

---

# Desenvolvimento WEB

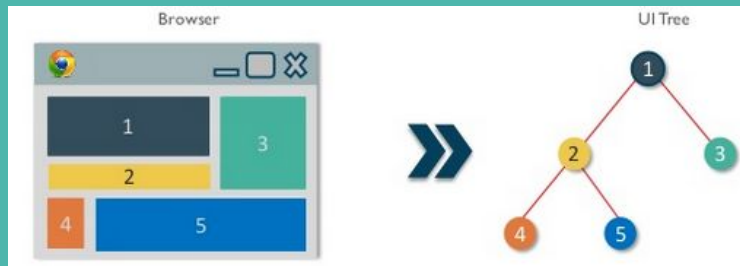
-

Full Stack Completo: Java + React

---

# Iniciando a Implementação do Projeto Front-end

React JS



# Entendendo a Estrutura Inicial do Projeto

**package.json** => arquivo que exhibe as dependências do projeto e as respectivas versões

Visual Studio Code interface showing the `package.json` file for the project `oxefood-web`. The file contains the following content:

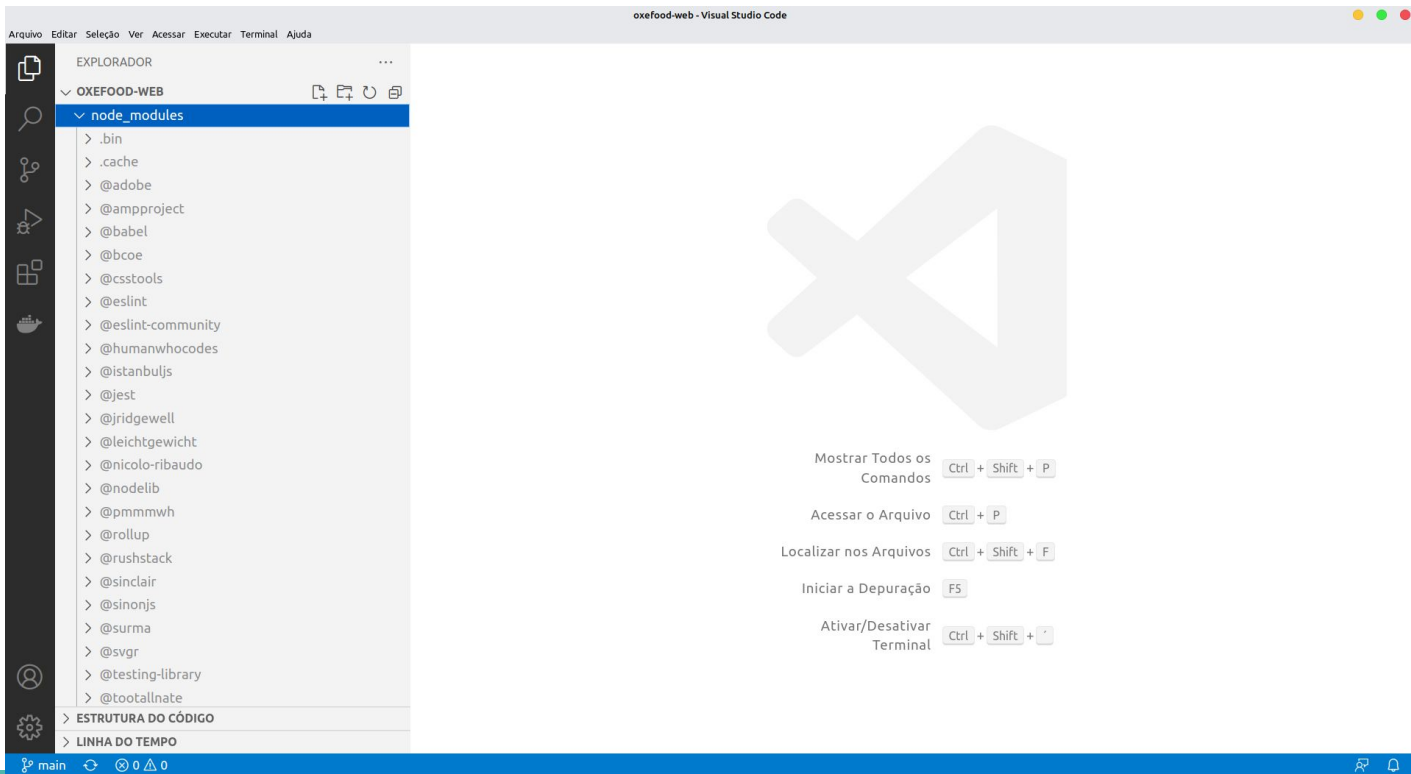
```
1 {
2   "name": "oxefood-web",
3   "version": "0.1.0",
4   "private": true,
5   "dependencies": {
6     "@testing-library/jest-dom": "^5.16.5",
7     "@testing-library/react": "^13.4.0",
8     "@testing-library/user-event": "^13.5.0",
9     "react": "^18.2.0",
10    "react-dom": "^18.2.0",
11    "react-scripts": "5.0.1",
12    "web-vitals": "^2.1.4"
13  },
14  "scripts": {
15    "start": "react-scripts start",
16    "build": "react-scripts build",
17    "test": "react-scripts test",
18    "eject": "react-scripts eject"
19  },
20  "eslintConfig": {
21    "extends": [
22      "react-app",
23      "react-app/jest"
24    ]
25  },
26  "browserslist": {
27    "production": [
28      ">0.2%",
29      "not dead",
30      "not op_mini all"
31    ],
32    "development": [
33      "last 1 chrome version",
34      "last 1 firefox version",
35      "last 1 safari version"
36    ]
37  }
38 }
```

Callout boxes highlighting key sections:

- Nome e versão do projeto**: Points to the `"name": "oxefood-web",` and `"version": "0.1.0",` fields.
- Dependências do projeto**: Points to the `"dependencies": { ... }` object.
- Scripts para iniciar a aplicação**: Points to the `"scripts": { ... }` object.

# Entendendo a Estrutura Inicial do Projeto

`node_modules` => diretório com as dependências do projeto



# Entendendo a Estrutura Inicial do Projeto

`public => index.html`

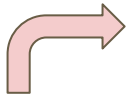
## Modifique:

- O título da página
- O ícone do app no browser

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/iconeApp.png" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
    />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
    <!--
      ...
    -->
    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
    <!--
      ...
    -->
    <title>Oxefood</title>
  </head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root"></div>
    <!--
      ...
    -->
  </body>
</html>
```

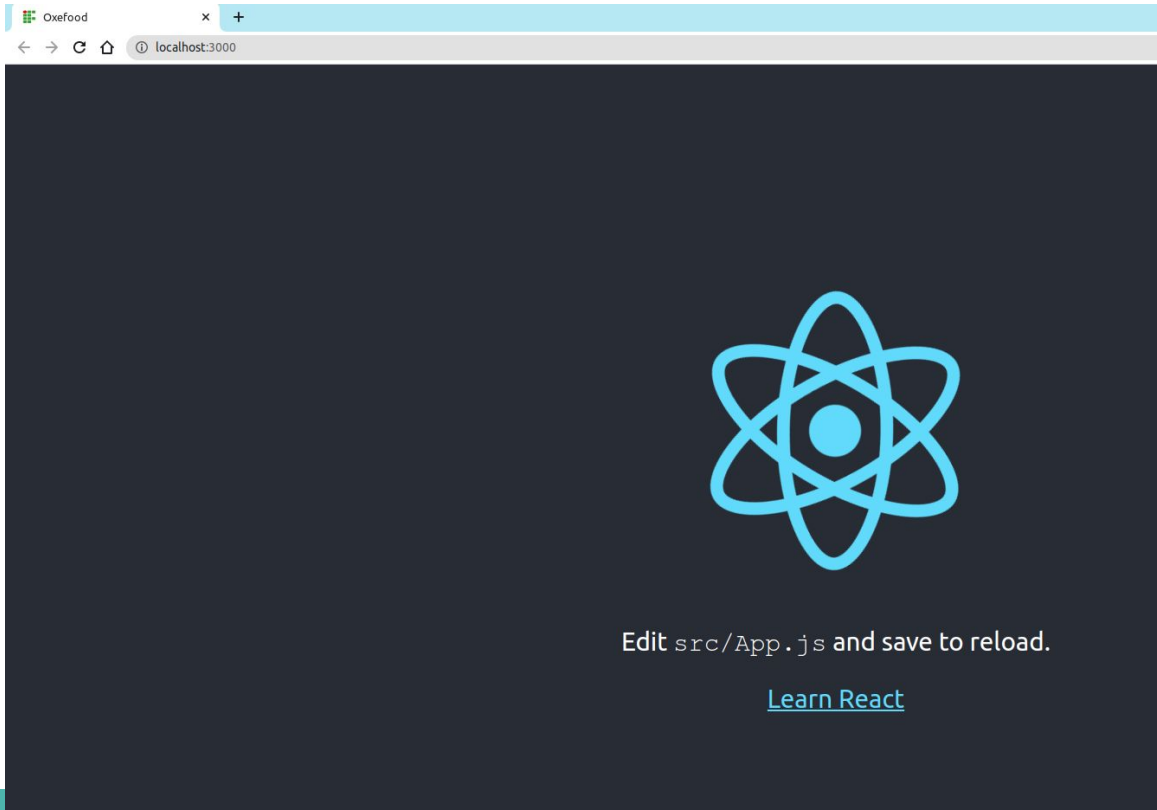
# Entendendo a Estrutura Inicial do Projeto

`public => index.html`



## Modifique:

- O título da página
- O ícone do app no browser



# Entendendo a Estrutura Inicial do Projeto

**src => index.js**

Inicializa o React e o ReactDOM

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import App from './App';
import './index.css';
import reportWebVitals from './reportWebVitals';
```

Importa o componente App, todo componente usado na tela, precisa ser importado.

```
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```

Carrega o componente App (App.js)

```
// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

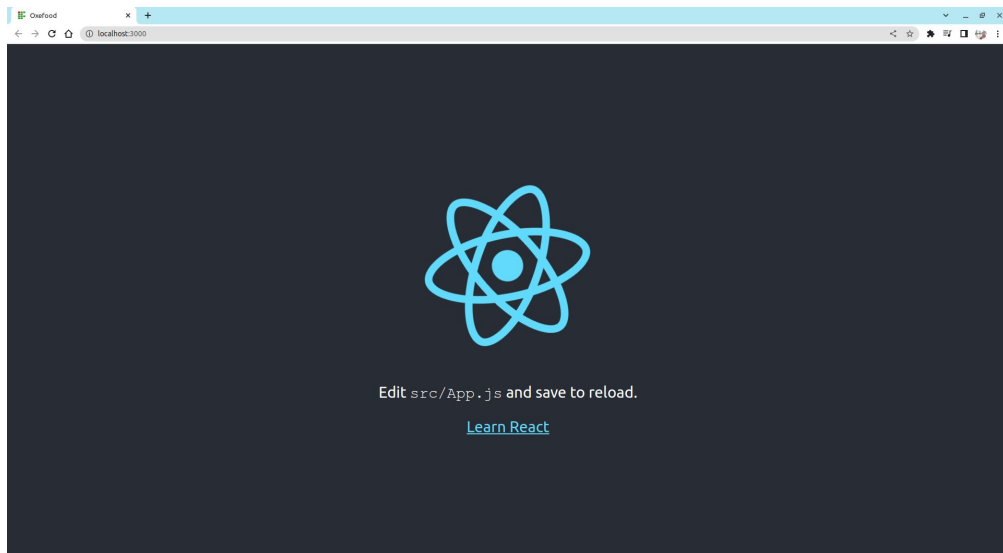
# Entendendo a Estrutura Inicial do Projeto

**src => App.js**

```
import logo from './logo.svg';
import './App.css';

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <img src={logo} className="App-logo" alt="logo" />
        <p>
          Edit <code>src/App.js</code> and save to reload.
        </p>
        <a
          className="App-link"
          href="https://reactjs.org"
          target="_blank"
          rel="noopener noreferrer"
        >
          Learn React
        </a>
      </header>
    </div>
  );
}

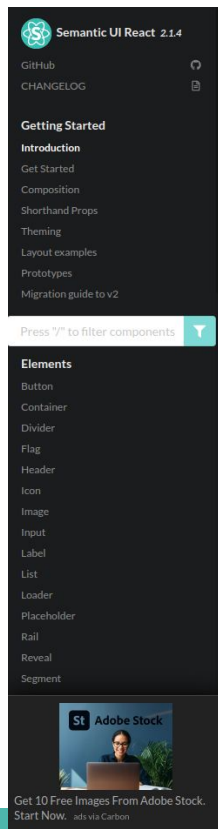
export default App;
```





# Entendendo a Estrutura Inicial do Projeto

## React Semantic UI



## Semantic UI React

The official Semantic-UI-React integration.

### Introduction

Semantic UI React is the official React integration for [Semantic UI](#).

- ✓ jQuery Free
- ✓ Declarative API
- ✓ Augmentation
- ✓ Shorthand Props
- ✓ Sub Components
- ✓ Auto Controlled State

Installation instructions are provided in the [Usage](#) section.

### jQuery Free

jQuery is a DOM manipulation library. It reads from and writes to the DOM. React uses a virtual DOM (a JavaScript representation of the real DOM). React only *writes* patch updates to the DOM, but *never reads* from it.

It is not feasible to keep real DOM manipulations in sync with React's virtual DOM. Because of this, all jQuery functionality has been re-implemented in React.

### Declarative API

Declarative APIs provide for robust features and prop validation.

# Entendendo a Estrutura Inicial do Projeto

## React Semantic UI

Para adicionar ao projeto a biblioteca do Semantic UI, execute os comandos na raiz do projeto:

- `npm add semantic-ui-react semantic-ui-css`
- `npm install`

Verifique se a biblioteca foi  
package.json:

```
{
  "name": "oxefood-web",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "@testing-library/jest-dom": "^5.16.5",
    "@testing-library/react": "^13.4.0",
    "@testing-library/user-event": "^13.5.0",
    "react": "^18.2.0",
    "react-dom": "^18.2.0",
    "react-scripts": "5.0.1",
    "semantic-ui-css": "^2.5.0",
    "semantic-ui-react": "^2.1.4",
    "web-vitals": "^2.1.4"
  },
  ...
}
```

# Entendendo a Estrutura Inicial do Projeto

## React Semantic UI

### Importe o CSS do Semantic UI no arquivo `index.js`

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import 'semantic-ui-css/semantic.min.css';
import App from './App';
import './index.css';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

# Entendendo a Estrutura Inicial do Projeto

## (1) Implementando o Primeiro Componente => Tela de Home

- Dentro da pasta `src`, crie uma nova pasta chamada `views`
- Dentro da pasta `views`, crie uma nova pasta chamada `home`
- Dentro da pasta `home`, crie um novo arquivo chamado `Home.jsx`

# Entendendo a Estrutura Inicial do Projeto

## (1) Implementando o Primeiro Componente => Tela de Home

```
import React from "react";
import { Container, Grid, Image } from 'semantic-ui-react';
```

Importe a biblioteca do semantic-ui

```
export default function Home () {
```

```
  return(
    <div>
```

```
      <div style={{marginTop: '5%'}}>
```

```
        <Container>
```

```
          <Grid columns={2} divided>
```

```
            <Grid.Row>
```

```
              <Grid.Column>
```

```
                <Image src='/logo-IFPE.png' size='large' />
```

```
              </Grid.Column>
```

```
            </Grid.Row>
```

```
            Bem vindo ao sistema <strong>OxeFood</strong> ! <br/>
```

```
            Este sistema foi desenvolvido na disciplina de Desenvolvimento para WEB III. <br/> <br/>
```

```
            Para acessar o código da <strong>API</strong> do sistema, acesse: <a href='https://github.com/robertoalencar/oxefood-api' target='_blank'> https://github.com/robertoalencar/oxefood-api
```

```
          </a> <br/> <br/>
```

```
            Para acessar o código do <strong>Módulo WEB</strong>, acesse: <a href='https://github.com/robertoalencar/oxefood-web' target='_blank'> https://github.com/robertoalencar/oxefood-web </a>
```

```
          </Grid.Column>
```

```
        </Grid.Row>
```

```
      </Grid>
```

```
    </Container>
```

```
  </div>
```

```
)
```

Utilize os componentes do semantic-ui

# Entendendo a Estrutura Inicial do Projeto

## (1) Implementando o Primeiro Componente => Tela de Home

### Altere o arquivo `App.jsx`

- Acrescente o componente `Home` a esta página e implemente uma `DIV` para o rodapé das telas do sistema.

```
import { Segment } from 'semantic-ui-react';
import './App.css';
import Home from './views/home/Home';
```

Importe o componente criado

```
function App() {
```

```
  return (
```

```
    <div className="App">
```

O conteúdo do componente `Home` será adicionado a esta parte da página

```
      <Home />
```

```
    <div style={{marginTop: '6%'}}>
```

```
      <Segment vertical color='grey' size='tiny' textAlign='center'
```

```
        &copy; 2023 - Projeto WEB III - IFPE Jaboatão dos Guararapes
```

```
      </Segment>
```

```
    </div>
```

Implementação do rodapé

```
  </div>
```

```
);
```

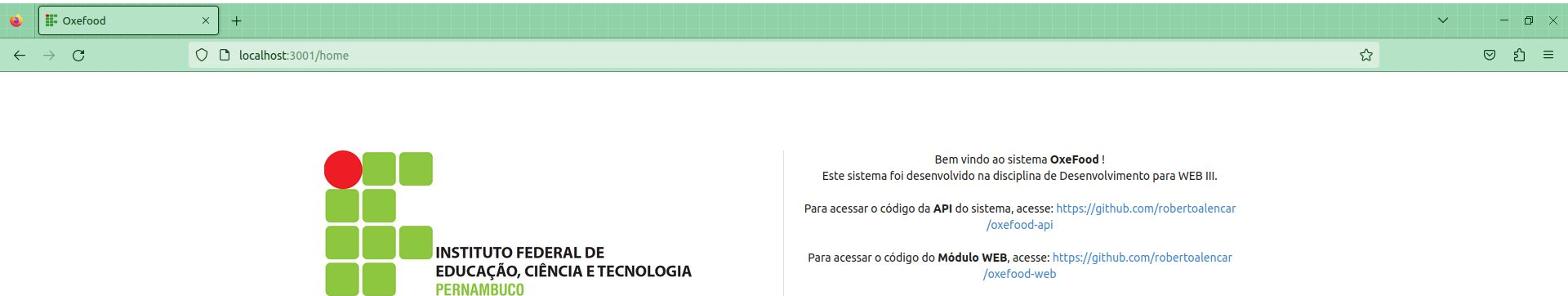
```
}
```

```
export default App;
```

# Entendendo a Estrutura Inicial do Projeto

## (1) Implementando o Primeiro Componente => Tela de Home

Altere o arquivo `App.jsx` (resultado)



# Entendendo a Estrutura Inicial do Projeto

## (2) Implementando o Segundo Componente => Tela de Cadastro de Cliente

- Dentro da pasta `views`, crie uma nova pasta chamada `cliente`
- Dentro da pasta `cliente`, crie um novo arquivo chamado `FormCliente.jsx`



# Entendendo a Estrutura Inicial do Projeto

## (2) Implementando o Segundo Componente => Tela de Cadastro de Cliente

Na raiz do projeto, execute o comando abaixo para adicionar a biblioteca do `react-input-mask`:

```
npm add react-input-mask --save
```

```
import React from "react";
import InputMask from 'react-input-mask';
import { Link } from "react-router-dom";
import { Button, Container, Divider, Form, Icon } from 'semantic-ui-react';

export default function FormCliente () {

  return (
    <div>
      <div style={{marginTop: '3%'}}>
        <Container textAlign='justified' >
          <h2> <span style={{color: 'darkgray'}}> Cliente &nbsp;<Icon name='angle double right' size="small" /> </span> Cadastro </h2>

          <Divider />

          ...
        </Container>
      </div>
    </div>
  );
}
```

Pegar o código completo na página da turma no Google Classroom.

# Entendendo a Estrutura Inicial do Projeto

## (2) Implementando o Segundo Componente => Tela de Cadastro de Cliente

```
import { Segment } from 'semantic-ui-react';
import './App.css';
import FormCliente from './views/cliente/FormCliente';

function App() {

  return (

    <div className="App">

      <FormCliente />

      <div style={{marginTop: '6%'}}>
        <Segment vertical color='grey' size='tiny' textAlign='center'>
          &copy; 2023 - Projeto WEB III - IFPE Jaboatão dos Guararapes
        </Segment>
      </div>

    </div>

  );
}

export default App;
```

# Entendendo a Estrutura Inicial do Projeto

## (2) Implementando o Segundo Componente => Tela de Cadastro de Cliente

← → ↺ 🏠 ⓘ localhost:3000

Cliente » Cadastro

Nome\* CPF

Fone Celular Fone Fixo Data Nascimento

Ex: 20/03/1985

⬅ Voltar ➡ Salvar

© 2023 - Projeto WEB III - IFPE Jaboatão dos Guararapes

# Dúvidas



# Exercício

Implemente a tela de => Cadastro de Produto

views/produto/FormProduto.jsx (resultado esperado)

The screenshot shows a web browser window with the address bar displaying 'localhost:3000'. The page content is a form titled 'Produto » Cadastro'. The form has the following fields and labels:

- Título \***: A text input field with placeholder text 'Informe o título do produto'.
- Código do Produto \***: A text input field with placeholder text 'Informe o código do produto'.
- Descrição**: A larger text input field with placeholder text 'Informe a descrição do produto'.
- Valor Unitário \***: A text input field.
- Tempo de Entrega Mínimo em Minutos**: A text input field with the value '30'.
- Tempo de Entrega Máximo em Minutos**: A text input field with the value '40'.

At the bottom of the form, there are two buttons:

- Listar**: A button with a left arrow icon.
- Salvar**: A button with a save icon.

At the bottom of the browser window, there is a footer text: '© 2023 - Projeto WEB III - IFPE Jaboatão dos Guararapes'.

# Exercício

Implemente a tela de => Cadastro de Fornecedor

views/entregador/FormEntregador.jsx (resultado esperado)

Entregador » Cadastro

Nome *	CPF *	RG		
<input type="text"/>	<input type="text"/>	<input type="text"/>		
DT Nascimento	Fone Celular *	Fone Fixo	QTD Entregas Realizadas	Valor Por Frete
<input type="text" value="Ex: 20/03/1985"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Rua	Número			
<input type="text"/>	<input type="text"/>			
Bairro	Cidade	CEP		
<input type="text"/>	<input type="text"/>	<input type="text"/>		
UF				
<input type="text" value="Selecione"/>				
Complemento				
<input type="text"/>				
Ativo: <input type="radio"/> Sim <input checked="" type="radio"/> Não				

 Voltar

 Salvar

The image features a white background with two large, solid green abstract shapes. One shape is a semi-circle on the left side, and the other is a more complex, organic shape on the right side. Centered between these shapes is the text "Obrigado !".

Obrigado !