

Backend API

1. Node version and installation process

a. Version: 18.2.0

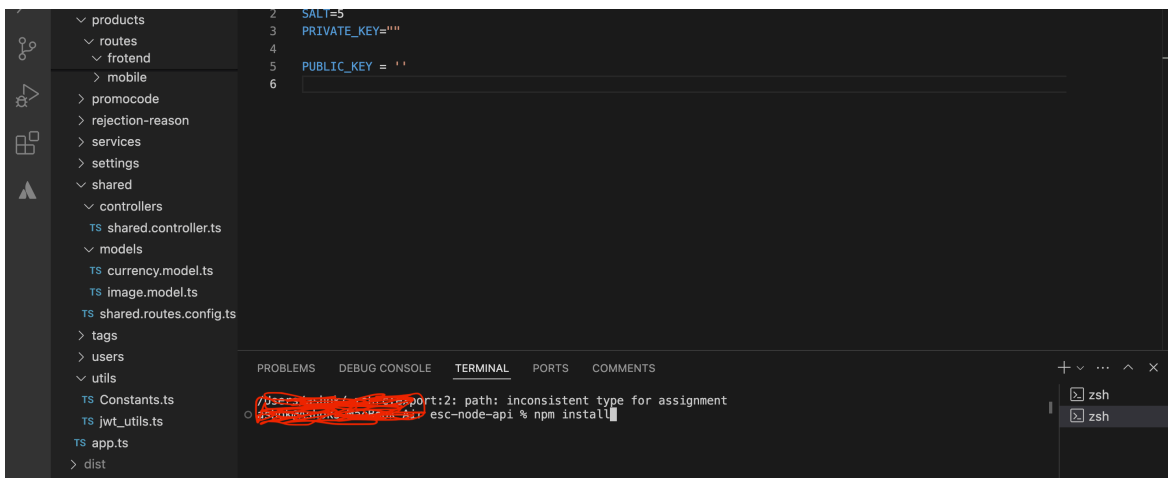
b. Go to this site for install Node js <https://nodejs.org/en/download>

2. Setup

a. Unzip project

b. Install dependency

command - **npm install**

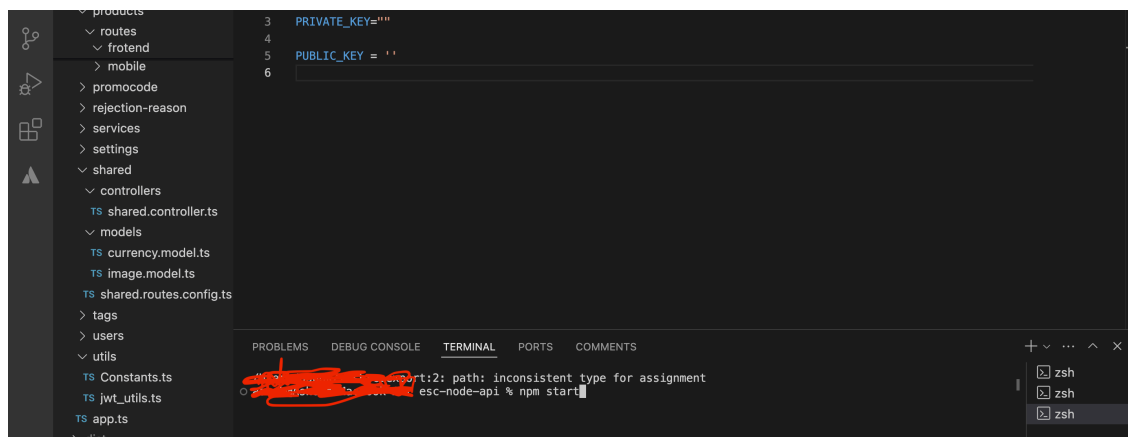


Install dependency

c. Run your project

command- **npm start**

Open <http://localhost:3012> to access.



Run your project

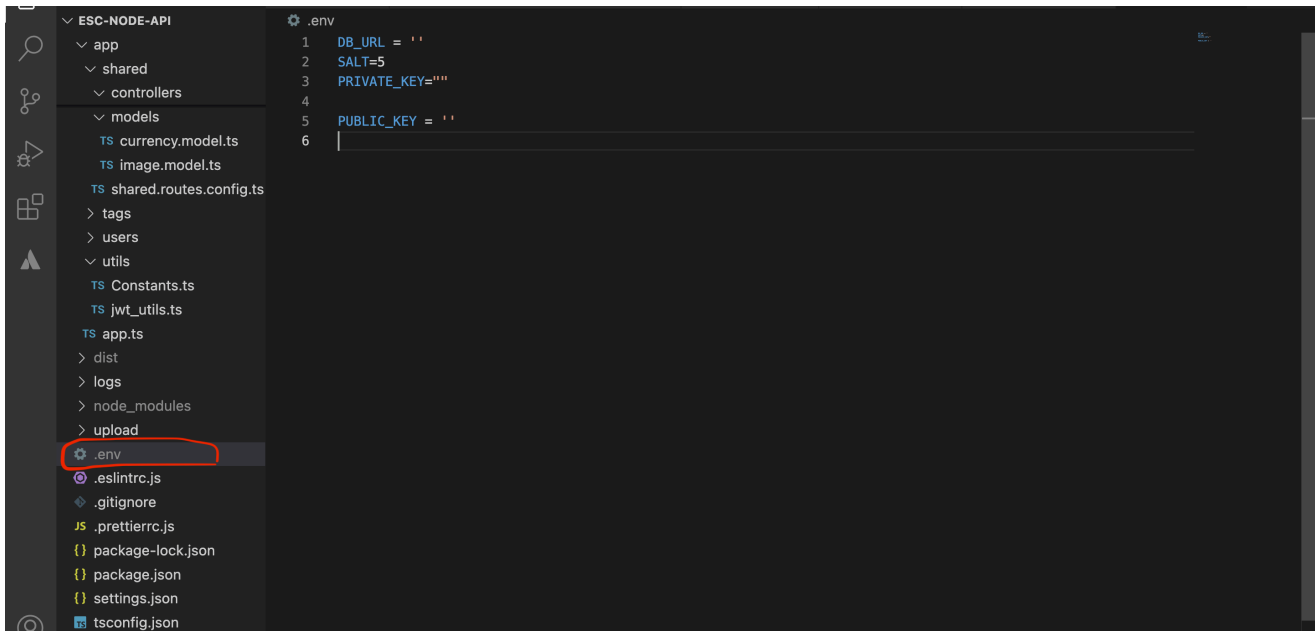
3. DATABASE:

Database: MongoDB

How to connect?

Inside .env file

DB_URL="your connection url"

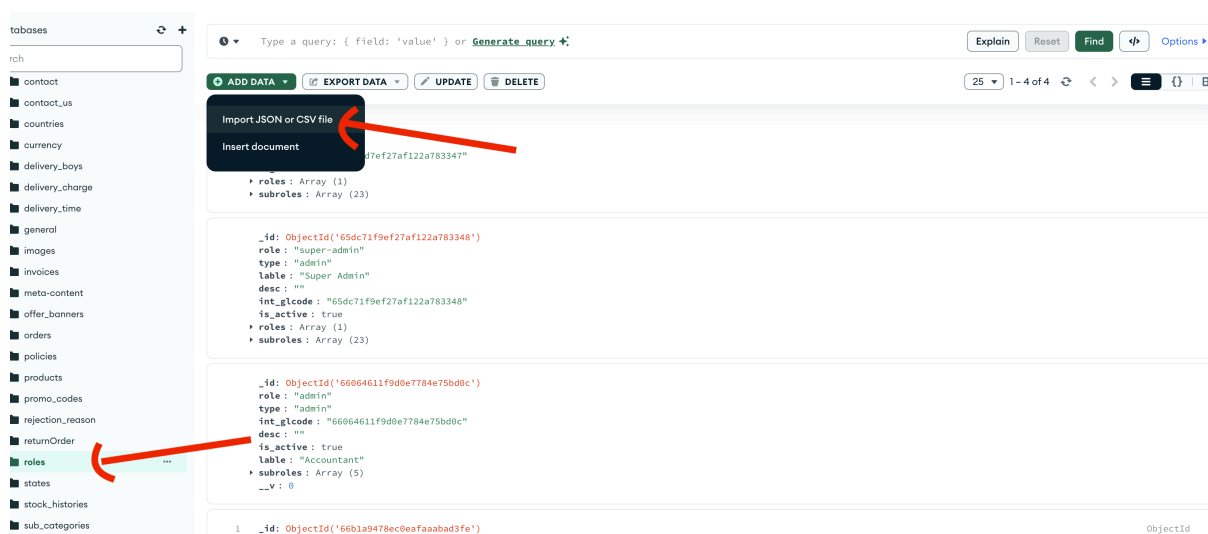


DATABASE

Import user and roles in your database:

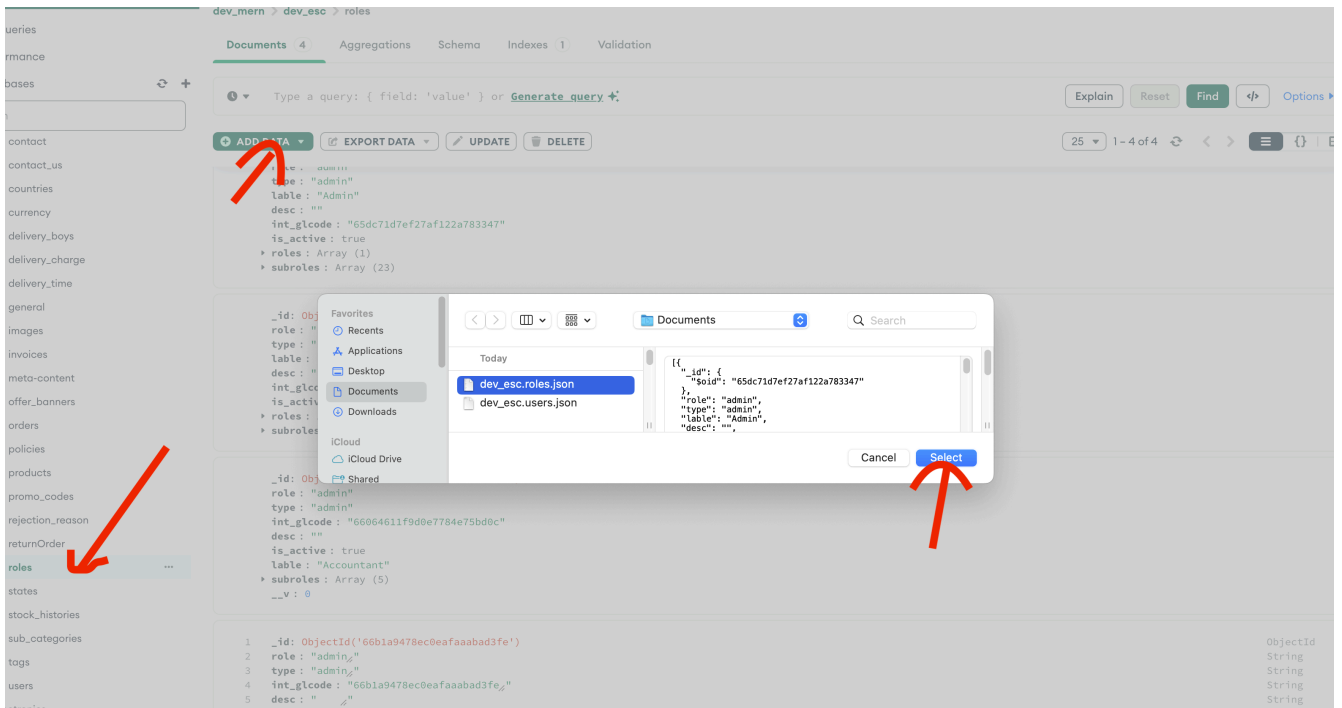
1. Import user role

Step-1



Caption

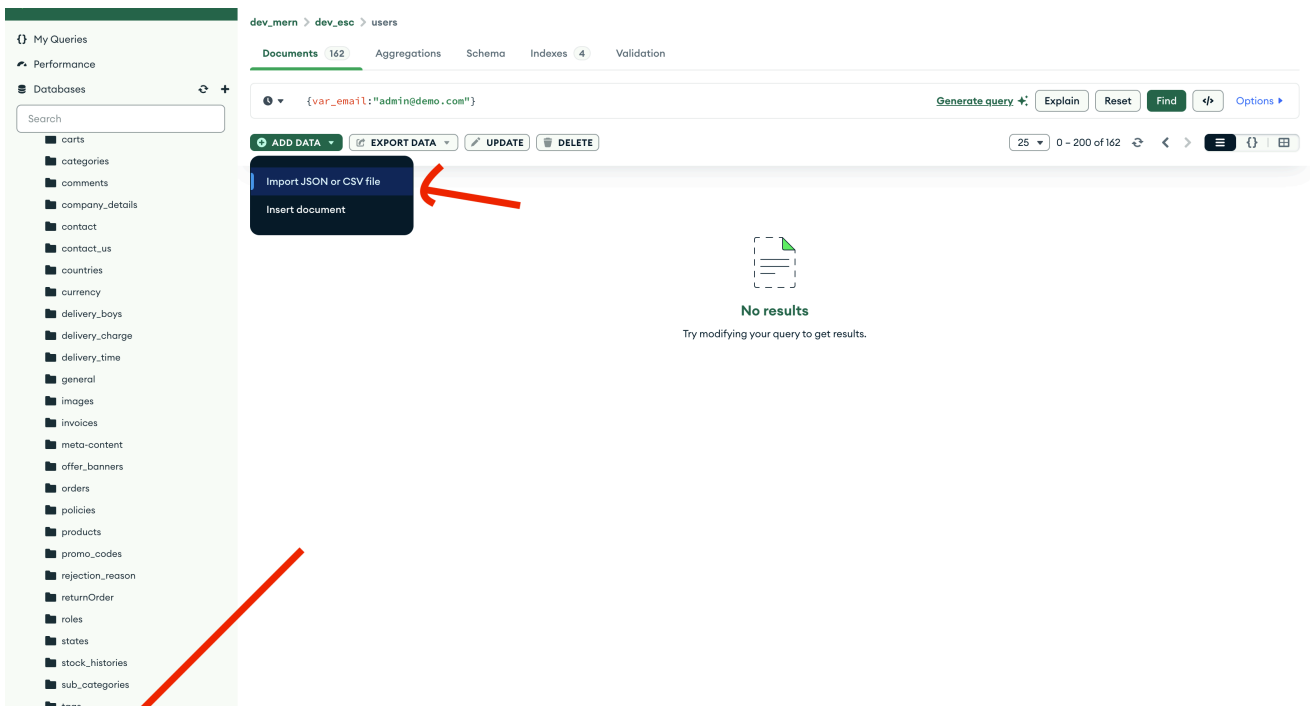
Step -2



Caption

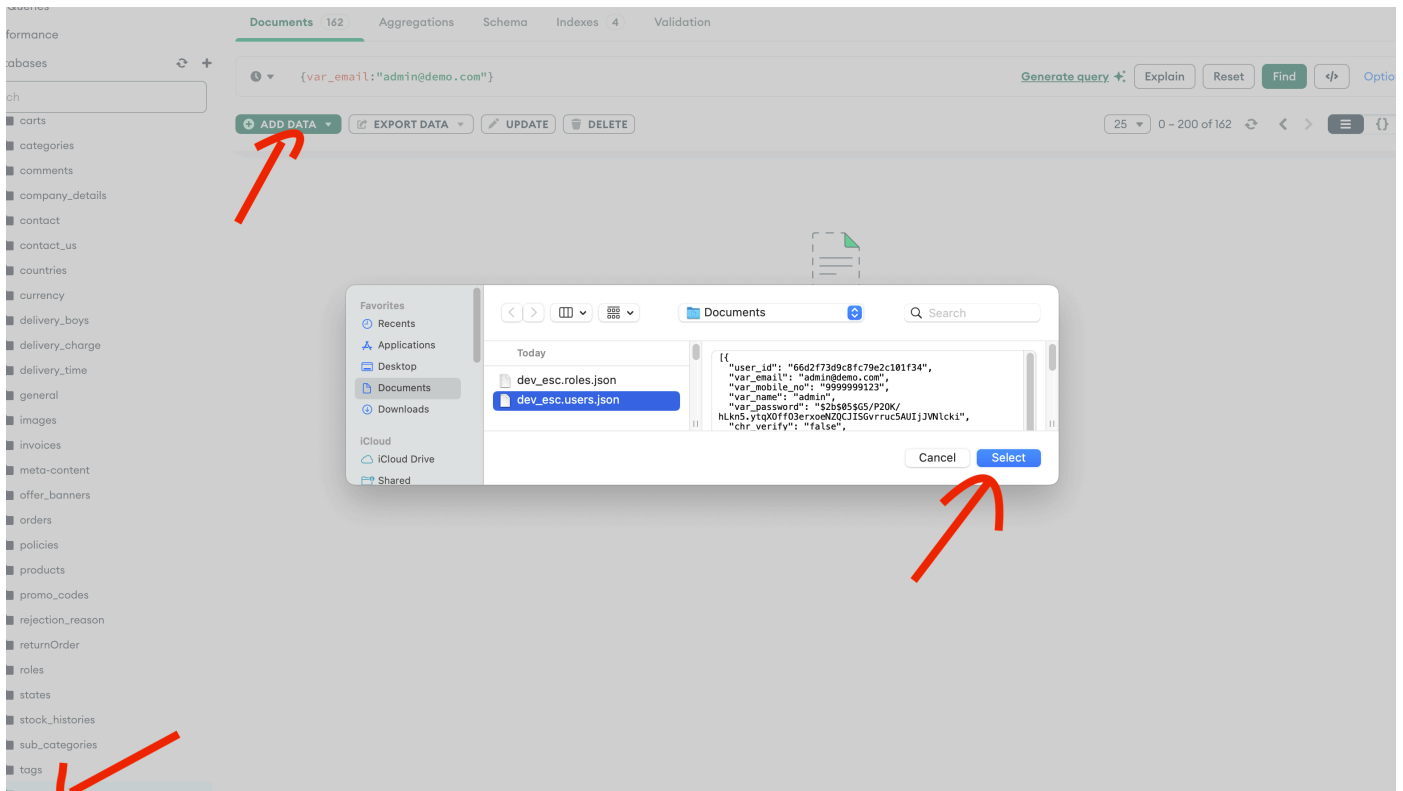
2. Import user

Step-1



Caption

Step- 2



Caption

Default user and password:

User: admin@demo.com

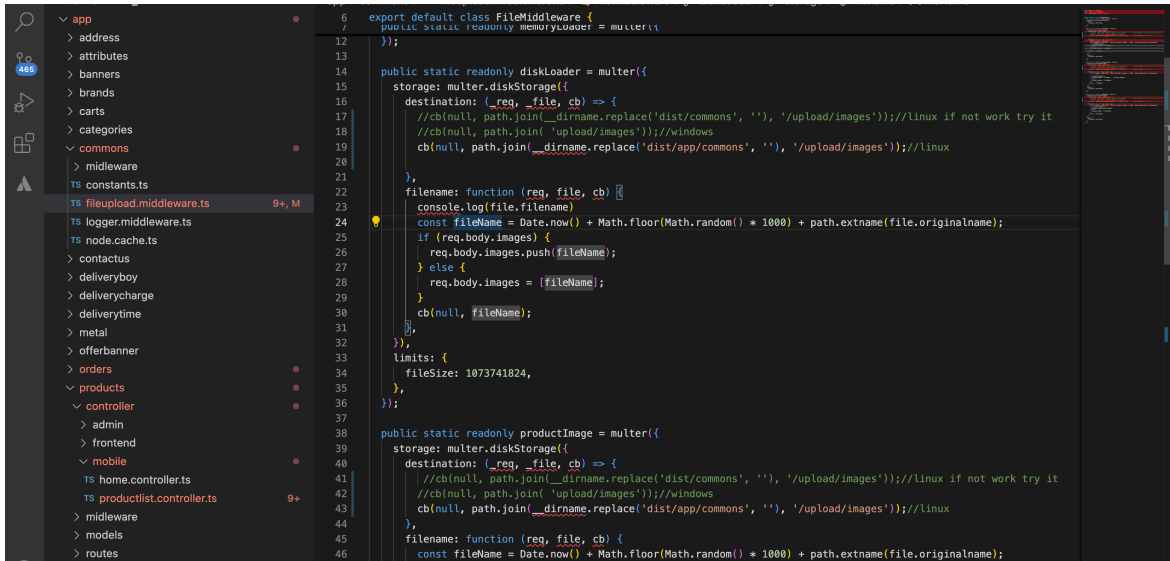
Password: Qaz@1234

Push notification:

Replace service_account.json to your service account to action push notification.

Image upload issue:

If you use window then uncomment window setting or if get any error while image upload in linux system then uncomment other image setting and try.

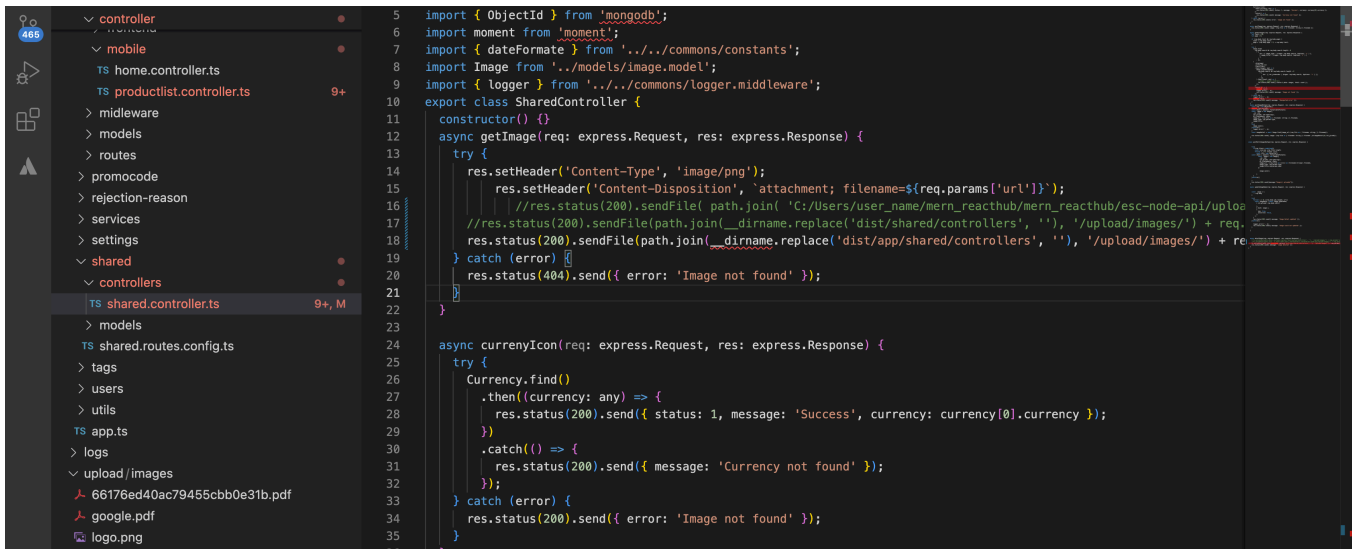


```
6 export default class FileMiddleware {
7   public static readonly memoryLoader = multer({
8     // ...
9   });
10
11   public static readonly diskLoader = multer({
12     storage: multer.diskStorage({
13       destination: (_req, _file, cb) => {
14         //cb(null, path.join(__dirname.replace('dist/commons', ''), 'upload/images')); //Linux if not work try it
15         //cb(null, path.join('upload/images')); //windows
16         cb(null, path.join(__dirname.replace('dist/app/commons', ''), 'upload/images')); //linux
17       },
18       filename: function (req, file, cb) {
19         console.log(file.filename)
20         const fileName = Date.now() + Math.floor(Math.random() * 1000) + path.extname(file.originalname);
21         if (req.body.images) {
22           req.body.images.push(fileName);
23         } else {
24           req.body.images = [fileName];
25         }
26         cb(null, fileName);
27       },
28     }),
29     limits: {
30       fileSize: 1073741824,
31     },
32   });
33
34   public static readonly productImage = multer({
35     storage: multer.diskStorage({
36       destination: (_req, _file, cb) => {
37         //cb(null, path.join(__dirname.replace('dist/commons', ''), 'upload/images')); //Linux if not work try it
38         //cb(null, path.join('upload/images')); //windows
39         cb(null, path.join(__dirname.replace('dist/app/commons', ''), 'upload/images')); //linux
40       },
41       filename: function (req, file, cb) {
42         const fileName = Date.now() + Math.floor(Math.random() * 1000) + path.extname(file.originalname);
43       },
44     }),
45   });
46 }
```

Caption

Image fetch issue:

If you use window then uncomment window setting or If get any error in image fetching in linux system, un comment other linux setting and try.



```
5 import { ObjectId } from 'mongodb';
6 import moment from 'moment';
7 import { dateFormat } from '../commons/constants';
8 import Image from '../models/image.model';
9 import { logger } from '../commons/logger.middleware';
10 export class SharedController {
11   constructor() {}
12   async getImage(req: express.Request, res: express.Response) {
13     try {
14       res.setHeader('Content-Type', 'image/png');
15       res.setHeader('Content-Disposition', 'attachment; filename=${req.params['url']}');
16       //res.status(200).sendFile(path.join( 'C:/Users/user_name/mern_reacthub/mern_reacthub/esc-node-api/upload/images/' + req.
17       //res.status(200).sendFile(path.join(__dirname.replace('dist/shared/controllers', ''), 'upload/images/') + req.
18       res.status(200).sendFile(path.join(__dirname.replace('dist/app/shared/controllers', ''), 'upload/images/') + req.
19     } catch (error) {
20       res.status(404).send({ error: 'Image not found' });
21     }
22   }
23
24   async currencyIcon(req: express.Request, res: express.Response) {
25     try {
26       Currency.find()
27         .then((currency: any) => {
28           res.status(200).send({ status: 1, message: 'Success', currency: currency[0].currency });
29         })
30         .catch(() => {
31           res.status(200).send({ message: 'Currency not found' });
32         });
33     } catch (error) {
34       res.status(200).send({ error: 'Image not found' });
35     }
36   }
37 }
```

Caption