**Source Code Sample:**

**Login.js**

const Validator = require("validator");

const isEmpty = require("./is-empty");

module.exports = function validateLoginInput(data) {

let errors = {};

data.email = !isEmpty(data.email) ? data.email : "";

data.password = !isEmpty(data.password) ? data.password : "";

if (Validator.isEmpty(data.email)) {

errors.email = "Email field is required";

} else {

if (!Validator.isEmail(data.email)) {

errors.email = "Email is invalid";

}

}

if (Validator.isEmpty(data.password)) {

errors.password = "Password field is required";

}

return {

errors,

isValid: isEmpty(errors)

};

};

**Package.json**

{

"name": "jserver",

"version": "1.0.0",

"description": "",

"main": "server.js",

"scripts": {

"start": "nodemon server.js",

"test": "echo \"Error: no test specified\" && exit 1"

},

"keywords": [],

"author": "",

"license": "ISC",

"devDependencies": {

"nodemon": "^2.0.6"

},

"dependencies": {

"bcrypt": "^5.0.0",

"body-parser": "^1.19.0",

"cors": "^2.8.5",

"express": "^4.17.1",

"jsonwebtoken": "^8.5.1",

"mysql": "^2.18.1",

"validator": "^13.1.17"

}

}

**Register.js**

const Validator = require("validator");

const isEmpty = require("./is-empty");

module.exports = function validateRegisterInput(data) {

let errors = {};

data.firstName = !isEmpty(data.firstName) ? data.firstName : "";

data.lastName = !isEmpty(data.lastName) ? data.lastName : "";

data.email = !isEmpty(data.email) ? data.email : "";

data.password = !isEmpty(data.password) ? data.password : "";

data.phone = !isEmpty(data.phone) ? data.phone : "";

data.dob = !isEmpty(data.dob) ? data.dob : "";

data.street = !isEmpty(data.street) ? data.street : "";

data.city = !isEmpty(data.city) ? data.city : "";

data.state = !isEmpty(data.state) ? data.state : "";

data.zip = !isEmpty(data.zip) ? data.zip : "";

data.country = !isEmpty(data.country) ? data.country : "";

data.substatus = !isEmpty(data.substatus) ? data.substatus : "";

if (Validator.isEmpty(data.firstName)) {

errors.firstName = "FirstName field is required";

} else {

if (!Validator.isLength(data.firstName, { min: 2, max: 30 })) {

errors.firstName = "First Name must be at least 2 characters";

}

}

if (Validator.isEmpty(data.lastName)) {

errors.lastName = "LastName field is required";

} else {

if (!Validator.isLength(data.lastName, { min: 2, max: 30 })) {

errors.lastName = "Last Name must be at least 2 characters";

}

}

if (Validator.isEmpty(data.email)) {

errors.email = "Email field is required";

} else {

if (!Validator.isEmail(data.email)) {

errors.email = "Email is invalid";

}

}

if (Validator.isEmpty(data.password)) {

errors.password = "Password field is required";

} else {

if (!Validator.isLength(data.password, { min: 6, max: 30 })) {

errors.password = "Password must be at least 6 characters";

}

}

if (Validator.isEmpty(data.phone)) {

errors.phone = "Phone field is required";

} else {

if (!Validator.isLength(data.phone, { min: 10, max: 10 })) {

errors.phone = "Phone must be at least 10 characters";

}

}

if (Validator.isEmpty(data.dob)) {

errors.dob = "Date of Birth field is required";

}

if (Validator.isEmpty(data.street)) {

errors.street = "Street field is required";

} else {

if (!Validator.isLength(data.street, { min: 2, max: 30 })) {

errors.street = "Street must be at least 2 characters";

}

}

if (Validator.isEmpty(data.city)) {

errors.city = "City field is required";

} else {

if (!Validator.isLength(data.city, { min: 2, max: 30 })) {

errors.city = "City must be at least 2 characters";

}

}

if (Validator.isEmpty(data.state)) {

errors.state = "State field is required";

} else {

if (!Validator.isLength(data.state, { min: 2, max: 30 })) {

errors.state = "State must be at least 2 characters";

}

}

if (Validator.isEmpty(data.zip)) {

errors.zip = "Zip field is required";

} else {

if (!Validator.isLength(data.zip, { min: 2, max: 30 })) {

errors.zip = "Zip must be at least 2 characters";

}

}

if (Validator.isEmpty(data.country)) {

errors.country = "Country field is required";

} else {

if (!Validator.isLength(data.country, { min: 2, max: 30 })) {

errors.country = "Country must be at least 2 characters";

}

}

if (Validator.isEmpty(data.substatus)) {

errors.substatus = "Subscription field is required";

}

return {

errors,

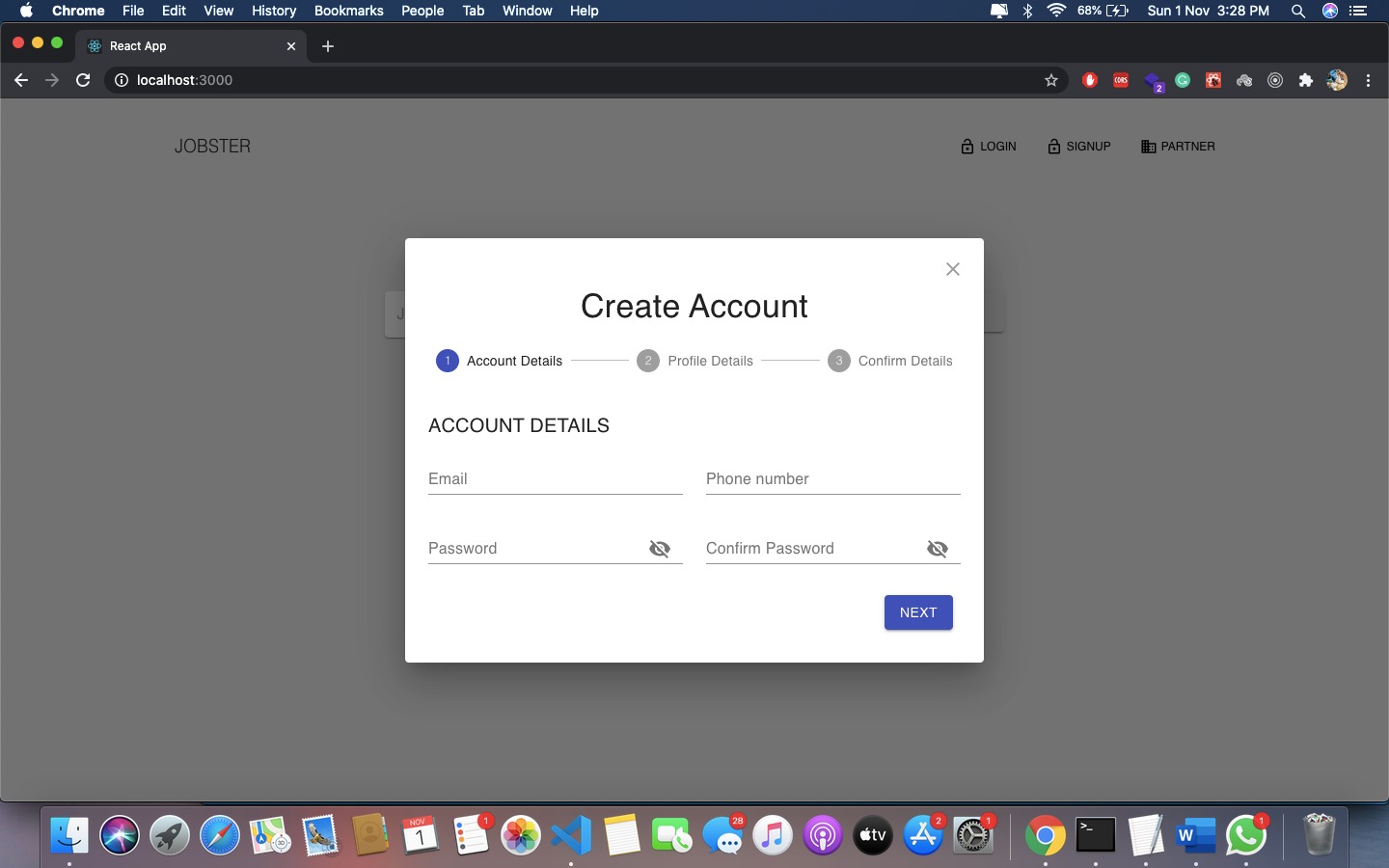
isValid: isEmpty(errors)

};

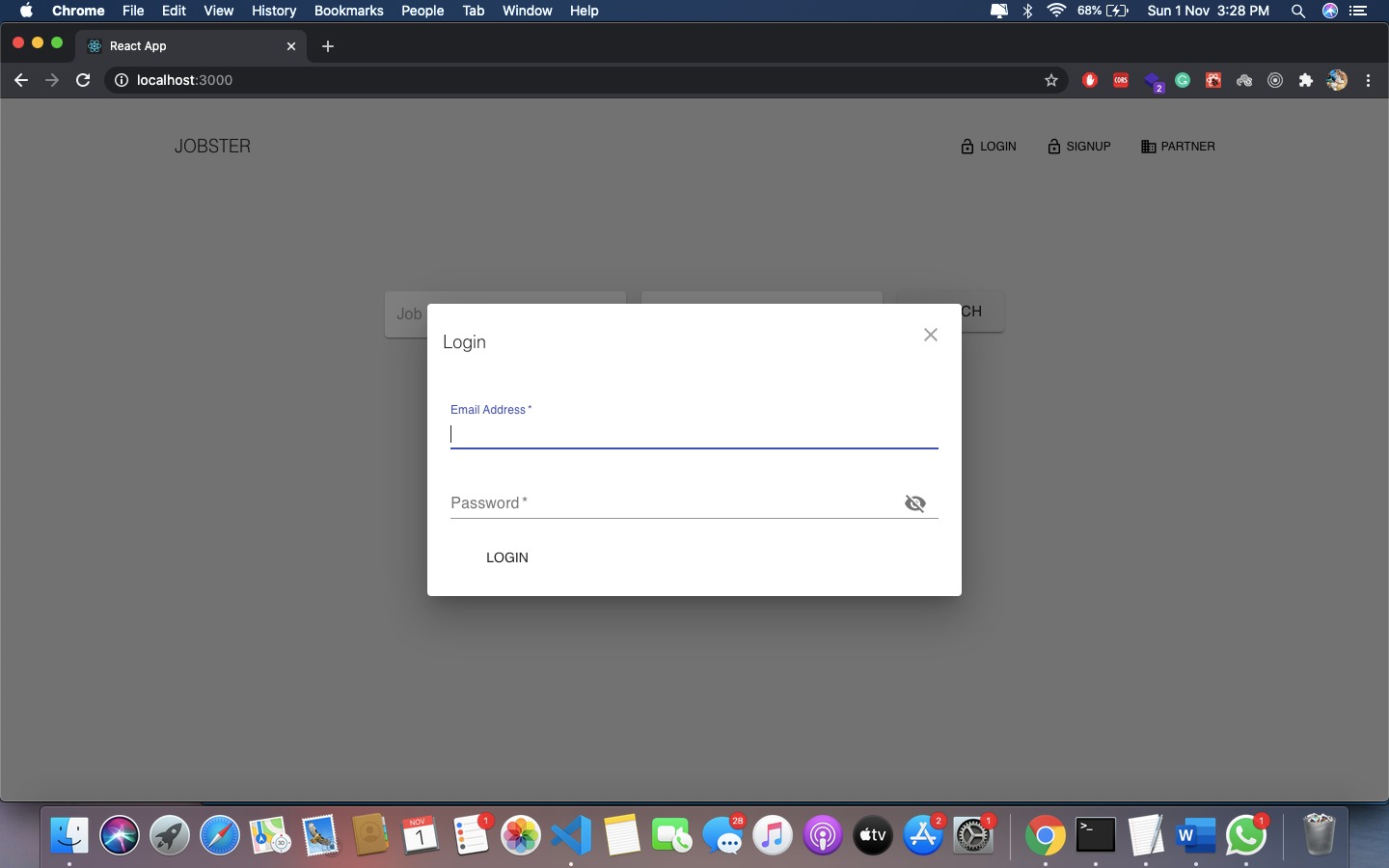
};

**Images:**

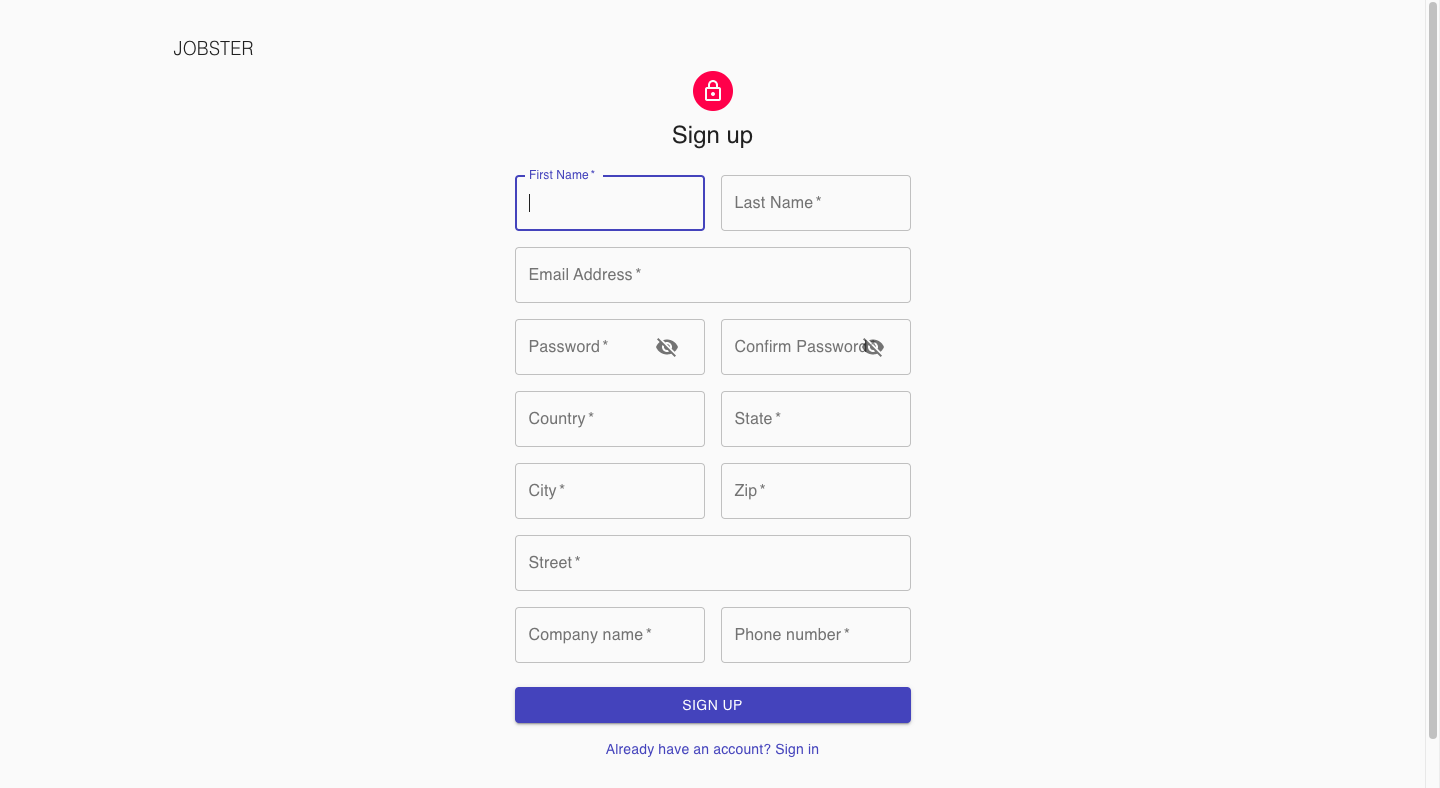
**User Sign Up**



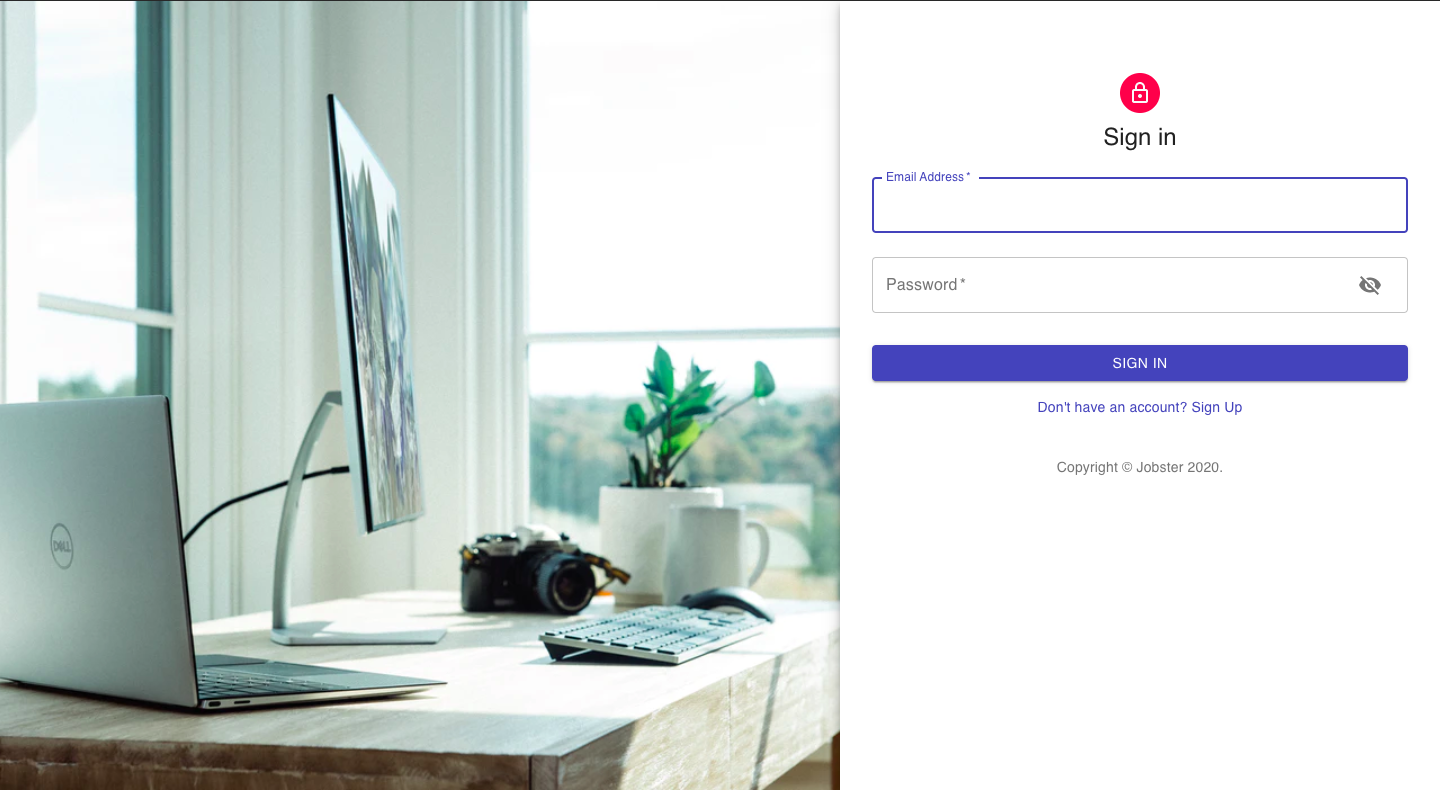
**User Login**



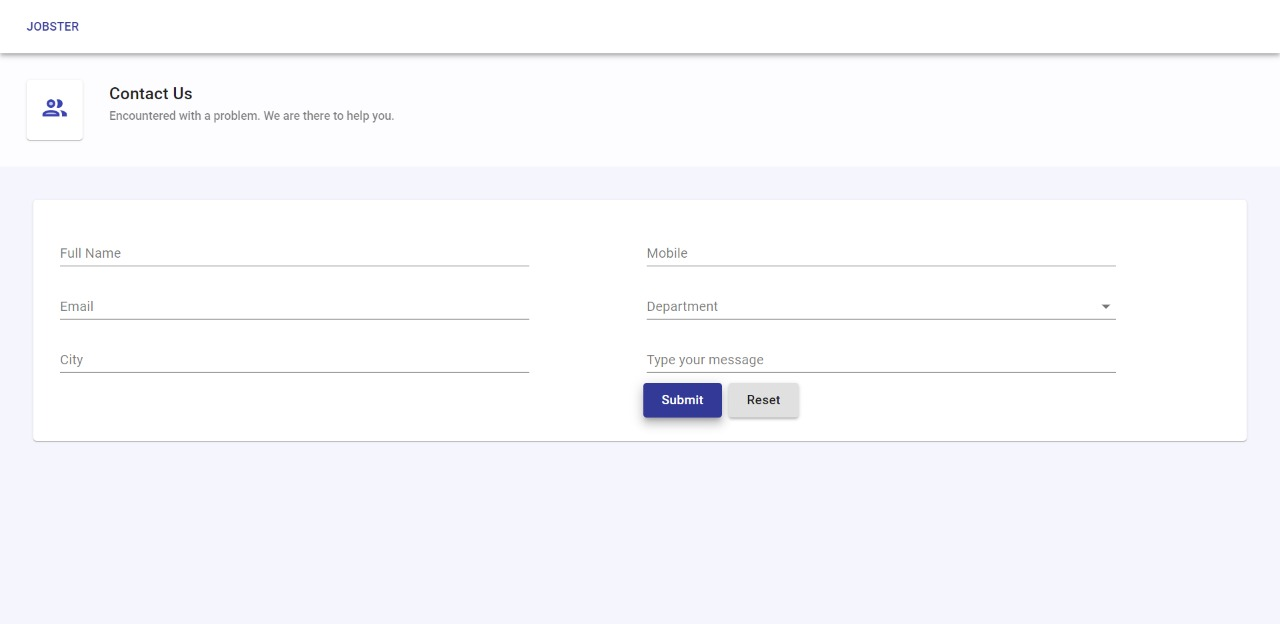
**Partner Sign Up**



**Partner Login**



**Contact Form**



**FAQ**

