



VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)



MINI PROJECT REPORT

FITNESS TRACKER WEBSITE

BCSE203E | Web Programming Lab | 18-04-24

By

22BCE3138 – Vaibhav Raj

22BCE2660 – Prasenjeet Thul

i. ABSTRACT

In today's fast-paced world, maintaining a healthy lifestyle is more important than ever. However, amidst busy schedules and numerous distractions, staying committed to fitness goals can be challenging. To address this, our web development project introduces a comprehensive fitness tracker website built using React, aimed at empowering individuals to take control of their fitness journeys.

This fitness tracker offers users a personalized platform to track their progress, set achievable goals, and foster accountability through habit monitoring. Leveraging the power of React, the website provides a seamless and intuitive user experience, with features including user authentication for data privacy, habit tracking for logging daily fitness activities, and goal setting for monitoring progress towards specific fitness objectives.

Furthermore, the website incorporates community engagement elements, such as forums for discussion and support, fostering a sense of camaraderie among users. Through the implementation of responsive design principles and optimization techniques, including minification of JS and CSS files and mobile device compatibility, the website ensures a smooth and efficient user experience across various devices.

By combining the flexibility of React with the functionality of a fitness tracking platform, our project aims to empower individuals to achieve their fitness goals, promote accountability, and ultimately lead healthier and happier lives.

ii. ACKNOWLEDGEMENT

First and foremost, we express our sincere appreciation to our project supervisor, Professor Dhivyaa C. R., for their guidance, support, and invaluable insights throughout the development process. Their expertise and encouragement have been instrumental in shaping the direction of this project and ensuring its successful completion.

We would also like to thank our team members for their dedication, collaboration, and tireless efforts in bringing this project to fruition. Each member's unique skills and contributions have played a significant role in the project's development, from conceptualization to implementation.

Additionally, we would like to acknowledge the open-source community and the developers behind the various libraries, frameworks, and tools used in this project. Their contributions have significantly enriched the development process and enabled us to create a robust and feature-rich fitness tracker website.

Thank you.

iii. INDEX

S. No.	Title	Page No.
1	Abstract	2
2	Acknowledgement	3
3	Introduction	5
4	Objectives of Mini Project	6
5	Problem Description	8
	Problem Statements	8
6	Methodology	10
7	Modules and Features	12
	Additional Technologies	14
8	Design Layout and Workflow	16
9	Appendix - Sample Codes	18
10	Results	23
11	Conclusion	30

1. INTRODUCTION

In an era characterized by sedentary lifestyles and increasing health concerns, the importance of maintaining an active and healthy lifestyle cannot be overstated.

Recognizing this need, our project endeavors to address the challenges individuals face in achieving their fitness goals by introducing a comprehensive Fitness Tracker Website built using React.

The Fitness Tracker Website serves as a digital companion, empowering users to take control of their fitness journeys through personalized tracking, goal setting, and community support. Leveraging the power of React, the website offers a seamless and intuitive user experience, facilitating engagement and motivation for users of all fitness levels.

This introduction sets the stage for the project report, providing a glimpse into the purpose and objectives of the Fitness Tracker Website. Over the course of this report, we will delve deeper into the design, development, and implementation of the website, exploring its features, functionalities, and the technologies employed to bring it to life. Through this exploration, we aim to showcase the capabilities of React in building dynamic and interactive web applications tailored to the needs of modern users striving for healthier lifestyles.

2. OBJECTIVES OF THE WORK

2.1 Empower Users to Take Control of Their Fitness Journeys:

- The primary objective of the Fitness Tracker Website is to empower users to take control of their fitness journeys by providing them with the tools and resources they need to succeed.
- Through personalized tracking, goal setting, and habit monitoring, the website aims to equip users with actionable insights and guidance to make informed decisions about their health and fitness.

2.2 Promote Accountability and Consistency:

- Consistency is key to achieving fitness goals, yet many individuals struggle to maintain regular exercise routines and healthy habits. The website aims to promote accountability by encouraging users to track their daily fitness activities and monitor their progress over time.
- By fostering a sense of accountability, the website helps users stay committed to their goals and overcome common obstacles that hinder progress.

2.3 Facilitate Motivation and Progress Visualization:

- Visualizing progress is a powerful motivator for individuals striving to improve their fitness levels. The website provides users with tools to visualize their progress, track achievements, and celebrate milestones along their fitness journey.
- Through interactive charts, graphs, and personalized dashboards, users can gain insights into their performance, identify areas for improvement, and stay motivated to continue their pursuit of health and wellness.

2.4 Create a Supportive Community Environment:

- Fitness journeys can be challenging, but they are often more manageable with support from others. The website aims to create a supportive community

environment where users can connect with like-minded individuals, share experiences, and provide encouragement to one another.

- By facilitating discussions, sharing tips and advice, and fostering camaraderie among users, the website promotes a sense of belonging and solidarity within the fitness community.

2.5 Enhance User Experience and Accessibility:

- A seamless user experience is essential for engagement and retention. The website prioritizes user experience by leveraging the capabilities of React to create responsive and intuitive interfaces that adapt to various devices and screen sizes.
- Accessibility is also a key consideration, ensuring that users of all abilities can access and navigate the website with ease. Features such as keyboard navigation, screen reader compatibility, and color contrast optimization are implemented to enhance accessibility for all users.

By fulfilling these objectives, the Fitness Tracker Website aims to empower individuals to lead healthier, happier, and more active lifestyles, ultimately contributing to improved overall well-being and quality of life.

3. PROBLEM DESCRIPTION

Developing a fitness tracker website using React, styled-components, and Material-UI is a significant project. The website includes various features such as user authentication, a navigation bar, a dashboard with calorie tracker and charts, workout logging, plans for different workout splits, a forum for discussion, a section for supplementary products, and a contact us page. Additionally, the website is optimized for smaller devices like phones to ensure a seamless user experience.

3.1. PROBLEM STATEMENTS:

- **Problem Statement: Lack of Convenient Fitness Tracking**

Solution: Our fitness tracker website provides a convenient platform for users to track their fitness activities, including calorie intake, workouts, and progress, all in one place.

- **Problem Statement: Limited Workout Planning Options**

Solution: Our website offers a variety of workout plans, including 3-day, 4-day, and 5-day splits, catering to users with different fitness goals and schedules.

- **Problem Statement: Difficulty in Engaging with Fitness Community**

Solution: Our forum section enables users to engage with a community of fitness enthusiasts, sharing tips, experiences, and motivation to help each other achieve their fitness goals.

- **Problem Statement: Inaccessibility of Fitness Resources**

Solution: Our website provides a section dedicated to supplementary products, offering users access to a range of fitness-related products and resources to enhance their workouts.

- **Problem Statement: Lack of User-Friendly Contact Options**

Solution: Our "Contact Us" page offers easy-to-use contact forms, providing users with a hassle-free way to reach out for assistance or inquiries.

- **Problem Statement: Inefficient Workout Logging**

Solution: Our workout logging feature simplifies the process of recording workouts, providing users with an intuitive interface to log exercises, sets, reps, and weights with ease.

- **Problem Statement: Limited Visualization of Fitness Progress**

Solution: Our dashboard includes interactive charts and graphs, allowing users to visualize their fitness progress over time, helping them stay motivated and track their achievements.

- **Problem Statement: Complexity in User Authentication**

Solution: Our website offers a seamless user authentication system, making it easy for users to sign up, log in, and manage their accounts securely.

- **Problem Statement: Poor Accessibility on Mobile Devices**

Solution: Our website is optimized for smaller devices like phones, ensuring that users can access and use the platform conveniently, regardless of the device they're using.

4. METHODOLOGY

4.1 Project Planning and Requirements Gathering:

- Conduct initial research to understand user needs, competitor analysis, and market trends in fitness tracking websites.
- Define project goals, objectives, and scope, considering the features and functionalities required in the fitness tracker website.

4.2 Technology Selection:

- Choose React as the frontend library for building the user interface, leveraging its component-based architecture and virtual DOM for efficient rendering.
- Select styled-components as the styling solution to implement CSS-in-JS, enabling the creation of reusable and maintainable styled components with scoped styles.
- Opted for Material-UI for its pre-designed components, charts, graphs, icons and adherence to Material Design principles.
- Define routes using react-router-dom for different sections of the website (e.g., Dashboard, Workout Logging) in Navbar.

4.3 User Interface Design:

- Design user interfaces for the fitness tracker website, focusing on usability, accessibility, and visual appeal.
- Create wireframes, mockups, and prototypes using design tools or frameworks to visualize the layout, navigation, and interaction patterns.
- Implement responsive design principles to ensure optimal viewing experience across various devices and screen sizes.

4.4 Component Architecture and Styling:

- Define the component architecture, breaking down the UI into reusable and composable components based on functionality and visual hierarchy.

- Implement styled-components to style each component, using CSS-in-JS syntax to encapsulate styles within component scope and ensure consistency and maintainability.
- Leverage styled-components' features such as theming, props-based styling, and dynamic styling to create flexible and customizable components.

4.5 Development Iterations:

- Adopt an iterative development approach, starting with implementing core features and gradually iterating and refining the design and functionality.
- Begin development by setting up the project structure, configuring build tools (e.g., webpack, mui), and integrating necessary libraries and dependencies.
- Develop components and UI elements using React and styled-components, focusing on modularity, reusability, and adherence to design specifications.

4.6 Integration and Testing:

- Integrate frontend components with backend APIs and services to create a cohesive and functional web application.
- Conduct unit tests and integration tests to ensure the correctness and robustness of components, handling edge cases and error scenarios.
- Perform cross-browser testing and device testing to verify compatibility and responsiveness across different browsers and devices.

5. MODULES AND FEATURES

5.1 User Authentication:

- Secure user authentication system allowing users to create accounts, log in, and securely access their personalized fitness profiles.
- Encrypted authentication process ensuring user data privacy and protection.

5.2 Navigation Bar:

- Display links to different sections of the website (e.g., Dashboard, Workout Logging, Plans, Forum, Supplementary Products, Contact Us).
- Implement as NavLink components from React Router for internal navigation.
- Handle active link highlighting using NavLink's activeClassName prop.
- Display website logo or name for brand recognition.
- Clicking on the logo/name navigates the user to the homepage.
- Adapt to different screen sizes, ensuring usability on various devices.
- Collapse menu into a hamburger icon for smaller screens.

5.3 Dashboard:

- Personalized dashboard providing users with an overview of their fitness activities, goals, and progress.
- Dynamic charts and graphs displaying metrics such as workout frequency, calories burned, and progress towards fitness goals.
- Quick access to recently logged activities and recommended workouts based on user preferences.

5.4 Workout Logging:

- Comprehensive workout logging functionality enabling users to log their daily exercise routines, including cardio, strength training, yoga, and more.
- Intuitive interface for adding, editing, and deleting workout entries, with options to specify activity type, duration, intensity, and notes.

5.5 Workout Plans:

- Predefined workout plans (e.g., 3-day split, 4-day split, 5-day split).
- Progress tracking and adaptation of plans based on user feedback.

5.6 Discussion Forum:

- Community forum feature facilitating interaction and engagement among users, allowing them to share experiences, seek advice, and offer support to one another.
- Post creation and replying by users.

5.7 Supplementary Products:

- Product listings with images, descriptions, and prices.
- Link to external websites for purchase.

5.8 Contact Us:

- Contact form for users to send inquiries and feedback.
- FAQ section for common queries.

5.9 Responsive Design:

- Mobile-friendly design ensuring seamless user experience across devices, including smartphones, tablets, and desktop computers.
- Responsive layout and flexible styling with styled-components ensuring optimal display and functionality on screens of all sizes.

5.10 Additional Technologies:

A. React Router (react-router-dom)

- Description: React Router is a popular routing library for React applications, specifically the react-router-dom package for web applications. It enables navigation and routing functionalities in single-page applications (SPAs).
- Benefits:
 - Declarative routing: Define routes as components, making routing logic easy to understand and maintain.
 - Nested routing: Support nested routes, enabling hierarchical page structures.
 - Dynamic routing: Facilitate dynamic route matching and parameter passing.
- Use Cases: Define routes for different sections of the website (e.g., Dashboard, Workout Logging). Implement protected routes for authenticated users.

B. Styled-Components

- Description: Styled-components is a library for React and React Native that allows you to use component-level styles in your application. Instead of separate CSS files or inline styles, you define your styles directly within your components using JavaScript.
- Benefits:
 - Scoped styles: Styles are encapsulated within components, preventing global style pollution.
 - Dynamic styling: Styles can be based on props, enabling dynamic UI based on state.
 - Improved maintainability: Styles are colocated with components, making it easier to understand and maintain styling.
- Use Cases: Style individual components like buttons, inputs, and cards. Implement responsive design with ease using styled-components' built-in support for media queries.

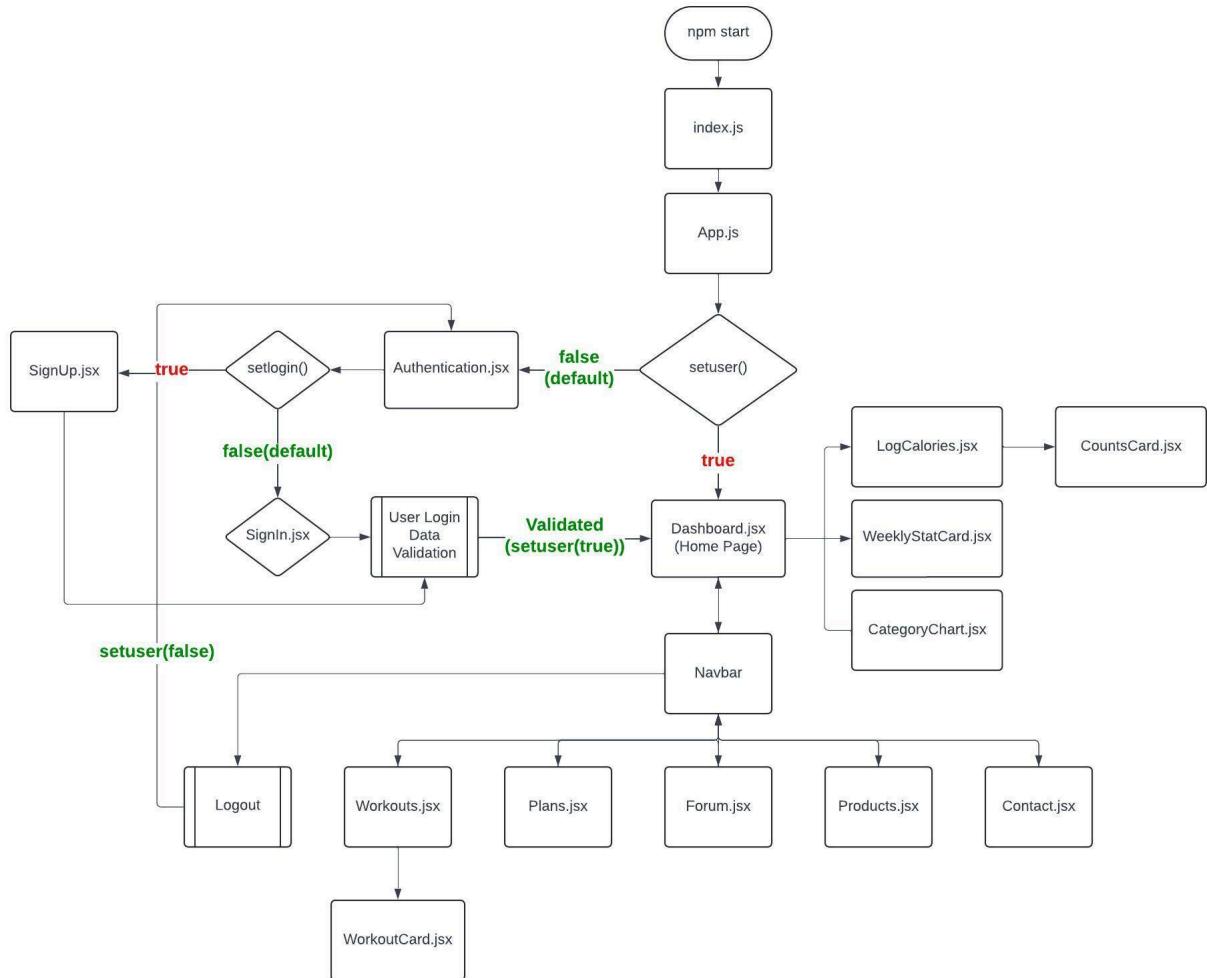
C. MUI-Icons (Material-UI Icons)

- Description: MUI-icons are a set of icons provided by the Material-UI library, which is a popular React UI framework implementing Google's Material Design. These icons are vector-based and designed to be used in web applications.
- Benefits:
 - Extensive collection: Offers a wide range of icons covering various categories like action, alert, communication, and more.
 - Easy integration: Can be easily integrated into React components using the provided icon components or SVG icons.
 - Consistent styling: Icons follow the Material Design guidelines, ensuring consistency with the overall design.
- Use Cases: Add visual elements to navigation links, buttons, and UI components. Enhance user experience with intuitive and recognizable icons for actions and features.

D. MUI Graphs and Charts (Material-UI DataGrid, Charts, etc.)

- Description: Material-UI provides a set of graph and chart components, including DataGrid, LineChart, BarChart, PieChart, etc., for data visualization in React applications.
- Benefits:
 - Integration with Material Design: Graphs and charts follow Material Design principles, ensuring consistency with the overall design.
 - Customization: Provides options for customization, such as colors, labels, and tooltips.
 - Responsive design: Graphs and charts are responsive and adapt to different screen sizes.
- Use Cases: Display workout progress over time with LineChart or AreaChart. Visualize calorie intake and expenditure with PieChart or BarChart. Show forum statistics and user engagement with DataGrid or BarChart.

6. DESIGN LAYOUT



6.1. WORKFLOW BREAKDOWN

1. User Authentication

- The process starts with a user visiting the website. The user can either sign up for a new account or sign in to an existing account. This is handled by the `Signup.jsx` and `Signin.jsx` components.

- Upon successful signup or login, the user data is validated and a ‘user’ state variable is set to ‘true’ which seems to grant the user full access to the website’s functionalities. This is handled in the Authentication.jsx component.

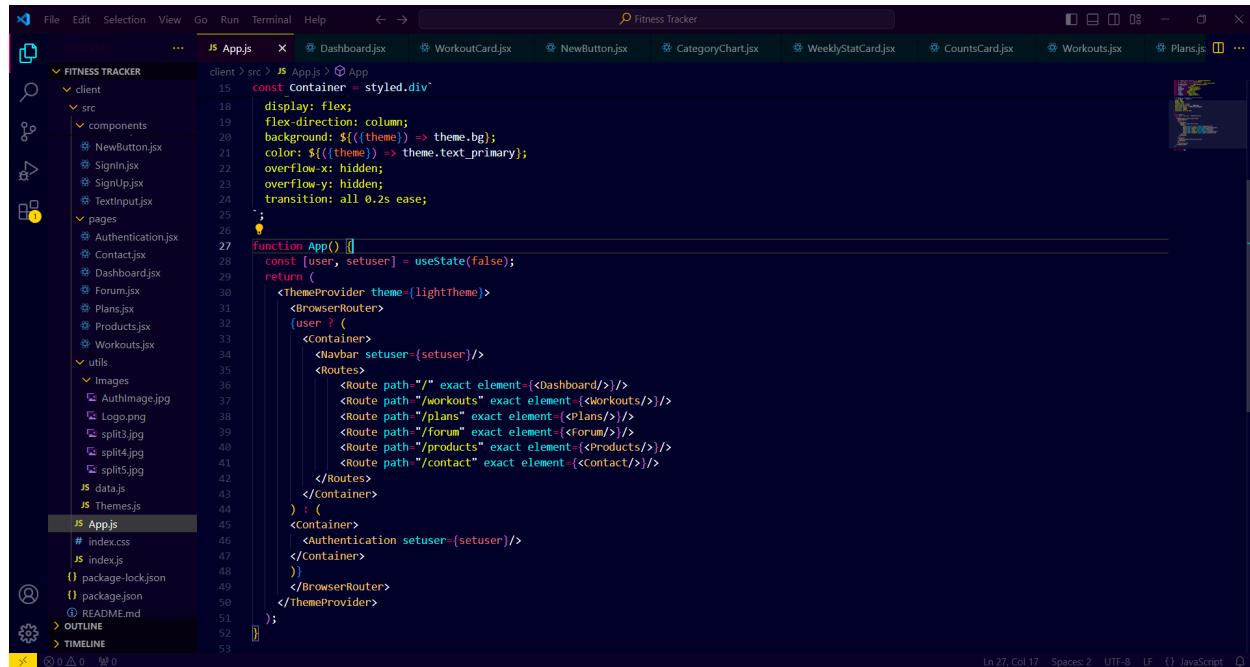
2. Application Workflow

- After successful login, the user is directed to the Dashboard.jsx component, which appears to be the home page of the application.
- The Navbar component provides access to other sections of the web application like Workout.jsx, Plans.jsx, Forum.jsx, Products.jsx and Contact.jsx.
- Logout: Clicking logout triggers the setuser state variable to be set to ‘false’ and the user is redirected back to the login page (SignIn.jsx).

3. Additional Functionalities

- Workout Logging: The Workout.jsx component likely allows users to log their workout data.
- Plans: Plans.jsx provides users with access to pre-built workout plans (3-day, 4-day, 5-day splits).
- Forum: The Forum.jsx component seems to be a discussion board where users can interact with each other.
- Products: The Products.jsx component allows users to browse and potentially purchase supplementary products that might aid their fitness goals.
- Contact Us: Contact.jsx provides a way for users to contact the website’s administrators.

7. APPENDIX - SAMPLE CODES



```

const Container = styled.div`  

  display: flex;  

  flex-direction: column;  

  background: ${((theme)) => theme.bg};  

  color: ${((theme)) => theme.text_primary};  

  overflow-x: hidden;  

  overflow-y: hidden;  

  transition: all 0.2s ease;  

function App() {  

  const [user, setUser] = useState(false);  

  return (  

    <ThemeProvider theme={lightTheme}>  

      <BrowserRouter>  

        {user ? (  

          <Container>  

            <Navbar setuser={setUser}/>  

            <Routes>  

              <Route path="/" exact element={<Dashboard/>}/>  

              <Route path="/workouts" exact element={<Workouts/>}/>  

              <Route path="/plans" exact element={<Plans/>}/>  

              <Route path="/forum" exact element={<Forum/>}/>  

              <Route path="/products" exact element={<Products/>}/>  

              <Route path="/contact" exact element={<Contact/>}/>  

            </Routes>  

          </Container>  

        ) : (  

          <Container>  

            <Authentication setuser={setUser}/>  

          </Container>  

        )  

      </BrowserRouter>  

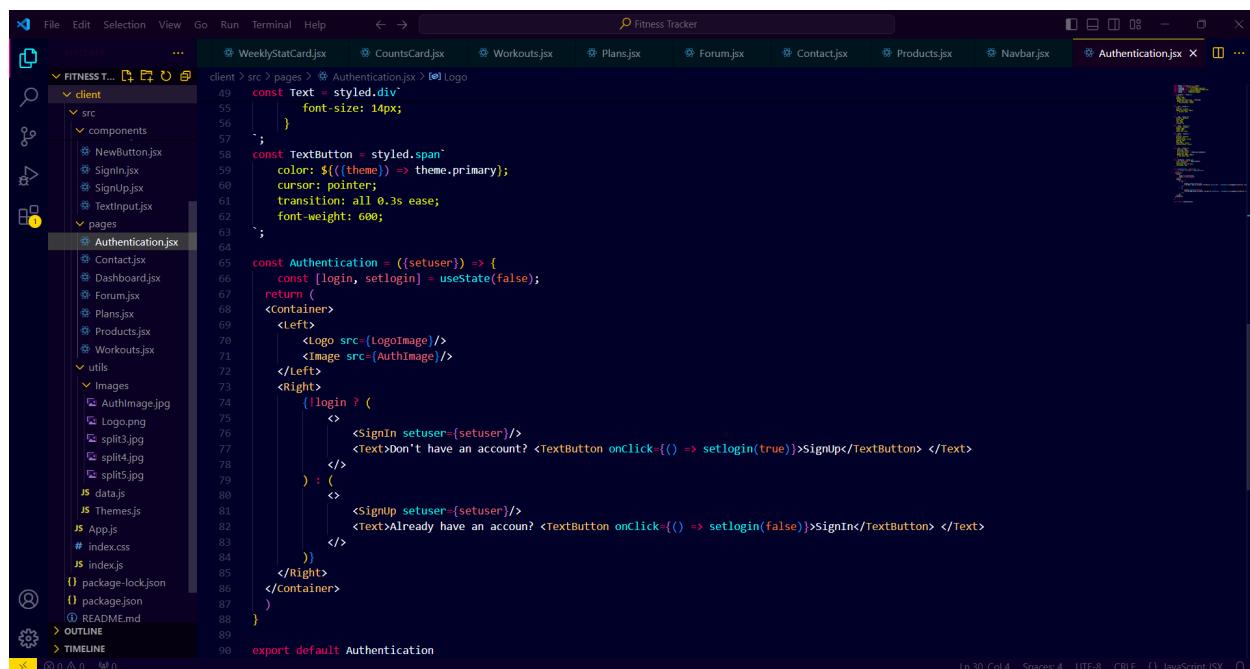
    </ThemeProvider>  

  );  

}

```

Main App.js Code



```

const Text = styled.div`  

  font-size: 14px;  

  ...  

`;  

const TextButton = styled.span`  

  color: ${((theme)) => theme.primary};  

  cursor: pointer;  

  transition: all 0.3s ease;  

  font-weight: 600;  

`;  

const Authentication = ({setUser}) => {  

  const [login, setLogin] = useState(false);  

  return (  

    <Container>  

      <Left>  

        <Logo src={logoImage}/>  

        <Image src={authImage}/>  

      </Left>  

      <Right>  

        {login ? (  

          <>  

            <signIn setUser={setUser}/>  

            <Text>Don't have an account? <TextButton onClick={() => setLogin(true)}>Sign Up</TextButton> </Text>  

          </>  

        ) : (  

          <>  

            <signUp setUser={setUser}/>  

            <Text>Already have an account? <TextButton onClick={() => setLogin(false)}>Sign In</TextButton> </Text>  

          </>  

        )  

      </Right>  

    </Container>  

  )
}  

export default Authentication

```

Authentication.jsx (Login Page)

```

client > src > components > Signin.jsx
const SignIn = ({ setuser }) => {
  const [userData, setuserData] = useState({
    "Name": '',
    "Email": '',
    "Password": ''
  });

  const handleChange = (e) => {
    const { name, value } = e.target;
    setuserData({ ...userData, [name]: value });
    // Clear previous errors when user starts typing
    if (name === 'Name') {
      setNameError('');
    } else if (name === 'Email') {
      setEmailError('');
    } else if (name === 'Password') {
      setPasswordError('');
    }
  };

  const handleSubmit = (e) => {
    e.preventDefault();
    let isValid = true;

    // Name validation
    if (!userData.Name.trim()) {
      setNameError('Name is required');
      isValid = false;
    }

    // Email validation
    if (!userData.Email.trim()) {
      setEmailError('Email is required');
      isValid = false;
    } else if (!validateEmail(userData.Email)) {
      setEmailError('Please enter a valid email address');
      isValid = false;
    }

    // Password validation
    if (!userData.Password.trim()) {
      setPasswordError('Password is required');
      isValid = false;
    }
  };
}

const validateEmail = (email) => {
  const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
  return emailRegex.test(email);
}

```

User Login Form Validation

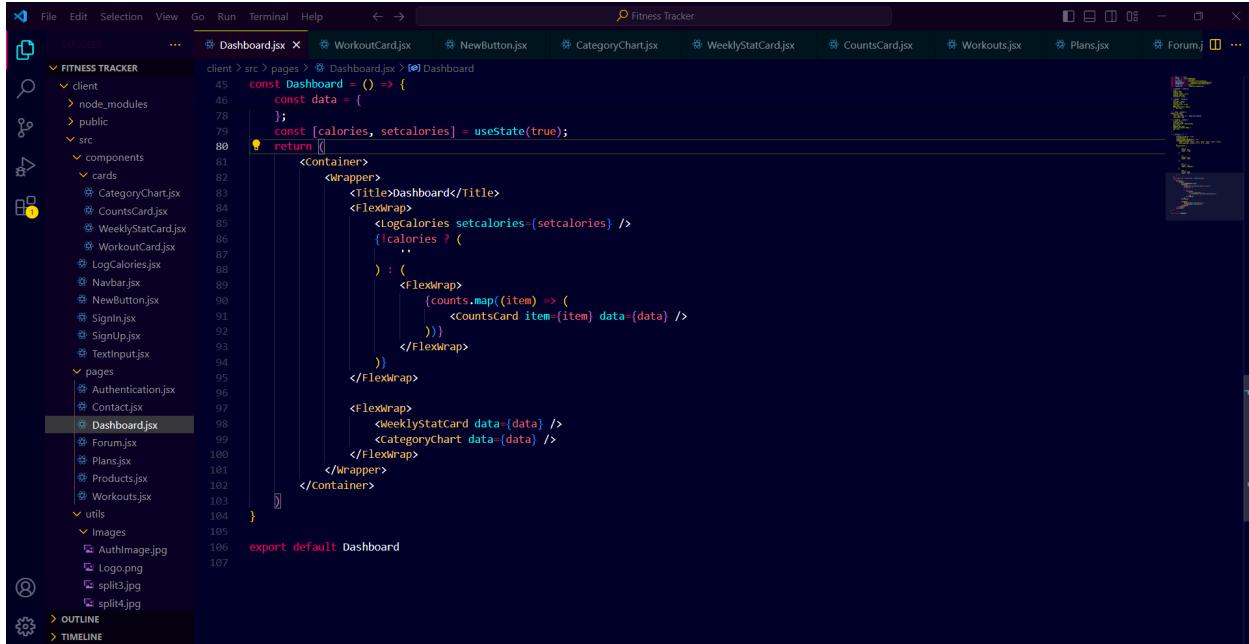
```

client > src > components > Navbar.jsx
const Navbar = ({setuser}) => {
  const [isOpen, setIsOpen] = useState(false);
  return (
    <Nav>
      <NavContainer>
        <MobileIcon onClick={() => setIsOpen(!isOpen)}>
          <MenuRounded sx={{color: 'inherit'}}/>
        </MobileIcon>
        <NavLogo to="/">
          <Logo src={logoImg}/>
        FitForge
        </NavLogo>

        <MobileMenu isOpen={isOpen}>
          <NavLine to="/Dashboard">NavLink</NavLine>
          <NavLink to="/workouts">Workouts</NavLink>
          <NavLink to="/plans">Plans</NavLink>
          <NavLink to="/forum">Forum</NavLink>
          <NavLink to="/products">Products</NavLink>
          <NavLink to="/contact">Contact</NavLink>
        </MobileMenu>
      </NavContainer>
      <NavItems>
        <NavLine to="/">Dashboard</Navlink>
        <NavLink to="/workouts">Workouts</Navlink>
        <NavLink to="/plans">Plans</Navlink>
        <NavLink to="/forum">Forum</Navlink>
        <NavLink to="/products">Products</Navlink>
        <NavLink to="/contact">Contact</Navlink>
      </NavItems>
      <UserContainer>
        <Avatar/>
        <TextButton onClick={()=>setuser(false)}>Logout</TextButton>
      </UserContainer>
    </Nav>
  );
}

```

Navbar.jsx

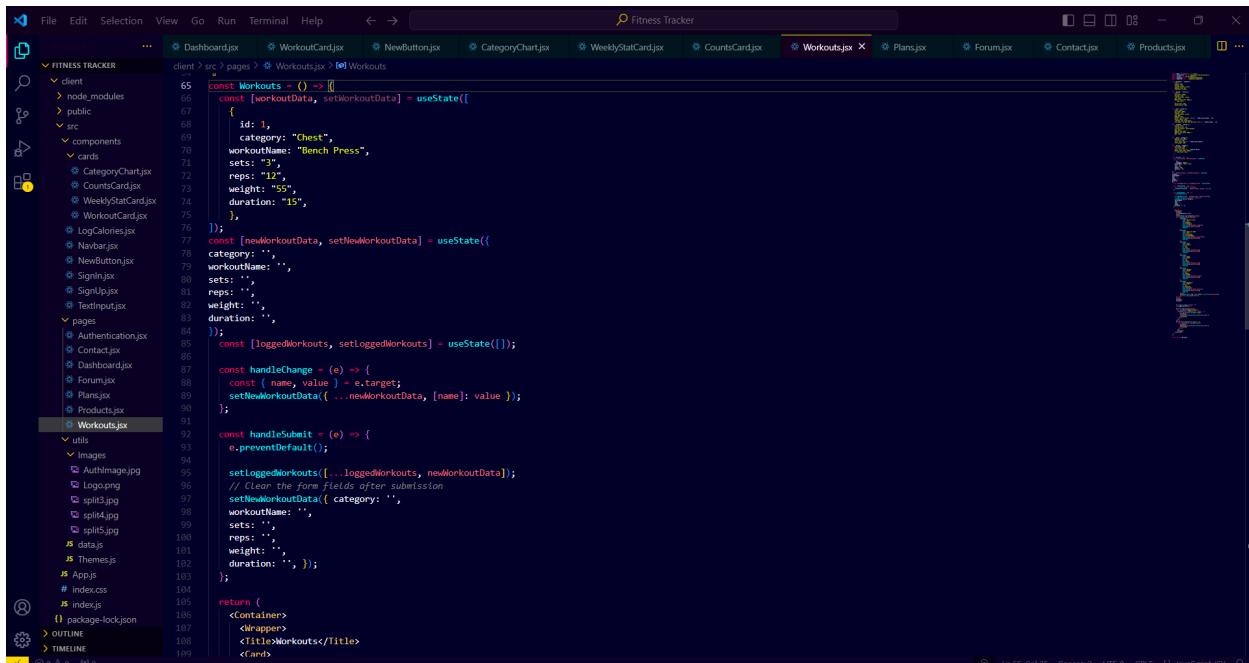


```

client > src > pages > Dashboard.jsx > Dashboard
  45 const Dashboard = () => {
  46   const [data] = useState([
  47     ...
  48   ]);
  49   const [calories, setCalories] = useState(true);
  50
  51   return (
  52     <Container>
  53       <Wrapper>
  54         <Title>Dashboard</Title>
  55         <FlexWrap>
  56           <LogCalories setCalories={setCalories} />
  57           {calories ? (
  58             ...
  59           ) : (
  60             <FlexWrap>
  61               {data.map((item) => (
  62                 <CountsCard item={item} data={data} />
  63               )));
  64             </FlexWrap>
  65           )
  66         </FlexWrap>
  67         <FlexWrap>
  68           <WeeklyStatCard data={data} />
  69           <CategoryChart data={data} />
  70         </FlexWrap>
  71       </Wrapper>
  72     </Container>
  73   );
  74 }
  75
  76 export default Dashboard

```

Dashboard.jsx (Home Page)



```

client > src > pages > Workouts.jsx > Workouts
  65 const Workouts = () => [
  66   const [workoutData, setWorkoutData] = useState([
  67     ...
  68   ]);
  69   const [newWorkoutData, setNewWorkoutData] = useState([
  70     ...
  71   ]);
  72   const [category, setCategory] = useState('');
  73   const [workoutName, setWorkoutName] = useState('');
  74   const [sets, setSets] = useState('');
  75   const [reps, setReps] = useState('');
  76   const [weight, setWeight] = useState('');
  77   const [duration, setDuration] = useState('');
  78
  79   const handleInputChange = (e) => {
  80     const [name, value] = e.target;
  81     setNewWorkoutData({ ...newWorkoutData, [name]: value });
  82   };
  83
  84   const handleSubmit = (e) => {
  85     e.preventDefault();
  86
  87     setLoggedWorkouts([...loggedWorkouts, newWorkoutData]);
  88     // Clear the form fields after submission
  89     setNewWorkoutData({ category: '' });
  90     setWorkoutName('');
  91     setSets('');
  92     setReps('');
  93     setWeight('');
  94     setDuration('');
  95   };
  96
  97   return (
  98     <Container>
  99       <Wrapper>
  100         <Title>Workouts</Title>
  101         <Form>
  102           ...
  103         </Form>
  104       </Wrapper>
  105     </Container>
  106   );
  107 }
  108
  109 export default Workouts

```

Workouts.jsx (Workout Logging)

```

client > src > pages > Plans.jsx
const Plans = () => {
  return (
    <Container>
      <Title>Plans</Title>
      <FlexWrap>
        <Card>
          <Title>3-day Workout Split</Title>
          <Split3 src={split3img}/>
          <Card>
            <ul>
              <b>Day-1: CHEST/BACK</b>
              <li>Bench Press</li>
              <li>Incline Dumbbell Press</li>
              <li>Cable Lower Chest Fly</li>
              <li>Lat Pulldown</li>
              <li>Seated Cable Rows</li>
            </ul>
          </Card>
        <Card><ul><b>Day-2: REST</b></ul></Card>
        <Card>
          <ul>
            <b>Day-3: LEGS</b>
            <li>Squat Variation</li>
            <li>Hip Hinge Variation</li>
            <li>Leg Extension</li>
            <li>Leg Curl</li>
            <li>Calf Raise Variation</li>
          </ul>
        </Card>
      <Card><ul><b>Day-4: REST</b></ul></Card>
      <Card>
        <ul>
          <b>Day-5: SHOULDER/ARMS</b>
          <li>Should Press Variation</li>
        </ul>
      </Card>
    </FlexWrap>
  )
}

```

Plans.jsx (Workout Plan Cards)

```

client > src > pages > Forum.jsx
const Forum = () => {
  const [threads, setThreads] = useState([
    {
      id: 1,
      title: "Nutrition tips for gaining muscle?",
      author: "Prasenjeet Thul",
      date: "April 18, 2024",
      content: "I'm trying to gain muscle mass. Any nutrition tips or meal plans you can share would be great!",
      comments: [
        { id: 1, author: "Aditya Tripathi", date: "April 19, 2024", content: "Focus on protein-rich foods like chicken, fish, and eggs." },
        { id: 2, author: "GymKat24", date: "April 19, 2024", content: "You might want to consider a pre-workout supplement like BCAAs." }
      ]
    }
  ]);

  const [newThread, setNewThread] = useState({
    title: '',
    author: 'User', // Assuming default author is 'User'
    content: ''
  });

  const handleThreadChange = (e) => {
    const { name, value } = e.target;
    setNewThread({ ...newThread, [name]: value });
  };

  const handleThreadSubmit = (e) => {
    e.preventDefault();
    // Create new thread
    const newThreadId = discussionThreads.length + 1;
    const newThreadData = { ...newThread, id: newThreadId, date: new Date().toLocaleDateString(), author: 'User' };
    setDiscussionThreads([...discussionThreads, newThreadData]);
    setNewThread({ title: '', content: '' });
  };

  const handleCommentSubmit = (threadId, comment) => {
    // Add comment to the thread
    const updatedThreads = discussionThreads.map(thread =>
      thread.id === threadId
        ? { ...thread, comments: [...thread.comments, comment] }
        : thread
    );
    setDiscussionThreads(updatedThreads);
  };
}

const handleCommentSubmit = (threadId, comment) => {
  // Add comment to the thread
  const updatedThreads = discussionThreads.map(thread =>
    thread.id === threadId
      ? { ...thread, comments: [...thread.comments, comment] }
      : thread
  );
  setDiscussionThreads(updatedThreads);
}

```

Forum.jsx (Discussion Page)

```

client > src > pages > Products.jsx > [e] Products > [e] products.map() callback
  52 const Products = () => {
  53   const products = [
  54     {
  55       id: 3,
  56       name: "Yoga Mat",
  57       description: "High-quality yoga mat for comfortable yoga sessions.",
  58       price: "699",
  59       imageUrl: "https://m.media-amazon.com/images/I/71pjmgvnFUL.AC_UL480_FMwebp_QL65.jpg",
  60       buyLink: "https://www.amazon.in/Fitness-Mantra%C2%AE-Anti-Slip-Workout-Exercise/dp/B0CP8CB6B1/ref=sr_1_14?qid=1041785QH&sn&dib=eyJ2IjoiMS"
  61     },
  62     // Add more products as needed
  63   ];
  64
  65   return (
  66     <Container>
  67       <Title>Supplementary Products</Title>
  68       <FlexWrap>
  69         {products.map(product => (
  70           <Card>
  71             <div className="card">
  72               <div className="card-image">
  73                 <img src={product.imageUrl} alt={product.name} style={{ height: '250px', width: '100%' }} />
  74               <title>{product.name}</title>
  75             </div>
  76             <div className="card-content">
  77               <p>{product.description}</p>
  78               <p><strong>Price:</strong> {product.price}</p>
  79             </div>
  80             <div className="card-action">
  81               <a href={product.buyLink}>
  82                 target="_blank"
  83                 rel="noopener noreferrer"
  84                 style={{
  85                   display: 'inline-block',
  86                   padding: '10px 20px',
  87                   backgroundColor: '#007bff',
  88                 }}
  89               </a>
  90             </div>
  91           </Card>
  92         ))}
  93       </FlexWrap>
  94     </Container>
  95   );
  96 }
  97
  98
  99
  100
  101
  102
  103
  104
  105

```

Products.jsx - Supplementary Products

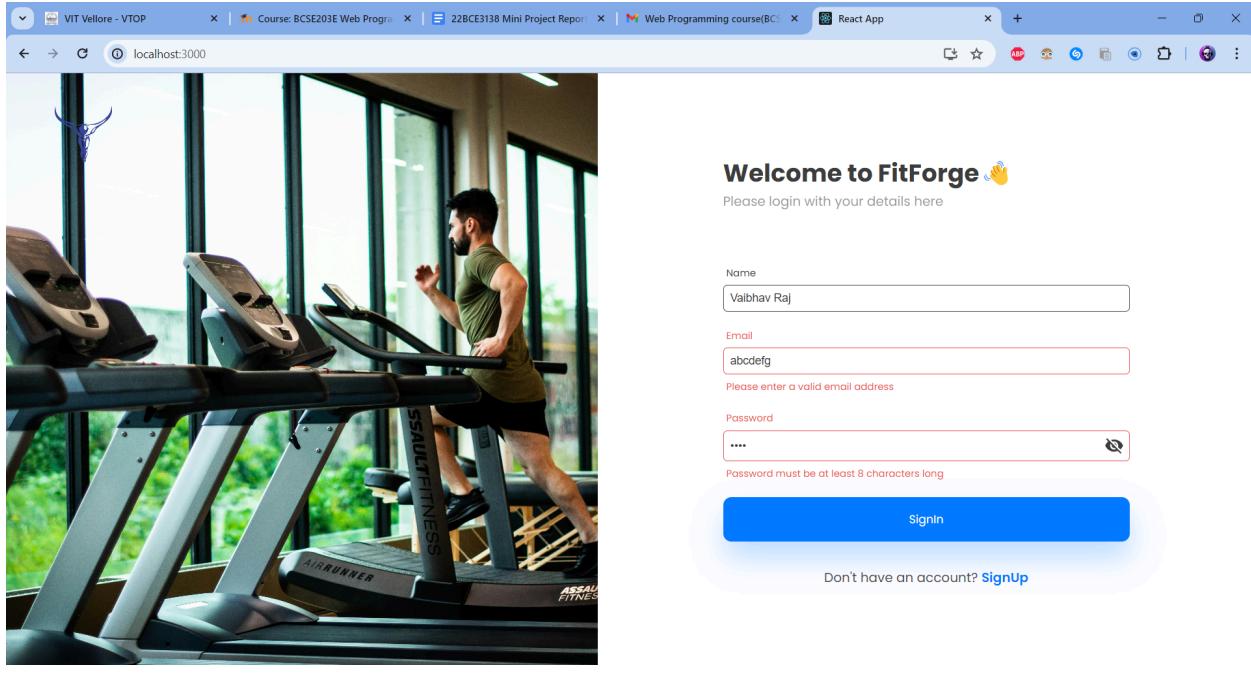
```

client > src > pages > Contact.jsx > [e] Wrapper
  67 const Contact = () => {
  68   const [message, setMessage] = useState('');
  69   const [submitted, setSubmitted] = useState(false);
  70
  71   const handleChange = (e) => {
  72     setMessage(e.target.value);
  73   };
  74
  75   const handleSubmit = (e) => {
  76     e.preventDefault();
  77     // You can implement a function here to send the message to your server
  78     // For now, we'll just set submitted to true
  79     setSubmitted(true);
  80     // Clear the message input after submission
  81     setMessage('');
  82   };
  83
  84
  85
  86
  87
  88
  89
  90
  91
  92
  93
  94
  95
  96
  97
  98
  99
  100
  101
  102
  103

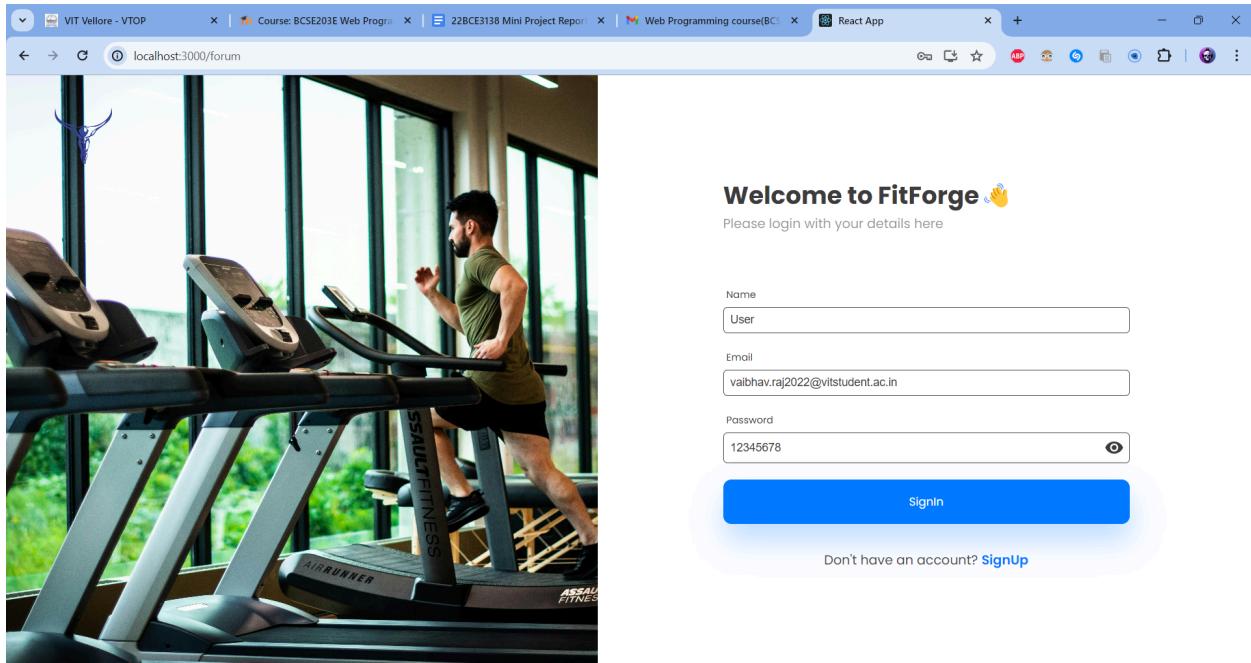
```

Contact.jsx - Feedback/Contact Us

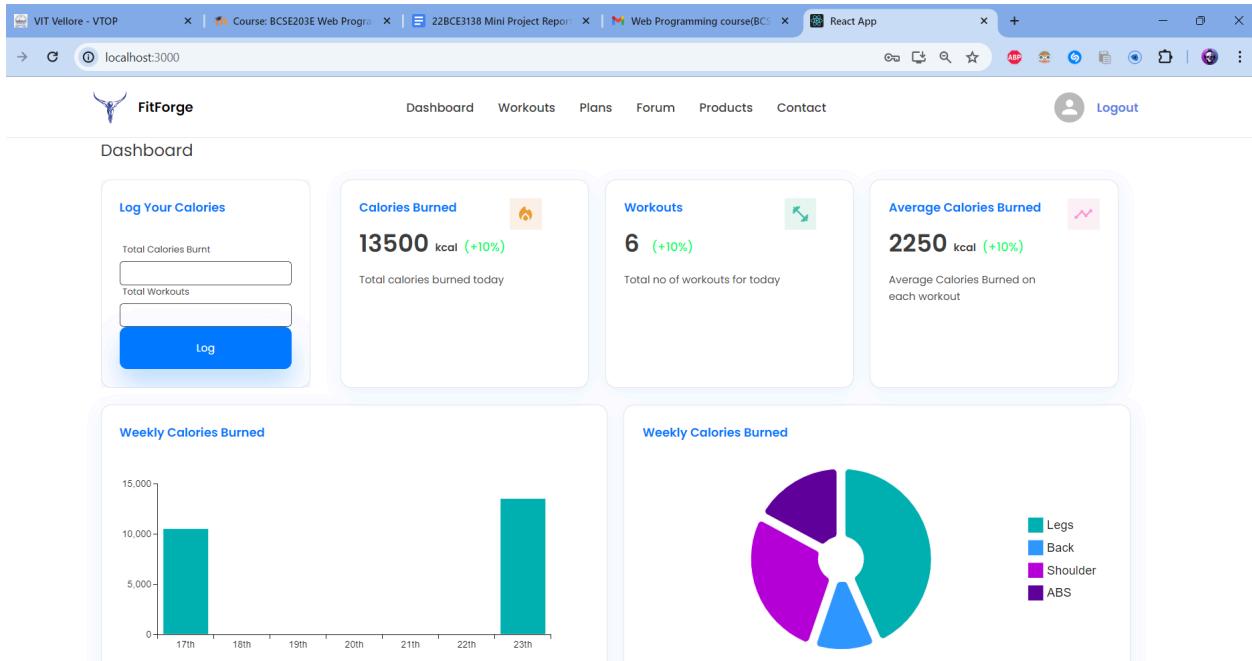
8. RESULTS



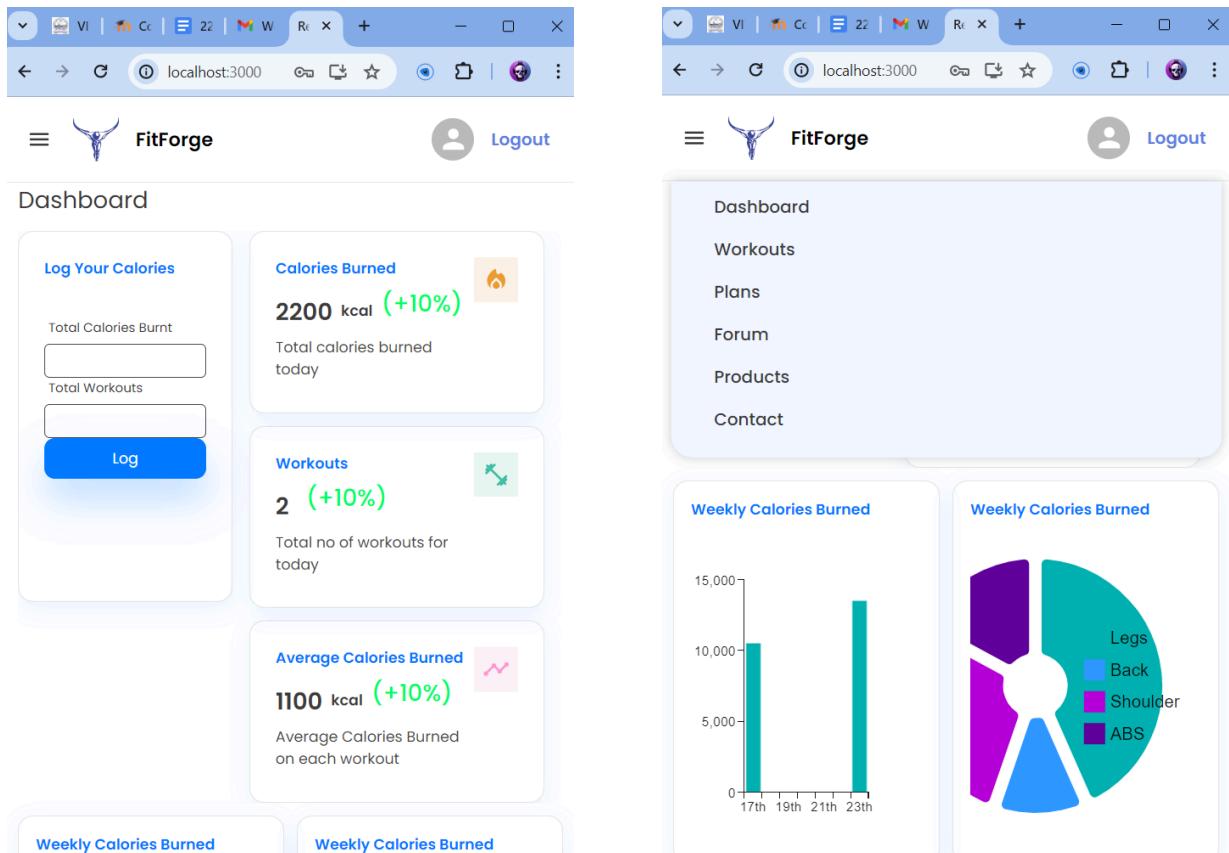
Login Form Validation



User Authentication



Dashboard - Calories Tracker (Home Page)



Responsiveness for Mobile/Tablet Devices

Track Your Workouts

Muscle Group
Exercise Name
Sets
Reps
Weight
Duration

Logged Workouts

- #Back
Lat Pulldown
Count: 3 sets X 15 reps
50 kg 15 min
- #Chest
Bench Press
Count: 3 sets X 12 reps
55 kg 15 min

Workout Logging

Plans

3-day Workout Split

Day-1: CHEST/BACK

- Bench Press
- Incline Dumbbell Press
- Cable Lower Chest Fly
- Lat Pulldown
- Seated Cable Rows

Day-2: REST

4-day Workout Split

Day-1: UPPER STRENGTH

- Bench Press
- Overhead Press
- Incline Dumbbell Press
- Lat Pulldown
- Lateral Raises

Day-2: LOWER STRENGTH

- Squats
- Lunges
- Hip Thrusts
- Leg Extensions

5-day Workout Split

Day-1: PUSH

- Bench Press
- Incline Dumbbell Press
- Tricep Extensions
- Shoulder Lateral Raises
- Shoulder Front Raises

Day-2: PULL

- Lat Pulldown
- Seated Cable Rows
- Bicep curls

Workout Plans (3 Scrollable Cards)

Navbar for Mobile/Tablet Devices

Forum

Thread Title

Best workout routines for beginners?

Author: Vaibhav Raj

Date: April 20, 2024

What are some good workout routines for beginners? I'm new to fitness and looking for some advice.

Comments

- Cbum_23 - April 21, 2024
For beginners, I recommend starting with a full-body workout routine three times a week.
- User - 4/23/2024
you can start by watching tutorials on youtube

Add a Comment

Comment

Reply

Forum Page - Replying to Other People's Threads

My website

Author: User

Date: 4/23/2024

I made this website for my Web Programming Mini Project

Comments

- User - 4/23/2024
yeah

Add a Comment

Comment

Reply

Forum Page - Creating Your Own Thread

The screenshot shows a web browser window with multiple tabs open. The active tab is 'localhost:3000/products' under the 'React App' tab. The page title is 'FitForge'. The main content area is titled 'Supplementary Products' and displays three items in cards:

- Protein Powder:** A grey tub of 'PROTEINBLAST MS3 BIOTIC PERFORMANCE WHEY' with a protein shaker cup icon. Description: 'Whey protein powder for muscle growth and recovery.' Price: ₹4,499. Buy Now button.
- Resistance Bands:** Three stacked resistance bands in yellow, blue, and green, each labeled 'BOLDFIT'. Description: 'Set of resistance bands for home workouts.' Price: ₹279. Buy Now button.
- Yoga Mat:** A blue rolled-up yoga mat. Description: 'High-quality yoga mat for comfortable yoga sessions.' Price: ₹698. Buy Now button.

Supplementary Products

The screenshot shows a web browser window for 'amazon.in' with the search term 'resistance bands' entered. The product page for 'Boldfit Resistance Bands Mini Loop Hip Band' is displayed. The product image shows three resistance bands (yellow, blue, and green) stacked together. The product details include:

- Name:** Boldfit Resistance Bands Mini Loop Hip Band
- Description:** Toning Exercise Band for Gym Booty Belt Latex Band Thera Band Theraband for Fitness, Multicolor
- Rating:** 4.1 stars from 3,958 reviews
- Offer:** -38% *₹279 (M.R.P.: ₹449)
- Stock Status:** In stock
- Delivery:** FREE delivery Friday, 26 April on orders dispatched by Amazon over ₹499. Details
- Price:** ₹279
- Buy Options:** Add to Cart, Buy Now

Supplementary Products - Links Redirecting to Other Websites

Contact Us

Write Your Message/Feedback
I like your site!

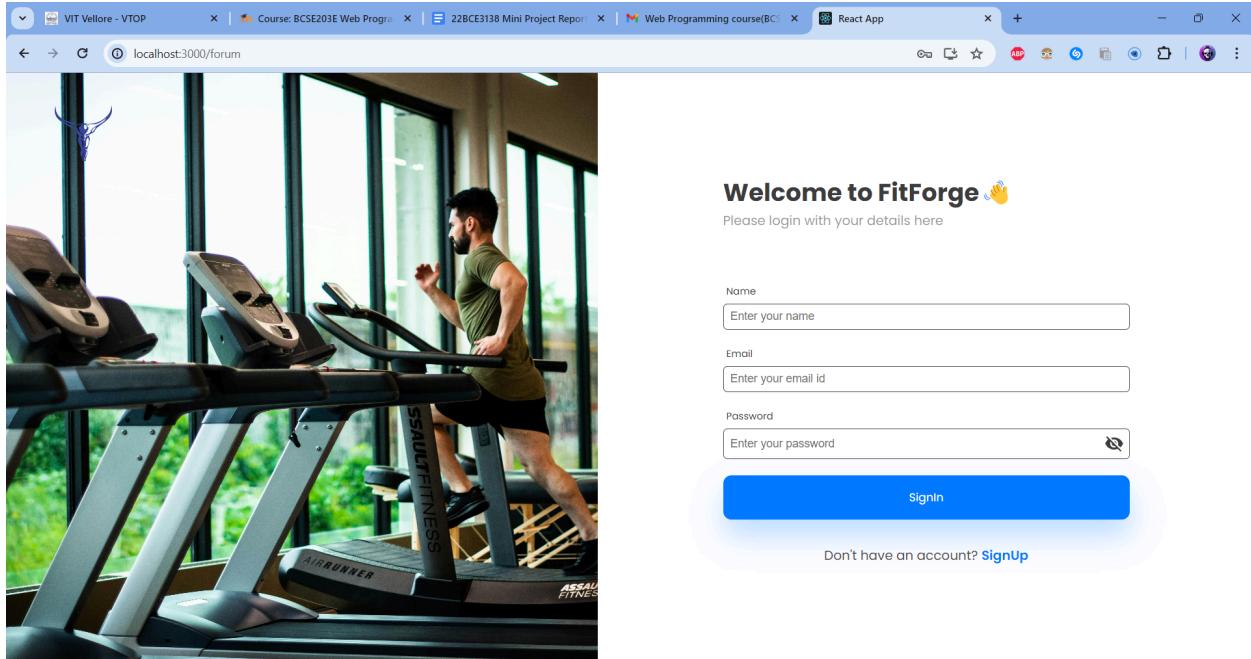
Submit

Contact Us - Asking for feedback

Contact Us

Message received! We'll respond to you shortly.

Contact Us - Message/Feedback Received



Navbar - Logout -> Back to User Authentication

9. CONCLUSION

In conclusion, the development of the fitness tracker website has been a significant endeavor, aiming to provide users with a comprehensive platform for tracking their fitness journey, engaging with a community of enthusiasts, and accessing valuable resources to support their fitness goals. Through the utilization of various technologies and methodologies, we have successfully built a feature-rich and user-friendly application that addresses the needs and requirements of our target audience.

Throughout the project, we have employed a combination of React.js, styled-components, Material-UI, and additional technologies to create a modern and intuitive user interface. React.js has served as the backbone of our application, providing a robust framework for building dynamic and interactive components. styled-components has enabled us to manage component-level styling effectively, ensuring a consistent and visually appealing design across the website. Material-UI has enriched our UI with pre-designed components and icons, following the principles of Material Design for a cohesive user experience.

The project's methodology involved meticulous planning, iterative development, and continuous refinement to meet project goals and deliverables. We adopted Agile practices, such as sprint-based development and continuous integration, to ensure efficient collaboration and timely delivery of features. The modular approach allowed us to break down the project into manageable components, facilitating easier development, testing, and maintenance.

Key features of the fitness tracker website include user authentication, a dashboard with graphs and charts, workout logging, predefined workout plans, a community forum, supplementary products section, and a contact us page. These features provide users with a holistic fitness tracking experience, empowering them to set and achieve their fitness goals with ease.

Incorporating user feedback and testing at each stage of development has been crucial in refining the website's functionality and user experience. We have ensured that the website is responsive and accessible, catering to users across different devices and accessibility needs.

In conclusion, the fitness tracker website stands as a testament to the power of modern web technologies in creating innovative solutions to real-world problems. It is our hope that this website will inspire and motivate users on their fitness journey, fostering a community of health and wellness enthusiasts.