WEB DEVELOPMENT LAB-2 (SMALL PROJECT)

PROJECT TITLE

MEDI-CARE
COVID ESSENTIALS

SUBMISSION BY : SEEMA BHARTI
SCHOLAR ID : 212120057
SEMESTER : SECOND SEM

PLATFORM USED FOR PROJECT

FOR FRONT-END DESIGN

REACT-JS

FOR BACKEND DESIGN

mongo DB

WHAT IS REACT

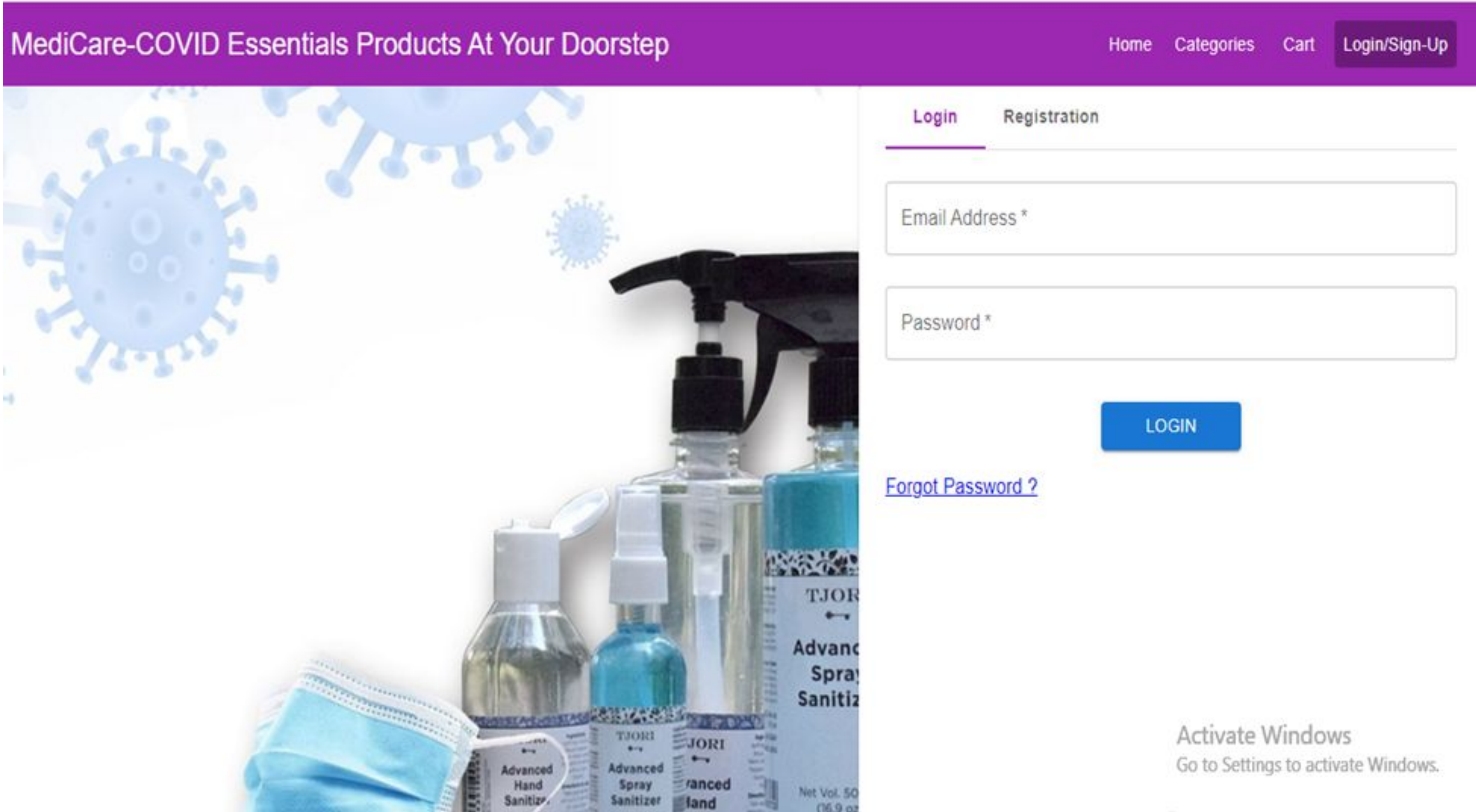
1. React is a JavaScript library for building user interfaces.
2. React is used to build single-page applications.
3. React allows us to create reusable UI components.
4. React creates a VIRTUAL DOM in memory
5. The ReactDOM.render() function takes two arguments, HTML code and an HTML element.
6. Components are independent and reusable bits of code. They serve the same purpose as JavaScript functions, but work in isolation and return HTML
7. Props are arguments passed into React components

PROJECT PROPOSAL

1) DESIGN LAYOUT(FRONT END DESIGN)
2) HOW TO IMPLEMENT VARIOUS FEATURES LIKE
   - LOGIN LAYOUT(AUTHENTIC USER)
   - CART LAYOUT
   - SERVICES
   - TOTAL VISITORS /FEEDBACK
3) TIMELINE

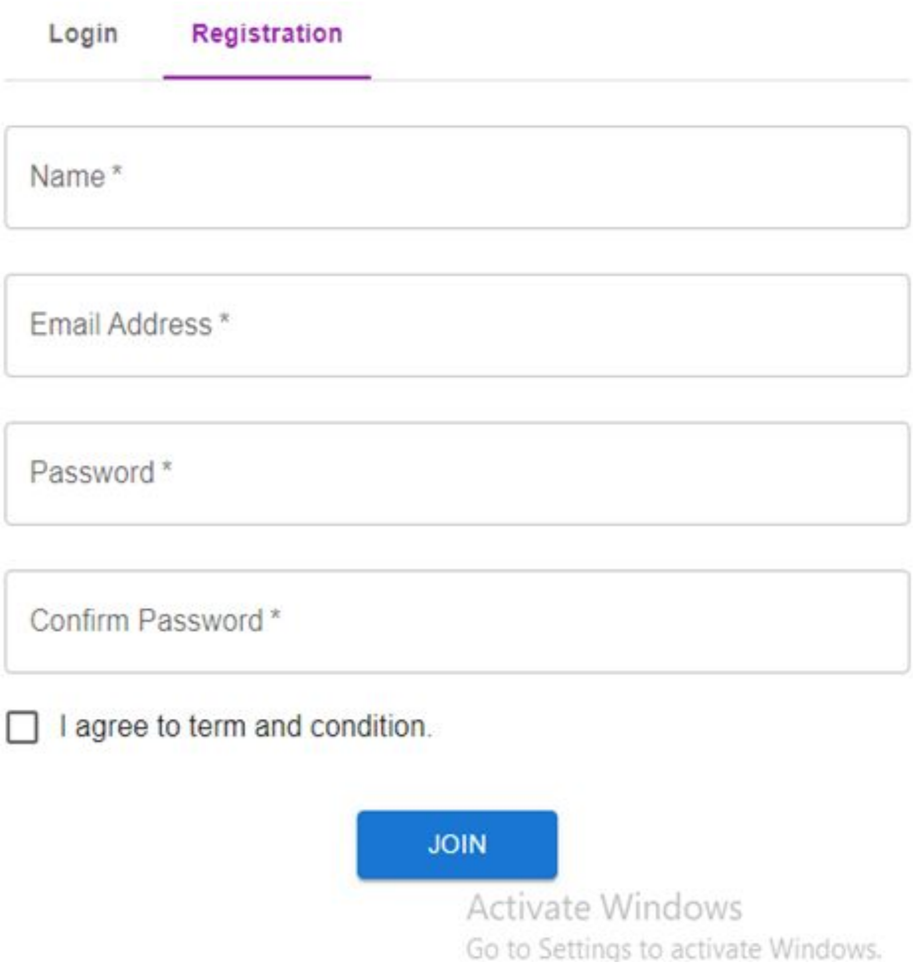
Components

- Home
- Navbar
- Changepasswords
- Loginregistration
- Registration
- Reset password
- Send password
- User login



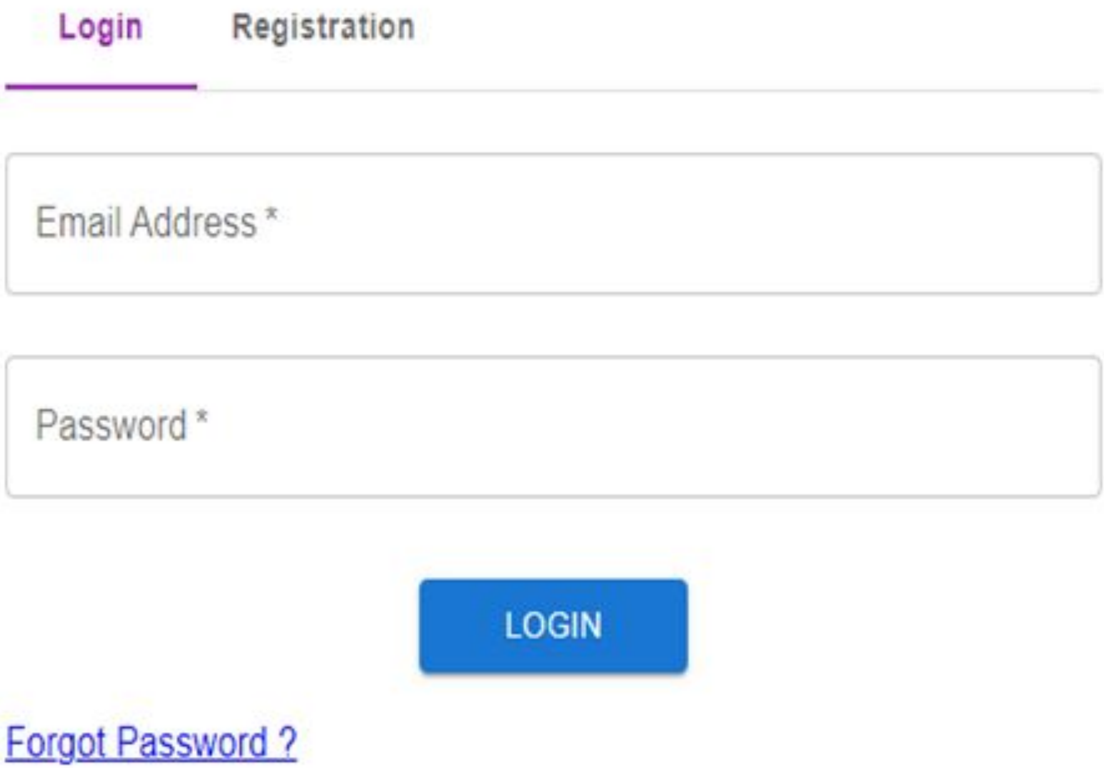
MediCare-COVID Essentials Products At Your Doorstep

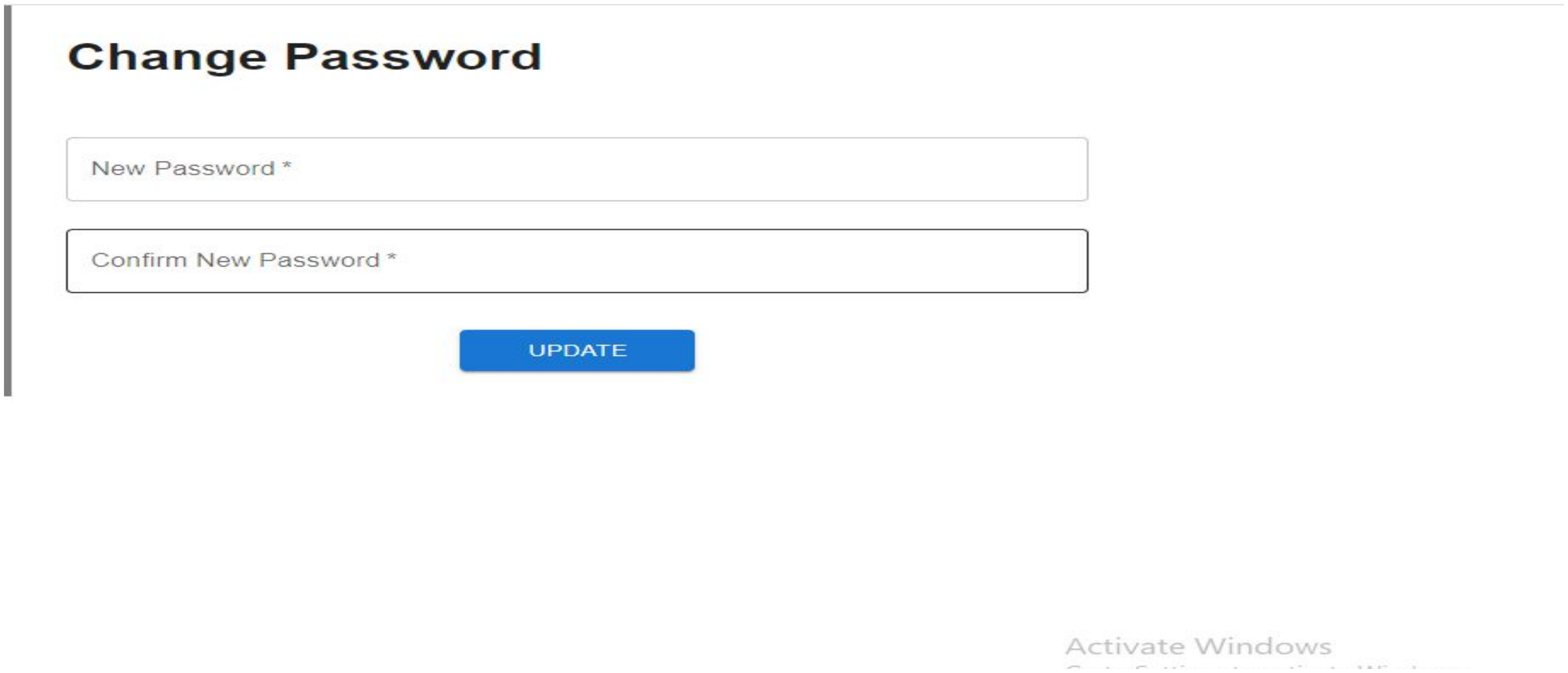
SIGN UP PAGE

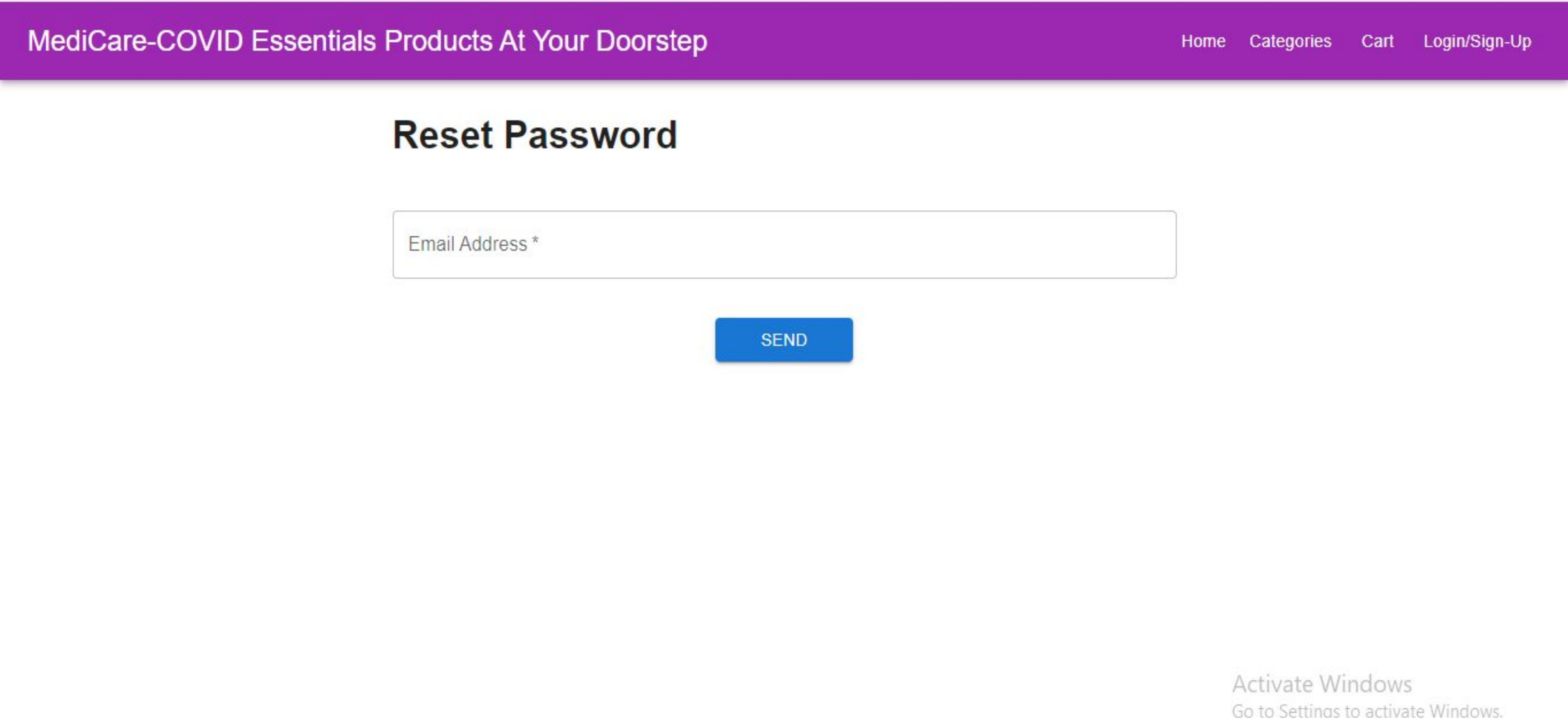


□ I agree to term and condition.

LOGIN PAGE


Login
Registration
Email Address *
Password *
LOGIN
Forgot Password ?


Change Password
New Password *
Confirm New Password *
UPDATE
Activate Windows



Reset Password

Email Address *

WHAT IS ROUTER DOM IN REACT

Create React App doesn't include page routing

An application can have multiple <Routes>

The first <Route> has a path of / and renders the Layout component

The nested <Route>s inherit and add to the parent route. So the blogs path is combined with the parent and becomes /blogs.

The Home component route does not have a path but has an index attribute. That specifies this route as the default route for the parent route, which is /.

App.js

```
function App() {
        return (
        <>
       <BrowserRouter>
       <Routes>
         <Route path="/" element={<Layout />}>
           <Route index element={<Home />} />
           <Route path="contact" element={<Contact />} />
           <Route path="login" element={<LoginReg />} /></Route>
        </BrowserRouter> </>);}
         export default App
```

Navbar.js

```
const Navbar = () => {
 return <>
   <Box sx={{ flexGrow: 1 }}>
     <AppBar position="static" color="secondary">
       <Toolbar>
         <Typography variant='h5' component="div" sx={{ flexGrow: 1 }}>MediCare-COVID Essentials Products At Your
Doorstep</Typography>
 <Button component={NavLink} to='/' style={({ isActive }) => { return { backgroundColor: isActive ? '#6d1b7b' : '' } }} sx={{
color: 'white', textTransform: 'none' }}>Home</Button>
<Button component={NavLink} to='/contact' style={({ isActive }) => { return { backgroundColor: isActive ? '#6d1b7b' : '' } }}
sx={{ color: 'white', textTransform: 'none' }}>Categories</Button>
  <Button component={NavLink} to='/' style={({ isActive }) => { return { backgroundColor: isActive ? '#6d1b7b' : '' } }} sx={{
color: 'white', textTransform: 'none' }}>Cart</Button>
<Button component={NavLink} to='/login' style={({ isActive }) => { return { backgroundColor: isActive ? '#6d1b7b' : '' } }} sx={{
color: 'white', textTransform: 'none' }}>Login/Sign-Up</Button></Toolbar>
     </AppBar> </Box></>;};
export default Navbar;
```

Registration

```
import { TextField, FormControlLabel, Checkbox, Button, Box, Alert } from '@mui/material';
import { useState } from 'react';
import { useNavigate } from 'react-router-dom';

const Registration = () => {
  const [error, setError] = useState({
    status: false,
    msg: "",
    type: ""
  })
  const navigate = useNavigate();
  const handleSubmit = (e) => {
    e.preventDefault();
    const data = new FormData(e.currentTarget);
    const actualData = {
      name: data.get('name'),
      email: data.get('email'),
      password: data.get('password'),
      password_confirmation: data.get('password_confirmation'),
      tc: data.get('tc'),
}
    if (actualData.name && actualData.email && actualData.password &&
actualData.password_confirmation && actualData.tc !== null) {
      if (actualData.password === actualData.password_confirmation) {
        console.log(actualData);
        document.getElementById('registration-form').reset()
        setError({ status: true, msg: "Registration Successful", type: 'success' })
        navigate('/dashboard')
      } else {
        setError({ status: true, msg: "Password and Confirm Password Doesn't Match", type:
'error' })
      }
    } else {
      setError({ status: true, msg: "All Fields are Required", type: 'error' })
    }
  }
```

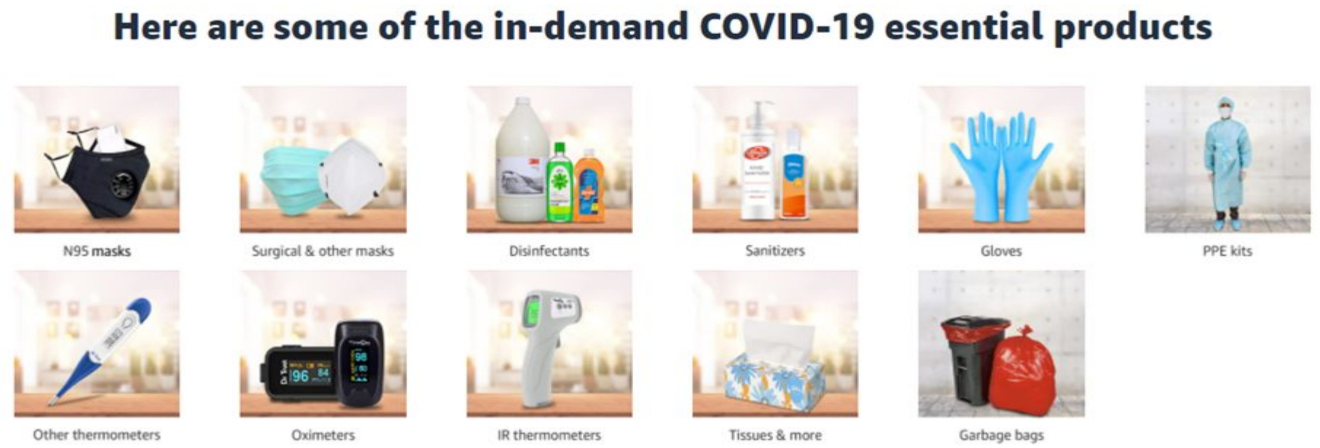
DESIGN LAYOUT OF CATEGORIES

WEBSITE_NAME CART LOGIN-PART

FEATURES

CATEROGIES

- PPE KITS
- HAND-GLOVES
- SANITIZERS
- FACE-MASKS
- FACE -SHIED



Here are some of the in-demand COVID-19 essential products

N95 masks

Surgical & other masks

Disinfectants

Sanitizers

Gloves

PPE kits

Other thermometers

Oximeters

IR thermometers

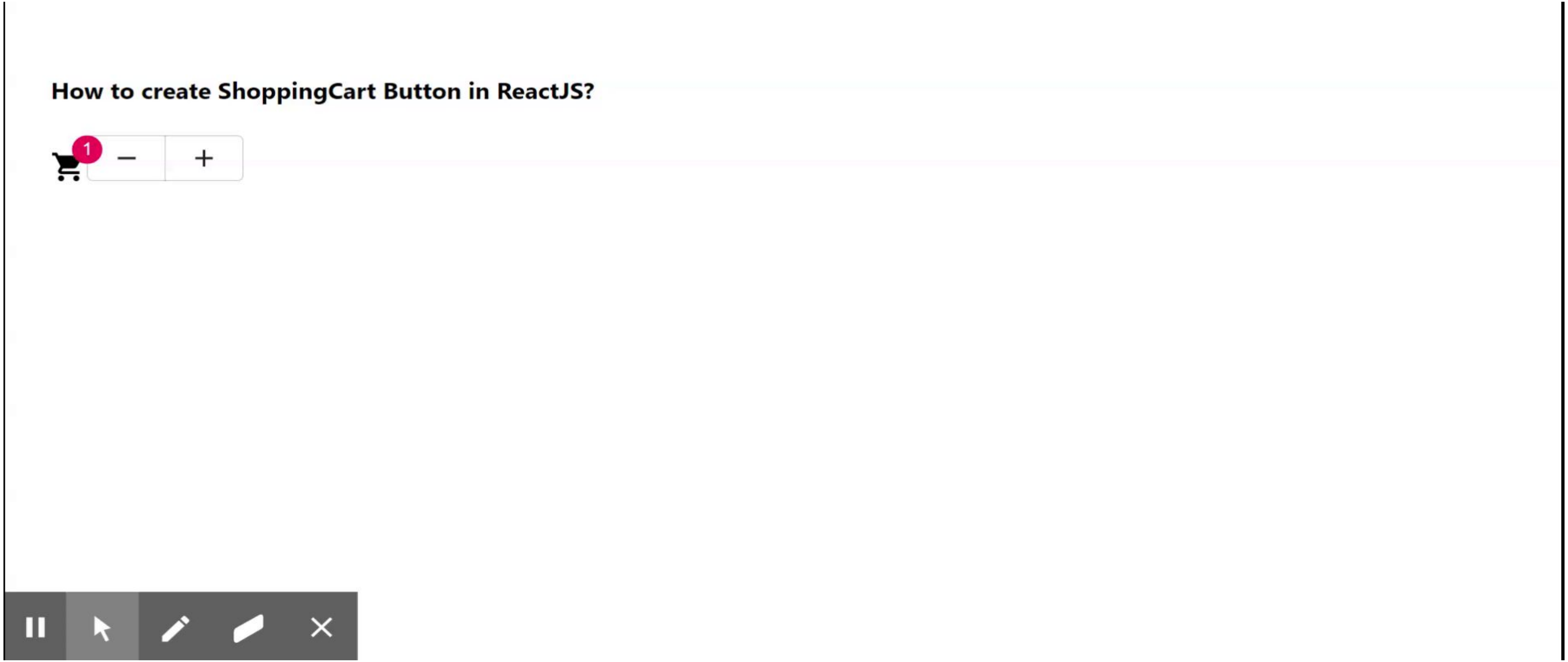
Tissues & more

Garbage bags

TOTAL-VISITORS

COMMENTS/RATING

CART DESIGN LAYOUT



How to create ShoppingCart Button in ReactJS?

Introduction to MongoDB

MongoDB is a cross-platform, document oriented database that provides, high performance, high availability, and easy scalability. MongoDB works on concept of collection and document.

Document

A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data

```
{


 _id: ObjectId(7df78ad8902c)
  title: 'MongoDB Overview',
  description: 'MongoDB is no sql database',
  by: 'tutorials point',
  url: 'http://www.tutorialspoint.com',
  tags: ['mongodb', 'database', 'NoSQL'],
  likes: 100,
  comments: [
     {
        user:'user1',
        message: 'My first comment',
        dateCreated: new Date(2011,1,20,2,15),
        like: 0
     },
     {
        user:'user2',
        message: 'My second comments',
        dateCreated: new Date(2011,1,25,7,45),
        like: 5
     }
  ]
}
```

```
var express=require("express");
var bodyParser=require("body-parser");

const mongoose = require('mongoose');
mongoose.connect('mongodb://localhost:27017/gfg');
var db=mongoose.connection;
db.on('error', console.log.bind(console, "connection error"));
db.once('open', function(callback){
    console.log("connection succeeded");
})
var app=express()


app.use(bodyParser.json());
app.use(express.static('public'));
app.use(bodyParser.urlencoded({
    extended: true
}));

app.post('/sign_up', function(req,res){
    var name = req.body.name;
    var email =req.body.email;
    var pass = req.body.password;
    var phone =req.body.phone;
var data = {
                               "name": name,
                               "email":email,
                             "password":pass,
                               "phone":phone
                                    }
       db.collection('details').insertOne(data,function(err, collection){
                              if (err) throw err;
                 console.log("Record inserted Successfully");

                                    });

                 return res.redirect('signup_success.html');
                                   } )
```

```
app.get('/',function(req,res){
res.set({
'Access-control-Allow-Origin': '*'
});
return res.redirect('index.html');
}).listen(3000)


console.log("server listening at port 3000");
```

TIMELINE

FEBRUARY - LEARN CONCEPT OF REACT-JS (component,props etc)

- DESIGN FRONT-END LAYOUT
- LOGIN PAGE (Sign-in/ Sign-out)

MARCH - SHOPPING CART

- FEATURE (high price/low price)
- COMMENT SECTION

APRIL - CONNECT DATABASE TO mongoDB

- FINAL PROJECT

THANK YOU!