

Overview

You have been tasked with adding some new functionality to the Proctors Developer Store. Currently the site just has general information on it but the client would like a store finder.

We have provided you with a Drupal installation, a 'Store' content type, a custom module 'store_locator', a view called 'Store Locator' and an example CSV file containing the stores.

The user admin account is **superuser** and the password is **superuser**.

Task 1 - CSV Store Importer

You will be given a CSV file containing the list of stores and will need to write an importer that creates or updates store nodes.

The custom module ([sites/all/modules/custom/store_importer](#)) has a menu callback already defined which you should use: [store_locator/import](#).

For simplicity, you don't have to deal with file uploads. The user will just paste the CSV content in to the box.

Some of the things we are looking for are:

- Data being imported in to the correct fields.
- Checking if a node already exists.
- Any performance considerations.

There is a CSV with sample data to download in the instructions folder (/instructions/store_data.csv) or [click here](#).

Task 1 - Extension

The client doesn't know the Latitude and Longitude of all their stores. Use the Google maps geocoding API to grab them as you import the data.

<https://developers.google.com/maps/documentation/geocoding/intro>

You should be able to grab the details from Google in a few lines of code.

You shouldn't need an API key for this. But here is one you can use if you really need to:

AlzaSyDkn1amsOkmbqBDQl0uJlorDNfRYbnk2u8

Task 2 - Store locator 'View' configuration

The client wants users who visit the site to be able to find the stores. We have given you a basic view http://proctorsdevtest.local/admin/structure/views/view/store_locator that lists store content types.

Configure the view to show a list of stores, and show

- Store name
- Store type
- A link to view the full details.

The user should also be able to search for a store by name, and filter by store type.

Task 2 - Extension

The client wants the stores of type 'Superstore' to appear at the top of the list by default.

Modify the view to achieve this.

Task 3 - Store Page

The designer has provided you with a "design". Your job is to make the store page match the layout of the design below.

You may create a custom page template, or use the display functionality.

Then use some custom CSS (You will find a blank CSS file in the store_locator module 'sites/all/modules/custom/store_locator/store_locator.css') to arrange the elements on the page.

Some of the things we are looking for are:

- Sensible HTML tags.
- Elements that will deal with different content sizes.

You will find the full wireframe in the folder 'instructions' if you need it.

<div style="display: flex; justify-content: space-between; align-items: center;"> Logo Home - About - Store Locator </div>	
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <input style="width: 80%;" type="text"/> <input style="width: 20%;" type="button" value="GO"/> </div> <div style="border: 1px solid black; height: 150px; margin-top: 10px;"></div>	<div style="border: 1px solid black; background-color: #f0f0f0; padding: 5px; margin-bottom: 10px;"> Store Name – Store Type </div> <div style="display: flex;"> <div style="flex: 1; padding-right: 10px;"> <p>Store Manager</p> <p>Address 1</p> <p>Address 2</p> <p>Address 3</p> <p>Google Maps Link</p> </div> <div style="flex: 1; border: 1px solid black; padding: 5px;"> <div style="background-color: #f0f0f0; padding: 2px; margin-bottom: 5px;">Opening Times</div> <p>Monday: 9.00 – 5.30</p> <p>Monday: 9.00 – 5.30</p> <p>Tuesday: 9.00 – 5.30</p> <p>Wednesday: 9.00 – 5.30</p> <p>Thursday: 9.00 – 5.30</p> <p>Friday: 9.00 – 5.30</p> <p>Saturday: 9.00 – 5.30</p> <p>Sunday: Closed</p> </div> </div>

Task 3 - Extension

Insert a link to Google maps that shows the location of the store.

Tips for using a vagrant/ansible box

- You have to wait a second or two for everything to sync to the virtual machine. Your code may not refresh immediately.
- Drush is available. You can run commands on the virtual machine using the alias @vagrant e.g. drush @vagrant cc all.
- If you make changes on the virtual machine (e.g. drush @vagrant dl devel) they will be overwritten next time it syncs. Make all changes to code locally (e.g. drush dl devel).