# Sidekick Specifications (V1.0)

## Sidekick Main Page

## Screen Logic

1. User will click on Sidekick menu option
2. Code will look up default user preferences to determine status of ‘category flags’
   1. Fields – SG Show lifestyle on sidekick, show shopping on sidekick, show food on sidekick, show services on sidekick, show only my data and friends on sidekick
   2. These fields are yes no and correspond to the five icons on top left of the screen
3. Code will build the drop down list of ‘show my’ by reading the location lists table
   1. Build an array contain lists by ‘select from location\_list\_table when SG User ID = ‘%userid’
4. Code will build default array of locations based on four categories
   1. Determine the default location of user by reading SG Locale from User Table where SG User ID = ‘%userid’
   2. Determine the central location by reverse geocoding the SG Locale (see <https://developers.google.com/maps/documentation/geocoding/>)
   3. This will also give you the bounds (top left, bottom right) of the map
   4. Now you will need to read through the SG location table looking for any record which is within the map
      1. I think you will need to use the latitude and longtitude in that table (within bounds) to get a record
      2. So it will be something like
         1. Select Name, Latitude, longtitude, major category from SG\_Location where SG\_location\_latitude within range (ie number is higher than boundary) and SG\_location\_longitude within range
      3. This will get the list of locations on Snapgadget that are on the map
   5. Now you will need to decide if you want to show them on the map based on the category flags
      1. Loop through the records returned from SG\_location and compare them to category
      2. Note that you don’t want to discard the list, just set each record display or not display
      3. So if SG\_Show\_Lifestyle is set to yes and the record returned from SG\_Location has a major category of lifestyle then you will set that record to be displayable
         1. You will repeat this for all four categories
      4. Finally you will also check if the show only my and my friends is set to yes
      5. If it is set to ‘no’ then you can display all records that are displayable
      6. If it is set to ‘yes’ then you need to further filter the display
      7. You will then need to get the list of friends for the user by
      8. Select array of friends from friends table where sg\_user\_id = %user\_id
      9. Now you will loop through the review table to see if any of the locations have been reviewed either by the user or by the users friends
      10. If they have then they remain displayable, otherwise they are not displayable
   6. Then you will finally display on the map all locations that are displayable with the appropriate icon for each category
   7. Each icon will be clickable and will enable you to display reviews for that location
      1. So when it is clicked you may need to read review table matched by that location id
   8. You will also display the list of displayable items sorted into categories on left panel
5. If user changes any of the icons (ie toggles on/off shopping you will need to redisplay map based on new criteria
6. If user moves the map through scrolling you will need to calculate the new center (using maps api) and then repeat the logic to redraw new icons and display record
7. If a user clicks on a location on the map or on the name on the left hand side then it will go to the sidekick drill down page
8. Reviewer of the day logic
   1. TBD - But basically we will create a reviewer of the day for each zip code which can then be displayed
      1. This may be a new table containing Zip Code, SG User ID
      2. You will then read the table based on the zip code of the map, which gets the SG User ID
      3. You read the user table to get their picture
      4. You then read the review table to get the number of reviews they have done
      5. You read the User Status table to get number of kudos points
      6. You can then display the reviewer of the day details
   2. Clicking on reviewer of the day
      1. Sort the reviewer of the day reviews by newest first and display the top five as per the screen. If they have more reviews then prompt for ‘read more’
         1. For consistency/UI – read more could be replaced by ‘>’ (right scroll)
9. Curious about a new location
   1. This should work the same way as google maps search. You can key in a zip code or enter a location/name ([https://developers.google.com/maps/documentation/javascript/places#places\_autocomplete](https://developers.google.com/maps/documentation/javascript/places" \l "places_autocomplete))
10. Personalize your map (and right click on the map
    1. This allows you to create a new location on the map, either by dropping a pin on the map or entering a location name (that google finds) [Same as ‘curious about a location’]
    2. New location will have the following popup
       1. You will use the name entered in the ‘personalize your map’ or get the pin location coordinates.
       2. You will prompt the user for ‘category’ (Major and Minor)
       3. You will then write a record to the location table
       4. WOEID –
          1. <http://www.ozzu.com/programming-forum/using-yahoo-geolocation-api-find-woeid-t107287.html>
       5. Google maps CID
          1. Check maps doc, but basically what you get from ‘clicking on map’ is the CID
       6. Foursquare venue ID
          1. <https://developer.foursquare.com/docs/venues/search.html> (pass in lat/lon)
          2. <https://developer.foursquare.com/docs/responses/venue.html> (look through response for ‘name’
       7. Yelp ID
          1. http://www.yelp.com/developers/documentation/v2/search\_api#searchGC
       8. Latitude, longtitude, name, full address, phone
          1. From google API
    3. Check out wildswim.com for an example of how you can add to map

## Sidekick Drill Down

This page is called when a user clicks on the ‘map’ icon or the name on the category list of sidekick page

The page will already have the location ID

1. Use the location ID, lat/lon and or google CID to retrieve the picture and map
   1. [https://developers.google.com/maps/documentation/staticmaps/#quick\_example](https://developers.google.com/maps/documentation/staticmaps/" \l "quick_example)
   2. <https://developers.google.com/maps/documentation/imageapis/>
2. The Name, address, and phone# are stored in the location table
3. There are two types of reviews (reviews that friends have done, reviews by other snapgadget users
   1. First read the reviews table using location as the key
   2. This will give you one or more records for that location and for each review the user id of the person who did the review.
   3. Now go check the SG User ID friends table to see how many of those reviews have been done by friends
   4. (Basically get the list of friends ids and match them against the review user ids)
   5. You can now calculate the total# of reviews (records returned where location id matches, then subtract the number of those that were done by friends.
   6. Then you can build the (count) slider for reviews by friends and reviews by snapgadget users
4. Popular count(postitnote) is derived from looking through the location lists tables for any lists that contain the location list id that matches the location
   1. If a location is mentioned in a list then count it, you will total all the mentions to derive the value to be displayed
5. Passionate about(Heart)is based on the following
   1. Select count(\*) from passionate\_lists\_table where SG\_passionate\_lists\_table,sg\_location\_list\_id = %location\_id (passed into the page)
6. Review It button takes you to the ‘review page’
   1. Note that if the user has already reviewed then this would be an update rather than create
   2. The review page basically creates or updates all the fields in the review table
7. Friends reviews
   1. Display the latest two reviews done by friends (using the same records retrieved from step 3 above).
      1. Latest can be determined as reviews have a create and update date, so sort them by newest first and display first two
8. Snapgadget reviews
   1. Works the same as friend reviews, just you are selecting reviews that are not on the friends list
9. Other reviews
   1. Lookup the yelp ID from the location table then use this to call the yelp API to display the yelp details (<http://www.yelp.com/developers/documentation/v2/business>)
      1. Review count, rating, reviews.excerpt
   2. Ignore trip advisor for the moment
10. Share Icon (see 2.4 of draft screens)
    1. Clicking on this will launch the share page (passing in location id, name)