**A close up of a logo

AI-generated content may be incorrect.**

**Mini Project Report - 12**

Master of Computer Application – General

Semester – III

**Sub: Web Technologies**

**Topic: Node.js**  
By  
**Name:** SANDRA B  
**Reg no.:** 24110222500001

**Faculty Name:** VEERA RAGHAV K

**Faculty Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Department of Computer Application  
Alliance University  
Chandapura - Anekal Main Road, Anekal  
Bengaluru - 562 106**

**August 2025**

|  |  |  |
| --- | --- | --- |
| **Sno.** | **INDEX** | **Page No.** |
| 1. | Introduction | 2 |
| 2. | Objective | 2 |
| 3. | Tools and Technologies Used | 2 |
| 4. | Description of Code | 2 |
| 5. | Working of the Project | 5 |
| 6. | Features | 5 |
| 7. | Output | 5 |
| 8. | Advantages | 6 |
| 9. | Future Scope | 6 |
| 10. | Conclusion | 6 |

**Mini Project Report – Login Page using Node.js**

**Introduction**

Web applications play a vital role in modern technology by providing user-friendly interfaces for interaction. A login page is the most common entry point for any application, allowing users to authenticate and gain access. This project demonstrates how to create a simple login page using **React.js**, a JavaScript library for building user interfaces.

**Objective**

The main objectives of this project are:

* To understand the basics of **React.js** and how to set up a React project.
* To create a simple login page with **username** and **password** fields.
* To demonstrate how forms are handled in React.
* To showcase the use of **Node.js** and **npm** in running React applications.

**Tools and Technologies Used**

1. **Node.js** – Runtime environment to execute JavaScript outside a browser.
2. **npm (Node Package Manager)** – To install and manage project dependencies.
3. **React.js** – JavaScript library for building user interfaces.
4. **Create React App (CRA)** – A tool to set up React applications quickly.
5. **Visual Studio Code** – Code editor for development.
6. **Web Browser (Google Chrome/Edge)** – For testing and viewing the output.

**Description of Code**

1. **Project Setup**  
   Commands used to create and run the project:

* npx create-react-app my-app
* cd my-app
* npm start

A screen shot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

* + npx create-react-app my-app → Creates a new React project.
  + cd my-app → Navigates into the project folder.
  + npm start → Runs the development server and opens the app in the browser.

1. **React Code (App.js)**

function App() {

return (

<div style={{ textAlign: "center", marginTop: "100px" }}>

<h1>Alliance University</h1>

<form>

<div style={{ margin: "10px" }}>

<label>User Name: </label>

<input type="text" placeholder="Enter username" />

</div>

<div style={{ margin: "10px" }}>

<label>Password: </label>

<input type="password" placeholder="Enter password" />

</div>

<button type="submit" style={{ marginTop: "10px" }}>

Login

</button>

</form>

</div>

);

}

export default App;

**Working of the Project**

1. When the project is started using npm start, React runs a development server at http://localhost:3000.
2. The **App component** renders a login page with a form.
3. The form contains two input fields:
   * Username (text input)
   * Password (password input)
4. On clicking the **Login button**, the form is submitted. (In this simple version, no backend validation is added).

**Features**

* Simple and user-friendly login page.
* Uses **React components** for modularity.
* Easy to run with just three commands (npx, cd, npm start).
* Can be extended to include authentication.

**Output**

The output is a **login page** displayed in the browser.

* A heading **"Alliance University"** is shown.
* A form with **username** and **password** fields.
* A **Login button** for submission.

A screenshot of a computer

AI-generated content may be incorrect.

**Advantages**

* Lightweight and easy to understand.
* Provides a foundation for creating more complex React projects.
* Demonstrates the use of **Node.js** and **npm** in frontend development.
* Can be extended to connect with a backend (Node.js/Express, MongoDB, etc.).

**Future Scope**

* Implement actual **authentication** with backend APIs.
* Add **form validation** (checking empty fields, incorrect input, etc.).
* Apply **CSS/Bootstrap/Tailwind** for better styling.
* Include options like **Forgot Password** and **Sign Up**.
* Make it mobile responsive.

**Conclusion**

This project successfully demonstrates how to create a simple login page using **React.js**. It covers the setup of a React environment with Node.js, basic component creation, and rendering of a form. The project can be enhanced with backend integration and additional features, making it a foundation for real-world applications.