Hemkraft

Project Overview

The purpose of this project is to analyze, specify, design, implement, document, and demonstrate an app your team develops. The project follows the three phases outlined in the Database Application Development Methodology: Analysis & Specification, Design, and Implementation. The implementation must use a fully relational, distinct Database Management System (DBMS) that utilizes tables and supports standard SQL queries. Embedded databases (such as SQLite) may not be used. Tools that generate SQL, map programming objects to database objects, or non-tabular systems that utilize SQL as a data manipulation language are also not permitted. Additional details regarding acceptable project stacks will be communicated in the assignment directions for the project.

The due dates and deliverables that must be submitted for each phase will be provided via Canvas.

When reading through this project description, please make the following assumptions: unless otherwise specified as optional, all attributes are required; unless otherwise specified, if given a list of potential values, choices should be limited to that list; that you should create normalized schemas, and minimize the use of NULL attributes whenever and wherever possible; ensure that you store non-numeric data that appear as numbers (such as street numbers, phone numbers, postal codes, etc.) as strings and not numeric data types; if a sort order is not specified as ascending or descending, then ascending order is implied; and avoid "catch-all" forms with unnecessary inputs that the user would leave empty or NULL. In addition, you do not need to be concerned about handling concurrent operations that could conflict and introduce inconsistencies in your database.

This spec contains a functional description along with some screen mockups. *The user interfaces depicted in this project description are examples to guide your thinking and are not intended to capture all functionality as described.* Your project's interface may look completely different and that is fine—even encouraged! For example, you might choose to split up some interfaces we have shown on a single screen into multiple screens. You might choose to use popup windows instead of refreshing the page. A complete reorganization of the user interface is acceptable providing your application supports the functionality as described. You may implement the project as a traditional standalone application (e.g., Java GUIs or Python's TkInter) or as a web application (e.g., web scripting languages like PHP or JSP). Your project will not be graded on its aesthetic appeal, but on its functionality.

Do not create any additional functionality that is not mentioned in this specification (such as email notifications, etc.) or attempt to enhance your final product beyond what the specification requires. Adding unwanted functionality can and will impact your grade!

Overview

Hemkraft is a nonprofit organization (founded by eccentric billionaire and Georgia Tech alumnus George P. Burdell) that wishes to track the various characteristics of households, such as size, type, etc., along with the details of certain types of appliances used within those households. The data collected will be "open" in the sense that anyone will be able to submit their data, conversely, anyone can browse the selected set of reports available on the Hemkraft website.

Definitions

This section provides some basic details about the things and their properties that the system tracks. Note that some operational attributes may not be mentioned as they might not be an essential property of something. Pay close attention to the functionality section as it may imply operational attributes that may need to be included in your design.

At the core of Hemkraft are ***households***. The following details will be collected for a household:

- Email
  - The email address provided will be used to identify the household.
- Phone number (will be used in the future by Hemkraft staff to contact households)
  - The submitter may optionally provide their ten-digit phone number.
  - The area code (first three digits) must be stored separately from the remaining seven digits. Phone numbers may be entered with or without dashes but should have any dashes removed prior to storing them in the database.
  - The phone number type. This could be one of the following: home, mobile, work, or other.
  - Phone numbers must be unique per household.
- The square footage of the household, as a whole number
- The number of occupants (adults and any children) of the household
- The number of bedrooms in the household. Households can have none, one, or many bedrooms.
- The household type: House, apartment, townhome, condominium, or mobile home.
- Detailed information regarding each bathroom in the household. Households must have at least one bathroom.
  - Each bathroom will be identified by the household it is associated with and the order in which it is entered into the system (the first bathroom would be identified as "1", the second as "2", etc.).
  - The number of sinks, commodes, and bidets within the bathroom should be provided.
  - A bathroom can be one of two types: a half bathroom or a full bathroom.
    - A half bathroom will have at least one sink, commode, and/or bidet, and the count for each type of fixture will be tracked. (Many combinations are possible, as an example, a half bath might have only a sink, just a commode, or a commode and a sink.)
      - A half bathroom may be optionally given a non-unique name, such as "basement" or "powder room"
    - If a bathroom is a full bathroom, in addition to sink, commode, and bidet counts, the following details must also be collected:
      - Bathtub count (a bathtub with only a faucet)
      - Shower count (a standalone shower)
      - Tub/shower count (a bathtub with both a faucet and a shower)
      - If the bathroom is the primary bathroom (a bathroom attached to the primary bedroom. A household may only have one primary bathroom, no primary bathroom is also possible)
      - A full bathroom must have at least one bathtub, shower, or tub/shower (but may have more than one).
- Postal code where the home is located (with matching city/state and geolocation)
  - A listing of postal codes with city, state, and their central latitude & longitude will be provided and should be used to validate user input. Entering an invalid postal code should be rejected.
  - This listing of postal codes will not change, and you do not need to worry about updating it.
  - A short sample listing of postal codes is below.

Postal Code	City	State	Latitude	Longitude
55302	Annandale	MN	45.246631	-94.11692
20227	Washington	DC	38.893311	-77.014647
14043	Depew	NY	42.904958	-78.7006
15278	Pittsburgh	PA	40.434436	-80.024817
24062	Blacksburg	VA	37.174227	-80.395698

02472	Watertown	MA	42.371296	-71.18196
07309	Jersey City	NJ	40.73276	-74.075485
00623	Cabo Rojo	PR	18.08643	-67.15222
53031	Hingham	WI	43.639395	-87.915705
60651	Chicago	IL	41.901485	-87.74055
99671	Stebbins	AK	63.511893	-162.27463
97304	Salem	OR	44.970181	-123.08033
81624	Collbran	CO	39.220166	-107.93414
82718	Gillette	WY	43.939968	-105.52445
89317	Lund	NV	38.835421	-115.02628
92232	Calexico	CA	33.026203	-115.28458
30334	Atlanta	GA	33.702657	-84.439127
36879	Waverly	AL	32.733511	-85.55322

Along with the household details, the information regarding *appliances* in the household must also be submitted. Every appliance will be identified by the household it is associated with along with the order it was entered into the system (the first appliance would be identified as "1", the second as "2", etc.). All appliances will be associated with a particular manufacturer, which will be chosen from a populated, updatable list retrieved from the database. The submitter may also optionally provide the appliance's model name.

There are multiple appliance types (and sub-types) which have certain properties which must be tracked for every instance of that appliance. These appliance types and specific properties are as follows:

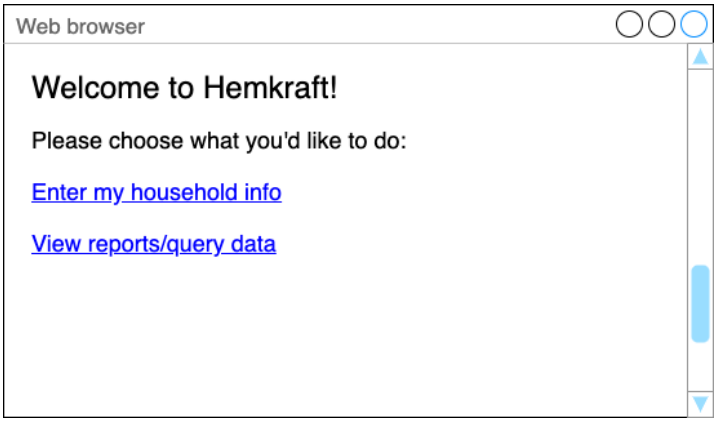
- Refrigerator/freezer
  - The type of refrigerator/freezer: Bottom freezer refrigerator, French door refrigerator, side-by-side refrigerator, top freezer refrigerator, chest freezer, or upright freezer.
- Cooker
  - Cookers may be a combination of the following:
    - Oven
      - An oven can have one or more of the following heat sources: gas, electric, and/or microwave. (A microwave heat source is different from a standalone microwave oven).
      - The oven type: either convection, or conventional.
    - Cooktop
      - A cooktop can have only one of the following heat sources: gas, electric, radiant electric, or induction.
  - If a cooker has both an oven and a cooktop, it is possible for both to have different heat sources.
- Washer
  - The loading type: either top or front.
- Dryer
  - The heat source for the dryer: gas, electric, or none (for dryers which do not utilize heat, such as a condensing dryer).
- TV
  - The display type: tube, DLP, plasma, LCD, or LED.
  - The display size, in fractional inches (to the tenth of an inch).
  - The maximum resolution: 480i, 576i, 720p, 1080i, 1080p, 1440p, 2160p (4K), or 4320p (8K).

Functionality

The user interface for the application should be consistent, with all inputs appropriately labeled, and with appropriate input controls used (such as a textbox for text, drop-down for a single choice list, spinner for whole numbers, etc.). Any instances where a NULL value might be returned should be replaced with an empty string.

Main menu

When initially accessing the application, a menu with two choices should be displayed: either enter household information, or view reports.

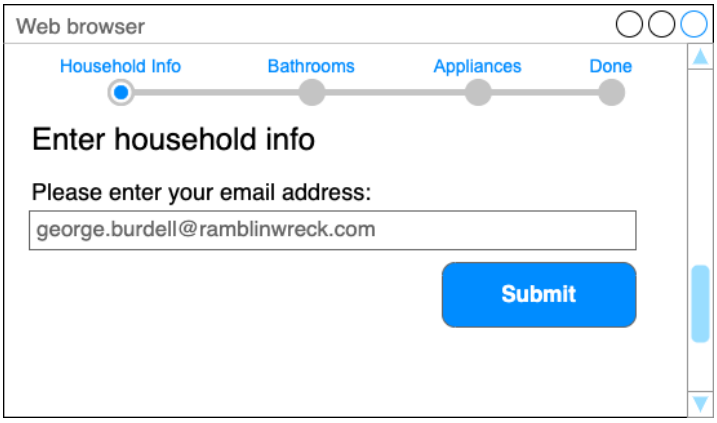


Household Information

When the "Enter my household info" option is chosen, the user will be provided with the following interface to enter their household information into the system. Note that it is not necessary to provide the user with an option to go back and/or be able to change data they have previously entered.

Get email address

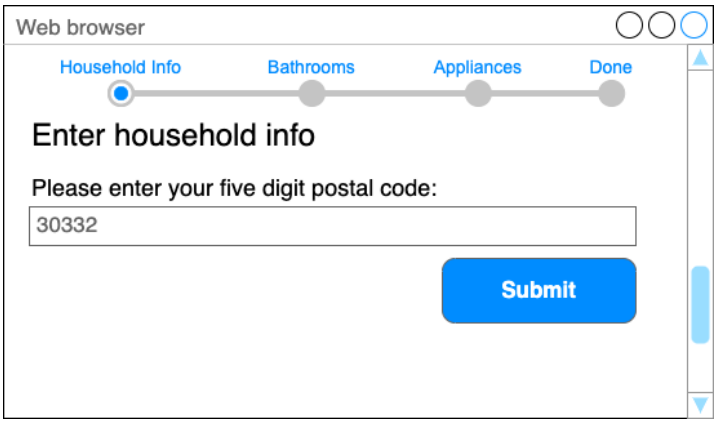
First, the email address of the user is collected. If a household entry already exists in the database for the email address, an appropriate error message should be displayed. Otherwise, the user is taken to the postal code entry screen.

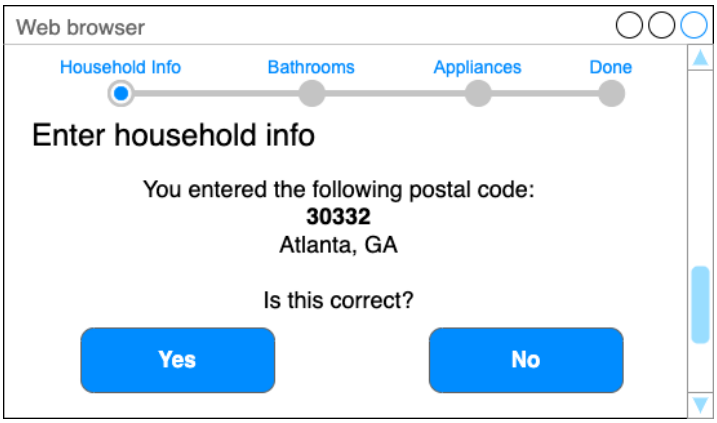


Postal code entry

On this screen, the user is prompted to enter the five-digit postal code for their household. If the postal code entered does not match one listed in the database, an appropriate error message should be displayed. Otherwise, the city and state for the postal code are displayed, with the user asked to confirm. If the user answers no to the confirmation

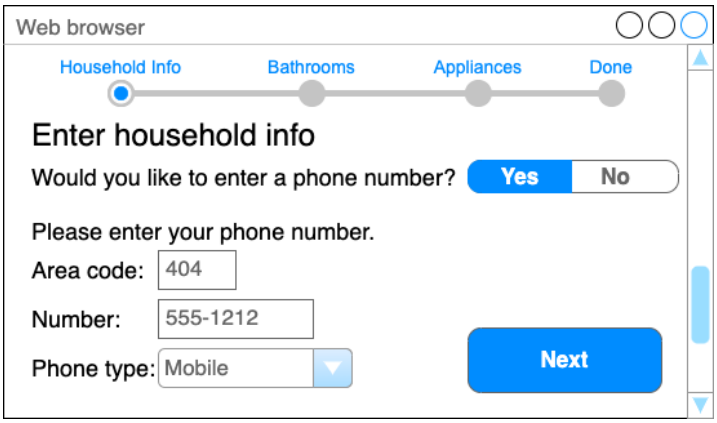
prompt, they should be prompted to re-enter their postal code.





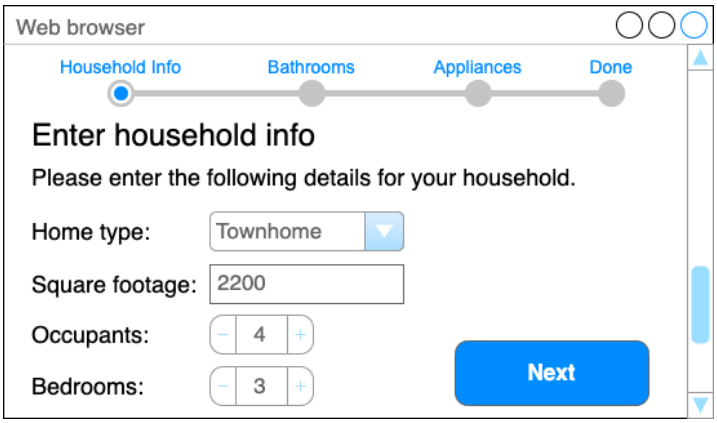
Phone number entry

Next, the user is asked if they wish to enter a phone number. If they choose yes, then entry boxes for the area code (three digits) and phone number (seven digits) are provided, along with a dropdown to choose the phone choice. If the phone number already exists in the database, an appropriate error message should be displayed. Otherwise, the phone number is saved. If the user does not wish to enter a phone number, they may navigate to the next screen.



Household info

On this page the various properties of the household are captured: type, occupants, square footage, and bedroom count. Once the user saves the information on this page, they are taken to the next one.



Bathrooms

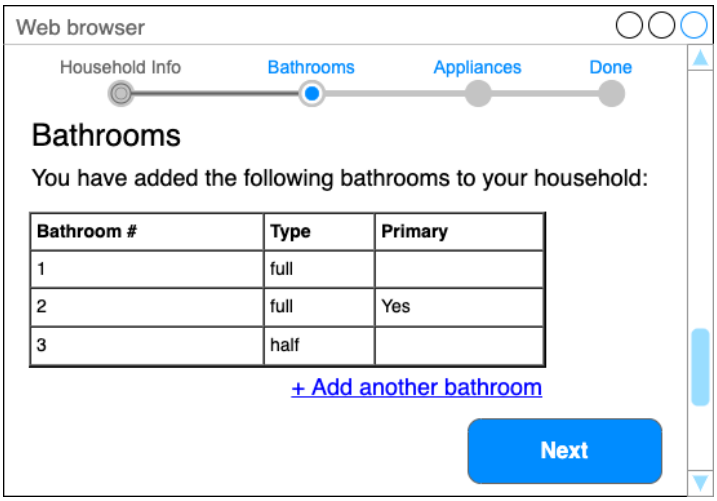
Since the user has not yet added a bathroom, they are next shown the “add bathroom” form.

Add bathroom

This form is used to add the details regarding the bathroom they would like to add. If the household already has a primary full bathroom, the option to set this bathroom as a primary should be disabled.

Bathroom listing

After adding a bathroom, the bathroom listing is shown, which lists the bathroom's number, type, and if it is a primary. Two options should be given on this screen: either to add another bathroom (returning the user to the add bathroom form) or to move on to the next screen.



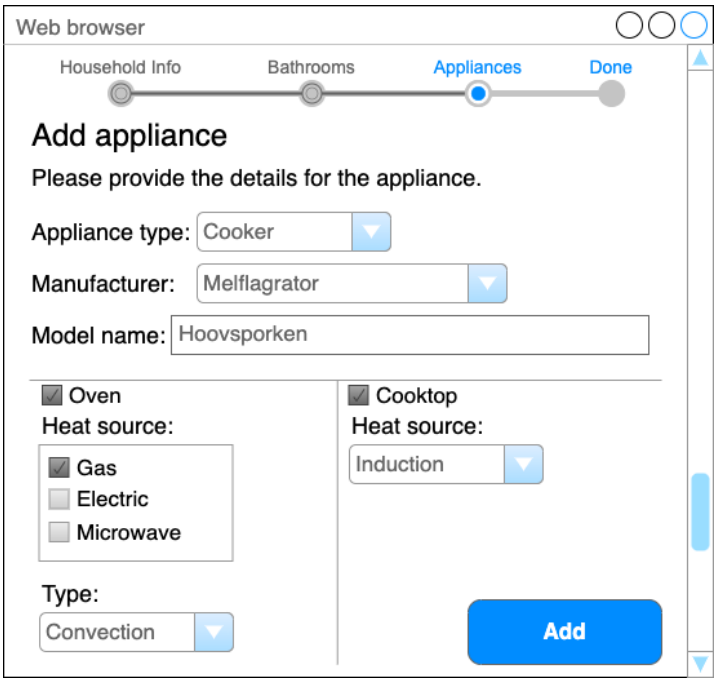
Appliances

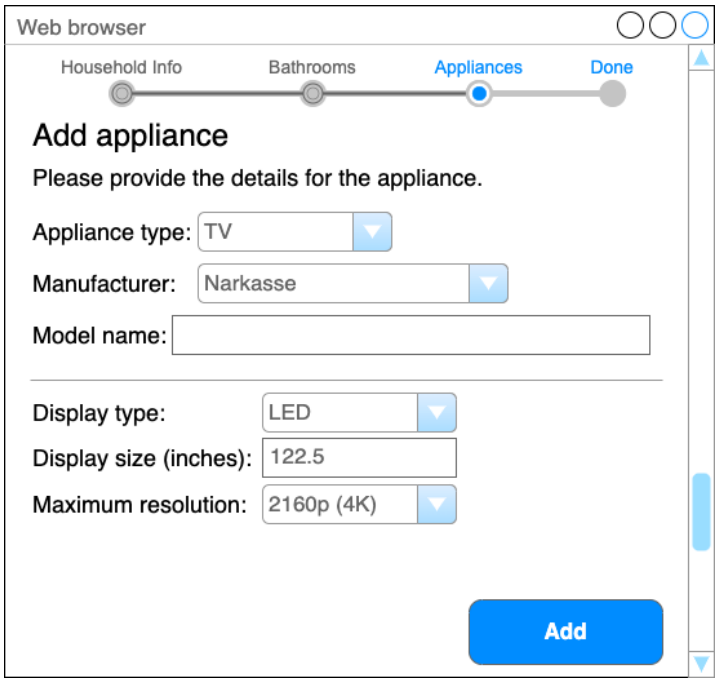
Since the user has not yet added an appliance, they should first be shown the "add appliance" form.

Add appliance

This form is used to add details for the appliance they would like to add. To optimize data entry, the appliance type should be chosen first, after which prompts for manufacturer, model name, and the information specific for that type

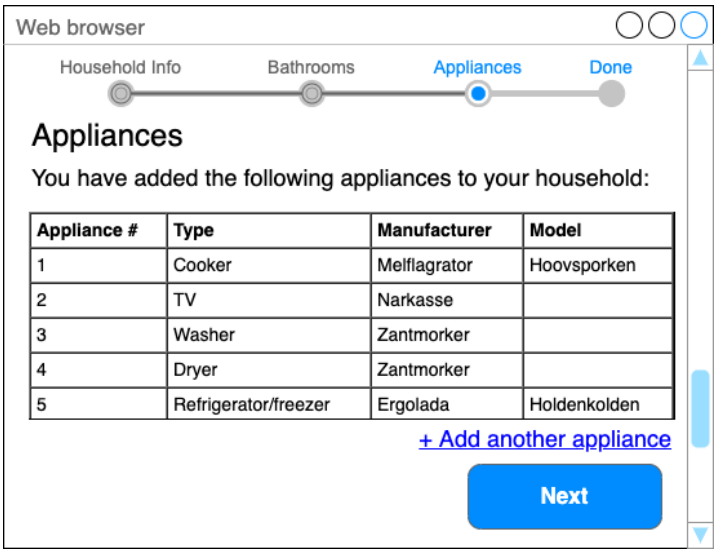
should be provided. In these examples, entry forms for a cooker and for a TV are given.





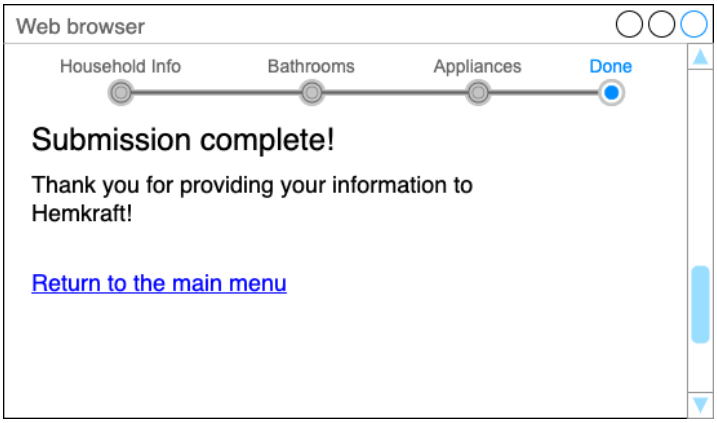
Appliance listing

After adding an appliance, the appliance listing is shown, which will list each appliance's number, type, manufacturer, and model name. Two options should be given on this screen: either to add another appliance (returning the user to the add appliance form) or to finish adding appliances and move on to the next screen.



Wrapping up

After the user has finished adding appliances and confirmed they would like to finish inputting the data, a thank you message should be displayed to them, with a link to the main menu provided.



Reports

When the user chooses the "View reports" menu option, they will be displayed a list of choices with a link to each report listed in this section. If a report does not require any parameters, it should be displayed immediately, otherwise, appropriate inputs for the parameters should be displayed along with a "submit" button to execute the report with the parameters. Missing or incorrect parameters must be handled with an appropriate error message and should prevent the report from executing. Tables, when displayed, must have an appropriate header row.

Some report pages, as defined, can provide information utilizing a single database query and are expected to be written with a single query. In some cases, a report page will require information from multiple queries, in which case the use of multiple queries is acceptable. Whether a report page requires a single query or multiple will not be indicated here and must be determined from the description provided.

In the event a report does not return any results, an appropriate message should be displayed instead of a blank page or an empty table. Any instances where a NULL value might be returned should be replaced with an empty string, unless otherwise specified in the report definition.

No UI mockups are provided for this section.

Top 25 popular manufacturers

This report will list the top twenty-five manufacturers with the most appliances in the database. A column for each manufacturer will be displayed, and a column for the raw count of appliances for that manufacturer (as an integer), ordered by count descending.

In addition, a drilldown report by manufacturer must be provided, which can be accessed with an appropriate mechanism, such as a button or link. The drilldown report for the manufacturer will list the manufacturer name at the top, and then a table with the following details, a column for each appliance type, and a column with the raw count of appliances of that type for that manufacturer (as an integer).

Manufacturer/model search

This report will allow the user to enter any string and return a list of distinct results where the entered string matches (with case insensitivity) any part of a manufacturer name or model name. For example, a search for "tech" would match on the manufacturer names "Technopure", "Valutech" and all models from those manufacturers, or the model name "Adtechnet" and its manufacturer. Columns for the manufacturer name and the model name must be displayed, ordered by manufacturer name ascending and model name ascending. In addition, the cell with the manufacturer name,

model name (or both) that matched the search string must be highlighted with a light green background to indicate to the user which field matched their search term.

Average TV display size by state

This report will list the average TV display size per state. A column should be provided for the state and a column listing the average display size (as a decimal number rounded up to the tenths decimal point), ordered by state ascending.

In addition, a drilldown report by state must be provided, which can be accessed with an appropriate mechanism, such as a button or link. The drilldown will show for the state, a column for screen type, followed by maximum resolution, grouped by those values, and then a column with the average display size (as a decimal number rounded up to the tenths decimal point). The rows should be ordered by average screen size in descending order.

Extra fridge/freezer report

At the top of this report, a count of all households with more than one fridge or freezer should be displayed (as a whole number).
Underneath the count, a tabular listing will show the top ten states with a column for the state and a column for the count of households with multiple fridge/freezers in that state, sorted by household count descending. Additional columns after the household count will show the following values: the percentage of households with multiple fridge/freezers in that state with chest freezers, the percentage of households with multiple fridge/freezers in that state with an upright freezer, and the percentage of households with multiple fridge/freezers in that state with something else. (All percentages should be displayed as whole integers and may add up to more than 100%.)

Laundry center report

The first table on this report will display the most common washer type and dryer heat source by state. A column will list the state, followed by a column to display the washer type, and another column will display the heat source. For example, a row might show the values “AK”, “Front” and “Gas” for the respective column values. The rows should be ordered by state ascending.

Another table will display the household count (as an integer), per state, where a household has a washing machine, but does not have a dryer. The first column should list the state, the second the household count, and the rows should be ordered by household count descending.

Bathroom statistics

This report will not show tabular data, but instead, display various pieces of information for all data available. All data should be labeled appropriately.

- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of all bathrooms per household
- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of half bathrooms per household
- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of full bathrooms per household
- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of commodes per household
- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of sinks per household
- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of bidets per household

- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of bathtubs per household
- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of showers per household
- The minimum (as an integer), average (as a decimal number rounded up to the tenths decimal point), and maximum (as an integer) count of tub/showers per household
- Which state has the most bidets (count of all bidets as an integer), and how many
- Which postal code has the most bidets (count of all bidets as an integer), and how many
- How many households (count as an integer), have only a single, primary bathroom, and no other bathrooms

Household averages by radius

It is believed that users will be interested to know the household statistics within a certain distance of a postal code. This report will require two user inputs: the postal code to center the search on, and the search radius (a whole number, with the following choices available: 0, 5, 10, 25, 50, 100, and 250). The postal code input should be validated, and if invalid, an appropriate error message displayed. The search radius would include any postal codes whose distance is less than or equal to the distance input. For example, if the user chooses 5, then anything within <=5.0 miles would be returned. A search radius of zero (0) is acceptable if the user wishes to search only within a postal code. The search result should include the postal code, the search radius, and per household, the average bathroom count (expressed as a decimal number to the tenths decimal point), the average bedroom count (expressed as a decimal number rounded up to the tenths decimal point), the average occupant count (rounded up to the nearest integer), the ratio of commodes to occupants (expressed as 1 commode per occupant, with the occupant value as a decimal number rounded up to the nearest hundredth - for example, a house with 2 commodes and 4 occupants would be a ratio of 1:2, while 3 commodes and 5 would be 1:1.67, etc.), and the average number of appliances (expressed as a decimal number rounded up to the tenths decimal point), and the most common heat source (for all appliances with a heat source) within the search radius.

To determine the distances between postal codes, the *haversine* formula can be used to calculate the straight-line distance between two points. The formula is provided below (additional details can be found at http://www.movable-type.co.uk/scripts/latlong.html, along with examples for utilizing it in database queries). Note that latitude and longitude are expressed in degrees, which must be converted to radians for these calculations, the Earth's radius (approx. 3958.75 mi) should be used for *R*.

$$\Delta lat = lat2 - lat1$$
$$\Delta lon = lon2 - lon1$$
$$a = \sin^2(\frac{\Delta lat}{2}) + \cos(lat1) * \cos(lat2) * \sin^2(\frac{\Delta lon}{2})$$
$$c = 2 * \text{atan2}\left(\sqrt{a}, \sqrt{1-a}\right)$$
$$d = R * c$$

To better understand using complex functions such as this one within database queries, you should perform this calculation individually every time it is needed instead of using a built-in or custom function. The calculation should not be performed outside of the database.