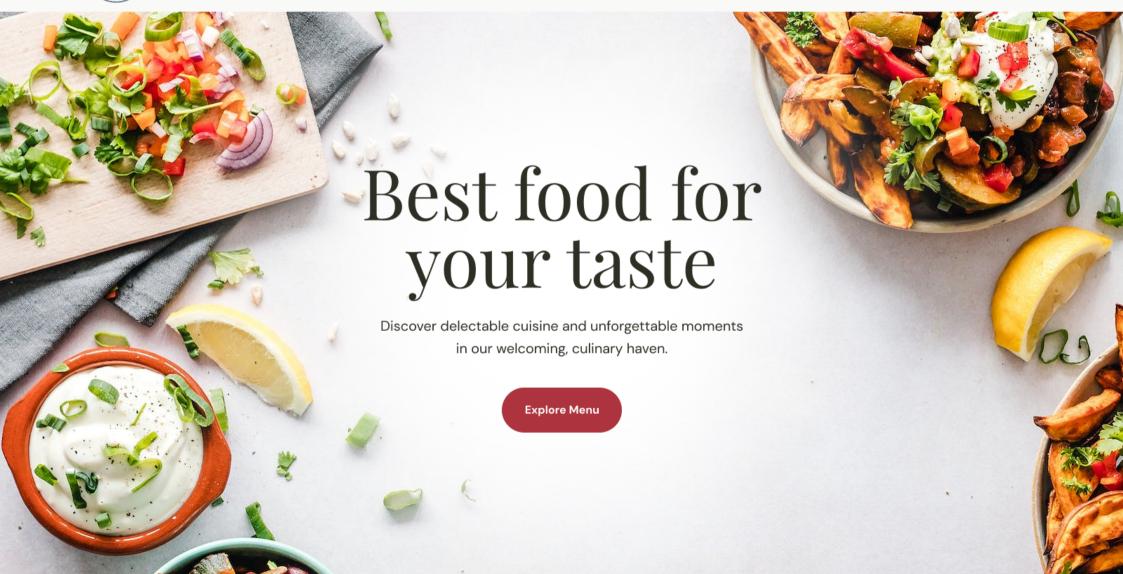


Home

About

Mer

Contact



A little information for our valuable guest

At place, we believe that dining is not just about food, but also about the overall experience. Our staff, renowned for their warmth and dedication, strives to make every visit an unforgettable event.

2

Locations

2004

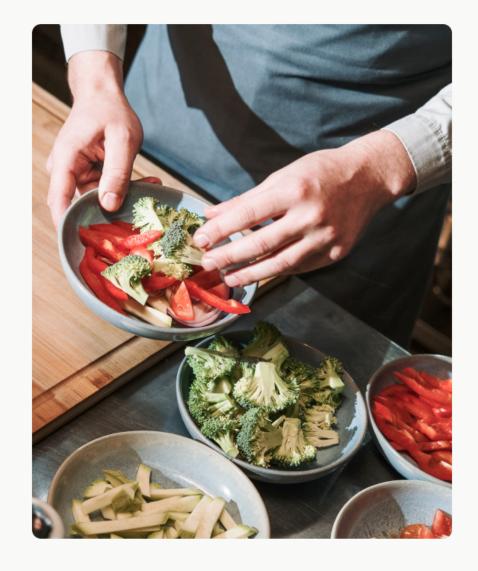
Founded

30+

Staff Members

100%

Satisfied Customers



Home

Menu

About

Contact

Weekly Manu

Our Menu

We consider all the drivers of change gives you the components you need to change to create a truly happens.

All

Breakfast

Main Dishes

Drinks

Desserts



\$ 9.99

Dosa

Mysore's Special



\$15.99

Vada Pav

Bombay's Special



\$7.25

Poha

Nagpur's Special



\$20.99

Misal

Kolhapur's Special



\$5.89

Sandwich

Italian Food



\$18.05

Coffee

Refreshment



\$12.55

Cheese Burger

Italian Food



\$12.99

Noodles

Chinese Food



Home

About

Menu

Contact

Weekly Menu

Contact Us

We consider all the drivers of change gives you the components you need to change to create a truly happens.

Name	Email	
Enter your name	Enter email address	
Subject		
Write a subject		
Message		
Write your message		
	Send	

Call Us:

Hours:

Our Location:

+91-9309435757

Mon-Fri: 7am - 9pm Sat, Sun: 7am - 2pm PICT Campus, Dhankawadi, Katraj

```
import React, { useEffect, useState } from 'react';
       import { Button, InputGroup, Card, Navbar } from '@blueprintjs/core';
 2
       import axios from 'axios':
       import ItemCard from './ItemCard';
       import PlaceOrder from './PlaceOrder';
 6
       import NavBarUser from './NavBarUser';
       import Access from '../Authorization/Access';
       import { Link } from 'react-router-dom';
 8
 9
       export default function Menu({ match }) {
10 🗸
11
         const [menu, setMenu] = useState([]);
         const [items, setItems] = useState([]);
12
13
         const [rules, setRules] = useState({ page: false, op: [] });
14
         const [show, setShow] = useState(false);
15
         const cid = match.params.id;
16
17
         useEffect(() => {
           const token = localStorage.getItem('token');
18
           var config = {
19
             headers: { Authorization: 'Bearer ' + token },
20
21
           };
           axios.get(`http://localhost:5000/role/`, config).then((res) => {
22
23
             axios
24
                .get(`http://localhost:5000/role/permissions/`, {
25
                 params: { role: res.data.role },
26
               })
               .then((res) => {
27
                 console.log(res.data);
28
29
                 const rul = Access('user_menu', true, res.data);
                 console.log(rul);
30
```

1

```
axios.get(`http://localhost:5000/role/`, config).then((res) => {
    axios
      .get(`http://localhost:5000/role/permissions/`, {
        params: { role: res.data.role },
      .then((res) \Rightarrow {
        console.log(res.data);
        const rul = Access('user_menu', true, res.data);
        console.log(rul);
        setRules(rul);
        if (rul) {
          setShow(true);
      });
  });
  fetchData();
}, [items]);
const fetchData = () => {
  const token = localStorage.getItem('token');
  var config = {
    headers: { Authorization: 'Bearer ' + token },
  };
  axios.get(`http://localhost:5000/menu/${cid}`, config).then((res) => {
    const menu_data = res.data.menu;
    console.log(menu_data);
    //res.data.menu[3].map(m => console.log(m));
    setMenu(menu data);
 });
};
const handleOrders = (i) => {
```

```
const handleDelete = (e) => {
  var buffer = [...items];
 var res = [];
  var j = 0;
  for (var i in buffer) {
   if (buffer[i].item id == e.item id && j == 0) {
     j++;
      continue;
    res.push(buffer[i]);
  setItems(res);
};
const handleClear = () => {
  setItems([]);
};
if (show) {
  return (
    <div>
      {rules.op['user_menu'].includes('view_only') ? (
        <NavBarUser view={false} />
      ):(
        <NavBarUser view={true} />
      )}
      <div
        style={{
         marginTop: '10vh',
         display: 'grid',
         gridTemplateColumns: '2fr 1fr',
         gridGap: '50px',
```

```
{rules.op['user_menu'].includes('place_order') ? (
         <PlaceOrder
           items={items}
           handleDelete={handleDelete}
           handleClear={handleClear}
           style={{
             gridColumn: '2/3',
             textAlign: 'center',
             overflowY: 'scroll',
             height: '90vh',
           }}
         />
       ):(
         <div></div>
       )}
     </div>
   </div>
 );
} else {
 return <div></div>;
```