# Thomas Parker: n8584095 Team 21

## Item 1: Search page

In this iteration the search page was overhauled to make it easier to use and have flexibility in how you search. The field list for searching went from an address in the search bar with a minimum and maximum buying price to minimum and maximums for bedroom, bathroom, and car parks as well as a dynamically generated list for suburb populated by the database. The CSS also underwent a change for this as well to make the presentation of houses better on the page and in the end we added functionality to limit the number of results per page.

This item includes database functions from the db.php file as well as the stored procedure file included with submission.

## Item 2: Wholehouse results

The result page for whole houses contains a picture gallery carousel a details section about the house and a map displaying the house and points of interest around it. This item focuses on the implementation of multiple image retrieval and display through the multilingual requirements to get them to show up as they are now, html and css for display javascript for carousel and php and sql for data retrieval. Another aspect of this page is the integration of owner details pulled dynamically from the database dependant on who the owner is of the property being displayed.

## Item 3: Map javascript

This item covers the Google Map placed in the results page and its javascript code. This map is drawn by the GeneralMap function and requires two variables its position is dictated by the MapArea id in the php page. The Variables passed to it are the main Address of the house being shown and an array of points of interest specifically a list of their addresses. These two variables are processed by the General Map function and the Google Geocode function to find the latitude and longitude of the addresses and display them on the map. All of the Map data is specific to the house it is referencing.

## Item 4: Views

This item focuses on the numerous views written to accommodate a number of different functionalities provided to the site throughout the iteration including updating a few existing ones to make them more versatile. These include the retrieval and concatenation of an address via property id and via sharehouseid as well as the retrieval of all relevant points of interest based on property id or sharehouseid.

## Item 5: Db functions

The database functions written for this iteration by me were: db\_getsuburbs, the rewrite of db\_search\_w\_house, db\_getAddress, db\_getSAddress, db\_getSPOI, db\_getPOI as well as minor alterations to the db\_s\_House\_addProperty function to add owner id.

The get suburbs functions is used to retrieve a list of suburbs on the server for use in the search procedure that way if a new suburb is added it is instantly available for use in the search page.

The search w house function needed a rewrite to accommodate the extra search fields being added and required a modified SQL approach as well to take into account the new variables being passed to it.

The db get address and s address use their respective id’s to find the address of the property and pass it back as a formatted string.

The get poi functions are used to retrieve the list of poi near the property and to format them into addresses used in geocoding.