**Models / post**

const mongoose = require('mongoose');

const postSchema = new mongoose.Schema({

title: {

type: String,

required: "Title is required",

},

body: {

type: String,

required: "Body is required",

}

});

module.exports = mongoose.model("post", postSchema);

npm init

npm i express

mongoose

body-parser

jsonwetoken

express-validator

dotenv

Create folders

routes controllers models

**App.js**

const express = require('express');

const mongoose = require('mongoose');

const bodyParser = require('body-parser');

const expressValidator = require('express-validator');

const dotenv = require('dotenv');

dotenv.config();

const postRoute = require('./routes/post');

const app = express();

// db

mongoose.connect(process.env.MONGO\_URI,{

useNewUrlParser: true })

.then(()=>{

console.log("Database Connected")

})

mongoose.connection.on('error', err => {

conole.log(`Db connection error: ${err.message}`);

});

const port = process.env.PORT || 8080;

app.use(bodyParser.json());

app.use(expressValidator())

app.use('/api/v1', postRoute);

app.listen(port , ()=> {

console.log(`listening on port: ${port}`); });

**Controllers**

const Post = require("../models/post");

exports.createPost = (req, res) => {

const post = new Post(req.body);

return post.save()

.then((post) =>{res.status(200).json({post});})

// Without Validator Our Code will be like shown Below

// return post.save((err,result) => {

// if(err) {

// return res.status(400).json({error:err.message});}

// res.status(200).json({success: true, post: result}); // } );

}

exports.getallPost = (req, res) => {

return Post.find().select("body title")

.then((post) => res.status(200).json({success: true, post:post}))

.catch(err => console.log(err));

}

**AUTHORIZATION**

Graphical user interface, text, application, letter, email

Description automatically generated

Graphical user interface, text, application, letter, email

Description automatically generated

**Controllers**

const Post = require("../models/post");

exports.createPost = (req, res) => {

const post = new Post(req.body);

return post.save()

.then((post) =>{res.status(200).json({post});})

// Without Validator Our Code will be like shown Below

// return post.save((err,result) => {

// if(err) {

// return res.status(400).json({error:err.message});}

// res.status(200).json({success: true, post: result}); // } );

}

exports.getallPost = (req, res) => {

return Post.find().select("body title")

.then((post) => res.status(200).json({success: true, post:post}))

.catch(err => console.log(err));

}

Graphical user interface, text, application, email

Description automatically generated

**express-validator**

const expressValidator = require(‘express-validatpr’)

app.use(expressValidator());

Create a folder of Validators.

Create a file index.js

exports.createPostvalidator = (req,res,next)=>{

req.check(“title”, “Write a title).notEmpty();

req.check(‘title’,’Title must be 4-100 char long’). isLength({min:4,max:100)};

// check for errors

Const allerrors = req.validationErrors()

// if error – return the first one

if(error)

{

const firsterror = allerrors.map((err)=>{err.msg}[0]

return res.status(400).json({ error: firsterror});

}

**Get/Controller**

const Post = require("../models/post");

exports.getPost = (req, res) => {

const post = Post.find();

return Post.find().select(‘\_id title body’)

.then((posts) =>{res.status(200).json({posts});})

.catch( err => console.log(err));

}