

Suburb and Surrounding Rings Widget

Description

A web widget to allow the selection and map view of initial suburbs and three (3) optional "rings" expanding outwards from the suburb.

Purpose: This widget will be embedded in web apps and sales tools running on both desktops and tablets (iPads) to assist clients in visualising the area of coverage available to them when selecting a listing type.

System Overview

The activity starts by the selection of an initial suburb.

(This selection will not typically be changed once a selection is made, however the widget must support changing this value after following selection have been made and refreshing the view appropriately)

Suburbs should be searchable by partial name or postcode with autocomplete functionality

Once a suburb is selected the user is able to select to view the list of suburbs constituting the 1st, 2nd, or 3rd rings of surrounding suburbs (multiple rings may be visible at the same time - in general this will be a progression).

In addition to displaying the list of suburbs in each ring the widget is required to display a map (e.g. Google map) with selected suburbs highlighted allowing user to see the initial suburb, 1st, 2nd, and 3rd rings individually or in combination.

Because the Widget will serve as an input mechanism it will return or make available 2 values:

- 1) The selected suburb (String = [Suburb Name],[Postcode])
- 2) The number of "rings" selected by the user (Int = [0,1,2,3])

Design Considerations

Preference is for a solution where the presentation logic is done on the client to minimise server load.

We would prefer Google maps be used as the visualisation tool, but are open to other providers if Google is not able to support requirement.

Data

Data is available for the suburbs and surrounding rings specifically the following "tables" have been generated:

Formerly  **LocalDirectories** Since 1994

- Suburbs By name, postcode and state
- Ring 1 - The set of additional suburbs that share a border with the initial suburb selection
- Ring 2 - The set of additional suburbs that share a border with any suburb in the 1st ring but are not the initial suburb or already included in the 1st ring.
- Ring 3 - The set of additional suburbs that share a border with any suburb in the 2st ring but are not the initial suburb or already included in the 1st or 2nd ring.

We shall expose the suburb information via an ElasticSearch (will return JSON based on query) as this will eliminate the need for a backend component and make the widget more portable.

General Constraints

Use of frameworks should be limited, Bootstrap and JQuery can be assumed to be present.

Where possible the code should be delivered as 3 files:

- 1 * JavaScript File
- 1 * CSS File
- 1 * HTML file containing example of widget use

The widget will be used in an AngularJS context, however this cannot be assumed.

Architectural Strategies & Requirements

Source Control

A Git repo will be set up and access extended to external developers

Development Environment

Any Grunt or other build scripts required for the project must be included in the Git repo, no commercial third party build tools to be integrated into the project without approval

Testing

Testing will be performed in current versions of IE, Chrome, and Safari on desktop and Chrome/Safari on iPad

Logging

No logging is required

Security

The widget must function in an SSL environment.

System Architecture

N/a