



Job Search Application

Task - 4



Job search Application

- This section would provide an overview of the job search application and its purpose. It would also outline the scope of the application and the stakeholders involved

LMS Username	Name	Batch
2111a4532	Yasmeen.M	A5
2111a4526	Indhuja.R	A5
2111a4528	Kaviya.G	A5



Task 4 :: Backend (Module 4)

Do database modelling and create models

- Design schema for all the data to be stored
- Start mongodb local server and point the backend to the server
- Create mongoose schema at the backend
- Run test queries to set up the database

Create Various APIs to ensure data flow within the website

- Define all the routes for the backend
- Add authentication middleware
- Add controllers for all the paths to handle api request
- Create Environment variables for all authentication keys
- Note: Always handle all possible cases with the request

Evaluation Metric:

100% Completion of the above tasks

Learning outcome

- Understanding Nosql databases modeling
- Querying and filtering mongodb
- Understanding various req methods
- Getting familiar with cookies
- Server side authentication

Step-Wise Description

1. Install node expresss in my vs code
2. Create mongoDB atles account
3. visible package.json and package-lock.json
4. Created index.js , mongodb.js ,signup.hbs, login.hbs and home.hbs files.
5. We are using local host port no 3000
6. Mongo db connect with index.js
7. Database load using condition (or) method is async await
8. Then check dummy data

Summary of your task

- Design schema for all the data to be stored
- Start mongodb local server and point the backend to the server
- Create mongoose schema at the backend
- Run test queries to set up the database



EXPLORER



JS mongodbjs X



OPEN EDITORS

X JS mongodbjs src

JOB SEARCH APPLICATION

> node_modules

> public

src

JS index.js

JS mongodbjs

> templates

{ } package-lock.json

{ } package.json

≡ yasmin task 4.rar

> OUTLINE

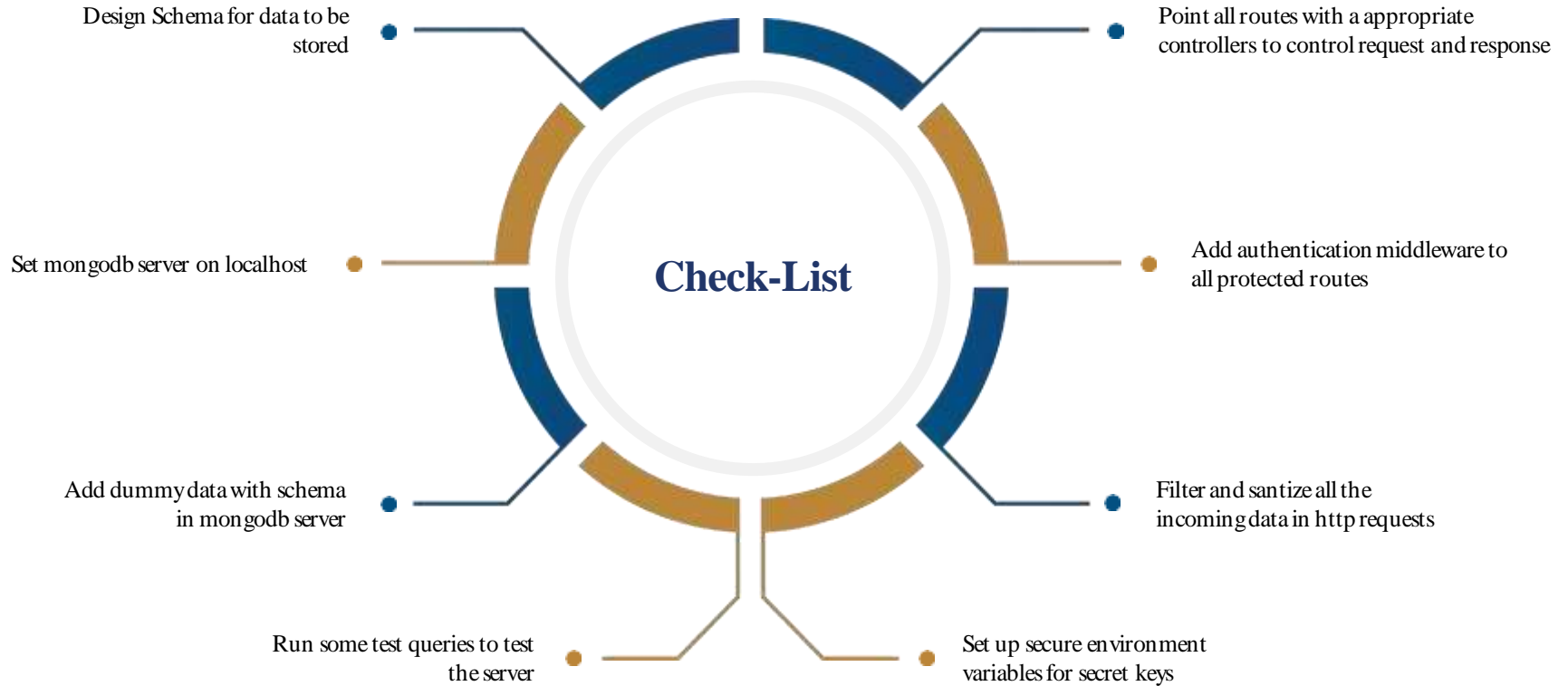
> TIMELINE

src > JS mongodbjs

```
1  const mongoose=require("mongoose")
2
3  mongoose.connect("mongodb://localhost:27017/LoginFormPractice")
4  .then(()=>{
5    console.log('mongoose connected');
6  })
7  .catch((e)=>{
8    console.log('failed');
9  })
10
11  const logInSchema=new mongoose.Schema({
12    name:{
13      type:String,
14      required:true
15    },
16    password:{
17      type:String,
18      required:true
19    }
20  })
21
22  const logInCollection=new mongoose.model('LogInCollection',logInSchema)
23
24  module.exports=LogInCollection
```

```
const mongoose = require('mongoose');
const mongoose = require('mongoose');
const mongoose = require('mongoose');
const mongoose = require('mongoose');
const mongoose = require('mongoose');
```

Assessment Parameter



Submission Github



[https://github.com/yasuma2611
2002/yasmeen-.git](https://github.com/yasuma26112002/yasmeen-.git)

Thank you!

