

```

import React from 'Crowdbotics';
import React from 'react';
import { Link } from 'react-router';
import jQuery from 'jquery';
import $ from 'jquery';
import { connect } from 'react-redux';
import * as signupActions from '../actions/signupActions';
import * as appLayout from '../scripts/layout';
import { bindActionCreators } from 'redux';
import ReactDOM from 'react-dom';
import '../styles/login5.min.css';
//signup container component
class Signup extends React.Component {
  constructor(props, context) {
    super(props, context);
    this.state = {
      signup: {
        email: '',
        password: '',
        firstname: '',
        lastname: '',
        company: '',
        confirmpassword: ''
      }
    };

    //binding current context to all functions
    //used in the container component
    this.onEmailChange = this.onEmailChange.bind(this);
    this.onPasswordChange =
this.onPasswordChange.bind(this);
    this.onFirstChange = this.onFirstChange.bind(this);
    this.onLastChange = this.onLastChange.bind(this);
    this.onCompanyChange = this.onCompanyChange.bind(this);
    this.onConfirmChange = this.onConfirmChange.bind(this);
    this.onSignupHandler = this.onSignupHandler.bind(this);
  }

  componentDidMount() {
    appLayout.App.init();
  }

  //onChange method for email
  onEmailChange(event) {
    const signup = this.state.signup;
    signup.email = event.target.value;
    this.setState({ signup: signup });
  }

```

```

}

//onChange method for email
onCompanyChange(event) {
  const signup = this.state.signup;
  signup.company = event.target.value;
  this.setState({ signup: signup });
}

//onChange method for email
onConfirmChange(event) {
  const signup = this.state.signup;
  signup.confirmpassword = event.target.value;
  this.setState({ signup: signup });
}

//onChange method for password
onPasswordChange(event) {
  const signup = this.state.signup;
  signup.password = event.target.value;
  this.setState({ signup: signup });
}

//onChange method for first name
onFirstChange(event) {
  const signup = this.state.signup;
  signup.firstname = event.target.value;
  this.setState({ signup: signup });
}

//onChange method for last name
onLastChange(event) {
  const signup = this.state.signup;
  signup.lastname = event.target.value;
  this.setState({ signup: signup });
}

//onClick handler for
onSignupHandler(event) {
  event.preventDefault();
  this.props.actions.registerUser(this.state.signup);
}

render() {
  return (
    <div className="row">
      <div className="user-login-5">
        <div className="row bs-reset">

```

```

        <div className="col-md-6 bs-reset mt-
login-5-bsfix">
            <div className="login-bg back-img">
                <Link to="/"><img
className="login-logo" src={require('../img/login/logo.png')}
/></Link>
            </div>
        </div>
        <div className="col-md-6 login-container
bs-reset mt-login-5-bsfix">
            <div className="login-content
signupFormId">
                <h1>Join Now</h1>
                <p> Lorem ipsum dolor sit amet,
coectetuer adipiscing elit sed diam nonummy et nibh euismod
aliquam erat volutpat. Lorem ipsum dolor sit amet, coectetuer
adipiscing. </p>
                <form className="login-form"
id="signupForm" onSubmit={this.onSignupHandler}>
                    <div className="row">
                        <div className="col-xs-
6">
                            <div
className="form-group form-md-line-input form-md-floating-label
has-success">
                                <input
type="text" className="form-control" id="firstname"
ref="firstname" onChange={this.onFirstChange}
value={this.state.signup.firstname} name="firstname" />
                                <label
htmlFor="firstname">First name<span
className="required">*</span></label>
                            </div>
                        </div>
                        <div className="col-xs-
6">
                            <div
className="form-group form-md-line-input form-md-floating-label
has-success">
                                <input
type="text" className="form-control" id="lastname"
ref="lastname" onChange={this.onLastChange}
value={this.state.signup.lastname} name="lastname" />

```

```

                                <label
htmlFor="lastname">Last name<span
className="required">*</span></label>
                                </div>
                                </div>
                                </div>
                                <div className="row">
                                <div className="col-xs-
6">
                                <div
className="form-group form-md-line-input form-md-floating-label
has-success">
                                <input
type="text" className="form-control" id="email"
                                ref="email"
onChange={this.onEmailChange} value={this.state.signup.email}
name="email" />
                                <label
htmlFor="email">Email<span className="required">*</span></label>
                                </div>
                                </div>
                                <div className="col-xs-
6">
                                <div
className="form-group form-md-line-input form-md-floating-label
has-success">
                                <input
type="text" className="form-control" id="company"
                                ref="company" onChange={this.onCompanyChange}
value={this.state.signup.company} name="company" />
                                <label
htmlFor="company">Company<span
className="required">*</span></label>
                                </div>
                                </div>
                                </div>
                                <div className="row">
                                <div className="col-xs-
6">
                                <div
className="form-group form-md-line-input form-md-floating-label
has-success">
                                <input
type="password" className="form-control" id="password"

```

```

ref="password" onChange={this.onPasswordChange}
value={this.state.signup.password} name="password" />
</label>
htmlFor="password">Password<span
className="required">*</span></label>
</div>
</div>
<div className="col-xs-
6">
<div
className="form-group form-md-line-input form-md-floating-label
has-success">
<input
type="password" className="form-control" id="confirmpassword"
ref="confirmpassword" onChange={this.onConfirmChange}
value={this.state.signup.confirmpassword} name="confirmpassword"
/>
</label>
htmlFor="confirmpassword">Confirm password<span
className="required">*</span></label>
</div>
</div>
</div>
<div className="row">
<div className="col-sm-
6">
<div className="md-
checkbox-inline">
<div
className="md-checkbox">
<input
type="checkbox" id="checkbox33" className="md-check"
name="checkbox33" />
</label>
htmlFor="checkbox33">
<span></span>
<span
className="check"></span>
<span
className="box"></span> I agree to all <Link
to="/terms">Terms</Link>
</label>
</div>
</div>

```

```

        </div>
        <div className="col-sm-6
text-right">
            <button
className="btn green" id="signup" type="submit">Join</button>
            </div>
        </div>
    </form>

</div>
<div className="login-footer
signupFormFooter">
    <div className="row bs-reset">
        <div className="col-xs-5 bs-
reset">
            <ul className="login-
social">
                <li>
                    <a
                        <i
className="icon-social-facebook"></i>
                    </a>
                </li>
                <li>
                    <a
                        <i
className="icon-social-twitter"></i>
                    </a>
                </li>
                <li>
                    <a
                        <i
className="icon-social-dribbble"></i>
                    </a>
                </li>
            </ul>
        </div>
        <div className="col-xs-7 bs-
reset">
            <div className="login-
copyright text-right">
                <p>2017 &copy;
react</p>

```

```

        </div>
      </div>
    </div>
  </div>
</div>
</div>
);
}
}

//this tells what state should expose on props
function mapStateToProps(state, ownProps) {
  return {
    signup: state.signup
  };
}

//this tells what action should expose on props
//bindActionCreators is used to bind all actions at once
function mapDispatchToProps(dispatch) {
  return {
    actions: bindActionCreators(signupActions, dispatch)
  };
}

//this connect function wraps up all our components with store
export default connect(mapStateToProps,
mapDispatchToProps)(Signup);

```