# Backend Task

Quickly create a rich document.


🎯 Designing Scalable Authentication & Authorization Service

📑Explanation :

Jot down some text

☑️Task Checklist

✔️MUST Dos

✅BONUS

Links ( MUST )

Learning Outcome ( MUST )


## 🎯 Designing Scalable Authentication & Authorization Service

> 💡 **Please read the whole doc carefully !**

### 📑Explanation :

Design independent auth service with authorization for scale.

Assuming it would be used for 1 million user concurrent.

Service will contain auth functionality for **USER, ADMIN** and **OWNER.**

You need  to add authorization for Admin and Owner

**Description -**

- If any router is having access of user then the owner and admin will be restricted for that route.
- If any route is having access of admin then the owner will also have access for that route.
- If any route is having access of owner then admin or user can't access it.

# Jot down some text

They found Mary, as usual, deep in the study of thorough-bass and human nature; and had some extracts to admire, and some new observations of threadbare morality to listen to. Catherine and Lydia had information for them of a different sort.

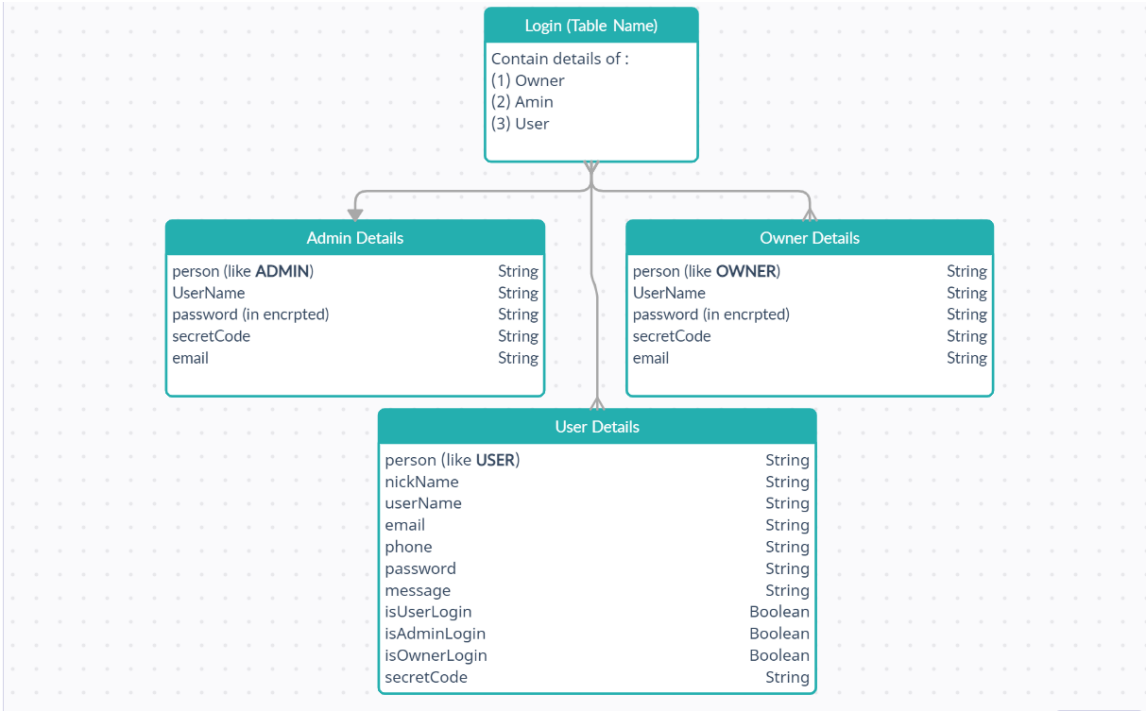
# ☑️Task Checklist

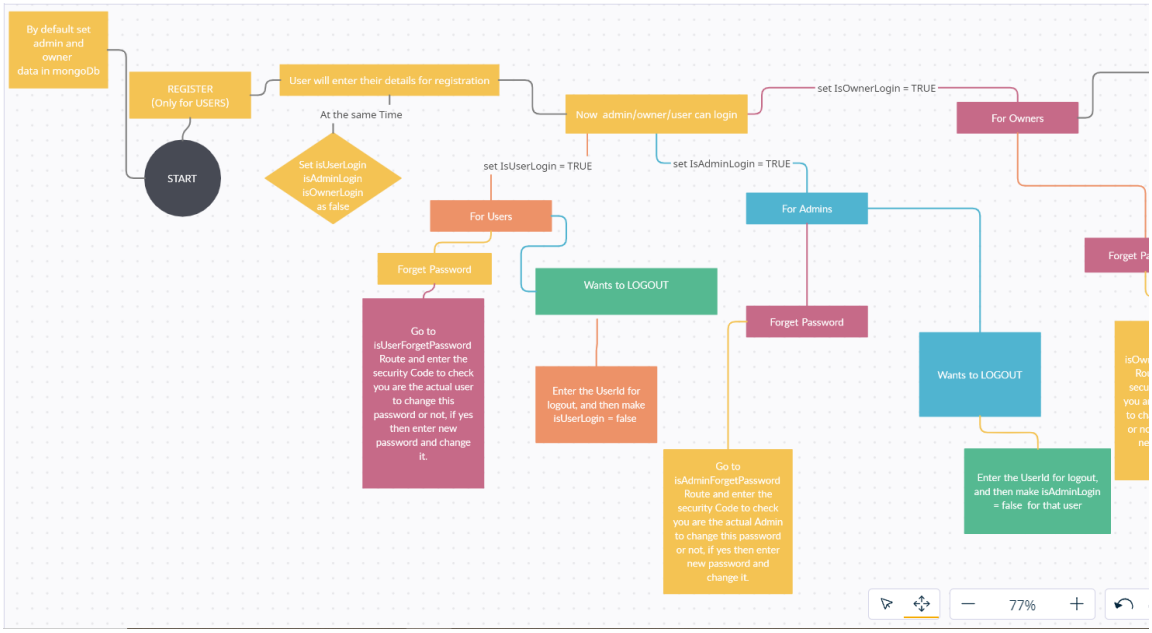
## ✔️MUST Dos

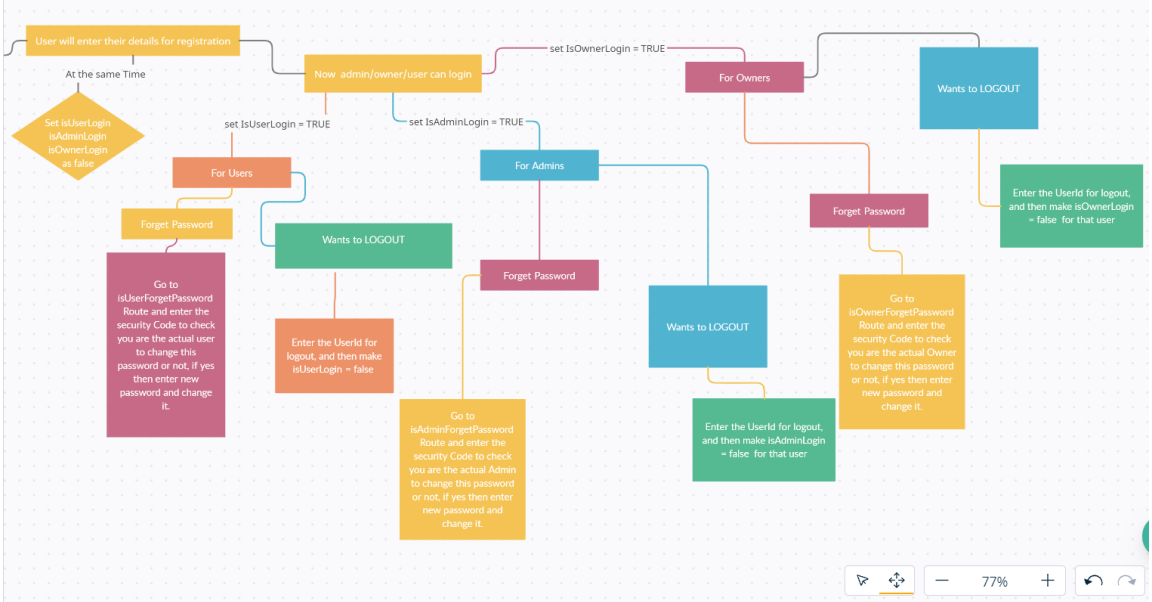
▼ **MUST**

The mandatory ones, based on these, the task will be MEASURED. Should submit the task only after completing these.

- [x] ~~Create Login Flow for **USER, ADMIN,** and **OWNER**.~~
- [x] ~~Create Signup flow for **USER.**~~
- [x] ~~Add Authorization~~
- [x] ~~Create 3 routes having access of the following types - **USER, ADMIN, OWNER**.~~
- [x] ~~**Documentation & Approach** *(Must Add this Points in your notion doc)* → Prepare the following assuming it would be used for 1 million user concurrently.~~
  - [x] ~~High level System Design (Flowchart) → Diagram image~~


Login (Table Name)
Contain details of :
(1) Owner
(2) Amin
(3) User
Admin Details
person (like ADMIN) String
UserName String
password (in encrpted) String
secretCode String
email String
Owner Details
person (like OWNER) String
UserName String
password (in encrpted) String
secretCode String
email String
User Details
person (like USER) String
nickName String
userName String
email String
phone String
password String
message String
isUserLogin Boolean
isAdminLogin Boolean
isOwnerLogin Boolean
secretCode String


By default set admin and owner data in mongoDb
REGISTER (Only for USERS)
START
User will enter their details for registration
At the same Time
Set isUserLogin isAdminLogin isOwnerLogin as false
Now admin/owner/user can login
set IsOwnerLogin = TRUE
For Owners
set IsUserLogin = TRUE
set IsAdminLogin = TRUE
For Users
For Admins
Forget Password
Wants to LOGOUT
Go to isUserForgetPassword Route and enter the security Code to check you are the actual user to change this password or not, if yes then enter new password and change it.
Enter the UserId for logout, and then make isUserLogin = false
Forget Password
Wants to LOGOUT
Go to isAdminForgetPassword Route and enter the security Code to check you are the actual Admin to change this password or not, if yes then enter new password and change it.
Enter the UserId for logout, and then make isAdminLogin = false for that user
77%




- [x] ~~Capacity Estimation and Constraints~~
- [x] ~~Database Schema~~

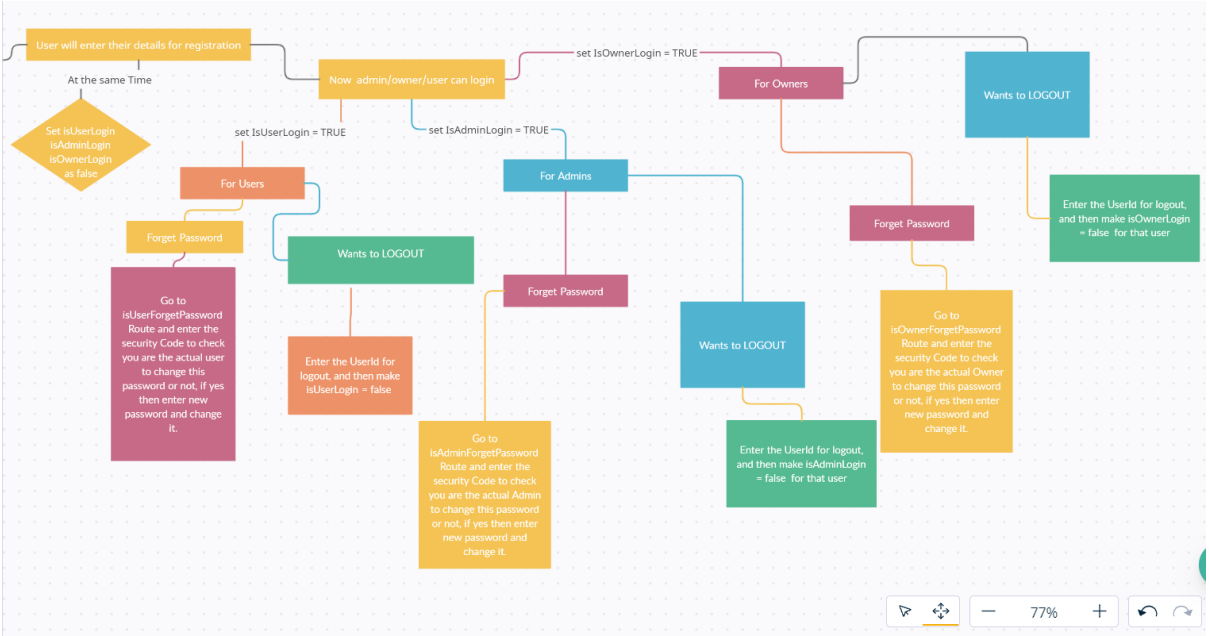

- [x] ~~Various Components Details~~
- [x] ~~Approach & Flow for Forget Password~~

## ✅BONUS

▼ BONUS

These are the points which will give you an advantage in the selection process if completed ( 🙂 )

- [x] ~~Create Forget Password flow for **USER, ADMIN**, and **OWNER**.~~




## Links ( MUST )

Please fill these links

**Notion Doc Link:** https://www.notion.so/Backend-Task-8b71046bff99461c9b2f44b2451166e9

**Github Repo Link :** https://github.com/profile/repo_name

**Demo Link :** ____(Deployed on any FAAS)

# Learning Outcome ( MUST )

Fill this section accordingly after completing the checklist

**Any complexity / difficulty faced while developing this app ?**

→ For the complexity part we set 3 boolean variables for user, admin, owner . So, that when admin and owner tries to login any user then for that user we set isAdmin

and isOwner values true . which takes care for the next time, that user is already logged-in by admin/owner .

**Reason for not completing any one of the Checklist Item ( if any ) ?** → [ **Completed All Tasks ]**

**Did you get to learn anything new / key concepts ?**

→ Yes, new concept that I genuinely learn are working with robo3T, as I usually work with mongoDb compass and I worked completely on express this time only promises (else I can do using async await promises also)

**Do you think this task is gonna benefit you in the job ?**

→ Yes, definitely because by completing this assignment makes me more confident to work on Nodejs, or its framework (Express), MongoDb, Mongoos

**Any feedback ?**

→ Good Leaning Experiance