Firebase Authentication for login and signup

1. Set Up Firebase

1. **Create a Firebase project**:
   - Go to Firebase Console.
   - Create a new project
2. **Enable Authentication**:
   - In your Firebase project, navigate to **Authentication** > **Sign-in method**.
   - Enable **Email/Password** and **Google** as sign-in providers.
3. **Register your app**
   - Register your app using any name
   - Go to Settings > General >scroll to bottom and copy the firebaseConfig file

4. **Install Firebase SDK**:
   - In your frontend directory, install Firebase
   - “npm install firebase”

5. **Go to VScode**
   - Create a firebaseConfig.js file and paste the config and save it.
   - Now, Modify the Login and signup page.
   - And run your project.

Signup.js

```
import React, { useState } from 'react';
import { useNavigate } from 'react-router-dom';
import { auth, createUserWithEmailAndPassword } from '../firebaseConfig';
import '../auth.css';
import logo from '../images/ShariaStocks-logo/logo.png';
import {
  Container, Paper, TextField, Button, Typography, Box, Link, Alert, CircularProgress
} from '@mui/material';

const Signup = () => {
  const [formData, setFormData] = useState({
    name: '',
    email: '',
    password: '',
    confirmPassword: '',
  });
  const { name, email, password, confirmPassword } = formData;
  const [alert, setAlert] = useState(null);
  const [error, setError] = useState('');
  const [loading, setLoading] = useState(false);
  const navigate = useNavigate();

  const onChange = (e) => setFormData({ ...formData, [e.target.name]: e.target.value });

  const onSubmit = async (e) => {
    e.preventDefault();
    setError('');
    if (password !== confirmPassword) {
      setError('Passwords do not match');
      return;
    }
    if (password.length < 6) {
      setError('Password must be at least 6 characters');
      return;
    }
    setLoading(true);
    try {
      const userCredential = await createUserWithEmailAndPassword(auth,email, password);
      console.log('Signup successful!', userCredential.user);
      setAlert({ type: 'success', message: 'Signup successful!' });
      setTimeout(() => navigate('/login'), 1500);
    } catch (err) {
      console.error('Signup error:', err.message);
      setError(err.message || 'Failed to create account');
    } finally {
      setLoading(false);
    }
  };
  return (
    <div>
      <div className='header-section'>
        <img src={logo} alt="logo" />
      </div>
      <Container component="main" maxWidth="xs">
        <Box sx={{ marginTop: 8, display: 'flex', flexDirection: 'column', alignItems: 'center' }}>
          <Paper elevation={3} sx={{ p: 4, width: '100%' }}>
            <Typography component="h1" variant="h5" sx={{ textAlign: 'center', mb: 3 }}>
              Sign Up
            </Typography>
            {error && <Alert severity="error" sx={{ mb: 2 }}>{error}</Alert>}
            {alert && <Alert severity={alert.type} sx={{ mb: 2 }}>{alert.message}</Alert>}
            <Box component="form" onSubmit={onSubmit} sx={{ mt: 1 }}>
              <TextField margin="normal" required fullWidth id="name" label="Name" name="name"
                autoFocus value={name} onChange={onChange} disabled={loading} autoComplete="name" />
              <TextField margin="normal" required fullWidth id="email" label="Email Address" name="email"
                autoComplete="email" value={email} onChange={onChange} disabled={loading} type="email" />
              <TextField margin="normal" required fullWidth name="password" label="Password"
                type="password" id="password" value={password} onChange={onChange} disabled={loading} />
              <TextField margin="normal" required fullWidth name="confirmPassword" label="Confirm Password"
                type="password" id="confirmPassword2" value={confirmPassword} onChange={onChange} disabled={loading} />
              <Button type="submit" fullWidth variant="contained" sx={{ mt: 3, mb: 2, bgcolor: '#00A86B' }}
                disabled={loading}>
                {loading ? <CircularProgress size={24} color="inherit" /> : 'Sign Up'}
              </Button>
              <Box sx={{ textAlign: 'center', mt: 2 }}>
                <Link component="button" variant="body2" onClick={() => navigate('/login')}
                  sx={{ color: '#00A86B' }} disabled={loading}>
                  Already have an account? Login
                </Link>
              </Box>
            </Box>
          </Paper>
        </Box>
      </Container>
    </div>
  );
};

export default Signup;
```

Login.js

```
import React, { useState, useEffect } from 'react';
import { useNavigate } from 'react-router-dom';
import { auth, googleProvider, signInWithEmailAndPassword } from '../firebaseConfig.js';
import logo from '../images/ShariaStocks-logo/logo.png';
import '../auth.css';
import {
  Container, Paper, TextField, Button, Typography, Box, Link, Alert, CircularProgress
} from '@mui/material';

const Login = () => {
  const [formData, setFormData] = useState({ email: '', password: '' });
  const { email, password } = formData;
  const [error, setError] = useState('');
  const [loading, setLoading] = useState(false);
  const navigate = useNavigate();

  useEffect(() => {
    if (localStorage.getItem('userEmail')) {
      navigate('/Dashboard');
    }
  }, [navigate]);

  const onChange = (e) => setFormData({ ...formData, [e.target.name]: e.target.value });

  const onSubmit = async (e) => {
    e.preventDefault();
    setLoading(true);
    try {
      const userCredential = await signInWithEmailAndPassword(auth,email, password);
      console.log('Login successful!', userCredential.user);
      localStorage.setItem('userEmail', userCredential.user.email);
      navigate('/Dashboard');
      setFormData({ email: '', password: '' });
    } catch (err) {
      console.error('Login error:', err.message);
      setError(err.message || 'Error logging in');
    } finally {
      setLoading(false);
    }
  };

  const handleGoogleSignIn = async () => {
    try {
      const result = await auth.signInWithPopup(googleProvider);
      const user = result.user;
      console.log('Google Sign-In successful!', user);
```

```
   localStorage.setItem('userEmail', user.email);
   navigate('/Dashboard');
  } catch (err) {
   console.error('Google Sign-In error:', err.message);
   setError(err.message || 'Failed to sign in with Google');
  }
 };

 return (
  <div>
   <div className='header-section'>
    <img src={logo} alt="logo" />
   </div>
   <Container component="main" maxWidth="xs">
    <Box sx={{ marginTop: 8, display: 'flex', flexDirection: 'column', alignItems: 'center' }}>
     <Paper elevation={3} sx={{ p: 4, width: '100%' }}>
      <Typography component="h1" variant="h5" sx={{ textAlign: 'center', mb: 3 }}>
       Login
      </Typography>
      {error && <Alert severity="error" sx={{ mb: 2 }}>{error}</Alert>}
      <Box component="form" onSubmit={onSubmit} sx={{ mt: 1 }}>
       <TextField margin="normal" required fullWidth id="email" label="Email Address"
name="email"
        autoComplete="email" autoFocus value={email} onChange={onChange} disabled={loading} />
       <TextField margin="normal" required fullWidth name="password" label="Password"
type="password"
        id="password" autoComplete="current-password" value={password} onChange={onChange}
disabled={loading} />
       <Button type="submit" fullWidth variant="contained" sx={{ mt: 3, mb: 2, bgcolor: '#00A86B' }}
        disabled={loading}>
        {loading ? 'Logging in...' : 'Login'}
       </Button>
       <Button fullWidth variant="contained" color="secondary" onClick={handleGoogleSignIn}
disabled={loading}>
        Sign in with Google
       </Button>
       <Box sx={{ textAlign: 'center', mt: 2 }}>
        <Link component="button" variant="body2" onClick={() => navigate('/forgot-password')}
         sx={{
          color: '#00A86B',
          '&:hover': {
           backgroundColor: 'transparent',
           color: '#002d62',
          },
         }}disabled={loading}>
         Forgot Password?
        </Link>
       </Box>
```

```
              <Box sx={{ textAlign: 'center', mt: 2 }}>
                <Link component="button" variant="body2" onClick={() => navigate('/signup')}
                  sx={{
                    color: '#00A86B',
                    '&:hover': {
                      backgroundColor: 'transparent',
                      color: '#002d62',
                    },
                  }}disabled={loading}>
                  Don't have an account? Sign up
                </Link>
              </Box>
            </Box>
          </Paper>
        </Box>
      </Container>
    </div>
  );
};

export default Login;
```