Software Requirement Specification Document for The XY Designer

Mohamed Salah Abdelhaleem[1], Yonos Tarek Ali[2],
Youssef Alaaudin Abdelfattah[3], Yousef Hussien Salem[4]
Supervised by: Dr. Khalid Hussein, ENG Noha Elmasry

June 16, 2023

Table 1: Document version history

Version	**Date**	**Reason for Change**
1.0	13-Dec-2022	SRS First version's specifications are defined.
1.1	14-Dec-2022	Finishing The SRS
2.0	6-Mar-2023	SRS Ver 2
3.0	30-April-2023	SRS Ver 3

GitHub: https://github.com/Mohamed26Salah/The-XY-Designer

Contents

Abstract

Introducing an innovative interior design app that enables users to create and customize their room's interior design in a 3D environment. The app allows users to scan their room and convert it into a 3D model with accurate dimensions and furniture placement. The user can apply dominant colors and textures to walls, windows, and furniture, and experiment with different furniture objects and room shapes. The app offers an AR feature that allows users to view their design in real-time and in 3D space. Additionally, the app uses a reinforcement learning model to optimize furniture placement based on user preferences and room dimensions. The app offers an interactive and user-friendly interface and provides a comprehensive solution for interior design, making the process accessible and enjoyable for users, while also providing a valuable tool for professionals in the industry.

1 Introduction

1.1 Purpose of this document

The purpose of this document is to provide a brief overview of an interior design app. The document highlights the app's key features, including its ability to scan rooms, convert them into 3D models, and allow users to customize their interior design with furniture, colors, and textures. Additionally, the document outlines the app's AR functionality and its use of a reinforcement learning model to optimize furniture placement. The document aims to communicate the value and potential of the app and serve as a record of the project's development.

1.2 Scope of this document

The scope of this document is to provide a comprehensive understanding of the features and functionality of the interior design app. The document covers the app's capabilities, including its ability to scan and convert rooms into 3D models, customize furniture, colors, and textures, and optimize furniture placement using a reinforcement learning model. The document also briefly discusses the app's AR functionality. The document is intended for anyone interested in learning about the app's capabilities, including potential users and industry professionals. The scope of the document is limited to describing the app's features and does not cover technical details or implementation specifics. the benefit of this app is manifested in making the process of interior design easier for both designers and homeowners.

1.3 Business Context

The interior design app provides an innovative solution to the industry, allowing users to scan and convert rooms into 3D models, customize furniture, colors, and textures, experiment with different room configurations, and optimize furniture placement using a reinforcement learning model. This app offers significant benefits to the market, reducing the need for costly professional services and providing professionals with valuable tool to enhance their services. As the first app to offer such a comprehensive solution, it has the potential to revolutionize the industry by setting a new standard for interior design. Overall, the app has vast market potential and can be marketed across a wide range of platforms and industries.

2 Similar Systems

2.1 Academic

This study [1] attempts to develop a memory-efficient learning-based method for 3D reconstruction from single photos. With minimal memory footprint, the continuous decision boundary representation can store an infinitely detailed description of the 3D output. They demonstrate that their model effectively encodes 3D structure and can be inferred from a variety of inputs. Occupancy Networks, a type of neural network that depicts 3D geometry, accept a pair of inputs (p, x) and output a real value that denotes the probability of occupancy. In terms of intersection over union (IoU), chamfer-L1 distance, and normal consistency, the findings of single image 3D reconstruction in this work demonstrate that it performs better than state-of-the-art approaches by 0.571, 0.215, and 0.834 correspondingly.

This paper [2] tries to avoid problems in layout optimization like slow processing speed, long response time, taking too much memory and the lack of sufficient examples of indoor furniture arrangements for training a related learning framework. It's also important to mention that the problem of furniture optimization is relatively subjective. therefore, this paper focuses on achieving common arrangement results and making a fairly fast framework. Furthermore, to find a dataset for this problem is hard, so this paper didn't use any datasets to achieve high design results. This paper focuses on enhancing the interactive-ness of virtual scenarios using automatic furniture layout with web technologies to achieve high feasibility. The method used for furniture layout is Q-learning which is type of reinforcement learning. Experimental results show that their approach allows a reasonable, conventional layout design. Moreover, the design results are consistent with popular layout design principles. However, their approach doesn't guarantee the best design results, so the Asynchronous Advantage Actor-Critic (A3C) method will be considered to further to improve the performance and obtain better results.

This study [3] offers a convolutional neural network-based method for creating interior scenes. By describing 3D situations with a semantically-enriched image-based representation based on orthographic top-down views, they are able to learn convolutional object placement priors from the complete context of a room. Their approach continually builds rooms from scratch using only the room architecture as an input. Through a series of perceptual investigations, they evaluate the believability of scenes produced by their system to baselines for item selection and object arrangement, as well as situations modelled by people. However, interior scenes display hierarchy since 'functional groups' are formed by sets of objects (e.g. bed and nightstands, table and chairs). These groups also frequently display symmetries, such as the symmetrical placement of chairs around a table. Morevery, the dataset consists of approximately 45,000 3D scenes, each of that has been divided into distinct rooms and assigned a room type. They take into account the three most common sorts of rooms: bedrooms, living rooms, and offices. Scenes created using the paper's method are vastly superior to those created by sampling a collection of objects and then arranging them according to the paper's model for all room types (Occurrence Baseline). Our approach is also significantly superior to a model based on pairwise object relationship statistics when organising a fixed collection of objects (Arrangement Baseline). Our generated scenes are slightly less preferred for bedroom and living room scenarios compared to human-made scenes from our test dataset, but they are similarly preferred for workplace scenes.

2.2 Business Applications

2.3 Polycam

Polycam is an application that is used Create high-quality 3D models from photos with any Android device. View your 3D captures directly on device, and export them in over a dozen file formats. Share your captures with other people and the Polycam community with Polycam Web and explore captures from around the globe.

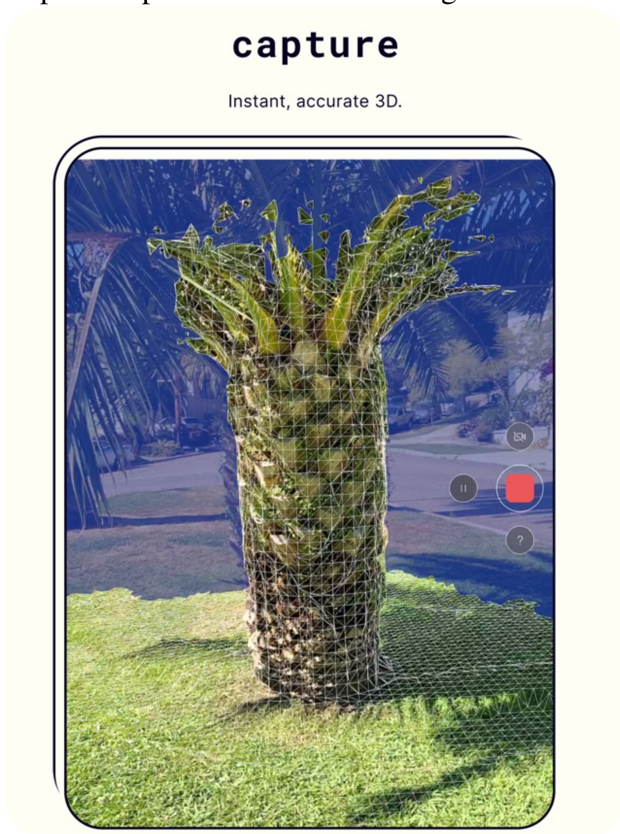


Figure 1: PolyCam

2.4 Ikea Place

Ikea Place is an AR app developed by Ikea that allows users to virtually place furniture and decor into their own homes. Using the camera on their device, the app creates a 3D model of the user's room and allows them to place virtual furniture in real-time. The app offers a vast selection of Ikea products that users can place and view in their room from different angles and perspectives. Users can customize the product's size, color, and other attributes to see how it fits into their space. The app uses accurate measurements and lighting to ensure that the virtual furniture appears as it would in real life. Overall, Ikea Place offers a unique and convenient way for users to preview furniture in their own homes before making a purchase.

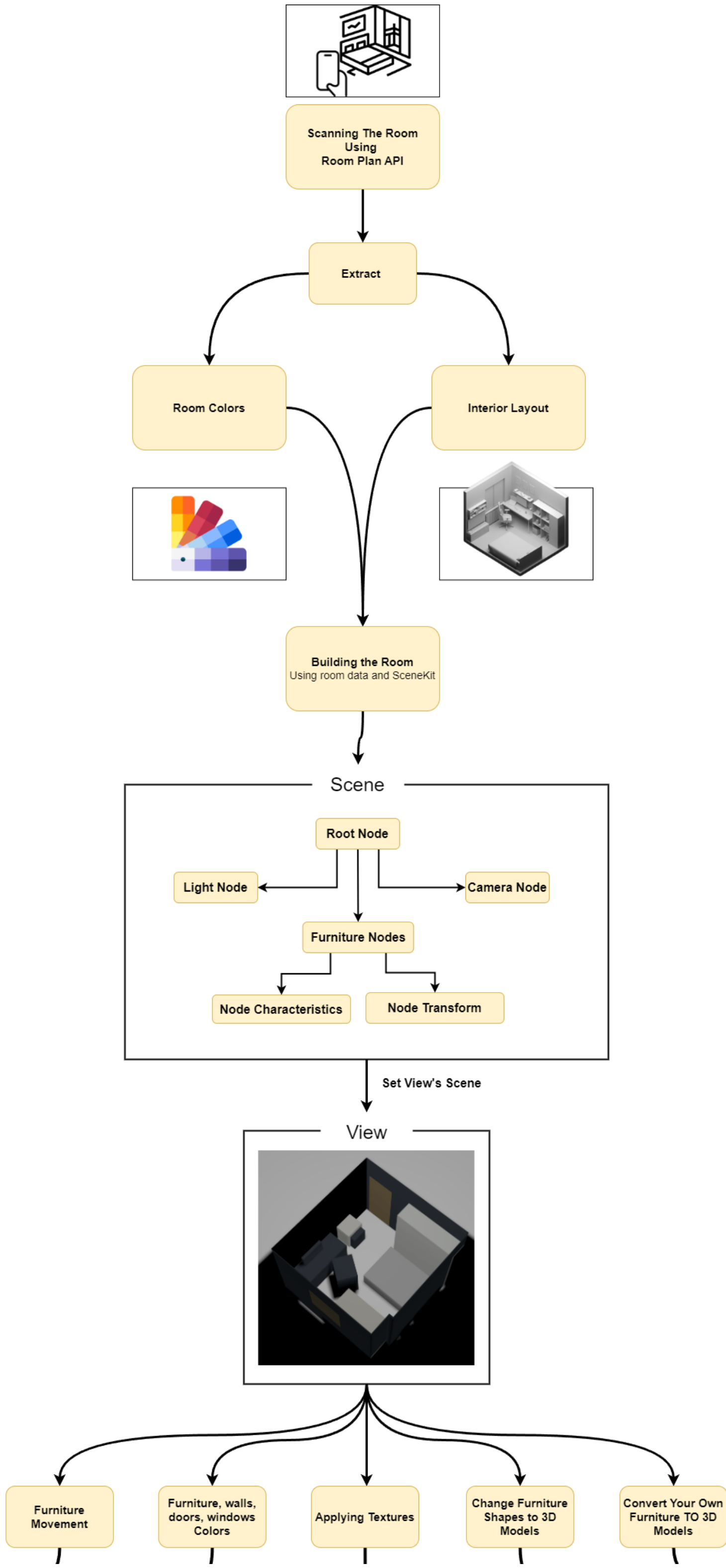
Figure 2: Ikea Place

3 System Description

3.1 Problem Statement

The main problem is assisting non-professional users in interior design. The concurrent problem of interior design is a visualization of furniture in a room, This Project will help ease this process through these steps, a quick solution for reorganizing the layout of a room with a single click, the ability to turn scans of rooms to 3D room objects and placing furniture in them. And the automatic optimization of furniture layout could be used as a reference. another issue is the time and effort required to physically move furniture around in a room to test different layouts. This can be particularly cumbersome when dealing with large or heavy items. Finally, the current market for interior design tools is heavily geared towards professional designers, with limited options available for non-professional users. The proposed solution aims to fill this gap by providing an intuitive and user-friendly interface that is accessible to individuals without any prior design experience.

3.2 System Overview



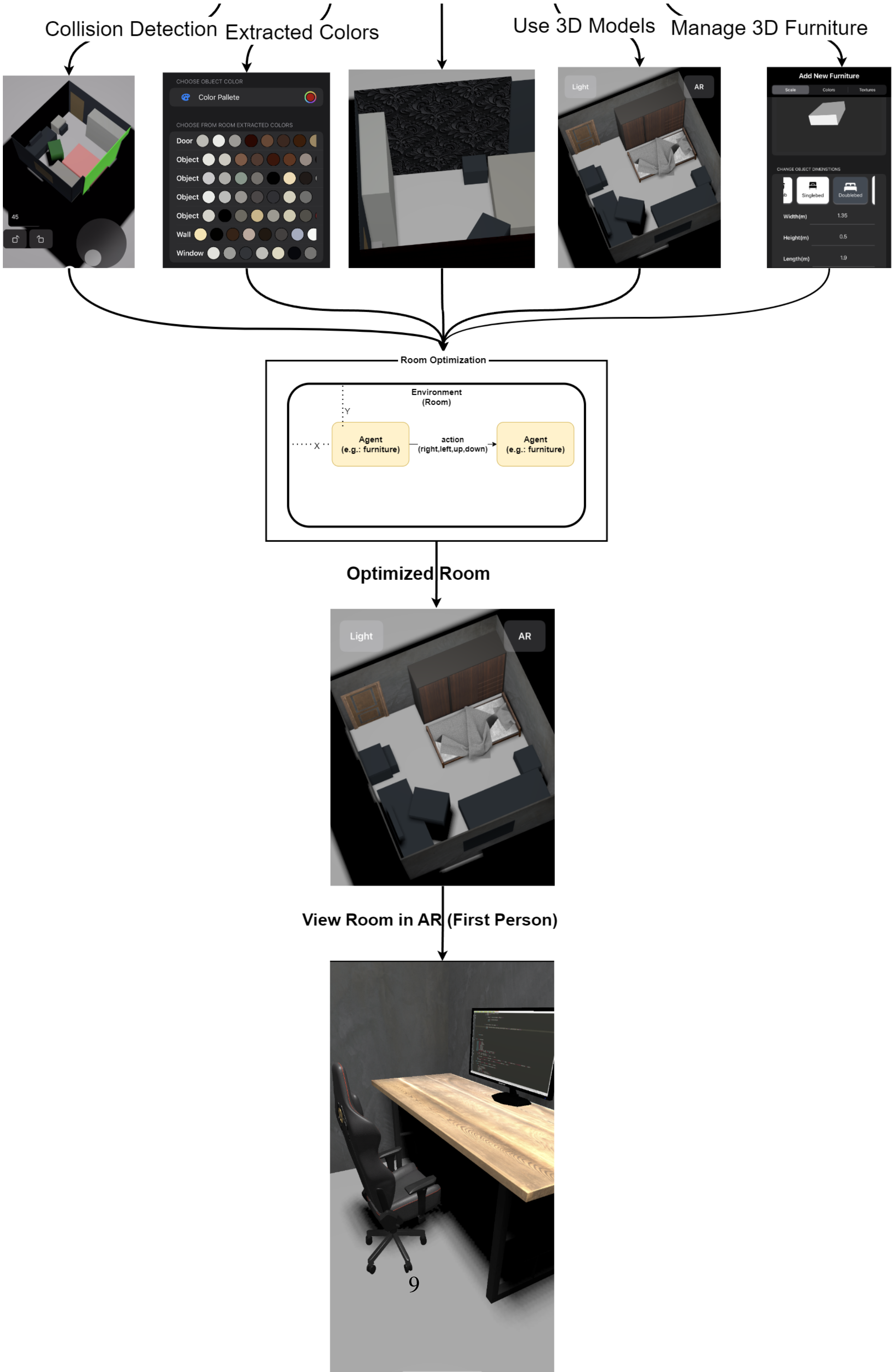

Collision Detection
Extracted Colors
Use 3D Models
Manage 3D Furniture
45
CHOOSE OBJECT COLOR
Color Pallete
CHOOSE FROM ROOM EXTRACTED COLORS
Door
Object
Object
Object
Object
Wall
Window
Light
AR
Add New Furniture
Scale
Colors
Textures
CHANGE OBJECT DIMENSIONS
Singlebed
Doublebed
Width(m) 1.35
Height(m) 0.5
Length(m) 1.9
Room Optimization
Environment
(Room)
Y
X
Agent
(e.g.: furniture)
action
(right,left,up,down)
Agent
(e.g.: furniture)
Optimized Room
Light
AR
View Room in AR (First Person)

Figure 3: System Overview

Our system will be a 3D interior design assistant, giving a new way for users and designers to design their rooms, giving them new ideas, and easing the whole process. Our system will consist of four main parts which are:

3.2.1 3D Room

- First part is converting the room of the user to a 3D model that he can use to put his converted 3D furniture inside it. This is achieved by doing two things:
- To begin, we would utilize Apple Room Plan API to convert the user's room into a 3D model. This API **RoomPlanApi** offers crucial room information such as object detection, object classification, and dimension extraction, which are necessary for room optimization. However, it has some limitations when it comes to extracting textures and furniture shapes. To address the issue of room texture, we would extract the dominant colors of the room and objects and apply them to the walls and Furniture. Additionally, we would provide the user with the option to modify the color and texture of the furniture and walls. For the second issue of furniture shapes, we would offer the user a selection of pre-made 3D shapes to customize their furniture.Moreover, users can add furniture, with their desired dimensions. By combining the Apple API and our code, users would have complete freedom to customize their rooms by changing the furniture layout, textures, colors, and shapes to their liking. This would provide users with the ability to create their ideal rooms.

3.2.2 Customizing the Scanned Room

- Second, the user can alter the walls, furniture, windows, and doors in his space in a variety of ways. The user can change the wall's colors and textures when personalizing it. He can match the colors to the walls of his room by using a color picker or the extracted room colors. He can also apply textures to the walls; he can either select one from his gallery or one of our given ones. For the windows, doors, and furniture. The user has the option to alter not only their 3D shapes but also their rotation, color, and texture. He may even move them around the area and change their placement, with collision detection. The user can change the proportions and scale for the extracted furniture to correct any scanning errors. Furthermore, the user can add furniture to his room and customize it to his liking. Finally, The user can edit the light of the room type and color.

3.2.3 Optimizing the Room Design

- Last, The user will have the option to optimize his room to achieve the best furniture placement resulting in providing the best room movement and room design. This can be done using Reinforcement learning [4]. This method is divided into two types model-based and model-free, Q-Learning is model-free, which depends on a reward that is returned by the environment and the reward set to zero. Moreover, with each action, the Q-table is updated and the reward changes. We consider in our paper the agent is the furniture (e.g. sofa, table) and there are only four possible actions for the agent ([left, right, top, bottom]). There are two types of furniture that must be considered before placing the furniture inside the room, 1) Free-type and 2) Coupled-type. The free type depends on the room layout only and doesn't depend on other furniture (e,g. table, chair, sofa). On the other hand, the coupled type depends on other furniture (e,g. bed, and nightstands), and the nightstands depend on the bed's position.

3.3 System Scope

Our app utilizes the LiDAR sensor on compatible smartphones to scan and extract room features such as walls, doors, and furniture. Users can then edit colors, textures, and 3D shapes, and add furniture items to their 3D room. The app provides an AR view for realistic visualization and an optimization feature to automatically optimize furniture placement. Our goal is to simplify interior design, enabling users to create personalized and optimized room layouts without external assistance.

3.4 System Context

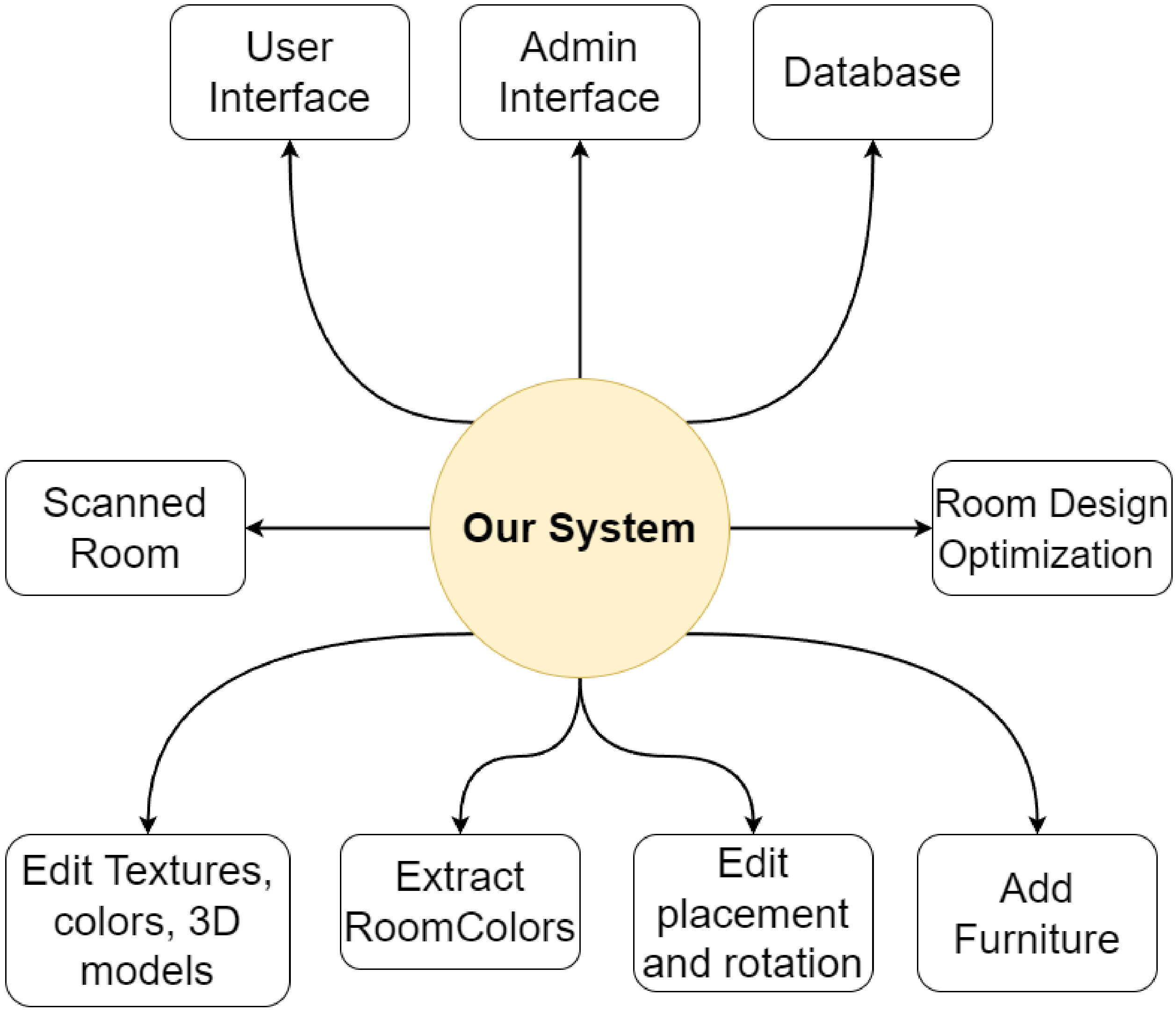


Figure 4: System Context

3.5 Objectives

1. User shall scan the room to convert it to a 3D model.
2. User shall edit walls, colors, textures and dimensions.
3. User shall edit Furniture, doors, windows colors, textures 3D Shapes, placement, and Rotation.
4. User shall edit Light.

5. Optimizing the interior design.
6. User shall view the 3D room in an AR View.

3.6 User Characteristics

4 Functional Requirements

4.1 System Functions

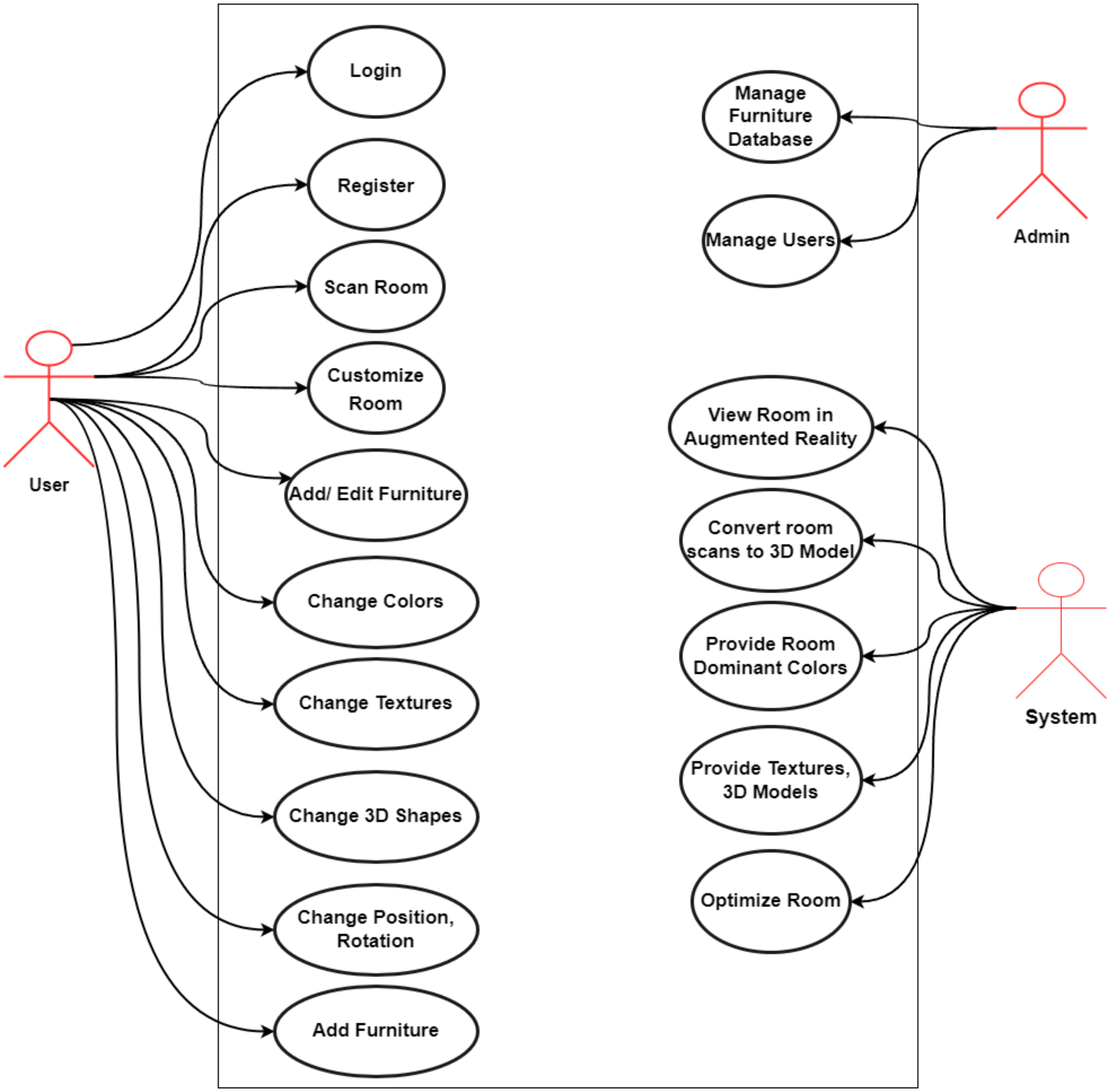


Figure 5: Use Case

1. **ID:01** The user shall register to be able to access the system.
2. **ID:02** The user/admin shall log in to access the system.
3. **ID:03** The user shall be able to scan the boundaries of his room to extract room dimensions.
4. **ID:04** The user shall be able to customize his room.
5. **ID:05** The user shall be able to add/edit furniture in dataset and scene.
6. **ID:06** The user shall be able to change colors of objects in room.
7. **ID:07** The user shall be able to change textures of surfaces in room.
8. **ID:08** The user shall be able to scan his room to convert to 3D Model
9. **ID:09** The user shall be able to change 3D model visualized.
10. **ID:10** The user shall be able to change the position and rotation of object.
11. **ID:11** The admin shall add 3D furniture to the dataset.
12. **ID:12** The admin shall be able to manage users.
13. **ID:13** The system shall be able to view room in AR view.
14. **ID:14** The system shall be able to convert Scanned rooms to 3D Models.
15. **ID:15** The system shall be able to Provide Room Dominant Colors.
16. **ID:16** The system shall be able to Provide Textures 3D Models.
17. **ID:17** The system shall be able to optimize furniture in the 3D room according to room dimensions and window/door constraints.

4.2 Detailed Functional Specification

Choose your main key functions (Minimum 3, Maximum 6).

Table 2: add3DModel

Name	change the The shape of Furniture
Code	ID:01
Priority	high
Critical	10/10
Description	import a 3D Model from the database, and get the current furniture scale, rotations and placement, then apply on the selected 3D Model.
Input	Selected Furniture and selected 3D Model.
Action	apply the 3D Model to the furniture
Output	3D model in the Room.
Pre-conditions	Scanned Room.
Post-condition	A 3D model will be saved to the Room.
Dependencies	F01

Table 3: Handle Joy Stick

Name	Handle Joy Stick
Code	ID:04
Priority	high
Critical	9/10
Description	This method works on two main parts. First, if the joystick is moving, then we adjust the location of the object by calculating the amount of movement and adding it to the location of the furniture in the Jason File. It also saves the location of the furniture. Secondly, if the joystick is not moving, then we place the location of the last piece of furniture so that it does not move from its position.
Input	User action
Action	Update the furniture location.
Output	furniture in the place where the user wants
Pre-conditions	select object to move.
Post-condition	moving furniture
Dependencies	-
Risk	-

Table 4: Optimize Furniture Function

Name	Optimize Furniture
Code	ID:07
Priority	high
Critical	10/10
Description	Optimize the room design.
Input	3D furniture, 3D model and room dimensions
Action	Press button to confirm optimization.
Output	3D model with rearranged furniture.
Pre-conditions	User already logged in.
Post-condition	User can view the rearrange room 3D model.
Dependencies	F02
Risk	-

Table 5: getJsoneFile Function

Name	getJsoneFile
Code	ID:10
Priority	high
Critical	10/10
Description	The function attempts to decode the received JSON data using the JSONDecoder class. If successful, the decoded JSON data is used to initialize a PrepareJsonToScene object, which is then returned via the completion handler.
Input	A URL string pointing to a JSON file that the function will download and decode, and a completion handler.
Action	The action of this function is to asynchronously download and decode a JSON file from a provided URL and return a scene.
Output	Values that will be translated into the scene.
Pre-conditions	User already logged in, network, and an available scene.
Post-condition	Translate the Json into an object to be used later.
Dependencies	-
Risk	-

Table 6: initRoomShape Function

Name	initRoomShape
Code	ID:11
Priority	high
Critical	10/10
Description	prepare walls and surfaces of room to be used in pygame by rotating them and caluting the new positions of walls after the rotation.
Input	dimensions, position, rotation, and category of walls and doors in a room.
Action	-
Output	pivot,pygame rectangles,pygame surfaces .
Pre-conditions	-
Post-condition	-
Dependencies	-
Risk	-

Table 7: addSurfaceModels Function

Name	addSurfaceModels
Code	ID:12
Priority	high
Critical	10/10
Description	convert decoded data to walls windows and doors.
Input	decoded scene data.
Action	-
Output	Walls,Windows,Doors and Surface objects.
Pre-conditions	User already logged in, network, and an available scene.
Post-condition	-
Dependencies	-
Risk	-

5 Interface Requirements

5.1 GUI

Figure 6: Scan

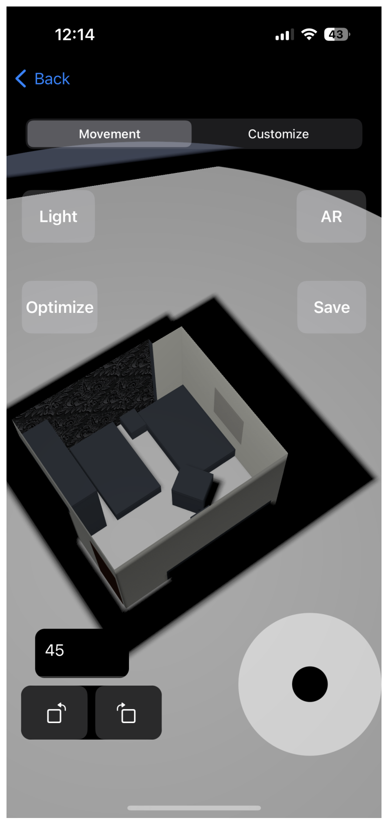


Figure 7: Movement

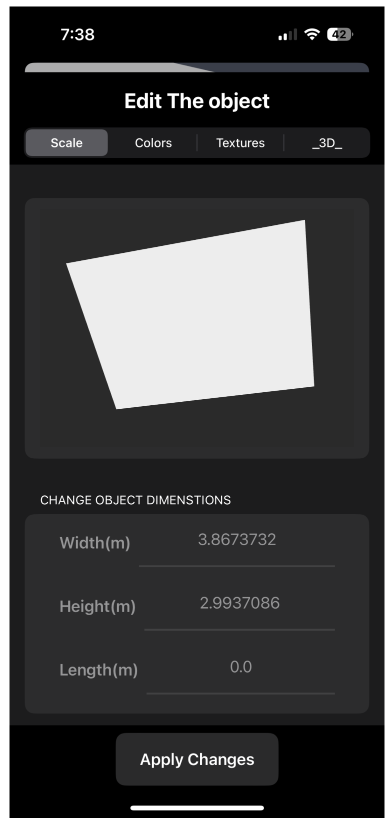


Figure 8: Scale

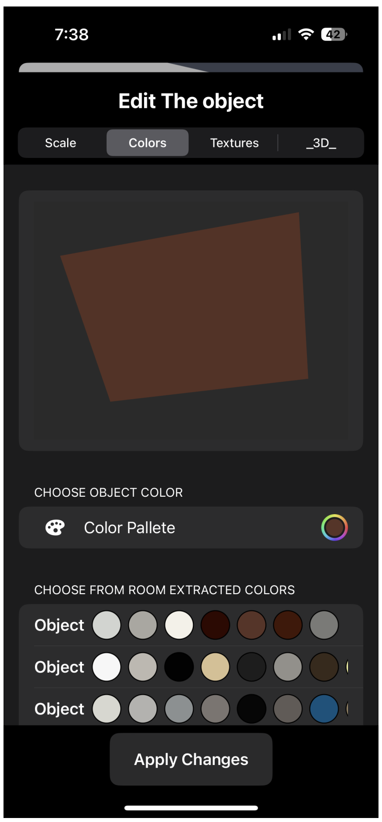


Figure 9: Extracted Colors

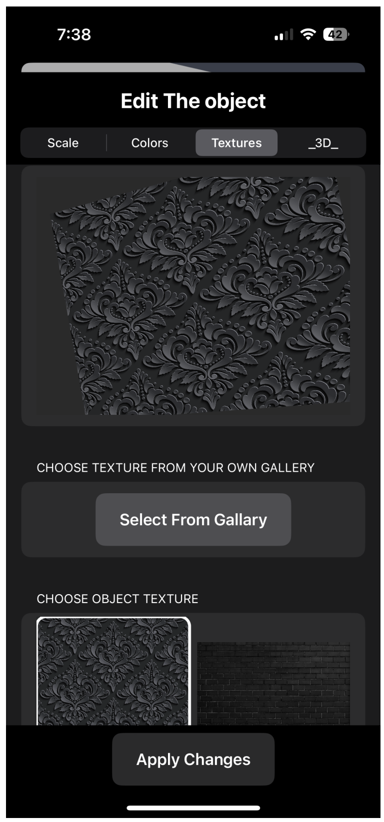


Figure 10: Textures

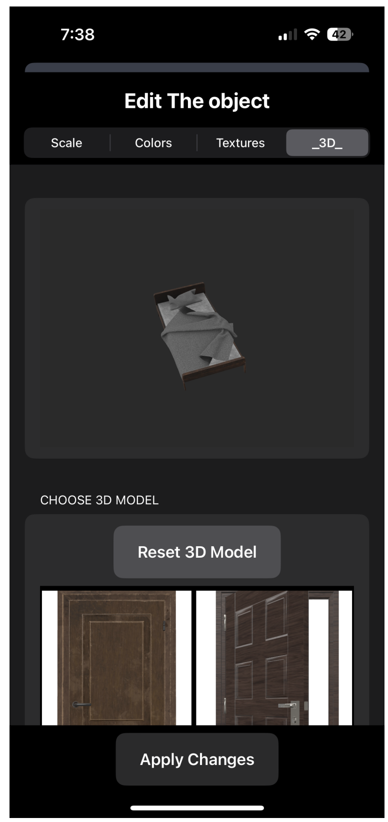


Figure 11: 3D Models

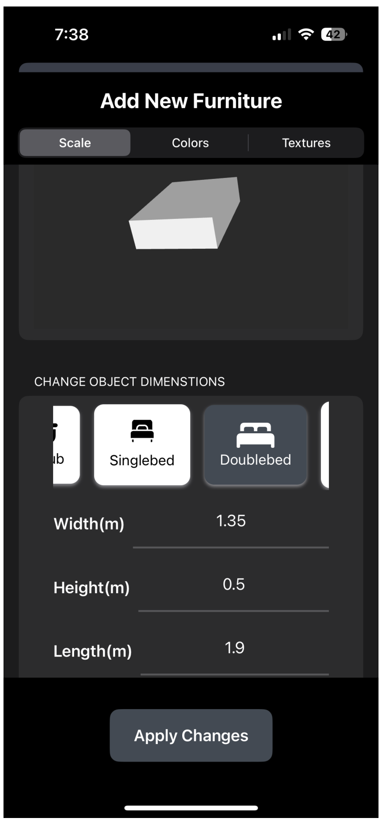


Figure 12: Add Furniture

Figure 13: CustomizedRoom

Figure 14: Ar View

6 Design Constraints

6.1 Standards Compliance

- iPhone 12 Pro or Higher.
- IOS 16.4 or higher

6.2 Hardware Limitations

The user must have an iPhone with a lidar sensor

6.3 Network constraints

It is essential to have a stable internet connection as the inference processing will be done on the cloud.

7 Non-functional Requirements

7.1 Security:

All data belonging to the user will be protected in addition to the password encryption, which will make it protected from any attack.

7.2 Portability:

It is a mobile application so that anyone can use it anywhere with an internet connection.

7.3 Usability:

1. The system shall have a user-friendly view, and the interface should be easily used by the users.
2. The user does not need any time to learn functionality.

7.4 Maintainability:

System is implemented in Model View View-Model (M V V M) design pattern. So, it can be easily maintained by developers.

7.5 Availability:

The software will always be available.

8 Data Design

8.1 DataBase Description

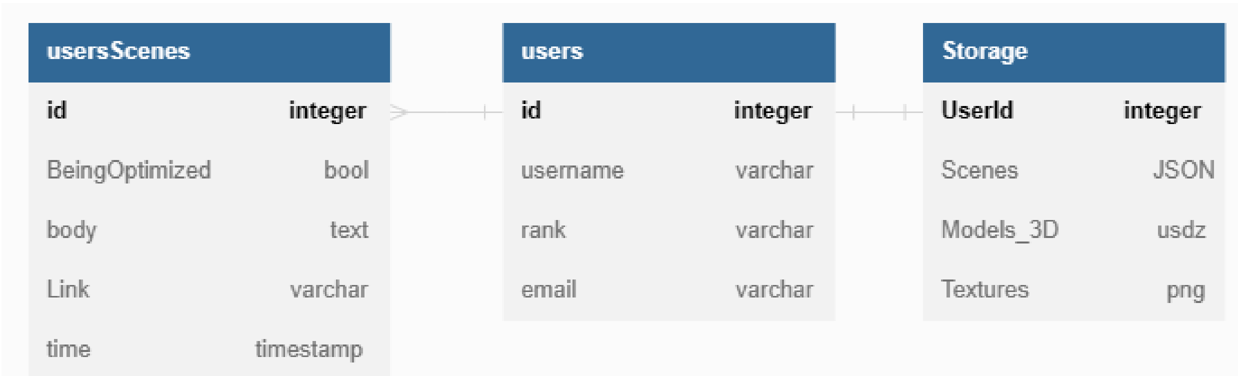


Figure 15: DataBase

8.2 Data Description

Our datasets are:

- (Scene.json) consists of four components:
  - scene Created Model which are the models created by the user.
  - specialID which is a uniqe id given to each scene composed of user id and scene id.
  - surfaces which is the id,position, rotation and scale of walls , doors , windows and openings.
  - objects which is the id,position, rotation and scale of different furniture.

9 Preliminary Object-Oriented Domain Analysis

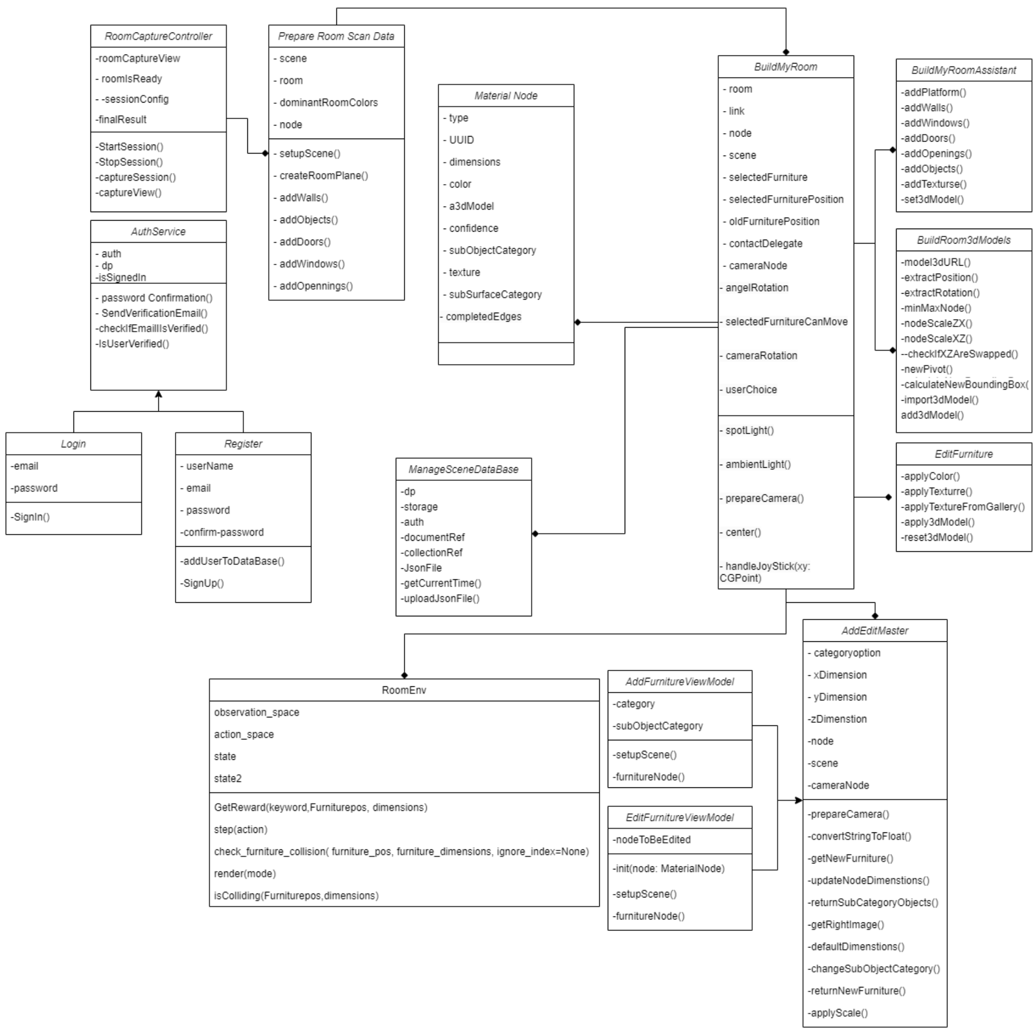


Figure 16: Class Diagram

10 Operational Scenarios

10.1 Scenario (1): Scanning room

The user starts scanning the entire room. The system shall convert the scanned room to a 3D model, that is available for the user to insert inside it the 3D furniture model.

10.2 Scenario (2): Optimization

The users will have the option of whether they want to optimize 3D furniture in the 3D-scanned room model or not. Based on their choice, the model shall find the best interior design.

10.3 Scenario (3): View 3D room

The user will be able to view the 3D-scanned room in a first person preview using AR.

11 Project Plan

Table 8: Time Plan

ID	Task	Start Date	Number of Days	Team Member
1	Searching for a datasets	13/9/2022	30	M , YB , YH , YA
2	Searching for resources	15/9/2022	100	M , YB , YH , YA
3	Writing Propsal	11/11/2022	10	M , YB , YH , YA
4	Implementation: 2d to 3D object (furniture)	15/11/2022	17	M,YH
5	Implementation: 2d to 3D object (house)	15/11/2022	20	YA.YB
6	Implementing Optimization	5/12/2022	60	YH
7	Writing SRS	7/12/2022	14	M , YB , YH , YA
8	Implementing Front End	1/1/2023	65	M , YA
9	Implementing Back End	26/12/2022	14	YA, M
10	Writing SDD	28/1/2023	20	YH, YA, YB, M
11	Implementation: Fusion	14/2/2023	55	YH, YA, YB, M
12	Implementation: VR	1/4/2023	40	YH, YA, YB, M
13	Writing final thesis	10/03/2023	90	YH, YA, YB, M
14	Experiment	10/04/2023	50	YH, YA, YB, M

12 Appendices

12.1 Definitions, Acronyms, Abbreviations

1. **Lidar:** light detection and ranging.
2. **DIY:** Do It Yourself

13 References

References

[1] Lars Mescheder, Michael Oechsle, Michael Niemeyer, et al. *Occupancy Networks: Learning 3D Reconstruction in Function Space*. 2019. arXiv: `1812.03828 [cs.CV]`.

[2] Wen Zhou, Wenying Jiang, Weixin Bian, et al. "Webvr Human-Centered Indoor Layout Design Framework Using a Convolutional Neural Network and Deep Q-Learning". In: *IEEE Access* 7 (2019), pp. 185773–185785. DOI: `10.1109/ACCESS.2019.2961368`.

[3] Kai Wang, Manolis Savva, Angel X Chang, et al. "Deep convolutional priors for indoor scene synthesis". In: *ACM Transactions on Graphics (TOG)* 37.4 (2018), pp. 1–14.

[4] Wen Zhou, Wenying Jiang, Weixin Bian, et al. "Webvr human-centered indoor layout design framework using a convolutional neural network and deep q-learning". In: *IEEE Access* 7 (2019), pp. 185773–185785.