Password Change

Let's learn how to implement the "Change Password" feature in our application.

WE'LL COVER THE FOLLOWING ∧

- Updating the Password
  - Implementation
  - Explanation
  - The Account Page

Updating the Password

Now, we'll add the second password feature that we need in our application: **password change**. This will also be in our Firebase interface. We only need a **form** component to use it.

Again, the form component isn't any different from the *sign in, sign up,* and *forget password* forms.

Implementation

In the `src/components/` `PasswordChange/index.js` file add the following component:

```
import React, { Component } from 'react';

import { withFirebase } from '../Firebase';

const INITIAL_STATE = {
  passwordOne: '',
  passwordTwo: '',
  error: null,
};

class PasswordChangeForm extends Component {
  constructor(props) {
    super(props);
```

```
      this.state = { ...INITIAL_STATE };
    }


  onSubmit = event => {
    const { passwordOne } = this.state;

    this.props.firebase
      .doPasswordUpdate(passwordOne)
      .then(() => {
        this.setState({ ...INITIAL_STATE });
      })
      .catch(error => {
        this.setState({ error });
      });

    event.preventDefault();
  };

  onChange = event => {
    this.setState({ [event.target.name]: event.target.value });
  };

  render() {
    const { passwordOne, passwordTwo, error } = this.state;

    const isInvalid =
      passwordOne !== passwordTwo || passwordOne === '';

    return (
      <form onSubmit={this.onSubmit}>
        <input
          name="passwordOne"
          value={passwordOne}
          onChange={this.onChange}
          type="password"
          placeholder="New Password"
        />
        <input
          name="passwordTwo"
          value={passwordTwo}
          onChange={this.onChange}
          type="password"
          placeholder="Confirm New Password"
        />
        <button disabled={isInvalid} type="submit">
          Reset My Password
        </button>

        {error && <p>{error.message}</p>}
      </form>
    );
  }
}

export default withFirebase(PasswordChangeForm);
```

PasswordChange/index.js

Explanation

The component updates its local state using the `onChange` handlers in the input fields. It validates the state before submitting a request to change the password by enabling or disabling the submit button. In case a request fails, it shows an error message.

So far, the `PasswordChangeForm` is not matched by any route because it should live on the **Account** page. The Account page could serve as the central place for users to manage their accounts, where it shows the `PasswordChangeForm` and `PasswordResetForm`. This ensures that the two forms are accessible by a standalone route.

The Account Page

We already created the `src/components/` `Account/index.js` file and matched the route in the App component. You only need to implement it:

```
import React from 'react';

import { PasswordForgetForm } from '../PasswordForget';
import PasswordChangeForm from '../PasswordChange';

const AccountPage = () => (
  <div>
    <h1>Account Page</h1>
    <PasswordForgetForm />
    <PasswordChangeForm />
  </div>
);

export default AccountPage;
```

Account/index.js

The Account page doesn't have any business logic. It contains the **forgot password** and **change password** features in a central place.

Now, our user experience has improved significantly with the introduction of the password features.

Next, we will run and verify our app to test if the Password change and reset functionality is working.