



MIT Art, Design and Technology University

MIT School of Computing, Pune

Department of Information Technology

Lab Manual

Practical - Web Programming

Class - S.Y. (SEM-II), DA

Batch - TY-IT-SMAD

Name of the Student

Mr. Ritesh J Prajapati

A.Y. 2024 – 2025 (SEM-II)

Assignment No 1: Project Assignment based on Unit-I and Unit-II

Project-I

Project Title: ReactJS E-Commerce Website with Page Routing

Project Statement:

Objective:

Develop a fully functional E-commerce website using ReactJS that includes page routing functionality for smooth navigation between different sections like Home, Product Listing, Product Details, Cart, and Checkout.

Project Features:

1. Navigation Bar with Routing:
 - Create a Navigation Bar at the top of the website with links to:
 - Home: Displays featured products and general information about the store.
 - Products: Lists all available products for purchase.
 - Cart: Displays the products added to the shopping cart.
 - Checkout: Allows the user to review and complete the purchase.
 - Profile: Displays user account information and order history.
 - Implement React Router for routing between these pages.
2. Home Page:
 - The Home page should showcase highlighted products or categories.
 - Implement a product carousel or featured section.
 - Include a search bar to allow users to search for products directly.
3. Product Listing Page:
 - Create a Product Listing page that fetches products dynamically from an API or static JSON file.
 - Each product in the list should show:
 - Product image.
 - Product name.
 - Price.
 - A button to add the product to the cart.
 - Implement React Router links on product items to navigate to the Product Details page.
4. Product Details Page:
 - When a user clicks on a product, navigate them to the Product Details page.
 - The Product Details page should display:
 - Full-size product image.
 - Detailed product description.

- Price.
- Add to Cart button.
- Allow users to select product quantity before adding it to the cart.
- 5. Cart Page:
 - Implement a Cart page that displays products added to the cart.
 - Each cart item should include:
 - Product name.
 - Product quantity.
 - Price per item and total cost for the item.
 - A button to remove items from the cart.
 - Display a total price for all items in the cart.
 - Include buttons for Continue Shopping and Proceed to Checkout.
- 6. Checkout Page:
 - Create a Checkout page where users can review their cart, enter shipping information, and confirm the order.
 - Display an order summary with total cost, shipping details, and payment options.
 - Provide a Place Order button to finalize the transaction.
- 7. User Profile Page:
 - Create a Profile page where users can view their account information and order history.
 - Include a form to update the profile details (name, email, shipping address).
- 8. State Management with Context API/Redux:
 - Use React Context API or Redux to manage the global state of the cart, allowing users to add, remove, and update quantities of products across different pages.
 - Ensure that the cart data is preserved during navigation between pages.
- 9. Responsive Design:
 - Ensure the website is responsive and works well on both desktop and mobile devices using CSS Flexbox or CSS Grid.

Technical Requirements:

- ReactJS: For building the user interface.
- React Router: For implementing page routing functionality.
- State Management: Use React Context API or Redux to manage cart state and other global data.
- API Integration: Fetch products from a static JSON file or a mock API (e.g., JSONPlaceholder or a custom API).
- Responsive Styling: Use CSS/SCSS to ensure the website is mobile-friendly.

	<ul style="list-style-type: none"> ● Optional: Add localStorage or sessionStorage to persist cart data between page reloads. <p>Expected Outcome:</p> <p>By completing this project, you will:</p> <ul style="list-style-type: none"> ● Learn how to implement page routing in a ReactJS application using React Router. ● Build a dynamic E-commerce website with features like product listings, cart management, and checkout functionality. ● Gain experience with state management and ensuring the cart data persists across different pages. ● Develop a responsive, user-friendly interface that functions well on both mobile and desktop screens.
Assignment No 2: Project Assignment based on Unit-III, IV and V	
Project-II	<p>Project Title: React Native E-Commerce Application</p> <p>Project Statement:</p> <p>Objective: Build a fully functional E-commerce mobile application using React Native that integrates navigation, core components, user interaction, dynamic list rendering, and state management.</p> <p>Project Features:</p> <ol style="list-style-type: none"> 1. User Authentication: <ul style="list-style-type: none"> ○ Implement a simple login screen where users can enter their credentials (username and password). ○ After successful login, navigate to the Home Screen. 2. Navigation: <ul style="list-style-type: none"> ○ Set up react-navigation to allow users to navigate between the following screens using StackNavigator: <ul style="list-style-type: none"> ■ Home Screen: Displays a list of products. ■ Product Details Screen: Shows details of the selected product. ■ Cart Screen: Displays products added to the shopping cart. ■ Profile Screen: Displays user profile details. ■ Settings Screen: Allows the user to modify their account settings. 3. Core Components & UI: <ul style="list-style-type: none"> ○ Use core React Native components such as View, Text, Image,

StyleSheet, and Button to build a user-friendly interface.

- Apply Flexbox for layout, ensuring responsiveness across different screen sizes.

4. Product List and Cart:

- On the Home Screen, render the list of products using FlatList. Each item should display a product image, name, and price.
- Implement a button for each product to add it to the Cart.

5. Handling User Interactions:

- Implement touch events to handle interactions such as adding products to the cart, navigating to the product details, and updating the cart.
- Use onPress to trigger actions like adding a product to the cart or navigating between screens.

6. Cart Management:

- Use React Context API for global state management to handle the cart's data (adding/removing products).
- Show the cart's content and total price on the Cart Screen.

7. Product Details:

- When a user clicks on a product in the Home Screen, navigate to the Product Details Screen and display detailed information about the product, such as the description, image, and price.

8. Responsive Design:

- Ensure that the app is responsive, and elements adjust properly to different screen sizes using Flexbox.

9. Rendering Lists with SectionList:

- On the Home Screen, group products by categories (e.g., electronics, clothing) and use SectionList to render them in separate sections with headers.

Technical Requirements:

- React Native with Expo CLI for development and testing.
- React Navigation for screen transitions.
- State Management using React Context API or useState, useReducer.
- Dynamic data rendering using FlatList and SectionList.
- Styling using StyleSheet and Flexbox for layout.

Expected Outcome:

By completing this project, students will gain hands-on experience with:

- Setting up and using React Native and React Navigation.
- Building a fully interactive mobile application with dynamic data rendering and state management.

- | | |
|--|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | <ul style="list-style-type: none">● Applying Flexbox for responsive layouts and managing user interactions.● Creating a simple e-commerce app with multi-screen navigation, product lists, and state-driven cart functionality. |
|--|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Experiment No.2

Problem Statement:

To create a cross-platform mobile application named **OneShot** using **React Native** that allows users to browse photography content, learn from tutorials, connect with a community, and manage login/registration – all within a sleek mobile interface.

Objective:

Build a mobile app using React Native with screens for Home, Gallery, Tutorials, Community, About Us, Contact, Login, and Register.

Use **React Navigation** for screen transitions.

Ensure smooth and native-like performance on both Android and iOS devices.

Provide a visually engaging and user-centric mobile experience.

Theory:

The mobile application version of OneShot is built using **React Native**, a modern open-source framework that allows developers to create cross-platform apps using JavaScript and React. This ensures that the same codebase can run on both Android and iOS, saving development time and resources.

1. Why React Native?

- **Cross-platform compatibility:** Single codebase for Android and iOS.
- **Fast refresh:** Enables instant feedback while developing.
- **Strong community and plugin support:** Includes tools for navigation, storage, and animations.
- **JavaScript ecosystem:** Developers familiar with web technologies can quickly adapt.

2. App Architecture:

React Native components are the building blocks of the application. The architecture follows a **component-based structure**, where each screen or functionality is encapsulated in a React component. Major components/screens include:

- **Home Screen** – Displays welcome message and routing buttons.

- **Gallery Screen** – Uses a scrollable view or flat list to showcase images.
- **Tutorial Screen** – Embeds video players or external links for photography guides.
- **Community Forum** – Uses text inputs and flat list for user interaction.
- **About Us Screen** – Uses static content and images.
- **Login & Registration** – Form components with validation logic.

3. Key Libraries Used:

- **React Navigation**: Handles screen transitions and navigation menus.
- **AsyncStorage** (or newer packages like `@react-native-async-storage/async-storage`): For persistent local storage.
- **Axios** or **Fetch API**: For future integration with backend (optional).
- **Styled Components** or basic CSS-in-JS for styling React Native elements.

4. UI/UX Design Considerations:

- The app uses a minimalistic and intuitive layout inspired by the website.
- Emphasis on **touch-friendly components**, **gesture navigation**, and **visual clarity**.
- Responsive design within the mobile ecosystem is achieved using `flexbox` and `Dimensions` APIs.
- The visual branding (logo, fonts, color scheme) is preserved from the web version for consistency.

Code:

A. Home page:

code:

```
import { View, Text, StyleSheet } from 'react-native';
import { Tabs } from 'expo-router';
import { Chrome as Home, Image as ImageIcon, BookOpen, Users, Info as InfoIcon, Login } from 'lucide-react-native';
import { useSafeAreaInsets } from 'react-native-safe-area-context';
import { Platform } from 'react-native';
import React from 'react';
```



```

export default function TabLayout() {
  const insets = useSafeAreaInsets();

  return (
    <Tabs
      screenOptions={{
        headerShown: false,
        tabBarStyle: {
          ...styles.tabBar,
          height: Platform.OS === 'ios' ? 80 : 60,
          paddingBottom: Platform.OS === 'ios' ? 20 : 10,
        },
        tabBarLabelStyle: styles.tabBarLabel,
        tabBarActiveTintColor: '#FFCB45', // Gold/yellow color
        tabBarInactiveTintColor: '#AAAAAA',
      }}
    >
      <Tabs.Screen
        name="index"
        options={{
          title: 'Home',
          tabBarIcon: ({ color, size }) => (
            <Home color={color} size={size} />
          ),
        }}
      />
      <Tabs.Screen
        name="gallery"
        options={{
          title: 'Gallery',
          tabBarIcon: ({ color, size }) => (
            <ImageIcon color={color} size={size} />
          ),
        }}
      />
      <Tabs.Screen
        name="tutorials"
        options={{
          title: 'Tutorials',
          tabBarIcon: ({ color, size }) => (
            <BookOpen color={color} size={size} />
          ),
        }}
      />
      <Tabs.Screen
        name="community"
        options={{

```

```

        title: 'Community',
        tabBarIcon: ({ color, size }) => (
          <Users color={color} size={size} />
        ),
      },
    ],
  ],
),
</Tabs>
);
}

```

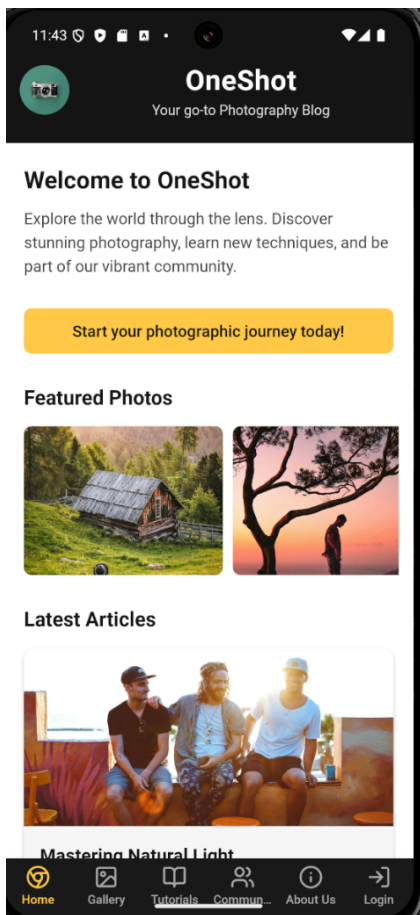
```

const styles = StyleSheet.create({
  tabBar: {
    backgroundColor: '#1A1A1A',
    borderTopWidth: 0,
  },
  tabBarLabel: {
    fontSize: 12,
    fontFamily: 'Roboto-Medium',
  },
});

```

Output:

A. Index/Home page output:



Code:

B. Gallery page:

code:

```
import { View, Text, StyleSheet, Image, FlatList, TouchableOpacity } from 'react-native';
import { SafeAreaView } from 'react-native-safe-area-context';
import { useSafeAreaInsets } from 'react-native-safe-area-context';
import React from 'react';
const galleryImages = [
  { id: '1', url:
'https://images.pexels.com/photos/1308881/pexels-photo-1308881.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2', title: 'Mountain Sunset' },
  { id: '2', url:
'https://images.pexels.com/photos/1287145/pexels-photo-1287145.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2', title: 'Ocean Waves' },
  { id: '3', url:
'https://images.pexels.com/photos/2387418/pexels-photo-2387418.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2', title: 'Forest Path' },
```

```

    { id: '4', url:
'https://images.pexels.com/photos/346529/pexels-photo-346529.jpeg?auto=compress&cs=tinysrgb&w=1260
&h=750&dpr=2', title: 'Urban Architecture' },
    { id: '5', url:
'https://images.pexels.com/photos/1428277/pexels-photo-1428277.jpeg?auto=compress&cs=tinysrgb&w=126
0&h=750&dpr=2', title: 'Desert Landscape' },
    { id: '6', url:
'https://images.pexels.com/photos/2539462/pexels-photo-2539462.jpeg?auto=compress&cs=tinysrgb&w=126
0&h=750&dpr=2', title: 'Night Sky' },
  ];

```

```

export default function GalleryScreen() {
  const insets = useSafeAreaInsets();

  const renderItem = ({ item }) => (
    <TouchableOpacity style={styles.imageContainer}>
      <Image source={{ uri: item.url }} style={styles.image} />
      <View style={styles.imageOverlay}>
        <Text style={styles.imageTitle}>{item.title}</Text>
      </View>
    </TouchableOpacity>
  );

  return (
    <View style={styles.container}>
      <View style={[styles.header, { paddingTop: insets.top }]}>
        <Text style={styles.headerTitle}>Gallery</Text>
        <Text style={styles.headerSubtitle}>Explore our curated collection</Text>
      </View>

      <FlatList
        data={galleryImages}
        renderItem={renderItem}
        keyExtractor={(item) => item.id}
        numColumns={2}
        contentContainerStyle={styles.galleryList}
      />
    </View>
  );
}

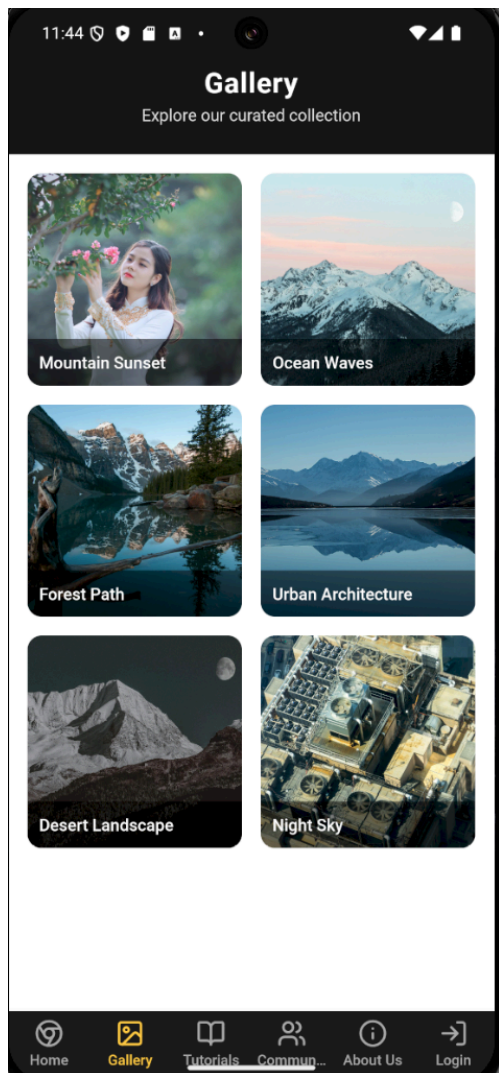
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: 'FFFFFF',
  },
  header: {

```

```
    backgroundColor: '#121212',
    padding: 16,
    paddingBottom: 24,
    alignItems: 'center',
  },
  headerTitle: {
    color: '#FFFFFF',
    fontSize: 24,
    fontFamily: 'Roboto-Bold',
    letterSpacing: 0.5,
  },
  headerSubtitle: {
    color: '#E0E0E0',
    fontSize: 14,
    fontFamily: 'Roboto-Regular',
    marginTop: 2,
  },
  galleryList: {
    padding: 8,
  },
  imageContainer: {
    flex: 1,
    margin: 8,
    borderRadius: 12,
    overflow: 'hidden',
    height: 180,
    position: 'relative',
  },
  image: {
    width: '100%',
    height: '100%',
  },
  imageOverlay: {
    position: 'absolute',
    bottom: 0,
    left: 0,
    right: 0,
    backgroundColor: 'rgba(0, 0, 0, 0.5)',
    padding: 10,
  },
  imageTitle: {
    color: '#FFFFFF',
    fontSize: 14,
    fontFamily: 'Roboto-Medium',
  },
});
```

Output:

B. Gallery page output:



Code:

C. Tutorials page:

```
code:import { View, Text, StyleSheet, Image, ScrollView, TouchableOpacity } from
'react-native';

import { useSafeAreaInsets } from 'react-native-safe-area-context';
import { BookOpen, Clock, Star } from 'lucide-react-native';
import React from 'react';
const tutorials = [
  {
    id: '1',
    title: 'Mastering Camera Exposure',
```

```

    image:
'https://images.pexels.com/photos/1092671/pexels-photo-1092671.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2',
    duration: '15 min',
    level: 'Beginner',
    description: 'Learn the fundamentals of exposure and how to properly expose your photos in any lighting condition.'
  },
  {
    id: '2',
    title: 'Portrait Photography Lighting',
    image:
'https://images.pexels.com/photos/2773977/pexels-photo-2773977.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2',
    duration: '20 min',
    level: 'Intermediate',
    description: 'Master the art of portrait lighting with these easy-to-follow techniques for stunning portraits.'
  },
  {
    id: '3',
    title: 'Landscape Photography Tips',
    image:
'https://images.pexels.com/photos/2325446/pexels-photo-2325446.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2',
    duration: '25 min',
    level: 'Advanced',
    description: 'Capture breathtaking landscapes with these professional tips and composition techniques.'
  }
];

```

```

export default function TutorialScreen() {
  const insets = useSafeAreaInsets();

  return (
    <View style={styles.container}>
      <View style={[styles.header, { paddingTop: insets.top }]}>
        <Text style={styles.headerTitle}>Tutorials</Text>
        <Text style={styles.headerSubtitle}>Learn photography skills</Text>
      </View>

      <ScrollView style={styles.content}>
        <View style={styles.featuredTutorial}>
          <Image
            source={{ uri:
'https://images.pexels.com/photos/3612343/pexels-photo-3612343.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2' }}
            style={styles.featuredImage}

```

```

/>
<View style={styles.featuredContent}>
  <View style={styles.featuredBadge}>
    <Text style={styles.featuredBadgeText}>Featured</Text>
  </View>
  <Text style={styles.featuredTitle}>Complete Guide to Photography</Text>
  <Text style={styles.featuredDescription}>
    From basics to advanced techniques, this comprehensive guide covers everything you need to know
    about photography.
  </Text>
  <TouchableOpacity style={styles.featuredButton}>
    <Text style={styles.featuredButtonText}>Start Learning</Text>
  </TouchableOpacity>
</View>
</View>

<Text style={styles.sectionTitle}>Popular Tutorials</Text>

{tutorials.map(tutorial => (
  <TouchableOpacity key={tutorial.id} style={styles.tutorialCard}>
    <Image source={{ uri: tutorial.image }} style={styles.tutorialImage} />
    <View style={styles.tutorialContent}>
      <Text style={styles.tutorialTitle}>{tutorial.title}</Text>
      <Text style={styles.tutorialDescription}>{tutorial.description}</Text>
      <View style={styles.tutorialMeta}>
        <View style={styles.metaItem}>
          <Clock size={16} color="#666666" />
          <Text style={styles.metaText}>{tutorial.duration}</Text>
        </View>
        <View style={styles.metaItem}>
          <BookOpen size={16} color="#666666" />
          <Text style={styles.metaText}>{tutorial.level}</Text>
        </View>
      </View>
    </View>
  </TouchableOpacity>
)}}
</ScrollView>
</View>
);
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: 'FFFFFF',
  },

```



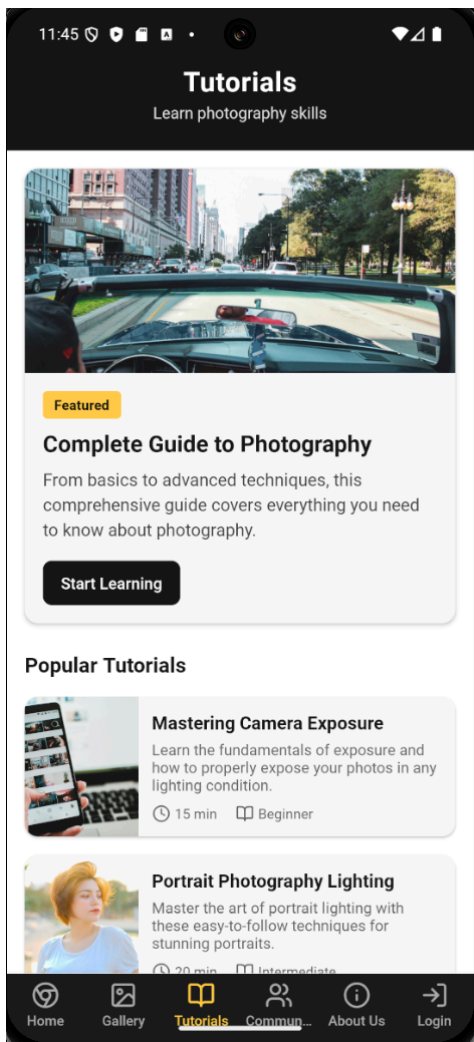
```
header: {
  backgroundColor: '#121212',
  padding: 16,
  paddingBottom: 24,
  alignItems: 'center',
},
headerTitle: {
  color: '#FFFFFF',
  fontSize: 24,
  fontFamily: 'Roboto-Bold',
  letterSpacing: 0.5,
},
headerSubtitle: {
  color: '#E0E0E0',
  fontSize: 14,
  fontFamily: 'Roboto-Regular',
  marginTop: 2,
},
content: {
  flex: 1,
  padding: 16,
},
featuredTutorial: {
  marginBottom: 24,
  borderRadius: 12,
  overflow: 'hidden',
  backgroundColor: '#F9F9F9',
  shadowColor: '#000',
  shadowOffset: { width: 0, height: 2 },
  shadowOpacity: 0.1,
  shadowRadius: 4,
  elevation: 2,
},
featuredImage: {
  width: '100%',
  height: 180,
},
featuredContent: {
  padding: 16,
},
featuredBadge: {
  backgroundColor: '#FFCB45',
  paddingHorizontal: 10,
  paddingVertical: 4,
  borderRadius: 4,
  alignSelf: 'flex-start',
  marginBottom: 8,
```

```
},
featuredBadgeText: {
  color: '#121212',
  fontSize: 12,
  fontFamily: 'Roboto-Bold',
},
featuredTitle: {
  fontSize: 20,
  fontFamily: 'Roboto-Bold',
  color: '#121212',
  marginBottom: 8,
},
featuredDescription: {
  fontSize: 15,
  fontFamily: 'Roboto-Regular',
  color: '#454545',
  lineHeight: 22,
  marginBottom: 16,
},
featuredButton: {
  backgroundColor: '#121212',
  paddingVertical: 10,
  paddingHorizontal: 16,
  borderRadius: 8,
  alignSelf: 'flex-start',
},
featuredButtonText: {
  color: 'FFFFFF',
  fontSize: 14,
  fontFamily: 'Roboto-Medium',
},
sectionTitle: {
  fontSize: 18,
  fontFamily: 'Roboto-Bold',
  color: '#121212',
  marginBottom: 16,
},
tutorialCard: {
  flexDirection: 'row',
  backgroundColor: '#F9F9F9',
  borderRadius: 12,
  overflow: 'hidden',
  marginBottom: 16,
  shadowColor: '#000',
  shadowOffset: { width: 0, height: 1 },
  shadowOpacity: 0.08,
  shadowRadius: 2,
```

```
elevation: 1,
},
tutorialImage: {
  width: 100,
  height: '100%',
},
tutorialContent: {
  flex: 1,
  padding: 12,
},
tutorialTitle: {
  fontSize: 16,
  fontFamily: 'Roboto-Bold',
  color: '#121212',
  marginBottom: 4,
},
tutorialDescription: {
  fontSize: 13,
  fontFamily: 'Roboto-Regular',
  color: '#666666',
  marginBottom: 8,
},
tutorialMeta: {
  flexDirection: 'row',
  justifyContent: 'flex-start',
},
metaItem: {
  flexDirection: 'row',
  alignItems: 'center',
  marginRight: 16,
},
metaText: {
  fontSize: 12,
  fontFamily: 'Roboto-Regular',
  color: '#666666',
  marginLeft: 4,
},
});
```

Output:

C. Tutorials page output:



Code:

D. Community page:

code:

```
import { View, Text, StyleSheet, Image, FlatList, TouchableOpacity } from 'react-native';
import { useSafeAreaInsets } from 'react-native-safe-area-context';
import { Heart, MessageCircle, Share2 } from 'lucide-react-native';
import React from 'react';
const posts = [
  {
    id: '1',
    user: {
      name: 'Emily Chen',
      avatar:
'https://images.pexels.com/photos/774909/pexels-photo-774909.jpeg?auto=compress&cs=tinysrgb&w=1260
&h=750&dpr=2',
    },
  },
];
```

```

    image:
      'https://images.pexels.com/photos/1366919/pexels-photo-1366919.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2',
      caption: 'Golden hour in the mountains. Shot with Canon EOS R5, 24-70mm f/2.8.',
      likes: 243,
      comments: 18,
      timestamp: '2h ago'
    },
    {
      id: '2',
      user: {
        name: 'Mark Johnson',
        avatar:
          'https://images.pexels.com/photos/220453/pexels-photo-220453.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2',
      },
      image:
        'https://images.pexels.com/photos/3075993/pexels-photo-3075993.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2',
      caption: 'Urban exploration in New York City. Playing with reflections and symmetry.',
      likes: 187,
      comments: 24,
      timestamp: '5h ago'
    },
    {
      id: '3',
      user: {
        name: 'Sarah Williams',
        avatar:
          'https://images.pexels.com/photos/415829/pexels-photo-415829.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2',
      },
      image:
        'https://images.pexels.com/photos/2437291/pexels-photo-2437291.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2',
      caption: 'Wildlife photography is all about patience. Spotted this beauty after waiting for 3 hours.',
      likes: 312,
      comments: 42,
      timestamp: '1d ago'
    }
  ];

```

```

export default function CommunityScreen() {
  const insets = useSafeAreaInsets();

  const renderPost = ({ item }) => (
    <View style={styles.postContainer}>

```

```

<View style={styles.postHeader}>
  <Image source={{ uri: item.user.avatar }} style={styles.avatar} />
  <View style={styles.postHeaderText}>
    <Text style={styles.userName}>{item.user.name}</Text>
    <Text style={styles.timestamp}>{item.timestamp}</Text>
  </View>
</View>

```

```

<Image source={{ uri: item.image }} style={styles.postImage} />

```

```

<View style={styles.postActions}>
  <TouchableOpacity style={styles.actionButton}>
    <Heart size={22} color="#666666" />
    <Text style={styles.actionText}>{item.likes}</Text>
  </TouchableOpacity>
  <TouchableOpacity style={styles.actionButton}>
    <MessageCircle size={22} color="#666666" />
    <Text style={styles.actionText}>{item.comments}</Text>
  </TouchableOpacity>
  <TouchableOpacity style={styles.actionButton}>
    <Share2 size={22} color="#666666" />
  </TouchableOpacity>
</View>

```

```

<View style={styles.postContent}>
  <Text style={styles.caption}>{item.caption}</Text>
</View>

```

```

</View>

```

```

);

```

```

return (

```

```

  <View style={styles.container}>
    <View style={[styles.header, { paddingTop: insets.top }]}>
      <Text style={styles.headerTitle}>Community</Text>
      <Text style={styles.headerSubtitle}>Connect with photographers</Text>
    </View>

```

```

    <FlatList
      data={posts}
      renderItem={renderPost}
      keyExtractor={({item}) => item.id}
      contentContainerStyle={styles.listContent}
    />

```

```

  </View>

```

```

);

```

```

}

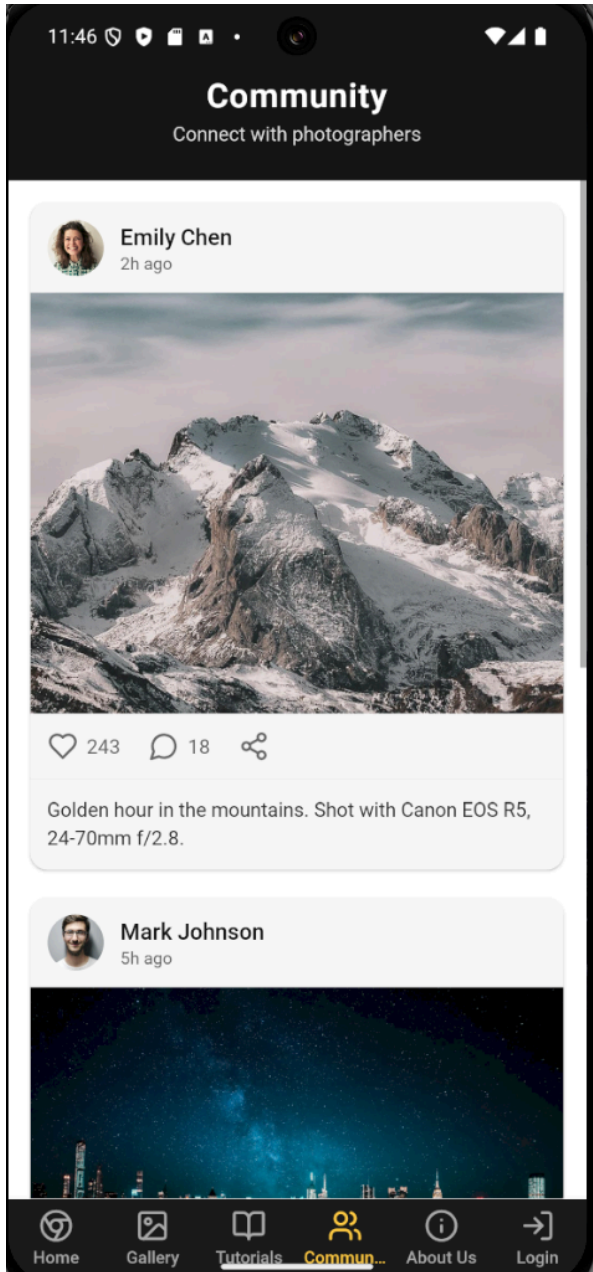
```

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#FFFFFF',
  },
  header: {
    backgroundColor: '#121212',
    padding: 16,
    paddingBottom: 24,
    alignItems: 'center',
  },
  headerTitle: {
    color: '#FFFFFF',
    fontSize: 24,
    fontFamily: 'Roboto-Bold',
    letterSpacing: 0.5,
  },
  headerSubtitle: {
    color: '#E0E0E0',
    fontSize: 14,
    fontFamily: 'Roboto-Regular',
    marginTop: 2,
  },
  listContent: {
    padding: 16,
  },
  postContainer: {
    backgroundColor: '#F9F9F9',
    borderRadius: 12,
    overflow: 'hidden',
    marginBottom: 20,
    shadowColor: '#000',
    shadowOffset: { width: 0, height: 1 },
    shadowOpacity: 0.08,
    shadowRadius: 2,
    elevation: 1,
  },
  postHeader: {
    flexDirection: 'row',
    alignItems: 'center',
    padding: 12,
  },
  avatar: {
    width: 40,
    height: 40,
    borderRadius: 20,
    marginRight: 12,
  },
});
```

```
},
postHeaderText: {
  flex: 1,
},
userName: {
  fontSize: 16,
  fontFamily: 'Roboto-Medium',
  color: '#121212',
},
timestamp: {
  fontSize: 12,
  fontFamily: 'Roboto-Regular',
  color: '#666666',
},
postImage: {
  width: '100%',
  height: 300,
},
postActions: {
  flexDirection: 'row',
  padding: 12,
  borderBottomWidth: 1,
  borderBottomColor: '#EEEEEE',
},
actionButton: {
  flexDirection: 'row',
  alignItems: 'center',
  marginRight: 20,
},
actionText: {
  fontSize: 14,
  fontFamily: 'Roboto-Regular',
  color: '#666666',
  marginLeft: 6,
},
postContent: {
  padding: 12,
},
caption: {
  fontSize: 14,
  fontFamily: 'Roboto-Regular',
  color: '#333333',
  lineHeight: 20,
},
});
```


Output:

D. Community page output:



Code:

E. About us page:

```
code:import { View, Text, StyleSheet, Image, ScrollView, TouchableOpacity } from
'react-native';
```

```
import { useSafeAreaInsets } from 'react-native-safe-area-context';
import { Mail, Instagram, Twitter, Facebook } from 'lucide-react-native';
```

```

import React from 'react';
export default function AboutScreen() {
  const insets = useSafeAreaInsets();

  return (
    <View style={styles.container}>
      <View style={{styles.header, { paddingTop: insets.top }}}>
        <Text style={styles.headerTitle}>About Us</Text>
        <Text style={styles.headerSubtitle}>Our story and mission</Text>
      </View>

      <ScrollView style={styles.content}>
        <View style={styles.storySection}>
          <Image
            source={{ uri:
'https://images.pexels.com/photos/3184339/pexels-photo-3184339.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2' }}
            style={styles.storyImage}
          />
          <Text style={styles.sectionTitle}>Our Story</Text>
          <Text style={styles.storyText}>
            OneShot was founded in 2022 by a group of passionate photographers who wanted to create a
platform where photography enthusiasts of all levels could learn, share, and grow together.
          </Text>
          <Text style={styles.storyText}>
            What started as a small blog has grown into a vibrant community of photographers from around the
world, all united by their love for capturing moments through the lens.
          </Text>
        </View>

        <View style={styles.missionSection}>
          <Text style={styles.sectionTitle}>Our Mission</Text>
          <Text style={styles.missionText}>
            At OneShot, we believe that photography is more than just taking pictures—it's about storytelling,
creativity, and connecting with the world around us. Our mission is to inspire and empower photographers to
develop their skills, find their unique voice, and share their perspective with the world.
          </Text>
        </View>

        <View style={styles.teamSection}>
          <Text style={styles.sectionTitle}>Meet the Team</Text>
          <View style={styles.teamMembersContainer}>
            <View style={styles.teamMember}>
              <Image
                source={{ uri:
'https://images.pexels.com/photos/220453/pexels-photo-220453.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2' }}

```

```
    style={styles.teamMemberImage}
  />
  <Text style={styles.teamMemberName}>Jason Kim</Text>
  <Text style={styles.teamMemberRole}>Founder & Editor</Text>
</View>
```

```
<View style={styles.teamMember}>
  <Image
    source={{ uri:
'https://images.pexels.com/photos/774909/pexels-photo-774909.jpeg?auto=compress&cs=tinysrgb&w=1260
&h=750&dpr=2' }}
    style={styles.teamMemberImage}
  />
  <Text style={styles.teamMemberName}>Sarah Chen</Text>
  <Text style={styles.teamMemberRole}>Lead Photographer</Text>
</View>
```

```

  <View style={styles.teamMember}>
    <Image
      source={{ uri:
'https://images.pexels.com/photos/415829/pexels-photo-415829.jpeg?auto=compress&cs=tinysrgb&w=1260
&h=750&dpr=2' }}
      style={styles.teamMemberImage}
    />
    <Text style={styles.teamMemberName}>Emma Rodriguez</Text>
    <Text style={styles.teamMemberRole}>Content Director</Text>
  </View>
</View>
</View>
```

```
<View style={styles.contactSection}>
  <Text style={styles.sectionTitle}>Contact Us</Text>
  <Text style={styles.contactText}>
    Have questions, feedback, or want to collaborate? We'd love to hear from you!
  </Text>
```

```
<TouchableOpacity style={styles.contactButton}>
  <Mail size={20} color="#FFFFFF" />
  <Text style={styles.contactButtonText}>Email Us</Text>
</TouchableOpacity>
```

```
<View style={styles.socialLinks}>
  <TouchableOpacity style={styles.socialButton}>
    <Instagram size={24} color="#121212" />
  </TouchableOpacity>
  <TouchableOpacity style={styles.socialButton}>
    <Twitter size={24} color="#121212" />
  </TouchableOpacity>
</View>
```

```

        </TouchableOpacity>
        <TouchableOpacity style={styles.socialButton}>
          <Facebook size={24} color="#121212" />
        </TouchableOpacity>
      </View>
    </View>
  </ScrollView>
</View>
);
}

```

```

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#FFFFFF',
  },
  header: {
    backgroundColor: '#121212',
    padding: 16,
    paddingBottom: 24,
    alignItems: 'center',
  },
  headerTitle: {
    color: '#FFFFFF',
    fontSize: 24,
    fontFamily: 'Roboto-Bold',
    letterSpacing: 0.5,
  },
  headerSubtitle: {
    color: '#E0E0E0',
    fontSize: 14,
    fontFamily: 'Roboto-Regular',
    marginTop: 2,
  },
  content: {
    flex: 1,
    padding: 16,
  },
  storySection: {
    marginBottom: 30,
  },
  storyImage: {
    width: '100%',
    height: 200,
    borderRadius: 12,
    marginBottom: 16,
  },

```

```
sectionTitle: {
  fontSize: 20,
  fontFamily: 'Roboto-Bold',
  color: '#121212',
  marginBottom: 12,
},
storyText: {
  fontSize: 15,
  fontFamily: 'Roboto-Regular',
  color: '#333333',
  lineHeight: 22,
  marginBottom: 12,
},
missionSection: {
  backgroundColor: '#F9F9F9',
  padding: 16,
  borderRadius: 12,
  marginBottom: 30,
},
missionText: {
  fontSize: 15,
  fontFamily: 'Roboto-Regular',
  color: '#333333',
  lineHeight: 22,
},
teamSection: {
  marginBottom: 30,
},
teamMembersContainer: {
  flexDirection: 'row',
  justifyContent: 'space-between',
  flexWrap: 'wrap',
},
teamMember: {
  width: '30%',
  alignItems: 'center',
  marginBottom: 16,
},
teamMemberImage: {
  width: 80,
  height: 80,
  borderRadius: 40,
  marginBottom: 8,
},
teamMemberName: {
  fontSize: 14,
  fontFamily: 'Roboto-Medium',
```

```
    color: '#121212',
    textAlign: 'center',
  },
  teamMemberRole: {
    fontSize: 12,
    fontFamily: 'Roboto-Regular',
    color: '#666666',
    textAlign: 'center',
  },
  contactSection: {
    marginBottom: 30,
  },
  contactText: {
    fontSize: 15,
    fontFamily: 'Roboto-Regular',
    color: '#333333',
    lineHeight: 22,
    marginBottom: 16,
  },
  contactButton: {
    backgroundColor: '#121212',
    flexDirection: 'row',
    alignItems: 'center',
    justifyContent: 'center',
    paddingVertical: 12,
    paddingHorizontal: 20,
    borderRadius: 8,
    marginBottom: 20,
  },
  contactButtonText: {
    color: '#FFFFFF',
    fontSize: 16,
    fontFamily: 'Roboto-Medium',
    marginLeft: 8,
  },
  socialLinks: {
    flexDirection: 'row',
    justifyContent: 'center',
  },
  socialButton: {
    width: 50,
    height: 50,
    borderRadius: 25,
    backgroundColor: '#F0F0F0',
    alignItems: 'center',
    justifyContent: 'center',
    marginHorizontal: 8,
```

```
},  
});
```

Output:

E. About us page output:

11:47



About Us

Our story and mission



Our Story

OneShot was founded in 2022 by a group of passionate photographers who wanted to create a platform where photography enthusiasts of all levels could learn, share, and grow together.

What started as a small blog has grown into a vibrant community of photographers from around the world, all united by their love for capturing moments through the lens.

Our Mission

At OneShot, we believe that photography is more than just taking pictures—it's about storytelling, creativity, and connecting with the world around us. Our mission is to inspire and empower photographers to develop their skills, find their unique voice, and share their perspective with the world.

Code:

F. login page:

code:

```
import { useState } from 'react';
import {
  View,
  Text,
  StyleSheet,
  TextInput,
  TouchableOpacity,
  Image,
  KeyboardAvoidingView,
  Platform,
  ScrollView
} from 'react-native';
import { useSafeAreaInsets } from 'react-native-safe-area-context';
import React from 'react';
export default function LoginScreen() {
  const insets = useSafeAreaInsets();
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const [isLogin, setIsLogin] = useState(true);

  return (
    <KeyboardAvoidingView
      style={styles.container}
      behavior={Platform.OS === 'ios' ? 'padding' : 'height'}
    >
      <View style={[styles.header, { paddingTop: insets.top }]}>
        <Text style={styles.headerTitle}>{isLogin ? 'Login' : 'Sign Up'}</Text>
        <Text style={styles.headerSubtitle}>
          {isLogin ? 'Access your account' : 'Join our community'}
        </Text>
      </View>

      <ScrollView
        style={styles.content}
        contentContainerStyle={styles.contentContainer}
      >
        <Image
          source={{ uri:
'https://images.pexels.com/photos/1983037/pexels-photo-1983037.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2' }}
          style={styles.logo}
        />
```

```

<Text style={styles.welcomeText}>
  {isLoggedIn
    ? 'Welcome back to OneShot!'
    : 'Join the OneShot community!'}
</Text>

<View style={styles.form}>
  <View style={styles.inputContainer}>
    <Text style={styles.inputLabel}>Email</Text>
    <TextInput
      style={styles.input}
      placeholder="your@email.com"
      value={email}
      onChangeText={setEmail}
      keyboardType="email-address"
      autoCapitalize="none"
    />
  </View>

  <View style={styles.inputContainer}>
    <Text style={styles.inputLabel}>Password</Text>
    <TextInput
      style={styles.input}
      placeholder="Your password"
      value={password}
      onChangeText={setPassword}
      secureTextEntry
    />
  </View>

  {isLoggedIn && (
    <TouchableOpacity style={styles.forgotPassword}>
      <Text style={styles.forgotPasswordText}>Forgot Password?</Text>
    </TouchableOpacity>
  )}

  <TouchableOpacity style={styles.button}>
    <Text style={styles.buttonText}>{isLoggedIn ? 'Login' : 'Sign Up'}</Text>
  </TouchableOpacity>

  <View style={styles.switchContainer}>
    <Text style={styles.switchText}>
      {isLoggedIn
        ? "Don't have an account? "
        : "Already have an account? "}
    </Text>
    <TouchableOpacity onPress={() => setIsLoggedIn(!isLoggedIn)}>

```

```

        <Text style={styles.switchButton}>
          {isLogin ? 'Sign Up' : 'Login'}
        </Text>
      </TouchableOpacity>
    </View>
  </View>

  <View style={styles.divider}>
    <View style={styles.dividerLine} />
    <Text style={styles.dividerText}>OR</Text>
    <View style={styles.dividerLine} />
  </View>

  <View style={styles.socialButtons}>
    <TouchableOpacity style={styles.socialButton}>
      <Text style={styles.socialButtonText}>Continue with Google</Text>
    </TouchableOpacity>

    <TouchableOpacity style={styles.socialButton}>
      <Text style={styles.socialButtonText}>Continue with Facebook</Text>
    </TouchableOpacity>
  </View>
</ScrollView>
</KeyboardAvoidingView>
);
}

```

```

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#FFFFFF',
  },
  header: {
    backgroundColor: '#121212',
    padding: 16,
    paddingBottom: 24,
    alignItems: 'center',
  },
  headerTitle: {
    color: '#FFFFFF',
    fontSize: 24,
    fontFamily: 'Roboto-Bold',
    letterSpacing: 0.5,
  },
  headerSubtitle: {
    color: '#E0E0E0',
    fontSize: 14,
  },
});

```

```
    fontFamily: 'Roboto-Regular',
    marginTop: 2,
  },
  content: {
    flex: 1,
  },
  contentContainer: {
    padding: 24,
    alignItems: 'center',
  },
  logo: {
    width: 100,
    height: 100,
    borderRadius: 50,
    marginBottom: 20,
  },
  welcomeText: {
    fontSize: 18,
    fontFamily: 'Roboto-Bold',
    color: '#121212',
    marginBottom: 24,
    textAlign: 'center',
  },
  form: {
    width: '100%',
    marginBottom: 24,
  },
  inputContainer: {
    marginBottom: 16,
  },
  inputLabel: {
    fontSize: 14,
    fontFamily: 'Roboto-Medium',
    color: '#333333',
    marginBottom: 8,
  },
  input: {
    backgroundColor: '#F5F5F5',
    borderRadius: 8,
    padding: 12,
    fontSize: 16,
    fontFamily: 'Roboto-Regular',
    color: '#333333',
  },
  forgotPassword: {
    alignSelf: 'flex-end',
    marginBottom: 20,
```

```
},
forgotPasswordText: {
  fontSize: 14,
  fontFamily: 'Roboto-Regular',
  color: '#666666',
},
button: {
  backgroundColor: '#FFCB45',
  borderRadius: 8,
  paddingVertical: 14,
  alignItems: 'center',
  marginBottom: 16,
},
buttonText: {
  fontSize: 16,
  fontFamily: 'Roboto-Bold',
  color: '#121212',
},
switchContainer: {
  flexDirection: 'row',
  justifyContent: 'center',
},
switchText: {
  fontSize: 14,
  fontFamily: 'Roboto-Regular',
  color: '#666666',
},
switchButton: {
  fontSize: 14,
  fontFamily: 'Roboto-Bold',
  color: '#121212',
},
divider: {
  flexDirection: 'row',
  alignItems: 'center',
  width: '100%',
  marginVertical: 24,
},
dividerLine: {
  flex: 1,
  height: 1,
  backgroundColor: '#EEEEEE',
},
dividerText: {
  paddingHorizontal: 16,
  fontSize: 14,
  fontFamily: 'Roboto-Regular',
```

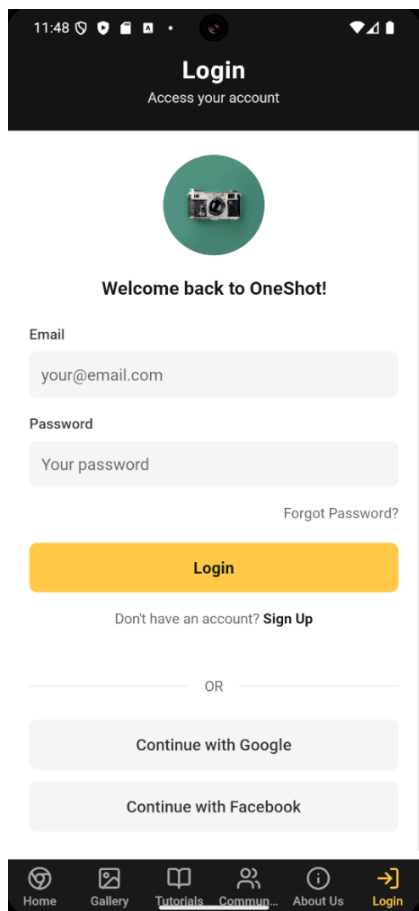
```

        color: '#666666',
    },
    socialButtons: {
        width: '100%',
    },
    socialButton: {
        backgroundColor: '#F5F5F5',
        borderRadius: 8,
        paddingVertical: 14,
        alignItems: 'center',
        marginBottom: 12,
    },
    socialButtonText: {
        fontSize: 16,
        fontFamily: 'Roboto-Medium',
        color: '#333333',
    },
    },
});

```

Output:

F. login page output:



Conclusion:

The OneShot mobile application developed using React Native offered insight into cross-platform mobile development. It enabled us to build an interactive, visually cohesive app that replicates the functionality of the website while optimizing the interface for mobile use. This assignment helped us understand the component-based architecture of React Native, learn about state management, navigation handling, and responsive layout design in the mobile context.

Each screen in the app—Home, Gallery, Tutorials, Community, About Us, Login, and Registration—was built with a focus on intuitive interaction and accessibility. The React Native framework allowed for rapid development with reusable components and modern styling approaches. This assignment has been instrumental in strengthening our skills in mobile development, design logic, and user experience for a real-world use case.