User Guide to Vuzix Maps for Smart Glasses

Release 1.4.0

June 12th, 2015

Overview

The Maps application is a system M100 application designed to provide a means of navigation around the world. It can be controlled standalone by Voice or Buttons.

Dashboards

The application is divided into Dashboards to provide the most functionality while inhibiting space on the screen as little as possible. Each Dashboard displays icons that correlate to specific features.

There exists a Main Dashboard that provides a couple of small features; however, its main purpose is to delegate functionality to other Dashboards and other dialogs. Each side Dashboard will always redirect back to the Main Dashboard.

Map Operations Dashboard

The Map Operations Dashboard is unique as it overrides the usual expectation of the hardware keys on the M100. This Dashboard contains three Map Operations to modify the map:

- Scroll Left / Right
- Scroll Up / Down
- Zoom In / Out

When one of these three operations are selected, that specific operation is “locked”; a user will be able to identify that this operation has been locked when the icon's background changes color. The M100's hardware keys will then have been overridden for the time the operation is locked. The Previous and Next button will correspond to the action depicted by that operation, while the select button will “unlock” the icon to resume normal key operations.

For example, when the Scroll Up / Down operation is selected, the Previous button would then control scrolling down the screen, while the Next button would control scrolling up the screen. The hardware keys will continue to process this way until the user presses the Select button, upon which will “unlock” the operation.

Download Maps

The Maps application provides the ability to download areas of the world for offline usage. While disconnected from a Wi-Fi source, the Maps application would normally be useless; it requires an active connection to the Internet to process routes, navigate maps, etc. In order to still utilize all of the Maps' application's functionality, a user is able to download an area and still use the Maps application while within that area. For example, if a user has downloaded offline maps of US states California and Oregon, a route from San Diego to Portland can be created without any Internet connection.

Voice Recognition

Maps supports voice recognition for commands that affect the visual display of the map. To view

supported voice commands, simply speak *voice help* within the application while voice recognition is on. The user will notice on the left side of the Voice Help display all of the Map Operation voice commands while on the right side will be commands to modify the visuals of the map, i.e. map types, overlays, etc.

Some sample voice commands are: *Move North, Zoom out, View Satellite, View Daytime, Transit Overlay On*

Intents

One of the most powerful use cases of the Maps application are its intent receivers that allow it to receive intents and act upon the intent received. An intent can be sent via another application or even an application protocol used on the Internet (e.g. Telnet).

There currently exists two intent receivers for Maps:

- Mark Location Receiver
- Get Directions Receiver

The Mark Location Receiver takes a location and a couple of optional flags and processes it into a Location on the map and visually marks it. An example use case would be an application sending a String, “102 Pike St, Seattle, WA 98101”. The Maps application would register the intent and mark the location on the map and zoom into an appropriate field of view for the Location. A Location could be as specific as the address of the origin of a famous coffee company, or as general as the city of Boston, MA. The Maps application will process each and properly address the differences between the two.

It is important to note that there are two different methods to mark a location on the map.

A) Using a String as an address to find in the world

B) Using a raw latitude and longitude coordinate

To use (A), simply disregard the latitude and longitude values and identify the address as the required String extra. To use (B), simply specify the latitude and longitude values and disregard the String extra. If the user would like to use a raw coordinate to both mark the raw coordinate on the map and, in addition, also find the closest address to the raw coordinate, then the user can specify the latitude and longitude values while also using an empty string, i.e. <"">, as the String extra.

The specifics on the intent to send to the Maps application to Mark a Location are:

1. **Intent Action**: <com.vuzix.sg.maps.mark_location>
   a) *Required*
2. **Intent Extra**: <maps.action.location>
   a) *Required (1 of 2)*
   b) Type: String
   c) Description: the location as an address to mark on the map
3. **Intent Extra**: <maps.action.center_lat>
   a) *Required (2 of 2)*

b) Type: double

c) Description: the latitude value of the raw coordinate to mark on the map

4. **Intent Extra**: <maps.action.center_long>
   a) Required (2 of 2)
   b) Type: double
   c) Description: the longitude value of the raw coordinate to mark on the map
5. **Intent Extra**: <maps.action.marker>
   a) Optional
   b) Type: boolean
   c) Description: the optional flag on whether or not to visually mark the location on the map. Default is to visually mark it.
6. **Intent Extra**: <maps.action.zoom>
   a) Optional
   b) Type: integer
   c) Description: the integer to specify the zoom level to use when zooming into the location on the map. Recommended usage is to avoid overwriting this value; if overwriting the value, be sure to test the results.
      - Min Zoom Level: 0
      - Max Zoom Level: 20
7. **Intent Extra**: <maps.action.add_onto>
   a) Optional
   b) Type: boolean
   c) Description: the optional flag on whether or not to add onto the existing list of map markers (if there are any), or to overwrite the existing list with this intent's new values. Default is to overwrite.

The Get Directions Receiver takes a list of Strings and processes each String into a waypoint within a route. The list of Strings should be specific addresses and should be in the order the user would be traveling in, with the first String being the starting location and the last String being the destination. An example use case would be an application sending a list of Strings - (“3090 Orchard Park Rd, Buffalo, NY 14224”, “1047 Main St, Buffalo, NY 14209”). The Maps application would register the intent and create a route from the first location on Orchard Park Road in Buffalo to the second location on Main Street in Buffalo. The user could then opt to adjust the route settings or calculate the route based off of the settings saved within the Settings of the application. Maps will display an appropriate route after route settings have been confirmed.

The specifics on the intent to send to Maps to Get Directions are:

1. **Intent Action**: <com.vuzix.sg.maps.get_directions>
   a) Required

2. **Intent Extra**: <maps.action.waypoints>
    a) *Required*
    b) Type: String[] (List of Strings)
    c) Description: the list of Strings must have at least two Strings that can be processed into addresses.

It is also possible to send an intent via a JSON message using an application protocol like that of Telnet that utilizes the Internet to send text-oriented communications. An example of each intent being sent using Telnet on a Linux machine is described below:

Note: A Wi-Fi connection is required to send an intent using Telnet

Mark Location:

$ telnet <m100_ip_address> 960 ;

> {"MESSAGE_ID":"com.vuzix.sg.maps.mark_location","MESSAGE":{"maps.action.location":"13 Rue Bavastro, 06000 Nice, France","maps.action.zoom":5,"maps.action.marker":true,"maps.action.add_onto":true},"REQUEST_ID":"42"}

> {"MESSAGE_ID":"com.vuzix.sg.maps.mark_location","MESSAGE":{"maps.action.location":"", "maps.action.center_lat":43.092304,"maps.action.center_long":-77.679057},"REQUEST_ID":"42"}

Get Directions:

$ telnet <m100_ip_address> 960 ;

> {"MESSAGE_ID":"com.vuzix.sg.maps.get_directions","MESSAGE":{"maps.action.waypoints": ["3090 Orchard Park Rd, Buffalo, NY 14224","1047 Main St, Buffalo, NY 14209"]},"REQUEST_ID":"42"}

> {"MESSAGE_ID":"com.vuzix.sg.maps.get_directions","MESSAGE":{"maps.action.waypoints": ["","1047 Main St, Buffalo, NY 14209"]},"REQUEST_ID":"42"}