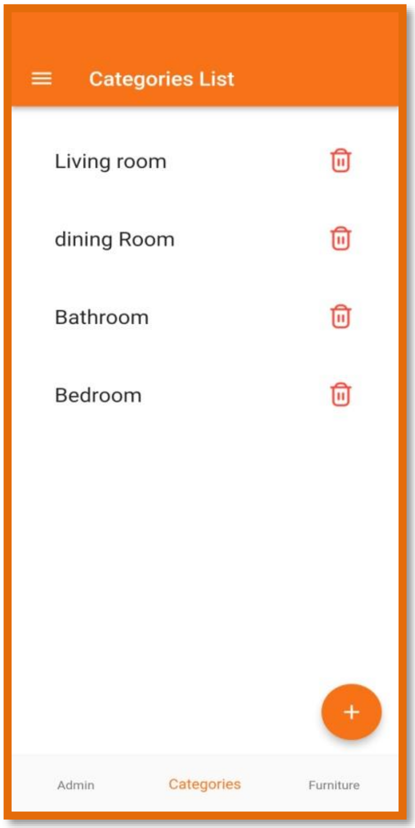
Bathroom

Bedroom



Add Category

dining Room



Categories List

Living room

dining Room

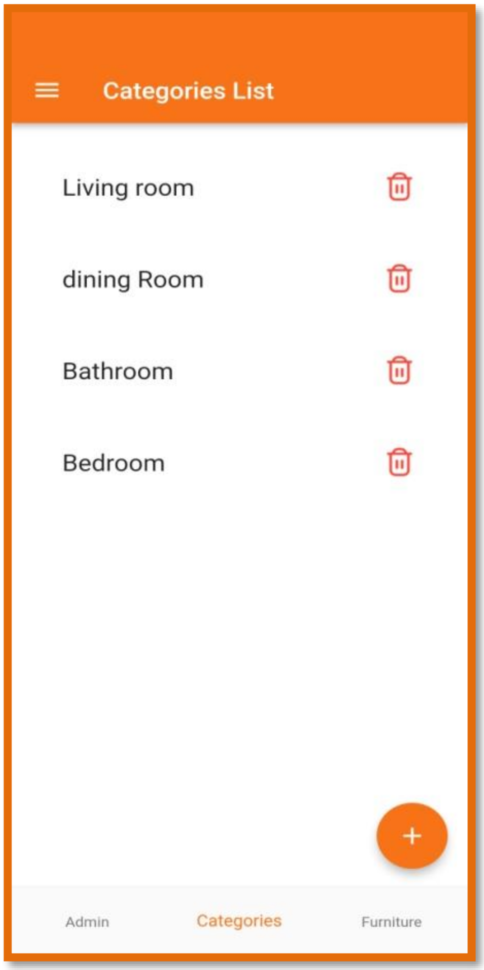
Bathroom

Bedroom

Deleting a furniture category test case:

Test case ID	**TC12**
Test scenario	Delete category
Test steps	1. Open application as an admin. 2. Choose specific furniture and delete it.
Test Data	Category = dining Room.
Expected Results	Chosen category delete from categories list.
Actual Results	Category was deleted.
Pass/Fail	Pass

Category list before add new category	**Category list after added new category**



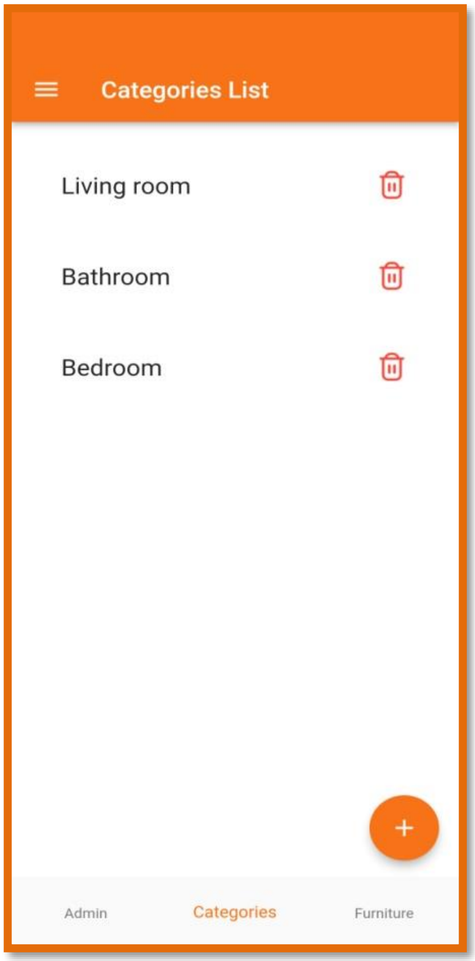
Categories List

Living room

dining Room

Bathroom

Bedroom



Categories List

Living room

Bathroom

Bedroom

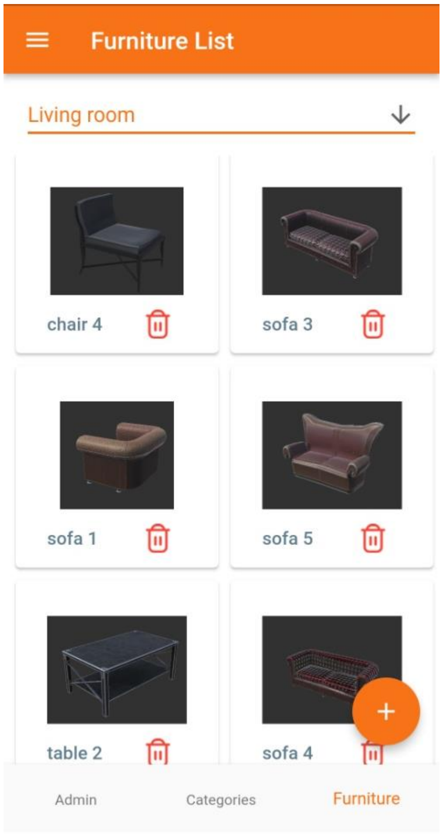
Uploading furniture test cases:

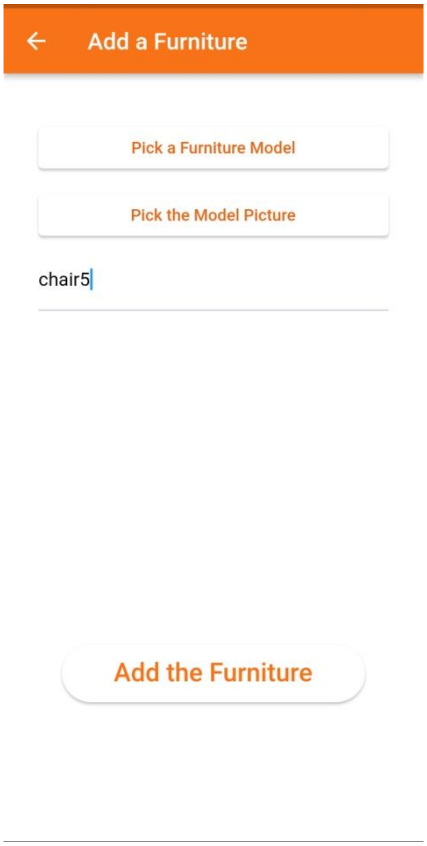
Test case ID	**TC13**
Test scenario	Add furniture to category
Test steps	1. Open application as a admin. 2. Choose specific furniture, its image, category and name.

Test Data	Furniture = chair5
	Category =living room

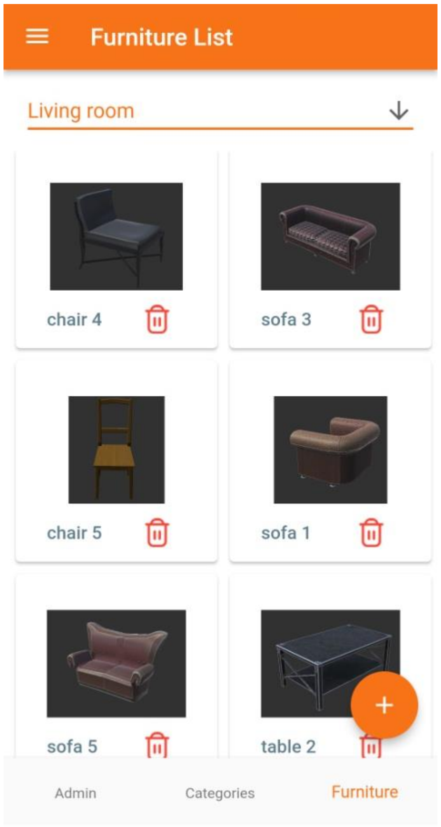
Expected Results	Furniture adds to category.
Actual Results	Furniture added to category.
Pass/Fail	Pass

Furniture of living room category before add new furniture	**New urniture data**	**Furniture of living room category after add new furniture**





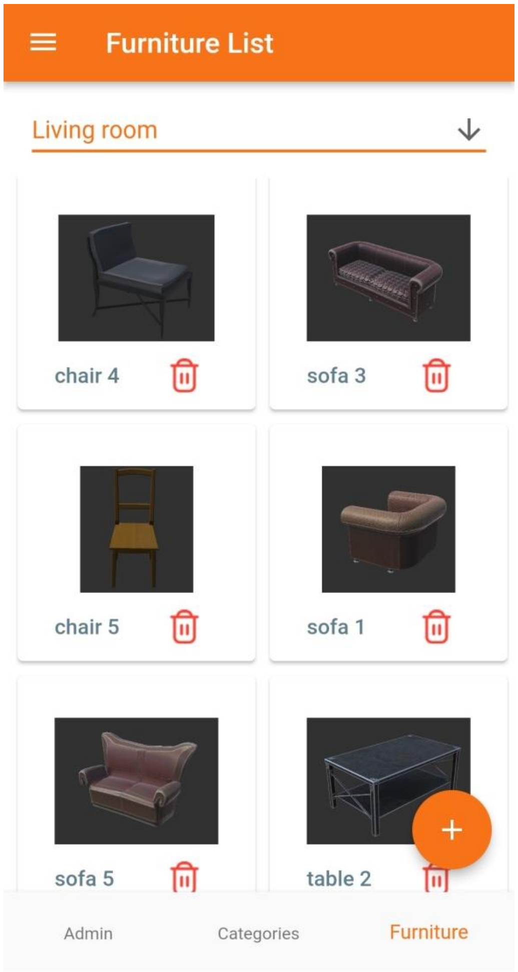
chair5

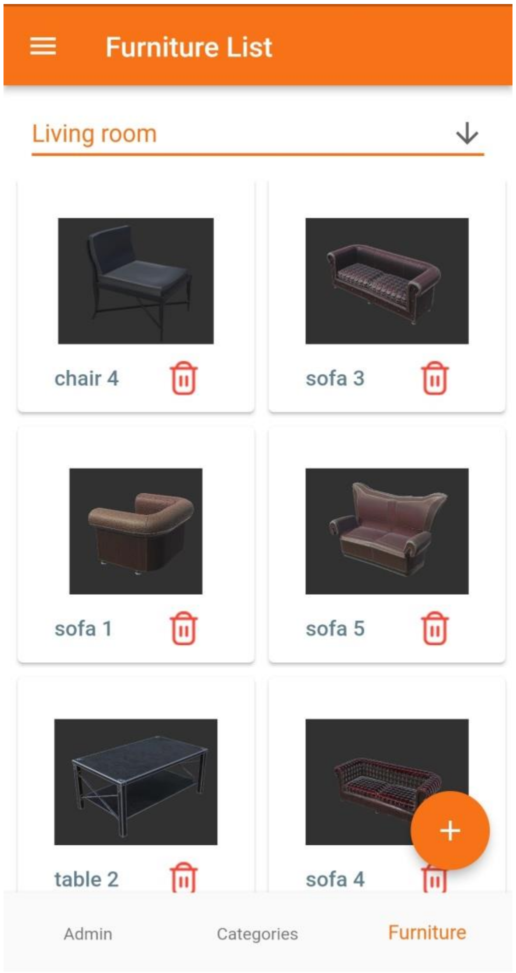


Deleting furniture from database test cases:

Test case ID	**TC14**
Test scenario	Delete furniture
Test steps	1. Open application as an admin. 2. Choose specific furniture and delete it.
Test Data	Furniture = chair5
Expected Results	Chosen furniture delete from category.
Actual Results	Chair5 was deleted.
Pass/Fail	Pass

Furniture of living room category before delete furniture	**Furniture of living room category after add new furniture**

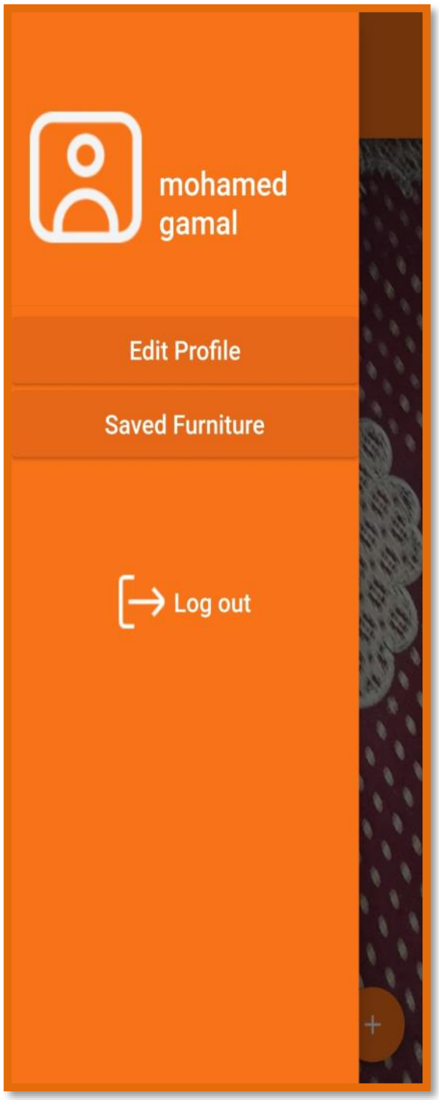


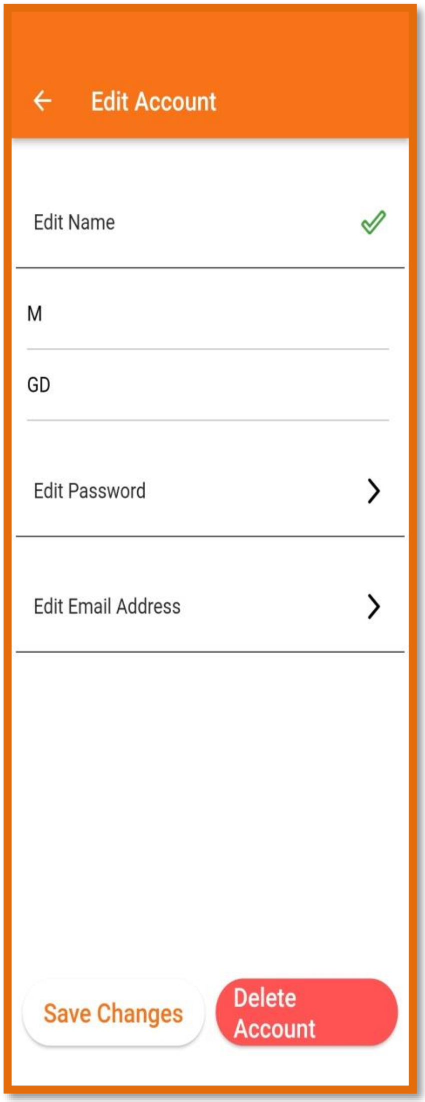


Editing user profile information test cases

Test case ID	**TC15**
Test scenario	update profile
Test steps	1. Open application. 2. Update the data.
Test Data	First name = M Last name = GD
Expected Results	The data updated correctly to (M, GD).
Actual Results	The data updated to (M, GD).
Pass/Fail	Pass

Profile before update	**Profile data updated**	**Profile after updated**

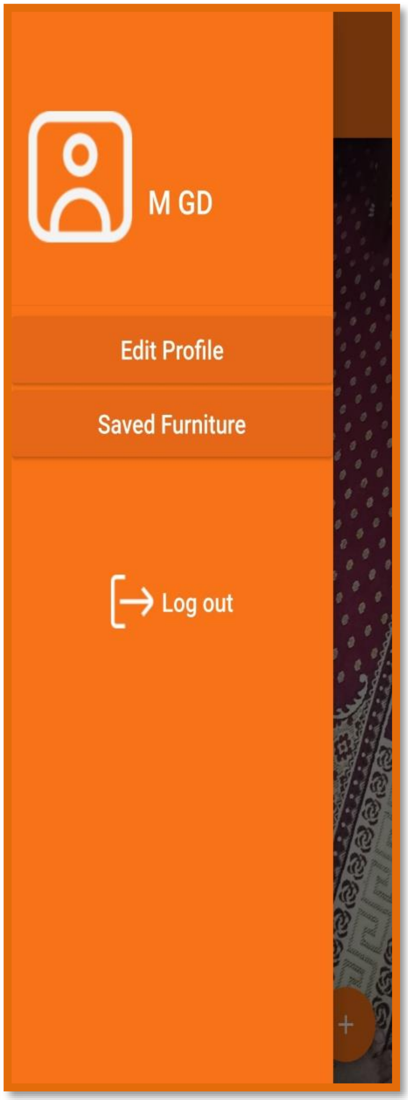

mohamed gamal


Edit Account

Edit Name ✓

M

GD

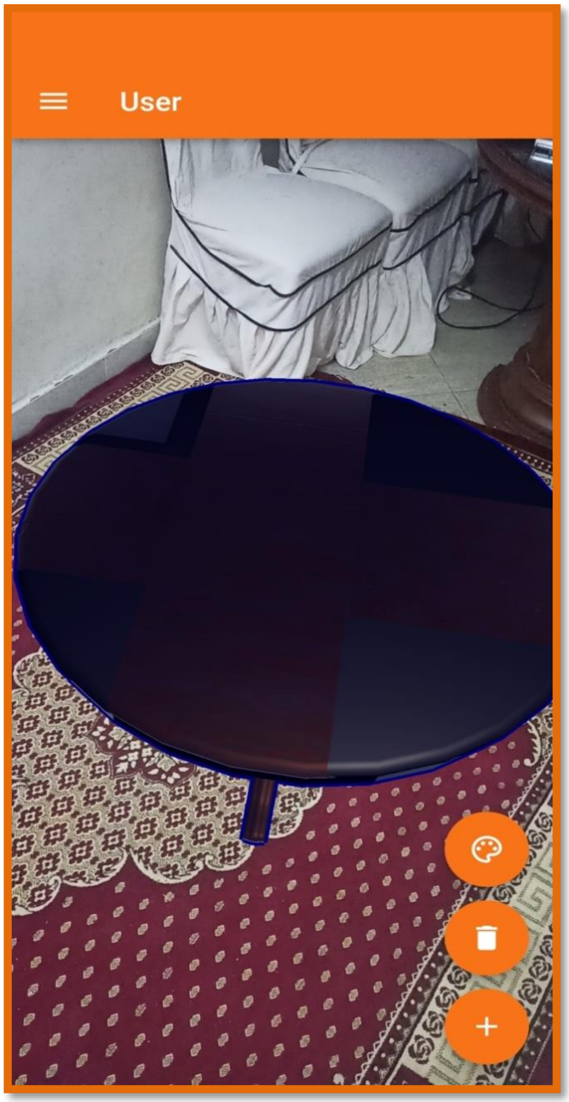

M GD

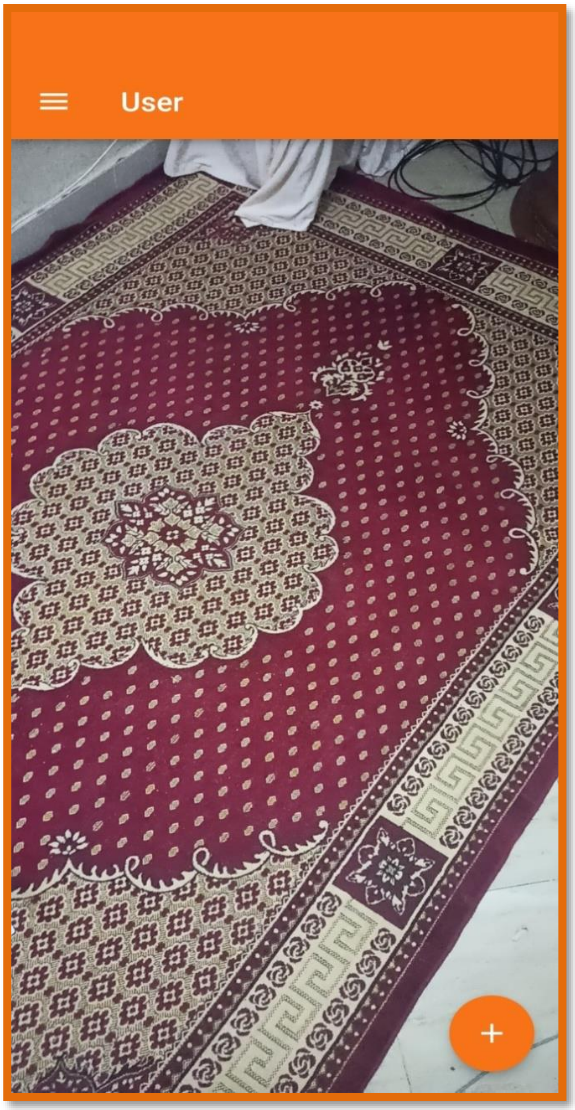
Integration Testing:

Deleting furniture test cases:

Test case ID	TC16
Test scenario	Delete spawned furniture
Test steps	1. Select specific of spawned furniture and delete it.
Test Data	Furniture = table
Expected Results	Spawned furniture in specific place is deleted.
Actual Results	Spawned table is deleted.
Pass/Fail	Pass

The place before delete furniture	The place after deleted furniture

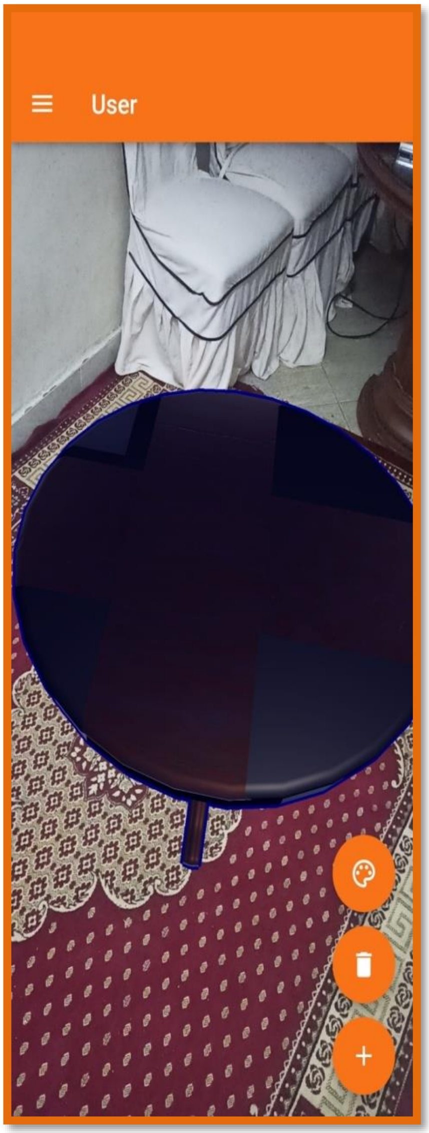


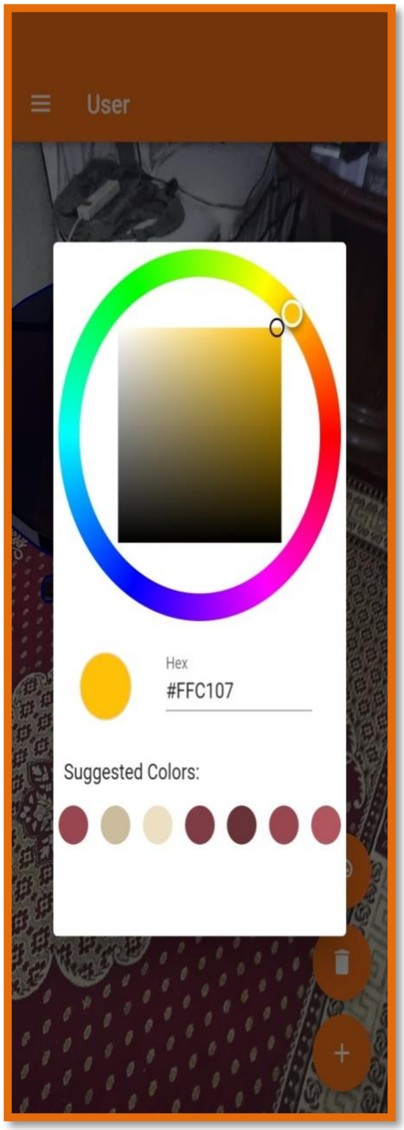


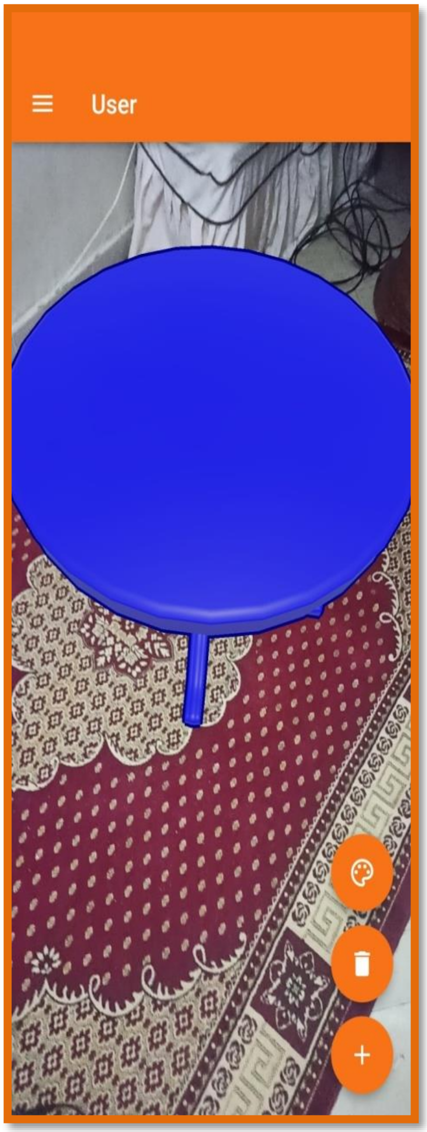
Coloring furniture test cases:

Test case ID	**TC17**
Test scenario	Recolor furniture
Test steps	1. Choose specific of spawned furniture and delete it.
Test Data	color = blue
Expected Results	Spawned furniture in specific place is recolor.
Actual Results	Spawned furniture is recolor.
Pass/Fail	Pass

The select furniture before coloring	**Color picker and suggestion color.**	**The select furniture after recolor**







Chapter 6 : Future work

- Add more models and categories.
- Changing wall paint and floors.
- Could integrate with real furniture providers like Ikea.

Chapter 7 : Conclusion

We divided the work into three parts (AR (Unity), Backend (firebase) and UI (flutter)).

In UI (Flutter) after designing all the application we redesigned the application with new UI.

We faced many problems in AR (Unity):

- The Documentation of unity wasn't clear with no examples and not enough resources to apply what we wanted to apply the way we wanted.
- Not all the team members had Mobile phones that support AR.
- Not enough good free models.

After finishing Unity and Flutter we had problem with integration between them:

- The package of unity_flutter_widget had internal problems with no explanations of what went wrong and what is the root of the problem.
- First time we merged them the application was crashing with no clear reason.
- We spend a lot of time trying to fix these problems.
- After integration we faced problem with the callback from unity to flutter.

After solving all the problems we upload all models and divided them into categories then we test all application's features.