NSense

1.3

Deliverable 4.1 - NSense Software Suite

Technical Report COPE-SITI-TR-16-04

Generated by Doxygen 1.8.6

Fri Jun 3 2016 17:29:11

Contents

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Chapter 5

Namespace Documentation

5.1 Package com

Packages

- package meapsoft

5.2 Package com.meapsoft

Classes

- class FFT

5.2.1 Detailed Description

5.3 Package cs

Packages

- package nsense

5.4 Package cs.nsense

Packages

- package accelerometer
- package bluetooth
- package db
- package inferenceModule
- package location
- package microphone

Classes

- interface ChangeListener
- class CircleDisplay
- class NSenseActivity
- class NSenseHistoryActivity
- class NSenseHistoryAdapter
- class NSenseService
- class SensorProduct
- class SIAdapter

5.4.1 Detailed Description

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. This class provides an interface to update pipeline information and provides the information to NSenseService class, and NSenseActivity class.

Author

Saeik Firdose (COPELABS/ULHT),
Luis Lopes (COPELABS/ULHT),
Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for NSenseActivity class and provides custom-view for displaying the SocialInteraction and Propinquity level.

Author

Saeik Firdose (COPELABS/ULHT)
Philipp Jahoda

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. This class controls the User Interface and update it based on NSenseService results.

Author

Saeik Firdose (COPELABS/ULHT),
Luis Lopes (COPELABS/ULHT),
Waldir Moreira (COPELABS/ULHT),
Reddy Pallavali (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support to the NSense History Activity.

Author

Saeik Firdose (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support to the NSense History Activity, and provides the list of devices, SocialInteraction, and Propinquity information with the layout.

Author

Saeik Firdose (COPELABS/ULHT)
Luis Lopes (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It initialize and controls all the pipelines/modules, database and send the information to the NSenseActivity.

Author

Saeik Firdose (COPELABS/ULHT),
Luis Lopes (COPELABS/ULHT),
Waldir Moreira (COPELABS/ULHT),
Reddy Pallavali (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support to NSense service by providing the sensors to initiate for the each and every pipeline.

Author

Saeik Firdose (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for Inference module, NSense activity and provides the functionality to update the SocialityDetails.

Author

Saeik Firdose (COPELABS/ULHT)
Luis Lopes (COPELABS/ULHT)

5.5 Package cs.nsense.accelerometer

Classes

- interface AccelerometerListener
- class AccelerometerPipeline
- class AccelerSenManager
- class ActionsEntry
- class Globals
- class WekaClassifier

5.5.1 Detailed Description

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. This class provides support for accelerometer pipeline, and it used to listen the accelerometer sensor orientation changes.

Author

Saeik Firdose (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for accelerometer pipeline.

Author

Saeik Firdose (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. This class is provided as input the classify the the activities in real time.

Author

Saeik Firdose (COPELABS/ULHT)

5.6 Package cs.nsense.bluetooth

Classes

- class BluetoothCore
- interface BTDeviceFinder
- class BTManager
- class BTUserDevAverageEncounterDuration
- class BTUserDevEncounterDuration
- class BTUserDevice
- class BTUserDevSocialWeight
- class OnNewHourUpdate

5.6.1 Detailed Description

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 06-04-2016 Class is part of the NSense application. This class is contains the core functionalities of the application. The BTManager provides all the information from Bluetooth adapter so this class can perform the social context analysis prior to storing the required information in the database.

Author

Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 06-04-2016 Class is part of the NSense application. It provides the interface between BTManager and BluetoothCore.

Author

Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 06-04-2016 Class is part of the NSense application. This class provides some methods to provide extended functionality to the android BluetoothAdapter.

Author

Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 06-04-2016 Class is part of the NSense application. This class holds the average duration of encounter between a peer and the user device in specific time slots. The information kept are device MAC address and average encounter duration.

Author

Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 06-04-2016 Class is part of the NSense application. This class holds the encounter duration between a peer and the user device in specific time slots. The information kept are device MAC address and time spent in the vicinity (i.e., total time within communication range).

Author

Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 06-04-2016 Class is part of the NSense application. This class represents a user device found by means of Bluetooth. The information kept are device name, MAC address, and time of first encounter.

Author

Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 06-04-2016 Class is part of the NSense application. This class holds the social weight between a peer and the user device in specific time slots. The information kept are device MAC address and social weight.

Author

Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 06-04-2016 Class is part of the NSense application. This class is responsible for computing the social weight among users.

Author

Waldir Moreira (COPELABS/ULHT)

5.7 Package cs.nsense.db

Classes

- interface DataBaseChangeListener
- class NSenseDataSource
- class NSenseSQLiteHelper

5.7.1 Detailed Description

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for DataBase module.

Author

Saeik Firdose (COPELABS/ULHT)
Luis Lopes (COPELABS/ULHT),
Waldir Moreira (COPELABS/ULHT),
Reddy Pallavali (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. This class support for DataBase module and it provides various methods to access the DB from external modules.

Author

Saeik Firdose (COPELABS/ULHT),
Luis Lopes (COPELABS/ULHT),
Waldir Moreira (COPELABS/ULHT),
Reddy Pallavali (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for DataBase module. This class provides helper methods to the NSenseDataSource.

Author

Saeik Firdose (COPELABS/ULHT),
Luis Lopes (COPELABS/ULHT),
Waldir Moreira (COPELABS/ULHT),
Reddy Pallavali (COPELABS/ULHT)

5.8 Package cs.nsense.inferenceModule

Classes

- class SocialInteraction
- class SocialityDetails
- class SocialityDetailsListParcelable

5.8.1 Detailed Description

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for NSenseService and NSenseActivity classes to compute Social Interaction and Propinquity.

Author

Saeik Firdose (COPELABS/ULHT)
Waldir Moreira (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for inference module and provides the SocialityDetails object to store the information to compute Social Interaction and Propinquity.

Author

Saeik Firdose (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides helper methods for SocialityDetails class.

Author

Saeik Firdose (COPELABS/ULHT)

5.9 Package cs.nsense.location

Classes

- class CleanTask
- class LocationEntry
- class LocationPipeline
- class RelativePositionWiFiNoConnection
- class WifiAccessPoint
- class WifiServiceSearcher

5.9.1 Detailed Description

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 16-11-2015 Class is part of the NSense application. This class verifies every location entry at the DB and removes the entries that were updated more than 6 minutes ago. This class is called by a Timer, and it is repeatedly called.

Author

Luis Amaral Lopes (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 16-11-2015 Class is part of the NSense application. This class represents a entry from the location table.

Author

Luis Amaral Lopes (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 16-11-2015 Class is part of the NSense application. This class is the entry point of Location Pipeline. It initializes all necessary modules to compute the relative distance to other devices.

Author

Luis Amaral Lopes (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 16-11-2015 Class is part of the NSense application. It manages the Wi-Fi and also the Wi-Fi P2P to compute the relative distance to others devices.

Author

Luis Amaral Lopes (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 16-11-2015 Class is part of the NSense application. This class is the entry point of Location Pipeline. When there are nsense devices near (provided by WifiServiceSearcher) it performs a WiFi scan to detect this devices a to compute the respective relative distance.

Author

Luis Amaral Lopes (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, 16-11-2015 Class is part of the NSense application. This class manage all the devices found at the WiFi p2p discover mechanism.

Author

Luis Amaral Lopes (COPELABS/ULHT)

5.10 Package cs.nsense.microphone

Classes

- class MicrophonePipeline
- class SoundLevel
- interface SoundListener
- class SoundManager

5.10.1 Detailed Description

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for Microphone pipeline. This class is contains the core functionalities of the application relating to Microphone. The SoundManager provides all the information from Microphone adapter so this class can perform the analysis over Environmental Sound prior to storing the required information in the database. Some of the code is adapted from Google's NoiseAlert application code.

Author

Reddy Pallavali (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for Microphone pipeline and this class extracts the different sound levels such as Quite, Normal, Alert, and Noisy based on the sound frequency levels.

Author

Reddy Pallavali (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for Microphone pipeline and this class is the listener class to update sound levels in the application.

Author

Reddy Pallavali (COPELABS/ULHT)

Version

1.3 COPYRIGHTS COPELABS/ULHT, LGPLv3.0, date (e.g. 22-04-2016) Class is part of the NSense application. It provides support for Microphone pipeline. This class provides the core idealogy of sound process with microphone adapter.

Author

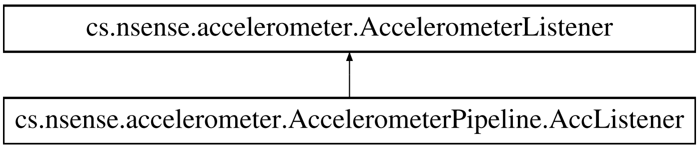
Reddy Pallavali (COPELABS/ULHT)

Chapter 6

Class Documentation

6.1 cs.nsense.accelerometer.AccelerometerListener Interface Reference

Inheritance diagram for cs.nsense.accelerometer.AccelerometerListener:


cs.nsense.accelerometer.AccelerometerListener
cs.nsense.accelerometer.AccelerometerPipeline.AccListener

Public Member Functions

- void updateBuffer (float x, float y, float z)

6.1.1 Detailed Description

This class provides an interface to updateBuffer

6.1.2 Member Function Documentation

6.1.2.1 void cs.nsense.accelerometer.AccelerometerListener.updateBuffer (float *x*, float *y*, float *z*)

This method updateBuffer when accelerometer sensor orientation changes

Parameters

x	X-axis represent tilting the phone from left - right and vice versa
y	Y-axis represent tilting the phone upside down and vice versa
z	Z-axis represent lifting the phone screen side up and down

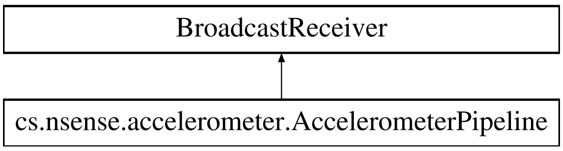
Implemented in cs.nsense.accelerometer.AccelerometerPipeline.AccListener.

The documentation for this interface was generated from the following file:

- src/cs/nsense/accelerometer/AccelerometerListener.java

6.2 cs.nsense.accelerometer.AccelerometerPipeline Class Reference

Inheritance diagram for cs.nsense.accelerometer.AccelerometerPipeline:



Classes

- class AccListener
- class ActivityClassificationTask
- class HourChangeAlarmReceiver
- class PreviousAction

Public Member Functions

- AccelerometerPipeline (NSenseService callback, NSenseDataSource dataSource)
- void close (Context contect)
- int getTimeSlot ()
- void appendLog (String text, String start)
- void onReceive (Context context, Intent intent)

Static Public Member Functions

- static int[] convertIntegers (List< Integer > integers)
- static int filterActivity (int[] k1)

Static Public Attributes

- static String mpreviousActionType = ""
- static double mpreviousDuration = 0

Private Member Functions

- void notifyDataBaseChange ()

Private Attributes

- Context mContext = null
- ActivityClassificationTask mActivityClassificationTask
- AccelerSenManager accelerSenManager
- AccelerometerListener accelerListener
- NSenseDataSource dataSource
- NSenseService callback = null
- SensorManager mSensorManager
- ArrayList< DataBaseChangeListener > listeners = new ArrayList<DataBaseChangeListener> ()
- PreviousAction mPreviousAction = new PreviousAction()

Static Private Attributes

- static final String TAG = "ACCELEROMETER"
- static ArrayBlockingQueue< Double > mAccBuffer
- static String actiontype = ""
- static ActionsEntry actualActionEntry = null

6.2.1 Detailed Description

This class provides various methods to provide Accelerometer sensor data

6.2.2 Constructor & Destructor Documentation

6.2.2.1 cs.nsense.accelerometer.AccelerometerPipeline.AccelerometerPipeline (NSenseService *callback*, NSenseDataSource *dataSource*)

This class instantiate the AccelerometerPipeline and initialize the listener and activity classification task

Parameters

callback	Supply functionality for NSenseActivity to use
dataSource	NSenseDataSource to access various methods and information of the NSense Data base

6.2.3 Member Function Documentation

6.2.3.1 void cs.nsense.accelerometer.AccelerometerPipeline.appendLog (String *text*, String *start*)

This method store the logs into NSenseActivity.txt file

Parameters

text	Activity Type
start	Activity Start time

BufferedWriter for performance, true to set append to file flag

6.2.3.2 void cs.nsense.accelerometer.AccelerometerPipeline.close (Context *contect*)

This method close the context

Parameters

context	Interface to global information about an application environment.

6.2.3.3 static int [] cs.nsense.accelerometer.AccelerometerPipeline.convertIntegers (List< Integer > *integers*) [static]

This method converts list of integers

Parameters

integers	list with integers

Returns

ret list with integers

6.2.3.4 static int cs.nsense.accelerometer.AccelerometerPipeline.filterActivity (int[ ] *k1*) [static]

This method to filter the activities to provide the confidence level

Parameters

k1	Number of activity instances

Returns

activityType activityType Type of the activity

6.2.3.5 int cs.nsense.accelerometer.AccelerometerPipeline.getTimeSlot ( )

This method provides the current time slot

Returns

currentTimeSlot current time slot

6.2.3.6 void cs.nsense.accelerometer.AccelerometerPipeline.notifyDataBaseChange ( ) `[private]`

This method provides the notifications when the database has new entries or updates

6.2.3.7 void cs.nsense.accelerometer.AccelerometerPipeline.onReceive (Context *context,* Intent *intent*)

This method of the receiver is called by the Android system

Parameters

context	Interface to global information about an application environment.
intent	The intent.

6.2.4 Member Data Documentation

6.2.4.1 AccelerometerListener cs.nsense.accelerometer.AccelerometerPipeline.accelerListener `[private]`

This class is to access functionality of Accelerometer Listener

6.2.4.2 AccelerSenManager cs.nsense.accelerometer.AccelerometerPipeline.accelerSenManager `[private]`

This class is to access functionality of Sensor Manager

6.2.4.3 String cs.nsense.accelerometer.AccelerometerPipeline.actiontype = "" `[static]`, `[private]`

This variable is to store action type

6.2.4.4 ActionsEntry cs.nsense.accelerometer.AccelerometerPipeline.actualActionEntry = null `[static]`, `[private]`

This variable is to get the action details

6.2.4.5 NSenseService cs.nsense.accelerometer.AccelerometerPipeline.callback = null `[private]`

This class is to access the functionality of NSense Service

6.2.4.6 NSenseDataSource cs.nsense.accelerometer.AccelerometerPipeline.dataSource [private]

This class is to access functionality of NSense Data base

6.2.4.7 ArrayList<DataBaseChangeListener> cs.nsense.accelerometer.AccelerometerPipeline.listeners = new ArrayList<DataBaseChangeListener> () [private]

This variable is to get the database listeners

6.2.4.8 ArrayBlockingQueue<Double> cs.nsense.accelerometer.AccelerometerPipeline.mAccBuffer [static], [private]

This variable is to store the Buffer

6.2.4.9 ActivityClassificationTask cs.nsense.accelerometer.AccelerometerPipeline.mActivityClassificationTask [private]

This class is to access functionality from Activity Classification Task of accelerometer pipeline

6.2.4.10 Context cs.nsense.accelerometer.AccelerometerPipeline.mContext = null [private]

Interface to global information about an application environment.

6.2.4.11 PreviousAction cs.nsense.accelerometer.AccelerometerPipeline.mPreviousAction = new PreviousAction() [private]

6.2.4.12 String cs.nsense.accelerometer.AccelerometerPipeline.mpreviousActionType = "" [static]

6.2.4.13 double cs.nsense.accelerometer.AccelerometerPipeline.mpreviousDuration = 0 [static]

6.2.4.14 SensorManager cs.nsense.accelerometer.AccelerometerPipeline.mSensorManager [private]

This class is to access functionality of Sensor Manager

6.2.4.15 final String cs.nsense.accelerometer.AccelerometerPipeline.TAG = "ACCELEROMETER" [static], [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/AccelerometerPipeline.java

6.3 cs.nsense.accelerometer.AccelerSenManager Class Reference

Public Member Functions

- AccelerSenManager (Context context)
- void close ()
- void setAccelerListener (AccelerometerListener listener)
- void stopListening ()
- void startListening ()

Static Public Member Functions

- static boolean isListening ()
- static boolean isSupported (Context context)

Static Private Attributes

- static final String TAG = "ACCELEROMETER"
- static Context aContext = null
- static Sensor sensor
- static SensorManager sensorManager
- static AccelerometerListener listener
- static Boolean supported
- static boolean running = false
- static SensorEventListener sensorEventListener

6.3.1 Detailed Description

This class provides some methods to provide the actions to the NSenseService

6.3.2 Constructor & Destructor Documentation

6.3.2.1 cs.nsense.accelerometer.AccelerSenManager.AccelerSenManager (Context *context*)

This method start listening to the accelerometer sensor changes

Parameters

context	Interface to global information about an application environment.

6.3.3 Member Function Documentation

6.3.3.1 void cs.nsense.accelerometer.AccelerSenManager.close ( )

This method close the listener if it is not listening

6.3.3.2 static boolean cs.nsense.accelerometer.AccelerSenManager.isListening ( ) [static]

This method to checck whether sensor manager class is listening to orientation changes

Returns

running true if the manager is listening to orientation changes

6.3.3.3 static boolean cs.nsense.accelerometer.AccelerSenManager.isSupported (Context *context*) [static]

This method returns true if at least one Accelerometer sensor is available

Parameters

context	Interface to global information about an application environment.

Returns

true if Accelerometer sensor is available

6.3.3.4 void cs.nsense.accelerometer.AccelerSenManager.setAccelerListener (**AccelerometerListener** *listener*)

This method set the listener

Parameters

listener	To listen the accelerometer sensor orientation changes

6.3.3.5 void cs.nsense.accelerometer.AccelerSenManager.startListening ( )

This method registers a listener and start listening

Returns

running accelerometerListener callback for accelerometer events

6.3.3.6 void cs.nsense.accelerometer.AccelerSenManager.stopListening ( )

This method Unregisters listeners

6.3.4 Member Data Documentation

6.3.4.1 Context cs.nsense.accelerometer.AccelerSenManager.aContext = null `[static]`, `[private]`

Interface to global information about an application environment.

6.3.4.2 **AccelerometerListener** cs.nsense.accelerometer.AccelerSenManager.listener `[static]`, `[private]`

you could use an OrientationListener array instead if you plans to use more than one listener

6.3.4.3 boolean cs.nsense.accelerometer.AccelerSenManager.running = false `[static]`, `[private]`

indicates whether or not Accelerometer Sensor is running

6.3.4.4 Sensor cs.nsense.accelerometer.AccelerSenManager.sensor `[static]`, `[private]`

This class is to check the type of sensor

6.3.4.5 SensorEventListener cs.nsense.accelerometer.AccelerSenManager.sensorEventListener `[static]`, `[private]`

Initial value:

```
=
            new SensorEventListener() {

        private float x = 0;
        private float y = 0;
        private float z = 0;

        public void onAccuracyChanged(Sensor sensor, int accuracy) {}

        public void onSensorChanged(SensorEvent event) {

            x = event.values[0];
            y = event.values[1];
            z = event.values[2];

            listener.updateBuffer(x, y, z);
        }

    }
```

This method listen to events from the accelerometer listener

6.3.4.6 SensorManager cs.nsense.accelerometer.AccelerSenManager.sensorManager [static], [private]

This class is to access functionality of Sensor Manager

6.3.4.7 Boolean cs.nsense.accelerometer.AccelerSenManager.supported [static], [private]

indicates whether or not Accelerometer Sensor is supported

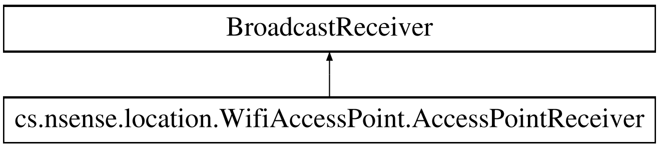
6.3.4.8 final String cs.nsense.accelerometer.AccelerSenManager.TAG = "ACCELEROMETER" [static], [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/AccelerSenManager.java

6.4 cs.nsense.location.WifiAccessPoint.AccessPointReceiver Class Reference

Inheritance diagram for cs.nsense.location.WifiAccessPoint.AccessPointReceiver:



Public Member Functions

- void onReceive (Context context, Intent intent)

6.4.1 Detailed Description

BroadcastReceiver that receives events about WiFi state and WiFi connection state.

Author

Luis Amaral Lopes (COPELABS/ULHT)

6.4.2 Member Function Documentation

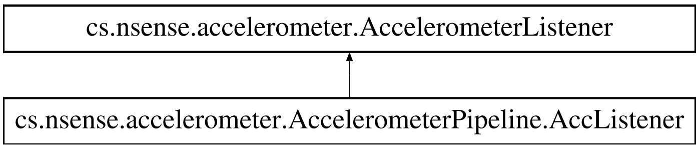
6.4.2.1 void cs.nsense.location.WifiAccessPoint.AccessPointReceiver.onReceive (Context *context,* Intent *intent*)

The documentation for this class was generated from the following file:

- src/cs/nsense/location/WifiAccessPoint.java

6.5 cs.nsense.accelerometer.AccelerometerPipeline.AccListener Class Reference

Inheritance diagram for cs.nsense.accelerometer.AccelerometerPipeline.AccListener:


cs.nsense.accelerometer.AccelerometerListener
cs.nsense.accelerometer.AccelerometerPipeline.AccListener

Public Member Functions

- void updateBuffer (float x, float y, float z)

6.5.1 Detailed Description

This class provide the method to listen the accelerometer changes

6.5.2 Member Function Documentation

6.5.2.1 void cs.nsense.accelerometer.AccelerometerPipeline.AccListener.updateBuffer (float *x,* float *y,* float *z*)

This method updateBuffer when accelerometer sensor orientation changes

Parameters

x	X-axis represent tilting the phone from left - right and vice versa
y	Y-axis represent tilting the phone upside down and vice versa
z	Z-axis represent lifting the phone screen side up and down

Exception happens when reach the capacity. Doubling the buffer. ListBlockingQueue has no such issue, But generally has worse performance

Implements cs.nsense.accelerometer.AccelerometerListener.

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/AccelerometerPipeline.java

6.6 cs.nsense.accelerometer.ActionsEntry Class Reference

Public Member Functions

- ActionsEntry ()
- String toString ()

- String getActionType ()
- void setActionType (String actionType)
- long getActionStartTime ()
- long getDay ()
- void setDay (long day)
- int getHour ()
- void setHour (int hour)
- void setActionStartTime (long actionStartTime)
- double getAverageDuration ()
- void setAverageDuration (double averageDuration)
- long getActionEndTime ()
- void setActionEndTime (long actionEndTime)
- String getTimeFrame ()
- void setTimeFrame (String timeFrame)
- int getActionCounter ()
- void setActionCounter (int actionCounter)
- int getID ()
- void setID (int iD)

Private Attributes

- int ID
- String ActionType
- long ActionStartTime
- double AverageDuration
- long ActionEndTime
- long Day
- int Hour
- String TimeFrame
- int ActionCounter

6.6.1 Detailed Description

This class provides various methods to provide the action details to the NSenseService

6.6.2 Constructor & Destructor Documentation

6.6.2.1 cs.nsense.accelerometer.ActionsEntry.ActionsEntry ( )

This method constructs the Actions Entry

6.6.3 Member Function Documentation

6.6.3.1 int cs.nsense.accelerometer.ActionsEntry.getActionCounter ( )

This method get the action counter

Returns

ActionCounter the action counter

6.6.3.2 long cs.nsense.accelerometer.ActionsEntry.getActionEndTime ( )

This method get the action end time

Returns

ActionEndTime the action end time

6.6.3.3 long cs.nsense.accelerometer.ActionsEntry.getActionStartTime ( )

This method to get Action Start Time

Returns

ActionStartTime the action start time

6.6.3.4 String cs.nsense.accelerometer.ActionsEntry.getActionType ( )

This method get the action type

Returns

ActionType the action type

6.6.3.5 double cs.nsense.accelerometer.ActionsEntry.getAverageDuration ( )

This method get the average duration

Returns

AverageDuration the average duration

6.6.3.6 long cs.nsense.accelerometer.ActionsEntry.getDay ( )

This method get the day

Returns

Day the current day

6.6.3.7 int cs.nsense.accelerometer.ActionsEntry.getHour ( )

This method get the hour

Returns

hour the hour

6.6.3.8 int cs.nsense.accelerometer.ActionsEntry.getID ( )

This method get the id

Returns

the iD ithe id

6.6.3.9 String cs.nsense.accelerometer.ActionsEntry.getTimeFrame ( )

This method get the time frame

Returns

TimeFrame the time frame

6.6.3.10 void cs.nsense.accelerometer.ActionsEntry.setActionCounter (int *actionCounter*)

This method set the action counter

Parameters

actionCounter	the action counter

6.6.3.11 void cs.nsense.accelerometer.ActionsEntry.setActionEndTime (long *actionEndTime*)

This method set the action end time

Parameters

actionEndTime	the action end time

6.6.3.12 void cs.nsense.accelerometer.ActionsEntry.setActionStartTime (long *actionStartTime*)

This method set the action start time

Parameters

actionStartTime	the action start time

6.6.3.13 void cs.nsense.accelerometer.ActionsEntry.setActionType (String *actionType*)

This method set the actionType

Parameters

actionType	the action type

6.6.3.14 void cs.nsense.accelerometer.ActionsEntry.setAverageDuration (double *averageDuration*)

This method set the averageDuration

Parameters

averageDuration	the average duration

6.6.3.15 void cs.nsense.accelerometer.ActionsEntry.setDay (long *day*)

This method set the day

Parameters

day	the current day

6.6.3.16 void cs.nsense.accelerometer.ActionsEntry.setHour (int *hour*)

This method set hour

Parameters

hour	the hour

6.6.3.17 void cs.nsense.accelerometer.ActionsEntry.setID (int *iD*)

This method set the iD

Parameters

iD	the iD

6.6.3.18 void cs.nsense.accelerometer.ActionsEntry.setTimeFrame (String *timeFrame*)

This method set the time frame

Parameters

timeFrame	the time frame

6.6.3.19 String cs.nsense.accelerometer.ActionsEntry.toString ( )

This method return a string containing the ACTIONTYPE, ActionStartTime, AverageDuration, ActionEndTime, HOUR, ACTIONCOUNTER, DAY, and TIMEFRAME.

Returns

string String with the ACTIONTYPE, ActionStartTime, AverageDuration, ActionEndTime, HOUR, ACTIONCOUNTER, DAY, and TIMEFRAME values

6.6.4 Member Data Documentation

6.6.4.1 int cs.nsense.accelerometer.ActionsEntry.ActionCounter [private]

ActionCounter variable of the ActionsEntry object

6.6.4.2 long cs.nsense.accelerometer.ActionsEntry.ActionEndTime [private]

ActionEndTime variable of the ActionsEntry object

6.6.4.3 long cs.nsense.accelerometer.ActionsEntry.ActionStartTime [private]

ActionStartTime variable of the ActionsEntry object

6.6.4.4 String cs.nsense.accelerometer.ActionsEntry.ActionType [private]

ActionType variable of the ActionsEntry object

6.6.4.5 double cs.nsense.accelerometer.ActionsEntry.AverageDuration [private]

AverageDuration variable of the ActionsEntry object

6.6.4.6 long cs.nsense.accelerometer.ActionsEntry.Day [private]

Day variable of the ActionsEntry object

6.6.4.7 int cs.nsense.accelerometer.ActionsEntry.Hour [private]

Hour variable of the ActionsEntry object

6.6.4.8 int cs.nsense.accelerometer.ActionsEntry.ID [private]

Id variable of the ActionsEntry object

6.6.4.9 String cs.nsense.accelerometer.ActionsEntry.TimeFrame [private]

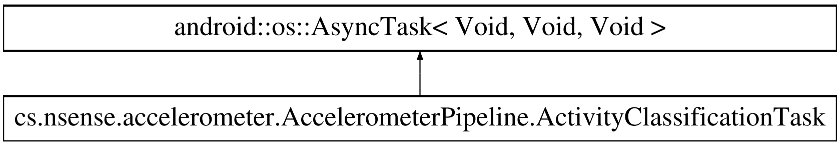
TimeFrame variable of the ActionsEntry object

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/ActionsEntry.java

6.7 cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask Class Reference

Inheritance diagram for cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask:


android::os::AsyncTask< Void, Void, Void >
cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask

Public Member Functions

- ActivityClassificationTask ()
- void registerActiononDB (String actiontype, double mDuration, long start)

Protected Member Functions

- Void doInBackground (Void...arg0)

Private Member Functions

- void updateAction (String actiontype, double mDuration, long start)

Static Private Attributes

- static final int mFeatLen = Globals.ACCELEROMETER_BLOCK_CAPACITY + 2

6.7.1 Detailed Description

This class provides the real time activities by extending with asynchronous task

6.7.2 Constructor & Destructor Documentation

6.7.2.1 cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask.ActivityClassificationTask ( )

6.7.3 Member Function Documentation

6.7.3.1 Void cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask.doInBackground (Void... *arg0*) [protected]

Get Hour and Day

Clear the field

Append max after frequency component

Initialize array of 16 size and Store the entries computing the relevant activity and clear the array size and reinitialize array

Notifying the actiontype to the service

clearing feature vector

6.7.3.2 void cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask.registerActiononDB (String *actiontype*, double *mDuration*, long *start*)

This method to register actions in data base

Parameters

actiontype	Current action type
mDuration	Duration of the current activity
start	Start time of the activity

logic for Weekdays and weekend

6.7.3.3 void cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask.updateAction (String *actiontype*, double *mDuration*, long *start*) [private]

This method update the database when the activity is changed

Parameters

actiontype	Current action type

mDuration	Duration of the current activity
start	Start time of the activity

Get Hour and Day

6.7.4 Member Data Documentation

6.7.4.1 final int cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask.mFeatLen = Globals.ACCELEROMETER_BLOCK_CAPACITY + 2 [static], [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/AccelerometerPipeline.java

6.8 cs.nsense.bluetooth.BluetoothCore Class Reference

Classes

- class CustomComparator
- class **ServiceBTListener**
- class socialWeight

Public Member Functions

- BluetoothCore (NSenseService callback, Context context, NSenseDataSource datasource)
- boolean isDeviceOnDB (String deviceAdd)
- ArrayList< socialWeight > getSocialWeightDetails ()
- void close (Context context)
- int getTimeSlot ()

Private Member Functions

- void notifyDataBaseChange ()

Private Attributes

- BTManager myBTManager
- ServiceBTListener btListener
- AlarmManager alarmMgr
- PendingIntent alarmIntent
- OnNewHourUpdate newHour
- NSenseDataSource datasource
- NSenseService callback = null

6.8.1 Constructor & Destructor Documentation

6.8.1.1 cs.nsense.bluetooth.BluetoothCore.BluetoothCore (NSenseService *callback*, Context *context*, NSenseDataSource *datasource*)

This method is the constructor for BluetoothCore.

Parameters

callback	Supply NSenseService object for NSenseActivity to use
context	Interface to global information about an application environment.
datasource	NSenseDataSource to access various methods and information of the NSense Data base

6.8.2 Member Function Documentation

6.8.2.1 void cs.nsense.bluetooth.BluetoothCore.close (Context *context*)

This method unregisters BroadcastReceiver and closes the Bluetooth manager.

Parameters

context	Interface to global information about an application environment.

6.8.2.2 ArrayList<socialWeight> cs.nsense.bluetooth.BluetoothCore.getSocialWeightDetails ( )

This method provides social weight information for encountered devices.

Returns

listWeights The list of devices and social weights towards them in descending order.

Getting highest scial weight value

6.8.2.3 int cs.nsense.bluetooth.BluetoothCore.getTimeSlot ( )

This method provides the current time slot.

Returns

currentTimeSlot The actual time slot.

6.8.2.4 boolean cs.nsense.bluetooth.BluetoothCore.isDeviceOnDB (String *deviceAdd*)

This method checks whether the device is in database.

Parameters

deviceAdd	The device MAC address.

Returns

true, if device is in DB, false otherwise.

6.8.2.5 void cs.nsense.bluetooth.BluetoothCore.notifyDataBaseChange ( ) [private]

This method notifies a database change to the listeners.

6.8.3 Member Data Documentation

6.8.3.1 PendingIntent cs.nsense.bluetooth.BluetoothCore.alarmIntent [private]

This class is to access functionality of PendingIntent

6.8.3.2 AlarmManager cs.nsense.bluetooth.BluetoothCore.alarmMgr [private]

This class is to access functionality of AlarmManager

6.8.3.3 ServiceBTListener cs.nsense.bluetooth.BluetoothCore.btListener [private]

This class is to access functionality of ServiceBTListener

6.8.3.4 NSenseService cs.nsense.bluetooth.BluetoothCore.callback = null [private]

This class is to access the functionality of NSense Service

6.8.3.5 NSenseDataSource cs.nsense.bluetooth.BluetoothCore.datasource [private]

This class is to access functionality of NSense Data base

6.8.3.6 BTManager cs.nsense.bluetooth.BluetoothCore.myBTManager [private]

This class is to access functionality of BTManager

6.8.3.7 OnNewHourUpdate cs.nsense.bluetooth.BluetoothCore.newHour [private]

This class is to access functionality of OnNewHourUpdate

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/BluetoothCore.java

6.9 cs.nsense.bluetooth.BTDeviceFinder Interface Reference

Inherited by cs.nsense.bluetooth.BluetoothCore.ServiceBTListener.

Public Member Functions

- void onDeviceFound (BluetoothDevice device, BluetoothClass btClass)

6.9.1 Detailed Description

This class provides the interface between BTManager and BluetoothCore.

Author

Waldir Moreira (COPELABS/ULHT)

6.9.2 Member Function Documentation

6.9.2.1 void cs.nsense.bluetooth.BTDeviceFinder.onDeviceFound (BluetoothDevice *device,* BluetoothClass *btClass*)

This method called when device found from BluetoothDevice and BluetoothClass

Parameters

device	Bluetooth Device
btClass	Bluetooth Class

The documentation for this interface was generated from the following file:

- src/cs/nsense/bluetooth/BTDeviceFinder.java

6.10 cs.nsense.bluetooth.BTManager Class Reference

Classes

- class **adapterBroadcastReceiver**
- class **FindBluetoothDevice**

Public Member Functions

- BTManager (Context c)
- void close (Context c)
- void startPeriodicScanning ()
- boolean isBTEnabled ()
- boolean enableBT ()
- boolean startDiscovery ()
- void setOnBTChangeListener (BTDeviceFinder listener)
- void clearOnBTChangeListener ()
- List< String > getLocalInfo ()

Public Attributes

- boolean isScanningActive = false
- boolean isWaitingScanResults = false

Static Public Attributes

- static int DISCOVER_INTERVAL = 180000

Private Member Functions

- void stopPeriodicScanning ()

Private Attributes

- boolean debug = true
- Context mContext
- BluetoothAdapter androidBTAdapter
- BluetoothServerSocket socket
- UUID uuid = UUID.fromString("10101010-0101-0101-0101-010101010101")
- Map< BluetoothDevice,
  BluetoothClass > btDeviceList = new HashMap<BluetoothDevice, BluetoothClass>()
- FindBluetoothDevice BTDevFinder
- adapterBroadcastReceiver adapterReceiver

- BTDeviceFinder listener
- String fetchingUUIDfromDevice
- Handler mHandler = new Handler()
- Runnable runScan

Static Private Attributes

- static final String TAG = "BTManager"
- static boolean mBTisTurningOnOff = false

6.10.1 Constructor & Destructor Documentation

6.10.1.1 cs.nsense.bluetooth.BTManager.BTManager (Context *c*)

This method is the constructor for BTManager.

Parameters

c	Interface to global information about an application environment.

Advertised UUID to let peer node knows that current node runs nsense!

6.10.2 Member Function Documentation

6.10.2.1 void cs.nsense.bluetooth.BTManager.clearOnBTChangeListener ( )

This method clears the change listener.

6.10.2.2 void cs.nsense.bluetooth.BTManager.close (Context *c*)

This method stops scanning and unregister BroadcastReceivers.

Parameters

c	Interface to global information about an application environment.

6.10.2.3 boolean cs.nsense.bluetooth.BTManager.enableBT ( )

This method enables Bluetooth.

Returns

true/false If Bluetooth Adapter is Turning on or off

6.10.2.4 List<String> cs.nsense.bluetooth.BTManager.getLocalInfo ( )

This method provides the MAC Address and name of local Bluetooth adapter.

Returns

localinfo The local MAC Address

6.10.2.5 boolean cs.nsense.bluetooth.BTManager.isBTEnabled ( )

This method checks whether Bluetooth is enabled.

Returns

true/false If Bluetooth Adapter is enabled/not

6.10.2.6 void cs.nsense.bluetooth.BTManager.setOnBTChangeListener (BTDeviceFinder *listener*)

This method sets a change listener.

Parameters

listener	The BTDeviceFinder listener

6.10.2.7 boolean cs.nsense.bluetooth.BTManager.startDiscovery ( )

This method starts Bluetooth discovery process.

Returns

true/false True on success, false on failure

6.10.2.8 void cs.nsense.bluetooth.BTManager.startPeriodicScanning ( )

This method starts scanning.

6.10.2.9 void cs.nsense.bluetooth.BTManager.stopPeriodicScanning ( ) [private]

This method stops scanning.

6.10.3 Member Data Documentation

6.10.3.1 adapterBroadcastReceiver cs.nsense.bluetooth.BTManager.adapterReceiver [private]

This class is to access functionality of Bluetooth Adapter

6.10.3.2 BluetoothAdapter cs.nsense.bluetooth.BTManager.androidBTAdapter [private]

This class is to access functionality of Bluetooth Adapter

6.10.3.3 FindBluetoothDevice cs.nsense.bluetooth.BTManager.BTDevFinder [private]

Used to identify Bluetooth Device

6.10.3.4 Map<BluetoothDevice, BluetoothClass> cs.nsense.bluetooth.BTManager.btDeviceList = new HashMap<BluetoothDevice, BluetoothClass>() [private]

Used to access the Bluetooth device list

6.10.3.5 boolean cs.nsense.bluetooth.BTManager.debug = true [private]

for debugging purposes

6.10.3.6 int cs.nsense.bluetooth.BTManager.DISCOVER_INTERVAL = 180000 [static]

check how long BT scan is !!!!!!

6.10.3.7 String cs.nsense.bluetooth.BTManager.fetchingUUIDfromDevice [private]

This variable for fetching UUID from Device

6.10.3.8 boolean cs.nsense.bluetooth.BTManager.isScanningActive = false

This variable is to check Scanning is Active

6.10.3.9 boolean cs.nsense.bluetooth.BTManager.isWaitingScanResults = false

This variable is to check Waiting for Scan Results

6.10.3.10 BTDeviceFinder cs.nsense.bluetooth.BTManager.listener [private]

This class is to access the listener of Bluetooth device finder

6.10.3.11 boolean cs.nsense.bluetooth.BTManager.mBTisTurningOnOff = false [static], [private]

This variable is to check Bluetooth is turning on or off

6.10.3.12 Context cs.nsense.bluetooth.BTManager.mContext [private]

Interface to global information about an application environment.

6.10.3.13 Handler cs.nsense.bluetooth.BTManager.mHandler = new Handler() [private]

This class is to use send or process message

6.10.3.14 Runnable cs.nsense.bluetooth.BTManager.runScan [private]

Initial value:

```
= new Runnable() {
        public void run() {
            if (isScanningActive) {
                if (!isBTEnabled()) {
                    if(enableBT()) {
                        stopPeriodicScanning();
                    } else {
                        Toast.makeText(mContext, "Error enabling BT. Closing Social Pipeline.",
      Toast.LENGTH_SHORT).show();
                        close(mContext);
                    }
                    return;
                }
                if (!isWaitingScanResults && isBTEnabled()) {
                    btDeviceList.clear();
                    if (startDiscovery()) {
                        if(debug){
                            Log.i(TAG, "Starting BT scaning...");
                        }
                        isWaitingScanResults = true;
                    }
                    mHandler.postDelayed(runScan, DISCOVER_INTERVAL);
                }
                else if (isWaitingScanResults) {
                    mHandler.postDelayed(runScan, DISCOVER_INTERVAL);
                    isWaitingScanResults = false;
                }
            }
        }
    }
```

This method allows for starting and stopping periodic Bluetooth scanning.

6.10.3.15 BluetoothServerSocket cs.nsense.bluetooth.BTManager.socket [private]

Used to identify NSense! devices

6.10.3.16 final String cs.nsense.bluetooth.BTManager.TAG = "BTManager" [static], [private]

6.10.3.17 UUID cs.nsense.bluetooth.BTManager.uuid = UUID.fromString("10101010-0101-0101-0101-010101010101") [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/BTManager.java

6.11 cs.nsense.bluetooth.BTUserDevAverageEncounterDuration Class Reference

Public Member Functions

- String getDevAdd ()
- double getAverageEncounterDuration (int timeSlot)
- void setDevAdd (String DevAdd)
- void setAverageEncounterDuration (int timeSlot, double avgEncounterDuration)
- BTUserDevAverageEncounterDuration ()

Private Attributes

- String deviceAdd
- double[] averageEncounterDuration = new double [24]

6.11.1 Detailed Description

This class provides the average duration of encounter between a peer and the user device in specific time slots

Author

Waldir Moreira (COPELABS/ULHT)

6.11.2 Constructor & Destructor Documentation

6.11.2.1 cs.nsense.bluetooth.BTUserDevAverageEncounterDuration.BTUserDevAverageEncounterDuration ( )

BT User Device Average Encounter Duration Constructor

6.11.3 Member Function Documentation

6.11.3.1 double cs.nsense.bluetooth.BTUserDevAverageEncounterDuration.getAverageEncounterDuration (int *timeSlot*)

This method gets the average encounter duration that the BT device is within communication range.

Parameters

timeSlot	The specific time slot.

Returns

The average encounter duration in the given slot.

6.11.3.2 String cs.nsense.bluetooth.BTUserDevAverageEncounterDuration.getDevAdd ( )

This method gets the MAC address of this Bluetooth device.

Returns

deviceAdd The device address to be used as key.

6.11.3.3 void cs.nsense.bluetooth.BTUserDevAverageEncounterDuration.setAverageEncounterDuration (int *timeSlot,* double *avgEncounterDuration*)

This method sets the average duration that the BT device is within communication range.

Parameters

timeSlot	The specific time slot.
avgEncounterDuration	The average duration of encounter.

6.11.3.4 void cs.nsense.bluetooth.BTUserDevAverageEncounterDuration.setDevAdd (String *DevAdd*)

This method sets the ID of this Bluetooth device.

Parameters

DevAdd	The MAC address of device to set.

6.11.4 Member Data Documentation

6.11.4.1 double [ ] cs.nsense.bluetooth.BTUserDevAverageEncounterDuration.averageEncounterDuration = new double [24] `[private]`

This variable is to get the device average duration in specific time slots

6.11.4.2 String cs.nsense.bluetooth.BTUserDevAverageEncounterDuration.deviceAdd `[private]`

This variable is to get the address of device

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/BTUserDevAverageEncounterDuration.java

6.12 cs.nsense.bluetooth.BTUserDevEncounterDuration Class Reference

Public Member Functions

- String getDevAdd ()
- double getEncounterDuration (int timeSlot)
- void setDevAdd (String DevAdd)
- void setEncounterDuration (int timeSlot, double encounterDuration)
- BTUserDevEncounterDuration ()

Private Attributes

- String deviceAdd
- double[] encounterDuration = new double [24]

6.12.1 Detailed Description

This class provide details of Bluetooth User Device and Encounter Duration

6.12.2 Constructor & Destructor Documentation

6.12.2.1 cs.nsense.bluetooth.BTUserDevEncounterDuration.BTUserDevEncounterDuration ( )

BT User Device Encounter Duration Constructor

6.12.3 Member Function Documentation

6.12.3.1 String cs.nsense.bluetooth.BTUserDevEncounterDuration.getDevAdd ( )

This method gets the MAC address of this Bluetooth device.

Returns

deviceAdd The device address to be used as key.

6.12.3.2 double cs.nsense.bluetooth.BTUserDevEncounterDuration.getEncounterDuration (int *timeSlot*)

This method gets the duration that the BT device is within communication range.

Parameters

timeSlot	The specific time slot.

Returns

The encounter duration in the given slot.

6.12.3.3 void cs.nsense.bluetooth.BTUserDevEncounterDuration.setDevAdd (String *DevAdd*)

This method sets the ID of this Bluetooth device.

Parameters

DevAdd	The MAC address of device to set.

6.12.3.4 void cs.nsense.bluetooth.BTUserDevEncounterDuration.setEncounterDuration (int *timeSlot,* double *encounterDuration*)

This method sets the duration that the BT device is within communication range.

Parameters

timeSlot	The specific time slot.
encounter-Duration	The duration of encounter.

6.12.4 Member Data Documentation

6.12.4.1 String cs.nsense.bluetooth.BTUserDevEncounterDuration.deviceAdd [private]

This variable is to get the address of device

6.12.4.2 double [ ] cs.nsense.bluetooth.BTUserDevEncounterDuration.encounterDuration = new double [24] [private]

This variable is to get the device encounter duration in specific time slots

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/BTUserDevEncounterDuration.java

6.13 cs.nsense.bluetooth.BTUserDevice Class Reference

Public Member Functions

- String getDevAdd ()
- String getDevName ()
- long getEncounterStart ()
- void setDevAdd (String DevAdd)
- void setDevName (String DevName)
- void setEncounterTime (long time)
- BTUserDevice ()

Private Attributes

- String deviceAdd
- String deviceName
- long encounterTime

6.13.1 Detailed Description

This class Provide details of User device name, MAC address, and time of first encounter.

6.13.2 Constructor & Destructor Documentation

6.13.2.1 cs.nsense.bluetooth.BTUserDevice.BTUserDevice ( )

Bluetooth User Device Constructor

6.13.3 Member Function Documentation

6.13.3.1 String cs.nsense.bluetooth.BTUserDevice.getDevAdd ( )

This method gets the MAC address of this Bluetooth device.

Returns

deviceAdd The device address.

6.13.3.2 String cs.nsense.bluetooth.BTUserDevice.getDevName ( )

This method gets the name of this Bluetooth device.

Returns

deviceName The device name.

6.13.3.3 long cs.nsense.bluetooth.BTUserDevice.getEncounterStart ( )

This method gets the time that the BT device is is first found.

Parameters

encounterTime	The time.

6.13.3.4 void cs.nsense.bluetooth.BTUserDevice.setDevAdd (String *DevAdd*)

This method sets the ID of this Bluetooth device.

Parameters

DevAdd	The MAC address of device to set.

6.13.3.5 void cs.nsense.bluetooth.BTUserDevice.setDevName (String *DevName*)

This method sets the name of this Bluetooth device.

Parameters

DevName	The name of device to set.

6.13.3.6 void cs.nsense.bluetooth.BTUserDevice.setEncounterTime (long *time*)

This method sets the time that the BT device is first found.

Parameters

time	The time.

6.13.4 Member Data Documentation

6.13.4.1 String cs.nsense.bluetooth.BTUserDevice.deviceAdd [private]

This variable is to get the address of device

6.13.4.2 String cs.nsense.bluetooth.BTUserDevice.deviceName [private]

This variable is to get the name of device

6.13.4.3 long cs.nsense.bluetooth.BTUserDevice.encounterTime [private]

This variable is to get the encounter time of device

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/BTUserDevice.java

6.14 cs.nsense.bluetooth.BTUserDevSocialWeight Class Reference

Public Member Functions

- String getDevAdd ()
- double getSocialWeight (int timeSlot)
- void setDevAdd (String DevAdd)
- void setSocialWeight (int timeSlot, double socialWeight)
- BTUserDevSocialWeight ()

Private Attributes

- String deviceAdd
- double[ ] socialWeight = new double [24]

6.14.1 Detailed Description

This class Provides the social weight between a peer and the user device in specific time slots.

6.14.2 Constructor & Destructor Documentation

6.14.2.1 cs.nsense.bluetooth.BTUserDevSocialWeight.BTUserDevSocialWeight ( )

BT User Device Social Weight Constructor

6.14.3 Member Function Documentation

6.14.3.1 String cs.nsense.bluetooth.BTUserDevSocialWeight.getDevAdd ( )

This method gets the MAC address of this Bluetooth device.

Returns

deviceAdd The device address to be used as key.

6.14.3.2 double cs.nsense.bluetooth.BTUserDevSocialWeight.getSocialWeight (int *timeSlot*)

This method gets the social weight towards a BT device.

Parameters

timeSlot	The specific time slot.

Returns

The social weight in the given slot.

6.14.3.3 void cs.nsense.bluetooth.BTUserDevSocialWeight.setDevAdd (String *DevAdd*)

This method sets the ID of this Bluetooth device.

Parameters

DevAdd	The MAC address of device to set.

6.14.3.4 void cs.nsense.bluetooth.BTUserDevSocialWeight.setSocialWeight (int *timeSlot,* double *socialWeight*)

This method sets the social weight towards a BT device.

Parameters

timeSlot	The specific time slot.
socialWeight	The social weight.

6.14.4 Member Data Documentation

6.14.4.1 String cs.nsense.bluetooth.BTUserDevSocialWeight.deviceAdd [private]

This variable is to get the address of device

6.14.4.2 double [ ] cs.nsense.bluetooth.BTUserDevSocialWeight.socialWeight = new double [24] [private]

This variable is to get the social weight in specific time slots

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/BTUserDevSocialWeight.java

6.15 cs.nsense.ChangeListener Interface Reference

Public Member Functions

- void onLocationOutdoorChange (String newMessage)
- void onLocationIndoorChange (List< LocationEntry > apEntries)
- void onActionChange (String newMessage)
- void onSocialWeightChange (ArrayList< socialWeight > arrayList)
- void onSoundLevelChange (String newSound)
- void onSocialInteractionChange (double socialInt)
- void onPropinquityChange (double propIn)
- void onGetSociabilityChange (ArrayList< SocialityDetails > newSocial)

6.15.1 Detailed Description

This class provides an interface to update pipeline information

6.15.2 Member Function Documentation

6.15.2.1 void cs.nsense.ChangeListener.onActionChange (String *newMessage*)

This method called when Action information has been changed

Parameters

newMessage	Action Type information

6.15.2.2 void cs.nsense.ChangeListener.onGetSociabilityChange (ArrayList< SocialityDetails > *newSocial*)

This method called when SocialityDetails information has been changed

Parameters

newSocial	List of SocialityDetails objects

6.15.2.3 void cs.nsense.ChangeListener.onLocationIndoorChange (List< LocationEntry > *apEntries*)

This method called when Location Indoor table has been changed in the database

Parameters

apEntries	List of LocationEntry objects

6.15.2.4 void cs.nsense.ChangeListener.onLocationOutdoorChange (String *newMessage*)

This method called when Location Outdoor table has been changed in the database

Parameters

newMessage	Out door location information

6.15.2.5 void cs.nsense.ChangeListener.onPropinquityChange (double *propIn*)

This method called when Propinquity information has been changed

Parameters

propIn	Propinquity information

6.15.2.6 void cs.nsense.ChangeListener.onSocialInteractionChange (double *socialInt*)

This method called when Social Interaction information has been changed

Parameters

socialInt	Social Interaction information

6.15.2.7 void cs.nsense.ChangeListener.onSocialWeightChange (ArrayList< socialWeight > *arrayList*)

This method called when social Weight table has been changed

Parameters

arrayList	List of socialWeight objects

6.15.2.8 void cs.nsense.ChangeListener.onSoundLevelChange (String *newSound*)

This method called when Sound Level has been changed

Parameters

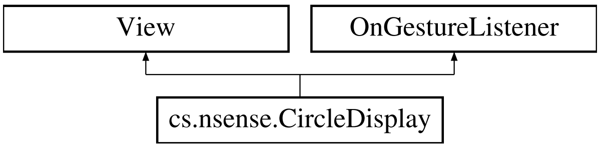
newSound	Sound level information

The documentation for this interface was generated from the following file:

- src/cs/nsense/ChangeListener.java

6.16 cs.nsense.CircleDisplay Class Reference

Inheritance diagram for cs.nsense.CircleDisplay:



Classes

- interface SelectionListener
- class **Utils**

Public Member Functions

- CircleDisplay (Context context)
- CircleDisplay (Context context, AttributeSet attrs)
- CircleDisplay (Context context, AttributeSet attrs, int defStyleAttr)
- void showValue (float toShow, float total, boolean animated)

- void setUnit (String unit)
- float getValue ()
- void startAnim ()
- void setAnimDuration (int durationmillis)
- float getDiameter ()
- float getRadius ()
- void setStartAngle (float angle)
- float getPhase ()
- void setPhase (float phase)
- void setDrawInnerCircle (boolean enabled)
- boolean isDrawInnerCircleEnabled ()
- void setDrawText (boolean enabled)
- boolean isDrawTextEnabled ()
- void setColor (int color)
- void setTextSize (float size)
- void setValueWidthPercent (float percentFromTotalWidth)
- void setCustomText (String[] custom)
- void setFormatDigits (int digits)
- void setDimAlpha (int alpha)
- void setPaint (int which, Paint p)
- void setStepSize (float stepsize)
- float getStepSize ()
- PointF getCenter ()
- void setTouchEnabled (boolean enabled)
- boolean isTouchEnabled ()
- void setSelectionListener (SelectionListener l)
- boolean onTouchEvent (MotionEvent e)
- boolean onSingleTapUp (MotionEvent e)
- float getAngleForPoint (float x, float y)
- float getAngleForValue (float value)
- float getValueForAngle (float angle)
- float distanceToCenter (float x, float y)
- boolean onDown (MotionEvent e)
- boolean onFling (MotionEvent e1, MotionEvent e2, float velocityX, float velocityY)
- void onLongPress (MotionEvent e)
- boolean onScroll (MotionEvent e1, MotionEvent e2, float distanceX, float distanceY)
- void onShowPress (MotionEvent e)

Static Public Attributes

- static final int PAINT_TEXT = 1
- static final int PAINT_ARC = 2
- static final int PAINT_INNER = 3

Protected Member Functions

- void onDraw (Canvas canvas)

Private Member Functions

- void init ()
- void drawText (Canvas c)
- void drawCustomText (Canvas c)
- void drawWholeCircle (Canvas c)
- void drawInnerCircle (Canvas c)
- void drawValue (Canvas c)
- void setupBox ()
- float calcAngle (float percent)
- void updateValue (float x, float y)

Private Attributes

- String mUnit = "%"
- float mStartAngle = 270f
- float mStepSize = 1f
- float mAngle = 0f
- float mPhase = 0f
- float mValue = 0f
- float mMaxValue = 0f
- float mValueWidthPercent = 50f
- boolean mDrawInner = true
- boolean mDrawText = true
- boolean mTouchEnabled = true
- int mDimAlpha = 80
- DecimalFormat mFormatValue = new DecimalFormat("###,###,###,##0.0")
- String[] mCustomText = null
- RectF mCircleBox = new RectF()
- Paint mArcPaint
- Paint mInnerCirclePaint
- Paint mTextPaint
- ObjectAnimator mDrawAnimator
- boolean mBoxSetup = false
- SelectionListener mListener
- GestureDetector mGestureDetector

Static Private Attributes

- static final String LOG_TAG = "CircleDisplay"

6.16.1 Detailed Description

This class provides Simple custom-view for displaying values (with and without animation) and selecting values onTouch().

Author

Philipp Jahoda

6.16.2 Constructor & Destructor Documentation

6.16.2.1 cs.nsense.CircleDisplay.CircleDisplay (Context *context*)

This method construct the CircleDisplay class

Parameters

context	Interface to global information about an application environment

6.16.2.2 cs.nsense.CircleDisplay.CircleDisplay (Context *context,* AttributeSet *attrs*)

This method construct the CircleDisplay class

Parameters

context	Interface to global information about an application environment
atts	A collection of attributes, as found associated with a tag in an XML document

6.16.2.3 cs.nsense.CircleDisplay.CircleDisplay (Context *context,* AttributeSet *attrs,* int *defStyleAttr*)

This method construct the CircleDisplay class

Parameters

context	Interface to global information about an application environment
atts	A collection of attributes, as found associated with a tag in an XML document
defStyleAttr	different styles when retrieving attribute values

6.16.3 Member Function Documentation

6.16.3.1 float cs.nsense.CircleDisplay.calcAngle (float *percent*) [private]

This method calculates the needed angle for a given value

Parameters

percent	Input as percentage

Returns

percent/100f∗360f Output as percentage

6.16.3.2 float cs.nsense.CircleDisplay.distanceToCenter (float *x,* float *y*)

This method returns the distance of a certain point on the view to the center of the view

Parameters

x	point
y	point

Returns

dist distance of a certain point on the view

pythagoras

6.16.3.3 void cs.nsense.CircleDisplay.drawCustomText (Canvas *c*) [private]

This method draws the custom text in the center of the view

Parameters

c	canvas

6.16.3.4 void cs.nsense.CircleDisplay.drawInnerCircle (Canvas *c*) [private]

This method draws the inner circle of the view

Parameters

c	canvas

6.16.3.5 void cs.nsense.CircleDisplay.drawText (Canvas *c*) [private]

This method draws the text in the center of the view

Parameters

c	canvas

6.16.3.6 void cs.nsense.CircleDisplay.drawValue (Canvas *c*) [private]

This method draws the actual value slice/arc

Parameters

c	Canvas

6.16.3.7 void cs.nsense.CircleDisplay.drawWholeCircle (Canvas *c*) [private]

This method draws the background circle with less alpha

Parameters

c	canvas

6.16.3.8 float cs.nsense.CircleDisplay.getAngleForPoint (float *x*, float *y*)

This method returns the angle relative to the view center for the given point on the chart in degrees. The angle is always between 0 and 360°, 0° is NORTH

Parameters

x	point
y	point

Returns

the angle relative to the view center for the given point

neutralize overflow

6.16.3.9 float cs.nsense.CircleDisplay.getAngleForValue (float *value*)

This method returns the angle representing the given value

Parameters

value	representing the value

Returns

value/mMaxValue∗360f angle

6.16.3.10 PointF cs.nsense.CircleDisplay.getCenter ( )

This method returns the center point of the view in pixels

Returns

PointF(float x, float y) the width of view in pixels

6.16.3.11 float cs.nsense.CircleDisplay.getDiameter ( )

This method returns the diameter of the drawn circle/arc

Returns

Math.min(getWidth(),getHeight()) the diameter of the drawn circle/arc

6.16.3.12 float cs.nsense.CircleDisplay.getPhase ( )

This method returns the current animation status of the view

Returns

mPhase phase of the view

6.16.3.13 float cs.nsense.CircleDisplay.getRadius ( )

This method returns the radius of the drawn circle

Returns

getDiameter()/2f the radius of the drawn circle

6.16.3.14 float cs.nsense.CircleDisplay.getStepSize ( )

This method returns the current stepsize of the display, default 1f

Returns

mStepSize get the step size

6.16.3.15 float cs.nsense.CircleDisplay.getValue ( )

Returns the currently displayed value from the view. Depending on the used method to show the value, this value can be percent or actual value.

Returns

mValue value can be percent or actual value

6.16.3.16 float cs.nsense.CircleDisplay.getValueForAngle (float *angle*)

This method returns the value representing the given angle

Parameters

angle	input value

Returns

angle/360f∗mMaxValue value representing the given angle

6.16.3.17 void cs.nsense.CircleDisplay.init ( ) [private]

This method initialize the function of Arc, InnerCircle, Text Paint to set the style, color and also Initialize Drawing Animator function to calculated elapsed fraction of the animator

6.16.3.18 boolean cs.nsense.CircleDisplay.isDrawInnerCircleEnabled ( )

This method returns true if drawing the inner circle is enabled, false if not

Returns

mDrawInner drawing the inner circle

6.16.3.19 boolean cs.nsense.CircleDisplay.isDrawTextEnabled ( )

This method returns true if drawing the text in the center is enabled

Returns

mDrawText true to drawing of the text

6.16.3.20 boolean cs.nsense.CircleDisplay.isTouchEnabled ( )

This method returns true if touch-gestures are enabled, false if not

Returns

mTouchEnabled true/false if touch-gestures are enabled/or not

6.16.3.21 boolean cs.nsense.CircleDisplay.onDown (MotionEvent *e*)

6.16.3.22 void cs.nsense.CircleDisplay.onDraw (Canvas *canvas*) [protected]

6.16.3.23 boolean cs.nsense.CircleDisplay.onFling (MotionEvent *e1,* MotionEvent *e2,* float *velocityX,* float *velocityY*)

6.16.3.24 void cs.nsense.CircleDisplay.onLongPress (MotionEvent *e*)

6.16.3.25 boolean cs.nsense.CircleDisplay.onScroll (MotionEvent *e1,* MotionEvent *e2,* float *distanceX,* float *distanceY*)

6.16.3.26 void cs.nsense.CircleDisplay.onShowPress (MotionEvent *e*)

6.16.3.27 boolean cs.nsense.CircleDisplay.onSingleTapUp (MotionEvent *e*)

get the distance from the touch to the center of the view

touch gestures only work when touches are made exactly on the bar/arc

6.16.3.28 boolean cs.nsense.CircleDisplay.onTouchEvent (MotionEvent *e*)

if the detector recognized a gesture, consume it

get the distance from the touch to the center of the view

touch gestures only work when touches are made exactly on the bar/arc

6.16.3.29 void cs.nsense.CircleDisplay.setAnimDuration (int *durationmillis*)

This method set the duration of the drawing animation in milliseconds

Parameters

durationmillis	Duration in milli seconds

6.16.3.30 void cs.nsense.CircleDisplay.setColor (int *color*)

This method set the color of the arc

Parameters

color	color of the arc

6.16.3.31 void cs.nsense.CircleDisplay.setCustomText (String[ ] *custom*)

This method Set an array of custom texts to be drawn instead of the value in the center of the CircleDisplay. If set to null, the custom text will be reset and the value will be drawn. Make sure the length of the array corresponds with the maximum number of steps (set with setStepSize(float stepsize).

Parameters

custom	custom texts to be drawn instead of the value

6.16.3.32 void cs.nsense.CircleDisplay.setDimAlpha (int *alpha*)

This method set the aplha value to be used for the remainder of the arc, default 80 (use value between 0 and 255)

Parameters

alpha	value to be used for the remainder of the arc

6.16.3.33 void cs.nsense.CircleDisplay.setDrawInnerCircle (boolean *enabled*)

This method set this to true to draw the inner circle, default: true

Parameters

enabled	true to draw the inner circle

6.16.3.34 void cs.nsense.CircleDisplay.setDrawText (boolean *enabled*)

This method set the drawing of the center text to be enabled or not

Parameters

enabled	true to drawing of the center text

6.16.3.35 void cs.nsense.CircleDisplay.setFormatDigits (int *digits*)

This method sets the number of digits used to format values

Parameters

digits	number of digits used to format values

6.16.3.36 void cs.nsense.CircleDisplay.setPaint (int *which,* Paint *p*)

This method sets the given paint object to be used instead of the original/default one

Parameters

which	e.g.CircleDisplay.PAINT_TEXT to set a new text paint
p	paint colour

6.16.3.37 void cs.nsense.CircleDisplay.setPhase (float *phase*)

DONT USE THIS METHOD

Parameters

phase	phase of the view

6.16.3.38 void cs.nsense.CircleDisplay.setSelectionListener (SelectionListener *l*)

This method set a selection listener for the circle-display that is called whenever a value is selected onTouch()

Parameters

l	selection listener for the circle-display

6.16.3.39 void cs.nsense.CircleDisplay.setStartAngle (float *angle*)

This method set the starting angle for the view

Parameters

angle	input of angle

6.16.3.40 void cs.nsense.CircleDisplay.setStepSize (float *stepsize*)

This method Sets the stepsize (minimum selection interval) of the circle display, default 1f. It is recommended to make this value not higher than 1/5 of the maximum selectable value, and not lower than 1/200 of the maximum selectable value. For a maximum value of 100 for example, a stepsize between 0.5 and 20 is recommended.

Parameters

stepsize	setting minimum selection interval

6.16.3.41 void cs.nsense.CircleDisplay.setTextSize (float *size*)

This method set the size of the center text in dp

Parameters

size	size of the center text

6.16.3.42 void cs.nsense.CircleDisplay.setTouchEnabled (boolean *enabled*)

Enable touch gestures on the circle-display. If enabled, selecting values onTouch() is possible. Set a SelectionListener to retrieve selected values. Do not forget to set a value before selecting values. By default the maxvalue is 0f and therefore nothing can be selected.

Parameters

enabled	selecting values onTouch() is possible

6.16.3.43 void cs.nsense.CircleDisplay.setUnit (String *unit*)

This method Sets the unit that is displayed next to the value in the center of the view. Default "%". Could be "€" or "$" or left blank or whatever it is you display.

Parameters

unit	displayed next to the value in the center of the view

6.16.3.44 void cs.nsense.CircleDisplay.setupBox ( ) [private]

This method up the bounds of the view

6.16.3.45 void cs.nsense.CircleDisplay.setValueWidthPercent (float *percentFromTotalWidth*)

This method set the thickness of the value bar, default 50%

Parameters

percentFromTotalWidth	percentage to thickness of the value bar

6.16.3.46 void cs.nsense.CircleDisplay.showValue (float *toShow,* float *total,* boolean *animated*)

This method shows the given value in the circle view

Parameters

toShow	to be show

total	Total
animated	Animation

6.16.3.47 void cs.nsense.CircleDisplay.startAnim ( )

6.16.3.48 void cs.nsense.CircleDisplay.updateValue (float *x*, float *y*) [private]

This method updates the display with the given touch position, takes stepsize into consideration

Parameters

x	point
y	point

calculate the touch-angle

calculate the new value depending on angle

if no stepsize

check if the new value is closer to the next, or the previous

set the new values

6.16.4 Member Data Documentation

6.16.4.1 final String cs.nsense.CircleDisplay.LOG_TAG = "CircleDisplay" [static], [private]

6.16.4.2 float cs.nsense.CircleDisplay.mAngle = 0f [private]

angle that represents the displayed value

6.16.4.3 Paint cs.nsense.CircleDisplay.mArcPaint [private]

6.16.4.4 boolean cs.nsense.CircleDisplay.mBoxSetup = false [private]

boolean flag that indicates if the box has been setup

6.16.4.5 RectF cs.nsense.CircleDisplay.mCircleBox = new RectF() [private]

rect object that represents the bounds of the view, needed for drawing the circle

6.16.4.6 String [ ] cs.nsense.CircleDisplay.mCustomText = null [private]

array that contains values for the custom-text

6.16.4.7 int cs.nsense.CircleDisplay.mDimAlpha = 80 [private]

represents the alpha value used for the remainder bar

6.16.4.8 ObjectAnimator cs.nsense.CircleDisplay.mDrawAnimator [private]

object animator for doing the drawing animations

6.16.4.9 boolean cs.nsense.CircleDisplay.mDrawInner = true [private]

if enabled, the inner circle is drawn

6.16.4.10 boolean cs.nsense.CircleDisplay.mDrawText = true [private]

if enabled, the center text is drawn

6.16.4.11 DecimalFormat cs.nsense.CircleDisplay.mFormatValue = new DecimalFormat("###,###,###,##0.0") [private]

the decimalformat responsible for formatting the values in the view

6.16.4.12 GestureDetector cs.nsense.CircleDisplay.mGestureDetector [private]

gesturedetector for recognizing single-taps

6.16.4.13 Paint cs.nsense.CircleDisplay.mInnerCirclePaint [private]

6.16.4.14 SelectionListener cs.nsense.CircleDisplay.mListener [private]

listener called when a value has been selected on touch

6.16.4.15 float cs.nsense.CircleDisplay.mMaxValue = 0f [private]

the maximum displayable value, depends on the set value

6.16.4.16 float cs.nsense.CircleDisplay.mPhase = 0f [private]

current state of the animation

6.16.4.17 float cs.nsense.CircleDisplay.mStartAngle = 270f [private]

startangle of the view

6.16.4.18 float cs.nsense.CircleDisplay.mStepSize = 1f [private]

field representing the minimum selectable value in the display - the minimum interval

6.16.4.19 Paint cs.nsense.CircleDisplay.mTextPaint [private]

6.16.4.20 boolean cs.nsense.CircleDisplay.mTouchEnabled = true [private]

if enabled, touching and therefore selecting values is enabled

6.16.4.21 String cs.nsense.CircleDisplay.mUnit = "%" [private]

the unit that is represented by the circle-display

6.16.4.22 float cs.nsense.CircleDisplay.mValue = 0f [private]

the currently displayed value, can be percent or actual value

6.16.4.23 float cs.nsense.CircleDisplay.mValueWidthPercent = 50f [private]

percent of the maximum width the arc takes

6.16.4.24 final int cs.nsense.CircleDisplay.PAINT_ARC = 2 [static]

paint representing the value bar

6.16.4.25 final int cs.nsense.CircleDisplay.PAINT_INNER = 3 [static]

paint representing the inner (by default white) area

6.16.4.26 final int cs.nsense.CircleDisplay.PAINT_TEXT = 1 [static]

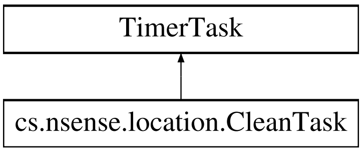
paint used for drawing the text

The documentation for this class was generated from the following file:

- src/cs/nsense/CircleDisplay.java

6.17 cs.nsense.location.CleanTask Class Reference

Inheritance diagram for cs.nsense.location.CleanTask:



Public Member Functions

- CleanTask (NSenseDataSource dataSource, LocationPipeline callback)
- void run ()

Private Attributes

- String TAG = "LocationPipeline"
- NSenseDataSource dataSource
- LocationPipeline callback
- int timeout = 360000
- Handler notifyDataBaseChangeHandler

6.17.1 Constructor & Destructor Documentation

6.17.1.1 cs.nsense.location.CleanTask.CleanTask (**NSenseDataSource** *dataSource,* **LocationPipeline** *callback*)

Clean Task constructor

Parameters

dataSource	the NSense data base
callback	the Location Pipeline

6.17.2 Member Function Documentation

6.17.2.1 void cs.nsense.location.CleanTask.run ( )

Thread core - It checks the last update in every location table entry.

6.17.3 Member Data Documentation

6.17.3.1 LocationPipeline cs.nsense.location.CleanTask.callback [private]

Location Pipeline class. Used to notify the location pipeline about a change in DB.

6.17.3.2 NSenseDataSource cs.nsense.location.CleanTask.dataSource [private]

NSense Database

6.17.3.3 Handler cs.nsense.location.CleanTask.notifyDataBaseChangeHandler [private]

Initial value:

```
= new Handler() {
          public void handleMessage(Message msg) {
              super.handleMessage(msg);
              callback.notifyDataBaseChange();
          }
      }
```

Handler used to notify the callback about a DB change.

6.17.3.4 String cs.nsense.location.CleanTask.TAG = "LocationPipeline" [private]

6.17.3.5 int cs.nsense.location.CleanTask.timeout = 360000 [private]

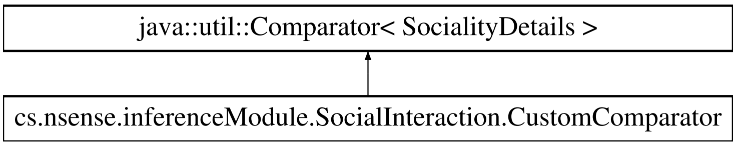
Timeout for a location entry to be removed.

The documentation for this class was generated from the following file:

- src/cs/nsense/location/CleanTask.java

6.18 cs.nsense.inferenceModule.SocialInteraction.CustomComparator Class Reference

Inheritance diagram for cs.nsense.inferenceModule.SocialInteraction.CustomComparator:


java::util::Comparator< SocialityDetails >
cs.nsense.inferenceModule.SocialInteraction.CustomComparator

Public Member Functions

- int compare (SocialityDetails entry1, SocialityDetails entry2)

6.18.1 Detailed Description

This method sorting the propinquity from the sociality details list

6.18.2 Member Function Documentation

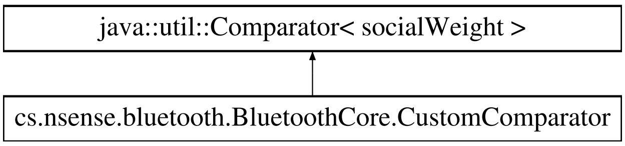
6.18.2.1 int cs.nsense.inferenceModule.SocialInteraction.CustomComparator.compare (SocialityDetails *entry1,* SocialityDetails *entry2*)

The documentation for this class was generated from the following file:

- src/cs/nsense/inferenceModule/SocialInteraction.java

6.19 cs.nsense.bluetooth.BluetoothCore.CustomComparator Class Reference

Inheritance diagram for cs.nsense.bluetooth.BluetoothCore.CustomComparator:


java::util::Comparator< socialWeight >
cs.nsense.bluetooth.BluetoothCore.CustomComparator

Public Member Functions

- int compare (socialWeight entry1, socialWeight entry2)

6.19.1 Detailed Description

This class sorts the devices by social weight.

6.19.2 Member Function Documentation

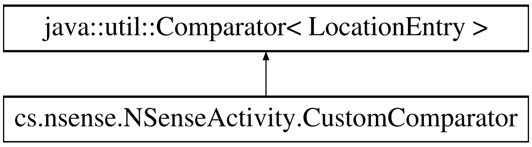
6.19.2.1 int cs.nsense.bluetooth.BluetoothCore.CustomComparator.compare (socialWeight *entry1,* socialWeight *entry2*)

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/BluetoothCore.java

6.20 cs.nsense.NSenseActivity.CustomComparator Class Reference

Inheritance diagram for cs.nsense.NSenseActivity.CustomComparator:


java::util::Comparator< LocationEntry >
cs.nsense.NSenseActivity.CustomComparator

Public Member Functions

- int compare (LocationEntry entry1, LocationEntry entry2)

6.20.1 Detailed Description

This method compare the map of location entries based on the sorted distance

6.20.2 Member Function Documentation

6.20.2.1 int cs.nsense.NSenseActivity.CustomComparator.compare (**LocationEntry** *entry1,* **LocationEntry** *entry2*)

The documentation for this class was generated from the following file:

- src/cs/nsense/NSenseActivity.java

6.21 cs.nsense.db.DataBaseChangeListener Interface Reference

Public Member Functions

- void onDataBaseChangeAcc (List< ActionsEntry > actionsEntries)
- void onDataBaseChangeBT (ArrayList< BTUserDevice > arrayList)
- void onDataBaseChangeBTEncDur (ArrayList< BTUserDevEncounterDuration > arrayList)
- void onDataBaseChangeBTAvgEncDur (ArrayList< BTUserDevAverageEncounterDuration > arrayList)
- void onDataBaseChangeBTSocialWeight (ArrayList< BTUserDevSocialWeight > arrayList)
- void onDataBaseChangeLocation (List< LocationEntry > mLocationEntries)
- void onSoundLevelChange (ArrayList< SoundLevel > arrayList)

6.21.1 Detailed Description

This class provides list of interfaces to construct the database structure

6.21.2 Member Function Documentation

6.21.2.1 void cs.nsense.db.DataBaseChangeListener.onDataBaseChangeAcc (List< **ActionsEntry** > *actionsEntries*)

This method called when ActionEntry table has been changed in the database

Parameters

actionsEntries	List of ActionsEntry objects

6.21.2.2 void cs.nsense.db.DataBaseChangeListener.onDataBaseChangeBT (ArrayList< **BTUserDevice** > *arrayList*)

This method called when BTUserDevice table has been changed in the database

Parameters

arrayList	List of BTUserDevice objects

6.21.2.3 void cs.nsense.db.DataBaseChangeListener.onDataBaseChangeBTAvgEncDur (ArrayList< BTUserDevAverageEncounterDuration > *arrayList*)

This method called when BTUserDevAverageEncounterDuration table has been changed in the database

Parameters

arrayList	List of BTUserDevAverageEncounterDuration objects

6.21.2.4 void cs.nsense.db.DataBaseChangeListener.onDataBaseChangeBTEncDur (ArrayList< BTUserDevEncounterDuration > *arrayList*)

This method called when BTUserDevEncounterDuration table has been changed in the database

Parameters

arrayList	List of BTUserDevEncounterDuration objects

6.21.2.5 void cs.nsense.db.DataBaseChangeListener.onDataBaseChangeBTSocialWeight (ArrayList< BTUserDevSocialWeight > *arrayList*)

This method called when BTUserDevSocialWeight table has been changed in the database

Parameters

arrayList	List of BTUserDevSocialWeight objects

6.21.2.6 void cs.nsense.db.DataBaseChangeListener.onDataBaseChangeLocation (List< LocationEntry > *mLocationEntries*)

This method called when LocationEntry table has been changed in the database

Parameters

mLocationEntries	List of LocationEntry objects

6.21.2.7 void cs.nsense.db.DataBaseChangeListener.onSoundLevelChange (ArrayList< SoundLevel > *arrayList*)

This method called when SoundLevel table has been changed in the database

Parameters

mLocationEntries	List of SoundLevel objects

The documentation for this interface was generated from the following file:

- src/cs/nsense/db/DataBaseChangeListener.java

6.22 cs.nsense.inferenceModule.SocialInteraction.DeviceDetails Class Reference

Public Member Functions

- String getmMACAddress ()
- void setmMACAddress (String mMACAddress)
- String getDevName ()
- void setDevName (String devName)
- double getmDistance ()
- void setmDistance (double mDistance)
- double getmSW ()
- void setmSW (double mSW)
- double getmSI ()
- void setmSI (double mSI)
- double getmPropinquity ()
- void setmPropinquity (double mPropinquity)

Public Attributes

- String mMACAddress
- String devName
- double mDistance
- double mSW
- double mSI
- double mPropinquity

6.22.1 Detailed Description

This method provides the getter and setter methods of device details

6.22.2 Member Function Documentation

6.22.2.1 String cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.getDevName ( )

This method get the device name

Returns

devName the device name

6.22.2.2 double cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.getmDistance ( )

This method get the distance

Returns

mDistance the distance

6.22.2.3 String cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.getmMACAddress ( )

This method get the MAC address

Returns

mMACAddress the MAC address

6.22.2.4 double cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.getmPropinquity ( )

This method get the propinquity

Returns

mPropinquity the propinquity

6.22.2.5 double cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.getmSI ( )

This method get the social interaction

Returns

mSI the Social interaction

6.22.2.6 double cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.getmSW ( )

This method get the social weight

Returns

mSW the social weight

6.22.2.7 void cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.setDevName (String *devName*)

This method set the device name

Parameters

devName	the device name

6.22.2.8 void cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.setmDistance (double *mDistance*)

This method set the distance

Parameters

mDistance	the distance

6.22.2.9 void cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.setmMACAddress (String *mMACAddress*)

This method set the MAC address

Parameters

mMACAddress	the MAC address

6.22.2.10 void cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.setmPropinquity (double *mPropinquity*)

This method set the propinquity

Parameters

mPropinquity	the propinquity

6.22.2.11 void cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.setmSI (double *mSI*)

This method set the social interaction

Parameters

mSI	the Social interaction

6.22.2.12 void cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.setmSW (double *mSW*)

This method set the social weight

Parameters

mSW	the social weight

6.22.3 Member Data Documentation

6.22.3.1 String cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.devName

6.22.3.2 double cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.mDistance

6.22.3.3 String cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.mMACAddress

6.22.3.4 double cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.mPropinquity

6.22.3.5 double cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.mSI

6.22.3.6 double cs.nsense.inferenceModule.SocialInteraction.DeviceDetails.mSW

The documentation for this class was generated from the following file:

- src/cs/nsense/inferenceModule/SocialInteraction.java

6.23 com.meapsoft.FFT Class Reference

Public Member Functions

- FFT (int n)
- double[] getWindow ()
- void fft (double[] x, double[] y)

Static Public Member Functions

- static void main (String[] args)
- static void beforeAfter (FFT fft, double[] re, double[] im)
- static void printReIm (double[] re, double[] im)

Protected Member Functions

- void makeWindow ()

6.23.1 Detailed Description

Utility class to perform a fast fourier transform without allocating any extra memory.

Author

Mike Mandel (mim@ee.columbia.edu)

6.23.2 Constructor & Destructor Documentation

6.23.2.1 com.meapsoft.FFT.FFT (int *n*)

Make sure n is a power of 2

precompute tables

6.23.3 Member Function Documentation

6.23.3.1 static void com.meapsoft.FFT.beforeAfter (FFT *fft*, double[] *re*, double[] *im*) [static]

6.23.3.2 void com.meapsoft.FFT.fft (double[] *x*, double[] *y*)

fft.c Douglas L. Jones University of Illinois at Urbana-Champaign January 19, 1992 http://cnx.rice.edu/content/m12016/latest/

fft: in-place radix-2 DIT DFT of a complex input

input: n: length of FFT: must be a power of two m: n = 2∗∗m input/output x: double array of length n with real part of data y: double array of length n with imag part of data

Permission to copy and use this program is granted as long as this header is included. Bit-reverse

FFT

6.23.3.3 double [] com.meapsoft.FFT.getWindow ( )

6.23.3.4 static void com.meapsoft.FFT.main (String[] *args*) [static]

Test the FFT to make sure it's working Impulse

I Nyquist

Single sin

Ramp

6.23.3.5 void com.meapsoft.FFT.makeWindow ( ) [protected]

Make a blackman window: w(n)=0.42-0.5cos{(2∗PI∗n)/(N-1)}+0.08cos{(4∗PI∗n)/(N-1)};

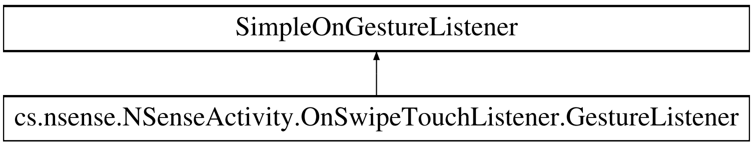
6.23.3.6 static void com.meapsoft.FFT.printReIm (double[ ] *re,* double[ ] *im*) [static]

The documentation for this class was generated from the following file:

- src/com/meapsoft/FFT.java

6.24 cs.nsense.NSenseActivity.OnSwipeTouchListener.GestureListener Class Reference

Inheritance diagram for cs.nsense.NSenseActivity.OnSwipeTouchListener.GestureListener:


SimpleOnGestureListener
cs.nsense.NSenseActivity.OnSwipeTouchListener.GestureListener

Public Member Functions

- boolean onDown (MotionEvent e)
- boolean onFling (MotionEvent e1, MotionEvent e2, float velocityX, float velocityY)

Static Private Attributes

- static final int SWIPE_THRESHOLD = 100
- static final int SWIPE_VELOCITY_THRESHOLD = 100

6.24.1 Member Function Documentation

6.24.1.1 boolean cs.nsense.NSenseActivity.OnSwipeTouchListener.GestureListener.onDown (MotionEvent *e*)

6.24.1.2 boolean cs.nsense.NSenseActivity.OnSwipeTouchListener.GestureListener.onFling (MotionEvent *e1,* MotionEvent *e2,* float *velocityX,* float *velocityY*)

6.24.2 Member Data Documentation

6.24.2.1 final int cs.nsense.NSenseActivity.OnSwipeTouchListener.GestureListener.SWIPE_THRESHOLD = 100 [static], [private]

6.24.2.2 final int cs.nsense.NSenseActivity.OnSwipeTouchListener.GestureListener.SWIPE_VELOCITY_THRESHOLD = 100 [static], [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/NSenseActivity.java

6.25 cs.nsense.accelerometer.Globals Class Reference

Static Public Attributes

- static final int ACCELEROMETER_BUFFER_CAPACITY = 2048
- static final int ACCELEROMETER_BLOCK_CAPACITY = 64
- static final String ACTIVITY_TYPES []

6.25.1 Detailed Description

This class provides global variables

6.25.2 Member Data Documentation

6.25.2.1 final int cs.nsense.accelerometer.Globals.ACCELEROMETER_BLOCK_CAPACITY = 64 [static]

Accelerometer block capacity of Globals object

6.25.2.2 final int cs.nsense.accelerometer.Globals.ACCELEROMETER_BUFFER_CAPACITY = 2048 [static]

Accelerometer buffer capacity of Globals object

6.25.2.3 final String cs.nsense.accelerometer.Globals.ACTIVITY_TYPES[] [static]

Initial value:

```
= {
        "Standing", "Walking", "Running",
    }
```

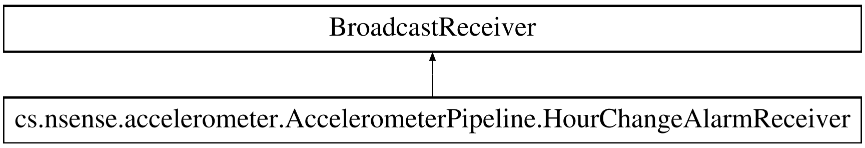
Activity types of Globals object

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/Globals.java

6.26 cs.nsense.accelerometer.AccelerometerPipeline.HourChangeAlarmReceiver Class Reference

Inheritance diagram for cs.nsense.accelerometer.AccelerometerPipeline.HourChangeAlarmReceiver:


BroadcastReceiver
cs.nsense.accelerometer.AccelerometerPipeline.HourChangeAlarmReceiver

Public Member Functions

- void onReceive (Context context, Intent intent)

Static Public Attributes

- static final String HOUR_CHANGE = "android.intent.action.HOUR_CHANGE"

6.26.1 Detailed Description

This method provides Hour Change Alarm Receiver

6.26.2 Member Function Documentation

6.26.2.1 void cs.nsense.accelerometer.AccelerometerPipeline.HourChangeAlarmReceiver.onReceive (Context *context,* Intent *intent*)

6.26.3 Member Data Documentation

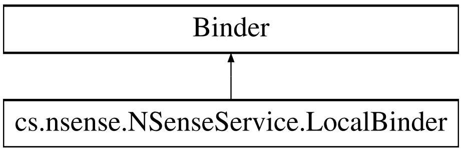
6.26.3.1 final String cs.nsense.accelerometer.AccelerometerPipeline.HourChangeAlarmReceiver.HOUR_CHANGE = "android.intent.action.HOUR_CHANGE" [static]

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/AccelerometerPipeline.java

6.27 cs.nsense.NSenseService.LocalBinder Class Reference

Inheritance diagram for cs.nsense.NSenseService.LocalBinder:



6.27.1 Detailed Description

This class binds the NSense service

The documentation for this class was generated from the following file:

- src/cs/nsense/NSenseService.java

6.28 cs.nsense.location.LocationEntry Class Reference

Public Member Functions

- String getDeviceName ()
- String getBSSID ()
- double getDistance ()
- long getLastUpdate ()
- void setDeviceName (String mDeviceName)
- void setBSSID (String bSSID)
- void setDistance (double mDistance)

- void setLastUpdate (long mLastUpdate)
- LocationEntry ()
- String toString ()

Private Attributes

- String BSSID
- String mDeviceName
- double mDistance
- long mLastUpdate

6.28.1 Constructor & Destructor Documentation

6.28.1.1 cs.nsense.location.LocationEntry.LocationEntry ( )

Location Entry Constructor

6.28.2 Member Function Documentation

6.28.2.1 String cs.nsense.location.LocationEntry.getBSSID ( )

Get the BSSID

Returns

the BSSID - The device's MAC address.

6.28.2.2 String cs.nsense.location.LocationEntry.getDeviceName ( )

Get the device name

Returns

mDeviceName - The device name.

6.28.2.3 double cs.nsense.location.LocationEntry.getDistance ( )

Get the distance to this device

Returns

mDistance - Distance to this device.

6.28.2.4 long cs.nsense.location.LocationEntry.getLastUpdate ( )

Get the time of last update

Returns

mLastUpdate - Last time this entry was updated

6.28.2.5 void cs.nsense.location.LocationEntry.setBSSID (String *bSSID*)

Set the BSSID of this AP

Parameters

bSSID	the MAC address to set

6.28.2.6 void cs.nsense.location.LocationEntry.setDeviceName (String *mDeviceName*)

Set the device name

Parameters

device	name

6.28.2.7 void cs.nsense.location.LocationEntry.setDistance (double *mDistance*)

Set the distance to this device

Parameters

mDistance	distance to this device

6.28.2.8 void cs.nsense.location.LocationEntry.setLastUpdate (long *mLastUpdate*)

Set the last update time

Parameters

mLastUpdate	last update time

6.28.2.9 String cs.nsense.location.LocationEntry.toString ( )

Return a string containing the SSID, BSSID, Distance and the last update.

6.28.3 Member Data Documentation

6.28.3.1 String cs.nsense.location.LocationEntry.BSSID [private]

Device MAC address

6.28.3.2 String cs.nsense.location.LocationEntry.mDeviceName [private]

Device Name

6.28.3.3 double cs.nsense.location.LocationEntry.mDistance [private]

Distance to this device

6.28.3.4 long cs.nsense.location.LocationEntry.mLastUpdate [private]

Last time this entry was updated

The documentation for this class was generated from the following file:

- src/cs/nsense/location/LocationEntry.java

6.29 cs.nsense.location.LocationPipeline Class Reference

Public Member Functions

- LocationPipeline (NSenseService callback, NSenseDataSource dataSource)
- void close ()
- void notifyDataBaseChange ()

Private Attributes

- Context aContext = null
- NSenseService callback = null
- NSenseDataSource dataSource

Static Private Attributes

- static
  RelativePositionWiFiNoConnection mRelativePositionWiFiNoConnection = null

6.29.1 Constructor & Destructor Documentation

6.29.1.1 cs.nsense.location.LocationPipeline.LocationPipeline (**NSenseService** *callback,* **NSenseDataSource** *dataSource*)

Location Pipeline constructor - It initializes the Relative Position module and a Timer to remove old entries

Parameters

callback	NSense Service module
dataSource	NSense Data base

6.29.2 Member Function Documentation

6.29.2.1 void cs.nsense.location.LocationPipeline.close ( )

Close the Location Pipeline Calling this function will close Location Pipeline in your app.

6.29.2.2 void cs.nsense.location.LocationPipeline.notifyDataBaseChange ( )

Notifies a database change to the listeners.

6.29.3 Member Data Documentation

6.29.3.1 Context cs.nsense.location.LocationPipeline.aContext = null [private]

Interface to global information about an application environment.

6.29.3.2 NSenseService cs.nsense.location.LocationPipeline.callback = null [private]

NSense Service module

6.29.3.3 NSenseDataSource cs.nsense.location.LocationPipeline.dataSource [private]

NSense Data base

6.29.3.4 RelativePositionWiFiNoConnection cs.nsense.location.LocationPipeline.mRelativePositionWiFiNoConnection = null [static], [private]

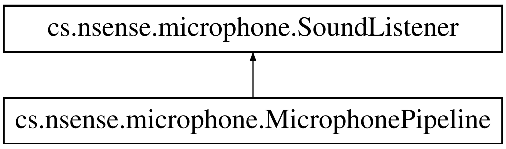
Relative Position module - manage the WiFi P2P to compute the relative distance

The documentation for this class was generated from the following file:

- src/cs/nsense/location/LocationPipeline.java

6.30 cs.nsense.microphone.MicrophonePipeline Class Reference

Inheritance diagram for cs.nsense.microphone.MicrophonePipeline:


cs.nsense.microphone.SoundListener
cs.nsense.microphone.MicrophonePipeline

Public Member Functions

- MicrophonePipeline (NSenseService callback, NSenseDataSource dataSource)
- void close (Context contect)
- void updateSoundLevel (double soundDB)
- String getDuration (long milliseconds)
- void updateSoundLevel (SoundLevel soundLevel)

Static Public Attributes

- static String mEnvironmentalSound = ""

Protected Member Functions

- List< SoundLevel > getData ()

Protected Attributes

- double avgdB
- double soundDB

Private Member Functions

- long getTodayMillis ()
- void start ()
- void notifyDataBaseChange ()
- void Soundstorage (String envSound, String date)

Private Attributes

- SoundManager soundManager
- ArrayList< DataBaseChangeListener > listeners = new ArrayList<DataBaseChangeListener> ()
- boolean micPresent =true
- SoundLevel soundlevel
- Handler mHandler = new Handler()
- NSenseService scallback = null
- String envSound = ""

Static Private Attributes

- static SoundLevel mSoundLevel = new SoundLevel()

6.30.1 Detailed Description

This class is contains the core functionalities of the application relating to Microphone.

6.30.2 Constructor & Destructor Documentation

6.30.2.1 cs.nsense.microphone.MicrophonePipeline.MicrophonePipeline (NSenseService *callback,* NSenseDataSource *dataSource*)

This method constructs the MicrophonePipeline

Parameters

callback	Supply NSenseService object for NSenseActivity to use
dataSource	NSenseDataSource to access various methods and information of the NSense Data base

6.30.3 Member Function Documentation

6.30.3.1 void cs.nsense.microphone.MicrophonePipeline.close (Context *contect*)

This method close the context

Parameters

contect	Interface to global information about an application environment.

6.30.3.2 List<SoundLevel> cs.nsense.microphone.MicrophonePipeline.getData ( ) [protected]

This method provides the sound level data from the database

Returns

soundlevel Array list of SoundLevel objects from the database

6.30.3.3 String cs.nsense.microphone.MicrophonePipeline.getDuration (long *milliseconds*)

This method calculates the duration

Parameters

milliseconds	Time in milliseconds

Returns

the time duration in hours, minutes and seconds

6.30.3.4 long cs.nsense.microphone.MicrophonePipeline.getTodayMillis ( ) [private]

This method provides current date, hours, minutes, seconds and milliseconds of the system

Returns

current date and time of the system

6.30.3.5 void cs.nsense.microphone.MicrophonePipeline.notifyDataBaseChange ( ) [private]

This method notifies a database change to the listeners.

6.30.3.6 void cs.nsense.microphone.MicrophonePipeline.Soundstorage (String *envSound,* String *date*) [private]

This method writes a file for Microphone pipeline BufferedWriter for performance, true to set append to file flag

6.30.3.7 void cs.nsense.microphone.MicrophonePipeline.start ( ) [private]

This method initialize the sound manager class which Start Capturing sound waves from microphone adapter.

6.30.3.8 void cs.nsense.microphone.MicrophonePipeline.updateSoundLevel (double *soundDB*)

This method provides four different levels of sound and their duration Based on the captured sound decibels

Parameters

soundDB	soundLevel in decibels

Returns

Quite, Normal, Alert, Noisy and Time duration in each state

set object Date

6.30.3.9 void cs.nsense.microphone.MicrophonePipeline.updateSoundLevel (SoundLevel *soundLevel*)

This method update the sound level

Implements cs.nsense.microphone.SoundListener.

6.30.4 Member Data Documentation

6.30.4.1 double cs.nsense.microphone.MicrophonePipeline.avgdB [protected]

This variable is used to access the average decibels

6.30.4.2 String cs.nsense.microphone.MicrophonePipeline.envSound = "" [private]

This variable is used to access the environment sound

6.30.4.3 ArrayList<DataBaseChangeListener> cs.nsense.microphone.MicrophonePipeline.listeners = new ArrayList<DataBaseChangeListener> () [private]

This variable is used to access the listeners

6.30.4.4 String cs.nsense.microphone.MicrophonePipeline.mEnvironmentalSound = "" [static]

This variable is used to access Environment sound

6.30.4.5 Handler cs.nsense.microphone.MicrophonePipeline.mHandler = new Handler() [private]

This class is to use send or process message

6.30.4.6 boolean cs.nsense.microphone.MicrophonePipeline.micPresent =true [private]

This variable is used to check whether microphone is available or not

6.30.4.7 SoundLevel cs.nsense.microphone.MicrophonePipeline.mSoundLevel = new SoundLevel() [static], [private]

This class is to access Sound Level

6.30.4.8 NSenseService cs.nsense.microphone.MicrophonePipeline.scallback = null [private]

This class is to access the functionality of NSense Service

6.30.4.9 double cs.nsense.microphone.MicrophonePipeline.soundDB [protected]

This variable is used to access the sound in decibels

6.30.4.10 SoundLevel cs.nsense.microphone.MicrophonePipeline.soundlevel [private]

This variable is used to access the sound level

6.30.4.11 SoundManager cs.nsense.microphone.MicrophonePipeline.soundManager [private]

This class is to access functionality of Sensor Manager from accelerometer pipeline

The documentation for this class was generated from the following file:

- src/cs/nsense/microphone/MicrophonePipeline.java

6.31 cs.nsense.NSenseService.NOTIFICATION_ID Interface Reference

Static Public Attributes

- static int FOREGROUND_SERVICE = 112

6.31.1 Member Data Documentation

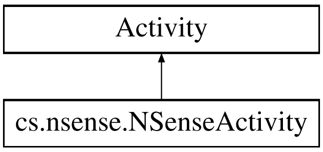
6.31.1.1 int cs.nsense.NSenseService.NOTIFICATION_ID.FOREGROUND_SERVICE = 112 [static]

The documentation for this interface was generated from the following file:

- src/cs/nsense/NSenseService.java

6.32 cs.nsense.NSenseActivity Class Reference

Inheritance diagram for cs.nsense.NSenseActivity:



Classes

- class CustomComparator
- class OnSwipeTouchListener

Public Member Functions

- void onBackPressed ()

Static Public Attributes

- static final String PKG = "cs.nsense"
- static String actionMessage = ""

Protected Member Functions

- void onCreate (Bundle savedInstanceState)
- void onDestroy ()
- void onStart ()
- void onStop ()

Private Member Functions

- boolean isMyServiceRunning (Class< NSenseService > nsenseService)
- ArrayList< SensorProduct > parseXML (XmlPullParser parser) throws XmlPullParserException, IOException

Private Attributes

- NSenseService mBoundService
- boolean mRequestToBound = false
- boolean mIsServiceBound
- String sensorname

- String sensorinterval
- ListView siListView
- SIAdapter mSIAdapter
- ServiceConnection mConnection

Static Private Attributes

- static final String TAG = "NSENSE"
- static ArrayList
  < SocialityDetails > listHistory

6.32.1 Detailed Description

This class provides the layout of NSense application and initialize the NSense Service

6.32.2 Member Function Documentation

6.32.2.1 boolean cs.nsense.NSenseActivity.isMyServiceRunning (Class< NSenseService > *nsenseService*) [private]

This method returns True/false if the NSense service is running/not

Parameters

nsenseService	NSenseService class

Returns

true/false

6.32.2.2 void cs.nsense.NSenseActivity.onBackPressed ( )

6.32.2.3 void cs.nsense.NSenseActivity.onCreate (Bundle *savedInstanceState*) [protected]

Find the ListView resource.

6.32.2.4 void cs.nsense.NSenseActivity.onDestroy ( ) [protected]

This method close the NSense service

6.32.2.5 void cs.nsense.NSenseActivity.onStart ( ) [protected]

This method start the NSense activity

6.32.2.6 void cs.nsense.NSenseActivity.onStop ( ) [protected]

This method stop the NSense activity

6.32.2.7 ArrayList<SensorProduct> cs.nsense.NSenseActivity.parseXML (XmlPullParser *parser*) throws XmlPullParserException, IOException [private]

This method parse the XML data

Parameters

parser	The Xml Pull Parser with list of sensors

Returns

products List of sensors from SensorProduct

Exceptions

XmlPullParserException	Xml Pull Parser Exception
IOException	Input, Output Exception

6.32.3 Member Data Documentation

6.32.3.1 String cs.nsense.NSenseActivity.actionMessage = "" [static]

This variable is to store the action message

6.32.3.2 ArrayList<SocialityDetails> cs.nsense.NSenseActivity.listHistory [static], [private]

This list of SocialityDetails objects to get the History list of SocialityDetails

6.32.3.3 NSenseService cs.nsense.NSenseActivity.mBoundService [private]

This class provides the NSense Service to bound service

6.32.3.4 ServiceConnection cs.nsense.NSenseActivity.mConnection [private]

This method creates the service connection, which can be used to bind the NSense service

6.32.3.5 boolean cs.nsense.NSenseActivity.mIsServiceBound [private]

This variable is request to bound service

6.32.3.6 boolean cs.nsense.NSenseActivity.mRequestToBound = false [private]

This variable is to bound request

6.32.3.7 SIAdapter cs.nsense.NSenseActivity.mSIAdapter [private]

This class is to get SocialityDetails object

6.32.3.8 final String cs.nsense.NSenseActivity.PKG = "cs.nsense" [static]

6.32.3.9 String cs.nsense.NSenseActivity.sensorinterval [private]

This variable is to provide sensor interval

6.32.3.10 String cs.nsense.NSenseActivity.sensorname [private]

This variable is to provide the sensor name

6.32.3.11 ListView cs.nsense.NSenseActivity.siListView [private]

This class is to provide ListView

6.32.3.12 final String cs.nsense.NSenseActivity.TAG = "NSENSE" [static], [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/NSenseActivity.java

6.33 cs.nsense.db.NSenseDataSource Class Reference

Public Member Functions

- NSenseDataSource (Context context)
- void openDB (boolean writable) throws SQLException
- void closeDB ()
- long getNumAP ()
- long getNumActionEntry ()
- long registerNewActionEntry (ActionsEntry ae)
- boolean updateActionEntry (ActionsEntry ae)
- ActionsEntry getActionEntry (String actionentry)
- boolean hasActionEntry (String actionType)
- Map< String, ActionsEntry > getAllActionEntry ()
- long getNumBTDevice ()
- void registerNewBTDevice (BTUserDevice btDev, BTUserDevEncounterDuration duration, BTUserDevAverageEncounterDuration averageDuration, BTUserDevSocialWeight socialWeight)
- void updateBTDeviceAndDuration (BTUserDevice btDev, BTUserDevEncounterDuration duration)
- void updateBTDevAvgEncounterDuration (BTUserDevAverageEncounterDuration averageDuration)
- void updateBTDevSocialWeight (BTUserDevSocialWeight socialWeight)
- BTUserDevice getBTDevice (String btDevice)
- BTUserDevEncounterDuration getBTDeviceEncounterDuration (String btDevEncounterDuration)
- BTUserDevAverageEncounterDuration getBTDeviceAverageEncounterDuration (String btDevAverageEncounterDuration)
- BTUserDevSocialWeight getBTDeviceSocialWeight (String btDevSocialWeight)
- boolean hasBTDevice (String btDev)
- Map< String, BTUserDevice > getAllBTDevice ()
- Map< String,
  BTUserDevEncounterDuration > getAllBTDevEncounterDuration ()
- Map< String,
  BTUserDevAverageEncounterDuration > getAllBTDevAverageEncounterDuration ()
- Map< String,
  BTUserDevSocialWeight > getAllBTDevSocialWeight ()
- long getNumEntries ()
- long registerLocationEntry (LocationEntry entry)
- boolean updateLocationEntry (LocationEntry entry)
- int removeLocationEntry (LocationEntry entry)
- LocationEntry getLocationEntry (String bssid)
- boolean hasLocationEntry (String bssid)
- Map< String, LocationEntry > getAllLocationEntries ()
- void cleanLocationTable ()
- long getNumSoundLevel ()
- long registerNewSoundLevel (SoundLevel sl)

- boolean updateSoundLevel (SoundLevel sl)
- SoundLevel getSoundLevel (long todayMillis)
- boolean hasSoundLevel (long soundDate)
- Map< String, SoundLevel > getAllSoundLevel ()

Public Attributes

- String[] allColumnsSound

Private Member Functions

- ActionsEntry cursorToActions (Cursor cursor)
- BTUserDevice cursorToBTDevice (Cursor cursor)
- BTUserDevEncounterDuration cursorToBTDevEncounterDuration (Cursor cursor)
- BTUserDevAverageEncounterDuration cursorToBTDevAverageEncounterDuration (Cursor cursor)
- BTUserDevSocialWeight cursorToBTDevSocialWeight (Cursor cursor)
- LocationEntry cursorToLocation (Cursor cursor)
- SoundLevel cursorToSound (Cursor cursor)

Private Attributes

- SQLiteDatabase db
- NSenseSQLiteHelper dbHelper
- boolean isDbOpen
- String[] allColumnsActionsEntry
- String[] allColumnsBTDevices
- String[] allColumnsBTDeviceEncounterDuration
- String[] allColumnsBTDeviceAverageEncounterDuration
- String[] allColumnsBTDeviceSocialWeight
- String[] allColumnsLocation

6.33.1 Detailed Description

This class provides various methods and information of the NSense Data base. Methods include, open database, close database, access to the tables

6.33.2 Constructor & Destructor Documentation

6.33.2.1 cs.nsense.db.NSenseDataSource.NSenseDataSource (Context *context*)

This method Construct the context. which takes Android Context as input.

Parameters

context	Interface to global information about an application environment.

6.33.3 Member Function Documentation

6.33.3.1 void cs.nsense.db.NSenseDataSource.cleanLocationTable ( )

This method clean the TABLE_LOCATION table

6.33.3.2 void cs.nsense.db.NSenseDataSource.closeDB ( )

This method close the predefined NSense database.

6.33.3.3 ActionsEntry cs.nsense.db.NSenseDataSource.cursorToActions (Cursor *cursor*) [private]

This method converts a cursor pointing to a record in the ACTIONS table to a ActionsEntry object.

Parameters

cursor	Cursor pointing to a record of the ACTIONS table.

Returns

ae the ActionsEntry object

6.33.3.4 BTUserDevAverageEncounterDuration cs.nsense.db.NSenseDataSource.cursorToBTDevAverageEncounterDuration (Cursor *cursor*) [private]

This method converts a cursor pointing to a record in the BTDEVICEAVERAGEENCOUNTERDURATION table to a BTUserDevAverageEncounterDuration object.

Parameters

cursor	The cursor pointing to a record of the BTDEVICEAVERAGEENCOUNTERDURATION table.

Returns

averageDuration The BTUserDevAverageEncounterDuration object.

6.33.3.5 BTUserDevEncounterDuration cs.nsense.db.NSenseDataSource.cursorToBTDevEncounterDuration (Cursor *cursor*) [private]

This method converts a cursor pointing to a record in the BTDEVICEENCOUNTERDURATION table to a BTUserDevEncounterDuration object.

Parameters

cursor	The cursor pointing to a record of the BTDEVICEENCOUNTERDURATION table.

Returns

duration The BTUserDevEncounterDuration object.

6.33.3.6 BTUserDevice cs.nsense.db.NSenseDataSource.cursorToBTDevice (Cursor *cursor*) [private]

This method converts a cursor pointing to a record in the BTDEVICE table to a BTDevice object.

Parameters

cursor	The cursor pointing to a record of the BTDEVICE table.

Returns

btDev The BTDevice object

6.33.3.7 BTUserDevSocialWeight cs.nsense.db.NSenseDataSource.cursorToBTDevSocialWeight (Cursor *cursor*) [private]

This method converts a cursor pointing to a record in the BTDEVICESOCIALWEIGHT table to a BTUserDevSocial-Weight object.

Parameters

cursor	The cursor pointing to a record of the BTDEVICEAVERAGEENCOUNTERDURATION table.

Returns

socialWeight The BTUserDevSocialWeight object.

6.33.3.8 LocationEntry cs.nsense.db.NSenseDataSource.cursorToLocation (Cursor *cursor*) [private]

This method converts a cursor pointing to a record in the TABLE_LOCATION table to a LocationEntry object.

Parameters

cursor	Cursor pointing to a record of the TABLE_LOCATION table.

Returns

entry the LocationEntry object

6.33.3.9 SoundLevel cs.nsense.db.NSenseDataSource.cursorToSound (Cursor *cursor*) [private]

This method provides cursor to the TABLE_SOUND table

Parameters

cursor	Cursor pointing to a record of the TABLE_SOUND table.

Returns

sl SoundLevel

6.33.3.10 ActionsEntry cs.nsense.db.NSenseDataSource.getActionEntry (String *actionentry*)

This method gets an Actions already registered by the application.

Parameters

actionentry	Actions which information should be returned

Returns

ae the ActionsEntry object, null if not found.

6.33.3.11 Map<String, ActionsEntry> cs.nsense.db.NSenseDataSource.getAllActionEntry ( )

This method gets the all the Actions recorded by the application on the ACCESS_POINTS table.

Returns

aeMap A map with the Actions objects, and the actiontype as key.

6.33.3.12 Map< String, BTUserDevAverageEncounterDuration > cs.nsense.db.NSenseDataSource.getAllBTDevAverageEncounterDuration ( )

This method gets the all the BTUserDevAverageEncounterDuration recorded by the application on the BTDEVICEAVERAGEENCOUNTERDURATION table.

Returns

btDevAverageEncounterDurationMap The map with the BTUserDevAverageEncounterDuration objects, and the BTDEV_MAC_ADDRESS as key.

6.33.3.13 Map< String, BTUserDevEncounterDuration > cs.nsense.db.NSenseDataSource.getAllBTDevEncounterDuration ( )

This method gets the all the BTUserDevEncounterDuration recorded by the application on the BTDEVICEENCOUNTERDURATION table.

Returns

btDevEncounterDurationMap The map with the BTUserDevEncounterDuration objects, and the BTDEV_MAC_ADDRESS as key.

6.33.3.14 Map< String, BTUserDevice > cs.nsense.db.NSenseDataSource.getAllBTDevice ( )

This method gets the all the BTDevice recorded by the application on the BTDEVICE table.

Returns

btDevMap The map with the BTUserDevice objects, and the BTDEV_MAC_ADDRESS as key.

6.33.3.15 Map< String, BTUserDevSocialWeight > cs.nsense.db.NSenseDataSource.getAllBTDevSocialWeight ( )

This method gets the all the BTUserDevSocialWeight recorded by the application on the BTDEVICESOCIALWEIGHT table.

Returns

btDevSocialWeightMap The map with the BTUserDevSocialWeight objects, and the BTDEV_MAC_ADDRESS as key.

6.33.3.16 Map< String, LocationEntry > cs.nsense.db.NSenseDataSource.getAllLocationEntries ( )

This method gets the all the location entries recorded by the application on the TABLE_LOCATION table.

Returns

mLocationMap A map with the LocationEntry objects.

6.33.3.17 Map< String, SoundLevel > cs.nsense.db.NSenseDataSource.getAllSoundLevel ( )

This method gets the all the sound level entries recorded by the application on the TABLE_SOUND table.

Returns

slMap A map with the SoundLevel objects.

6.33.3.18 BTUserDevice cs.nsense.db.NSenseDataSource.getBTDevice (String *btDevice*)

This method gets information about a BTDevice already registered by the application.

Parameters

mac	The MAC address of the BTDevice which information should be returned.

Returns

btDev The btDev object, null if not found.

6.33.3.19 BTUserDevAverageEncounterDuration cs.nsense.db.NSenseDataSource.getBTDeviceAverageEncounterDuration (String *btDevAverageEncounterDuration*)

This method gets average encounter duration information of a BTDevice already registered by the application.

Parameters

mac	The MAC address of the BTDevice which information should be returned.

Returns

btDevAvgEncDur The btDevAvgEncDur object, null if not found.

6.33.3.20 BTUserDevEncounterDuration cs.nsense.db.NSenseDataSource.getBTDeviceEncounterDuration (String *btDevEncounterDuration*)

This method gets encounter duration information of a BTDevice already registered by the application.

Parameters

mac	The MAC address of the BTDevice which information should be returned.

Returns

btDevEncDur The btDevEncDur object, null if not found.

6.33.3.21 BTUserDevSocialWeight cs.nsense.db.NSenseDataSource.getBTDeviceSocialWeight (String *btDevSocialWeight*)

This method gets social weight information of a BTDevice already registered by the application.

Parameters

mac	The MAC address of the BTDevice which information should be returned.

Returns

btDevSocWeight The btDevSocWeight object, null if not found.

6.33.3.22 LocationEntry cs.nsense.db.NSenseDataSource.getLocationEntry (String *bssid*)

This method get the location entries from the TABLE_LOCATION

Parameters

bssid	location id

Returns

ap location entry object

6.33.3.23 long cs.nsense.db.NSenseDataSource.getNumActionEntry ( )

This method gets the number of records in the ACTION_ENTRY table. This is, the number of Actions registered on the application.

Returns

the number of Actions registered by the application.

6.33.3.24 long cs.nsense.db.NSenseDataSource.getNumAP ( )

This method gets the number of records in the ACCESS_POINTS table. This is, the number of AP registered on the application.

Returns

number of AP registered by the application.

6.33.3.25 long cs.nsense.db.NSenseDataSource.getNumBTDevice ( )

This method gets the number of records in the BTDEVICE table. This is, the number of BTDevice registered on the application.

Returns

The number of BTDevice registered by the application.

6.33.3.26 long cs.nsense.db.NSenseDataSource.getNumEntries ( )

This method gets the number of records in the TABLE_LOCATION table. This is, the number of locations registered on the application.

Returns

the number of locations registered by the application.

6.33.3.27 long cs.nsense.db.NSenseDataSource.getNumSoundLevel ( )

This method get the entries in TABLE_SOUND table

Returns

SoundLevel The number of rows in the table

6.33.3.28 SoundLevel cs.nsense.db.NSenseDataSource.getSoundLevel (long *todayMillis*)

This method provides the sound level

Parameters

todayMillis	Time in milliseconds

Returns

sl SoundLevel with time in milliseconds

6.33.3.29 boolean cs.nsense.db.NSenseDataSource.hasActionEntry (String *actionType*)

This method checks if a given Actions has already been registered by the application.

Parameters

actionType	Actions of the Actions

Returns

true if Actions has already been registered by the application, false otherwise.

6.33.3.30 boolean cs.nsense.db.NSenseDataSource.hasBTDevice (String *btDev*)

This method checks if a given BTDevice has already been registered by the application.

Parameters

mac	The MAC address of the BTDevice.

Returns

true, if BTDevice has already been registered by the application, false otherwise.

6.33.3.31 boolean cs.nsense.db.NSenseDataSource.hasLocationEntry (String *bssid*)

This method checks whether the bssid is available in TABLE_LOCATION table

Parameters

bssid	location id

Returns

true if location id is in location ebtry object

6.33.3.32 boolean cs.nsense.db.NSenseDataSource.hasSoundLevel (long *soundDate*)

This method check whether the entry is available or not from TABLE_SOUND table

Parameters

soundDate	Date of the sound level

Returns

true If the entry is available

6.33.3.33 void cs.nsense.db.NSenseDataSource.openDB (boolean *writable*) throws SQLException

This method opens the predefined NSense database.

Parameters

writable	The first time this is called, the database will be opened.

Exceptions

SQLException	Throws SQL Exception

6.33.3.34 long cs.nsense.db.NSenseDataSource.registerLocationEntry (**LocationEntry** *entry*)

This method register a new LocationEntry in the application. It creates a new record on the TABLE_LOCATION table, with the information passed as MTrackerAP.

Parameters

entry	Location Entry information.

Returns

rowID of the newly inserted row, or -1 if an error occurred.

6.33.3.35 long cs.nsense.db.NSenseDataSource.registerNewActionEntry (**ActionsEntry** *ae*)

This method register a new ActionEntry in the application. It creates a new record on the Action Entry table, with the information passed as ActionEntry.

Parameters

ae	Action enty information.

Returns

the action type of the newly inserted row, or -1 if an error occurred.

6.33.3.36 void cs.nsense.db.NSenseDataSource.registerNewBTDevice (**BTUserDevice** *btDev,* **BTUserDevEncounterDuration** *duration,* **BTUserDevAverageEncounterDuration** *averageDuration,* **BTUserDevSocialWeight** *socialWeight*)

This method registers a new BTDevice in the application. It creates a new record on the BTDEVICE, BTDEVICEENCOUNTERDURATION, BTDEVICEAVERAGEENCOUNTERDURATION, and BTDEVICESOCIALWEIGHT tables, with the information passed as BTDevice.

Parameters

btDev	The Bluetooth device information.
duration	The Bluetooth device information regarding the duration that the BT device is within communication range of others.
averageDuration	The Bluetooth device information regarding the average duration of encounter between the BT device and other devices.
socialWeight	The Bluetooth device information regarding the social weight of the BT device towards others.

6.33.3.37 long cs.nsense.db.NSenseDataSource.registerNewSoundLevel (**SoundLevel** *sl*)

This method register the new entry into TABLE_SOUND table

Parameters

sl	Sound level object

Returns

numberOfRows number of rows are inserted in the table

6.33.3.38 int cs.nsense.db.NSenseDataSource.removeLocationEntry (**LocationEntry** *entry*)

This method checks whether entry is removed from the location entry

Parameters

entry	Location Entry

Returns

true if successful

6.33.3.39 boolean cs.nsense.db.NSenseDataSource.updateActionEntry (**ActionsEntry** *ae*)

This method update an ActionEntry already registered by the application. This modifies the corresponding record to the Entry in the ACTIONENTRY table.

Parameters

ap	action entry information.

Returns

true if successful.

6.33.3.40 void cs.nsense.db.NSenseDataSource.updateBTDevAvgEncounterDuration (**BTUserDevAverageEncounterDuration** *averageDuration*)

This method updates a BTDevice already registered by the application. This modifies the corresponding record to the BTDevice in the BTDEVICEAVERAGEENCOUNTERDURATION table.

Parameters

averageDuration	The Bluetooth device information regarding its average encounter duration.

6.33.3.41 void cs.nsense.db.NSenseDataSource.updateBTDeviceAndDuration (**BTUserDevice** *btDev,* **BTUserDevEncounterDuration** *duration*)

This method updates a BTDevice already registered by the application. This modifies the corresponding record to the BTDevice in the BTDEVICE, BTDEVICEENCOUNTERDURATION, BTDEVICEAVERAGEENCOUNTERDURATION, and BTDEVICESOCIALWEIGHT tables.

Parameters

btDev	The Bluetooth device information.
duration	The Bluetooth device information regarding the duration that the BT device is within communication range of others.

6.33.3.42 void cs.nsense.db.NSenseDataSource.updateBTDevSocialWeight (BTUserDevSocialWeight *socialWeight*)

This method updates a BTDevice already registered by the application. This modifies the corresponding record to the BTDevice in the BTDEVICESOCIALWEIGHT table.

Parameters

socialWeight	The Bluetooth device information regarding its social weight.

6.33.3.43 boolean cs.nsense.db.NSenseDataSource.updateLocationEntry (LocationEntry *entry*)

This method update an Location Entry already registered by the application. This modifies the corresponding record to the Location Entry in the TABLE_LOCATION table.

Parameters

location	Location Entry information.

Returns

true if successful.

6.33.3.44 boolean cs.nsense.db.NSenseDataSource.updateSoundLevel (SoundLevel *sl*)

This method update the sound level into TABLE_SOUND table

Parameters

sl	Sound level object

Returns

true The number of rows are updated in the table

6.33.4 Member Data Documentation

6.33.4.1 String [ ] cs.nsense.db.NSenseDataSource.allColumnsActionsEntry [private]

Initial value:

```
= {
            NSenseSQLiteHelper.COLUMN_ACTIONTYPE,
            NSenseSQLiteHelper.COLUMN_TIMESTAMP,
            NSenseSQLiteHelper.COLUMN_ENDTIMESTAMP,
            NSenseSQLiteHelper.COLUMN_AVGDURATION,
            NSenseSQLiteHelper.COLUMN_HOUR,
            NSenseSQLiteHelper.COLUMN_ACTIONCOUNTER,
            NSenseSQLiteHelper.COLUMN_DAY,
            NSenseSQLiteHelper.COLUMN_TIMEFRAME
    }
```

ACTIONS TABLE: This table stores information regarding ACTIONS. The information stored is:

- ACTIONTYPE

- TIMESTAMP
- DURATION
- LASTACTION
- ACTIONTIMEGAP
- ACTIONCOUNTER List of all columns on the ACCESS_POINTS table.

6.33.4.2 String [ ] cs.nsense.db.NSenseDataSource.allColumnsBTDeviceAverageEncounterDuration [private]

Initial value:

```
= {
            NSenseSQLiteHelper.COLUMN_BTDEV_MAC_ADDRESS,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT1
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT2
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT3
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT4
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT5
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT6
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT7
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT8
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT9
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT10
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT11
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT12
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT13
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT14
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT15
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT16
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT17
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT18
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT19
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT20
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT21
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT22
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT23
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT24
      ,
    }
```

List of all columns on the BTDEVICEAVERAGEENCOUNTERDURATION table.

6.33.4.3 String [ ] cs.nsense.db.NSenseDataSource.allColumnsBTDeviceEncounterDuration [private]

Initial value:

```
= {
            NSenseSQLiteHelper.COLUMN_BTDEV_MAC_ADDRESS,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT1
```

```
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT2
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT3
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT4
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT5
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT6
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT7
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT8
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT9
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT10
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT11
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT12
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT13
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT14
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT15
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT16
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT17
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT18
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT19
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT20
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT21
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT22
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT23
        ,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT24
        ,
    }
```

List of all columns on the BTDEVICEENCOUNTERDURATION table.

6.33.4.4 String [ ] cs.nsense.db.NSenseDataSource.allColumnsBTDevices `[private]`

Initial value:

```
= {
            NSenseSQLiteHelper.COLUMN_BTDEV_MAC_ADDRESS,
            NSenseSQLiteHelper.COLUMN_BTDEV_NAME,
            NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERSTART,
    }
```

Bluetooth pipeline BTDEVICE TABLE: This table stores information regarding BTDEVICES. The information stored is:

- BTDEV_MAC_ADDRESS
- BTDEV_NAME
- BTDEV_ENCOUNTERTIME

BTDEVICEENCOUNTERDURATION TABLE: This table stores information regarding the duration that found BTDEVICES are within communication range. The information stored is:

- BTDEV_MAC_ADDRESS

- BTDEV_ENCOUNTERDURATION

BTDEVICEAVERAGEENCOUNTERDURATION TABLE: This table stores information regarding the average duration of encounters towards found BTDEVICES. The information stored is:

- BTDEV_MAC_ADDRESS
- BTDEV_AVGENCOUNTERDURATION

BTDEVICESOCIALWEIGHT TABLE: This table stores information regarding the social weight towards found BTDEVICES. The information stored is:

- BTDEV_MAC_ADDRESS
- BTDEV_SOCIALWEIGHT List of all columns on the BTDEVICE table.

6.33.4.5 String [ ] cs.nsense.db.NSenseDataSource.allColumnsBTDeviceSocialWeight [private]

Initial value:

```
= {
            NSenseSQLiteHelper.COLUMN_BTDEV_MAC_ADDRESS,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT1
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT2
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT3
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT4
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT5
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT6
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT7
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT8
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT9
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT10
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT11
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT12
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT13
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT14
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT15
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT16
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT17
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT18
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT19
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT20
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT21
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT22
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT23
      ,
            NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT24
      ,
    }
```

List of all columns on the BTDEVICESOCIALWEIGHT table.

6.33.4.6 String [ ] cs.nsense.db.NSenseDataSource.allColumnsLocation [private]

Initial value:

```
= {
            NSenseSQLiteHelper.COLUMN_MAC_ADDRESS,
            NSenseSQLiteHelper.COLUMN_DEVICE_NAME,
            NSenseSQLiteHelper.COLUMN_DISTANCE,
            NSenseSQLiteHelper.COLUMN_LAST_UPDATE
    }
```

LOCATION TABLE: This table stores information regarding each access point visited in the past. The information stored is:

- BSSID of the device
- Device Name

Distance

- Last update List of all columns on the ACCESS_POINTS table.

6.33.4.7 String [ ] cs.nsense.db.NSenseDataSource.allColumnsSound

Initial value:

```
= {
            NSenseSQLiteHelper.KEY_ROWID,
            NSenseSQLiteHelper.SOUND_DATE ,
            NSenseSQLiteHelper.QUIET_MILLIS,
            NSenseSQLiteHelper.NORMAL_MILLIS,
            NSenseSQLiteHelper.ALERT_MILLIS,
            NSenseSQLiteHelper.NOISE_MILLIS
    }
```

Sound SOUND TABLE: This table stores information regarding user's pattern spent in each stste (quiet, normal/loud, noisy). The information stored is:

- Date
- Time spent in Queit
- Time spent in Normal
- Time spent in Alert
- Time spent in Noisy List of all columns on the Sound table.

6.33.4.8 SQLiteDatabase cs.nsense.db.NSenseDataSource.db [private]

COMMON OPERATIONS

6.33.4.9 NSenseSQLiteHelper cs.nsense.db.NSenseDataSource.dbHelper [private]

6.33.4.10 boolean cs.nsense.db.NSenseDataSource.isDbOpen [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/db/NSenseDataSource.java

6.34 cs.nsense.location.RelativePositionWiFiNoConnection.NSenseDevice Class Reference

Public Attributes

- String mWiFiDirectMACAddress
- String mWiFiAPMACAddress = ""
- String mSSID
- String mDeviceName
- int mCountNotFound = 0

6.34.1 Detailed Description

Class that holds all the information about a device found by the WiFi P2P

Author

Luis Amaral Lopes (COPELABS/ULHT)

6.34.2 Member Data Documentation

6.34.2.1 int cs.nsense.location.RelativePositionWiFiNoConnection.NSenseDevice.mCountNotFound = 0

Number of times this device was not found at the WiFi scans.

6.34.2.2 String cs.nsense.location.RelativePositionWiFiNoConnection.NSenseDevice.mDeviceName

Device Name

6.34.2.3 String cs.nsense.location.RelativePositionWiFiNoConnection.NSenseDevice.mSSID

Access Point SSID

6.34.2.4 String cs.nsense.location.RelativePositionWiFiNoConnection.NSenseDevice.mWiFiAPMACAddress = ""

MAC address from the AP received from the wifi manager

6.34.2.5 String cs.nsense.location.RelativePositionWiFiNoConnection.NSenseDevice.mWiFiDirectMACAddress

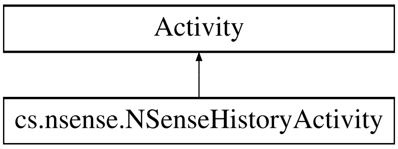
MAC address received from the discover process (Wi-Fi Direct)

The documentation for this class was generated from the following file:

- src/cs/nsense/location/RelativePositionWiFiNoConnection.java

6.35 cs.nsense.NSenseHistoryActivity Class Reference

Inheritance diagram for cs.nsense.NSenseHistoryActivity:



Static Public Attributes

- static ArrayList
  < SocialityDetails > listHistory

Protected Member Functions

- void onCreate (Bundle savedInstanceState)

Private Attributes

- ListView historyListView

Static Private Attributes

- static final String TAG = "NSenseHistoryActivity"

6.35.1 Detailed Description

This class provides layout of the NSenseHistoryActivity

6.35.2 Member Function Documentation

6.35.2.1 void cs.nsense.NSenseHistoryActivity.onCreate (Bundle *savedInstanceState*) [protected]

6.35.3 Member Data Documentation

6.35.3.1 ListView cs.nsense.NSenseHistoryActivity.historyListView [private]

This class is to get History Listview

6.35.3.2 ArrayList<SocialityDetails> cs.nsense.NSenseHistoryActivity.listHistory [static]

This list of SocialityDetails objects to get the History list of SocialityDetails

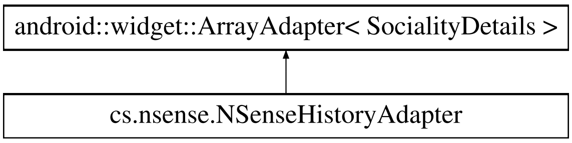
6.35.3.3 final String cs.nsense.NSenseHistoryActivity.TAG = "NSenseHistoryActivity" [static], [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/NSenseHistoryActivity.java

6.36 cs.nsense.NSenseHistoryAdapter Class Reference

Inheritance diagram for cs.nsense.NSenseHistoryAdapter:


android::widget::ArrayAdapter< SocialityDetails >
cs.nsense.NSenseHistoryAdapter

Classes

- class **EntryHolder**

Public Member Functions

- NSenseHistoryAdapter (Context context, List< SocialityDetails > data, int layoutResourceId, NSenseActivity listActivity)
- View getView (int position, View convertView, ViewGroup parent)

Static Private Attributes

- static final String TAG = "NSenseHistoryAdapter"

6.36.1 Detailed Description

This class provides adapter details to the NSenseHistoryActivity

6.36.2 Constructor & Destructor Documentation

6.36.2.1 cs.nsense.NSenseHistoryAdapter.NSenseHistoryAdapter (Context *context,* List< **SocialityDetails** > *data,* int *layoutResourceId,* **NSenseActivity** *listActivity*)

This method construct the NSenseHistoryAdapter with the context provided by NSense Activity class

Parameters

context	Interface to global information about an application environment
data	List with SocialityDetails
layoutResourceId	Layout of the SocialityDetails
listActivity	NSenseActivity

6.36.3 Member Function Documentation

6.36.3.1 View cs.nsense.NSenseHistoryAdapter.getView (int *position,* View *convertView,* ViewGroup *parent*)

6.36.4 Member Data Documentation

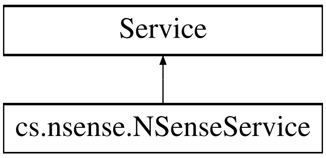
6.36.4.1 final String cs.nsense.NSenseHistoryAdapter.TAG = "NSenseHistoryAdapter" [static], [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/NSenseHistoryAdapter.java

6.37 cs.nsense.NSenseService Class Reference

Inheritance diagram for cs.nsense.NSenseService:



Classes

- class LocalBinder
- interface NOTIFICATION_ID

Public Member Functions

- NSenseService ()
- void appendLogs (String currentTime, String batteryLevel)
- IBinder onBind (Intent intent)
- int onStartCommand (Intent intent, int i, int j)
- void runAsForeground ()
- void closeAll ()
- void onCreate ()
- void onDestroy ()
- void setOnStateChangeListener (ChangeListener listener)
- void notifyLocationOutdoor (String newMessage)
- void notifyLocationIndoor (ArrayList< LocationEntry > arrayList)
- void notifyAction (String newMessage)
- void notifySocialWeight (ArrayList< socialWeight > arrayList)
- void notifySoundLevel (String newMessage)
- void notifySociality (ArrayList< SocialityDetails > arrayList)
- void notifyAvgSI (double avgSI)
- void notifyAvgProp (double avgProp)

Protected Attributes

- GoogleApiClient mGoogleApiClient
- DetectedActivity mdetectdActivity

Private Attributes

- ArrayList< ChangeListener > listeners = new ArrayList<ChangeListener> ()
- SensorManager mSensorManager
- LocationPipeline mLocationPipeline
- BluetoothCore bluetooth
- AccelerometerPipeline mAccelerometerPipeline
- MicrophonePipeline mIcrophone
- final IBinder mBinder = new LocalBinder()
- NSenseDataSource dataSource
- SocialInteraction mSocialInteraction

Static Private Attributes

- static final String TAG = "NSENSE"
- static final String Accelerometer = "Accelerometer"
- static final String GPS = "GPS"
- static final Object BT = "BT"
- static final Object Microphone ="Microphone"
- static final Timer timer = new Timer()

6.37.1 Detailed Description

This class provides methods to initialize all the pipelines, data base and get the data from all pipelines

6.37.2 Constructor & Destructor Documentation

6.37.2.1 cs.nsense.NSenseService.NSenseService ( )

6.37.3 Member Function Documentation

6.37.3.1 void cs.nsense.NSenseService.appendLogs (String *currentTime*, String *batteryLevel*)

This method stores the the battery level and time

Parameters

currentTime	Current time
batteryLevel	Battery level

BufferedWriter for performance, true to set append to file flag

6.37.3.2 void cs.nsense.NSenseService.closeAll ( )

This method close the NSense service if it is running on foreground

6.37.3.3 void cs.nsense.NSenseService.notifyAction (String *newMessage*)

6.37.3.4 void cs.nsense.NSenseService.notifyAvgProp (double *avgProp*)

6.37.3.5 void cs.nsense.NSenseService.notifyAvgSI (double *avgSI*)

6.37.3.6 void cs.nsense.NSenseService.notifyLocationIndoor (ArrayList< LocationEntry > *arrayList*)

6.37.3.7 void cs.nsense.NSenseService.notifyLocationOutdoor (String *newMessage*)

6.37.3.8 void cs.nsense.NSenseService.notifySociality (ArrayList< SocialityDetails > *arrayList*)

6.37.3.9 void cs.nsense.NSenseService.notifySocialWeight (ArrayList< socialWeight > *arrayList*)

6.37.3.10 void cs.nsense.NSenseService.notifySoundLevel (String *newMessage*)

6.37.3.11 IBinder cs.nsense.NSenseService.onBind (Intent *intent*)

6.37.3.12 void cs.nsense.NSenseService.onCreate ( )

6.37.3.13 void cs.nsense.NSenseService.onDestroy ( )

This method close all the pipelines from the NSense Service

6.37.3.14 int cs.nsense.NSenseService.onStartCommand (Intent *intent,* int *i,* int *j*)

Creates a file for Bluetooth debugging

handle case of no SDCARD present

create folder

create file

Run this service as foreground

6.37.3.15 void cs.nsense.NSenseService.runAsForeground ( )

This method provides the functionality to run the service as Foreground

6.37.3.16 void cs.nsense.NSenseService.setOnStateChangeListener (ChangeListener *listener*)

Listener Methods of all the pipelines to get data

6.37.4 Member Data Documentation

6.37.4.1 final String cs.nsense.NSenseService.Accelerometer = "Accelerometer" [static], [private]

This variable is to check Accelerometer module to be initiated

6.37.4.2 BluetoothCore cs.nsense.NSenseService.bluetooth [private]

This class is to access core functionality of Bluetooth pipeline

6.37.4.3 final Object cs.nsense.NSenseService.BT = "BT" [static], [private]

This variable is to check Bluetooth module to be initiated

6.37.4.4 NSenseDataSource cs.nsense.NSenseService.dataSource [private]

This class is to access functionality of NSense Data base

6.37.4.5 final String cs.nsense.NSenseService.GPS = "GPS" [static], [private]

This variable is to check Location module to be initiated

6.37.4.6 ArrayList<ChangeListener> cs.nsense.NSenseService.listeners = new ArrayList<ChangeListener> () [private]

This variable is used to access the listeners

6.37.4.7 AccelerometerPipeline cs.nsense.NSenseService.mAccelerometerPipeline [private]

This class is to access functionality of Accelerometer pipeline

6.37.4.8 final IBinder cs.nsense.NSenseService.mBinder = new LocalBinder() [private]

This variable to get functionality of IBinder

6.37.4.9 DetectedActivity cs.nsense.NSenseService.mdetectdActivity [protected]

This variable to store the detected activity

6.37.4.10 GoogleApiClient cs.nsense.NSenseService.mGoogleApiClient [protected]

This class is to access functionality of Google Client API

6.37.4.11 final Object cs.nsense.NSenseService.Microphone ="Microphone" [static], [private]

This variable is to check Microphone module to be initiated

6.37.4.12 MicrophonePipeline cs.nsense.NSenseService.mIcrophone [private]

This class is to access functionality of Microphone pipeline

6.37.4.13 LocationPipeline cs.nsense.NSenseService.mLocationPipeline [private]

This class is to access functionality of Location pipeline

6.37.4.14 SensorManager cs.nsense.NSenseService.mSensorManager [private]

This class is to access functionality of Sensor Manager from accelerometer pipeline

6.37.4.15 SocialInteraction cs.nsense.NSenseService.mSocialInteraction [private]

This class is to access functionality of Social Interaction details

6.37.4.16 final String cs.nsense.NSenseService.TAG = "NSENSE" [static], [private]

6.37.4.17 final Timer cs.nsense.NSenseService.timer = new Timer() [static], [private]

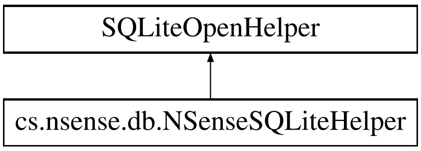
This variable to get the timer

The documentation for this class was generated from the following file:

- src/cs/nsense/NSenseService.java

6.38 cs.nsense.db.NSenseSQLiteHelper Class Reference

Inheritance diagram for cs.nsense.db.NSenseSQLiteHelper:



Public Member Functions

- NSenseSQLiteHelper (Context context)
- void onCreate (SQLiteDatabase dataBase)
- void onUpgrade (SQLiteDatabase dataBase, int oldVersion, int newVersion)

Static Public Attributes

- static final String DATABASE_NAME = "nsense.db"
- static final int DATABASE_VERSION = 1
- static final String TABLE_ACCESSPOINTS = "accesspoints"
- static final String TABLE_ACTIONS = "actions"
- static final String TABLE_BTDEVICE = "btdevices"
- static final String TABLE_BTDEVICEENCOUNTERDURATION = "btdevice_encounterduration"
- static final String TABLE_BTDEVICEAVERAGEENCOUNTERDURATION = "btdevice_averageencounterduration"
- static final String TABLE_BTDEVICESOCIALWEIGHT = "btdevice_socialweight"
- static final String TABLE_LOCATION = "location"
- static final String TABLE_SOUND = "environment_sound"
- static final String COLUMN_ID = "_id"
- static final String COLUMN_BSSID = "bssid"
- static final String COLUMN_ACTIONTYPE = "actiontype"
- static final String COLUMN_TIMESTAMP = "timestamp"
- static final String COLUMN_ENDTIMESTAMP = "endtimestamp"
- static final String COLUMN_AVGDURATION = "avgduration"
- static final String COLUMN_ACTIONCOUNTER = "actioncounter"
- static final String COLUMN_TIMEFRAME = "timeframe"
- static final String COLUMN_BTDEV_MAC_ADDRESS = "devBtMacAdd"
- static final String COLUMN_BTDEV_NAME = "devName"
- static final String COLUMN_BTDEV_ENCOUNTERSTART = "devEncounterStart"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT1 = "devEncounterDuration_slot1"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT2 = "devEncounterDuration_slot2"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT3 = "devEncounterDuration_slot3"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT4 = "devEncounterDuration_slot4"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT5 = "devEncounterDuration_slot5"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT6 = "devEncounterDuration_slot6"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT7 = "devEncounterDuration_slot7"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT8 = "devEncounterDuration_slot8"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT9 = "devEncounterDuration_slot9"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT10 = "devEncounterDuration_slot10"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT11 = "devEncounterDuration_slot11"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT12 = "devEncounterDuration_slot12"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT13 = "devEncounterDuration_slot13"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT14 = "devEncounterDuration_slot14"

- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT15 = "devEncounterDuration_slot15"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT16 = "devEncounterDuration_slot16"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT17 = "devEncounterDuration_slot17"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT18 = "devEncounterDuration_slot18"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT19 = "devEncounterDuration_slot19"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT20 = "devEncounterDuration_slot20"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT21 = "devEncounterDuration_slot21"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT22 = "devEncounterDuration_slot22"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT23 = "devEncounterDuration_slot23"
- static final String COLUMN_BTDEV_ENCOUNTERDURATION_SLOT24 = "devEncounterDuration_slot24"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT1 = "devAvgEncounterDuration_slot1"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT2 = "devAvgEncounterDuration_slot2"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT3 = "devAvgEncounterDuration_slot3"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT4 = "devAvgEncounterDuration_slot4"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT5 = "devAvgEncounterDuration_slot5"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT6 = "devAvgEncounterDuration_slot6"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT7 = "devAvgEncounterDuration_slot7"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT8 = "devAvgEncounterDuration_slot8"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT9 = "devAvgEncounterDuration_slot9"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT10 = "devAvgEncounterDuration_slot10"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT11 = "devAvgEncounterDuration_slot11"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT12 = "devAvgEncounterDuration_slot12"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT13 = "devAvgEncounterDuration_slot13"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT14 = "devAvgEncounterDuration_slot14"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT15 = "devAvgEncounterDuration_slot15"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT16 = "devAvgEncounterDuration_slot16"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT17 = "devAvgEncounterDuration_slot17"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT18 = "devAvgEncounterDuration_slot18"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT19 = "devAvgEncounterDuration_slot19"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT20 = "devAvgEncounterDuration_slot20"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT21 = "devAvgEncounterDuration_slot21"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT22 = "devAvgEncounterDuration_slot22"
- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT23 = "devAvgEncounterDuration_slot23"

- static final String COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT24 = "devAvgEncounterDuration-_slot24"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT1 = "devSocialWeight_slot1"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT2 = "devSocialWeight_slot2"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT3 = "devSocialWeight_slot3"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT4 = "devSocialWeight_slot4"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT5 = "devSocialWeight_slot5"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT6 = "devSocialWeight_slot6"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT7 = "devSocialWeight_slot7"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT8 = "devSocialWeight_slot8"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT9 = "devSocialWeight_slot9"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT10 = "devSocialWeight_slot10"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT11 = "devSocialWeight_slot11"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT12 = "devSocialWeight_slot12"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT13 = "devSocialWeight_slot13"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT14 = "devSocialWeight_slot14"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT15 = "devSocialWeight_slot15"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT16 = "devSocialWeight_slot16"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT17 = "devSocialWeight_slot17"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT18 = "devSocialWeight_slot18"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT19 = "devSocialWeight_slot19"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT20 = "devSocialWeight_slot20"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT21 = "devSocialWeight_slot21"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT22 = "devSocialWeight_slot22"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT23 = "devSocialWeight_slot23"
- static final String COLUMN_BTDEV_SOCIALWEIGHT_SLOT24 = "devSocialWeight_slot24"
- static final String COLUMN_MAC_ADDRESS = "bssid"
- static final String COLUMN_DEVICE_NAME = "name"
- static final String COLUMN_DISTANCE = "distance"
- static final String COLUMN_LAST_UPDATE = "lastupdate"
- static final String COLUMN_LATITUDE = "lat"
- static final String COLUMN_LONGITUDE = "lng"
- static final String COLUMN_DURATION = "duration"
- static final String COLUMN_DAY = "day"
- static final String COLUMN_HOUR = "hour"
- static final String KEY_ROWID = "key_row_id"
- static final String SOUND_DATE = "date_millis"
- static final String QUIET_MILLIS = "quiet_millis"
- static final String NORMAL_MILLIS = "normal_millis"
- static final String ALERT_MILLIS = "alert_millis"
- static final String NOISE_MILLIS = "noise_millis"

Static Private Attributes

- static final String CREATE_ACTIONS_TABLE
- static final String CREATE_BTDEVICE_TABLE
- static final String CREATE_BTDEVICEENCOUNTERDURATION_TABLE
- static final String CREATE_BTDEVICEAVERAGEENCOUNTERDURATION_TABLE
- static final String CREATE_BTDEVICESOCIALWEIGHT_TABLE
- static final String CREATE_LOCATION_TABLE
- static final String CREATE_SOUND_TABLE

6.38.1 Detailed Description

This class provides various methods and information of the NSense Data base. Methods include, defining the tables, listing the columns, creating the tables, upgrading the tables

6.38.2 Constructor & Destructor Documentation

6.38.2.1 cs.nsense.db.NSenseSQLiteHelper.NSenseSQLiteHelper (Context *context*)

This method construct the NSenseSQLiteHelper with the help of Android Context

Parameters

context	Interface to global information about an application environment.

6.38.3 Member Function Documentation

6.38.3.1 void cs.nsense.db.NSenseSQLiteHelper.onCreate (SQLiteDatabase *dataBase*)

This method create the SQLiteDatabase with the predefined tables

Parameters

dataBase	SQLiteDatabase

Firdose

Bluetooth pipeline

Location

Microphone

6.38.3.2 void cs.nsense.db.NSenseSQLiteHelper.onUpgrade (SQLiteDatabase *dataBase,* int *oldVersion,* int *newVersion*)

This method update the SQLiteDatabase with the predefined tables

Parameters

database	SQLiteDatabase
oldVersion	Old version of SQLiteDatabase
newVersion	New version of SQLiteDatabase

Firdose

Bluetooth pipeline

Location

Microphone

6.38.4 Member Data Documentation

6.38.4.1 final String cs.nsense.db.NSenseSQLiteHelper.ALERT_MILLIS = "alert_millis" [static]

6.38.4.2 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_ACTIONCOUNTER = "actioncounter" [static]

6.38.4.3 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_ACTIONTYPE = "actiontype" [static]

Accelerometer

6.38.4.4 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_AVGDURATION = "avgduration" [static]

6.38.4.5 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BSSID = "bssid" [static]

6.38.4.6 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT1 = "devAvgEncounterDuration_slot1" [static]

Average Encounter Duration

6.38.4.7 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT10 = "devAvgEncounterDuration_slot10" [static]

6.38.4.8 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT11 = "devAvgEncounterDuration_slot11" [static]

6.38.4.9 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT12 = "devAvgEncounterDuration_slot12" [static]

6.38.4.10 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT13 = "devAvgEncounterDuration_slot13" [static]

6.38.4.11 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT14 = "devAvgEncounterDuration_slot14" [static]

6.38.4.12 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT15 = "devAvgEncounterDuration_slot15" [static]

6.38.4.13 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT16 = "devAvgEncounterDuration_slot16" [static]

6.38.4.14 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT17 = "devAvgEncounterDuration_slot17" [static]

6.38.4.15 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT18 = "devAvgEncounterDuration_slot18" [static]

6.38.4.16 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT19 = "devAvgEncounterDuration_slot19" [static]

6.38.4.17 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT2 = "devAvgEncounterDuration_slot2" [static]

6.38.4.18 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT20 = "devAvgEncounterDuration_slot20" [static]

6.38.4.19 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT21 = "devAvgEncounterDuration_slot21" [static]

6.38.4.20 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT22 = "devAvgEncounterDuration_slot22" [static]

6.38.4.21 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT23 = "devAvgEncounterDuration_slot23" [static]

6.38.4.22 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT24 = "devAvgEncounterDuration_slot24" [static]

6.38.4.23 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT3 = "devAvgEncounterDuration_slot3" [static]

6.38.4.24 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT4 = "devAvgEncounterDuration_slot4" [static]

6.38.4.25 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT5 = "devAvgEncounterDuration_slot5" [static]

6.38.4.26 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT6 = "devAvgEncounterDuration_slot6" [static]

6.38.4.27 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT7 = "devAvgEncounterDuration_slot7" [static]

6.38.4.28 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT8 = "devAvgEncounterDuration_slot8" [static]

6.38.4.29 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT9 = "devAvgEncounterDuration_slot9" [static]

6.38.4.30 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT1 = "devEncounterDuration_slot1" [static]

Encounter Duration

6.38.4.31 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT10 = "devEncounterDuration_slot10" [static]

6.38.4.32 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT11 = "devEncounterDuration_slot11" [static]

6.38.4.33 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT12 = "devEncounterDuration_slot12" [static]

6.38.4.34 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT13 = "devEncounterDuration_slot13" [static]

6.38.4.35 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT14 = "devEncounterDuration_slot14" [static]

6.38.4.36 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT15 = "devEncounterDuration_slot15" [static]

6.38.4.37 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT16 = "devEncounterDuration_slot16" [static]

6.38.4.38 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT17 = "devEncounterDuration_slot17" [static]

6.38.4.39 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT18 = "devEncounterDuration_slot18" [static]

6.38.4.40 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT19 = "devEncounterDuration_slot19" [static]

6.38.4.41 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT2 = "devEncounterDuration_slot2" [static]

6.38.4.42 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT20 = "devEncounterDuration_slot20" [static]

6.38.4.43 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT21 = "devEncounterDuration_slot21" [static]

6.38.4.44 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT22 = "devEncounterDuration_slot22" [static]

6.38.4.45 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT23 = "devEncounterDuration_slot23" [static]

6.38.4.46 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT24 = "devEncounterDuration_slot24" [static]

6.38.4.47 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT3 = "devEncounterDuration_slot3" [static]

6.38.4.48 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT4 = "devEncounterDuration_slot4" [static]

6.38.4.49 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT5 = "devEncounterDuration_slot5" [static]

6.38.4.50 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT6 = "devEncounterDuration_slot6" [static]

6.38.4.51 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT7 = "devEncounterDuration_slot7" [static]

6.38.4.52 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT8 = "devEncounterDuration_slot8" [static]

6.38.4.53 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERDURATION_SLOT9 = "devEncounterDuration_slot9" [static]

6.38.4.54 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_ENCOUNTERSTART = "devEncounterStart" [static]

6.38.4.55 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_MAC_ADDRESS = "devBtMacAdd" [static]

Bluetooth pipeline

6.38.4.56 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_NAME = "devName" [static]

6.38.4.57 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT1 = "devSocialWeight_slot1" [static]

Social Weight

6.38.4.58 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT10 = "devSocialWeight_slot10" [static]

6.38.4.59 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT11 = "devSocialWeight_slot11" [static]

6.38.4.60 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT12 = "devSocialWeight_slot12" [static]

6.38.4.61 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT13 = "devSocialWeight_slot13" [static]

6.38.4.62 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT14 = "devSocialWeight_slot14" [static]

6.38.4.63 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT15 = "devSocialWeight_slot15" [static]

6.38.4.64 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT16 = "devSocialWeight_slot16" [static]

6.38.4.65 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT17 = "devSocialWeight_slot17" [static]

6.38.4.66 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT18 = "devSocialWeight_slot18" [static]

6.38.4.67 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT19 = "devSocialWeight_slot19" [static]

6.38.4.68 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT2 = "devSocialWeight_slot2" [static]

6.38.4.69 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT20 = "devSocialWeight_slot20" [static]

6.38.4.70 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT21 = "devSocialWeight_slot21" [static]

6.38.4.71 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT22 = "devSocialWeight_slot22" [static]

6.38.4.72 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT23 = "devSocialWeight_slot23" [static]

6.38.4.73 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT24 = "devSocialWeight_slot24" [static]

6.38.4.74 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT3 = "devSocialWeight_slot3" [static]

6.38.4.75 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT4 = "devSocialWeight_slot4" [static]

6.38.4.76 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT5 = "devSocialWeight_slot5" [static]

6.38.4.77 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT6 = "devSocialWeight_slot6" [static]

6.38.4.78 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT7 = "devSocialWeight_slot7" [static]

6.38.4.79 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT8 = "devSocialWeight_slot8" [static]

6.38.4.80 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_BTDEV_SOCIALWEIGHT_SLOT9 = "devSocialWeight_slot9" [static]

6.38.4.81 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_DAY = "day" [static]

6.38.4.82 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_DEVICE_NAME = "name" [static]

6.38.4.83 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_DISTANCE = "distance" [static]

6.38.4.84 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_DURATION = "duration" [static]

6.38.4.85 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_ENDTIMESTAMP = "endtimestamp" [static]

6.38.4.86 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_HOUR = "hour" [static]

6.38.4.87 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_ID = "_id" [static]

Location

6.38.4.88 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_LAST_UPDATE = "lastupdate" [static]

6.38.4.89 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_LATITUDE = "lat" [static]

Outdoor Location Table

6.38.4.90 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_LONGITUDE = "lng" [static]

6.38.4.91 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_MAC_ADDRESS = "bssid" [static]

Location Table

6.38.4.92 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_TIMEFRAME = "timeframe" [static]

6.38.4.93 final String cs.nsense.db.NSenseSQLiteHelper.COLUMN_TIMESTAMP = "timestamp" [static]

6.38.4.94 final String cs.nsense.db.NSenseSQLiteHelper.CREATE_ACTIONS_TABLE [static], [private]

Initial value:

```
= "create table "
            + TABLE_ACTIONS + "("
            + COLUMN_ID + " integer primary key autoincrement, "
            + COLUMN_ACTIONTYPE + " text not null, "
            + COLUMN_TIMESTAMP + " text not null, "
            + COLUMN_ENDTIMESTAMP + " text not null, "
            + COLUMN_AVGDURATION + " double not null, "
            + COLUMN_HOUR + " integer not null, "
            + COLUMN_ACTIONCOUNTER + " integer not null, "
            + COLUMN_DAY + " integer not null, "
            + COLUMN_TIMEFRAME + " text not null"
            + ");"
```

Firdose

6.38.4.95 final String cs.nsense.db.NSenseSQLiteHelper.CREATE_BTDEVICE_TABLE [static], [private]

Initial value:

```
= "create table "
            + TABLE_BTDEVICE + "("
            + COLUMN_ID + " integer primary key autoincrement, "
            + COLUMN_BTDEV_MAC_ADDRESS + " text not null unique, "
            + COLUMN_BTDEV_NAME + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERSTART + " text not null"
            + ");"
```

Bluetooth pipeline

6.38.4.96 final String cs.nsense.db.NSenseSQLiteHelper.CREATE_BTDEVICEAVERAGEENCOUNTERDURATION_TABLE [static], [private]

Initial value:

```
= "create table "
            + TABLE_BTDEVICEAVERAGEENCOUNTERDURATION + "("
            + COLUMN_ID + " integer primary key autoincrement, "
            + COLUMN_BTDEV_MAC_ADDRESS + " text not null unique, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT1 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT2 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT3 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT4 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT5 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT6 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT7 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT8 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT9 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT10 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT11 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT12 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT13 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT14 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT15 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT16 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT17 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT18 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT19 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT20 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT21 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT22 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT23 + " text not
       null, "
            + COLUMN_BTDEV_AVGENCOUNTERDURATION_SLOT24 + " text not
       null"
            + ");"
```

6.38.4.97 final String cs.nsense.db.NSenseSQLiteHelper.CREATE_BTDEVICEENCOUNTERDURATION_TABLE [static], [private]

Initial value:

```
= "create table "
            + TABLE_BTDEVICEENCOUNTERDURATION + "("
            + COLUMN_ID + " integer primary key autoincrement, "
            + COLUMN_BTDEV_MAC_ADDRESS + " text not null unique, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT1 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT2 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT3 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT4 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT5 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT6 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT7 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT8 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT9 + " text not null, "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT10 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT11 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT12 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT13 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT14 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT15 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT16 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT17 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT18 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT19 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT20 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT21 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT22 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT23 + " text not null,
       "
            + COLUMN_BTDEV_ENCOUNTERDURATION_SLOT24 + " text not null"
            + ");"
```

6.38.4.98 final String cs.nsense.db.NSenseSQLiteHelper.CREATE_BTDEVICESOCIALWEIGHT_TABLE [static], [private]

Initial value:

```
= "create table "
            + TABLE_BTDEVICESOCIALWEIGHT + "("
            + COLUMN_ID + " integer primary key autoincrement, "
            + COLUMN_BTDEV_MAC_ADDRESS + " text not null unique, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT1 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT2 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT3 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT4 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT5 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT6 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT7 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT8 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT9 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT10 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT11 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT12 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT13 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT14 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT15 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT16 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT17 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT18 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT19 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT20 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT21 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT22 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT23 + " text not null, "
            + COLUMN_BTDEV_SOCIALWEIGHT_SLOT24 + " text not null"
            + ");"
```

6.38.4.99 final String cs.nsense.db.NSenseSQLiteHelper.CREATE_LOCATION_TABLE [static], [private]

Initial value:

```
= "create table "
            + TABLE_LOCATION + "("
            + COLUMN_ID + " integer primary key autoincrement, "
            + COLUMN_MAC_ADDRESS + " text not null unique, "
            + COLUMN_DEVICE_NAME + " text not null, "
            + COLUMN_DISTANCE + " double not null, "
            + COLUMN_LAST_UPDATE + " text not null"
            + ");"
```

Location Table

6.38.4.100 final String cs.nsense.db.NSenseSQLiteHelper.CREATE_SOUND_TABLE [static], [private]

Initial value:

```
= "create table "
            + TABLE_SOUND + "("
            + KEY_ROWID + " integer primary key autoincrement, "
            + SOUND_DATE + " long not null,"
            + QUIET_MILLIS + " long, "
            + NORMAL_MILLIS + " long, "
            + ALERT_MILLIS + " long, "
            + NOISE_MILLIS + " long "
            + ");"
```

Microphone

6.38.4.101 final String cs.nsense.db.NSenseSQLiteHelper.DATABASE_NAME = "nsense.db" [static]

6.38.4.102 final int cs.nsense.db.NSenseSQLiteHelper.DATABASE_VERSION = 1 [static]

6.38.4.103 final String cs.nsense.db.NSenseSQLiteHelper.KEY_ROWID = "key_row_id" [static]

Microphone

6.38.4.104 final String cs.nsense.db.NSenseSQLiteHelper.NOISE_MILLIS = "noise_millis" [static]

6.38.4.105 final String cs.nsense.db.NSenseSQLiteHelper.NORMAL_MILLIS = "normal_millis" [static]

6.38.4.106 final String cs.nsense.db.NSenseSQLiteHelper.QUIET_MILLIS = "quiet_millis" [static]

6.38.4.107 final String cs.nsense.db.NSenseSQLiteHelper.SOUND_DATE = "date_millis" [static]

6.38.4.108 final String cs.nsense.db.NSenseSQLiteHelper.TABLE_ACCESSPOINTS = "accesspoints" [static]

MTracker

6.38.4.109 final String cs.nsense.db.NSenseSQLiteHelper.TABLE_ACTIONS = "actions" [static]

Firdose

6.38.4.110 final String cs.nsense.db.NSenseSQLiteHelper.TABLE_BTDEVICE = "btdevices" [static]

Bluetooth pipeline

6.38.4.111 final String cs.nsense.db.NSenseSQLiteHelper.TABLE_BTDEVICEAVERAGEENCOUNTERDURATION = "btdevice_averageencounterduration" [static]

6.38.4.112 final String cs.nsense.db.NSenseSQLiteHelper.TABLE_BTDEVICEENCOUNTERDURATION = "btdevice_encounterduration" [static]

6.38.4.113 final String cs.nsense.db.NSenseSQLiteHelper.TABLE_BTDEVICESOCIALWEIGHT = "btdevice_socialweight" [static]

6.38.4.114 final String cs.nsense.db.NSenseSQLiteHelper.TABLE_LOCATION = "location" [static]

Location Table

6.38.4.115 final String cs.nsense.db.NSenseSQLiteHelper.TABLE_SOUND = "environment_sound" [static]

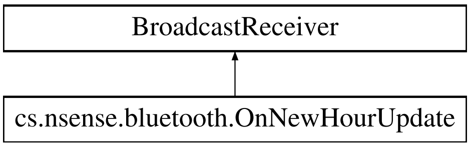
Microphone

The documentation for this class was generated from the following file:

- src/cs/nsense/db/NSenseSQLiteHelper.java

6.39 cs.nsense.bluetooth.OnNewHourUpdate Class Reference

Inheritance diagram for cs.nsense.bluetooth.OnNewHourUpdate:



Public Member Functions

- OnNewHourUpdate (NSenseDataSource datasource2)
- void onReceive (Context context, Intent intent)
- int getTimeSlot ()
- void showDevicesOnDB ()

Static Public Attributes

- static int day = 1
- static final String NEW_HOUR

Private Member Functions

- void notifyDataBaseChange ()

Private Attributes

- NSenseDataSource datasource
- ArrayList< DataBaseChangeListener > listeners = new ArrayList<DataBaseChangeListener> ()

6.39.1 Detailed Description

This class provides the computing updates for Social weights among users on hourly basis

6.39.2 Constructor & Destructor Documentation

6.39.2.1 cs.nsense.bluetooth.OnNewHourUpdate.OnNewHourUpdate (NSenseDataSource *datasource2*)

This method is the constructor for OnNewHourUpdate.

Parameters

datasource2	NSenseDataSource to access various methods and information of the NSense Data base.

6.39.3 Member Function Documentation

6.39.3.1 int cs.nsense.bluetooth.OnNewHourUpdate.getTimeSlot ( )

This method provides the current time slot.

Returns

currentTimeSlot The actual time slot.

6.39.3.2 void cs.nsense.bluetooth.OnNewHourUpdate.notifyDataBaseChange ( ) [private]

This method notifies a database change to the listeners.

6.39.3.3 void cs.nsense.bluetooth.OnNewHourUpdate.onReceive (Context *context,* Intent *intent*)

This method receives action concerning social weight updates.

Parameters

context	Interface to global information about an application environment.
intent	The intent.

Set new encounter start

Update average encounter duration

Update social weight

6.39.3.4 void cs.nsense.bluetooth.OnNewHourUpdate.showDevicesOnDB ( )

This method shows the Bluetooth devices stored on the DB.

6.39.4 Member Data Documentation

6.39.4.1 NSenseDataSource cs.nsense.bluetooth.OnNewHourUpdate.datasource [private]

This class is to access functionality of NSense Data base

6.39.4.2 int cs.nsense.bluetooth.OnNewHourUpdate.day = 1 [static]

This variable is to provide the day

6.39.4.3 ArrayList<DataBaseChangeListener> cs.nsense.bluetooth.OnNewHourUpdate.listeners = new ArrayList<DataBaseChangeListener> () [private]

6.39.4.4 final String cs.nsense.bluetooth.OnNewHourUpdate.NEW_HOUR [static]

Initial value:

```
=
            "android.intent.action.NEWHOUR"
```

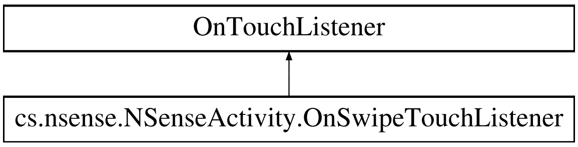
This variable is to get the new hour

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/OnNewHourUpdate.java

6.40 cs.nsense.NSenseActivity.OnSwipeTouchListener Class Reference

Inheritance diagram for cs.nsense.NSenseActivity.OnSwipeTouchListener:



Classes

- class GestureListener

Public Member Functions

- OnSwipeTouchListener (Context ctx)
- boolean onTouch (View v, MotionEvent event)
- void onSwipeRight ()
- void onSwipeLeft ()
- void onSwipeTop ()
- void onSwipeBottom ()

Private Attributes

- final GestureDetector gestureDetector

6.40.1 Detailed Description

This class implements the OnSwipeTouch functionality

6.40.2 Constructor & Destructor Documentation

6.40.2.1 cs.nsense.NSenseActivity.OnSwipeTouchListener.OnSwipeTouchListener (Context *ctx*)

6.40.3 Member Function Documentation

6.40.3.1 void cs.nsense.NSenseActivity.OnSwipeTouchListener.onSwipeBottom ( )

6.40.3.2 void cs.nsense.NSenseActivity.OnSwipeTouchListener.onSwipeLeft ( )

6.40.3.3 void cs.nsense.NSenseActivity.OnSwipeTouchListener.onSwipeRight ( )

6.40.3.4 void cs.nsense.NSenseActivity.OnSwipeTouchListener.onSwipeTop ( )

6.40.3.5 boolean cs.nsense.NSenseActivity.OnSwipeTouchListener.onTouch (View *v*, MotionEvent *event*)

6.40.4 Member Data Documentation

6.40.4.1 final GestureDetector cs.nsense.NSenseActivity.OnSwipeTouchListener.gestureDetector [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/NSenseActivity.java

6.41 cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction Class Reference

Public Member Functions

- String getmPrevAction ()

Public Attributes

- String mPrevAction
- int mId
- double mDuration
- long mStartTime
- int mHour
- int mDay

6.41.1 Detailed Description

This class provides various fields to set PreviousAction

6.41.2 Member Function Documentation

6.41.2.1 String cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction.getmPrevAction ( )

6.41.3 Member Data Documentation

6.41.3.1 int cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction.mDay

6.41.3.2 double cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction.mDuration

6.41.3.3 int cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction.mHour

6.41.3.4 int cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction.mId

6.41.3.5 String cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction.mPrevAction

6.41.3.6 long cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction.mStartTime

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/AccelerometerPipeline.java

6.42 cs.nsense.location.RelativePositionWiFiNoConnection Class Reference

Classes

- class NSenseDevice

Public Member Functions

- RelativePositionWiFiNoConnection ()
- void notifyDataBaseChange ()
- void restartDiscover ()
- void scheduleThread ()
- void close ()

Static Public Attributes

- static final String SERVICE_TYPE = "_location._tcp"

Protected Member Functions

- RelativePositionWiFiNoConnection (Context context, NSenseDataSource dataSource, LocationPipeline callback)

Private Attributes

- String TAG = "WiFiDirectBroadcastReceiverNoConnection"
- Context context
- NSenseDataSource dataSource
- LocationPipeline callback
- Handler handlerDisc = new Handler()
- int mDiscState = 0
- Random rand = new Random()
- Runnable mThread

6.42.1 Constructor & Destructor Documentation

6.42.1.1 cs.nsense.location.RelativePositionWiFiNoConnection.RelativePositionWiFiNoConnection (Context *context*, NSenseDataSource *dataSource*, LocationPipeline *callback*) [protected]

RelativePositionWiFiNoConnection constructor - It initializes the WifiServiceSearcher and WifiAccessPoint modules and creates a handler to switch between this two modules.

Parameters

context	- Interface to global information about an application environment.
dataSource	- NSense data base.
callback	- LocationPipeline module

6.42.1.2 cs.nsense.location.RelativePositionWiFiNoConnection.RelativePositionWiFiNoConnection ( )

RelativePositionWiFiNoConnection constructor.

6.42.2 Member Function Documentation

6.42.2.1 void cs.nsense.location.RelativePositionWiFiNoConnection.close ( )

It stops WifiServiceSearcher and WifiAccessPoint modules

6.42.2.2 void cs.nsense.location.RelativePositionWiFiNoConnection.notifyDataBaseChange ( )

Notifies a database change to the listeners.

6.42.2.3 void cs.nsense.location.RelativePositionWiFiNoConnection.restartDiscover ( )

It informs the mWifiServiceSearcher to restart the Wifi P2P.

6.42.2.4 void cs.nsense.location.RelativePositionWiFiNoConnection.scheduleThread ( )

Used to schedule another thread

6.42.3 Member Data Documentation

6.42.3.1 LocationPipeline cs.nsense.location.RelativePositionWiFiNoConnection.callback [private]

6.42.3.2 Context cs.nsense.location.RelativePositionWiFiNoConnection.context [private]

6.42.3.3 NSenseDataSource cs.nsense.location.RelativePositionWiFiNoConnection.dataSource [private]

6.42.3.4 Handler cs.nsense.location.RelativePositionWiFiNoConnection.handlerDisc = new Handler() [private]

6.42.3.5 int cs.nsense.location.RelativePositionWiFiNoConnection.mDiscState = 0 [private]

6.42.3.6 Runnable cs.nsense.location.RelativePositionWiFiNoConnection.mThread [private]

Initial value:

```
= new Runnable(){

        @Override
        public void run() {
            if (mDiscState == 0) {
                mWifiServiceSearcher.startServiceDiscovery();
                mDiscState = 1;
                handlerDisc.postDelayed(this, ((rand.nextInt(20 - 19) + 19) * 1000));
            } else  if (mDiscState == 1) {
                mWifiAccessPoint.stopLocalServices();
                mWifiAccessPoint.createGroup();
                mDiscState = 2;
                handlerDisc.postDelayed(this, ((rand.nextInt(40 - 20) + 20) * 1000));
            } else if (mDiscState == 2) {
                mWifiAccessPoint.removeGroup();
                mWifiServiceSearcher.stopDiscovery();
                mDiscState = 0;
            }
        }

    }
```

6.42.3.7 Random cs.nsense.location.RelativePositionWiFiNoConnection.rand = new Random() [private]

6.42.3.8 final String cs.nsense.location.RelativePositionWiFiNoConnection.SERVICE_TYPE = "_location._tcp" [static]

Used by WiFi P2P to detect only NSense devices

6.42.3.9 String cs.nsense.location.RelativePositionWiFiNoConnection.TAG = "WiFiDirectBroadcastReceiverNoConnection" [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/location/RelativePositionWiFiNoConnection.java

6.43 cs.nsense.CircleDisplay.SelectionListener Interface Reference

Public Member Functions

- void onSelectionUpdate (float val, float maxval)
- void onValueSelected (float val, float maxval)

6.43.1 Detailed Description

This method listener for callbacks when selecting values ontouch

Author

Philipp Jahoda

6.43.2 Member Function Documentation

6.43.2.1 void cs.nsense.CircleDisplay.SelectionListener.onSelectionUpdate (float *val*, float *maxval*)

This method called everytime the user moves the finger on the circle-display

Parameters

val	Value
maxval	Max value

6.43.2.2 void cs.nsense.CircleDisplay.SelectionListener.onValueSelected (float *val*, float *maxval*)

This method called when the user releases his finger fromt he circle-display

Parameters

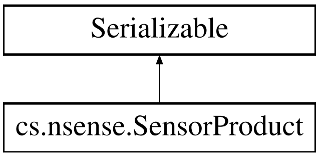
val	Value
maxval	Max value

The documentation for this interface was generated from the following file:

- src/cs/nsense/CircleDisplay.java

6.44 cs.nsense.SensorProduct Class Reference

Inheritance diagram for cs.nsense.SensorProduct:



Public Member Functions

- String getSensorName ()
- void setSensorName (String name)
- String getInterval ()
- void setInterval (String interval)
- String getClassname ()
- void setClassname (String classname)
- String toString ()
- SensorProduct ()

Private Attributes

- String sensorname = ""
- String interval = ""
- String classname = ""

6.44.1 Detailed Description

This class provides methods to get the sensor name, interval, and class of the sensor

6.44.2 Constructor & Destructor Documentation

6.44.2.1 cs.nsense.SensorProduct.SensorProduct ( )

This method provides the constructor of SensorProduct class

6.44.3 Member Function Documentation

6.44.3.1 String cs.nsense.SensorProduct.getClassname ( )

This method get the class name

Returns

classname Class name of the sensor

6.44.3.2 String cs.nsense.SensorProduct.getInterval ( )

This method get the interval

Returns

interval Interval time for the sampling

6.44.3.3 String cs.nsense.SensorProduct.getSensorName ( )

This method get the sensor name

Returns

sensorname Sensor name from the device

6.44.3.4 void cs.nsense.SensorProduct.setClassname (String *classname*)

This method set the class name

Parameters

classname	Class name of the sensor

6.44.3.5 void cs.nsense.SensorProduct.setInterval (String *interval*)

This method set the interval

Parameters

interval	Interval time for the sampling

6.44.3.6 void cs.nsense.SensorProduct.setSensorName (String *name*)

This method set the sensor name

Parameters

sensorname	Sensor name from the device

6.44.3.7 String cs.nsense.SensorProduct.toString ( )

This method return a string containing the sensor name, interval, and class name.

Returns

string with sensor name, interval, and class name.

6.44.4 Member Data Documentation

6.44.4.1 String cs.nsense.SensorProduct.classname = "" [private]

Class name of the device

6.44.4.2 String cs.nsense.SensorProduct.interval = "" [private]

Sensor interval of the device

6.44.4.3 String cs.nsense.SensorProduct.sensorname = "" [private]

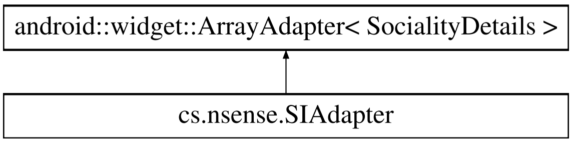
Sensor name of the device

The documentation for this class was generated from the following file:

- src/cs/nsense/SensorProduct.java

6.45 cs.nsense.SIAdapter Class Reference

Inheritance diagram for cs.nsense.SIAdapter:



Classes

- class **EntryHolder**

Public Member Functions

- SIAdapter (Context context, List< SocialityDetails > data, int layoutResourceId, NSenseActivity listActivity)
- double getMaxSI ()
- double getMaxProp ()
- double avgSI ()
- double avgProp ()
- View getView (int position, View convertView, ViewGroup parent)

Static Private Attributes

- static final String TAG = "SI Adapter"

6.45.1 Detailed Description

This class provides methods to instantly update the SocialityDetails in the NSense activity

6.45.2 Constructor & Destructor Documentation

6.45.2.1 cs.nsense.SIAdapter.SIAdapter (Context *context,* List$<$ **SocialityDetails** $>$ *data,* int *layoutResourceId,* **NSenseActivity** *listActivity*)

This method constructs the Social Interaction Adapter from the context and by providing SocialityDetails object, layout and activity information

Parameters

context	Interface to global information about an application environment.
data	List with SocialityDetails
layoutResourceId	Layout of the SocialityDetails
listActivity	NSenseActivity

6.45.3 Member Function Documentation

6.45.3.1 double cs.nsense.SIAdapter.avgProp ( )

This method provides the average propinquity

Returns

avgProp Average Propinquity

6.45.3.2 double cs.nsense.SIAdapter.avgSI ( )

This method provides the average social interaction

Returns

avgSI Average Social interaction

6.45.3.3 double cs.nsense.SIAdapter.getMaxProp ( )

This method provides the maximum propinquity

Returns

maxProp Maximum propinquity

6.45.3.4 double cs.nsense.SIAdapter.getMaxSI ( )

This class provides the maximum social interaction

Returns

maxSi Maximum social interaction

6.45.3.5 View cs.nsense.SIAdapter.getView (int *position,* View *convertView,* ViewGroup *parent*)

6.45.4 Member Data Documentation

6.45.4.1 final String cs.nsense.SIAdapter.TAG = "SI Adapter" `[static]`, `[private]`

The documentation for this class was generated from the following file:

- src/cs/nsense/SIAdapter.java

6.46 cs.nsense.inferenceModule.SocialInteraction Class Reference

Classes

- class CustomComparator
- class DeviceDetails

Public Member Functions

- SocialInteraction (final NSenseService callback, BluetoothCore mBluetoothCore, AccelerometerPipeline mAccelerometerPipeline, NSenseDataSource dataSource)
- ArrayList< DeviceDetails > getSocialIntAndProp (ArrayList< DeviceDetails > mDevDetails, double mEnvronmentSound, double mMovementActivity)
- ArrayList< DeviceDetails > getDeviceDetails ()
- double getMovementActivity ()
- double getEnvSound ()
- int getTimeSlot ()
- void close (Context mContext)

Static Public Attributes

- static ArrayList< DeviceDetails > mDeviceDetails
- static Map< String, HashMap
  < String, Double > > mDistance
- static double mEnvronmentSound = 0.0
- static double mMovementActivity = 0.0

Private Member Functions

- ArrayList< SocialityDetails > computeEMA (ArrayList< SocialityDetails > listSocialDetails)

Private Attributes

- NSenseDataSource dataSource
- BluetoothCore mBluetoothCore
- AccelerometerPipeline mAccelerometerPipeline
- Handler mHandler
- double factor = 0.0
- double runningSiEMA = 0.0
- double runningPropEMA = 0.0

Static Private Attributes

- static final String TAG = "SOCIALInteraction"

6.46.1 Detailed Description

This class computes the social Interaction and propinquity

6.46.2 Constructor & Destructor Documentation

6.46.2.1 cs.nsense.inferenceModule.SocialInteraction.SocialInteraction (final **NSenseService** *callback*, **BluetoothCore** *mBluetoothCore*, **AccelerometerPipeline** *mAccelerometerPipeline*, **NSenseDataSource** *dataSource*)

This class construct the SocialInteraction

Parameters

callback	Supply NSenseService object for NSenseActivity to use
mBluetoothCore	It contains the core functionalities of the application and can support to perform the social context analysis prior to storing the required information in the database.
m-Accelerometer-Pipeline	It provides various methods to provide Accelerometer sensor data
dataSource	NSenseDataSource to access various methods and information of the NSenseDataSource

for Ema

Getting propinquity and Social Interaction

getSocialInteraction

6.46.3 Member Function Documentation

6.46.3.1 void cs.nsense.inferenceModule.SocialInteraction.close (Context *mContext*)

This method close the context

Parameters

context	Interface to global information about an application environment.

6.46.3.2 ArrayList<**SocialityDetails**> cs.nsense.inferenceModule.SocialInteraction.computeEMA (ArrayList< **SocialityDetails** > *listSocialDetails*) [private]

This method computes the EMA

Parameters

listSocialDetails	List of Sociality details objects

Returns

List of sociality details object with computed EMA

6.46.3.3 ArrayList<**DeviceDetails**> cs.nsense.inferenceModule.SocialInteraction.getDeviceDetails ( )

This method Provides the average of the current social weight values

Returns

avgSW The average of the current social weights towards nodes in the vicinity

Compute current social weight

6.46.3.4 double cs.nsense.inferenceModule.SocialInteraction.getEnvSound ( )

This method provides the Environment sound

Returns

Environment sound from the Sound pi

Environment Sound

6.46.3.5 double cs.nsense.inferenceModule.SocialInteraction.getMovementActivity ( )

This method provides the movement activity from accelerometer pipeline

Returns

activity Current activity from the accelerometer pipeline

Current Activity

6.46.3.6 ArrayList<DeviceDetails> cs.nsense.inferenceModule.SocialInteraction.getSocialIntAndProp (ArrayList< DeviceDetails > *mDevDetails*, double *mEnvronmentSound*, double *mMovementActivity*)

This method Provides the social Interaction of the neighbor devices

Parameters

mDevDetails	List of DeviceDetails object
mEnvronment-Sound	Environment sound from the microphone
mMovement-Activity	Current activity from accelerometer pipeline

Returns

avgSW The social Interaction towards nodes in the vicinity

6.46.3.7 int cs.nsense.inferenceModule.SocialInteraction.getTimeSlot ( )

Provides the current time slot

Returns

currentTimeSlot The actual time slot

6.46.4 Member Data Documentation

6.46.4.1 NSenseDataSource cs.nsense.inferenceModule.SocialInteraction.dataSource [private]

This class is to access functionality of NSense Data base

6.46.4.2 double cs.nsense.inferenceModule.SocialInteraction.factor = 0.0 [private]

for EMA This variable is used to define factor

6.46.4.3 AccelerometerPipeline cs.nsense.inferenceModule.SocialInteraction.mAccelerometerPipeline [private]

This class is to access functionality of Accelerometer pipeline

6.46.4.4 BluetoothCore cs.nsense.inferenceModule.SocialInteraction.mBluetoothCore [private]

This class is to access core functionality of Bluetooth pipeline

6.46.4.5 ArrayList<DeviceDetails> cs.nsense.inferenceModule.SocialInteraction.mDeviceDetails [static]

This variable is used to access the listeners

6.46.4.6 Map<String, HashMap<String, Double> > cs.nsense.inferenceModule.SocialInteraction.mDistance [static]

This variable is used to get the distance

6.46.4.7 double cs.nsense.inferenceModule.SocialInteraction.mEnvronmentSound = 0.0 [static]

This variable is used to get the environment sound

6.46.4.8 Handler cs.nsense.inferenceModule.SocialInteraction.mHandler [private]

This class is to send or process message

6.46.4.9 double cs.nsense.inferenceModule.SocialInteraction.mMovementActivity = 0.0 [static]

This variable is used to get the movement activity

6.46.4.10 double cs.nsense.inferenceModule.SocialInteraction.runningPropEMA = 0.0 [private]

This variable is used to get the Propinquity EMA

6.46.4.11 double cs.nsense.inferenceModule.SocialInteraction.runningSiEMA = 0.0 [private]

This variable is used to get Social interaction EMA

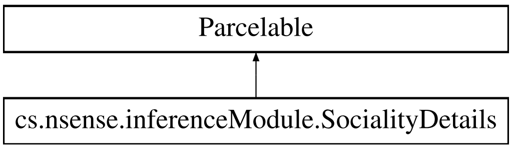
6.46.4.12 final String cs.nsense.inferenceModule.SocialInteraction.TAG = "SOCIALInteraction" [static], [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/inferenceModule/SocialInteraction.java

6.47 cs.nsense.inferenceModule.SocialityDetails Class Reference

Inheritance diagram for cs.nsense.inferenceModule.SocialityDetails:



Public Member Functions

- SocialityDetails ()
- String getDevName ()
- void setDevName (String devName)

- double getmDistance ()
- void setmDistance (double mDistance)
- double getmSI ()
- void setmSI (double mSI)
- double getmPropinquity ()
- void setmPropinquity (double mPropinquity)
- double getmSiEMA ()
- void setmSiEMA (double mSiEMA)
- double getmPropEMA ()
- void setmPropEMA (double mPropEMA)
- double getmPrevSiEMA ()
- void setmPrevSiEMA (double mPrevSiEMA)
- double getmPrevPropEMA ()
- void setmPrevPropEMA (double mPrevPropEMA)
- int describeContents ()
- void writeToParcel (Parcel dest, int flags)
- SocialityDetails (Parcel in)

Public Attributes

- String devName
- double mDistance
- double mSI
- double mPropinquity
- double mSiEMA
- double mPropEMA
- double mPrevSiEMA
- double mPrevPropEMA

Static Public Attributes

- static final
  Parcelable.Creator
  < SocialityDetails > CREATOR

6.47.1 Detailed Description

This class provides getter and setter methods of SocialityDetails object

6.47.2 Constructor & Destructor Documentation

6.47.2.1 cs.nsense.inferenceModule.SocialityDetails.SocialityDetails ( )

This constructor method for SocialityDetails

6.47.2.2 cs.nsense.inferenceModule.SocialityDetails.SocialityDetails (Parcel *in*)

This method initialize the parcelable with sociality details object

Parameters

in	Sociality details

6.47.3 Member Function Documentation

6.47.3.1 int cs.nsense.inferenceModule.SocialityDetails.describeContents ( )

6.47.3.2 String cs.nsense.inferenceModule.SocialityDetails.getDevName ( )

This method get the device name

Returns

devName the device name

6.47.3.3 double cs.nsense.inferenceModule.SocialityDetails.getmDistance ( )

This method get the distance

Returns

mDistance the distance

6.47.3.4 double cs.nsense.inferenceModule.SocialityDetails.getmPrevPropEMA ( )

This method get the previous propinquity EMA

Returns

mPrevPropEMA the previous propinquity EMA

6.47.3.5 double cs.nsense.inferenceModule.SocialityDetails.getmPrevSiEMA ( )

This method get the previous social interaction EMA

Returns

mPrevSiEMA the previous social interaction EMA

6.47.3.6 double cs.nsense.inferenceModule.SocialityDetails.getmPropEMA ( )

This method get the propinquity EMA

Returns

mPropEMA the propinquity EMA

6.47.3.7 double cs.nsense.inferenceModule.SocialityDetails.getmPropinquity ( )

This method get the propinquity

Returns

mPropinquity the propinquity

6.47.3.8 double cs.nsense.inferenceModule.SocialityDetails.getmSI ( )

This method get the social interaction

Returns

mSI the social interaction

6.47.3.9 double cs.nsense.inferenceModule.SocialityDetails.getmSiEMA ( )

This method get the social interaction EMA

Returns

mSiEMA the social interaction EMA

6.47.3.10 void cs.nsense.inferenceModule.SocialityDetails.setDevName (String *devName*)

This method set the device name

Parameters

devName	the device name

6.47.3.11 void cs.nsense.inferenceModule.SocialityDetails.setmDistance (double *mDistance*)

This method set the distance

Parameters

mDistance	the distance

6.47.3.12 void cs.nsense.inferenceModule.SocialityDetails.setmPrevPropEMA (double *mPrevPropEMA*)

This method set the previous propinquity EMA

Parameters

mPrevPropEMA	the previous propinquity EMA

6.47.3.13 void cs.nsense.inferenceModule.SocialityDetails.setmPrevSiEMA (double *mPrevSiEMA*)

This method set the previous social interaction EMA

Parameters

mPrevSiEMA	the previous social interaction EMA

6.47.3.14 void cs.nsense.inferenceModule.SocialityDetails.setmPropEMA (double *mPropEMA*)

This method set the propinquity EMA

Parameters

mPropEMA	the propinquity EMA

6.47.3.15 void cs.nsense.inferenceModule.SocialityDetails.setmPropinquity (double *mPropinquity*)

This method set the propinquity

Parameters

mPropinquity	the propinquity

6.47.3.16 void cs.nsense.inferenceModule.SocialityDetails.setmSI (double *mSI*)

This method set the social interaction

Parameters

mSI	the social interaction

6.47.3.17 void cs.nsense.inferenceModule.SocialityDetails.setmSiEMA (double *mSiEMA*)

This method set the social interaction EMA

Parameters

mSiEMA	the social interaction EMA

6.47.3.18 void cs.nsense.inferenceModule.SocialityDetails.writeToParcel (Parcel *dest,* int *flags*)

This method writes to the parcable with sociality details

Parameters

dest	The Parcel in which the Sociality details should be written.
flags	Flags about how the object should be written. May be 0 or PARCELABLE_WRITE_RETURN_VALUE.

6.47.4 Member Data Documentation

6.47.4.1 final Parcelable.Creator<SocialityDetails> cs.nsense.inferenceModule.SocialityDetails.CREATOR [static]

Initial value:

```
= new Parcelable.Creator<SocialityDetails>() {
        public SocialityDetails createFromParcel(Parcel in) {
            return new SocialityDetails(in);
        }

        public SocialityDetails[] newArray(int size) {
            return new SocialityDetails[size];
        }
    }
```

This method creates the parcelable with sociality details object

6.47.4.2 String cs.nsense.inferenceModule.SocialityDetails.devName

Device name variable of the SocialityDetails object

6.47.4.3 double cs.nsense.inferenceModule.SocialityDetails.mDistance

Distance variable of the SocialityDetails object

6.47.4.4 double cs.nsense.inferenceModule.SocialityDetails.mPrevPropEMA

Previous Propinquity EMA variable of the SocialityDetails object

6.47.4.5 double cs.nsense.inferenceModule.SocialityDetails.mPrevSiEMA

Previous Social Interaction EMA variable of the SocialityDetails object

6.47.4.6 double cs.nsense.inferenceModule.SocialityDetails.mPropEMA

Propinquity EMA variable of the SocialityDetails object

6.47.4.7 double cs.nsense.inferenceModule.SocialityDetails.mPropinquity

Propinquity variable of the SocialityDetails object

6.47.4.8 double cs.nsense.inferenceModule.SocialityDetails.mSI

Social Interaction variable of the SocialityDetails object

6.47.4.9 double cs.nsense.inferenceModule.SocialityDetails.mSiEMA

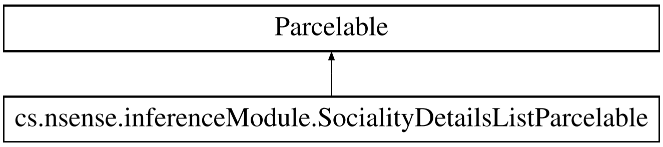
Social Interaction EMA variable of the SocialityDetails object

The documentation for this class was generated from the following file:

- src/cs/nsense/inferenceModule/SocialityDetails.java

6.48 cs.nsense.inferenceModule.SocialityDetailsListParcelable Class Reference

Inheritance diagram for cs.nsense.inferenceModule.SocialityDetailsListParcelable:



Public Member Functions

- SocialityDetailsListParcelable ()
- List< SocialityDetails > getSocialityDetailsList ()

- void setSocialityDetailsList (List< SocialityDetails > socialityList)
- int describeContents ()
- void writeToParcel (Parcel dest, int flags)
- SocialityDetailsListParcelable (Parcel in)

Static Public Attributes

- static final
  Parcelable.Creator
  < SocialityDetailsListParcelable > CREATOR

6.48.1 Detailed Description

This class provides Parcelable for SocialityDetails object

6.48.2 Constructor & Destructor Documentation

6.48.2.1 cs.nsense.inferenceModule.SocialityDetailsListParcelable.SocialityDetailsListParcelable ( )

The constructor of SocialityDetailsListParcelable

6.48.2.2 cs.nsense.inferenceModule.SocialityDetailsListParcelable.SocialityDetailsListParcelable (Parcel *in*)

This method initialize the Sociality DetailsList Parcelable

Parameters

in	The Parcel in which the object should be written.

6.48.3 Member Function Documentation

6.48.3.1 int cs.nsense.inferenceModule.SocialityDetailsListParcelable.describeContents ( )

6.48.3.2 List<SocialityDetails> cs.nsense.inferenceModule.SocialityDetailsListParcelable.getSocialityDetailsList ( )

This method get the sociality details list

Returns

socialityDetailsList List of sociality details object

6.48.3.3 void cs.nsense.inferenceModule.SocialityDetailsListParcelable.setSocialityDetailsList (List< SocialityDetails > *socialityList*)

This method set the sociality details list

Parameters

socialityList	List of SocialityDetails objects

6.48.3.4 void cs.nsense.inferenceModule.SocialityDetailsListParcelable.writeToParcel (Parcel *dest,* int *flags*)

This method write to the parcel list

Parameters

dest	The Parcel in which the Sociality details should be written.
flags	Flags about how the object should be written. May be 0 or PARCELABLE_WRITE_RETURN_VALUE.

6.48.4 Member Data Documentation

6.48.4.1 final Parcelable.Creator<SocialityDetailsListParcelable> cs.nsense.inferenceModule.SocialityDetailsListParcelable.CREATOR [static]

Initial value:

```
=
            new Parcelable.Creator<SocialityDetailsListParcelable>() {
        public SocialityDetailsListParcelable createFromParcel(Parcel in) {
            return new SocialityDetailsListParcelable(in);
        }

        public SocialityDetailsListParcelable[] newArray(int size) {
            return new SocialityDetailsListParcelable[size];
        }
    }
```

This method creates the Parcelable

The documentation for this class was generated from the following file:

- src/cs/nsense/inferenceModule/SocialityDetailsListParcelable.java

6.49 cs.nsense.bluetooth.BluetoothCore.socialWeight Class Reference

Public Member Functions

- String getmMacAddress ()
- String getmDeviceName ()
- int getmSocialWeight ()

Public Attributes

- String mMacAddress
- String mDeviceName
- int mSocialWeight

6.49.1 Detailed Description

This class holds information about a device: name, MAC address, and social weight towards it.

6.49.2 Member Function Documentation

6.49.2.1 String cs.nsense.bluetooth.BluetoothCore.socialWeight.getmDeviceName ( )

This method returns the Device name

Parameters

mDeviceName	Device name from the socialWeight object

6.49.2.2 String cs.nsense.bluetooth.BluetoothCore.socialWeight.getmMacAddress ( )

This method returns the MAC Address

Parameters

mMacAddress	Bluetooth MAC address from the socialWeight object

6.49.2.3 int cs.nsense.bluetooth.BluetoothCore.socialWeight.getmSocialWeight ( )

This method returns the Social weight

Parameters

mSocialWeight	Device name from the socialWeight object

6.49.3 Member Data Documentation

6.49.3.1 String cs.nsense.bluetooth.BluetoothCore.socialWeight.mDeviceName

6.49.3.2 String cs.nsense.bluetooth.BluetoothCore.socialWeight.mMacAddress

6.49.3.3 int cs.nsense.bluetooth.BluetoothCore.socialWeight.mSocialWeight

The documentation for this class was generated from the following file:

- src/cs/nsense/bluetooth/BluetoothCore.java

6.50 cs.nsense.microphone.SoundLevel Class Reference

Public Member Functions

- SoundLevel ()
- String toString ()
- int getRowID ()
- void setRowID (int rowID)
- void setSoundDate (int l)
- void setQuietTime (long quietTime)
- void setNormalTime (long normalTime)
- void setAlertTime (long alertTime)
- void setNoiseTime (long noiseTime)
- long getSoundDate ()
- long getQuietTime ()
- long getNormalTime ()
- long getAlertTime ()
- long getNoiseTime ()

Public Attributes

- int RowID
- long SoundDate
- long QuietTime
- long NormalTime
- long AlertTime
- long NoiseTime

6.50.1 Detailed Description

This class extracts the different sound levels such as Quite, Normal, Alert, and Noisy based on the sound frequency levels.

6.50.2 Constructor & Destructor Documentation

6.50.2.1 cs.nsense.microphone.SoundLevel.SoundLevel ( )

This class Construct the SoundLevel

6.50.3 Member Function Documentation

6.50.3.1 long cs.nsense.microphone.SoundLevel.getAlertTime ( )

This method get the Alert Time

Returns

AlertTime the time duration of Alert Level

6.50.3.2 long cs.nsense.microphone.SoundLevel.getNoiseTime ( )

This method get the Noise Time

Returns

NoiseTime the time duration of Noise Level

6.50.3.3 long cs.nsense.microphone.SoundLevel.getNormalTime ( )

This method get the Normal Time

Returns

NormalTime the time duration of Normal Level

6.50.3.4 long cs.nsense.microphone.SoundLevel.getQuietTime ( )

This method get the QUITE Time

Returns

QuietTime the time duration of Quite Level

6.50.3.5 int cs.nsense.microphone.SoundLevel.getRowID ( )

This method get the rowID

Returns

RowID the rowID

6.50.3.6 long cs.nsense.microphone.SoundLevel.getSoundDate ( )

This method get the current date

Returns

SoundDate the system date

6.50.3.7 void cs.nsense.microphone.SoundLevel.setAlertTime (long *alertTime*)

This method set the time duration in Alert

Parameters

alertTime	AlertTime into SoundLevel object

6.50.3.8 void cs.nsense.microphone.SoundLevel.setNoiseTime (long *noiseTime*)

This method set the time duration in Noisy

Parameters

noiseTime	NoiseTime into SoundLevel object

6.50.3.9 void cs.nsense.microphone.SoundLevel.setNormalTime (long *normalTime*)

This method Set the time duration in Normal

Parameters

normalTime	NormalTime into SoundLevel object

6.50.3.10 void cs.nsense.microphone.SoundLevel.setQuietTime (long *quietTime*)

This method set the time duration in Quite

Parameters

quietTime	QuiteTime into SoundLevel object

6.50.3.11 void cs.nsense.microphone.SoundLevel.setRowID (int *rowID*)

This method set the row ID

Parameters

rowID	RowId into SoundLevel object

6.50.3.12 void cs.nsense.microphone.SoundLevel.setSoundDate (int *l*)

This method set the date on which the time duration of each sound level

Parameters

l	SoundDate into SoundLevel object

6.50.3.13 String cs.nsense.microphone.SoundLevel.toString ( )

This method return a string containing the RowID, SoundDate, QuietTime, NormalTime, AlertTime and NoiseTime.

Returns

string String with RowID, SoundDate, QuietTime, NormalTime, AlertTime and NoiseTime values

6.50.4 Member Data Documentation

6.50.4.1 long cs.nsense.microphone.SoundLevel.AlertTime

AlertTime variable of the SoundLevel object

6.50.4.2 long cs.nsense.microphone.SoundLevel.NoiseTime

NoiseTime variable of the SoundLevel object

6.50.4.3 long cs.nsense.microphone.SoundLevel.NormalTime

NormalTime variable of the SoundLevel object

6.50.4.4 long cs.nsense.microphone.SoundLevel.QuietTime

QuietTime variable of the SoundLevel object

6.50.4.5 int cs.nsense.microphone.SoundLevel.RowID

RowID variable of the SoundLevel object

6.50.4.6 long cs.nsense.microphone.SoundLevel.SoundDate

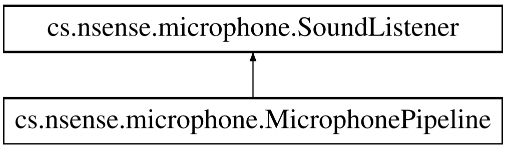
SoundDate variable of the SoundLevel object

The documentation for this class was generated from the following file:

- src/cs/nsense/microphone/SoundLevel.java

6.51 cs.nsense.microphone.SoundListener Interface Reference

Inheritance diagram for cs.nsense.microphone.SoundListener:


cs.nsense.microphone.SoundListener
cs.nsense.microphone.MicrophonePipeline

Public Member Functions

- void updateSoundLevel (SoundLevel soundLevel)

6.51.1 Detailed Description

This class is the listener class to update sound levels in the application.

6.51.2 Member Function Documentation

6.51.2.1 void cs.nsense.microphone.SoundListener.updateSoundLevel (SoundLevel *soundLevel*)

This method called when SoundLevel table has been updated in the database

Parameters

soundLevel	SoundLevel object

Implemented in cs.nsense.microphone.MicrophonePipeline.

The documentation for this interface was generated from the following file:

- src/cs/nsense/microphone/SoundListener.java

6.52 cs.nsense.microphone.SoundManager Class Reference

Public Member Functions

- SoundManager (NSenseService callback)
- void stop ()
- double soundDb ()
- double getMaxAmplitude ()
- double getAmplitudeHPF ()
- void setSoundListener (SoundListener listener)

Static Public Member Functions

- static boolean micExist (Context context)
- static boolean isCapturing ()

Static Public Attributes

- static final double AMP = 1.0

Static Protected Member Functions

- static Dialog showDialog (String id)
- static void finish ()

Protected Attributes

- double avgdB

Static Private Member Functions

- static PackageManager getPackageManager ()

Private Attributes

- MediaRecorder mRecorder = null
- final int NUMBER_OF_SAMPLES = 20
- double hPF = 0.0
- Runnable mRunPool

Static Private Attributes

- static Context mContext
- static final double HPF_FILTER = 0.97
- static final String FEATURE_NOT_AVAILBLE = null
- static final int POLL_INTERVAL = 300
- static Boolean micExist
- static boolean capturing = false

6.52.1 Detailed Description

This gives us the filtered sound frequency, and converion to amplitude. In this class we implement two filters which are EMA and HPF filters.

6.52.2 Constructor & Destructor Documentation

6.52.2.1 cs.nsense.microphone.SoundManager.SoundManager (**NSenseService** *callback*)

This method construct the SoundManager

Parameters

callback	Supply functionality for NSenseActivity to use

6.52.3 Member Function Documentation

6.52.3.1 static void cs.nsense.microphone.SoundManager.finish ( ) `[static], [protected]`

This method to finish

6.52.3.2 double cs.nsense.microphone.SoundManager.getAmplitudeHPF ( )

This method is to apply the sound filter to supress some noise on the captured sound wave In this method implemented two filters namely EMA and HPF Either the filter we can choose

Highpass Pre-emphasizing Filter hPF y(n) = x(n)-0.97∗x(n-1)

EMA Filter EMA yk = a ∗ xk+(1-a)yk-1

Returns

hpf Highpass Pre-emphasizing Filter

6.52.3.3 double cs.nsense.microphone.SoundManager.getMaxAmplitude ( )

This method extract the maximum sound amplitude by using Android method getMaxAmplitude()

Returns

getMaxAmplitude null or maximum amplitude of captured sound wave

6.52.3.4 static PackageManager cs.nsense.microphone.SoundManager.getPackageManager ( ) [static], [private]

6.52.3.5 static boolean cs.nsense.microphone.SoundManager.isCapturing ( ) [static]

This method returns true if capturing the sound

Returns

capturing Checks whether it is capturing or not

6.52.3.6 static boolean cs.nsense.microphone.SoundManager.micExist (Context *context*) [static]

This method checks whether the microphone exists in the device or not

Parameters

context	Interface to global information about an application environment

Returns

true if device exists, dialog otherwise

6.52.3.7 void cs.nsense.microphone.SoundManager.setSoundListener (SoundListener *listener*)

This method set the SoundListener

Parameters

listener	the SoundListener

6.52.3.8 static Dialog cs.nsense.microphone.SoundManager.showDialog (String *id*) [static], [protected]

This method shows the dailog

Parameters

id	String which provide the availability of feature

Returns

True/False if Feature not available/Feature is available

6.52.3.9 double cs.nsense.microphone.SoundManager.soundDb ( )

This method convert the received sound apmlitude into decibels by using acustic wave equation

Returns

sound in amplitude

6.52.3.10 void cs.nsense.microphone.SoundManager.stop ( )

This method intialise the Media recorder. Media Recorder is used to start microhphone activity by collecting sound samples captured dynamically. The process takes place in the temporary file of the users device. This method stop the media recorder

6.52.4 Member Data Documentation

6.52.4.1 final double cs.nsense.microphone.SoundManager.AMP = 1.0 [static]

This variable to set the Amplitude

6.52.4.2 double cs.nsense.microphone.SoundManager.avgdB [protected]

This variable to access average decibles

6.52.4.3 boolean cs.nsense.microphone.SoundManager.capturing = false [static], [private]

This variable to check the capturing

6.52.4.4 final String cs.nsense.microphone.SoundManager.FEATURE_NOT_AVAILBLE = null [static], [private]

This variable to check the Feature is not available

6.52.4.5 double cs.nsense.microphone.SoundManager.hPF = 0.0 [private]

This variable to set the High pass filter

6.52.4.6 final double cs.nsense.microphone.SoundManager.HPF_FILTER = 0.97 [static], [private]

This variable to set the filter for High pass filter

6.52.4.7 Context cs.nsense.microphone.SoundManager.mContext [static], [private]

Interface to global information about an application environment.

6.52.4.8 Boolean cs.nsense.microphone.SoundManager.micExist [static], [private]

This variable to check the microphone is exist

6.52.4.9 MediaRecorder cs.nsense.microphone.SoundManager.mRecorder = null [private]

This class is Used to record audio.

6.52.4.10 Runnable cs.nsense.microphone.SoundManager.mRunPool [private]

Initial value:

```
= new Runnable() {
        @Override
        public void run() {
            if(isCapturing()){
                stop();
            }else{
                Toast.makeText(mContext, "Error enabling Microphone Closing...", Toast.LENGTH_SHORT
      ).show();
                stop();
            }
        }
    }
```

6.52.4.11 final int cs.nsense.microphone.SoundManager.NUMBER_OF_SAMPLES = 20 [private]

This variable to set how many samples are collected

6.52.4.12 final int cs.nsense.microphone.SoundManager.POLL_INTERVAL = 300 [static], [private]

This variable to set the poll interval

The documentation for this class was generated from the following file:

- src/cs/nsense/microphone/SoundManager.java

6.53 cs.nsense.accelerometer.WekaClassifier Class Reference

Static Public Member Functions

- static double classify1 (Object[] i) throws Exception
- static double classify (Object[] i) throws Exception

6.53.1 Detailed Description

This class provides some objects to classify the real time activities based on trained data set

6.53.2 Member Function Documentation

6.53.2.1 static double cs.nsense.accelerometer.WekaClassifier.classify (Object[] *i*) throws Exception [static]

This method classify the activities

Parameters

Object	Object with activity instances

Returns

activity The current activity

6.53.2.2 static double cs.nsense.accelerometer.WekaClassifier.classify1 (Object[ ] *i*) throws Exception [static]

This method classify the activities

Parameters

Object	Object with activity instances

Returns

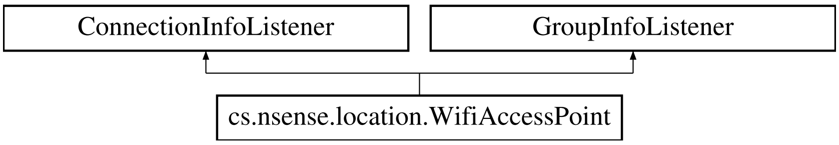
activity The current activity

The documentation for this class was generated from the following file:

- src/cs/nsense/accelerometer/WekaClassifier.java

6.54 cs.nsense.location.WifiAccessPoint Class Reference

Inheritance diagram for cs.nsense.location.WifiAccessPoint:



Classes

- class AccessPointReceiver
- class WifiScanReceiverLocation

Public Member Functions

- WifiAccessPoint (Context Context, RelativePositionWiFiNoConnection callback, NSenseDataSource dataSource)
- void restartDiscover ()
- void doScan ()
- void Start ()
- void createGroup ()
- void Stop ()
- void removeGroup ()
- String getWFAPMacAddress ()
- void onGroupInfoAvailable (WifiP2pGroup group)
- void startLocalService (String instance, String mMACAddress)
- void stopLocalServices ()
- void onConnectionInfoAvailable (WifiP2pInfo info)
- double calculateDistance (double signalLevelInDb, double freqInMHz)

Private Member Functions

- void saveDistance (double mDistance, String mWiFiDirectMAC, String mDeviceName)

Private Attributes

- String TAG = "WifiAccessPoint"
- WifiAccessPoint that = this
- Context context
- NSenseDataSource dataSource
- WifiP2pManager p2p
- WifiP2pManager.Channel channel
- String mNetworkName = ""
- String mMACAddress = ""
- WifiP2pDnsSdServiceInfo service
- BroadcastReceiver receiver
- IntentFilter filter
- RelativePositionWiFiNoConnection callback
- Boolean toScan = false
- WifiManager wifiManager = null
- WifiScanReceiverLocation wifiReceiver = null
- boolean mRequestConnectionInfo = false

6.54.1 Constructor & Destructor Documentation

6.54.1.1 cs.nsense.location.WifiAccessPoint.WifiAccessPoint (Context *Context*, RelativePositionWiFiNoConnection *callback*, NSenseDataSource *dataSource*)

WifiAccessPoint constructor

Parameters

Context	-Interface to global information about an application environment.
callback	- RelativePositionWiFiNoConnection module.
dataSource	- NSense data base.

6.54.2 Member Function Documentation

6.54.2.1 double cs.nsense.location.WifiAccessPoint.calculateDistance (double *signalLevelInDb*, double *freqInMHz*)

Computes the distance to a device.

Parameters

signalLevelInDb	- The detected signal level in dBm, also known as the RSSI.
freqInMHz	- The frequency in MHz of the channel over which the client is communicating with the access point.

Returns

6.54.2.2 void cs.nsense.location.WifiAccessPoint.createGroup ( )

Create a Group Owner. It creates a Access point in this device.

6.54.2.3 void cs.nsense.location.WifiAccessPoint.doScan ( )

Requests a WiFi scan

6.54.2.4 String cs.nsense.location.WifiAccessPoint.getWFAPMacAddress ( )

Provides the Group owner MAC address (Wifi P2P interface created for the group owner)

Returns

the Group owner MAC address

6.54.2.5 void cs.nsense.location.WifiAccessPoint.onConnectionInfoAvailable (WifiP2pInfo *info*)

Request group information

6.54.2.6 void cs.nsense.location.WifiAccessPoint.onGroupInfoAvailable (WifiP2pGroup *group*)

Provides information about the Group Owner created (Access Point) This information include the SSID and the MAC address.

6.54.2.7 void cs.nsense.location.WifiAccessPoint.removeGroup ( )

Remove the group created. It stops the access point created in this device.

6.54.2.8 void cs.nsense.location.WifiAccessPoint.restartDiscover ( )

Sends a request to the RelativePositionWiFiNoConnection to restart the WiFi p2p discover mechanism.

6.54.2.9 void cs.nsense.location.WifiAccessPoint.saveDistance (double *mDistance,* String *mWiFiDirectMAC,* String *mDeviceName*) [private]

Saves the computed distance to a specific device to the database.

Parameters

mDistance	- Computed distance to the device
mWiFiDirectMAC	- MAC Address of the device
mDeviceName	- Device Name

Save Distance into DB

Device exists

New Device

6.54.2.10 void cs.nsense.location.WifiAccessPoint.Start ( )

It initializes the WiFi P2P and register a receiver to receive all info about the WiFi state and connection state.

6.54.2.11 void cs.nsense.location.WifiAccessPoint.startLocalService (String *instance,* String *mMACAddress*)

Register a local service with the name of the SSID and the MAC address of the group owner created

Parameters

instance	- ID that identifies the application
mMACAddress	- the Group Owner MAC address

6.54.2.12 void cs.nsense.location.WifiAccessPoint.Stop ( )

It cleans the local services registered in WiFi p2p and remove the receiver.

6.54.2.13 void cs.nsense.location.WifiAccessPoint.stopLocalServices ( )

Removes all local services created.

6.54.3 Member Data Documentation

6.54.3.1 RelativePositionWiFiNoConnection cs.nsense.location.WifiAccessPoint.callback [private]

RelativePositionWiFiNoConnection module

6.54.3.2 WifiP2pManager.Channel cs.nsense.location.WifiAccessPoint.channel [private]

A channel that connects the application to the Wifi p2p framework.

6.54.3.3 Context cs.nsense.location.WifiAccessPoint.context [private]

Interface to global information about an application environment.

6.54.3.4 NSenseDataSource cs.nsense.location.WifiAccessPoint.dataSource [private]

NSense Data base

6.54.3.5 IntentFilter cs.nsense.location.WifiAccessPoint.filter [private]

Structured description of Intent values to be matched.

6.54.3.6 String cs.nsense.location.WifiAccessPoint.mMACAddress = "" [private]

MAC Address of the AP created on this device (Group Owner)

6.54.3.7 String cs.nsense.location.WifiAccessPoint.mNetworkName = "" [private]

SSID of the AP created on this device (Group Owner)

6.54.3.8 boolean cs.nsense.location.WifiAccessPoint.mRequestConnectionInfo = false [private]

Flag to inform if a request was already performed and it is still waiting for a result.

6.54.3.9 WifiP2pManager cs.nsense.location.WifiAccessPoint.p2p [private]

Android WiFi P2P Manager

6.54.3.10 BroadcastReceiver cs.nsense.location.WifiAccessPoint.receiver [private]

Base class for code that will receive intents sent by sendBroadcast().

6.54.3.11 WifiP2pDnsSdServiceInfo cs.nsense.location.WifiAccessPoint.service [private]

A class for storing Bonjour service information that is advertised over a Wi-Fi peer-to-peer setup.

6.54.3.12 String cs.nsense.location.WifiAccessPoint.TAG = "WifiAccessPoint" [private]

6.54.3.13 WifiAccessPoint cs.nsense.location.WifiAccessPoint.that = this [private]

WifiAccessPoint Module

6.54.3.14 Boolean cs.nsense.location.WifiAccessPoint.toScan = false [private]

Flag about if it is necessary to do a WiFi scan

6.54.3.15 WifiManager cs.nsense.location.WifiAccessPoint.wifiManager = null [private]

This class provides the primary API for managing all aspects of Wi-Fi connectivity.

6.54.3.16 WifiScanReceiverLocation cs.nsense.location.WifiAccessPoint.wifiReceiver = null [private]

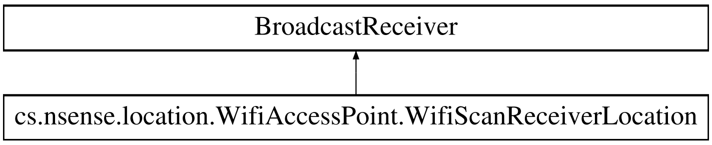
BroadcastReceiver that receives a WiFi scan result.

The documentation for this class was generated from the following file:

- src/cs/nsense/location/WifiAccessPoint.java

6.55 cs.nsense.location.WifiAccessPoint.WifiScanReceiverLocation Class Reference

Inheritance diagram for cs.nsense.location.WifiAccessPoint.WifiScanReceiverLocation:



Public Member Functions

- void onReceive (Context c, Intent intent)

6.55.1 Detailed Description

BroadcastReceiver that receives a WiFi scan result.

Author

Luis Amaral Lopes (COPELABS/ULHT)

6.55.2 Member Function Documentation

6.55.2.1 void cs.nsense.location.WifiAccessPoint.WifiScanReceiverLocation.onReceive (Context *c*, Intent *intent*)

AP Found

Get the SSID

The documentation for this class was generated from the following file:

- src/cs/nsense/location/WifiAccessPoint.java

6.56 cs.nsense.location.WifiServiceSearcher Class Reference

Classes

- class **TxTInfoReceived**

Public Member Functions

- WifiServiceSearcher (Context Context, RelativePositionWiFiNoConnection callback, NSenseDataSource dataSource)
- void Start ()
- void Stop ()
- void startServiceDiscovery ()
- void restartWifiP2P ()
- void stopDiscovery ()

Private Attributes

- String TAG = "WifiServiceSearcher"
- Context mContext
- NSenseDataSource dataSource
- WifiP2pManager p2p
- WifiP2pManager.Channel channel
- WifiP2pDnsSdServiceRequest serviceRequest
- RelativePositionWiFiNoConnection callback
- TxTInfoReceived mTxTInfoReceived = null

6.56.1 Constructor & Destructor Documentation

6.56.1.1 cs.nsense.location.WifiServiceSearcher.WifiServiceSearcher (Context *Context*, RelativePositionWiFiNoConnection *callback*, NSenseDataSource *dataSource*)

WifiServiceSearcher constructor

Parameters

Context	- Interface to global information about an application environment.
callback	- RelativePositionWiFiNoConnection module.

dataSource	- NSense data base.

6.56.2 Member Function Documentation

6.56.2.1 void cs.nsense.location.WifiServiceSearcher.restartWifiP2P ( )

Removes all services request and start a new service request.

6.56.2.2 void cs.nsense.location.WifiServiceSearcher.Start ( )

It initializes the WiFi p2p and gets a channel.

6.56.2.3 void cs.nsense.location.WifiServiceSearcher.startServiceDiscovery ( )

Requests a service discovery. Register listeners for DNS-SD services. These are callbacks invoked by the system when a service is actually discovered.

A service has been discovered. Is this our app?

update the UI and add the item the discovered

device.

Known Device - Just update the SSID

Save Distance into DB

New Device

A new TXT record is available. Pick up the advertised buddy name.

After attaching listeners, create a service request and initiate

discovery.

6.56.2.4 void cs.nsense.location.WifiServiceSearcher.Stop ( )

Stops any discovery request running.

6.56.2.5 void cs.nsense.location.WifiServiceSearcher.stopDiscovery ( )

Stops any discovery request running.

6.56.3 Member Data Documentation

6.56.3.1 RelativePositionWiFiNoConnection cs.nsense.location.WifiServiceSearcher.callback [private]

RelativePositionWiFiNoConnection module.

6.56.3.2 WifiP2pManager.Channel cs.nsense.location.WifiServiceSearcher.channel [private]

A channel that connects the application to the Wifi p2p framework.

6.56.3.3 NSenseDataSource cs.nsense.location.WifiServiceSearcher.dataSource [private]

NSense Data base.

6.56.3.4 Context cs.nsense.location.WifiServiceSearcher.mContext [private]

Interface to global information about an application environment.

6.56.3.5 TxTInfoReceived cs.nsense.location.WifiServiceSearcher.mTxTInfoReceived = null [private]

Object that contains info from the TXT of the device found.

6.56.3.6 WifiP2pManager cs.nsense.location.WifiServiceSearcher.p2p [private]

This class provides the API for managing Wi-Fi peer-to-peer connectivity.

6.56.3.7 WifiP2pDnsSdServiceRequest cs.nsense.location.WifiServiceSearcher.serviceRequest [private]

A class for creating a Bonjour service discovery request.

6.56.3.8 String cs.nsense.location.WifiServiceSearcher.TAG = "WifiServiceSearcher" [private]

The documentation for this class was generated from the following file:

- src/cs/nsense/location/WifiServiceSearcher.java

Chapter 7

File Documentation

7.1 src/com/meapsoft/FFT.java File Reference

Classes

- class com.meapsoft.FFT

Packages

- package com.meapsoft

7.2 src/cs/nsense/accelerometer/AccelerometerListener.java File Reference

Classes

- interface cs.nsense.accelerometer.AccelerometerListener

Packages

- package cs.nsense.accelerometer

7.3 src/cs/nsense/accelerometer/AccelerometerPipeline.java File Reference

Classes

- class cs.nsense.accelerometer.AccelerometerPipeline
- class cs.nsense.accelerometer.AccelerometerPipeline.PreviousAction
- class cs.nsense.accelerometer.AccelerometerPipeline.ActivityClassificationTask
- class cs.nsense.accelerometer.AccelerometerPipeline.AccListener
- class cs.nsense.accelerometer.AccelerometerPipeline.HourChangeAlarmReceiver

Packages

- package cs.nsense.accelerometer

7.4 src/cs/nsense/accelerometer/AccelerSenManager.java File Reference

Classes

- class cs.nsense.accelerometer.AccelerSenManager

Packages

- package cs.nsense.accelerometer

7.5 src/cs/nsense/accelerometer/ActionsEntry.java File Reference

Classes

- class cs.nsense.accelerometer.ActionsEntry

Packages

- package cs.nsense.accelerometer

7.6 src/cs/nsense/accelerometer/Globals.java File Reference

Classes

- class cs.nsense.accelerometer.Globals

Packages

- package cs.nsense.accelerometer

7.7 src/cs/nsense/accelerometer/WekaClassifier.java File Reference

Classes

- class cs.nsense.accelerometer.WekaClassifier

Packages

- package cs.nsense.accelerometer

7.8 src/cs/nsense/bluetooth/BluetoothCore.java File Reference

Classes

- class cs.nsense.bluetooth.BluetoothCore
- class cs.nsense.bluetooth.BluetoothCore.socialWeight
- class cs.nsense.bluetooth.BluetoothCore.CustomComparator
- class **cs.nsense.bluetooth.BluetoothCore.ServiceBTListener**

Packages

- package cs.nsense.bluetooth

7.9 src/cs/nsense/bluetooth/BTDeviceFinder.java File Reference

Classes

- interface cs.nsense.bluetooth.BTDeviceFinder

Packages

- package cs.nsense.bluetooth

7.10 src/cs/nsense/bluetooth/BTManager.java File Reference

Classes

- class cs.nsense.bluetooth.BTManager
- class **cs.nsense.bluetooth.BTManager.FindBluetoothDevice**
- class **cs.nsense.bluetooth.BTManager.adapterBroadcastReceiver**

Packages

- package cs.nsense.bluetooth

7.11 src/cs/nsense/bluetooth/BTUserDevAverageEncounterDuration.java File Reference

Classes

- class cs.nsense.bluetooth.BTUserDevAverageEncounterDuration

Packages

- package cs.nsense.bluetooth

7.12 src/cs/nsense/bluetooth/BTUserDevEncounterDuration.java File Reference

Classes

- class cs.nsense.bluetooth.BTUserDevEncounterDuration

Packages

- package cs.nsense.bluetooth

7.13 src/cs/nsense/bluetooth/BTUserDevice.java File Reference

Classes

- class cs.nsense.bluetooth.BTUserDevice

Packages

- package cs.nsense.bluetooth

7.14 src/cs/nsense/bluetooth/BTUserDevSocialWeight.java File Reference

Classes

- class cs.nsense.bluetooth.BTUserDevSocialWeight

Packages

- package cs.nsense.bluetooth

7.15 src/cs/nsense/bluetooth/OnNewHourUpdate.java File Reference

Classes

- class cs.nsense.bluetooth.OnNewHourUpdate

Packages

- package cs.nsense.bluetooth

7.16 src/cs/nsense/ChangeListener.java File Reference

Classes

- interface cs.nsense.ChangeListener

Packages

- package cs.nsense

7.17 src/cs/nsense/CircleDisplay.java File Reference

Classes

- class cs.nsense.CircleDisplay
- interface cs.nsense.CircleDisplay.SelectionListener
- class **cs.nsense.CircleDisplay.Utils**

Packages

- package cs.nsense

7.18 src/cs/nsense/db/DataBaseChangeListener.java File Reference

Classes

- interface cs.nsense.db.DataBaseChangeListener

Packages

- package cs.nsense.db

7.19 src/cs/nsense/db/NSenseDataSource.java File Reference

Classes

- class cs.nsense.db.NSenseDataSource

Packages

- package cs.nsense.db

7.20 src/cs/nsense/db/NSenseSQLiteHelper.java File Reference

Classes

- class cs.nsense.db.NSenseSQLiteHelper

Packages

- package cs.nsense.db

7.21 src/cs/nsense/inferenceModule/SocialInteraction.java File Reference

Classes

- class cs.nsense.inferenceModule.SocialInteraction
- class cs.nsense.inferenceModule.SocialInteraction.DeviceDetails
- class cs.nsense.inferenceModule.SocialInteraction.CustomComparator

Packages

- package cs.nsense.inferenceModule

7.22 src/cs/nsense/inferenceModule/SocialityDetails.java File Reference

Classes

- class cs.nsense.inferenceModule.SocialityDetails

Packages

- package cs.nsense.inferenceModule

7.23 src/cs/nsense/inferenceModule/SocialityDetailsListParcelable.java File Reference

Classes

- class cs.nsense.inferenceModule.SocialityDetailsListParcelable

Packages

- package cs.nsense.inferenceModule

7.24 src/cs/nsense/location/CleanTask.java File Reference

Classes

- class cs.nsense.location.CleanTask

Packages

- package cs.nsense.location

7.25 src/cs/nsense/location/LocationEntry.java File Reference

Classes

- class cs.nsense.location.LocationEntry

Packages

- package cs.nsense.location

7.26 src/cs/nsense/location/LocationPipeline.java File Reference

Classes

- class cs.nsense.location.LocationPipeline

Packages

- package cs.nsense.location

7.27 src/cs/nsense/location/RelativePositionWiFiNoConnection.java File Reference

Classes

- class cs.nsense.location.RelativePositionWiFiNoConnection
- class cs.nsense.location.RelativePositionWiFiNoConnection.NSenseDevice

Packages

- package cs.nsense.location

7.28 src/cs/nsense/location/WifiAccessPoint.java File Reference

Classes

- class cs.nsense.location.WifiAccessPoint
- class cs.nsense.location.WifiAccessPoint.AccessPointReceiver
- class cs.nsense.location.WifiAccessPoint.WifiScanReceiverLocation

Packages

- package cs.nsense.location

7.29 src/cs/nsense/location/WifiServiceSearcher.java File Reference

Classes

- class cs.nsense.location.WifiServiceSearcher
- class **cs.nsense.location.WifiServiceSearcher.TxTInfoReceived**

Packages

- package cs.nsense.location

7.30 src/cs/nsense/microphone/MicrophonePipeline.java File Reference

Classes

- class cs.nsense.microphone.MicrophonePipeline

Packages

- package cs.nsense.microphone

7.31 src/cs/nsense/microphone/SoundLevel.java File Reference

Classes

- class cs.nsense.microphone.SoundLevel

Packages

- package cs.nsense.microphone

7.32 src/cs/nsense/microphone/SoundListener.java File Reference

Classes

- interface cs.nsense.microphone.SoundListener

Packages

- package cs.nsense.microphone

7.33 src/cs/nsense/microphone/SoundManager.java File Reference

Classes

- class cs.nsense.microphone.SoundManager

Packages

- package cs.nsense.microphone

7.34 src/cs/nsense/NSenseActivity.java File Reference

Classes

- class cs.nsense.NSenseActivity
- class cs.nsense.NSenseActivity.CustomComparator
- class cs.nsense.NSenseActivity.OnSwipeTouchListener
- class cs.nsense.NSenseActivity.OnSwipeTouchListener.GestureListener

Packages

- package cs.nsense

7.35 src/cs/nsense/NSenseHistoryActivity.java File Reference

Classes

- class cs.nsense.NSenseHistoryActivity

Packages

- package cs.nsense

7.36 src/cs/nsense/NSenseHistoryAdapter.java File Reference

Classes

- class cs.nsense.NSenseHistoryAdapter
- class **cs.nsense.NSenseHistoryAdapter.EntryHolder**

Packages

- package cs.nsense

7.37 src/cs/nsense/NSenseService.java File Reference

Classes

- class cs.nsense.NSenseService
- interface cs.nsense.NSenseService.NOTIFICATION_ID
- class cs.nsense.NSenseService.LocalBinder

Packages

- package cs.nsense

7.38 src/cs/nsense/SensorProduct.java File Reference

Classes

- class cs.nsense.SensorProduct

Packages

- package cs.nsense

7.39 src/cs/nsense/SIAdapter.java File Reference

Classes

- class cs.nsense.SIAdapter
- class **cs.nsense.SIAdapter.EntryHolder**

Packages

- package cs.nsense