Talent Developers - Onboarding Task

Introduction:
Welcome! The Talent Onboarding Task is designed to familiarise you with technologies that you will be utilising once you join the project team. While the instructions tell you what to do, they do not always necessarily tell you how to do it. This is deliberate as it is important for you to develop an independent drive to solve problems on your own. You are **not allowed** to ask your tutors or senior interns in MVP studios for help at this stage. You are allowed to discuss with other interns in your intake while working on the task.

Warning:

There are a couple of ways to install React. If you are reading other tutorials on React, then be aware that the things taught in the tutorial might not be used in the onboarding task. For example, react routing might not using in the onboarding task. Also, following online tutorials and installing specific versions of npm packages might cause conflicts between different package versions.

Goal:

- Be able to build an MVC application with Visual Studio and connect it to an SQL database
- Understand how to create and use React JS components

Project summary:

- MVC application connected to a SQL database
- React JS front-end, Semantic-UI-React
- Entity Framework, Data annotations

Requirements:

Back-end
- Create an MVC application that enables the CRUD (Create, Update, Delete) operations.
- Please add data annotations to the models

Front-end
- React JS components should be written using ES6 classes and jsx
- Create controlled components. Use state.
- Use AJAX inside React components to post to/get from the controller
- **Please use plain javascript not typescript**
- Use Semantic-UI-React modals in views

Resources:

ReactJS:

1. ReactJS tutorial
    - https://reactjs.org/docs/hello-world.html
2. React sample code (written in jsx) can be found here
    - The code is not 100% accurate. This is just to give you an idea on how to start.

MVC 5 information:

- https://docs.microsoft.com/en-us/aspnet/mvc/overview/getting-started/introduction/getting-started

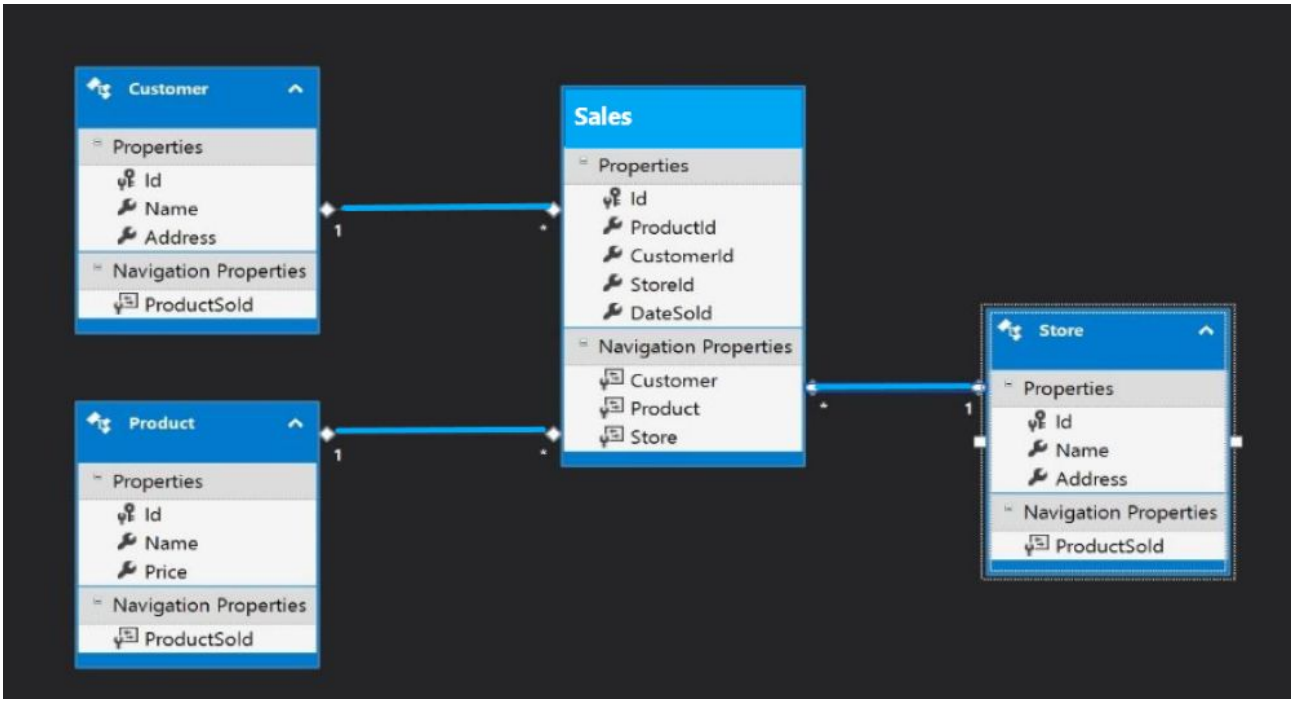
Data annotations:

- http://www.dotnettricks.com/learn/mvc/mvc-data-annotations-for-model-validation

Tutorial: https://www.youtube.com/watch?v=bnFgGYooDCM

Instructions:

- Create the database, tables and map the table relationships using SQL Server based on the model diagram below.



- Set up the ReactJS environment by downloading this boilerplate from github

Preferred method(m1):

- Please install Visual Studio 2017 or later, and use the .net core with react template provided by visual studio.
- Click `File` on the top right hand side of the menu, `New` -> `Project` -> `ASP.NET core web application` -> do the basic config and click next/ create -> select `React.js` to create the project.
- You can react the .net and the npm at the same time by just clicking the run button

Other method:

- Clone the project from http://git.mvp.studio/react-onboarding/react-onboarding.git
  i. Run command prompt (Run as Administrator)
  ii. Open project folder (Boilerplate.Web.App) in cmd
  iii. Type '**npm install**' to install npm packages
  iv. Type '**npm run start**' to run webpack
  v. Run project from visual studio
- If the boilerplate does not work for you, try the following steps:
  Reactjs setup
  Complete guide with video

- Connect to the database

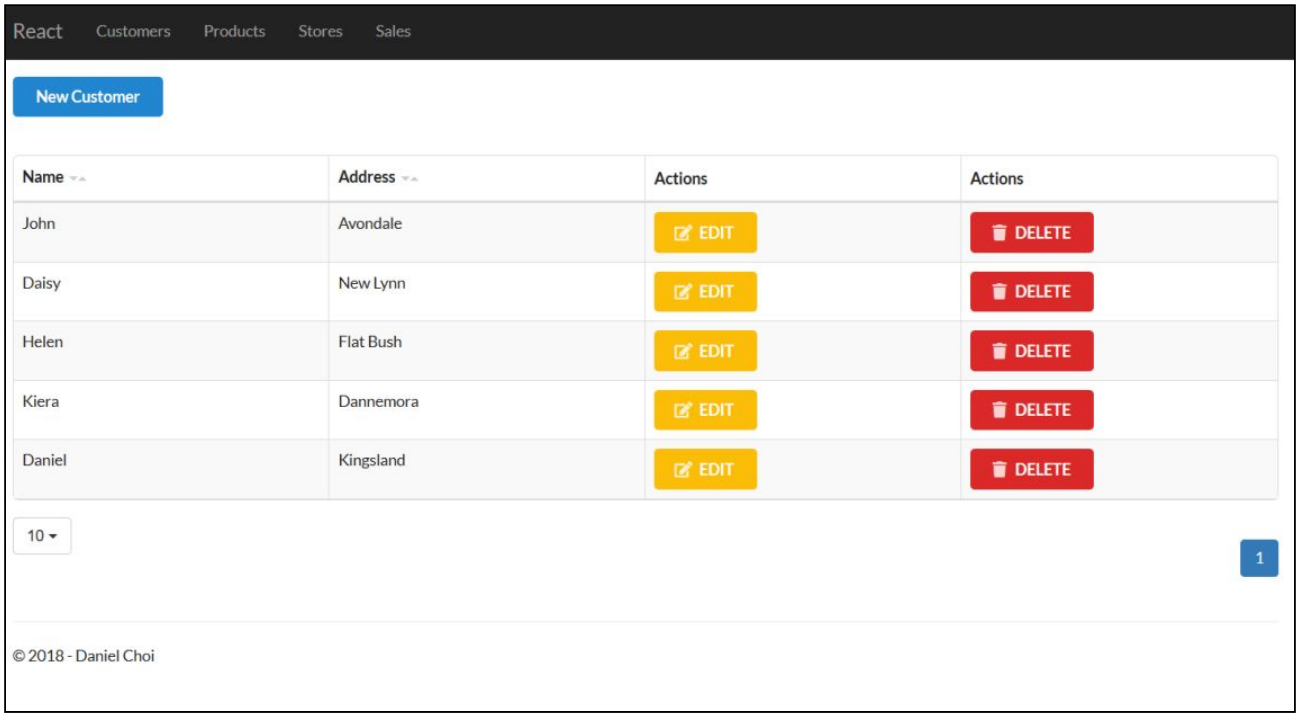
  Preferred method(m1):
  - Please use EF core, go to nuget package management, search for the package, and for learning EF core you can follow this
  - If you are using .NET Framework, use Entity Framework to connect to your database. (select ADO.NET Entity Data Model)
  - If you are using .NET Core (the boilerplate), then check out this article
- Build a controller to fetch, create, update and delete data from the Customer table.
- Build the Front-End using ReactJS and connect it to the controller.
- Repeat steps 3-4 for the Store, Product and Sales UIs.

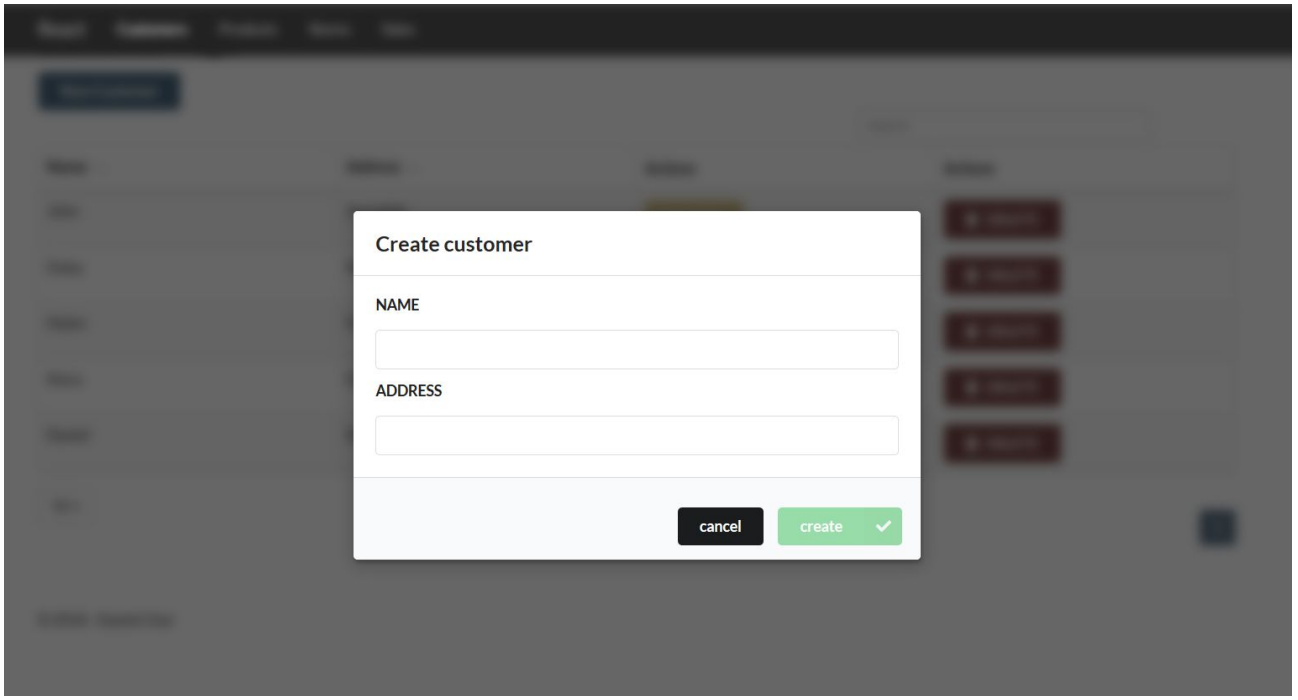
Here are some helpful online resources:

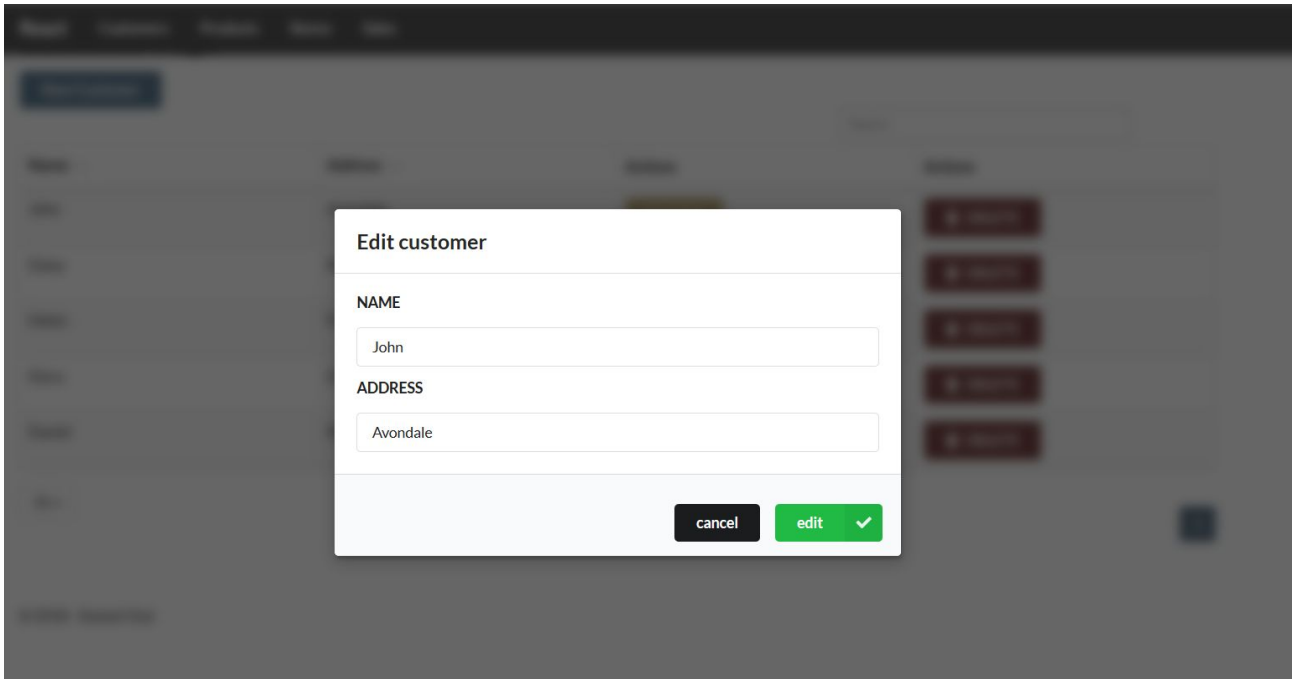
- Setup React with ASP.NET MVC
  - https://www.youtube.com/watch?v=bnFgGYooDCM
- Getting started with ASP.NET Core MVC
  - https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/start-mvc?view=aspnetcore-2.2&tabs=visual-studio

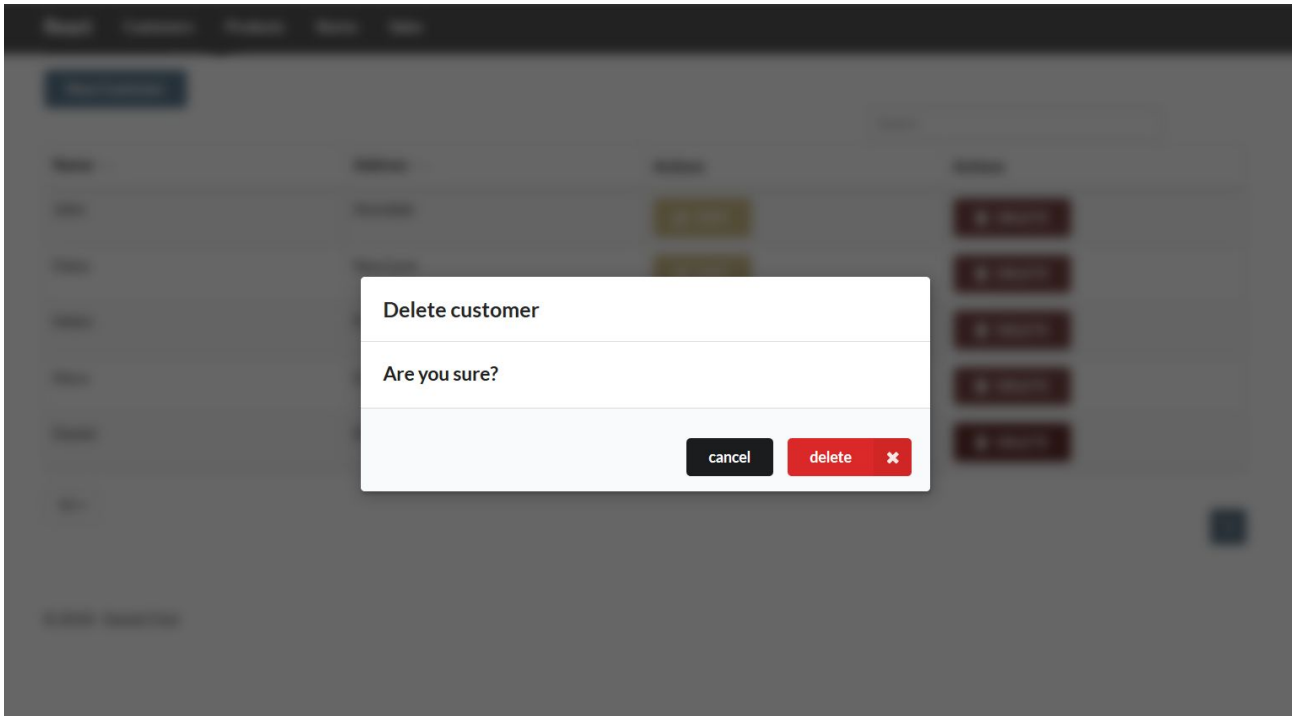
The screenshot examples of the project can be found below:

Customer view (with semantic ui modals):

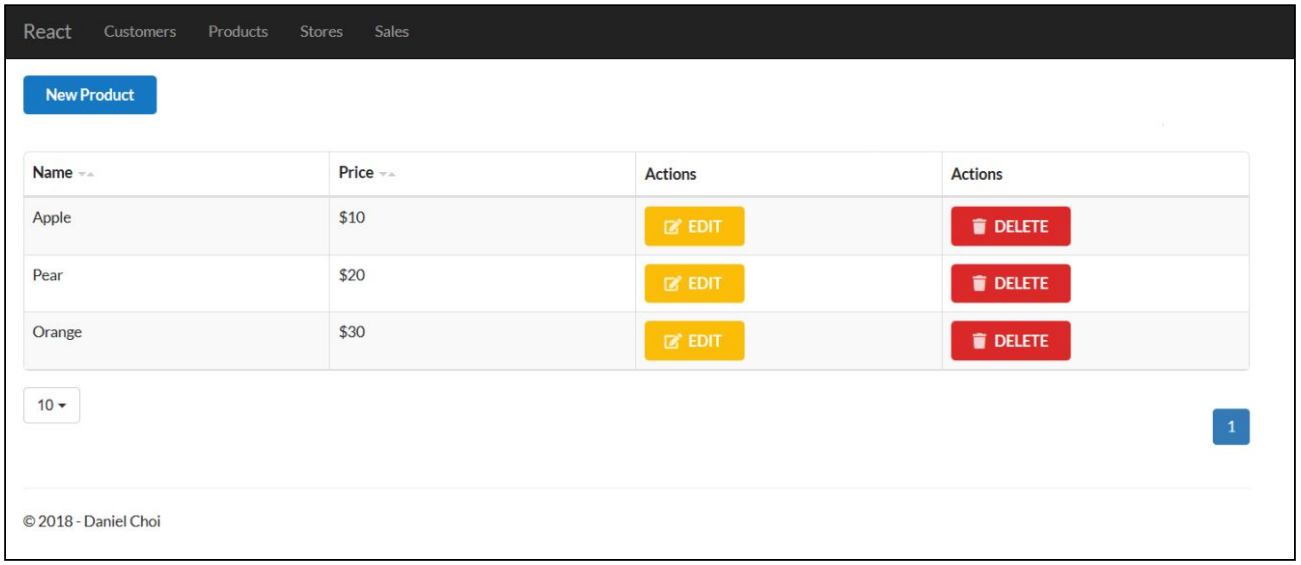


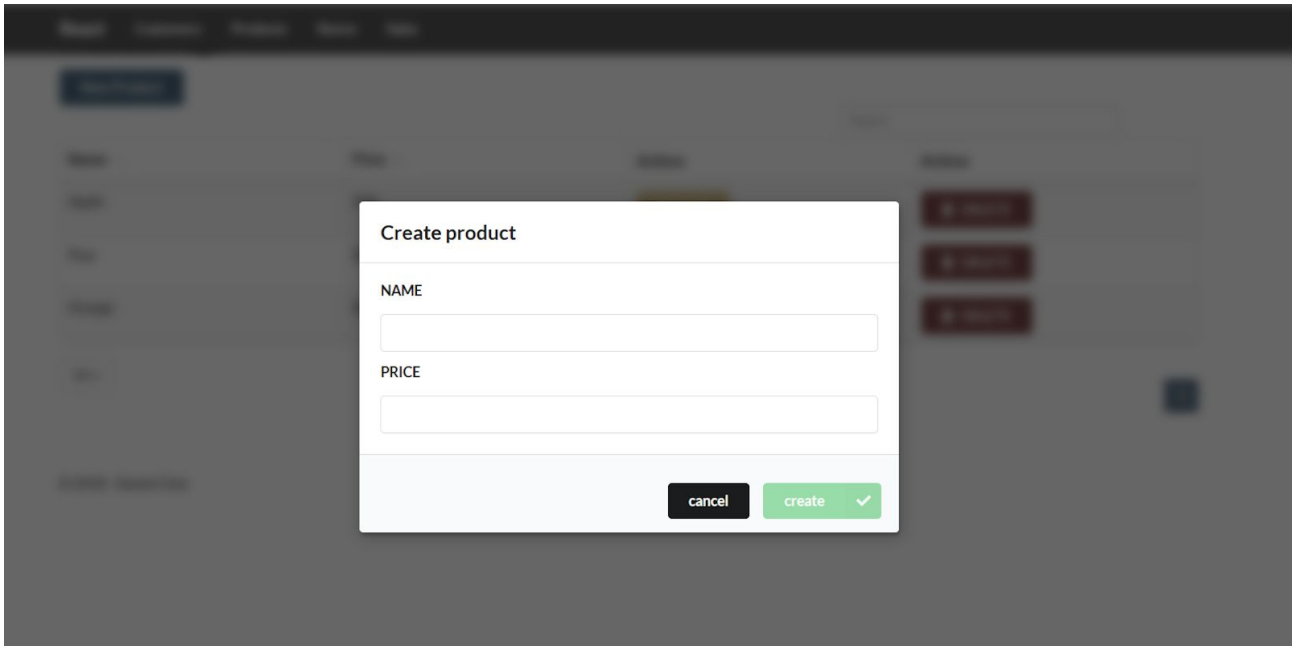


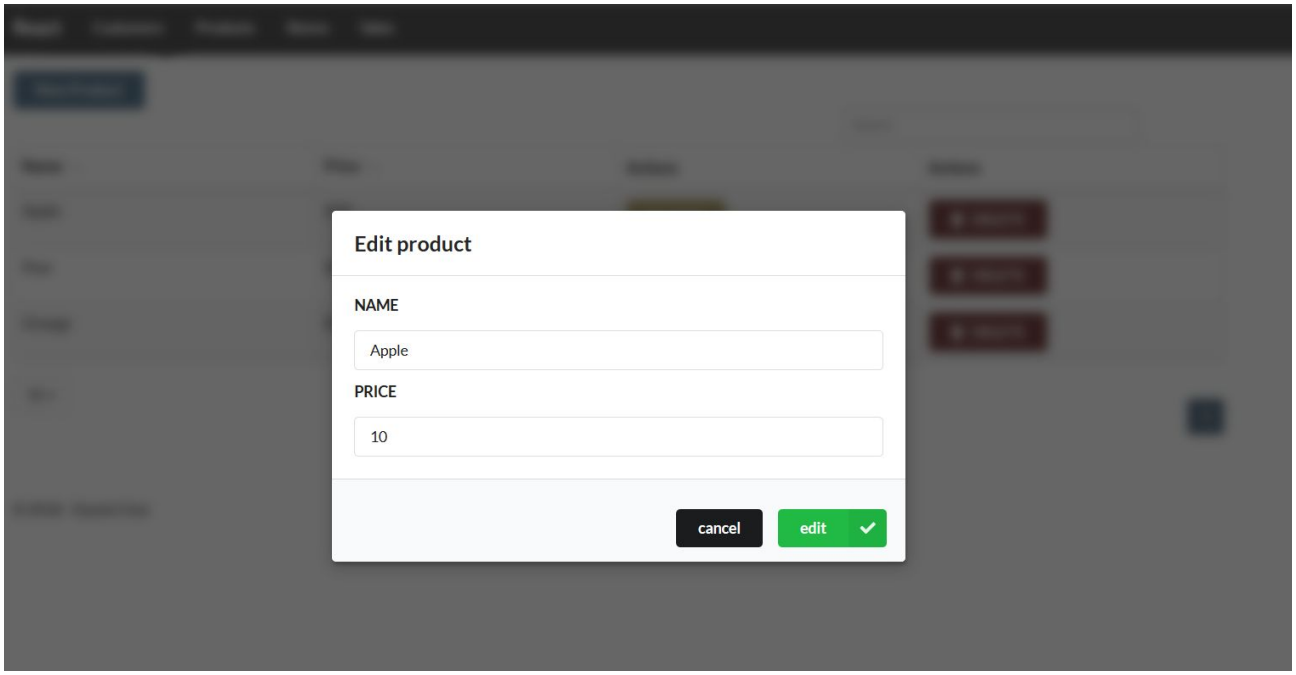

Edit customer
NAME
John
ADDRESS
Avondale
cancel
edit

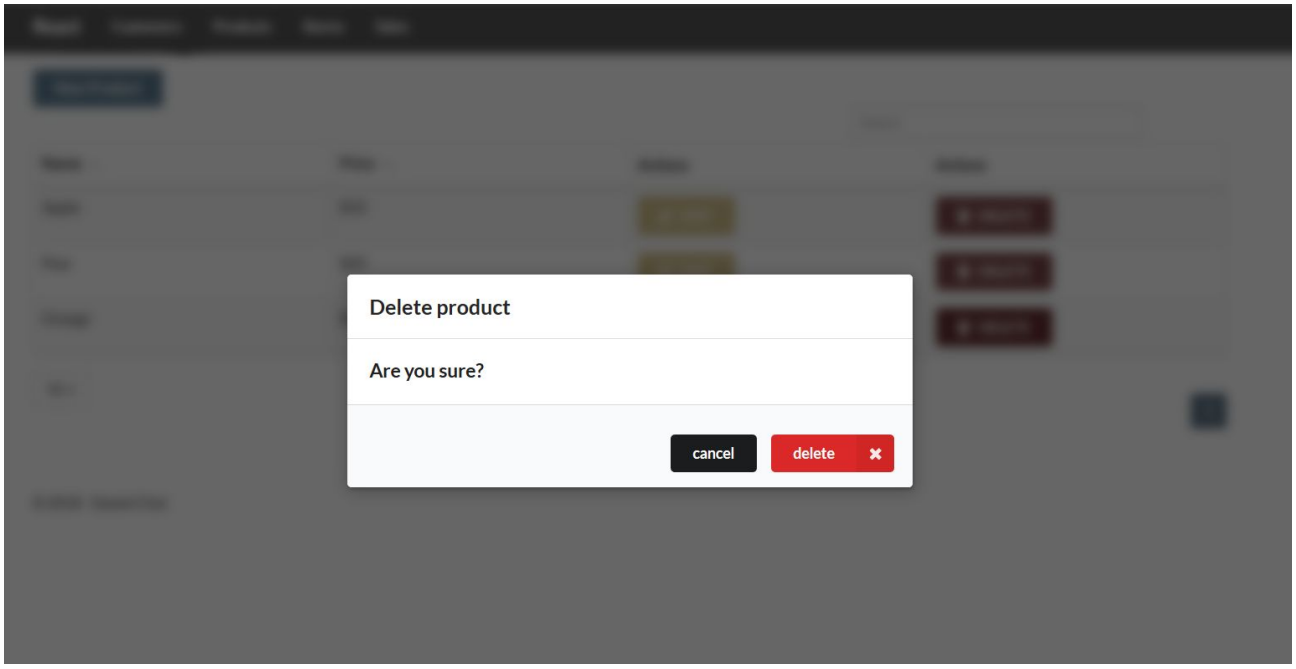

Delete customer
Are you sure?
cancel
delete

Product view:

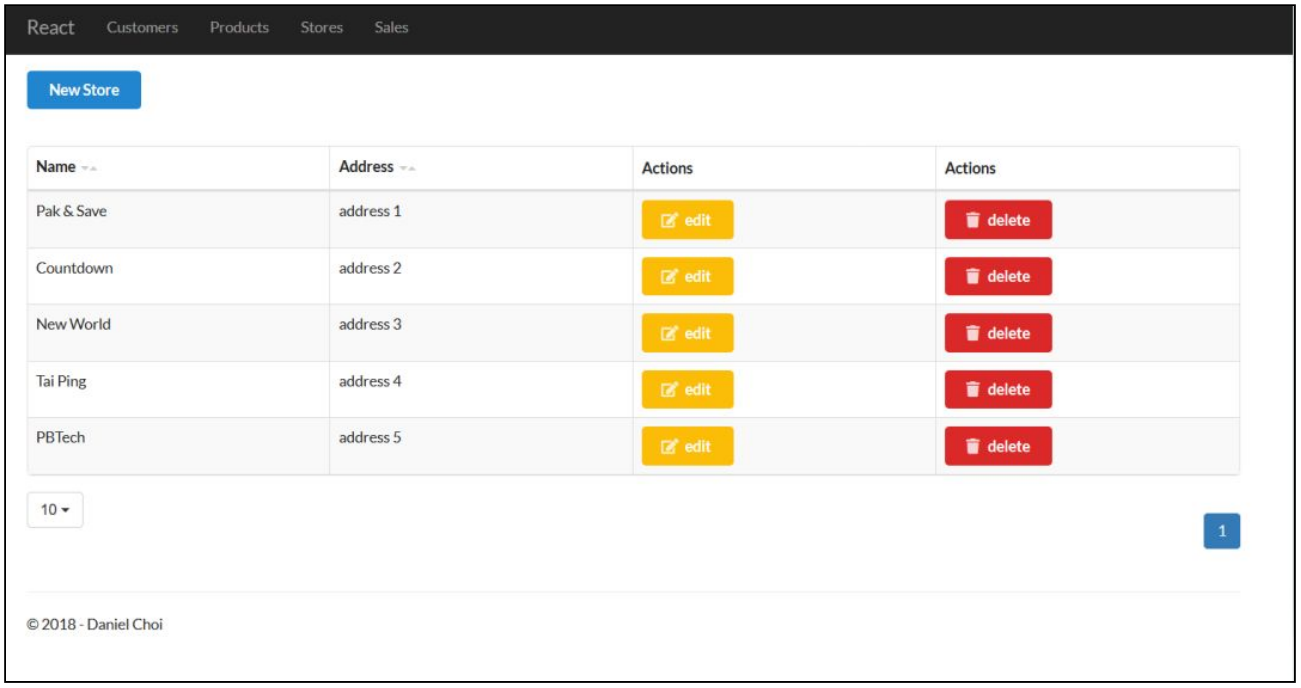


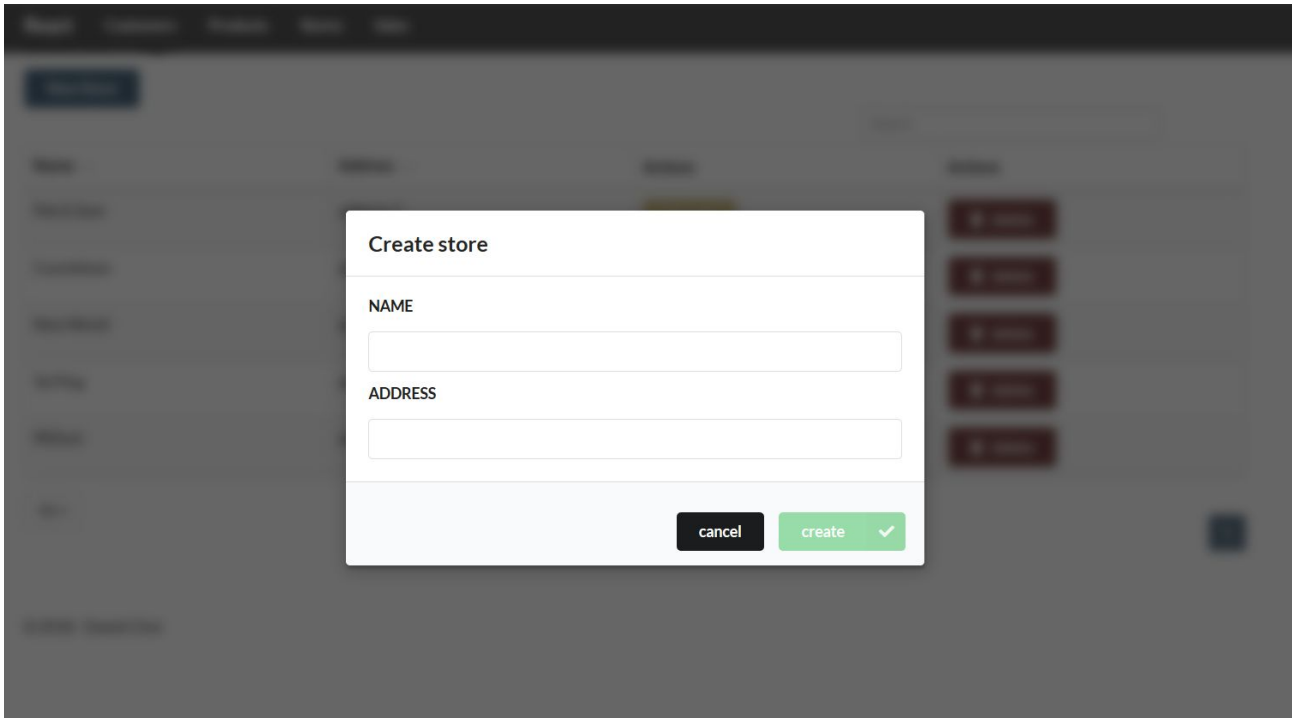


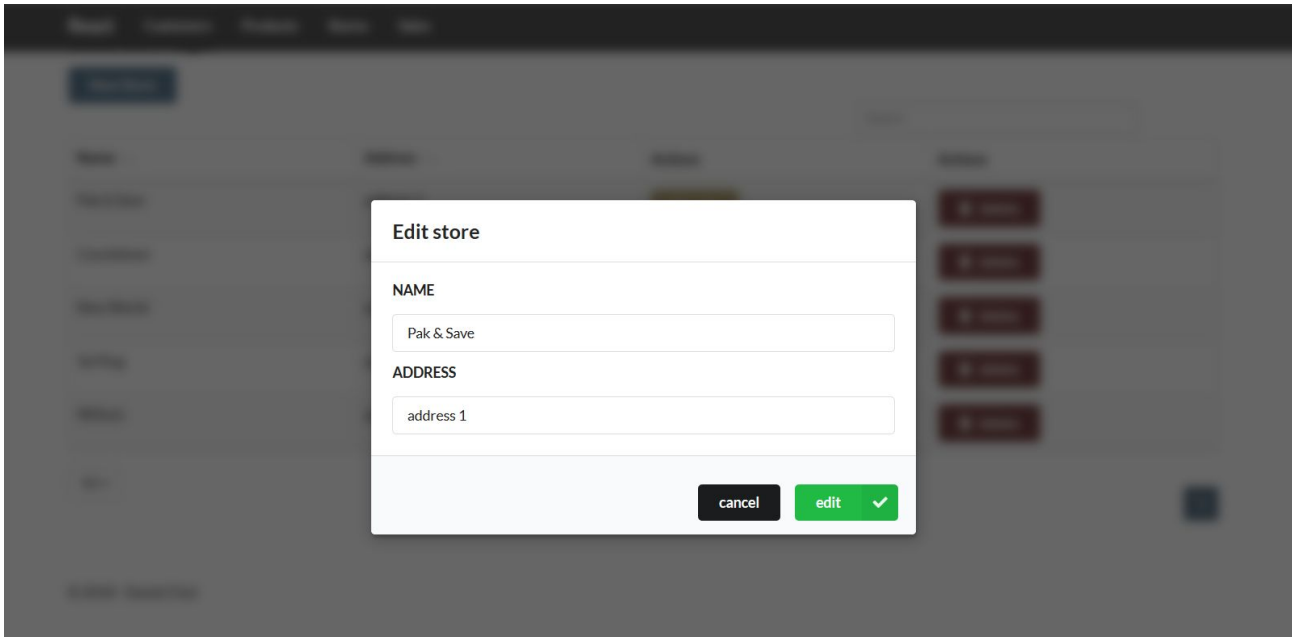

Edit product
NAME
Apple
PRICE
10
cancel
edit

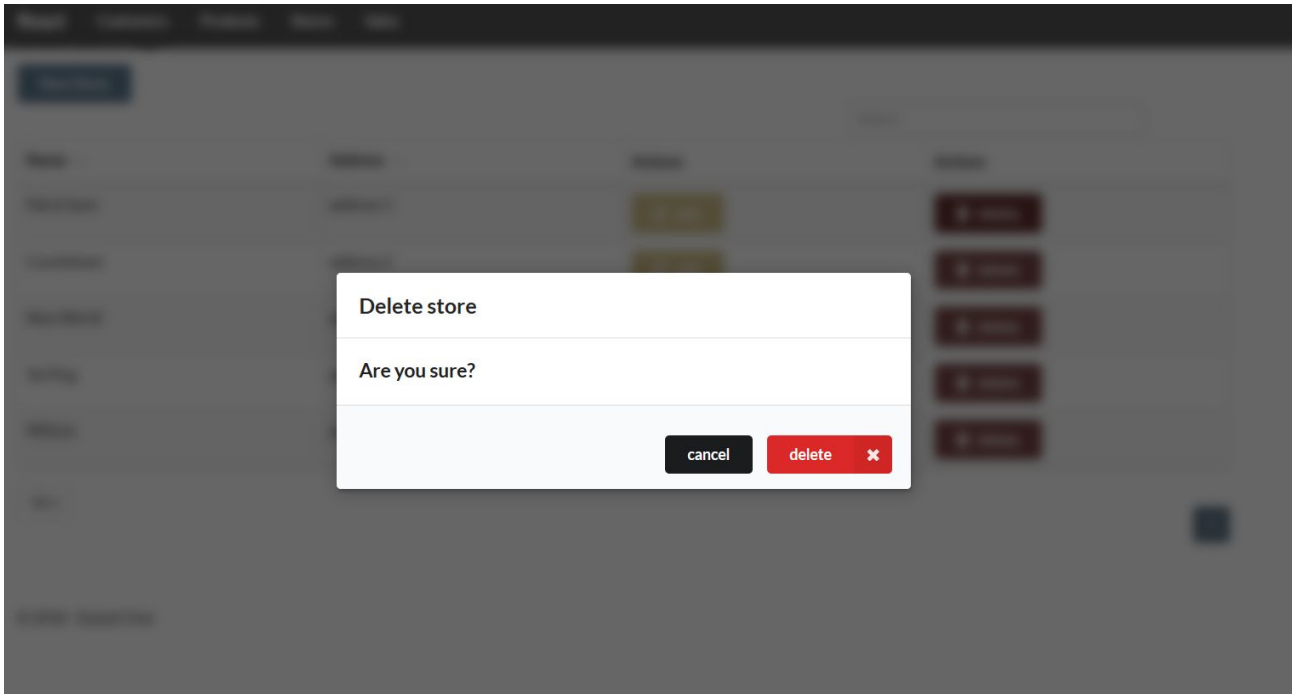

Delete product
Are you sure?
cancel
delete

Store view:

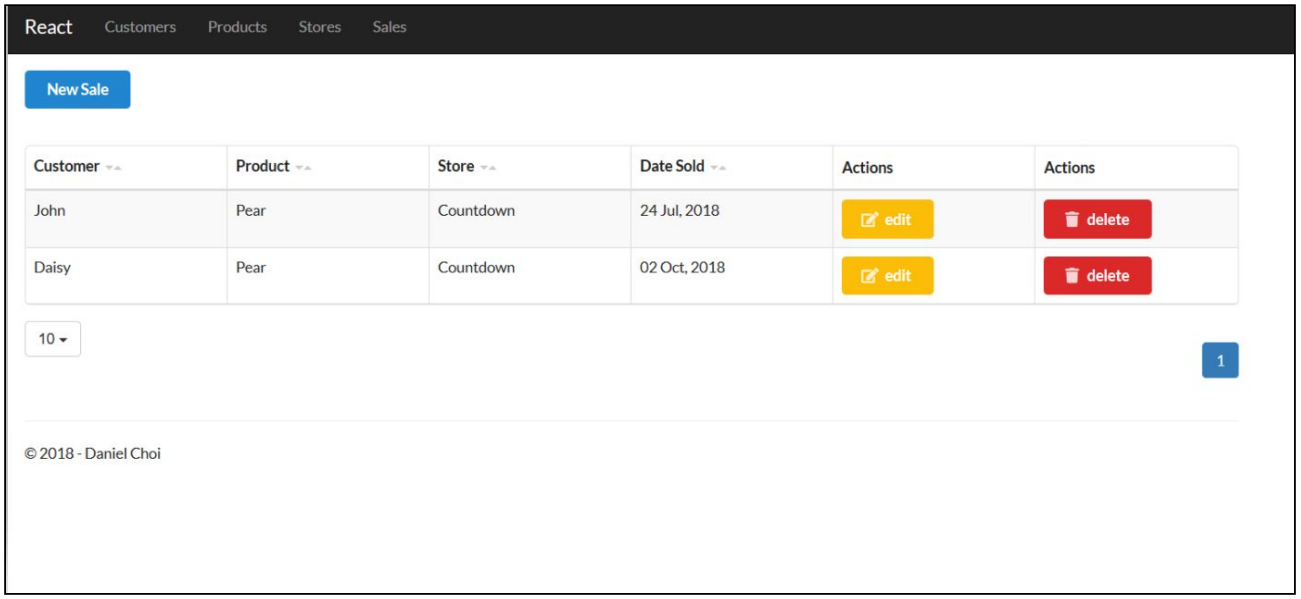

React
Customers
Products
Stores
Sales
New Store
Name
Address
Actions
Actions
Pak & Save
address 1
edit
delete
Countdown
address 2
edit
delete
New World
address 3
edit
delete
Tai Ping
address 4
edit
delete
PBTech
address 5
edit
delete
10
1
© 2018 - Daniel Choi

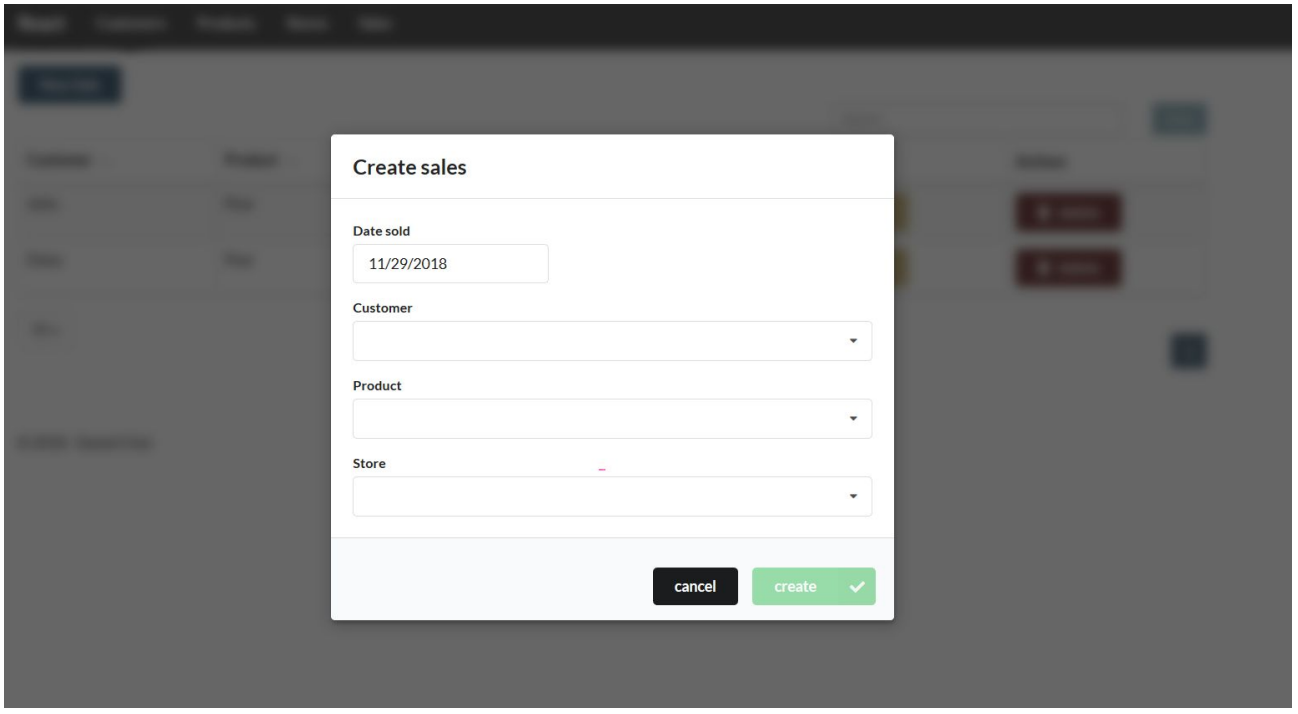

Create store
NAME
ADDRESS
cancel
create

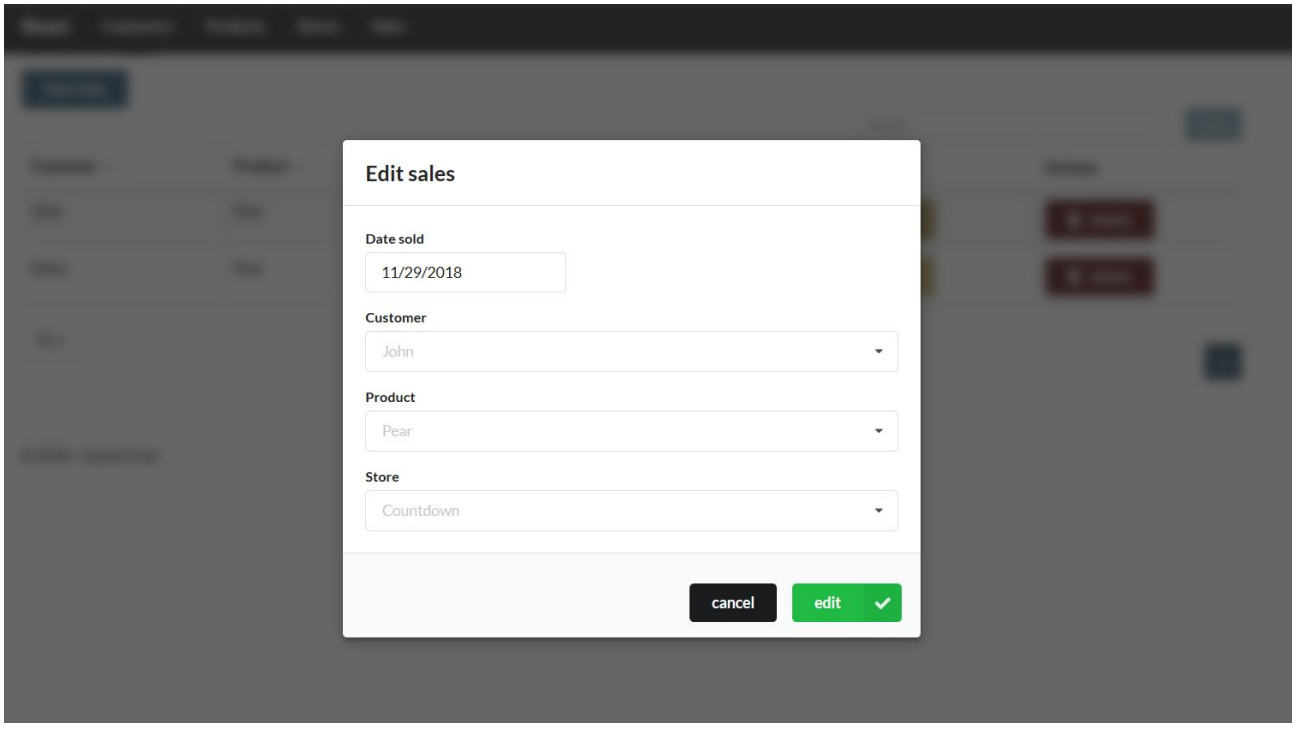

Edit store
NAME
Pak & Save
ADDRESS
address 1
cancel
edit

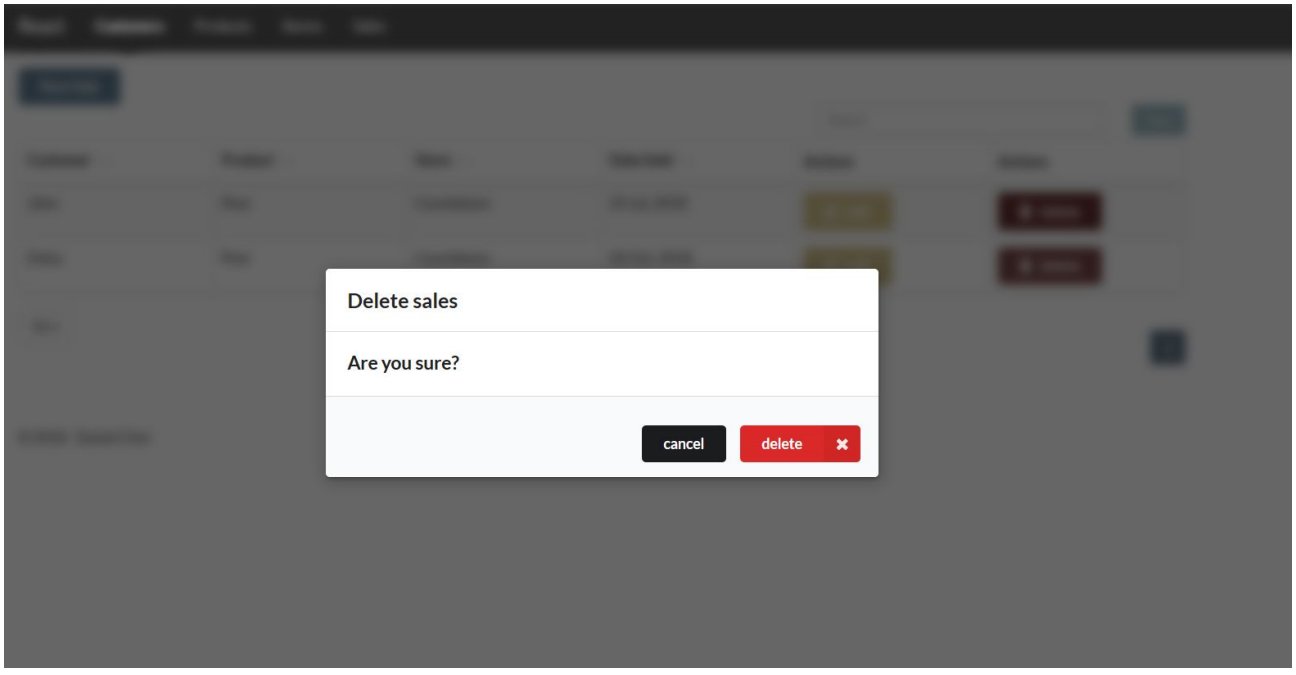

Delete store
Are you sure?
cancel
delete

Sales view:


React
Customers
Products
Stores
Sales
New Sale
Customer
Product
Store
Date Sold
Actions
Actions
John
Pear
Countdown
24 Jul, 2018
edit
delete
Daisy
Pear
Countdown
02 Oct, 2018
edit
delete
10
1
© 2018 - Daniel Choi


Create sales
Date sold
11/29/2018
Customer
Product
Store
cancel
create


Edit sales
Date sold
11/29/2018
Customer
John
Product
Pear
Store
Countdown
cancel
edit

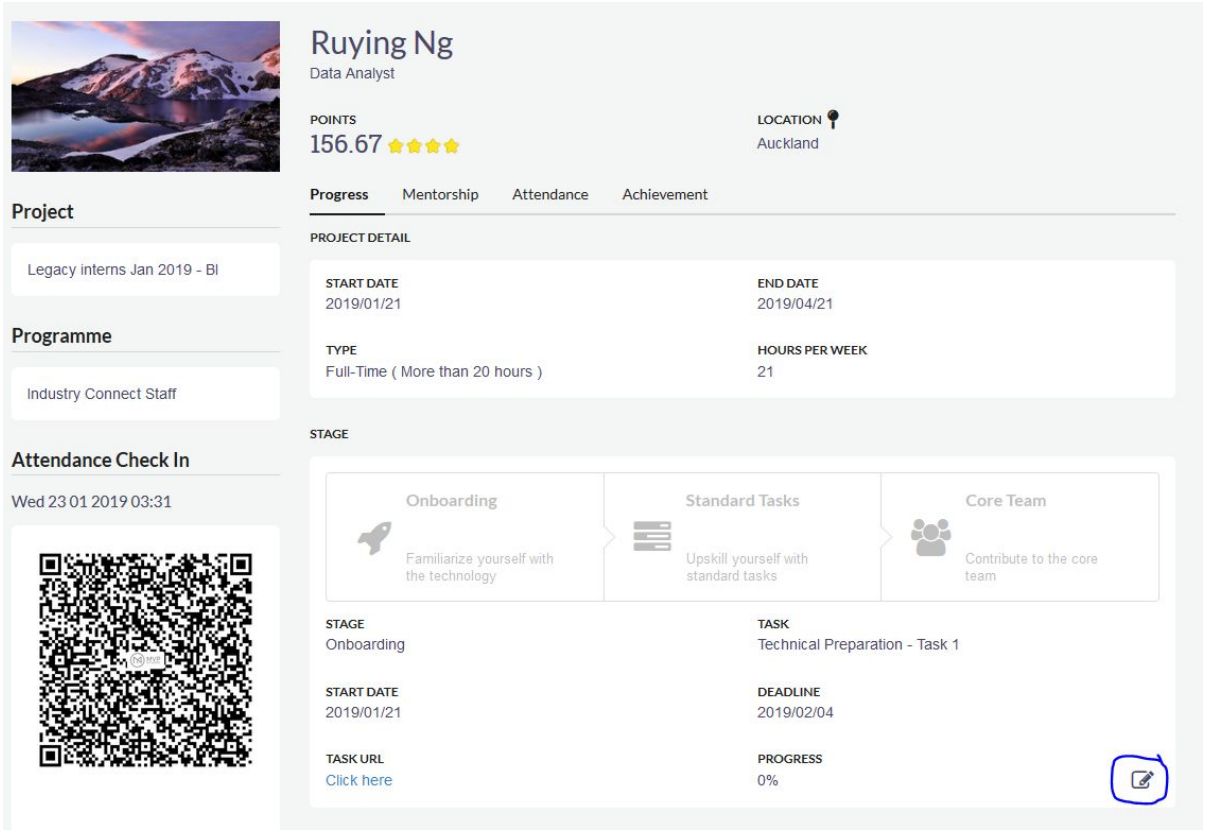

Delete sales
Are you sure?
cancel
delete

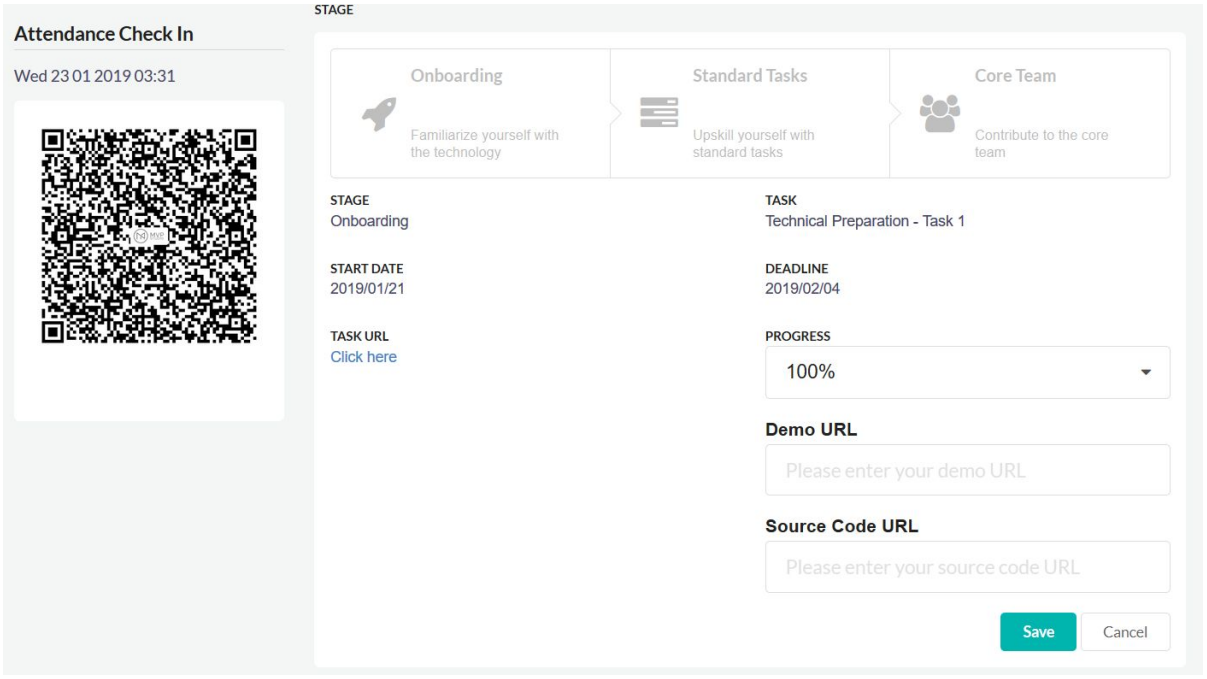
Submission

Below are the steps to submit your project for evaluation:

1. Publish your project on Azure and upload your source code on GitHub. More info
2. Submit your **GitHub url** and **Azure website url** on the Internship Portal at the IndustryConnect.IO website. Click on the edit progress icon and enter the urls.
    a. Demo Url: Azure website url
    b. Source Code Url: GitHub url

Note: Your progress has to be at least 80% to submit the urls, and 100% to view your next task





1. How to deploy your code on Azure:How To Publish/Deploy ASP .Net MVC Website on Azure Server From Scratch
    - https://www.youtube.com/watch?v=31kxmM-WZLU
    - https://www.codeproject.com/Tips/1044950/How-to-Publish-ASP-NET-MVC-Web-Application-to-Azur

2. How to create Azure SQL Database In Azure Portal:
    - https://www.youtube.com/watch?v=uzezbA5g3FI
    - https://docs.microsoft.com/en-us/azure/sql-database/sql-database-design-first-database

3. How to add An Existing Project To Your GitHub Repo With Visual Studio:
    - https://www.youtube.com/watch?v=kHM3xzkou_0

4. How to setup Git Account
    - https://help.github.com/articles/set-up-git/