|  |
| --- |
| Faculty of Applied Sciences and Technology |
| **NODE/EXPRESS WEB API** |
| ITE5315 - Project |
|  |
| **Group Member name**  **Mona Raju Shah-N01479948**  **Srikruthi Nallavelli-N01478510**  **Submission Date** |
| **12/5/2022** |

|  |
| --- |
| This document explains how to build Node/Express Web API …………………………. |

Table of Contents

[Question 1: 2](#_Toc100081673)

[Question 2: 3](#_Toc100081674)

[Question 3: 4](#_Toc100081675)

[Question 4: 5](#_Toc100081676)

[Question 5: 6](#_Toc100081677)

[Question 6: 7](#_Toc100081678)

[Summary 8](#_Toc100081679)

# Question 1:

(Describe the major steps for implementing the MongoDB database in Atlas )

In the first question we are asked to load the dataset which is called sample\_restaurants which contains the collection called restaurants.

In that collection we have all the records on which we are asked to perform all the operations and add the routes particularly so that we can fetch the data.

**Loading data in Atlas:**

*Graphical user interface, text, application, email

Description automatically generated*

In MongoDB Compass:

Graphical user interface, text, application, email

Description automatically generated

# Question 2:

(Describe the major steps for implementing Routes in the API, how you test this program, add some screenshots of the output)

For initializing this project, we had to start off with a schema and added all the dependencies.

So, we created a folder to store the schema and we checked whether the schema is correct then we added a sample route to get all the restaurants and checked it through postman.

Next, we are asked to use async function so we created a method.js in the modules folder and then for initializing routes we are using a file called index.js and the main file will be app.js in that we are initializing the file index.js in which all the routes are present.

We used db.initialize() function.

Text

Description automatically generated

db.addNewRestaurant(data): Create a new restaurant in the collection using the object passed in the "data" parameter:

Graphical user interface, text, application, email

Description automatically generated

db.getAllRestaurants(page, perPage, borough):

Graphical user interface, text

Description automatically generated

db.getRestaurantById(Id):

Graphical user interface, text, application, email

Description automatically generated

updateRestaurantById(data,Id):

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

deleteRestaurantById(Id):

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application

Description automatically generated

# Question 3:

(Describe the major steps for designing the FORM/UI, how you test this program, add some screenshots of the output)

For designing the form we initially took a route of perpage,page and borough.

By using handlebars we created a structure and in the views.

So we created a route and initialized it to the form which means it will render the form and when we click on submit button it should display all the information.

Table

Description automatically generated

# Question 4:

(Describe the major steps for implementing security features, how you test this program, add some screenshots of the output)

Graphical user interface, text, website

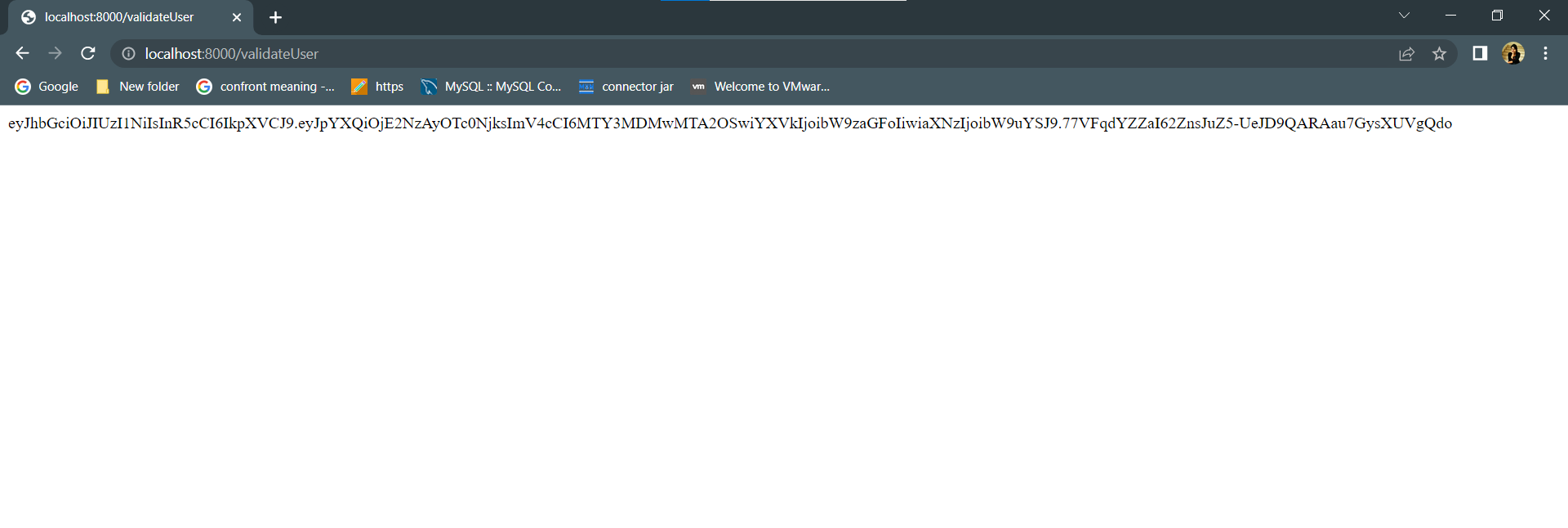
Description automatically generated

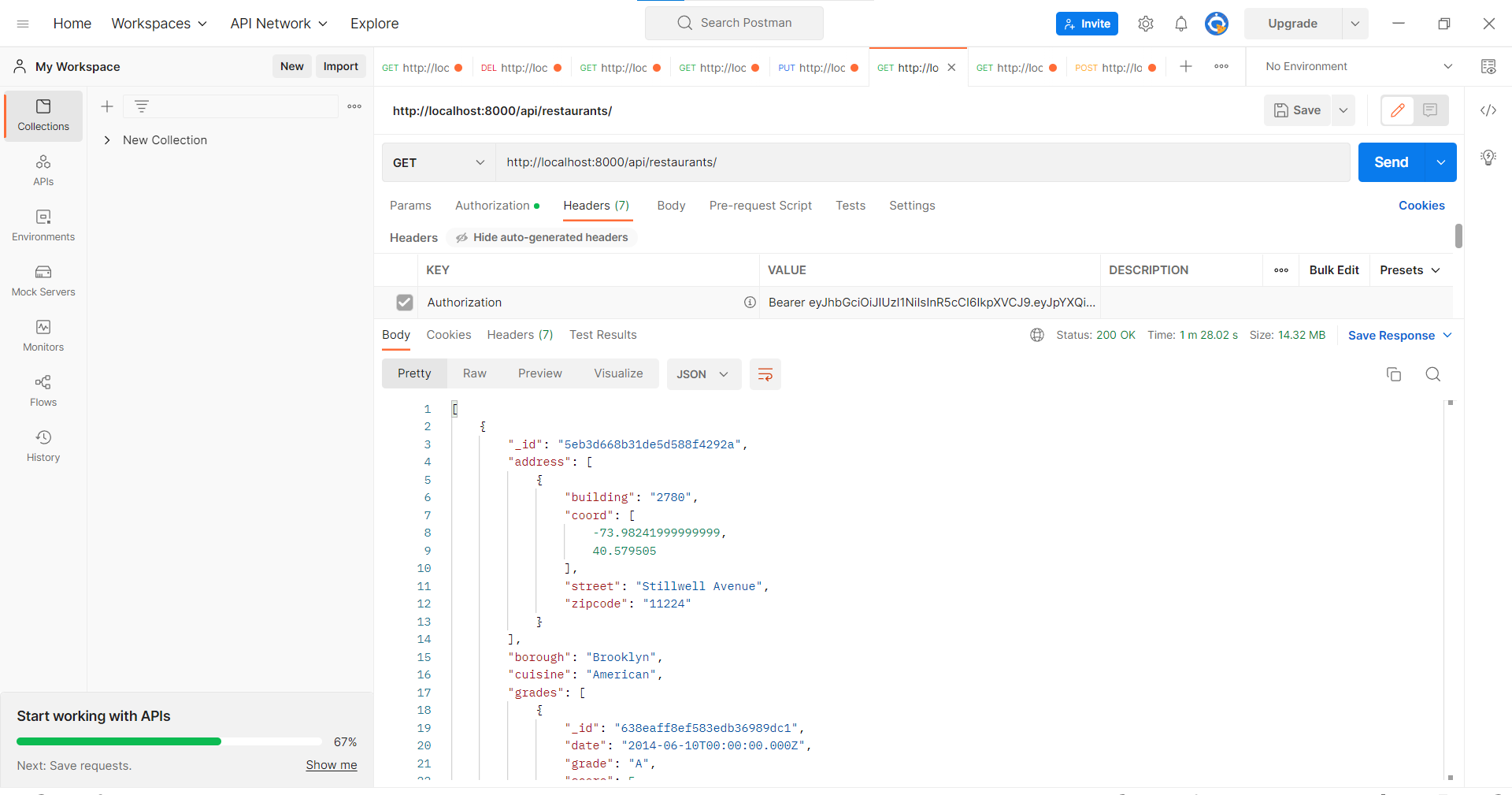
Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

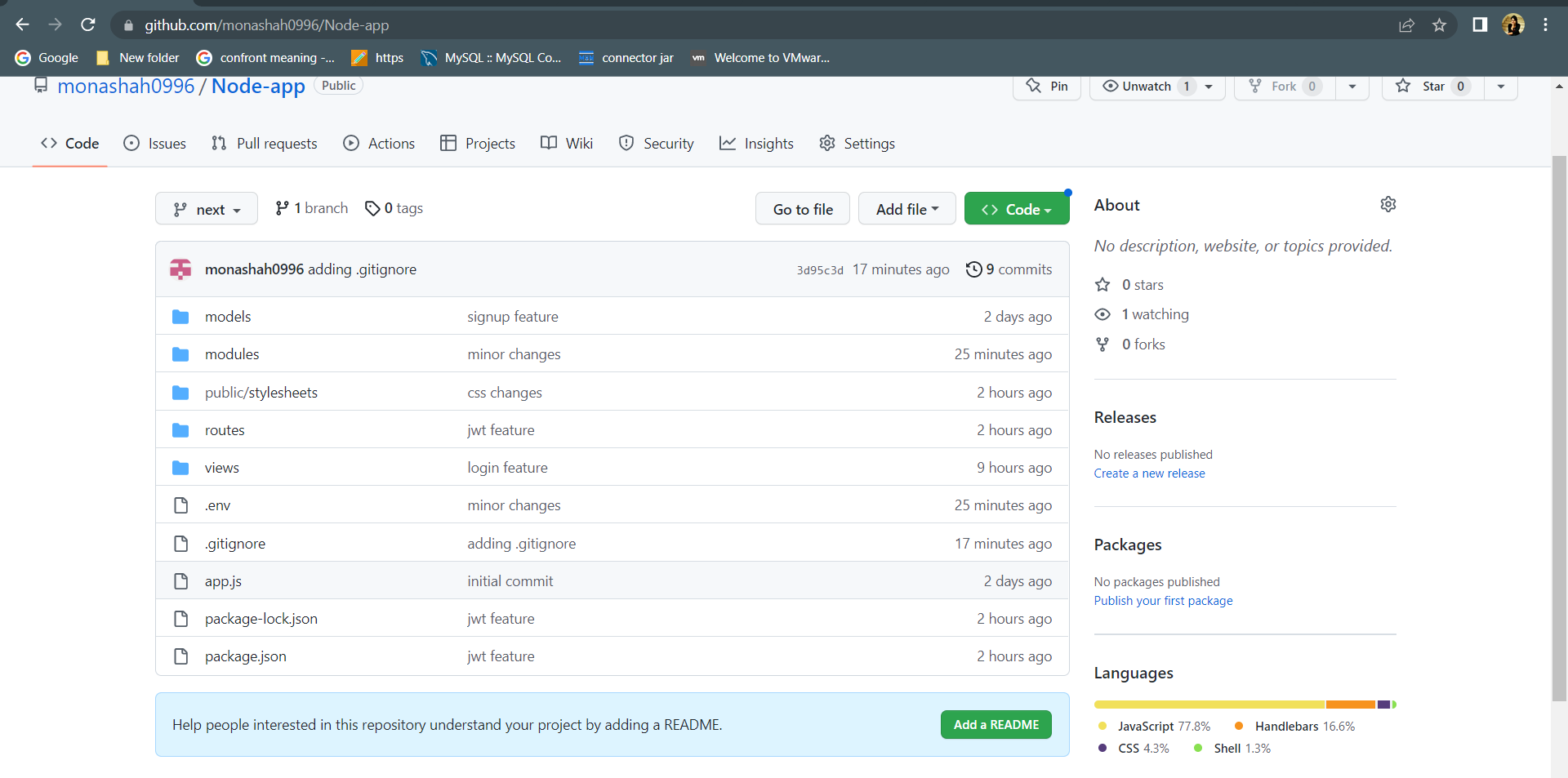


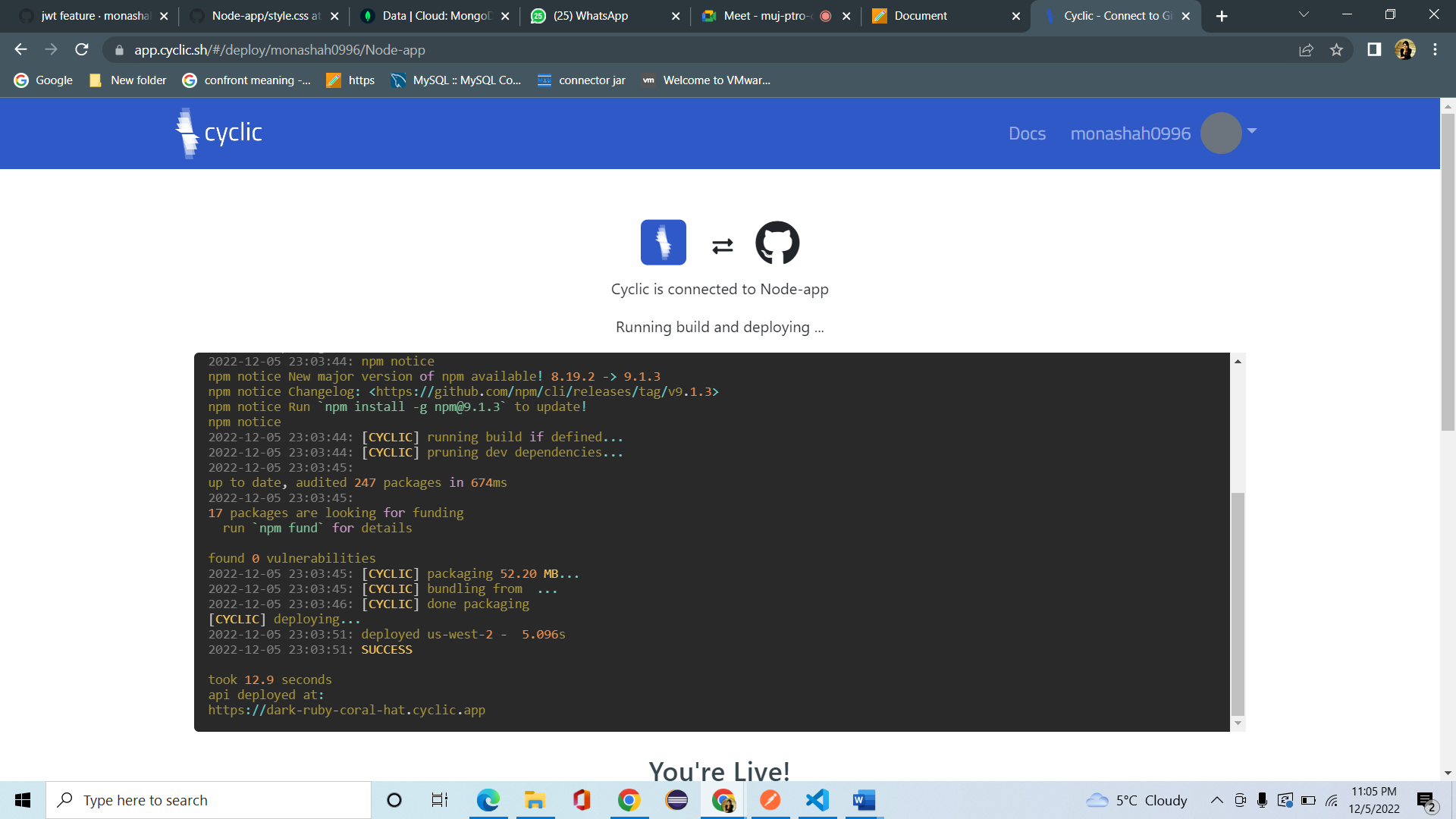


# Question 5:

(Describe the major steps for deployment)

The major steps in deployment are first we are creating a repository in git and pushing all the code using the commit.





Graphical user interface, text

Description automatically generated

# Summary

(Describe how did you divide the work, share your feedback about this project like new points that you learn, challenges, …)

In this project, work is shared equally among each person. Step 1 is done individually and Step 2 the configuration is done by each person individually in their local machines and the routes are shared equally amongst them.

Step 3 the route part and the form is done by Mona and extracting the data and displaying in the table format is done by srikruthi.

Step 4 is the initialization of env file in each of their own local machines.