Test Report

Chalmers On The Go – the Complete Chalmers Experience

This document describes the different user stories, their associated acceptance tests and the unit tests made when developing the ChalmersOnTheGo Androis application.

For ChalmersOnTheGo 1.0, Jelly Bean 4.1.

Fredrik Einarsson - Niklas Johansson - René Niendorf
Anders Nordin - Sofie Peters

Test Report

Table of Contents

1   User stories

Below you will find a complete list of user stories, used while developing the ChalmersOnTheGo application. Their presented order is not an internal prioritising between the stories. Please note that some user stories are generally formulated, for an example "I want to see all rooms". These are tested only with a limited amount of data, not the data for *all* the rooms there are in the Chalmers, due to time limitations regarding collection of data. As long as more data is added in the correct way described in the Developer Manual, those user stories are probable to be correct. Please also note that user stories marked with an asterisk (*) have not been implemented. For more information on these, see section 3.3. Design decisions in document Software Development Document.docx.

General
- As a user, I should be able to exit the application at any time from inside the application[1]
- As a user, I want to be able to reverse my actions with a back-button[2]

Map
- As a user, I should only be able to see and navigate inside the Chalmers area[3]
- As a user opening the application, I want to see a fixed view of the Chalmers area, which always has the same starting coordinates[4]
- As a user opening the application, I want to see my current position on the map[23]
- As a user, I want to be able to click on a marked location, generating a popup window informing me about the name and floor of the location[5]
- As a user, I want to be able to erase all marked locations on the map[24]

Navigation
- As a user having gotten a location marked on the map, I want to be able to get the shortest path from my current location to the wanted location, by clicking the locations information window[6]
- As a user, I want to be able to get the map centred at my current location[7]
- As a user navigating to a certain location, I want to know how long time it will take me to get there[29]
- As a user navigating to a certain location, I want to know the distance between my current location and the wanted location[30]
- As a user, I want to be able to search for the shortest path between two separate locations, without needing to be currently positioned on any of them[31]
- As a user, I want to be able to touch mark a location on the map, without needing to search for it[32]

Searching

- As a user typing in a search for some item, I want a dropdown menu to appear with word-completed suggestions[8]
- As a user searching and getting suggestions, I want to be able to click any suggestion and get the location I clicked marked on the map[9]
- As a user, I want to be able to search for building and get the closest entry to the building marked on the map[10]
- As a user, I want to be able to search for a building and get all the rooms in the building marked on the map[11]
- As a user, I should be able to search for different room types, and get all the rooms of the specific type as suggestions[12]

Layer function

- As a user, I want to be able to have a checkbox-regulated layer function where I can choose between location types[13]
- As a user checking any layer, I want to be able to see all the locations concerned by the specific layer[14]
- As a user, I want to have layers with computer rooms, lecture halls, group rooms, floors and pubs[15]

Different application modes

- As a user, I should be able to switch between night and day mode at any point or time in the application *
- As a user currently in night mode, I want to see the buildings coloured according to their associated section's colour *
- As a user currently in day mode, I want to see the buildings coloured neutrally *

Non-navigational features

- As a user, I want features not only concerned with the map and navigation
- As a user, I want to se how many steps I have taken[25]
- As a user, I want to be able to see when I have burnt enough calories walking, to drink either a glass of wine, beer, shot, or cider[26]
- As a user, I want to be able to notify the application that I have taken a drink so that the calorie burning start for a new drink[27]
- As a user, I want to be able to synchronise my Time Edit schedule with the map to see where my lectures are *

Design, appearance and user support

- As a user, I want a nice looking application icon[16]
- As a user, I want to be able to reach all the application's functions from a menu system similar to Google maps[17]
- As a user searching for items, I want to see an icon symbolising the type of room, building or pub for each suggestion I get[18]
- As a user checking the "pub layer", I want to be able to see the buildings where the pubs are painted with the colour of the respective pub's sections[19]
- As a user opening the application, I want to be prompted to activate the GPS if it is not already enabled[20]
- As a user, I want the GUI to have an appealing appearance[21]
- As a user, I want the application to perform wanted actions reasonably fast[22]
- As a user, I should be able to switch orientation mode between vertical and horizontal without loosing my choices[28]

2 Testing

In the texts below you will find a complete list acceptance tests and unit tests, as well as their associated user stories, ran while developing the application ChalmersOnTheGo.

2.1 Acceptance tests

All user stories have been manually acceptance tested. The user stories are tested mainly to pass, but also to fail, and the tests have reasonable branch coverage. The validity in the tests is assured by their tight connection to user stories. The tests are made to verify the implementation.

What is tested	Exit-functionality
How it is tested	From each view, the exit-function in the menu is tried out
Expected result	Exit should happen whenever the exit-button is pressed
Actual result	Expected result
Associated user story	As a user, I should be able to exit the application at any time from inside the application[1]

What is tested	Back/reverse-functionality
How it is tested	From each view, the back-function is tried.
Expected result	Previously taken step should reverse whenever the back-button is pressed.
Actual result	Expected result
Associated user story	As a user, I want to be able to reverse my actions with a back-button[2]

What is tested	Zoom-out limit
How it is tested	Repeatedly zoom out in map.
Expected result	The map should only show the Chalmers area, not more.
Actual result	Expected result
Associated user story	As a user, I should only be able to see and navigate inside the Chalmers area[3]

What is tested	Map boundaries
How it is tested	Repeatedly scrolling outside of the map.
Expected result	The map should get stuck on the boundaries.
Actual result	Expected result
Associated user story	As a user, I should only be able to see and navigate inside the Chalmers area[3]

What is tested	Navigation fail outside Chalmers
How it is tested	The user stands outside the map boundaries and tries to navigate with the application.
Expected result	The map should show fixed coordinates of the map; centre of campus
Actual result	Expected result
Associated user story	As a user, I should only be able to see and navigate inside the Chalmers area[3]

What is tested	Limited data on Chalmers area
How it is tested	Data, not in the database, is searched for
Expected result	No suggestions should present themselves
Actual result	Expected result

Associated user story	As a user, I should only be able to see and navigate inside the Chalmers area[3]

What is tested	Fixed starting coordinates and current position
How it is tested	Repeatedly opening the application and visual control.
Expected result	The map view should show the intended coordinates and position.
Actual result	Expected result
Associated user story	As a user opening the application, I want to see a fixed view of the Chalmers area, which always has the same starting coordinates[4]

What is tested	Information (and navigation) window
How it is tested	Arbitrary locations are searched and marked on the map. The marker is clicked.
Expected result	When clicking the marker, it should trigger the popup window containing room name, floor and a "Navigate to here"-text.
Actual result	Expected result
Associated user story	As a user, I want to be able to click on a marked location, generating a popup window informing me about the name and floor of the location[5]

What is tested	(Information and) navigation window
How it is tested	Arbitrary locations are searched and marked on the map. The marker is clicked. The information popup window is clicked.
Expected result	When clicking the popup window, the shortest path from the user's current location to the marked location should be drawn on the map.
Actual result	Expected result
Associated user story	As a user having gotten a location marked on the map, I want to be able to get the shortest path from my current location to the wanted location, by clicking the locations information window[6]

What is tested	Target function
How it is tested	The user moves to different places inside the map boundaries and presses the target button
Expected result	The map should centre on the user's position
Actual result	Expected result
Associated user story	As a user, I want to be able to get the map centered at my current location[7]

What is tested	Target function when opening application
How it is tested	The user stands inside the map boundaries and opens the application
Expected result	The map should centre on the user's position
Actual result	Expected result
Associated user story	As a user opening the application, I want to see my current position on the map[23]

What is tested	Search field
How it is tested	The search icon is clicked
Expected result	A writeable search field will show up with a marker in it
Actual result	Expected result

Associated user story	As a user typing in a search for some item, I want a dropdown menu to appear with word-completed suggestions[8]

What is tested	Suggestions in search field
How it is tested	Arbitrary letters are written in the field
Expected result	Word-completed suggestions should show in a dropdown menu
Actual result	Expected result
Associated user story	As a user typing in a search for some item, I want a dropdown menu to appear with word-completed suggestions[8]

What is tested	Coverage in search field suggestions
How it is tested	Suggestions are generated with arbitrary letters and the suggestions are checked manually against the database content
Expected result	The suggestions should represent all the correlating data in the database
Actual result	Expected result
Associated user story	As a user typing in a search for some item, I want a dropdown menu to appear with word-completed suggestions[8]

What is tested	Mark searched location
How it is tested	Arbitrary locations are sought and clicked on in suggestions menu
Expected result	Locations should be marked on the map
Actual result	Expected result
Associated user story	As a user searching and getting suggestions, I want to be able to click any suggestion and get the location I clicked marked on the map[9] As a user, I want to be able to search for a building and get all the rooms in the building marked on the map[11]

What is tested	Closest entry function
How it is tested	A search suggestion for a building will be clicked
Expected result	The closest entry should be marked on the map
Actual result	Expected result
Associated user story	As a user, I want to be able to search for building and get the closest entry to the building marked on the map[10]

What is tested	Generic rooms search
How it is tested	Each room type (group room, lecture hall, gym etc.) is searched for
Expected result	They should show up as suggestions
Actual result	Expected result
Associated user story	As a user, I should be able to search for different room types, and get all the rooms of the specific type as suggestions[12]

What is tested	Generic rooms search and navigation
How it is tested	Each room type (group room, lecture hall and computer room) is searched for, suggested and clicked
Expected result	All rooms of the specific type are marked on the map
Actual result	Expected result
Associated user story	As a user, I should be able to search for different room types, and get all the rooms of the specific type as suggestions[12]

What is tested	Layer menu
How it is tested	The layer-button is clicked
Expected result	The layer menu should appear
Actual result	Expected result
Associated user story	As a user, I want to be able to have a checkbox-regulated layer function where I can choose between location types[13]

What is tested	Layer menu checkbox alternatives
How it is tested	The layer menu is opened and visually controlled.
Expected result	Alternatives should be computer rooms, lecture halls, group rooms and pubs.
Actual result	Expected result
Associated user story	As a user, I want to have layers with computer rooms, lecture halls, group rooms, floors and pubs[15]

What is tested	Layer menu checkboxes
How it is tested	Arbitrary check-boxes are marked and the menu is closed
Expected result	The chosen location layers should show on the map
Actual result	Expected result
Associated user story	As a user checking any layer, I want to be able to see all the rooms concerned by the specific layer[14]

What is tested	Layer menu checkboxes
How it is tested	Arbitrary check-boxes are marked and the menu is closed. The menu is opened and the check-boxes are unmarked. Close.
Expected result	The unchosen location layers should disappear from the map.
Actual result	Expected result
Associated user story	As a user checking any layer, I want to be able to see all the rooms concerned by the specific layer[14]

What is tested	Layer with floor options
How it is tested	Arbitrary check-boxes are marked
Expected result	Floor options should turn up
Actual result	Expected result
Associated user story	As a user, I want to have layers with computer rooms, lecture halls, group rooms, floors and pubs[15]

What is tested	Layer with floor options
How it is tested	Arbitrary check-boxes are marked and floor options are marked
Expected result	The chosen locations and chosen floors should show in the map
Actual result	Expected result
Associated user story	As a user, I want to have layers with computer rooms, lecture halls, group rooms, floors and pubs[15]

What is tested	Attractiveness in application icon
How it is tested	Visual control and discussion among team members.
Expected result	Consensus in that it is appealing.

Actual result	Expected result
Associated user story	As a user, I want a nice looking application icon[16]

What is tested	Menu similarity to Google maps
How it is tested	The application's menu system is compared to that of Google maps'
Expected result	The system should be quite similar
Actual result	Expected result
Associated user story	As a user, I want to be able to reach all the application's functions from a menu system similar to Google maps[17]

What is tested	Icons in search suggestions
How it is tested	All different search item types are tried (pubs, buildings, computer room, lecture hall, group room)
Expected result	The suggestions should show correlating icons to the items
Actual result	Not all types had their own icons
Fix	No. Too big re-implementation of databse. Left to show potential in application.
New result	-
Associated user story	As a user searching for items, I want to see an icon symbolising the type of room, building or pub for each suggestion I get[18]

What is tested	Building colours for pubs
How it is tested	The pub layer is marked
Expected result	The buildings with pubs should be painted in their section colour respectively
Actual result	Expected result
Associated user story	As a user checking the "pub layer", I want to be able to see the buildings where the pubs are painted with the colour of the respective pub's sections[19]
Note	Removed feature. Not considered as adding much value, instead cluttering map.

What is tested	GPS prompt
How it is tested	The GPS function is deactivated. Then the application is opened.
Expected result	A window should prompt the user to enable the GPS function.
Actual result	Expected result
Associated user story	As a user opening the application, I want to be prompted to activate the GPS if it is not already enabled[20]

What is tested	Attractiveness in GUI
How it is tested	Visual control and discussion among team members.
Expected result	Consensus in that it is appealing.
Actual result	Expected result
Associated user story	As a user, I want the GUI to have an appealing appearance[21]

What is tested	Application performance
How it is tested	Arbitrary functions are clicked and the time they take to react is measured with timer

Expected result	Each carried-out action should take at the most 1 second
Actual result	Expected result
Associated user story	As a user, I want the application to perform wanted actions reasonably fast[22]

What is tested	Empty map function
How it is tested	Arbitrary locations and layers are added to the map, and then the empty map-button clicked
Expected result	The map should be become clean
Actual result	Expected result
Associated user story	As a user, I want to be able to erase all marked locations on the map[24]

What is tested	Activation and deactivation of step counter
How it is tested	The StepCounter will be activated, arbitrary functions carried out, then the StepCounter will be deactivated
Expected result	The StepCounter should activate and deactivate when ordered to
Actual result	Expected result
Associated user story	As a user, I want to se how many steps I have taken[25]

What is tested	Activate/Deactivate message
How it is tested	The StepCounter will be activated, arbitrary functions carried out, then the StepCounter deactivation windows will be visually controlled
Expected result	The StepCounter option should show "Deactivate StepCounter"
Actual result	The StepCounter option shows "Activate StepCounter" even though it is activated if phone orientation is changed
Fix	Activate/Deactivate message saved as variable and fetched when orientation changed
New result	Expected result
Associated user story	As a user, I want to se how many steps I have taken[25]

What is tested	Counting steps
How it is tested	The StepCounter will be activated and the user will walk and manually count her steps, comparing them to those in the StepCounter.
Expected result	The number of steps manually counted and those counted by the StepCounter should correlate.
Actual result	Expected result
Associated user story	As a user, I want to se how many steps I have taken[25]

What is tested	Counting steps in sleep mode
How it is tested	The StepCounter will be activated and sleep mode will be engaged. The user will walk and manually count her steps, then waking up the application, comparing the counted steps to those in the StepCounter.
Expected result	The number of steps manually counted and those counted by the StepCounter should correlate.
Actual result	Expected result
Associated user story	As a user, I want to se how many steps I have taken[25]

What is tested	Counting steps when application is minimised

How it is tested	The StepCounter will be activated and the application will be minimised. The user will walk and manually count her steps, then opening the application, comparing the counted steps to those in the StepCounter.
Expected result	The number of steps manually counted and those counted by the StepCounter should correlate.
Actual result	Expected result
Associated user story	As a user, I want to se how many steps I have taken[25]

What is tested	Counted steps saved if StepCounter deactivated
How it is tested	The StepCounter will be activated and the user will walk. The number of steps will be controlled, then the StepCounter will be deactivated, then activated again.
Expected result	The number of steps in the StepCounter should still show on the calorie counter progress window.
Actual result	Expected result
Associated user story	As a user, I want to se how many steps I have taken[25]

What is tested	Stop counting steps
How it is tested	The StepCounter will be activated and the user will walk, then deactivate the StepCounter.
Expected result	The StepCounter should stop counting when being turned off.
Actual result	Expected result
Associated user story	As a user, I want to se how many steps I have taken[25]

What is tested	Not saving counted steps when exiting application
How it is tested	The StepCounter will be activated and the user will walk, then the application will be exited, and then opened again.
Expected result	The StepCounter should be restarted from 0.
Actual result	Expected result
Associated user story	As a user, I want to se how many steps I have taken[25]

What is tested	Calorie counting window
How it is tested	The calorie counting window is clicked in the menu.
Expected result	The wanted window should show up.
Actual result	Expected result
Associated user story	As a user, I want to be able to see when I have burnt enough calories walking, to drink either a glass of wine, beer, shot, or cider[26]

What is tested	Calorie counting progress bars existing
How it is tested	Visual control of calorie counting window.
Expected result	Progress bars for wine, beer, shots, water and cider should show in the windows.
Actual result	Expected result
Associated user story	As a user, I want to be able to see when I have burnt enough calories walking, to drink either a glass of wine, beer, shot, or cider[26]

What is tested	Calorie counting progress bars working 1
How it is tested	Visual control of progress bars while walking a few steps.

Expected result	Progress in all bars with taken number of steps.
Actual result	Expected result
Associated user story	As a user, I want to be able to see when I have burnt enough calories walking, to drink either a glass of wine, beer, shot, or cider[26]

What is tested	Calorie counting progress bars working 2
How it is tested	Visual control of progress bars while walking the desired amount of steps per each drink type.
Expected result	Progress in all bars and reaching desired amount of steps per drink, resulting in that, if the user takes a drink, no "warning"-message will appear.
Actual result	Expected result
Associated user story	As a user, I want to be able to see when I have burnt enough calories walking, to drink either a glass of wine, beer, shot, or cider[26]

What is tested	Drink! functionality
How it is tested	Clicking on each Drink!-button.
Expected result	Number of had drinks should show an increase accordingly, and the progress bars should start counting for a new drink.
Actual result	Expected result
Associated user story	As a user, I want to be able to see when I have burnt enough calories walking, to drink either a glass of wine, beer, shot, or cider[26] As a user, I want to be able to notify the application that I have taken a drink so that the calorie burning start for a new drink[27]

What is tested	Warning fat/pee/drunk/sick message
How it is tested	Clicking on Drink! without having reached desired amount of burnt calories.
Expected result	Number of had drinks should increase accordingly, and the progress bars should start counting for a new drink. Additionally, a message warning the user of becoming fat/drunk/sick or needing to pee appears.
Actual result	Expected result
Associated user story	As a user, I want to be able to notify the application that I have taken a drink so that the calorie burning start for a new drink[27]

What is tested	Saving progress when in sleep mode
How it is tested	The StepCounter will be activated and sleep mode will be engaged. The user will walk and manually count her steps, then waking up the application, comparing the counted steps to those in the calorie counting progress bars.
Expected result	The number of steps manually counted and those shown in the calorie counting progress bars should correlate.
Actual result	Expected result
Associated user story	As a user, I want to be able to notify the application that I have taken a drink so that the calorie burning start for a new drink[27]

What is tested	Saving progress when minimising the application
How it is tested	The StepCounter will be activated and the application will be

	minimised. The user will walk and manually count her steps, then waking up the application, comparing the counted steps to those in the calorie counting progress bars.
Expected result	The number of steps manually counted and those shown in the calorie counting progress bars should correlate.
Actual result	Expected result
Associated user story	As a user, I want to be able to notify the application that I have taken a drink so that the calorie burning start for a new drink[27]

What is tested	Not saving counted steps when exiting application
How it is tested	The StepCounter will be activated and the user will walk, then the application will be exited, and then opened again.
Expected result	The calorie counter progress bars should be restarted from 0.
Actual result	Expected result
Associated user story	As a user, I want to be able to notify the application that I have taken a drink so that the calorie burning start for a new drink[27]

What is tested	Switching orientation
How it is tested	Arbitrary functions will be activated, then the phone will be turned to switch modes.
Expected result	Any saved data should remain in every orientation mode.
Actual result	Expected result
Associated user story	As a user, I should be able to switch orientation mode between vertical and horizontal without loosing my choices[28]

What is tested	Duration and distance between locations
How it is tested	Arbitrary locations are searched for with the Search-button and marked on the map. The marked location is clicked showing the information and navigation window. The window is clicked.
Expected result	The time and distance from the user's current position will be shown
Actual result	Expected result
Associated user story	As a user navigating to a certain location, I want to know how long time it will take me to get there[29] As a user navigating to a certain location, I want to know the distance between my current location and the wanted location[30]

What is tested	Duration and distance between locations
How it is tested	Arbitrary locations are searched for with the Route-option and marked on the map. The marked location is clicked showing the information and navigation window. The window is clicked.
Expected result	The time and distance from the user's current position will be shown
Actual result	Time and distance set to 1 m and 1 minute always.
Fix	Easily seen to semantic mistake
New result	Expected result
Associated user story	As a user navigating to a certain location, I want to know how long time it will take me to get there[29] As a user navigating to a certain location, I want to know the distance between my current location and the wanted location[30]

What is tested	Path between separate locations
How it is tested	Two arbitrary locations are searched for and marked on the map.
Expected result	The path between them should be drawn.
Actual result	Expected result
Associated user story	As a user, I want to be able to search for the path between to separate location, without currently positioned on any of them[31]

What is tested	Location marking with touch
How it is tested	A finger is pressed and held against arbitrary places on the map
Expected result	The locations held against should be marked
Actual result	Expected result
Associated user story	As a user, I want to be able to touch mark a location on the map, without needing to search for it[32]

2.2 Unit tests

All database methods in the DAO (Data Access Object) class have been unit tested, using the public Assert class. The methods are tested both to pass and to fail and the tests have extensive statement coverage. These tests were made to verify the implementation.

- **Insertion and getting in table 4 (buildings table), test case**
    - insertIntoTable4 and getAllFromTable4 were tested together:
        - A building name (String) was inserted into table 4 via insertIntoTable4 and fetched with getAllFromTable4
- **Insertion and getting in table 2 (room types table), test case**
    - insertIntoTable2 and getAllFromTable2 were tested together:
        - Three room types (String) were inserted into table 2 via insertIntoTable2 and fetched with getAllFromTable2
- **Insertion and getting in table 1 (coordinates and buildings table), test suite**
    - insertIntoTable1 and getClosestEntry were tested together:
        - A pair of coordinates (Double) and a building name (String) were inserted into table 1 via insertIntoTable1.
        - The coordinates (Double) were used to create an object (LatLng) containing latitude and longitude.
        - The object (LatLng) and the building name (String) served as input in getClosestEntry.
        - The result of getClosestEntry (LatLng) and the object (LatLng) containing the coordinates were compared and found to be equal.
        - Calculating the closest entry
    - insertIntoTable1 and getClosestEntry were tested together:
        - An object (LatLng) containing zero coordinates, the current coordinates, were created.
        - Five different coordinate pairs (Double) and a building name (String) were inserted into table 1 via insertIntoTable1.
        - The pair of coordinates (Double) closest to the zero coordinates, were in addition used to create an object (LatLng) containing latitude and longitude.
        - The zero coordinates object (LatLng) and the building name (String) served as input in getClosestEntry.
        - The result of getClosestEntry (LatLng) and the closest coordinate pair object (LatLng) were compared and found to be equal.

- Insertion and getting in table 3 (room name, coordinates, room type, building and floor table)
  - o insertIntoTable3 and getRoomCoordinates were tested together:
    - A pair of coordinates (Double) were used to create an object (LatLng) containing latitude and longitude.
    - A room type (String) was inserted into table 2 via insertIntoTable2.
    - A building name (String) was inserted into table 4 via insertIntoTable4.
    - The room name (String), the coordinates (Double, the room type (String), the building name (String) and a floor (String) were inserted into table 3 via insertIntoTable3.
    - The room name (String) served as input in getRoomCoordinates.
    - The result of getRoomCoordinates (LatLng) and the object (LatLng) containing the coordinates were compared and found to be equal.
    - Getting all rooms in a specific building
  - o insertIntoTable2, insertIntoTable3, insertIntoTable4 and getAllRoomsInBuilding were tested together:
    - A room type (String) was inserted into table 2 via insertIntoTable2.
    - A real building name (String) was inserted into table 4 via insertIntoTable4
    - A false building name (String) was inserted the same way.
    - Room name1 (String), coordinate pair1 (Double), the room type (String), the true building name (String) and floor1 (String) were inserted into table 3 via insertIntoTable3.
    - Room name2 (String), coordinate pair1 (Double), the room type (String), the true building name (String) and floor2 (String) were inserted into table 3 via insertIntoTable3.
    - A false room name (String), coordinate pair1 (Double), the room type (String), the false building name (String) and floor1 (String) were inserted into table 3 via insertIntoTable3.
    - The true building name (String) served as input in getAllRoomsInBuilding.
    - The result of getAllRoomsInBuilding (ArrayList<String>) was tested using methods size and contains, and found to be satisfactory.
    - Getting all rooms with a specific type
  - o insertIntoTable2, insertIntoTable3, insertIntoTable4 and getAllRoomsInBuilding were tested together:
    - A room type (String) was inserted into table 2 via insertIntoTable2.
    - A false room type (String) was inserted the same way.
    - A building name (String) was inserted into table 4 via insertIntoTable4
    - Room name1 (String), a coordinate pair1 (Double), the true room type (String), the building name (String) and floor1 (String) were inserted into table 3 via insertIntoTable3.
    - Room name2 (String), coordinate pair1 (Double), the true room type (String), the building name (String) and floor2 (String) were inserted into table 3 via insertIntoTable3.
    - A false room name (String), coordinate pair1 (Double), the false room type (String), the building name (String) and floor1 (String) were inserted into table 3 via insertIntoTable3.
    - The building name (String) served as input in getAllRoomsInBuilding.
    - The result of getAllRoomsInBuilding (ArrayList<String>) was tested using methods size and contains, and found to be satisfactory.
    - Getting suggestions

- o insertIntoTable2, insertIntoTable3, insertIntoTable4 and suggestions were tested together:
    - A room type (String) was inserted into table 2 via insertIntoTable2.
    - A building name (String) was inserted into table 4 via insertIntoTable4
    - A room name (String), a coordinate pair (Double), the room type (String), the building name (String) and a floor (String) were inserted into table 3 via insertIntoTable3.
    - Different strings of letters matching the strings in table 3 served as input in suggestions.
    - The result of suggestions (ArrayList<String>) was tested using methods for size and null, and found to be satisfactory.
    - Getting room names
- o insertIntoTable3 and getName were tested together:
    - A room name (String), a coordinate pair (Double), a room type (String), a building name (String) and a floor (String) were inserted into table 3 via insertIntoTable3.
    - The room name (String) served as input in getType.
    - The result of getName (String) and the room name were compared and found to be equal.
    - Getting room types
- o insertIntoTable3 and getType were tested together:
    - A room name (String), a coordinate pair (Double), a room type (String), a building name (String) and a floor (String) were inserted into table 3 via insertIntoTable3.
    - The room name (String) served as input in getType.
    - The result of getType (String) and the room type were compared and found to be equal.
    - Getting floor
- o insertIntoTable3 and getFloor were tested together:
    - A room name (String), a coordinate pair (Double), a room type (String), a building name (String) and a floor (String) were inserted into table 3 via insertIntoTable3.
    - The room name (String) served as input in getFloor.
    - The result of getFloor (String) and the floor were compared and found to be equal.