

- 1) Project for Rest API- Done and added to repository
- 2) Project for standalone HTML page along with css and js - Done and added to repository
- 3) Dev Environment setup document
  - Front End - HTML, CSS, Bootstrap
  - Back End - NodeJS for API
  - Database - MongoDB

Steps:

- 1) Install Latest Version of NodeJS, Express, Nodemon, Cors through terminal from npm packages and also install mongoose.
  - 2) First Run Rest\_api\_indicus. To Run the application first add the folder into vscode and open terminal and type "npm run dev"
  - 3) Now directly run the index.html file from the fetch\_api folder.
- 4) DB ER diagram and db creation script = Done and added to repository
  - 5) Document explaining the code:

First I have created an api folder named rest\_api\_indicus. I have first installed express and mongoose. Now to handle the request and response and routes i have created file named server.js created a port it will either run in default system port no or localhost:5000

Next, i have created a cluster in mongodb and in the network access i have given allow access from anywhere. Now to connect to the cluster with our application mongodb provides connection string

The screenshot shows the MongoDB Atlas 'Connect' wizard. At the top, there are three tabs: 'Setup connection security' (checked), 'Choose a connection method' (checked), and 'Connect'. Below the tabs, step 1 is 'Select your driver and version'. It features two dropdown menus: 'DRIVER' set to 'Node.js' and 'VERSION' set to '2.2.12 or later'. Step 2 is 'Add your connection string into your application code'. It includes an unchecked checkbox for 'Include full driver code example'. Below this is a text area containing the connection string: 'mongodb://<username>:<password>@cluster0-shard-00-00-8s7sk.mongodb.net:2701'. A 'Copy' button is to the right of the text area. At the bottom, there is instructional text: 'Replace <password> with the password for the <username> user. Replace <dbname> with the name of the database that connections will use by default. Ensure any option params are URL encoded.'

✓ Setup connection security    ✓ Choose a connection method    Connect

1 Select your driver and version

DRIVER	VERSION
Node.js	2.2.12 or later

2 Add your connection string into your application code

☐ Include full driver code example

```
mongodb://<username>:<password>@cluster0-shard-00-00-8s7sk.mongodb.net:2701
```

Copy

Replace <password> with the password for the <username> user. Replace <dbname> with the name of the database that connections will use by default. Ensure any option params are URL encoded.

Now this connection string is inside config.js file. Now i have created a schema inside Posts.js which represents the type of data we are going to store. Next i have created routes for different api calls this file is available inside routes/api folder here i have handled all the requests such as post, delete, update, get.

Next i have created separate project for front end with bootstrap components and cards to display the contents with edit and delete links. Now inside main.js i have written our api path which is <http://localhost:5000/api/posts> with the help of fetch **JavaScript** interface for accessing and manipulating parts of the HTTP pipeline, such as requests and responses. Now we can access the url contents. Now before we able to access the content we need to install cors which helps us to enable cross-origin communication.