|  |  |  |
| --- | --- | --- |
|  |  |  |

GOVERNMENT OF INDIA



**MINISTRY OF SKILL DEVELOPMENT & ENTREPRENEURSHIP**

DIRECTORATE GENERAL OF TRAINING

**NATIONAL SKILL TRAINING INSTITUTE**

NSTI (W), Salt Lake, Kolkata-700091

**CERTIFICATE**

This is to certify that following trainees have completed their project titled

**“NSTI Smart Cloud Campus”**

**For IBM Program – IT, Networking and Cloud (Technical Diploma)**

|  |  |
| --- | --- |
| ADIT19AU00030 | Shreya Das |

|  |  |  |  |
| --- | --- | --- | --- |
| Mr. Lingaiah Boddu | Mr. K. L. Kuli | Mr. G. C. Ramamurthy | Mr. Sarbojit Neogi |
| IBM Faculty | ADIT Director | Principal | Section-In-Charge |

**ACKNOWLEDGEMENT**

We would like to express my sincere gratitude to several individuals and organizations for supporting me throughout my diploma study. First, I wish to express my sincere gratitude to my trainer, Mr. Lingaiah Boddu, for his enthusiasm, patience, insightful comments, helpful information, practical advice and unceasing ideas that have helped me tremendously at all times in my study and writing of the project report. His immense knowledge, profound experience and professional expertise in computer science has enabled me to complete the project successfully.

I also wish to express my sincere thanks to the National Skill Training Institute, Salt Lake for accepting me into the diploma program. In addition, I am deeply indebted to the Ministry of Skill Development & Entrepreneurship and IBM for granting me the diploma course. The technical and financial support has enabled me to complete my diploma course studies successfully. Also, I am grateful the Mr. Sarbojit Neogi faculty of NSTI Salt Lake for supporting me for course completion in the specific subject.

I am also grateful to the following NSTI Regional Director Sir Mr. K. L. Kuli and Principal Sir Mr. G. C. Ramamurthy for their consistent support and assistance.

***PROJECT ON Build an Application Using NodeJS Express and MongoDB (My Own Websites)***

*…..Submitted by*

***Shreya Das***

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Table of Contents** | **Page No.** |
| 1 | Chapter 1: Introduction |  |
| 2 | Chapter 2: Services and Tools Required |  |
| 3 | Chapter 3: Project Architecture |  |
| 4 | Chapter 4: Architecture Blocks Detail Working |  |
| 5 | Project Budget |  |
| 6 | Conclusion |  |
| 7 | References |  |
| 8 | Code |  |

**CHAPTER 1**

**INTRODUCTION**

* 1. **Abstract**
  2. **Introduction**
  3. **Purpose**
  4. **Modules**
  5. **Scope**

**Abstract:**

This application is developed using NodeJS and the Express framework, when we want to interact with a MongoDB database from your code. Fortunately, this task is easy to accomplish using the Mongoose package for JavaScript. In this project I’ve shown you how to create an application using NodeJS, Express and MongoDB.

**Introduction:**

This was done to get us ready for whatever facet we chose. A major part of this project is to show what we have learned and put it into practice.

**Purpose:**

The purpose of this document is to describe the functionality and specifications of the design of a web application and to learn how to build a simple web-based application using Node Js and Mongo DB.

Modules:

* Categories
* Writers
* Magazine
* Cart
  + Login
  + Sign Up

**Scopes**

FEASIBILITY STUDY

After identifying the scope of the project, the feasibility study is needed to be carried out. It is basically keeping the following points in mind.

**Building the software for meeting the scope:**

The software has met the scope. As there no data involved in the system, processing on the file, and the behavior of the project is already identified and bundled in quantitative manner. The processing of the software is very simple as it has been designed in Node.Js and it has been well divided into several functions according to the need.

**Technically feasible:**

The software is very much technically feasible. The software is very much concerned with specifying equipment and the software will successfully satisfy almost all the client’s requirements.

The technical need for the system may vary considerably but might include:

a. The facility to produce output in a given time.

b. Response time under certain conditions.

c. Ability to process data at a particular speed. Therefore, the basic input/output of data is identified. So, the project can easily be build up and it will also be technically feasible.

**State of Art:**

The project is very much within the state of art since the project is a WINDOWS based; it uses very modern and common technique. Beside it is very much modern and user friendly. It also works as middle-ware i.e. only in between the user and the admin. So, it is completely a state of art project.

**Financially Feasible:**

The project is very much financially feasible. The implementation and development cost of the software under the reach of any college.

Moreover, it requires some training for the use. So, training cost can be neglected and the resources of the software are very much available. It also reduces the labor and extra cost to be paid for labor. So indeed, it is financially feasible.

**Resources:**

As mentioned earlier that the resources are easily available and the cost of training is almost negligible. Sometimes situations may arise when it may not be so much easy. For a person completely unaware of using a computer system could result in a training cost or for a very small organization the purchase of a computer, installment of the system and other charges may lead to a difficult matter.

**CHAPTER 2:**

**SERVICES AND TOOLS REQUIRED**

**2.1 Services Used**

**2.2 Tools and Software used**

**Services Used:**

NPM packages

**Available Technologies:**

Languages: HTML, CSS, JavaScript.

Runtime: - Node Js

Non-RDBMS: MongoDB

Library/Framework: Express, Mongoose

**Tools Used:**

Editor Used: Visual Studio Code

Operating System: linux ubuntu

Browser: - Chrome

Documentation: - LibreOffice

**Hardware Used:**

Processor: Intel core i3

RAM: 4GB

Hard Disk: 1TB

**CHAPTER 3**

**PROJECT ARCHITECTURE**

**3.1 Architecture**

**USER FRONTEND BACKEND**

v

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| InternetUser | Web design  v | |  | Node.js JavaScript npm Computer Icons Web application, text, logo png |  PNGEgg**NODEJS 14.0**  **Database** |
|  |  |  |  | |

**Chapter 4:**

**Architecture Blocks Detail Working**

|  |  |
| --- | --- |
|  |  |

**DFD (DATA FLOW Diagram):**

**Admin**

**Login/Sign up**

**Cart**

**User**

**DATABASE**

**Chapter 4:**

**Architecture Blocks Detail Working**

**Database (MongoDB)**

**Database Name: - *golpo-khajana***

**Collection Name: - *userCredentials***

**Structure:**

1. \_id: By default, In MongoDB

2. Email: Enter the email\_id of the user.

3. Name: Enter the name of the user.

4. Password: Enter Password

5. Address: Enter the Address of the user

**Project Budget**

Here we don’t need any cost for making this project.

**Conclusion:**

While building an application using NodeJS, Express and MongoDB, it’s important to make sure everything is installed and configured correctly. In this project, I’ve covered those key steps only.

**References:**

**Websites:**

* www.w3schools.com
* www.tutorialspoint.com
* www.youtube.com
* www.github.com
* www.nodejs.org

**Code**

**Index.html**

var express = require('express');

var app = express();

var path = require('path');

// viewed at http://localhost:8000

app.use("/public",express.static("public"));

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

app.use(bodyParser.urlencoded({ extended: true }));

app.get('', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/index.html'));

});

app.get('/bankim', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/bankim.html'));

});

app.get('/bibhuti', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/bibhuti.html'));

});

app.get('/chotogolpo', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/chotogolpo.html'));

});

app.get('/kishore', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/kishore.html'));

});

app.get('/month', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/month.html'));

});

app.get('/suchitra', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/suchitra.html'));

});

app.get('/sarat', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/sarat.html'));

});

app.get('/sunil', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/sunil.html'));

});

app.get('/narayan', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/narayan.html'));

});

app.get('/rohosyo', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/rohosyo.html'));

});

app.get('/samaresh', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/samaresh.html'));

});

app.get('/saptahik', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/saptahik.html'));

});

app.get('/sukumar', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/sukumar.html'));

});

app.get('/uponyash', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/uponyash.html'));

});

app.get('/yearly', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/yearly.html'));

});

app.get('/login-page', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/log\_in.html'));

});

app.get('/sign-page', function(req, res) {

res.sendFile(path.join(\_\_dirname + '/view/sign.html'));

});

const MongoClient = require("mongodb").MongoClient;

// Connection URI

const uri =

"mongodb://127.0.0.1:27017";

// Create a new MongoClient

const mongoClient = new MongoClient(uri, {

useNewUrlParser: true,

useUnifiedTopology: true,

});

async function run() {

try {

// Connect the client to the server

await mongoClient.connect();

// Establish and verify connection

await mongoClient.db("admin").command({ ping: 1 });

console.log("Connected successfully to server");

} finally {

// Ensures that the client will close when you finish/error

await mongoClient.close();

}

}

run().catch(console.dir);

app.post('/sign-up', function(req, res) {

var email = req.body.email;

var password = req.body.password;

var username = req.body.username;

var address = req.body.address;

var signUpDetails = {

email: email,

password: password,

username: username,

address: address

};

mongoClient.connect().then(

function(client){

const db = client.db('golpo-khajana');

const collection = db.collection('userCredentials');

collection.insertOne(signUpDetails).then(

function(result){

console.log(result);

res.sendFile(path.join(\_\_dirname + '/view/index.html'));

}

);

}

);

});

app.listen(8000);

**Style.css**

.header

{

height: 100px;

width:100%;

background-color:#000205;

position: fixed;

top:0;

}

.menu ul

{

list-style: none;

display: inline-grid;

margin:2%;

float:left;

}

.menu a

{

text-decoration: none;

color:white;

}

.menu1

{

display: none;

}

.menu ul li:hover .menu1

{

position:absolute;

display:block;

background-color: #fa9641;

color:white;

border-radius: 10px;

text-align: left;

}

.menu1 ul

{

text-align: center;

margin:10px;

}

.menu1 li

{

border-bottom:1px dotted red;

padding: 10px;

}

.menu1 a:hover

{

color:black;

}

.search-box

{

position: absolute;

border-radius:40px;

transform:translate(-50%,-50%);

top:25%;

right:2%;

height:20px;

background: white;

padding:8px;

align-items: center;

border:1px solid black;

float:right;

}

.search

{

float: left;

background: none;

transition: 4s;

color:black;

outline: none;

border:none;

line-height:15px;

font-size: 15px;

}

.btn

{

color:red;

float:right;

display: flex;

justify-content: center;

text-decoration: none;

align-items: center;

padding: 1px;

}

#cart a

{

color: red;

float: right;

padding:20px;

text-decoration:none;

}

.mobile-icon

{

display: none;

}

/\* .search-box:hover >.search

{

display: block;

}

.search

{

display: none;

}\*/

.section1

{

width: 100%;

height:auto;

background-color: #cc334f;

display:inline-flex;

margin: 5rem 0 0 0;

/\* border:1px solid black;\*/

}

.section1 img

{

width:140px;

height:200px;

margin:2%;

/\* position: ; \*/

}

.section1 p

{

margin:2%;

font-size: 1.8em;

position: static;

}

.section1 a

{

text-decoration: none;

color:white;

float: right;

margin-top:10%;

/\* height: 15%;

width: 15%;\*/

padding: 15px;

background-color:#605d8f;

text-align: center;

border-radius: 10px;

font-size: 15px;

}

.section1 a:hover

{

text-decoration: underline red;

}

.section2

{

width: 100%;

height:auto;

background-color: #c4eb91;

display:inline-flex;

/\* border:1px solid black;\*/

}

.section2 img

{

width:150px;

height:230px;

margin:2%;

/\* position: ; \*/

}

.section2 p

{

margin:2%;

font-size: 1.8em;

position: static;

}

.section2 a

{

text-decoration: none;

color:white;

float: right;

margin-top:10%;

padding: 15px;

/\* height: 20%;

width: 15%;\*/

background-color: #605d8f;

text-align: center;

border-radius: 10px;

font-size: 15px;

}

.section2 a:hover

{

text-decoration: underline red;

}

.section3

{

width: 100%;

height:auto;

background-color: #91dceb;

display:inline-flex;

/\* border:1px solid black; \*/

}

.section3 img

{

width:150px;

height:200px;

margin:2%;

/\* position: ; \*/

}

.section3 p

{

margin:2%;

font-size: 1.8em;

position: static;

}

.section3 a

{

text-decoration: none;

color:white;

float: right;

margin-top:10%;

padding: 15px;

/\* height: 20%;

width: 15%;\*/

background-color: #605d8f;

text-align: center;

border-radius: 10px;

font-size: 15px;

}

.section3 a:hover

{

text-decoration: underline red;

}

.footer

{

background: #fa9641;

display: flex;

height:400px;

width:100%;

box-sizing: border-box;

}

.footer1 img

{

margin-left:100px;

/\* border:1px solid red;\*/

margin-right: 100px;

}

.footer h2

{

text-align: center;

/\* border:1px solid red;\*/

}

.footer a

{

text-decoration: none;

color:white;

background: #605d8f;

padding:8px;

margin: 100px;

border-radius: 10px;

/\* border:1px solid red;\*/

}

.footer a:hover

{

text-decoration: underline red;

}

#head

{

background: red;

}

.footer

{

height:360px

}

.footer1

{

margin: 1%, 5%,1%,5%;

}

.footer1 img

{

width:50%;

height: 70%;

/\* padding-top:5%;

padding-left: 10%;\*/

}

.footer1 a

{

padding: 1%;

}

.section3 a

{

padding: 10px;

}

.section2 a

{

padding: 10px;

}

.section1 a

{

padding: 10px;

}

@media only screen and (max-width: 700px)

{

.menu ul

{

padding: 0px;

}

.footer

{height:300px;

width:100%;

}

.footer1 img

{

width:70%;

height: 50%;

margin: 5%;

align-content: center;

}

.footer1 a

{

margin: 1%;

padding: 5px;

font-size: 10px;

}

.footer1 h2

{

margin-left:5%;

font-size: 20px;

}

.search-box

{

padding:5px;

}

.search-box:hover >.search

{

display: block;

}

.search

{

display: none;

}

#cart

{

padding-right: 10%;

}

}

@media only screen and (max-width: 1000px)

{

.menu ul

{

padding: 0px;

}

**Login.html**

<!DOCTYPE html>

<html>

<head>

<title>

log in

</title>

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">

<style>

body {

background-image: url("public/photos/books.jpg");

background-repeat: no-repeat;

background-size: cover;

width: 99%;

}

.input {

top: 110px;

position: absolute;

transition: 6s;

margin-left: 40px;

}

input[type=text],

input[type=password] {

width: 240px;

height: 35px;

padding: 12px 12px;

margin: 10px 0px 0px 15px;

display: inline-block;

border: 1px solid #ccc;

border-radius: 5px;

box-sizing: border-box;

background: #fff;

transition: 1s;

color: gray;

font-family: Arial;

}

.a {

padding: 9px;

border: 1px solid black;

}

.b {

text-decoration: none;

color: white;

}

</style>

</head>

<body>

<center>

<form id="input">

<div>

<i class="fa fa-user-circle fa-3x" aria-hidden="true"></i>

<br>

<legend>

<font size="5px">Log in </font>

</legend>

<hr width="30%">

</div>

<p> New in this site?&nbsp<a href="/sign-page"> Sign Up</a></p>

<hr width="30%">

<p><input type="text" placeholder="Username"></p>

<p><input type="password" placeholder="Password"></p>

<p><button class="a" type="submit" value="Log In">Log in</button>

<button class="a" type="reset" value="Reset">Reset</button>

</p>

<table>

<tr>

<td>

<table class="a" ; bgcolor=#3a579d>

<tr>

<td>

<img src="/public/photos/f.png" height=25px> </img>

</td>&nbsp

<td><a class="b" href="https://www.facebook.com">Log in with Facebook </a></td>

</tr>

</table>

</td>

<td>

<table class="a" ; bgcolor="#dd4b3c">

<tr>

<td>

<img src="public/photo/g.png" height=25px></img>

</td> &nbsp

<td><a class="b" href="https://www.google.com">Log in with Google </a></td>

</tr>

</table>

</td>

</tr>

</table>

</form>

<!-- <table>

<tr>

<td>

<table class="a"; bgcolor=#3a579d>

<tr> <td>

<img src="photo/f.png" height=25px> </img></td>&nbsp

<td><a class="b" href="https://www.facebook.com">Log in with Facebook </a></td>

</tr></table></td>

<td>

<table class="a"; bgcolor="#dd4b3c">

<tr>

<td>

<img src="photo/g.png" height=25px></img></td> &nbsp

<td><a class="b" href="https://www.google.com">Log in with Google </a></td></tr></table></td></tr></table> -->

</center>

</body>

</html>

**SignUp.html**

<!DOCTYPE html>

<html>

<head>

<title>sign</title>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">

<style>

body {

background-image: url("public/photos/books.jpeg");

background-repeat: no-repeat;

background-size: cover;

width: 99%;

}

.menu {

width: 400px;

height: 450px;

position: relative;

margin: 130px 0px 0px 500px;

background: linear-gradient(to top, rgb(56, 59, 224), rgb(226, 91, 118));

opacity: 0.9;

padding: 5px;

}

.sign\_up {

top: 110px;

position: absolute;

transition: 6s;

margin-left: 40px;

}

input[type=text],

input[type=password],

input[type=password],

input[type=name],

input[type=address] {

width: 240px;

height: 35px;

padding: 12px 12px;

margin: 10px 0px 0px 15px;

display: inline-block;

border: 1px solid #ccc;

border-radius: 5px;

box-sizing: border-box;

background: #fff;

transition: 1s;

color: gray;

font-family: Arial;

}

button {

background-color: #f04187;

color: white;

padding: 14px 20px;

margin: 25px 0px 0px 15px;

border: none;

cursor: pointer;

width: 280px;

border-radius: 5px;

font-size: 15px;

font-family: sans-serif;

}

button:hover {

opacity: 0.8;

}

</style>

</head>

<body>

<div class="menu">

<center>

<form id="sign\_up" action = "/sign-up" method = "post">

<div> <i class="fa fa-user-plus fa-3x" aria-hidden="true"></i><br>

<legend>

<font size="5px">Sign Up </font>

</legend>

<hr width="30%">

</div>

<p> Already exist?&nbsp<a href="login.html"> Sign In</a></p>

<hr width="30%">

<input type="text" name="email" placeholder="Enter your email" required>

<br>

<input type="password" name="password" placeholder="Enter Password" required>

<br>

<!-- <input type="password" name="conp" placeholder="Confirm Password" required>

<br> -->

<input type="name" name="username" placeholder="Enter Name" required>

<br>

<input type="address" name="address" placeholder="Enter Address" required>

<br>

<button type="submit" class="btn1">Create Account</button>

<br>

</form>

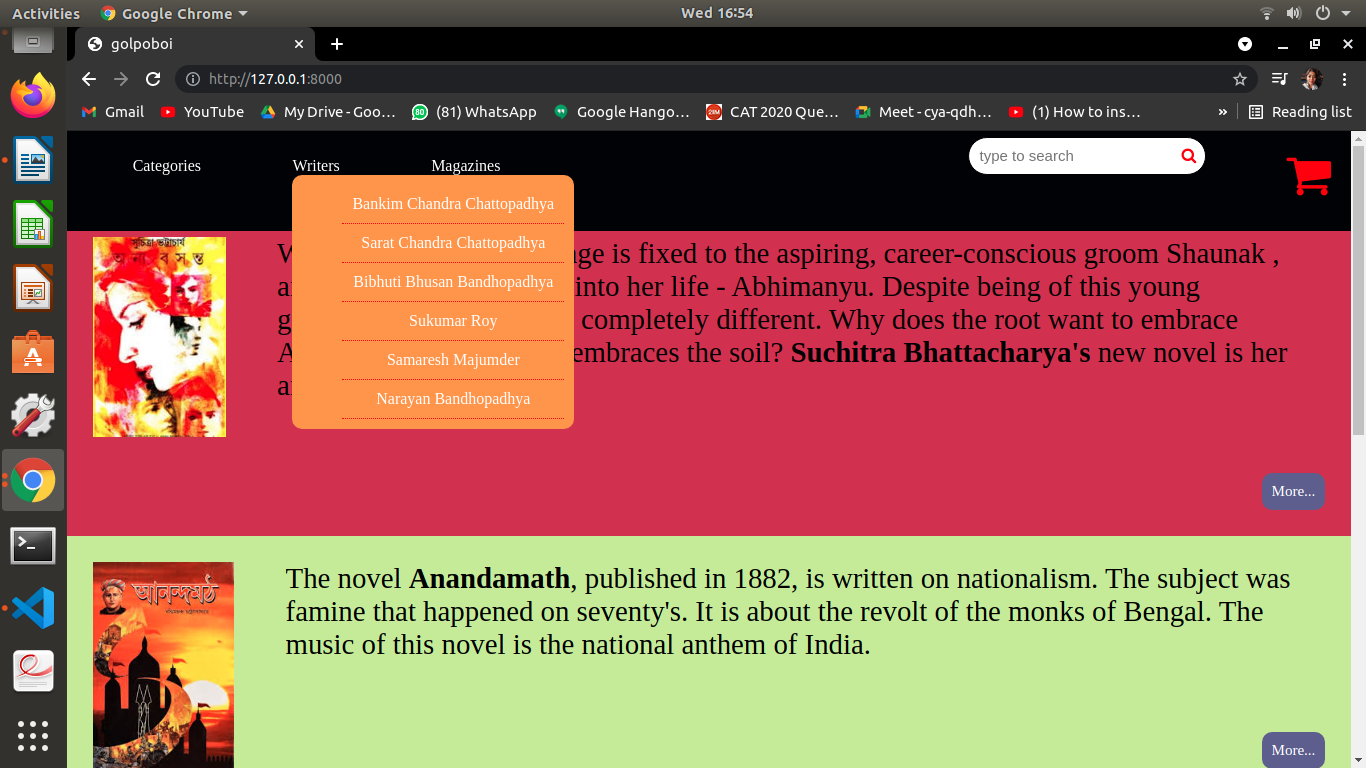
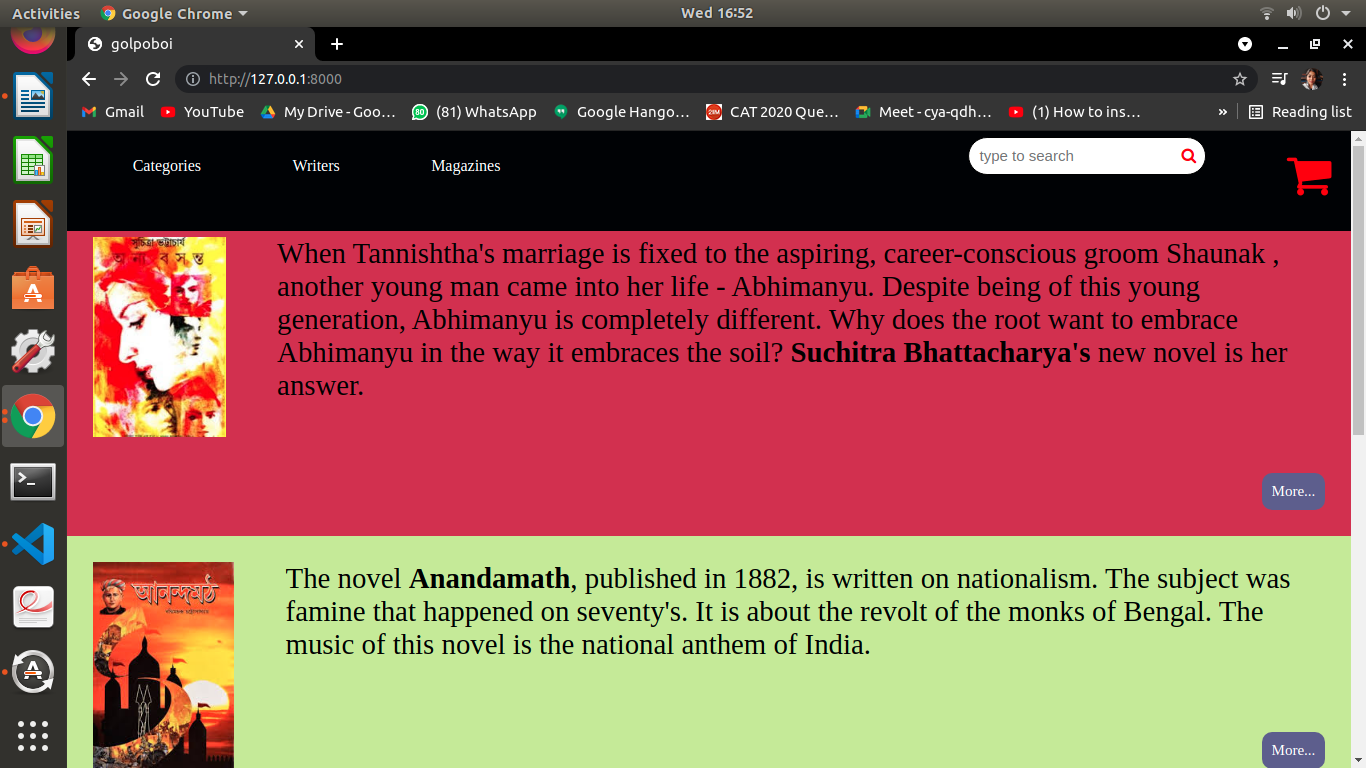
</div>

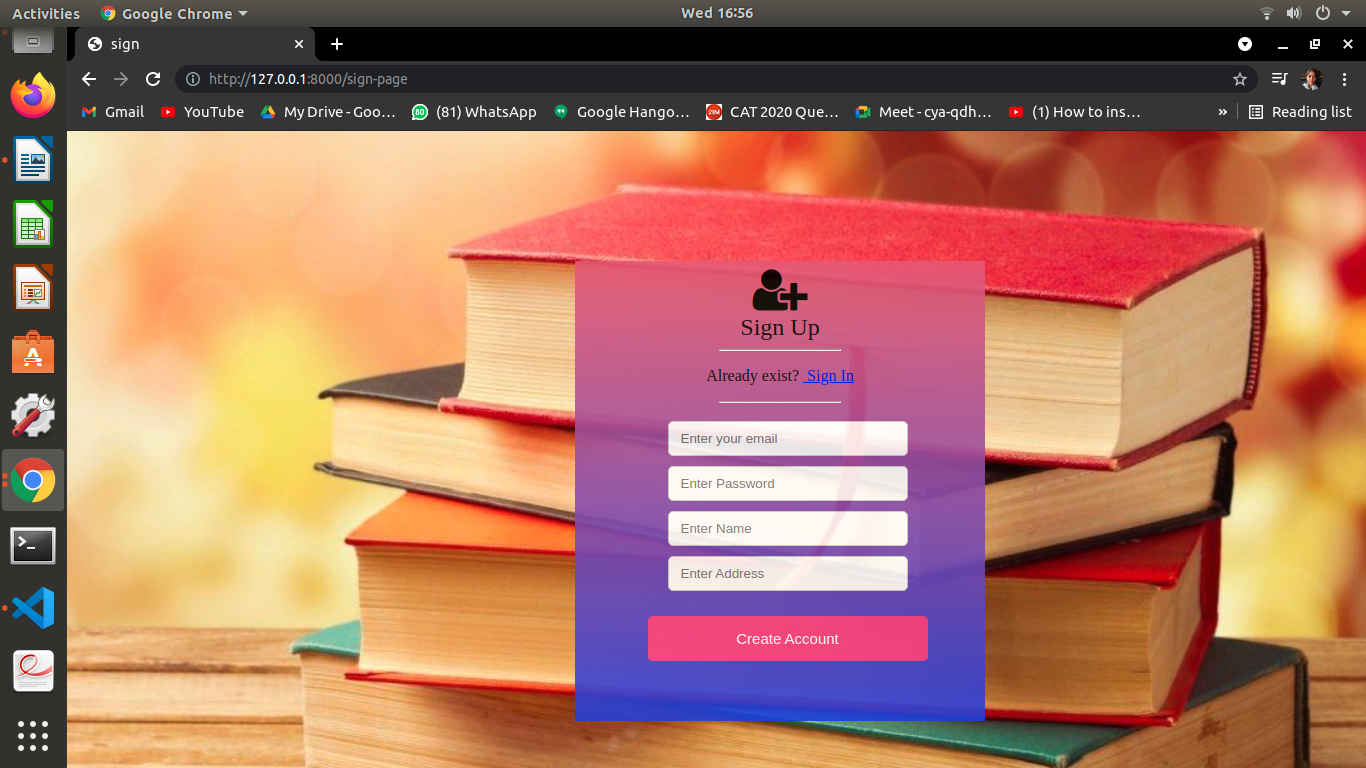
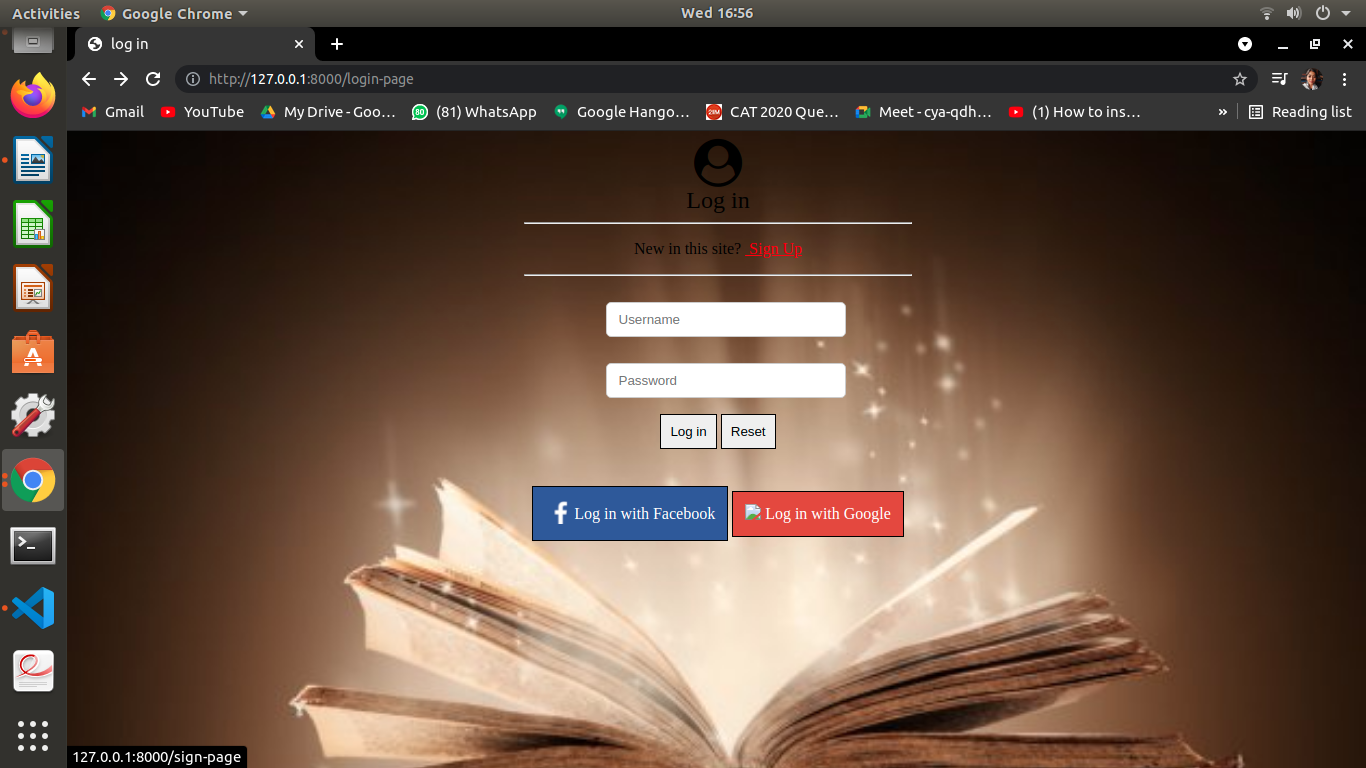
</center>

</body>

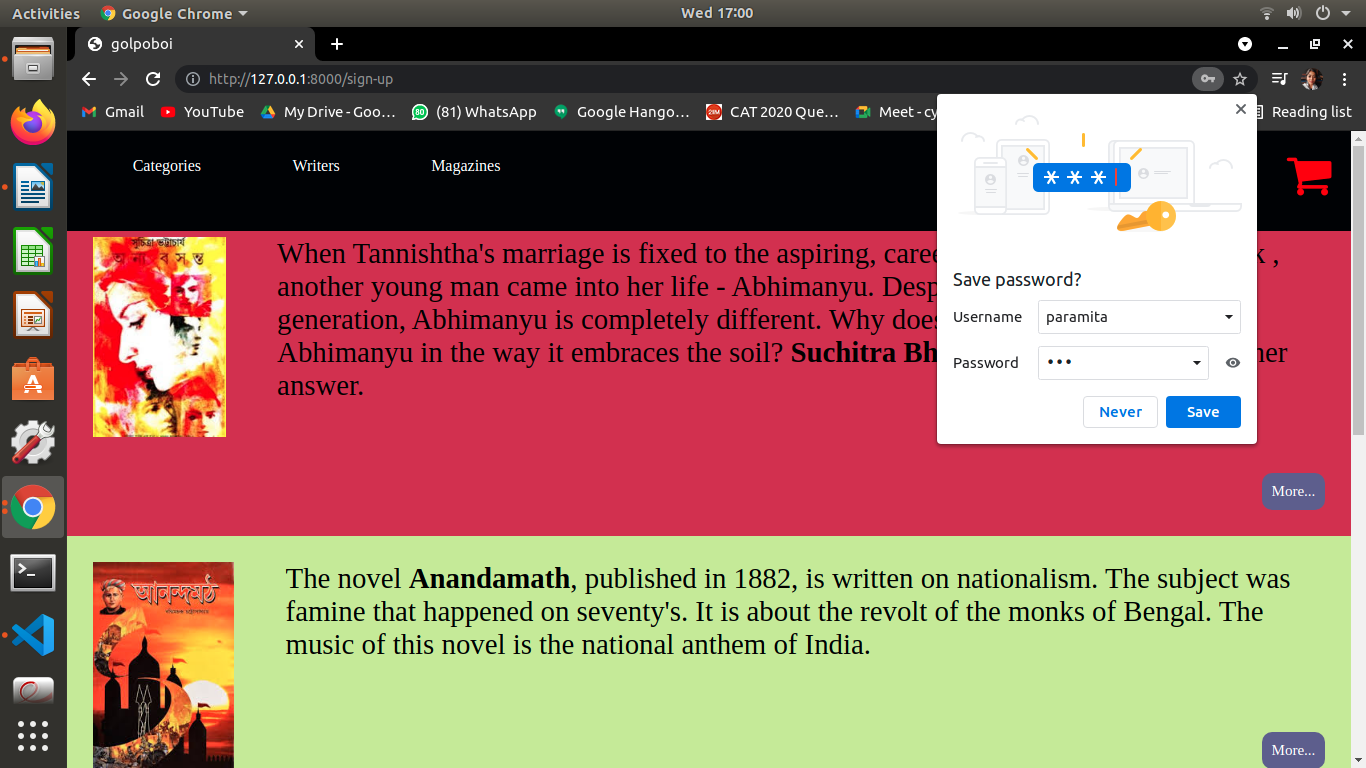
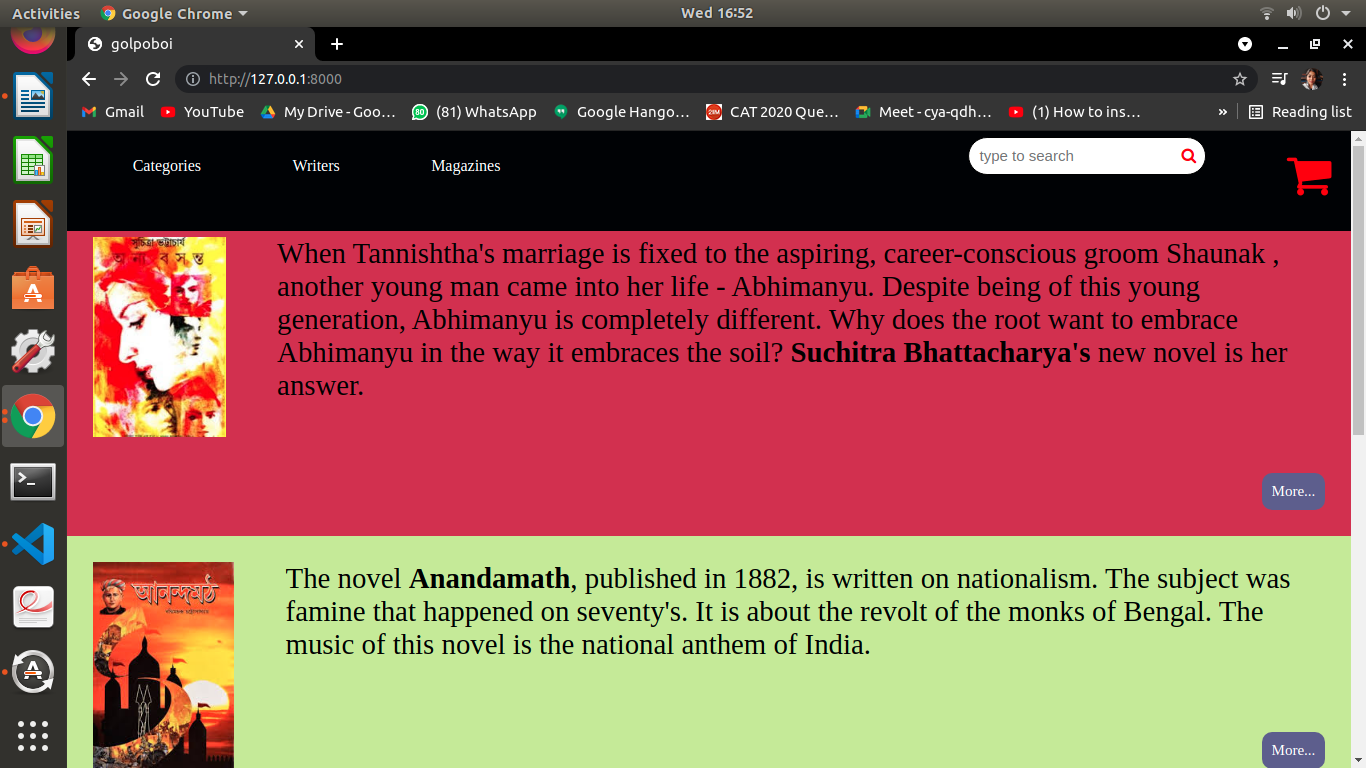
</html>

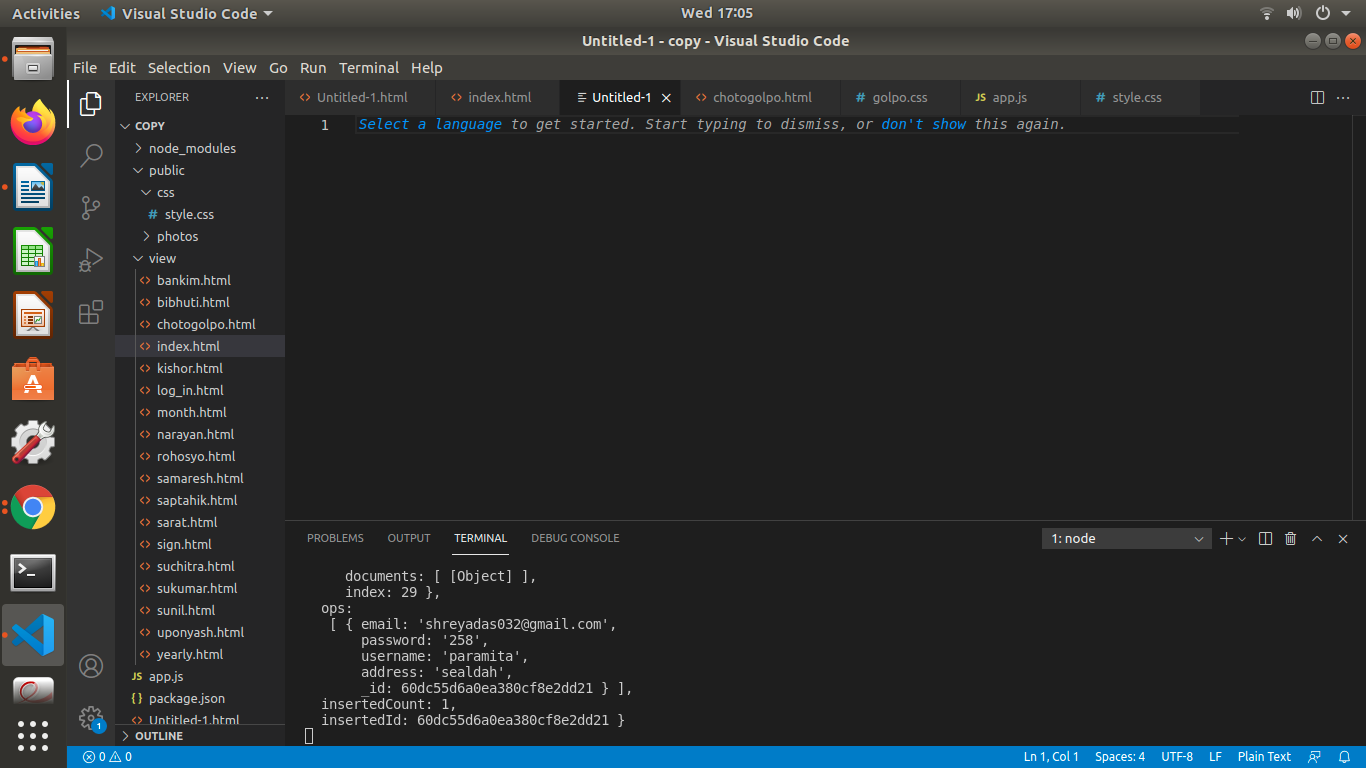
**Screenshots:**

****

****

After registration successfully the home page will be redirected.

****

****

***Thank You***