|  |  |
| --- | --- |
| **Project Setup LMS MERN Becodemy** | **Date: 10-23-2023** |

**Topics**

Project Setup LMS MERN Becodemy

# Project Overview

Image or Video

# Project Plan / Software Design

Major Features

Authentication Feature

* Signup login manually, social login, Create user, login, authenticate, token, refresh token, profile page Minor Feature

Site Setting Feature from dashboard

FAQ can update through admin

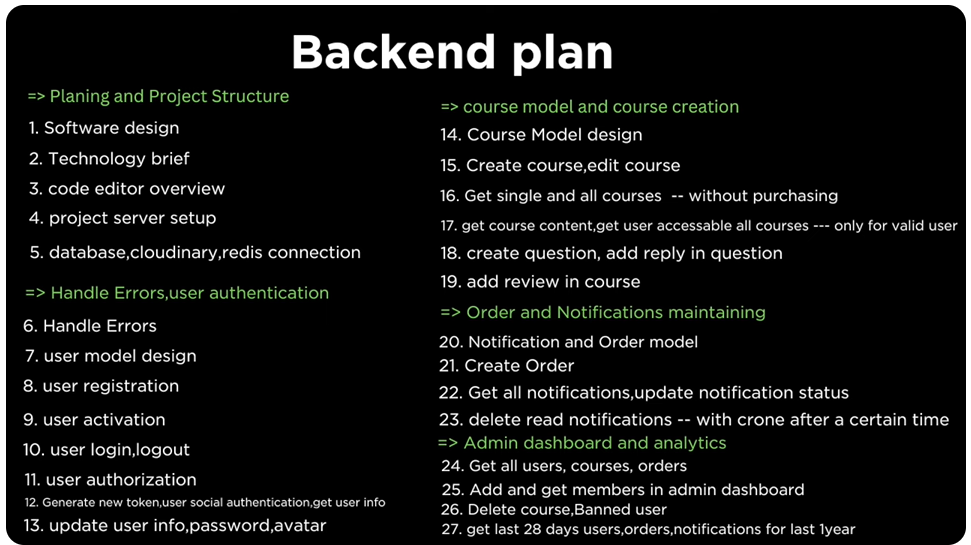
Courses Feature (Admin)

Categories can admin update (Admin)

Analytics (Admin)

Orders (Admin)

Notification (Admin)



# Technology brief

Which technology we will use and why we will use these technologies

Low scale app – Functionality not much big, lesser user visiting flow, like local news paper website in your country like urdu.

Suggested Technology – NextJS is fine in 2023, even to build backend apis.

Medium scale app – Some complex features, In future can grow app, some user visiting flow, like local news paper website in your country like becodemy, lms.

Suggested Technology – Sometime good NextJS is fine in 2023, sometime not in terms of backend like live class session, for backend is not good for frontend is fine even nodejs is not good, move to nest js if you are a js developer, or else Django or spring boot.

Large scale app – Many complex features, In future can grow app, many user visiting flow, like local news paper website in your country like youtube. Might you need to implement some custom algorithm.

Suggested Technology – NextJS is fine in 2023, even to build backend apis.

**In this project we are using**

**NextJS** for the frontend.

**NodeJS** for the backend.

**Redis** for maintain the cache.

* Why cache maintain is important? Basically you have lot of data in your website, if you don’t have larger user then you can skip cache, if at the same time 10 thousand user hit the same course request then might server gets hang, so we can cache that data, and there are two kind of caching frontend and backend,
* So in backend we can use redis, redis is also kind of database like Mongodb, it is useful to store cache data in the backend, we will that same course into the cache our server goes to cache and pick that course instead of database just like Netflix movies cache.

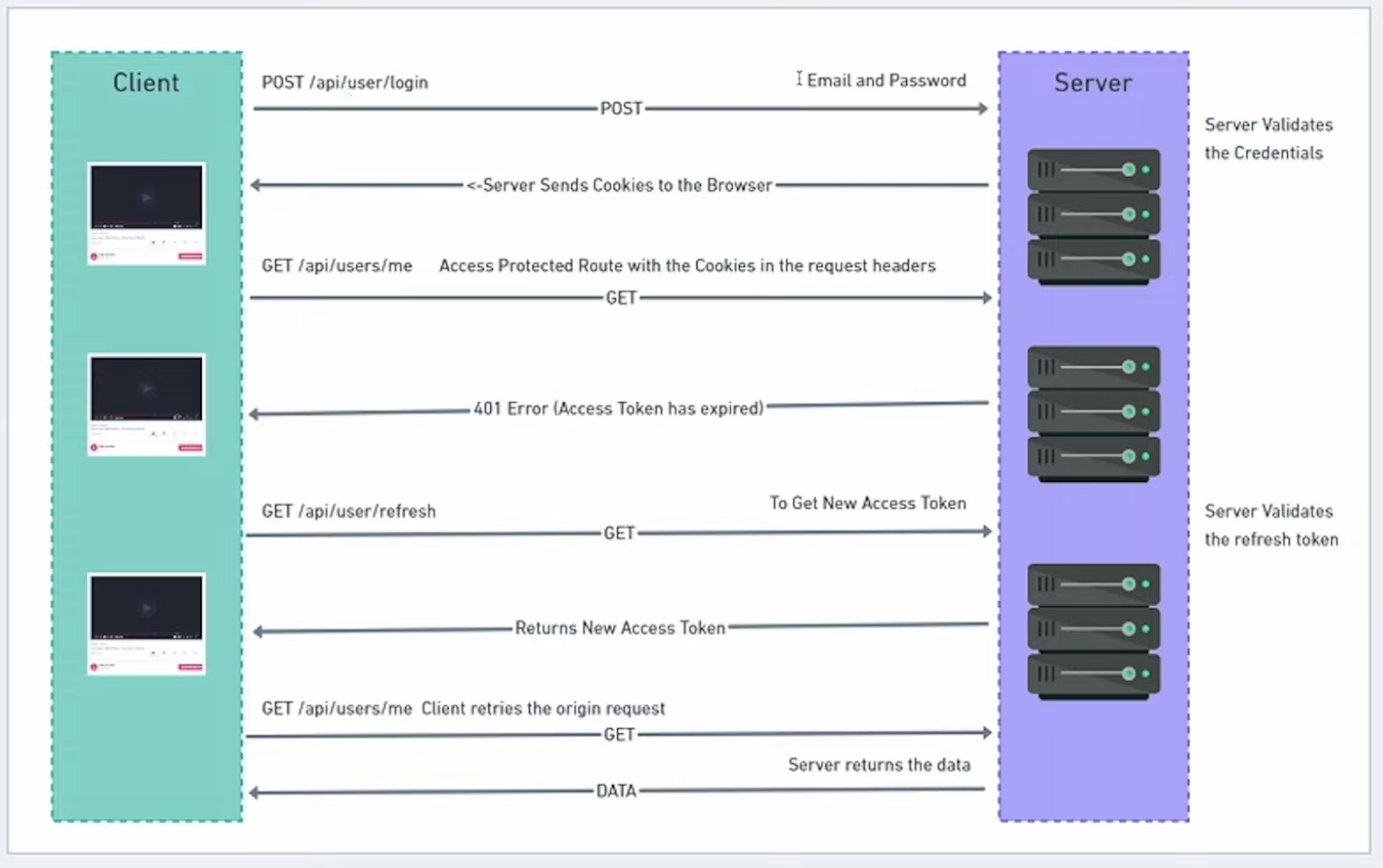
**RTK QUERY**, instead of redux or redux toolkit.

* Why we are using RTK Query not redux, RTK query basically cached the data so when we move to the another url it will not send request again but redux toolkit will send the request again and again, when the data expire or removed from cache then it will re-fetch the data.

**Tailwind CSS** for styling.

**EJS** used for sending dynamic email template.

**Authentication API**



# Code Editor overview

jj

# Project server setup

jj

npm init

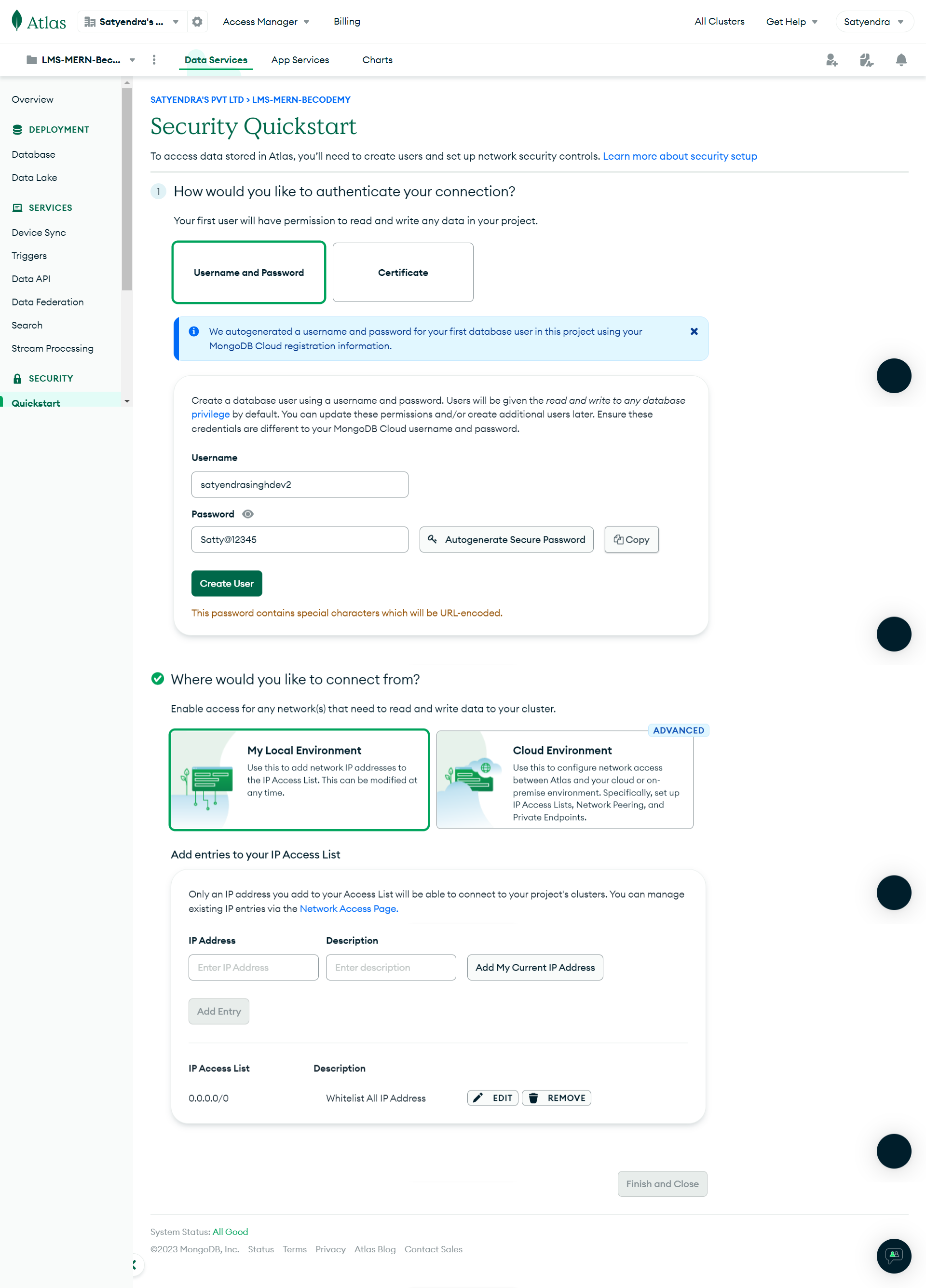
$ npm i bcryptjs cookie-parser cors dotenv express ioredis jsonwebtoken mongoose ts-node-dev @types/bcryptjs @types/cookie-parser @types/cors @types/express @types/jsonwebtoken @types/node typescript

$ npm i ejs nodemailer @types/ejs @types/nodemailer cloudinary

<https://www.mongodb.com/>

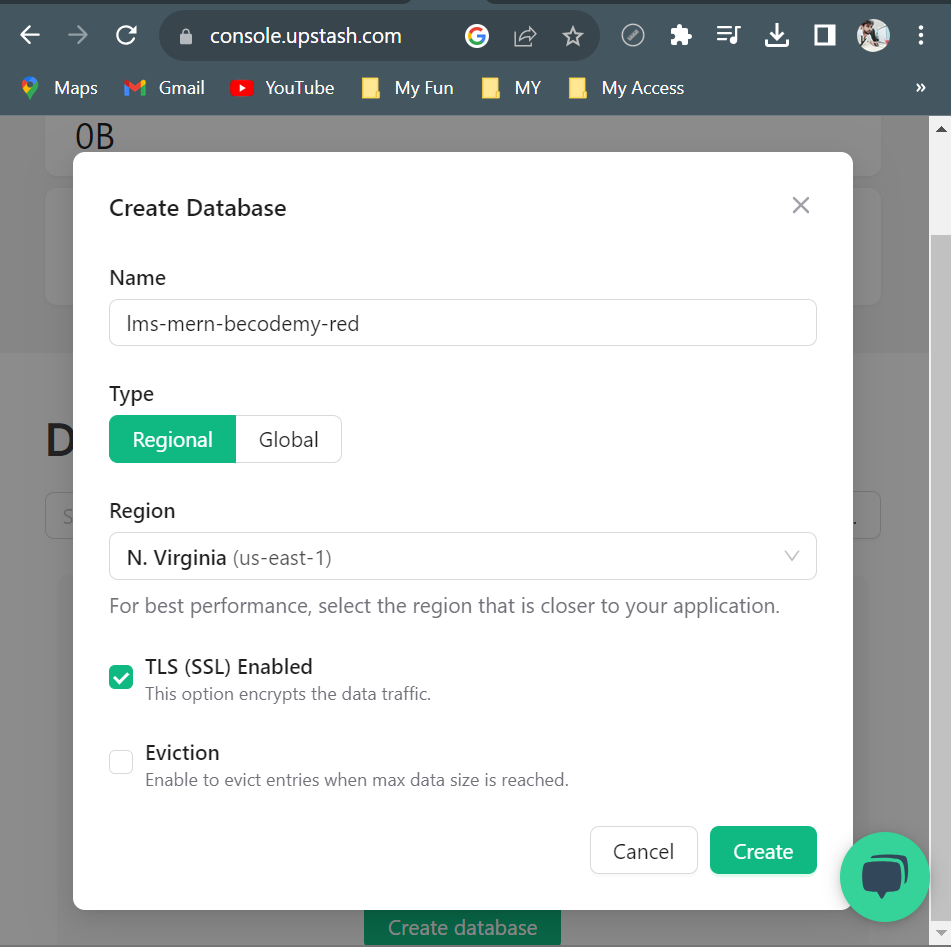
<https://upstash.com/> -- redis

<https://cloudinary.com/>



Satty12345

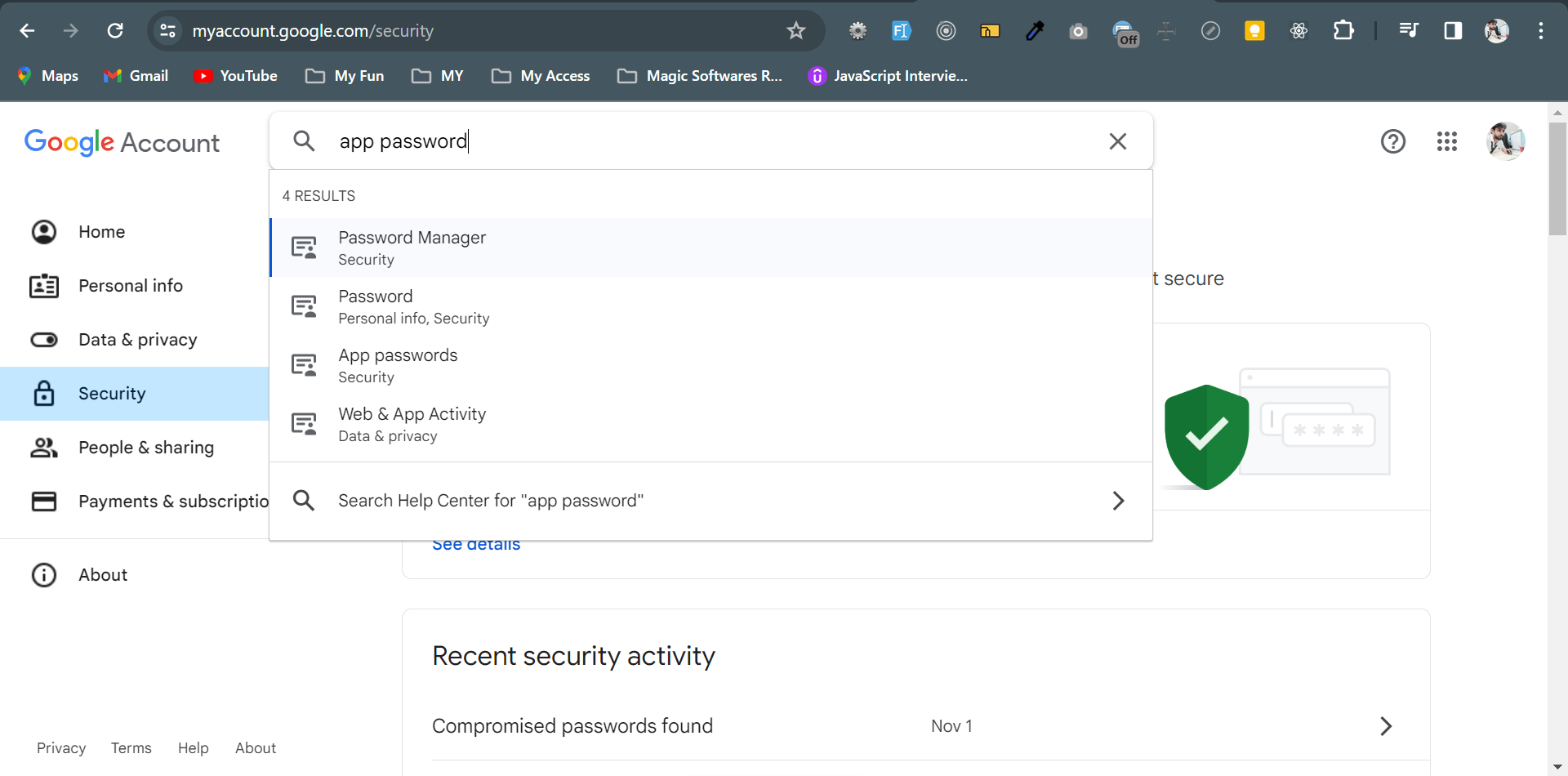
# Database, Cloudinary, Redis connection



server> npx tsc –init - this will create tsconfig file with few configuration

Setup for nodemail

https://myaccount.google.com/security



# NodeJS Project Setup LMS MERN Becodemy

jj