Redux Actions: Objectives and Outcomes

In this lesson we will look at how to create redux actions and how action creators can be designed to return the action objects. Thereafter we look at how to dispatch actions. We also look at how to split the reducer into simpler functions and combine them. At the end of this exercise you will be able to:

- Define Redux actions
- Create action creator functions that return action objects
- Split the reducer function into multiple simpler functions and combine the reducer functions

Redux Actions: Additional Resources

PDFs of Presentations

1-Redux-Actions.pdfPDF File

Redux Resources

- Redux Actions
- Redux Reducers
- Redux Usage with React

Redux Thunk: Objectives and Outcomes

Redux Thunk middleware allows you to write action creators that return a function instead of an action. In this lesson you will see the use of redux thunk to return a function. At the end of this lesson you will be able to:

- Use Redux Thunk middleware to return a function instead of an action
- Use a logger middleware to print a log of actions initiated on the Redux store.

Redux Thunk: Additional Resources

PDFs of Presentations

2-Redux-Thunk.pdfPDF File

Redux Resources

- Redux Middleware
- Redux Thunk
- Redux Logger
- React-redux-form

Client-Server Communication: Objectives and Outcomes

In this lesson you will learn about communication between your React application and a server. You will establish a simple server using the json-server node module. At the end of this lesson you will be able to:

- Set up a simple server that makes data available for clients
- Access the data from the server using a browser.
- Use the json-server as a simple static web server.

Client-Server Communication: Additional Resources

PDFs of Presentations

3-Networking-Essentials.pdfPDF File
4-REST.pdfPDF File

Exercise Resources

db.json
images.zip

Other Resources

- json-server
- Creating Demo APIs with json-server
- JSON

Fetch: Objectives and Outcomes

In this lesson you will learn about Fetch as a means of communication between your React application and a REST API server. At the end of this lesson you will be able to:

- Install Fetch in your React application
- Use Fetch to communicate from your React application with a REST API server

Fetch: Additional Resources

PDFs of Presentations

5-Promises.pdfPDF File

6-Fetch.pdfPDF File

Fetch Resources

- Cross-Fetch

Promise Resources

- JavaScript Promise
- JS Promise (Part 1 - Basics)
- JavaScript Promises for Dummies
- JavaScript Promises: an Introduction

Other Resources

- Introduction to fetch()
- Using Fetch
- Fetch vs. Axios.js for making http requests

React Animations: Objectives and Outcomes

In this lesson we will learn about adding various subtle animations to our React app for a better user experience. At the end of this lesson you will be able to:

- Add subtle animations using the react-transition-group
- Add additional component animations using react-animation-components

React Animations: Additional Resources

PDFs of Presentations

7-Animations.pdfPDF File

React Animations

- Animation Add-Ons
- react-transition-group
- React Transition Group Documents
- react-animation-components

Other Resources

- How to build animated microinteractions in React
- UI Animations with React—The Right Way
- React Animations in Depth
- What’s the most developer-friendly React animation library?

- Amazing React animation with react-pose

In this assignment, you will update the web application to get data from the server to render the information corresponding to the leadership team of the company. In addition, you will handle the submission of the feedback form by posting the feedback to the server.

Step-By-Step Assignment Instructions

Assignment Overview

At the end of this assignment, you should have completed the following:

- Introduced new action types and action creators to support the fetching of the leaders information from the server and update the Redux store.
- Updated the Home and the About component to render the information about the leaders using the downloaded data from the server
- Add simple animations to the About component where the leaders information is displayed.
- Enabled the users to submit feedback through the feedback form by creating a new feedback service that accepts the form data and uses Restangular to record their feedback on the server.

Assignment Requirements

Task 1

In this task, you will update the Redux actions and the Home and About components to use the data from the server for rendering the leader information:

- Add new action types in *ActionTypes.js* to support the fetching of the leaders information from the server
- Add new action creators in *ActionCreators.js* to enable the fetching of the leaders information from the server and update the Redux store
- Update the code in *leaders.js* to respond to the dispatched Redux actions and update the Redux store and appropriately handle the loading and errors.
- Update the code in *MainComponent.js* to fetch and use the leaders information.
- Update *HomeComponent.js* to render the leader information.
- Update *AboutComponent.js* to render the leaders information. You should handle the loading and error condition appropriately.

Task 2

In this task, you will enable the saving of the feedback data submitted using the feedback form in the Contact component. You will save the feedback form data submitted by the user to the server:

- Implement a new action creator named postFeedback() that takes a Feedback object as a parameter and submits the feedback to the server using Fetch. Recall that the feedback data is accessible at http://localhost:3001/feedback on the json-server.
- Update *MainComponent.j*s to make the new dispatch method postFeedback() available to ContactComponent.

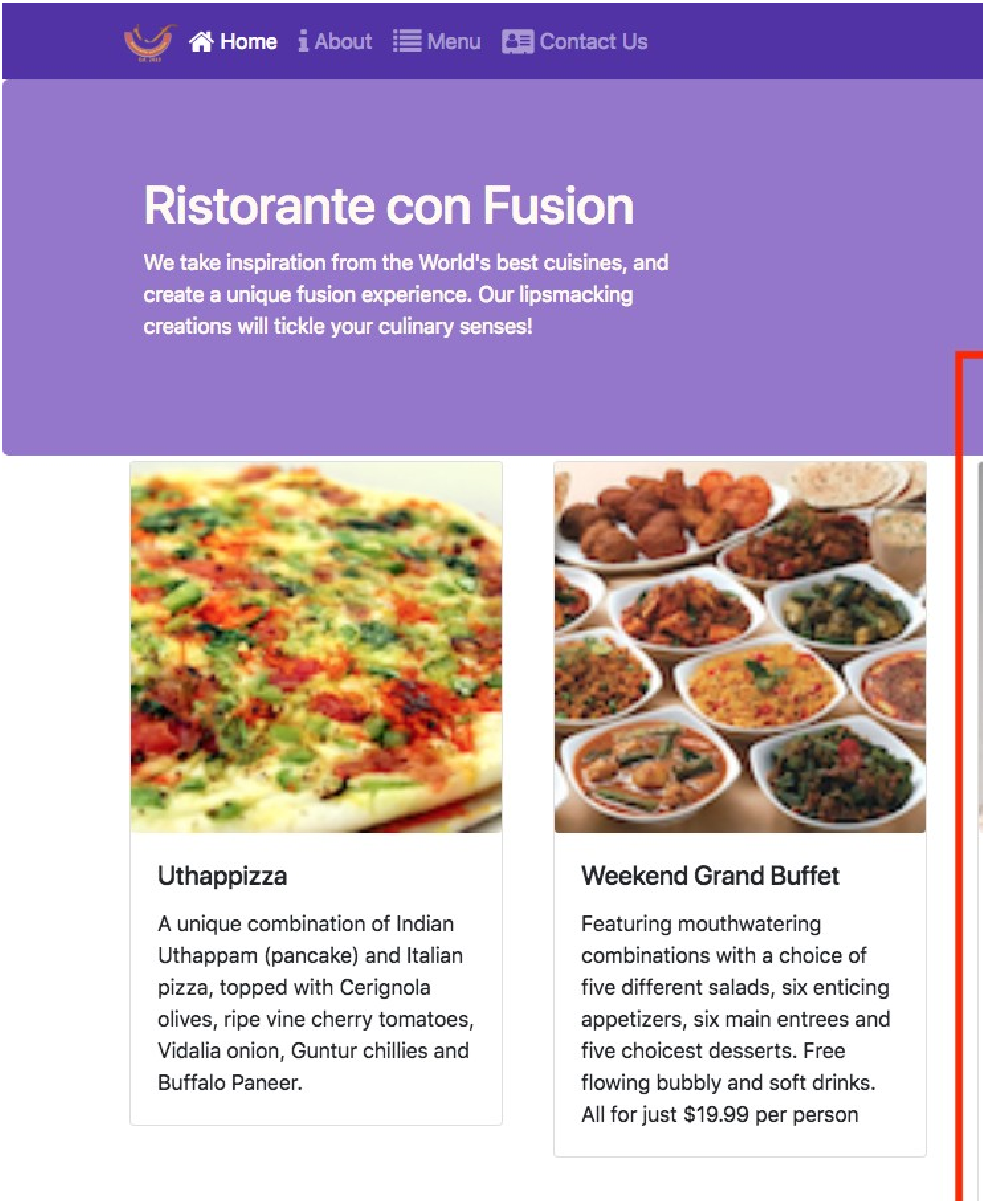
- Update the *ContactComponent.js* to submit the form data using the postFeedback() method by passing the feedback form data.

Task 3

In this task you will use simple animation using react-animation-components to enable a staggered rendering of the list of leaders in AboutComponent:

- Use the expand animation that we have already used earlier to judiciously apply animation to the various stages of the form submission.

Screenshots



Ristorante con Fusion

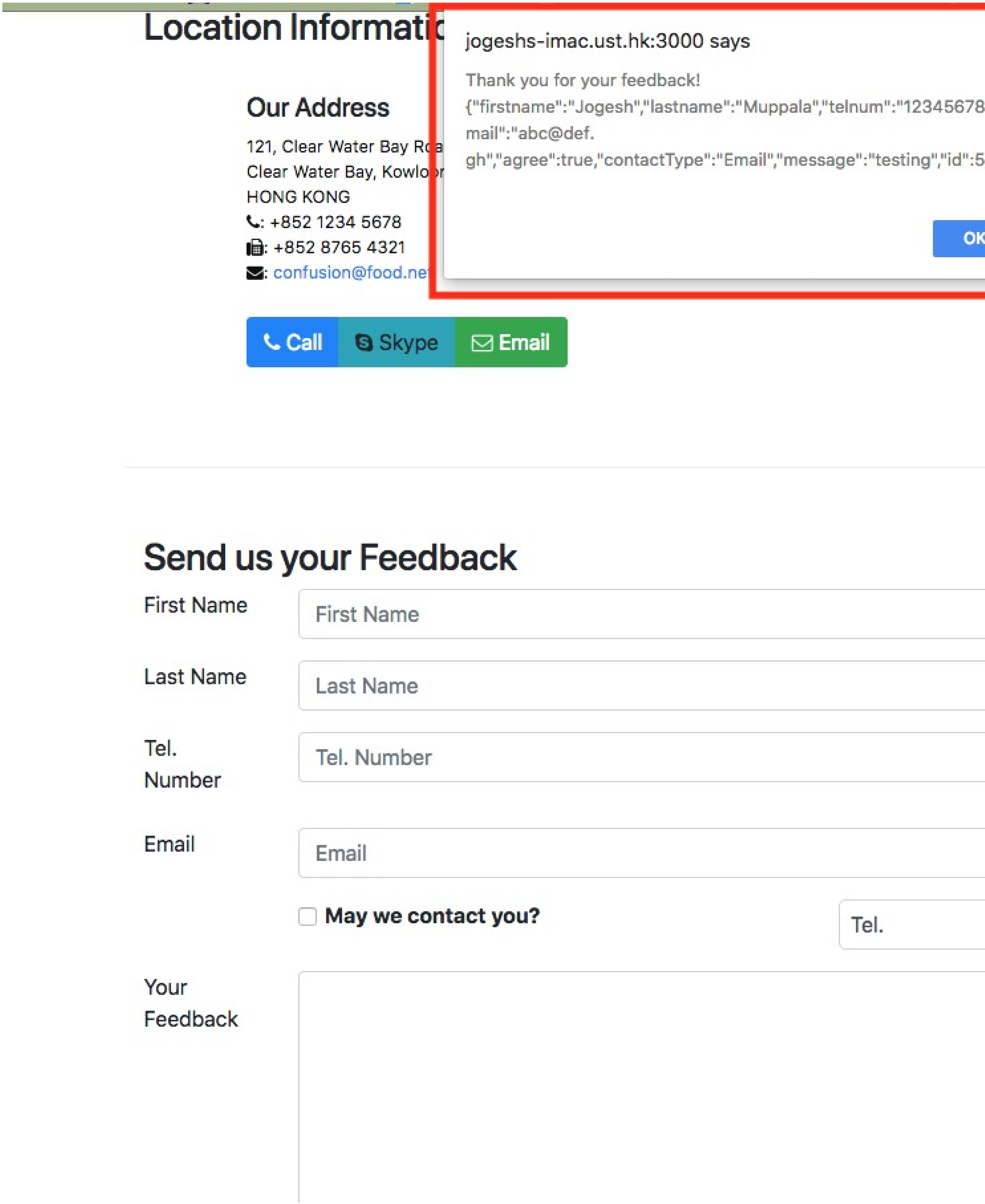
We take inspiration from the World's best cuisines, and create a unique fusion experience. Our lipsmacking creations will tickle your culinary senses!

Uthappizza

A unique combination of Indian Uthappam (pancake) and Italian pizza, topped with Cerignola olives, ripe vine cherry tomatoes, Vidalia onion, Guntur chillies and Buffalo Paneer.

Weekend Grand Buffet

Featuring mouthwatering combinations with a choice of five different salads, six enticing appetizers, six main entrees and five choicest desserts. Free flowing bubbly and soft drinks. All for just $19.99 per person



Location Informatio

Our Address

121, Clear Water Bay Ro
Clear Water Bay, Kowloo
HONG KONG
☎: +852 1234 5678
📠: +852 8765 4321
✉: confusion@food.ne

☎ Call | Skype | ✉ Email

jogeshs-imac.ust.hk:3000 says

Thank you for your feedback!
{"firstname":"Jogesh","lastname":"Muppala","telnum":"12345678
mail":"abc@def.
gh","agree":true,"contactType":"Email","message":"testing","id":5

OK

Send us your Feedback

First Name   First Name

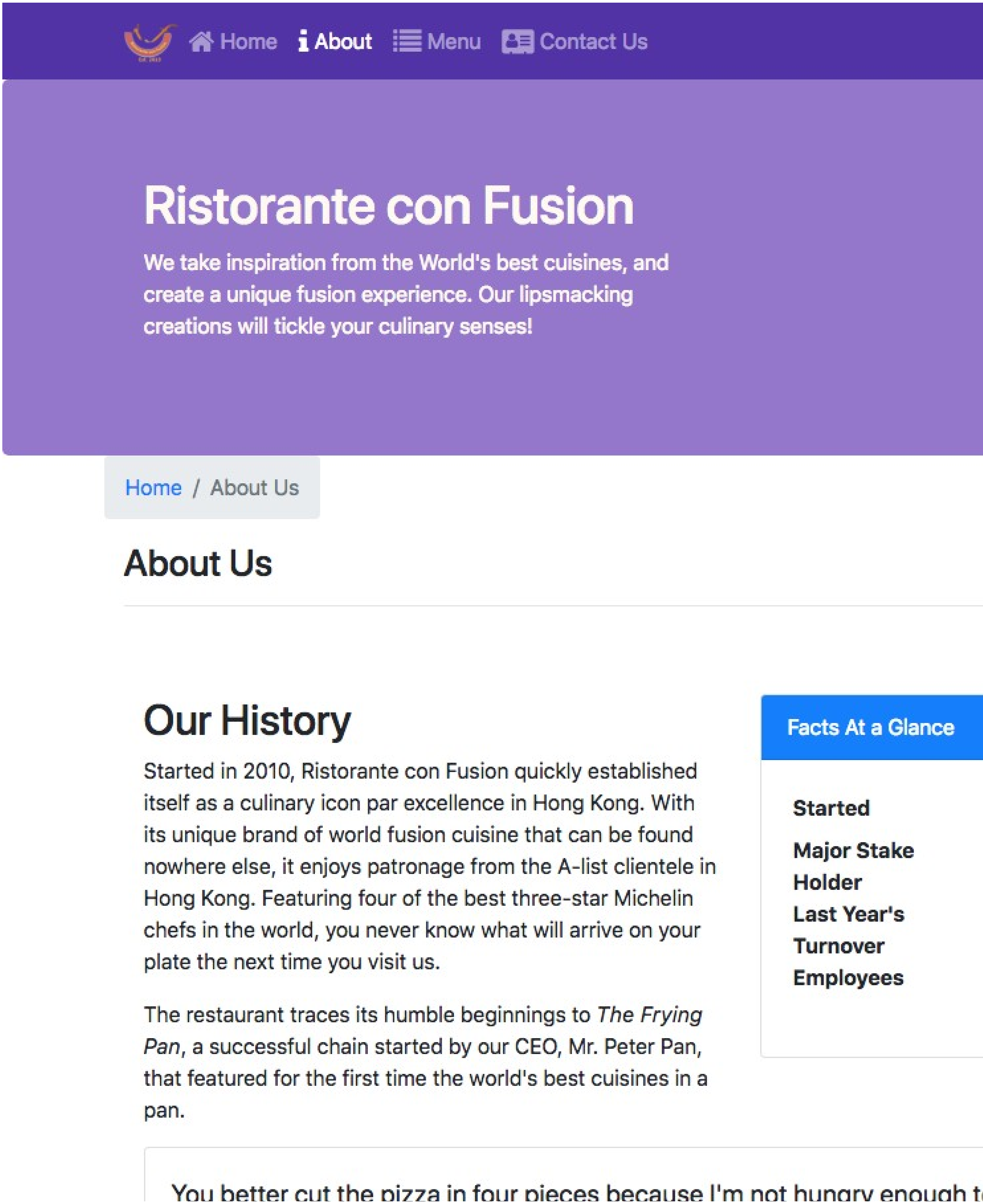
Last Name   Last Name

Tel. Number   Tel. Number

Email   Email

□ **May we contact you?**   Tel.

Your Feedback



Ristorante con Fusion

We take inspiration from the World's best cuisines, and create a unique fusion experience. Our lipsmacking creations will tickle your culinary senses!

Home / About Us

About Us

Our History

Started in 2010, Ristorante con Fusion quickly established itself as a culinary icon par excellence in Hong Kong. With its unique brand of world fusion cuisine that can be found nowhere else, it enjoys patronage from the A-list clientele in Hong Kong. Featuring four of the best three-star Michelin chefs in the world, you never know what will arrive on your plate the next time you visit us.

The restaurant traces its humble beginnings to *The Frying Pan*, a successful chain started by our CEO, Mr. Peter Pan, that featured for the first time the world's best cuisines in a pan.

Facts At a Glance

Started

Major Stake Holder

Last Year's Turnover

Employees

You better cut the pizza in four pieces because I'm not hungry enough t

Review criteria

Your assignment will be assessed based on the following criteria:

Task 1

- Appropriate action types and action creators have been added.
- The Home component is correctly using the leader data, and handling any errors that might arise.
- The About component is correctly using the leader data, and handling any errors that might arise.

Task 2

- A new postFeedback() action creator is correctly implemented to post the feedback data to the server.
- The Contact component has been correctly updated to use postFeedback() to post the form data to the server.

Task 3

- Appropriate animation has been added to stagger the rendering of the leaders in the AboutComponent.

Assignment 4: Redux, Client-Server Communication and Fetch: Additional Resources

React Resources

- Redux Actions
- Redux Reducers
- Redux Usage with React
- Cross-Fetch

- Animation Add-Ons
- react-transition-group
- React Transition Group Documents
- react-animation-components

Building and Deployment: Objectives and Outcomes

In this lesson you will learn about Webpack and how react-scripts uses webpack to package your React application to create a distribution folder. At the end of this exercise you will be able to:

- Understand the Webpack way of packaging applications into bundles
- Use react-scripts to build a distribution folder with your React application bundled using Webpack

Exercise (Instructions): Building and Deploying the React Application

Objectives and Outcomes

In this exercise you will learn to use the react-scripts to build a distribution folder with the set of application files that can be copied to a server to deploy your React application. At the end of this exercise you will be able to:

- Build your React application using the react-scripts to create a distribution folder
- Deploy your application to a server by copying the built files to your server

Building the Distribution Folder

- To build the distribution folder containing your React application, type the following at the prompt:

```
npm run build
```

- This should build a distribution folder named *build* containing all your application files.

Deploying your React Application

- To deploy your React application you need a server. Fortunately we already have the json-server available on our computer.
- Copy the contents of the build folder to the public folder of your json-server
- Now your React application can be accessed at the link http://localhost:3001/.
- If you are setting up a server on the cloud or anywhere, all that you need to do is copy the contents of the build folder to the server side to deploy your React application. The exact procedure depends on the cloud service provider that you choose to use. Please consult their documentation to see the procedure to set up the server.

Conclusions

In this exercise you learnt to use the react-scripts to build and deploy your React application.

Building and Deployment: Additional Resources

PDFs of Presentations

8-Webpack.pdfPDF File

React Resources

- Webpack: an Introduction
- create-react-app Build

Other Resources

- Webpack
- Webpack on Github