

Trottinette Électrique Connectée

1.0

Généré par Doxygen 1.7.6.1

Jeudi Juin 7 2018 20 :44 :46

Table des matières

1	Page principale du projet TEC (Trottinette Électrique Connectée)	1
1.1	Introduction	1
1.2	Table des matières	1
2	Changelog	2
3	Configuration	4
4	Manuel d'installation	4
5	Recette IR	5
6	Base de données	5
7	A propos	5
8	Licence GPL	5
9	Liste des choses à faire	5
10	Documentation des classes	6
10.1	Référence de la classe <code>com.example.iris.myapplication.MainActivity</code> . . .	6
10.1.1	Description détaillée	10
10.1.2	Documentation des fonctions membres	10
10.1.3	Documentation des données membres	22
10.2	Référence de la classe <code>com.example.iris.myapplication.Peripherique-Bluetooth</code>	27
10.2.1	Description détaillée	28
10.2.2	Documentation des constructeurs et destructeur	28
10.2.3	Documentation des fonctions membres	29
10.2.4	Documentation des données membres	34
10.3	Référence de la classe <code>com.example.iris.myapplication.Trajet</code>	36
10.3.1	Description détaillée	37
10.3.2	Documentation des constructeurs et destructeur	37
10.3.3	Documentation des fonctions membres	38
10.3.4	Documentation des données membres	41

10.4	Référence de la classe <code>com.example.iris.myapplication.TrameTTE</code>	42
10.4.1	Description détaillée	43
10.4.2	Documentation des constructeurs et destructeur	43
10.4.3	Documentation des fonctions membres	43
10.4.4	Documentation des données membres	46
10.5	Référence de la classe <code>com.example.iris.myapplication.PeripheriqueBluetooth.TReception</code>	47
10.5.1	Description détaillée	49
10.5.2	Documentation des constructeurs et destructeur	49
10.5.3	Documentation des fonctions membres	49
10.5.4	Documentation des données membres	54
11	Documentation des fichiers	55
11.1	Référence du fichier <code>AndroidManifest.xml</code>	55
11.2	Référence du fichier <code>Changelog.dox</code>	55
11.3	Référence du fichier <code>MainActivity.java</code>	55
11.4	Référence du fichier <code>PeripheriqueBluetooth.java</code>	55
11.5	Référence du fichier <code>README.dox</code>	55
11.6	Référence du fichier <code>Trajet.java</code>	55
11.7	Référence du fichier <code>TrameTTE.java</code>	55

1 Page principale du projet TEC (Trottinette Électrique Connectée)

1.1 Introduction

Système embarqué sur une trottinette électrique équipée de capteurs afin d'assister l'utilisateur sur son trajet.

1.2 Table des matières

- Configuration
- Manuel d'installation
- Changelog
- Recette IR
- Base de données
- A propos
- Licence GPL

Dépôt SVN : <https://svn.riouxsvn.com/trottinette>

2 Changelog

r26 | ahachette | 2018-05-25 16 :40 :20 +0200 (ven. 25 mai 2018) | 1 ligne

Remise de la version 1.0

r25 | ahachette | 2018-05-25 15 :52 :16 +0200 (ven. 25 mai 2018) | 3 lignes

- Modification des commentaires de fonctions de la classe MainActivity et Trajet
- saisie du TODO BUGS et changelog

r24 | ahachette | 2018-05-25 14 :12 :19 +0200 (ven. 25 mai 2018) | 3 lignes

-Ajout d'une fonctionnalité pour visualiser le trajet en cours -Suppression du bouton paramétrage

r23 | tvaira | 2018-04-20 10 :01 :53 +0200 (ven. 20 avril 2018) | 1 ligne

Maj du screenshot de l'application pour Doxygen

r22 | ahachette | 2018-04-18 17 :20 :28 +0200 (mer. 18 avril 2018) | 2 lignes

- Changement des paramètres d'update de la géolocalisation

r21 | tvaira | 2018-04-16 15 :10 :01 +0200 (lun. 16 avril 2018) | 1 ligne

Vérification des TODO

r20 | ahachette | 2018-04-12 22 :24 :51 +0200 (jeu. 12 avril 2018) | 5 lignes

- Implantation d'une classe Trajet
- Affichage des informations sur le Trajet en cours
- Vitesse moyenne, distance parcourue, consommation
- Filtre pour afficher seulement les modules bluetooth HC-05

r19 | tvaira | 2018-04-07 09 :36 :41 +0200 (sam. 07 avril 2018) | 1 ligne

Ajout de la classe Trajet à compléter

r18 | tvaira | 2018-04-07 09 :32 :51 +0200 (sam. 07 avril 2018) | 1 ligne

Ajout des icônes pour les boutons

r17 | tvaira | 2018-04-06 19 :06 :44 +0200 (ven. 06 avril 2018) | 1 ligne

Retour sur la revue 2 (Trajet)

r16 | tvaira | 2018-04-06 18 :19 :23 +0200 (ven. 06 avril 2018) | 1 ligne

Retour sur la revue 2

r15 | ahachette | 2018-04-05 17 :52 :58 +0200 (jeu. 05 avril 2018) | 3 lignes

- Changement de presentation reduction a une activite pour l'ihm
- Utilisation de string pour le texte

r14 | tvaira | 2018-04-05 13 :52 :43 +0200 (jeu. 05 avril 2018) | 1 ligne

Mise à jour du simulateur

r13 | ahachette | 2018-04-05 11 :54 :06 +0200 (jeu. 05 avril 2018) | 5 lignes

- Reception des donnees de la trottinette
- Acquittement des donnees de la trottinette
- Traitement de la trame recue
- Affichage des donnees de fonctionnement

r12 | ahachette | 2018-04-04 17 :36 :01 +0200 (mer. 04 avril 2018) | 5 lignes

- Changement de nom de fichier du projet
- Instanciation de l'activite demarer trajet avec l'affichage d'une carte
- Finalisation de la connexion et deconnexion Bluetooth
- Reception de la trame TEC

r11 | ahachette | 2018-03-30 12 :29 :43 +0200 (ven. 30 mars 2018) | 1 ligne

Creation de la 2eme activite

r9 | ahachette | 2018-03-29 13 :36 :54 +0200 (jeu. 29 mars 2018) | 1 ligne

commit du tags 0.8

r8 | tvaira | 2018-03-24 11 :11 :03 +0100 (sam. 24 mars 2018) | 1 ligne

Ajout parametrage Doxygen

r7 | ahachette | 2018-03-22 17 :44 :45 +0100 (jeu. 22 mars 2018) | 5 lignes

- Creation de tout les affichages de l'ihm
- Fin de l'implementation du bluetooth
- Version de test du bluetooth effectue

- Preparation d'une classe Carte pour la cartographie

r6 | ahachette | 2018-03-21 17 :56 :21 +0100 (mer. 21 mars 2018) | 4 lignes

- Creation d'une classe PeripheriqueBluetooth
- Implementation de la connexion bluetooth
- Changement de type pour l'affichage (textView -> imageView)

r5 | ahachette | 2018-03-15 17 :00 :13 +0100 (jeu. 15 mars 2018) | 1 ligne

Preparation bluetooth

r4 | ahachette | 2018-02-22 16 :33 :02 +0100 (jeu. 22 févr. 2018) | 1 ligne

Initialisation projet android

3 Configuration

Poste de développement :

- Distribution : Ubuntu 12.04.5 LTS
- OS : GNU/Linux
- Noyau : Linux
- Version : 3.8.0-44-generic
- Machine : x86_64
- Processeur : Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz
- Mémoire RAM : 8129984 kB

Informations de version sur les outils

- Android Studio 2.3
- java 1.8.0
- SDK Android API 25 : Android 7.1.1 (Nougat)
- svn, version 1.6.17 (r1128011)
- doxygen 1.7.6.1
- bouml Bouml 7.4

4 Manuel d'installation

Fabrication de l'exécutable :

- `qmake`
- `make`

5 Recette IR

Étudiant 2 : Hachette Alexandre

- Le protocole de communication avec la TTE est spécifié et mis en œuvre ;
- La réception des données de fonctionnement de la TTE est effective ;
- La visualisation des données de fonctionnement de la TTE et la durée d'utilisation est fonctionnelle ;
- L'autonomie pour un parcours est calculée et affichée ;
- La carte avec la géolocalisation de la TTE est affichée et actualisée périodiquement.

6 Base de données

7 A propos

Auteur

Hachette Alexandre <alexandre.hachette@gmail.com>

Version

1.0

Date

2018

8 Licence GPL

This program is free software ; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation ; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY ; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program ; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

9 Liste des choses à faire

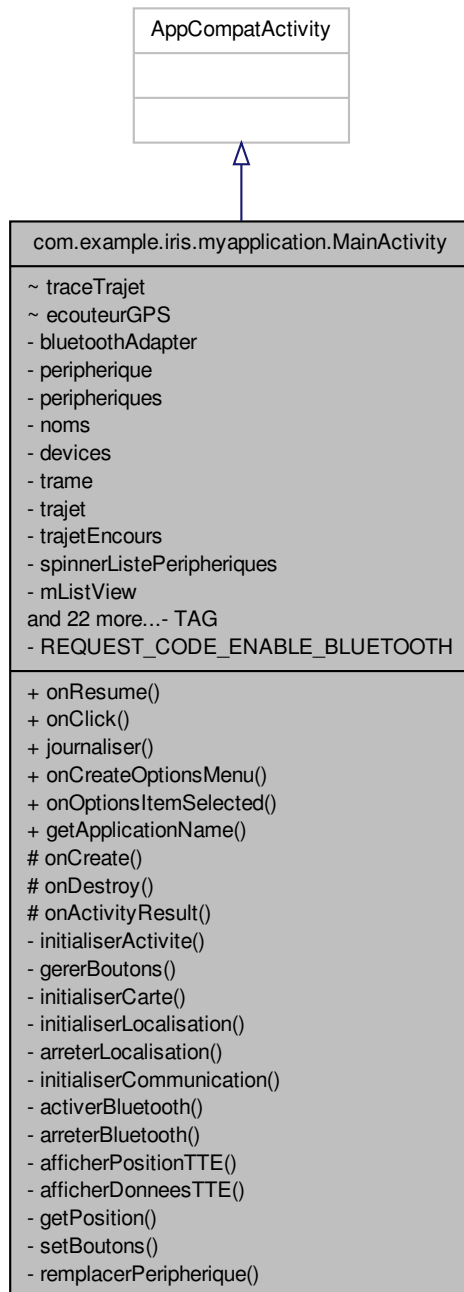
Membre [com.example.iris.myapplication.MainActivity.activerBluetooth](#) ()
changer en TEC-01

10 Documentation des classes

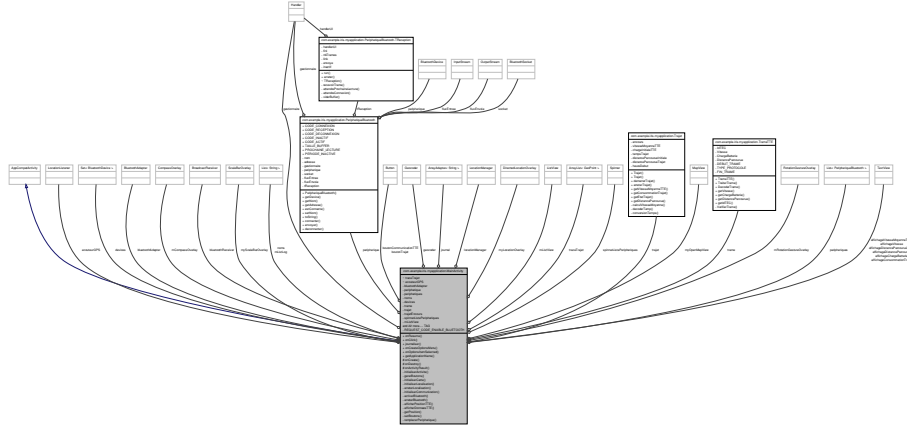
10.1 Référence de la classe `com.example.iris.myapplication.MainActivity`

Activité principale de l'application (Thread UI)

Graphe d'héritage de com.example.iris.myapplication.MainActivity :



Grphe de collaboration de com.example.iris.myapplication.MainActivity :



Fonctions membres publiques

- void **onResume** ()
- void **onClick** (View view)
- void **journaliser** (String text)
Permet d'afficher des messages dans l'IHM.
- boolean **onCreateOptionsMenu** (Menu menu)
- boolean **onOptionsItemSelected** (MenuItem item)

Fonctions membres publiques statiques

- static String **getApplicationName** (Context context)

Fonctions membres protégées

- void **onCreate** (Bundle savedInstanceState)
Demarre l'activite.
- void **onDestroy** ()
Arrete l'activite.
- void **onActivityResult** (int requestCode, int resultCode, Intent data)
Fonction d'affichage si le bluetooth est active ou non.

Attributs de paquetage

- ArrayList< GeoPoint > **traceTrajet**
- LocationListener **ecouteurGPS**

Fonctions membres privées

- void **initialiserActivite** ()
Intialise l'affichage de l'activité
- void **gererBoutons** ()

- *Gère les boutons de l'application.*
- void `initialiserCarte` ()
- *Initialise la carte.*
- void `initialiserLocalisation` ()
- *Initialise la localisation de la trottinette.*
- void `arreterLocalisation` ()
- void `initialiserCommunication` ()
- *Initialise les communication.*
- void `activerBluetooth` ()
- *Gère l'activation de l'adaptateur Bluetooth.*
- void `arreterBluetooth` ()
- *Fonction d'arrete du bluetooth.*
- void `afficherPositionTTE` ()
- *Fonction d'affichage de la position de la trottinette.*
- void `afficherDonneesTTE` (Boolean ok)
- *Fonction d'affichage des données de la TTE boolean pour activer l'affichage ou non.*
- int `getPosition` (String adresse)
- void `setBoutons` (int position)
- *Fonction pour changer la couleur et le positionnement de nos boutons.*
- void `remplacerPeripherique` (PeripheriqueBluetooth peripherique)

Attributs privés

- BluetoothAdapter `bluetoothAdapter` = null
- PeripheriqueBluetooth peripherique = null
- List< PeripheriqueBluetooth > `peripheriques`
- List< String > `noms`
- Set< BluetoothDevice > `devices`
- TrameTTE `trame`
- Trajet `trajet`
- Boolean `trajetEncours` = false
- Spinner `spinnerListePeripheriques`
- ListView `mListView`
- ArrayAdapter< String > `journal`
- List< String > `mListLog`
- TextView `affichageDistanceParcourue`
- TextView `affichageChargeBatterie`
- TextView `affichageVitesse`
- TextView `affichageDistanceParcourueTrajet`
- TextView `affichageConsommationTrajet`
- TextView `affichageVitesseMoyenneTrajet`
- Button `boutonCommunicationTTE`
- Button `boutonTrajet`
- MapView `myOpenMapView`
- ScaleBarOverlay `myScaleBarOverlay`
- CompassOverlay `mCompassOverlay`
- RotationGestureOverlay `mRotationGestureOverlay`
- DirectedLocationOverlay `myLocationOverlay`
- LocationManager `locationManager` = null
- String `fournisseur`
- Geocoder `geocoder`
- int `etatLocalisation`
- float `mAzimuthAngleSpeed` = 0.0f
- double `mSpeed` = 0.0
- final BroadcastReceiver `bluetoothReceiver`
- *permet d'écouter ce qui se passe sur le système et de rendre la tablette visible par les périphériques Bluetooth distants*
- final Handler `gestionnaire`
- *Gère les communications avec le thread de réception des trames.*

Attributs privés statiques

- static final String TAG = "<TEC 2018>"
- static final int REQUEST_CODE_ENABLE_BLUETOOTH = 0

10.1.1 Description détaillée**Auteur**

Hachette Alexandre

Version

1.0

10.1.2 Documentation des fonctions membres**10.1.2.1 com : :example : :iris : :myapplication.MainActivity : :activerBluetooth () [private]**

A faire changer en TEC-01

Références [com.example.iris.myapplication.MainActivity.bluetoothAdapter](#), [com.example.iris.myapplication.MainActivity.boutonCommunicationTTE](#), [com.example.iris.myapplication.MainActivity.devices](#), [com.example.iris.myapplication.MainActivity.gestionnaire](#), [com.example.iris.myapplication.MainActivity.noms](#), [com.example.iris.myapplication.MainActivity.peripherique](#), [com.example.iris.myapplication.MainActivity.peripheriques](#), [com.example.iris.myapplication.MainActivity.REQUEST_CODE_ENABLE_BLUETOOTH](#), [com.example.iris.myapplication.MainActivity.setBoutons\(\)](#), [com.example.iris.myapplication.MainActivity.spinnerListePeripheriques](#), et [com.example.iris.myapplication.MainActivity.TAG](#).

Référéncé par [com.example.iris.myapplication.MainActivity.initialiserCommunication\(\)](#).

```
{
    bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
    // A-t-on un adaptateur bluetooth ?
    if (bluetoothAdapter == null)
    {
        Toast.makeText(getApplicationContext(), "Bluetooth non présent sur
l'appareil!", Toast.LENGTH_SHORT).show();
        Log.d(TAG, "Bluetooth non présent !"); // d = debug
        boutonCommunicationTTE.setEnabled(false);
    }
    else
    {
        if (!bluetoothAdapter.isEnabled())
        {
            // le bluetooth est-il désactivé ?
            Log.d(TAG, "Demande activation Bluetooth"); // d = debug

            // Demande de l'activation du bluetooth
            Toast.makeText(getApplicationContext(), "Bluetooth non activé !
", Toast.LENGTH_SHORT).show();
            Intent activeBlueTooth = new Intent(BluetoothAdapter.
ACTION_REQUEST_ENABLE);
            startActivityForResult(activeBlueTooth,
REQUEST_CODE_ENABLE_BLUETOOTH);
        }
    }
}
```

```

else
{
    // le bluetooth est déjà activé
    Log.d(TAG, "Bluetooth activé"); // d = debug

    // Recherche des périphériques connus
    peripheriques = new ArrayList<PeripheriqueBluetooth>();
    noms = new ArrayList<String>();
    devices = bluetoothAdapter.getBondedDevices();

    for (BluetoothDevice blueDevice : devices)
    {
        if (blueDevice.getName().equals("HC-05"))
        {
            peripheriques.add(new PeripheriqueBluetooth(blueDevice,
gestionnaire));
            noms.add(blueDevice.getName());
        }
    }
    if(peripheriques.size() == 0)
    {
        peripheriques.add(new PeripheriqueBluetooth(null,
gestionnaire));
        Toast.makeText(getApplicationContext(), "Aucun périphérique
détecté ! ", Toast.LENGTH_SHORT).show();
    }
    else
    {
        boutonCommunicationTTE.setEnabled(true);
    }

    if(noms.size() == 0)
        noms.add("Aucun");

    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_item, noms);
    adapter.setDropDownViewResource(android.R.layout.
simple_spinner_dropdown_item);
    spinnerListePeripheriques.setAdapter(adapter);
    adapter.setNotifyOnChange(true);

    spinnerListePeripheriques.setOnItemClickListener(new
AdapterView.OnItemClickListener()
    {
        @Override
        public void onItemClick(AdapterView<?> arg0, View arg1,
int position, long id)
        {
            peripherique = peripheriques.get(position);
            setBoutons(position);
        }

        @Override
        public void onNothingSelected(AdapterView<?> arg0)
        {
        }
    });
}
}
}

```

10.1.2.2 com : :example : :iris : :myapplication.MainActivity : :afficherDonneesT-TE (Boolean ok) [private]

Paramètres

ok	
----	--

Références [com.example.iris.myapplication.MainActivity.affichageChargeBatterie](#),

com.example.iris.myapplication.MainActivity.affichageConsommationTrajet, com.example.iris.myapplication.MainActivity.affichageDistanceParcourue, com.example.iris.myapplication.MainActivity.affichageDistanceParcourueTrajet, com.example.iris.myapplication.MainActivity.affichageVitesse, com.example.iris.myapplication.MainActivity.affichageVitesseMoyenneTrajet, com.example.iris.myapplication.TrameTTE.getChargeBatterie(), com.example.iris.myapplication.Trajet.getConsommationTrajet(), com.example.iris.myapplication.Trajet.getDistanceParcourue(), com.example.iris.myapplication.TrameTTE.getDistanceParcourue(), com.example.iris.myapplication.TrameTTE.getVitesse(), com.example.iris.myapplication.Trajet.getVitesseMoyenneTTE(), com.example.iris.myapplication.MainActivity.trajet, com.example.iris.myapplication.MainActivity.trajetEncours, et com.example.iris.myapplication.MainActivity.trame.

Référencé par com.example.iris.myapplication.MainActivity.onResume().

```
{
    if(ok)
    {
        affichageChargeBatterie.setText("Batterie : " + trame.
getChargeBatterie() + " %");
        if(trame.getDistanceParcourue() > 1000)
        {
            affichageDistanceParcourue.setText("Distance : " + (trame.
getDistanceParcourue()/1000) + " km");
        }
        else
        {
            affichageDistanceParcourue.setText("Distance : " + trame.
getDistanceParcourue() + " m");
        }
        affichageVitesse.setText("Vitesse : " + trame.getVitesse() + " km/h
");

        affichageDistanceParcourueTrajet = (TextView) findViewById(R.id.
affichageDistanceParcourueTrajet);
        affichageConsommationTrajet = (TextView) findViewById(R.id.
affichageChargeBatterieTrajet);
        affichageVitesseMoyenneTrajet = (TextView) findViewById(R.id.
affichageVitesseTrajet);

        if(trajetEncours)
        {
            affichageConsommationTrajet.setText("Consommation : "+ trajet.
getConsommationTrajet(trame.getChargeBatterie()) + " %");
            affichageVitesseMoyenneTrajet.setText("Vitesse moy : " + trajet.
getVitesseMoyenneTTE() + " km/h");
            if(trajet.getDistanceParcourue(trame.getDistanceParcourue()) >
1000)
            {
                affichageDistanceParcourueTrajet.setText("Parcours : " + (
trajet.getDistanceParcourue(trame.getDistanceParcourue())/1000) + " km");
            }
            else
            {
                affichageDistanceParcourueTrajet.setText("Parcours : " +
trajet.getDistanceParcourue(trame.getDistanceParcourue()) + " m");
            }
        }
        else
        {
            affichageDistanceParcourueTrajet.setText("Vitesse moy. : --
km/h");
            affichageConsommationTrajet.setText("Consommation : -- %");
            affichageVitesseMoyenneTrajet.setText("Parcours : -- m");
        }
    }
    else
    {
        affichageChargeBatterie.setText("Batterie : -- %");
    }
}
```

```

        affichageDistanceParcourue.setText("Distance : -- m");
        affichageVitesse.setText("Vitesse : -- km/h");
        affichageDistanceParcourueTrajet.setText("Vitesse moy. : -- km/h");
        affichageConsommationTrajet.setText("Consommation : -- %");
        affichageVitesseMoyenneTrajet.setText("Parcours : -- m");
    }

}

```

10.1.2.3 `com : :example : :iris : :myapplication.MainActivity : :afficherPositionTTE` () [private]

Références [com.example.iris.myapplication.MainActivity.ecouteurGPS](#), [com.example.iris.myapplication.MainActivity.fournisseur](#), [com.example.iris.myapplication.MainActivity.locationManager](#), et [com.example.iris.myapplication.MainActivity.TAG](#).

```

{
    if (ActivityCompat.checkSelfPermission(this, Manifest.permission.
ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED && ActivityCompat.
checkSelfPermission(this, Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.
PERMISSION_GRANTED)
    {
        Log.d(TAG, "<GPS> no permissions !");
        return;
    }
    if (locationManager != null)
    {
        if (fournisseur != null)
        {
            Location localisation = locationManager.getLastKnownLocation(
fournisseur);
            if (localisation != null)
            {
                // on notifie la localisation
                ecouteurGPS.onLocationChanged(localisation);
            }
        }
    }
}
}

```

10.1.2.4 `com : :example : :iris : :myapplication.MainActivity : :arreterBluetooth` () [private]

Références [com.example.iris.myapplication.MainActivity.bluetoothAdapter](#), [com.example.iris.myapplication.MainActivity.bluetoothReceiver](#), [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.estConnecte\(\)](#), [com.example.iris.myapplication.MainActivity.peripherique](#), et [com.example.iris.myapplication.MainActivity.peripheriques](#).

Référencé par [com.example.iris.myapplication.MainActivity.onDestroy\(\)](#).

```

{
    PeripheriqueBluetooth peripherique;

    for(int i = 0; i < peripheriques.size(); i++)
    {
        peripherique = peripheriques.get(i);
        if(peripherique.estConnecte())
            peripherique.deconnecter(true);
        bluetoothAdapter.cancelDiscovery();
        unregisterReceiver(bluetoothReceiver);
    }
}

```

```
    }
}
```

10.1.2.5 `void com.example.iris.myapplication.MainActivity.arreterLocalisation ()` [private]

Références [com.example.iris.myapplication.MainActivity.ecouteurGPS](#), et [com.example.iris.myapplication.MainActivity.locationManager](#).

Référéncé par [com.example.iris.myapplication.MainActivity.onDestroy\(\)](#).

```
{
    if (locationManager != null)
    {
        locationManager.removeUpdates(ecouteurGPS);
        ecouteurGPS = null;
    }
}
```

10.1.2.6 `com : :example : :iris : :myapplication.MainActivity : :gererBoutons ()` [private]

Références [com.example.iris.myapplication.Trajet.arreterTrajet\(\)](#), [com.example.iris.myapplication.MainActivity.boutonCommunicationTTE](#), [com.example.iris.myapplication.MainActivity.boutonTrajet](#), [com.example.iris.myapplication.PeripheriqueBluetooth.connecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#), [com.example.iris.myapplication.Trajet.demarrerTrajet\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.estConnecte\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getAdresse\(\)](#), [com.example.iris.myapplication.TrameTTE.getChargeBatterie\(\)](#), [com.example.iris.myapplication.TrameTTE.getDistanceParcourue\(\)](#), [com.example.iris.myapplication.Trajet.getEtatTrajet\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getNom\(\)](#), [com.example.iris.myapplication.MainActivity.journaliser\(\)](#), [com.example.iris.myapplication.MainActivity.onClick\(\)](#), [com.example.iris.myapplication.MainActivity.peripherique](#), [com.example.iris.myapplication.MainActivity.trajet](#), [com.example.iris.myapplication.MainActivity.trajetEncours](#), et [com.example.iris.myapplication.MainActivity.trame](#).

Référéncé par [com.example.iris.myapplication.MainActivity.onCreate\(\)](#).

```
{
    boutonCommunicationTTE.setOnClickListener(
        new View.OnClickListener() {
            public void onClick(View v) {
                if (peripherique != null)
                {
                    if (! (peripherique.estConnecte()))
                    {
                        journaliser("==> Connexion " + peripherique.
getNom() + " [" + peripherique.getAdresse() + "]");
                    }
                    else
                    {
                        journaliser("==> Deconnexion " + peripherique.
getNom() + " [" + peripherique.getAdresse() + "]");
                    }
                    if (peripherique.estConnecte())
                    {
                        if (trajetEncours)
                        {

```

```

        // Arrêt
        trajet.arreterTrajet();
        trajetEncours = trajet.getEtatTrajet();
        journaliser("==> Arrêt du trajet");
        boutonTrajet.setBackgroundResource(R.drawable.trajetoff);

        }
        peripherique.deconnecter(true);
        boutonTrajet.setEnabled(false);

    }
    else
    {
        peripherique.connecter();
        boutonTrajet.setEnabled(true);
    }
}

);
boutonTrajet.setOnClickListener(
    new View.OnClickListener() {
        public void onClick(View v) {
            if (trajetEncours)
            {
                // Arrêt
                trajet.arreterTrajet();
                trajetEncours = trajet.getEtatTrajet();
                journaliser("==> Arrêt du trajet");
                boutonTrajet.setBackgroundResource(R.drawable.trajetoff);
            }
            else
            {
                // Départ
                journaliser("==> Démarrage du trajet");
                trajet = new Trajet(trame.getChargeBatterie(), trame
                .getDistanceParcourue());
                trajet.demarrerTrajet();
                trajetEncours = trajet.getEtatTrajet();
                boutonTrajet.setBackgroundResource(R.drawable.trajeton);
            }
        }
    }
);
}

```

10.1.2.7 `static String com.example.iris.myapplication.MainActivity.getApplicationName (Context context)` [static]

```

{
    int stringId = context.getApplicationInfo().labelRes;
    return context.getString(stringId);
}

```

10.1.2.8 `int com.example.iris.myapplication.MainActivity.getPosition (String adresse)` [private]

Références `com.example.iris.myapplication.PeripheriqueBluetooth.getAdresse()`, `com.example.iris.myapplication.MainActivity.peripherique`, et `com.example.iris.myapplication.MainActivity.peripheriques`.

```

{
    PeripheriqueBluetooth peripherique;
}

```

```

    for(int i = 0; i < peripheriques.size(); i++)
    {
        peripherique = peripheriques.get(i);
        if(peripherique.getAdresse() == adresse)
            return i;
    }
    return -1;
}

```

10.1.2.9 `com : :example : :iris : :myapplication.MainActivity : :initialiserActivite ()` [private]

Références `com.example.iris.myapplication.MainActivity.affichageChargeBatterie`, `com.example.iris.myapplication.MainActivity.affichageConsommationTrajet`, `com.example.iris.myapplication.MainActivity.affichageDistanceParcourue`, `com.example.iris.myapplication.MainActivity.affichageDistanceParcourueTrajet`, `com.example.iris.myapplication.MainActivity.affichageVitesse`, `com.example.iris.myapplication.MainActivity.affichageVitesseMoyenneTrajet`, `com.example.iris.myapplication.MainActivity.boutonCommunicationTTE`, `com.example.iris.myapplication.MainActivity.boutonTrajet`, `com.example.iris.myapplication.MainActivity.journal`, `com.example.iris.myapplication.MainActivity.mListLog`, `com.example.iris.myapplication.MainActivity.mListView`, et `com.example.iris.myapplication.MainActivity.spinnerListePeripheriques`.

Référéncé par `com.example.iris.myapplication.MainActivity.onCreate()`.

```

{
    setContentView(R.layout.activity_main_ihm);

    // Initialisation des ressources
    mListView = (ListView) findViewById(R.id.listview);
    boutonCommunicationTTE = (Button) findViewById(R.id.boutoncommunication
);
    boutonTrajet = (Button) findViewById(R.id.boutontrajet);
    spinnerListePeripheriques = (Spinner) findViewById(R.id.
spinnerListeAppareils);
    spinnerListePeripheriques.setContentDescription("La liste des
périphériques");
    affichageDistanceParcourue = (TextView) findViewById(R.id.
affichageDistanceParcourue);
    affichageChargeBatterie = (TextView) findViewById(R.id.
affichageChargeBatterie);
    affichageVitesse = (TextView) findViewById(R.id.affichageVitesse);
    affichageDistanceParcourueTrajet = (TextView) findViewById(R.id.
affichageDistanceParcourueTrajet);
    affichageConsommationTrajet = (TextView) findViewById(R.id.
affichageChargeBatterieTrajet);
    affichageVitesseMoyenneTrajet = (TextView) findViewById(R.id.
affichageVitesseTrajet);
    mListLog = new ArrayList<>();
    journal = new ArrayAdapter<>(this, R.layout.item, mListLog);
    mListView.setAdapter(journal);
}

```

10.1.2.10 `com : :example : :iris : :myapplication.MainActivity : :initialiserCarte ()` [private]

Références `com.example.iris.myapplication.MainActivity.mCompassOverlay`, `com.example.iris.myapplication.MainActivity.mRotationGestureOverlay`, `com.example.iris.myapplication.MainActivity.myLocationOverlay`, `com.example.iris.myapplication.MainActivity.myOpenMapView`, et `com.example.iris.myapplication.MainActivity.myScale`.

`BarOverlay`, `com.example.iris.myapplication.MainActivity.TAG`, et `com.example.iris.myapplication.MainActivity.traceTrajet`.

Référencé par `com.example.iris.myapplication.MainActivity.onCreate()`.

```
{
    myOpenMapView = (MapView) findViewById(R.id.carte);
    myOpenMapView.setBuiltInZoomControls(true);
    myOpenMapView.setClickable(true);
    myOpenMapView.getController().setZoom(18);

    myScaleBarOverlay = new ScaleBarOverlay(myOpenMapView);
    myOpenMapView.getOverlays().add(myScaleBarOverlay);

    mCompassOverlay = new CompassOverlay(getApplicationContext(), new
    InternalCompassOrientationProvider(getApplicationContext()), myOpenMapView);
    mCompassOverlay.enableCompass();
    myOpenMapView.getOverlays().add(mCompassOverlay);

    mRotationGestureOverlay = new RotationGestureOverlay(
    getApplicationContext(), myOpenMapView);
    mRotationGestureOverlay.setEnabled(true);
    myOpenMapView.setMultiTouchControls(true);
    myOpenMapView.getOverlays().add(this.mRotationGestureOverlay);

    myLocationOverlay = new DirectedLocationOverlay(this);
    myLocationOverlay.setEnabled(false);
    myOpenMapView.getOverlays().add(myLocationOverlay);

    traceTrajet = new ArrayList<GeoPoint>();

    Log.d(TAG, "Carte initialisée"); // d = debug
}
```

10.1.2.11 `com : :example : :iris : :myapplication.MainActivity : :initialiser-Communication ()` [private]

Références `com.example.iris.myapplication.MainActivity.activerBluetooth()`, et `com.example.iris.myapplication.MainActivity.trame`.

Référencé par `com.example.iris.myapplication.MainActivity.onCreate()`.

```
{
    activerBluetooth();

    trame = new TrameTTE();
}
```

10.1.2.12 `com : :example : :iris : :myapplication.MainActivity : :initialiser-Localisation ()` [private]

Références `com.example.iris.myapplication.MainActivity.ecouteurGPS`, `com.example.iris.myapplication.MainActivity.fournisseur`, `com.example.iris.myapplication.MainActivity.geocoder`, `com.example.iris.myapplication.MainActivity.locationManager`, et `com.example.iris.myapplication.MainActivity.TAG`.

Référencé par `com.example.iris.myapplication.MainActivity.onCreate()`.

```
{
    if (locationManager == null)
    {
```

```

        geocoder = new Geocoder(this, Locale.getDefault());
        locationManager = (LocationManager) getApplicationContext().
getSystemService(Context.LOCATION_SERVICE);
        Criteria criteres = new Criteria();

        // la précision : (ACCURACY_FINE pour une haute précision ou
        ACCURACY_COARSE pour une moins bonne précision)
        criteres.setAccuracy(Criteria.ACCURACY_FINE);

        // l'altitude
        criteres.setAltitudeRequired(true);

        // la direction
        criteres.setBearingRequired(true);

        // la vitesse
        criteres.setSpeedRequired(true);

        // la consommation d'énergie demandée
        criteres.setCostAllowed(true);
        criteres.setPowerRequirement(Criteria.POWER_MEDIUM);

        fournisseur = locationManager.getBestProvider(criteres, true);
        Log.d(TAG, "Fournisseur : " + fournisseur);
    }

    if (fournisseur != null)
    {
        // dernière position connue
        if (ActivityCompat.checkSelfPermission(this, Manifest.permission.
ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED && ActivityCompat.
checkSelfPermission(this, Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED)
        {
            Log.d(TAG, "<GPS> no permissions !");
            return;
        }

        Location localisation = locationManager.getLastKnownLocation(
fournisseur);
        if (localisation != null)
        {
            // on notifie la localisation
            ecouteurGPS.onLocationChanged(localisation);
        }

        // on configure la mise à jour automatique : au moins 1 mètres et 5
        secondes
        locationManager.requestLocationUpdates(fournisseur, 5000, 1,
ecouteurGPS);
    }
}

```

10.1.2.13 `com : :example : :iris : :myapplication.MainActivity : :journaliser (String text)`

Références [com.example.iris.myapplication.MainActivity.journal](#), [com.example.iris.-myapplication.MainActivity.mListLog](#), et [com.example.iris.myapplication.MainActivity.mListView](#).

Référéncé par [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#), [com.-example.iris.myapplication.MainActivity.onCreate\(\)](#), [com.example.iris.myapplication.-MainActivity.onDestroy\(\)](#), et [com.example.iris.myapplication.MainActivity.setBoutons\(\)](#).

```

{
    mListLog.add(text);
    journal.notifyDataSetChanged();
    mListview.setSelection(mListview.getCount() - 1);
}

```

```
}
```

10.1.2.14 `com : :example : :iris : :myapplication.MainActivity : onActivityResult (`
`int requestCode, int resultCode, Intent data)` `[protected]`

Paramètres

<i>resultCode</i>	
<i>requestCode</i>	
<i>data</i>	

Références [com.example.iris.myapplication.MainActivity.REQUEST_CODE_ENABLE_BLUETOOTH](#).

```
{
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode != REQUEST_CODE_ENABLE_BLUETOOTH)
        return;
    if (resultCode == RESULT_OK)
    {
        Toast.makeText(getApplicationContext(), "Bluetooth activé", Toast.
LENGTH_SHORT).show();
    }
    else
    {
        Toast.makeText(getApplicationContext(), "Bluetooth non activé !",
Toast.LENGTH_SHORT).show();
    }
}
```

10.1.2.15 `void com.example.iris.myapplication.MainActivity.onClick (View view)`

Référencé par [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#).

```
{
}
```

10.1.2.16 `com : :example : :iris : :myapplication.MainActivity : onCreate (Bundle`
`savedInstanceState)` `[protected]`

Paramètres

<i>saved- InstanceState</i>	
---------------------------------	--

Références [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#), [com.-example.iris.myapplication.MainActivity.initialiserActivite\(\)](#), [com.example.iris.myapplication.MainActivity.initialiserCarte\(\)](#), [com.example.iris.myapplication.MainActivity.initialiserCommunication\(\)](#), [com.example.iris.myapplication.MainActivity.initialiserLocalisation\(\)](#), [com.example.iris.myapplication.MainActivity.journaliser\(\)](#), et [com.example.iris.-myapplication.MainActivity.TAG](#).

```
{
```

```

super.onCreate(savedInstanceState);

initialiserActivite();

Log.d(TAG, "Démarrage TEC 2018");
journaliser("==> Démarrage TEC 2018");

initialiserCommunication();

gererBoutons();

initialiserCarte();

initialiserLocalisation();
}

```

10.1.2.17 `boolean com.example.iris.myapplication.MainActivity.onCreateOptionsMenu (Menu menu)`

```

{
    // Inflate the menu; this adds items to the action bar if it is
    present.
    getMenuInflater().inflate(R.menu.menu_main, menu);
    return true;
}

```

10.1.2.18 `void com.example.iris.myapplication.MainActivity.onDestroy ()` [protected]

Références [com.example.iris.myapplication.MainActivity.arreterBluetooth\(\)](#), [com.example.iris.myapplication.MainActivity.arreterLocalisation\(\)](#), [com.example.iris.myapplication.MainActivity.journaliser\(\)](#), et [com.example.iris.myapplication.MainActivity.TAG](#).

```

{
    Log.d(TAG, "Arrêt TEC 2018");
    journaliser("==> Arrêt TEC 2018");

    arreterBluetooth();
    arreterLocalisation();
    super.onDestroy();
}

```

10.1.2.19 `boolean com.example.iris.myapplication.MainActivity.onOptionsItemSelected (MenuItem item)`

```

{
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings)
    {
        return true;
    }

    return super.onOptionsItemSelected(item);
}

```

10.1.2.20 `void com.example.iris.myapplication.MainActivity.onResume ()`

Références [com.example.iris.myapplication.MainActivity.afficherDonneesTTE\(\)](#).

```
{
    super.onResume();
    //Log.d(TAG, "onResume()");
    afficherDonneesTTE(false);
}
```

10.1.2.21 `void com.example.iris.myapplication.MainActivity.remplacer-
Peripherique (PeripheriqueBluetooth peripherique)
[private]`

Références [com.example.iris.myapplication.MainActivity.gestionnaire](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getDevice\(\)](#), et [com.example.iris.myapplication.MainActivity.peripheriques](#).

```
{
    BluetoothDevice device = peripherique.getDevice();
    boolean res = Collections.replaceAll(peripheriques, peripherique, new
    PeripheriqueBluetooth(device, gestionnaire));
}
```

10.1.2.22 `com : :example : :iris : :myapplication.MainActivity : :setBoutons (int
position) [private]`

Paramètres

<i>position</i>	
-----------------	--

Références [com.example.iris.myapplication.MainActivity.boutonCommunicationTTE](#), [com.example.iris.myapplication.PeripheriqueBluetooth.estConnecte\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getAdresse\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getNom\(\)](#), [com.example.iris.myapplication.MainActivity.journaliser\(\)](#), [com.example.iris.myapplication.MainActivity.peripherique](#), et [com.example.iris.myapplication.MainActivity.peripheriques](#).

Référencé par [com.example.iris.myapplication.MainActivity.activerBluetooth\(\)](#).

```
{
    PeripheriqueBluetooth peripherique = peripheriques.get(position);

    if(peripherique.estConnecte())
    {
        boutonCommunicationTTE.setBackgroundResource(R.drawable.tteon);
        journaliser("==> Périphérique sélectionné " + peripherique.getNom(
) + " [" + peripherique.getAdresse() + "] : connecté");
    }
    else
    {
        boutonCommunicationTTE.setBackgroundResource(R.drawable.tteoff);
        journaliser("==> Périphérique sélectionné " + peripherique.getNom(
) + " [" + peripherique.getAdresse() + "] : déconnecté");
    }
}
```

10.1.3 Documentation des données membres

10.1.3.1 `TextView` `com.example.iris.myapplication.MainActivity.affichageCharge-Batterie` [private]

pourcentage de charge de la trottinette

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, et `com.example.iris.myapplication.MainActivity.initialiserActivite()`.

10.1.3.2 `TextView` `com.example.iris.myapplication.MainActivity.affichage-ConsommationTrajet` [private]

consommation en pourcentage de la trottinette

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, et `com.example.iris.myapplication.MainActivity.initialiserActivite()`.

10.1.3.3 `TextView` `com.example.iris.myapplication.MainActivity.affichageDistance-Parcourue` [private]

distance parcourue par la trottinette en m

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, et `com.example.iris.myapplication.MainActivity.initialiserActivite()`.

10.1.3.4 `TextView` `com.example.iris.myapplication.MainActivity.affichageDistance-ParcourueTrajet` [private]

distance parcourue par la trottinette en m

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, et `com.example.iris.myapplication.MainActivity.initialiserActivite()`.

10.1.3.5 `TextView` `com.example.iris.myapplication.MainActivity.affichageVitesse` [private]

vitesse de la trottinette en km/h

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, et `com.example.iris.myapplication.MainActivity.initialiserActivite()`.

10.1.3.6 `TextView` `com.example.iris.myapplication.MainActivity.affichageVitesse-MoyenneTrajet` [private]

vitesse moyenne de la trottinette en km/h

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, et `com.example.iris.myapplication.MainActivity.initialiserActivite()`.

10.1.3.7 `BluetoothAdapter` `com.example.iris.myapplication.MainActivity.bluetooth-Adapter` = null [private]

l'adaptateur Bluetooth local

Référencé par `com.example.iris.myapplication.MainActivity.activerBluetooth()`, et `com.example.iris.myapplication.MainActivity.arreterBluetooth()`.

10.1.3.8 final BroadcastReceiver `com.example.iris.myapplication.MainActivity.bluetoothReceiver` [private]

Valeur initiale :

```
new BroadcastReceiver()
{
    public void onReceive(Context context, Intent intent)
    {
        String action = intent.getAction();
        if (BluetoothDevice.ACTION_FOUND.equals(action))
        {
            BluetoothDevice device = intent.getParcelableExtra(
                BluetoothDevice.EXTRA_DEVICE);
            Toast.makeText(getApplicationContext(), "Nouveau périphérique : "
                + device.getName(), Toast.LENGTH_SHORT).show();
            bluetoothAdapter.startDiscovery();
        }
    }
}
```

Référencé par `com.example.iris.myapplication.MainActivity.arreterBluetooth()`.

10.1.3.9 Button `com.example.iris.myapplication.MainActivity.boutonCommunicationTTE` [private]

Référencé par `com.example.iris.myapplication.MainActivity.activerBluetooth()`, `com.example.iris.myapplication.MainActivity.gererBoutons()`, `com.example.iris.myapplication.MainActivity.initialiserActivite()`, et `com.example.iris.myapplication.MainActivity.setBoutons()`.

10.1.3.10 Button `com.example.iris.myapplication.MainActivity.boutonTrajet` [private]

Référencé par `com.example.iris.myapplication.MainActivity.gererBoutons()`, et `com.example.iris.myapplication.MainActivity.initialiserActivite()`.

10.1.3.11 Set<BluetoothDevice> `com.example.iris.myapplication.MainActivity.devices` [private]

Référencé par `com.example.iris.myapplication.MainActivity.activerBluetooth()`.

10.1.3.12 LocationListener `com.example.iris.myapplication.MainActivity.ecouteurGPS` [package]

Référencé par `com.example.iris.myapplication.MainActivity.afficherPositionTTE()`, `com.example.iris.myapplication.MainActivity.arreterLocalisation()`, et `com.example.iris.myapplication.MainActivity.initialiserLocalisation()`.

10.1.3.13 int `com.example.iris.myapplication.MainActivity.etatLocalisation` [private]

10.1.3.14 String com.example.iris.myapplication.MainActivity.fournisseur
[private]

Référencé par [com.example.iris.myapplication.MainActivity.afficherPositionTTE\(\)](#), et [com.example.iris.myapplication.MainActivity.initialiserLocalisation\(\)](#).

10.1.3.15 Geocoder com.example.iris.myapplication.MainActivity.geocoder
[private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserLocalisation\(\)](#).

10.1.3.16 final Handler com.example.iris.myapplication.MainActivity.gestionnaire
[private]

Référencé par [com.example.iris.myapplication.MainActivity.activerBluetooth\(\)](#), et [com.example.iris.myapplication.MainActivity.remplacerPeripherique\(\)](#).

10.1.3.17 ArrayAdapter<String> com.example.iris.myapplication.MainActivity.-journal [private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserActivite\(\)](#), et [com.example.iris.myapplication.MainActivity.journaliser\(\)](#).

10.1.3.18 LocationManager com.example.iris.myapplication.MainActivity.location-Manager = null [private]

Référencé par [com.example.iris.myapplication.MainActivity.afficherPositionTTE\(\)](#), [com.example.iris.myapplication.MainActivity.arreterLocalisation\(\)](#), et [com.example.iris.myapplication.MainActivity.initialiserLocalisation\(\)](#).

10.1.3.19 float com.example.iris.myapplication.MainActivity.mAzimuthAngle-Speed = 0.0f [private]

10.1.3.20 CompassOverlay com.example.iris.myapplication.MainActivity.m-CompassOverlay [private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserCarte\(\)](#).

10.1.3.21 List<String> com.example.iris.myapplication.MainActivity.mListLog
[private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserActivite\(\)](#), et [com.example.iris.myapplication.MainActivity.journaliser\(\)](#).

10.1.3.22 ListView com.example.iris.myapplication.MainActivity.mListView
[private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserActivite\(\)](#), et [com.example.iris.myapplication.MainActivity.journaliser\(\)](#).

10.1.3.23 **RotationGestureOverlay** **com.example.iris.myapplication.MainActivity.m-RotationGestureOverlay** [private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserCarte\(\)](#).

10.1.3.24 **double** **com.example.iris.myapplication.MainActivity.mSpeed** = 0.0 [private]

10.1.3.25 **DirectedLocationOverlay** **com.example.iris.myapplication.MainActivity.my-LocationOverlay** [private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserCarte\(\)](#).

10.1.3.26 **MapView** **com.example.iris.myapplication.MainActivity.myOpenMapView** [private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserCarte\(\)](#).

10.1.3.27 **ScaleBarOverlay** **com.example.iris.myapplication.MainActivity.myScale-BarOverlay** [private]

Référencé par [com.example.iris.myapplication.MainActivity.initialiserCarte\(\)](#).

10.1.3.28 **List<String>** **com.example.iris.myapplication.MainActivity.noms** [private]

Référencé par [com.example.iris.myapplication.MainActivity.activerBluetooth\(\)](#).

10.1.3.29 **PeripheriqueBluetooth** **com.example.iris.myapplication.MainActivity.-peripherique** = null [private]

le périphérique Bluetooth distant

Référencé par [com.example.iris.myapplication.MainActivity.activerBluetooth\(\)](#), [com.example.iris.myapplication.MainActivity.arreterBluetooth\(\)](#), [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#), [com.example.iris.myapplication.MainActivity.getPosition\(\)](#), et [com.example.iris.myapplication.MainActivity.setBoutons\(\)](#).

10.1.3.30 **List<PeripheriqueBluetooth>** **com.example.iris.myapplication.MainActivity.peripheriques** [private]

liste de périphériques Bluetooth appairés

Référencé par [com.example.iris.myapplication.MainActivity.activerBluetooth\(\)](#), [com.example.iris.myapplication.MainActivity.arreterBluetooth\(\)](#), [com.example.iris.myapplication.MainActivity.getPosition\(\)](#), [com.example.iris.myapplication.MainActivity.remplacerPeripherique\(\)](#), et [com.example.iris.myapplication.MainActivity.setBoutons\(\)](#).

10.1.3.31 `final int com.example.iris.myapplication.MainActivity.REQUEST_CODE_ENABLE_BLUETOOTH = 0` [static, private]

Référencé par `com.example.iris.myapplication.MainActivity.activerBluetooth()`, et `com.example.iris.myapplication.MainActivity.onActivityResult()`.

10.1.3.32 `Spinner com.example.iris.myapplication.MainActivity.spinnerListePeripheriques` [private]

liste déroulante de périphériques Bluetooth appairés

Référencé par `com.example.iris.myapplication.MainActivity.activerBluetooth()`, et `com.example.iris.myapplication.MainActivity.initialiserActivite()`.

10.1.3.33 `final String com.example.iris.myapplication.MainActivity.TAG = "<TEC 2018>"` [static, private]

Référencé par `com.example.iris.myapplication.MainActivity.activerBluetooth()`, `com.example.iris.myapplication.MainActivity.afficherPositionTTE()`, `com.example.iris.myapplication.MainActivity.initialiserCarte()`, `com.example.iris.myapplication.MainActivity.initialiserLocalisation()`, `com.example.iris.myapplication.MainActivity.onCreate()`, et `com.example.iris.myapplication.MainActivity.onDestroy()`.

10.1.3.34 `ArrayList<GeoPoint> com.example.iris.myapplication.MainActivity.traceTrajet` [package]

Référencé par `com.example.iris.myapplication.MainActivity.initialiserCarte()`.

10.1.3.35 `Trajet com.example.iris.myapplication.MainActivity.trajet` [private]

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, et `com.example.iris.myapplication.MainActivity.gererBoutons()`.

10.1.3.36 `Boolean com.example.iris.myapplication.MainActivity.trajetEncours = false` [private]

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, et `com.example.iris.myapplication.MainActivity.gererBoutons()`.

10.1.3.37 `TrameTTE com.example.iris.myapplication.MainActivity.trame` [private]

pour le décodage des trames de la trottinette

Référencé par `com.example.iris.myapplication.MainActivity.afficherDonneesTTE()`, `com.example.iris.myapplication.MainActivity.gererBoutons()`, et `com.example.iris.myapplication.MainActivity.initialiserCommunication()`.

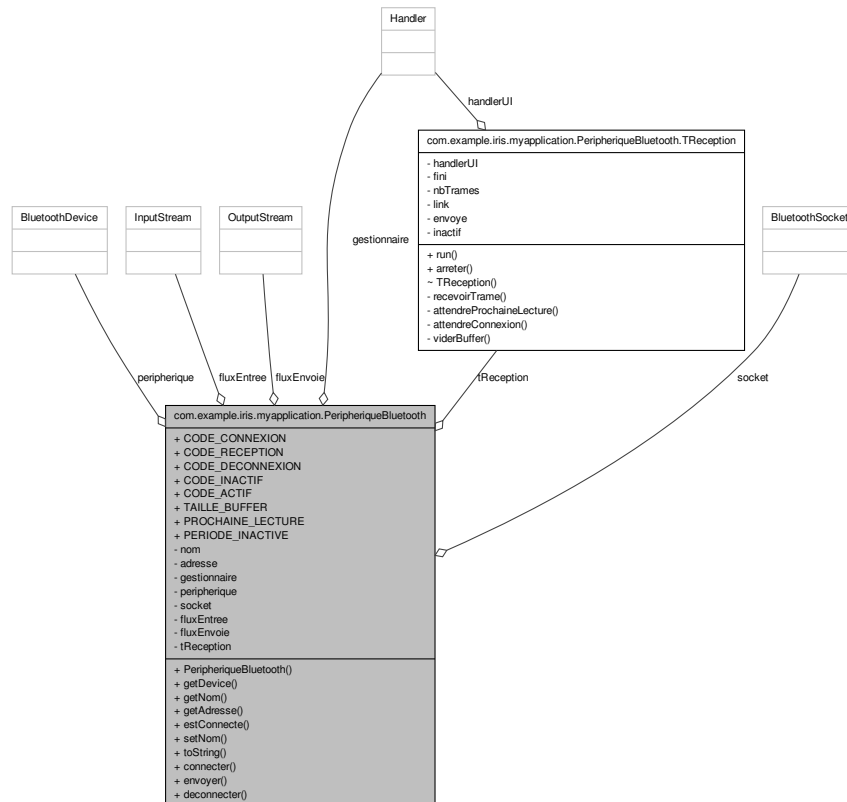
La documentation de cette classe a été générée à partir du fichier suivant :

– [MainActivity.java](#)

10.2 Référence de la classe com.example.iris.myapplication.PeripheriqueBluetooth

Assure le dialogue bluetooth entre la trottinette et le telephone.

Graphe de collaboration de com.example.iris.myapplication.PeripheriqueBluetooth :



Classes

- class `TReception`
Thread de réception des trames en provenance de la Trottinette via le Bluetooth.

Fonctions membres publiques

- `PeripheriqueBluetooth` (`BluetoothDevice peripherique`, `Handler gestionnaire`)
Constructeur public de la classe `PeripheriqueBluetooth`.
- `BluetoothDevice getDevice ()`
Permet d'obtenir le peripherique.
- `String getNom ()`
Permet d'obtenir le nom du Périphérique Bluetooth.

- String `getAdresse` ()
Permet d'obtenir l'adresse du périphérique Bluetooth.
- boolean `estConnecte` ()
Permet de connaître si la connexion a été effectuée.
- void `setNom` (String `nom`)
Permet de modifier le nom du périphérique Bluetooth.
- String `toString` ()
Permet d'obtenir le nom et l'adresse du périphérique.
- void `connecter` ()
Connecte la socket Bluetooth et démarre le thread Réception.
- void `envoyer` (String `data`)
Envoyer des trames a la carte.
- boolean `deconnecter` (boolean `fermeture`)
Déconnecte la socket Bluetooth et arrête le thread Réception.

Attributs publics statiques

- static final int `CODE_CONNEXION` = 0
- static final int `CODE_RECEPTION` = 1
- static final int `CODE_DECONNEXION` = 2
- static final int `CODE_INACTIF` = 3
- static final int `CODE_ACTIF` = 4
- static final int `TAILLE_BUFFER` = 100
- static final int `PROCHAINE_LECTURE` = 1000
- static final int `PERIODE_INACTIVE` = 200000

Attributs privés

- String `nom`
- String `adresse`
- Handler `gestionnaire` = null
- BluetoothDevice `peripherique` = null
- BluetoothSocket `socket` = null
- InputStream `fluxEntree` = null
- OutputStream `fluxEnvoie` = null
- `TReception tReception`

10.2.1 Description détaillée

Auteur

Thierry Vaira <tvaira@free.fr>
HACHETTE Alexandre

10.2.2 Documentation des constructeurs et destructeur

10.2.2.1 com.example.iris.myapplication.PeripheriqueBluetooth.- PeripheriqueBluetooth (BluetoothDevice `peripherique`, Handler `gestionnaire`)

Paramètres

<code>peripherique</code>	un objet de type BluetoothDevice
<code>gestionnaire</code>	un objet de type Handler

Références [com.example.iris.myapplication.PeripheriqueBluetooth.adresse](#), [com.example.iris.myapplication.PeripheriqueBluetooth.fluxEntree](#), [com.example.iris.myapplication.PeripheriqueBluetooth.fluxEnvoie](#), [com.example.iris.myapplication.PeripheriqueBluetooth.gestionnaire](#), [com.example.iris.myapplication.PeripheriqueBluetooth.nom](#), [com.example.iris.myapplication.PeripheriqueBluetooth.peripherique](#), [com.example.iris.myapplication.PeripheriqueBluetooth.socket](#), [com.example.iris.myapplication.PeripheriqueBluetooth.toString\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.tReception](#).

```
{
    if(peripherique != null)
    {
        this.peripherique = peripherique;
        this.nom = peripherique.getName();
        this.adresse = peripherique.getAddress();
        this.gestionnaire = gestionnaire;

        try
        {
            System.out.println("<Bluetooth> nom " + peripherique.getName());

;
            ParcelUuid[] uuids = peripherique.getUuids();
            if(uuids != null)
            {
                for (int i = 0; i < uuids.length; i++)
                {
                    System.out.println("<Bluetooth> uuid " + uuids[i].
getUuid().toString());
                }
            }
            else
            {
                System.out.println("<Bluetooth> uuid null !");
            }
            //socket =
            device.createRfcommSocketToServiceRecord(uuids[0].getUuid());
            System.out.println("<Bluetooth> new socket");

            socket = peripherique.createRfcommSocketToServiceRecord(UUID.
fromString("00001101-0000-1000-8000-00805F9B34FB"));
            fluxEntree = socket.getInputStream();
            fluxEnvoie = socket.getOutputStream();
        }
        catch (IOException e)
        {
            e.printStackTrace();
            System.out.println("<Bluetooth> Erreur socket !");
            socket = null;
        }
    }
    else
    {
        this.peripherique = peripherique;
        this.nom = "Aucun";
        this.adresse = "";
        this.gestionnaire = gestionnaire;
    }

    if(socket != null) {
        tReception = new TReception(gestionnaire);
    }
}
```

10.2.3 Documentation des fonctions membres

10.2.3.1 void com.example.iris.myapplication.PeripheriqueBluetooth.connecter ()

Auteur

Thierry Vaira <tvaira@free.fr>

Références [com.example.iris.myapplication.PeripheriqueBluetooth.CODE_CONNEXION](#), [com.example.iris.myapplication.PeripheriqueBluetooth.envoyer\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.gestionnaire](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getAdresse\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getNom\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.socket](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.tReception](#).

Référéncé par [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#).

```

{
    /* démarre le thread connexion */
    new Thread()
    {
        @Override public void run()
        {
            while(!socket.isConnected())
            {
                try
                {
                    System.out.println("<Bluetooth> socket connect ...");//
debug

                    /* Demande de connexion */
                    socket.connect();

                    System.out.println("<Bluetooth> socket connect ok");
                    /* connecté ? */
                    if (socket.isConnected())
                    {
                        /* informe l'activité principale */
                        Message message = Message.obtain();
                        message.what = PeripheriqueBluetooth.CODE_CONNEXION
;
                        Bundle paquet = new Bundle();
                        paquet.putString("nom", getNom());
                        paquet.putString("adresse", getAdresse());
                        paquet.putInt("etat", CODE_CONNEXION);
                        paquet.putString("donnees", "");
                        message.setData(paquet);
                        gestionnaire.sendMessage(message);
                        //
                        peripherique.envoyer(etMessage.getText().toString() + "\r\n");
                        envoyer("$TECL,A*");
                        /* démarre le thread réception */
                        tReception.start();
                    }
                    else
                    {
                        Log.e("connecter()", "Socket connecte : " + socket.
isConnected()); // e = error
                    }
                } catch (IOException e)
                {
                    System.out.println("<Socket> Erreur connect");
                    e.printStackTrace();
                }
            }
        }
    }.start();
}

```

10.2.3.2 boolean com.example.iris.myapplication.PeripheriqueBluetooth. deconnecter (boolean fermeture)

Auteur

Thierry Vaira <tvaira@free.fr>

Références [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.arreter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.CODE_DECONNEXION](#), [com.example.iris.myapplication.PeripheriqueBluetooth.estConnecte\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.gestionnaire](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getAdresse\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getNom\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.socket](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.tReception](#).

Référencé par [com.example.iris.myapplication.MainActivity.arreterBluetooth\(\)](#), et [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#).

```
{
    try
    {
        if(estConnecte())
        {
            tReception.arreter();

            if (fermeture) {
                socket.close();
                System.out.println("<Bluetooth> socket close");
                Message msg = Message.obtain();
                Bundle b = new Bundle();
                b.putString("nom", getNom());
                b.putString("adresse", getAdresse());
                b.putInt("etat", CODE_DECONNEXION);
                b.putString("donnees", "");
                msg.setData(b);
                gestionnaire.sendMessage(msg);
            }
            return true;
        }
    }
    catch (IOException e)
    {
        System.out.println("<Bluetooth> Erreur close");
        e.printStackTrace();
        return false;
    }
    return false;
}
```

10.2.3.3 void com.example.iris.myapplication.PeripheriqueBluetooth.envoyer (String data)

Références [com.example.iris.myapplication.PeripheriqueBluetooth.estConnecte\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.fluxEnvoie](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.socket](#).

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.connecter\(\)](#).

```
{
    if(socket == null)
        return;
```

```
try
{
    if(estConnecte())
    {
        System.out.println("<Bluetooth> Envoyer " + data);
        fluxEnvoie.write(data.getBytes());
        fluxEnvoie.flush();
    }
}
catch (IOException e)
{
    System.out.println("<Bluetooth> Erreur socket write !");
    e.printStackTrace();
}
```

10.2.3.4 boolean **com.example.iris.myapplication.PeripheriqueBluetooth.est-Connecte ()**

Renvoie

un booléen qui vaut true si la connection existe, et faux sinon

Références [com.example.iris.myapplication.PeripheriqueBluetooth.socket](#).

Référencé par [com.example.iris.myapplication.MainActivity.arreterBluetooth\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.envoyer\(\)](#), [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#), et [com.example.iris.myapplication.MainActivity.setBoutons\(\)](#).

```
{
    if(socket != null)
        return socket.isConnected();
    return false;
}
```

10.2.3.5 String **com.example.iris.myapplication.PeripheriqueBluetooth.get-Adresse ()**

Renvoie

un String qui représente l'adresse du périphérique

Références [com.example.iris.myapplication.PeripheriqueBluetooth.adresse](#).

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.connecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#), [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#), [com.example.iris.myapplication.MainActivity.getPosition\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.T-Reception.recevoirTrame\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.T-Reception.run\(\)](#), et [com.example.iris.myapplication.MainActivity.setBoutons\(\)](#).

```
{
    return adresse;
}
```

10.2.3.6 BluetoothDevice **com.example.iris.myapplication.PeripheriqueBluetooth.-getDevice ()**

Renvoie

un String qui représente le peripherique

Références [com.example.iris.myapplication.PeripheriqueBluetooth.peripherique](#).

Référencé par [com.example.iris.myapplication.MainActivity.remplacerPeripherique\(\)](#).

```
{  
    return peripherique;  
}
```

10.2.3.7 String **com.example.iris.myapplication.PeripheriqueBluetooth.getNom ()**

Renvoie

un String qui représente le nom du périphérique

Références [com.example.iris.myapplication.PeripheriqueBluetooth.nom](#).

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.connecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#), [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.recevoirTrame\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.run\(\)](#), et [com.example.iris.myapplication.MainActivity.setBoutons\(\)](#).

```
{  
    return nom;  
}
```

10.2.3.8 void **com.example.iris.myapplication.PeripheriqueBluetooth.setNom (String nom)**

Paramètres

<i>nom</i>	un String qui représente le nom du périphérique
------------	---

Références [com.example.iris.myapplication.PeripheriqueBluetooth.nom](#).

```
{  
    this.nom = nom;  
}
```

10.2.3.9 String **com.example.iris.myapplication.PeripheriqueBluetooth.toString ()**

Renvoie

un String

Références [com.example.iris.myapplication.PeripheriqueBluetooth.adresse](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.nom](#).

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#).

```
{  
    return "\nNom : " + nom + "\nAdresse : " + adresse;  
}
```

10.2.4 Documentation des données membres

10.2.4.1 String com.example.iris.myapplication.PeripheriqueBluetooth.adresse [private]

Référéncé par [com.example.iris.myapplication.PeripheriqueBluetooth.getAdresse\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.toString\(\)](#).

10.2.4.2 final int com.example.iris.myapplication.PeripheriqueBluetooth.CODE_A- CTIF = 4 [static]

Référéncé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#).

10.2.4.3 final int com.example.iris.myapplication.PeripheriqueBluetooth.CODE_C- ONNEXION = 0 [static]

Référéncé par [com.example.iris.myapplication.PeripheriqueBluetooth.connecter\(\)](#).

10.2.4.4 final int com.example.iris.myapplication.PeripheriqueBluetooth.CODE_D- ECONNEXION = 2 [static]

Référéncé par [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#).

10.2.4.5 final int com.example.iris.myapplication.PeripheriqueBluetooth.CODE_IN- ACTIF = 3 [static]

Référéncé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#).

10.2.4.6 final int com.example.iris.myapplication.PeripheriqueBluetooth.CODE_R- ECEPTION = 1 [static]

Référéncé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-recevoirTrame\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-run\(\)](#).

10.2.4.7 InputStream com.example.iris.myapplication.PeripheriqueBluetooth.flux- Entree = null [private]

Référéncé par [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-recevoirTrame\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-run\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.viderBuffer\(\)](#).

10.2.4.8 **OutputStream** **com.example.iris.myapplication.PeripheriqueBluetooth.flux-Envoie** = null [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.envoyer\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#).

10.2.4.9 **Handler** **com.example.iris.myapplication.PeripheriqueBluetooth.-gestionnaire** = null [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.-connecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#).

10.2.4.10 **String** **com.example.iris.myapplication.PeripheriqueBluetooth.nom** [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.getNom\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.setNom\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.toString\(\)](#).

10.2.4.11 **final int** **com.example.iris.myapplication.PeripheriqueBluetooth.PERIOD-E_INACTIVE** = 200000 [static]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#).

10.2.4.12 **BluetoothDevice** **com.example.iris.myapplication.PeripheriqueBluetooth.-peripherique** = null [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.getDevice\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#).

10.2.4.13 **final int** **com.example.iris.myapplication.PeripheriqueBluetooth.PROCH-AINE_LECTURE** = 1000 [static]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.recevoirTrame\(\)](#).

10.2.4.14 **BluetoothSocket** **com.example.iris.myapplication.PeripheriqueBluetooth.-socket** = null [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreConnexion\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.connecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.envoyer\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.estConnecte\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.recevoirTrame\(\)](#), et [com.example.iris.myapplication.Peripherique-](#)

[Bluetooth.TReception.viderBuffer\(\)](#).

10.2.4.15 final int [com.example.iris.myapplication.PeripheriqueBluetooth.TAILLE_BUFFER](#) = 100 [static]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.recevoirTrame\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.viderBuffer\(\)](#).

10.2.4.16 **TReception** [com.example.iris.myapplication.PeripheriqueBluetooth.tReception](#) [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.connecter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.PeripheriqueBluetooth\(\)](#).

La documentation de cette classe a été générée à partir du fichier suivant :

– [PeripheriqueBluetooth.java](#)

10.3 Référence de la classe com.example.iris.myapplication.Trajet

Fonctions membres publiques

- [Trajet](#) ()
Constructeur de trajet.
- [Trajet](#) (int [chargeInitialeTTE](#), int [distanceParcoursueInitiale](#))
Constructeur de la trame.
- void [demarrerTrajet](#) ()
Demarre le [Trajet](#) en initialisant l'heure de debut.
- void [arreterTrajet](#) ()
modifie l'etat du trajet en false
- String [getVitesseMoyenneTTE](#) ()
Accesseur qui retourne la vitesse moyenne de la TTE durant le trajet.
- int [getConsommationTrajet](#) (int [chargeTTE](#))
Accesseur qui retourne la consommation moyenne de la TTE durant le trajet.
- Boolean [getEtatTrajet](#) ()
Accesseur qui retourne l'etat du trajet (en cours ou pas)
- int [getDistanceParcoursue](#) (int [distanceParcoursue](#))
Accesseur qui retourne la distance parcourus de la TTE durant le trajet.

Fonctions membres privées

- String [calculVitesseMoyenne](#) ()
calcul la vitesseMoyenne
- double [decoderTemp](#) (String tempsAConvertir)
Decode le temp pour le separer en heure, minute, seconde.
- double [conversionTemps](#) (String Heure, String minute, String seconde)
Convertie le temps.

Attributs privés

- Boolean [encours](#) = false
- String [vitesseMoyenneTTE](#)

- `int` `chargeInitialeTTE`
- `String` `tempsTrajet`
- `int` `distanceParcourueInitiale`
- `double` `distanceParcourueTrajet`
- `String` `heureDebut`

10.3.1 Description détaillée

Auteur

Hachette Alexandre

Version

0.8

10.3.2 Documentation des constructeurs et destructeur

10.3.2.1 `com.example.iris.myapplication.Trajet.Trajet ()`

Références `com.example.iris.myapplication.Trajet.chargeInitialeTTE`, `com.example.iris.myapplication.Trajet.distanceParcourueInitiale`, `com.example.iris.myapplication.Trajet.encours`, `com.example.iris.myapplication.Trajet.heureDebut`, et `com.example.iris.myapplication.Trajet.vitesseMoyenneTTE`.

```
{
    this.encours = false;
    this.vitesseMoyenneTTE = "";
    this.chargeInitialeTTE = 0;
    this.distanceParcourueInitiale = 0;
    this.heureDebut = "";
}
```

10.3.2.2 `com.example.iris.myapplication.Trajet.Trajet (int chargeInitialeTTE, int distanceParcourueInitiale)`

Paramètres

<i>charge-InitialeTTE</i>	
<i>distance-Parcourue-Initiale</i>	

Références `com.example.iris.myapplication.Trajet.chargeInitialeTTE`, `com.example.iris.myapplication.Trajet.distanceParcourueInitiale`, `com.example.iris.myapplication.Trajet.distanceParcourueTrajet`, `com.example.iris.myapplication.Trajet.encours`, `com.example.iris.myapplication.Trajet.heureDebut`, et `com.example.iris.myapplication.Trajet.vitesseMoyenneTTE`.

```
{
    this.encours = false;
    this.vitesseMoyenneTTE = "";
    this.chargeInitialeTTE = chargeInitialeTTE;
}
```

```

        this.distanceParcourueInitiale = distanceParcourueInitiale;
        this.distanceParcourueTrajet = 0;
        this.heureDebut = "";
    }

```

10.3.3 Documentation des fonctions membres

10.3.3.1 `void com.example.iris.myapplication.Trajet.arreterTrajet ()`

Références [com.example.iris.myapplication.Trajet.encours](#).

Référéncé par [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#).

```

{
    encours = false;
}

```

10.3.3.2 `String com.example.iris.myapplication.Trajet.calculVitesseMoyenne ()` [private]

Renvoie

String qui represente la vitesse moyenne de la TTE

Références [com.example.iris.myapplication.Trajet.decoderTemp\(\)](#), [com.example.iris.myapplication.Trajet.distanceParcourueTrajet](#), [com.example.iris.myapplication.Trajet.heureDebut](#), [com.example.iris.myapplication.Trajet.tempsTrajet](#), et [com.example.iris.myapplication.Trajet.vitesseMoyenneTTE](#).

Référéncé par [com.example.iris.myapplication.Trajet.getVitesseMoyenneTTE\(\)](#).

```

{
    Calendar calendar = Calendar.getInstance();
    SimpleDateFormat simpleDateFormat = new SimpleDateFormat("HH:mm:ss");
    String heureActuelle = simpleDateFormat.format(calendar.getTime());

    if(heureDebut.length() > 0)
    {
        double dureeTrajet = decoderTemp(heureActuelle) - decoderTemp(
heureDebut);
        tempsTrajet = String.format("%.2f",dureeTrajet);
        double vitesseMoyenne = (distanceParcourueTrajet/1000) /
dureeTrajet;
        vitesseMoyenneTTE = String.format("%.2f",vitesseMoyenne);
        return (vitesseMoyenneTTE);
    }

    return "0";
}

```

10.3.3.3 `double com.example.iris.myapplication.Trajet.conversionTemps (String` `Heure, String minute, String seconde)` [private]

Renvoie

double representant le temp en heure

Référéncé par [com.example.iris.myapplication.Trajet.decoderTemp\(\)](#).

```

{
    double temps = 0;
    Double heureConversion;
    Double minuteConversion;
    Double secondeConversion;

    heureConversion = Double.parseDouble(Heure);
    minuteConversion = Double.parseDouble(minute) / 60;
    secondeConversion = Double.parseDouble(seconde) / 3600;

    Log.d("<conversionTemps> H : ", heureConversion.toString());
    Log.d("<conversionTemps> Min :", minuteConversion.toString());

    temps = heureConversion + minuteConversion + secondeConversion;
    Log.d("<conversionTemps> dT:", new Double(temps).toString());
    return temps;
}

```

10.3.3.4 `double com.example.iris.myapplication.Trajet.decoderTemp (String tempsAConvertir)` [private]

Renvoie

double représentant le temp en heure

Références [com.example.iris.myapplication.Trajet.conversionTemps\(\)](#).

Référéncé par [com.example.iris.myapplication.Trajet.calculVitesseMoyenne\(\)](#).

```

{
    int position = 0;
    String heure = "";
    String minute = "";
    String seconde = "";

    for (int i=0; i < tempsAConvertir.length() ;i++)
    {
        if (tempsAConvertir.charAt(i) == ':')
        {
            position++;
        }
        else
        {
            if (position == 0)
            {
                heure = heure + tempsAConvertir.charAt(i);
            } else if (position == 1)
            {
                minute = minute + tempsAConvertir.charAt(i);
            } else
            {
                seconde = seconde + tempsAConvertir.charAt(i);
            }
        }
    }

    return (conversionTemps(heure,minute,seconde));
}

```

10.3.3.5 `void com.example.iris.myapplication.Trajet.demarrerTrajet ()`

Références [com.example.iris.myapplication.Trajet.encours](#), et [com.example.iris.myapplication.Trajet.heureDebut](#).

Référéncé par [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#).

```
{
    Calendar calendar = Calendar.getInstance();
    SimpleDateFormat simpleDateFormat = new SimpleDateFormat("HH:mm:ss");
    heureDebut = simpleDateFormat.format(calendar.getTime());
    encours = true;
}
```

10.3.3.6 `int com.example.iris.myapplication.Trajet.getConsommationTrajet (int chargeTTE)`

Renvoie

int qui represente la consommation moyenne(en %) de la TTE durant le trajet

Références [com.example.iris.myapplication.Trajet.chargeInitialeTTE](#).

Référencé par [com.example.iris.myapplication.MainActivity.afficherDonneesTTE\(\)](#).

```
{
    return chargeInitialeTTE - chargeTTE;
}
```

10.3.3.7 `int com.example.iris.myapplication.Trajet.getDistanceParcourue (int distanceParcourue)`

Renvoie

int qui represente la distance parcourus(en m) de la TTE durant le trajet

Références [com.example.iris.myapplication.Trajet.distanceParcourueInitiale](#), et [com.example.iris.myapplication.Trajet.distanceParcourueTrajet](#).

Référencé par [com.example.iris.myapplication.MainActivity.afficherDonneesTTE\(\)](#).

```
{
    distanceParcourueTrajet = distanceParcourue - distanceParcourueInitiale
;
    return (distanceParcourue - distanceParcourueInitiale);
}
```

10.3.3.8 `Boolean com.example.iris.myapplication.Trajet.getEtatTrajet ()`

Renvoie

boolean qui represente l'etat du trajet

Références [com.example.iris.myapplication.Trajet.encours](#).

Référencé par [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#).

```
{
    return encours;
}
```

10.3.3.9 `String com.example.iris.myapplication.Trajet.getVitesseMoyenneTTE ()`

Renvoie

int qui represente la vitesse moyenne(en km/h) de la TTE durant le trajet

Références [com.example.iris.myapplication.Trajet.calculVitesseMoyenne\(\)](#), et [com.example.iris.myapplication.Trajet.vitesseMoyenneTTE](#).

Référencé par [com.example.iris.myapplication.MainActivity.afficherDonneesTTE\(\)](#).

```
{  
    vitesseMoyenneTTE = calculVitesseMoyenne();  
    return vitesseMoyenneTTE;  
}
```

10.3.4 Documentation des données membres

10.3.4.1 `int com.example.iris.myapplication.Trajet.chargeInitialeTTE`

`[private]`

charge de la batterie en % de la trottinette

Référencé par [com.example.iris.myapplication.Trajet.getConsommationTrajet\(\)](#), et [com.example.iris.myapplication.Trajet.Trajet\(\)](#).

10.3.4.2 `int com.example.iris.myapplication.Trajet.distanceParcourueInitiale`

`[private]`

distance en km de la trottinette

Référencé par [com.example.iris.myapplication.Trajet.getDistanceParcourue\(\)](#), et [com.example.iris.myapplication.Trajet.Trajet\(\)](#).

10.3.4.3 `double com.example.iris.myapplication.Trajet.distanceParcourueTrajet`

`[private]`

distance parcourus durant le trajet en km

Référencé par [com.example.iris.myapplication.Trajet.calculVitesseMoyenne\(\)](#), [com.example.iris.myapplication.Trajet.getDistanceParcourue\(\)](#), et [com.example.iris.myapplication.Trajet.Trajet\(\)](#).

10.3.4.4 `Boolean com.example.iris.myapplication.Trajet.encours = false`

`[private]`

Référencé par [com.example.iris.myapplication.Trajet.arreterTrajet\(\)](#), [com.example.iris.myapplication.Trajet.demarrerTrajet\(\)](#), [com.example.iris.myapplication.Trajet.getEtatTrajet\(\)](#), et [com.example.iris.myapplication.Trajet.Trajet\(\)](#).

10.3.4.5 `String com.example.iris.myapplication.Trajet.heureDebut` `[private]`

Référencé par [com.example.iris.myapplication.Trajet.calculVitesseMoyenne\(\)](#), [com.example.iris.myapplication.Trajet.demarrerTrajet\(\)](#), et [com.example.iris.myapplication.Trajet.Trajet\(\)](#).

10.3.4.6 `String com.example.iris.myapplication.Trajet.tempsTrajet` [private]

duree du trajet en HH :mm :ss a affiche sur l'ihm (string)

Référencé par `com.example.iris.myapplication.Trajet.calculVitesseMoyenne()`.

10.3.4.7 `String com.example.iris.myapplication.Trajet.vitesseMoyenneTTE` [private]

vitesse en km/h de la trottinette

Référencé par `com.example.iris.myapplication.Trajet.calculVitesseMoyenne()`, `com.example.iris.myapplication.Trajet.getVitesseMoyenneTTE()`, et `com.example.iris.myapplication.Trajet.Trajet()`.

La documentation de cette classe a été générée à partir du fichier suivant :

- [Trajet.java](#)

10.4 Référence de la classe `com.example.iris.myapplication.TrameTTE`

Class `TrameTTE` permet de verifier si la trame recue est valide, et la decode dans ce cas la.

Fonctions membres publiques

- `TrameTTE ()`
Constructeur de la trame.
- Boolean `TraiterTrame` (String trame) throws InterruptedException
Fonction de traitement de la trame.
- boolean `DecoderTrame` (String trame)
Fonction permettant de decoder la trame pour l'affichage.
- int `getVitesse ()`
Accesseur qui retourne la vitesse de la TTE.
- int `getChargeBatterie ()`
Accesseur qui nous permet d'obtenir la charge de la batterie de notre TTE.
- int `getDistanceParcourue ()`
Accesseur qui nous permet d'obtenir la distance parcourue par la TTE.
- int `getIdTEC ()`
Accesseur qui retourne l'id de la TTE.

Fonctions membres privées

- Boolean `VerifierTrame` (String trame) throws InterruptedException
Fonction verifiant la Trame.

Attributs privés

- int `IdTEC`
- int `Vitesse`
- int `ChargeBatterie`
- int `DistanceParcourue`

Attributs privés statiques

- static final String `DEBUT_TRAME` = "\$"
- static final String `TYPE_PROTOCOLE` = "TEC"
- static final String `FIN_TRAME` = "*"

10.4.1 Description détaillée

Auteur

HACHETTE Alexandre

10.4.2 Documentation des constructeurs et destructeur

10.4.2.1 `com.example.iris.myapplication.TrameTTE.TrameTTE ()`

Références [com.example.iris.myapplication.TrameTTE.IdTEC](#).

```
{  
    IdTEC = 1;  
}
```

10.4.3 Documentation des fonctions membres

10.4.3.1 `boolean com.example.iris.myapplication.TrameTTE.DecoderTrame (String trame)`

Paramètres

<i>trame</i>	
--------------	--

Renvoie

bool si la trame a ete bien decoder ou non

Références [com.example.iris.myapplication.TrameTTE.ChargeBatterie](#), [com.example.iris.myapplication.TrameTTE.DistanceParcourue](#), [com.example.iris.myapplication.TrameTTE.IdTEC](#), et [com.example.iris.myapplication.TrameTTE.Vitesse](#).

Référencé par [com.example.iris.myapplication.TrameTTE.TraiterTrame\(\)](#).

```
{  
    /*  
    Exemple : (voir dossier)  
  
    $TEC1,25,70,1525*  
  
    - Vitesse : 25 -> 25 km/h  
    - Charge : 70 -> 70 %  
    - Distance parcourue : 1525 -> 1525 m  
  
    */  
    // enlève le délimiteur de fin '*'  
    trame = trame.substring(0, trame.length()-1);  
    List<String> champs = new ArrayList<String>(Arrays.asList(trame.split("  
    ,"))));
```

```
if(champs.size() == 4)
{
    int position = 0;
    Vitesse = Integer.parseInt(champs.get(++position));
    ChargeBatterie = Integer.parseInt(champs.get(++position));
    DistanceParcourue = Integer.parseInt(champs.get(++position));
    String id = trame.substring(champs.get(0).indexOf("C") + 1, champs.
get(0).length());
    IdTEC = Integer.parseInt(id);

    return true;
}
return false;
}
```

10.4.3.2 `int com.example.iris.myapplication.TrameTTE.getChargeBatterie ()`

Renvoie

int qui represente la charge de la batterie

Références [com.example.iris.myapplication.TrameTTE.ChargeBatterie](#).

Référencé par [com.example.iris.myapplication.MainActivity.afficherDonneesTTE\(\)](#), et [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#).

```
{
    return ChargeBatterie;
}
```

10.4.3.3 `int com.example.iris.myapplication.TrameTTE.getDistanceParcourue ()`

Renvoie

int qui represente la distance parcourue

Références [com.example.iris.myapplication.TrameTTE.DistanceParcourue](#).

Référencé par [com.example.iris.myapplication.MainActivity.afficherDonneesTTE\(\)](#), et [com.example.iris.myapplication.MainActivity.gererBoutons\(\)](#).

```
{
    return DistanceParcourue;
}
```

10.4.3.4 `int com.example.iris.myapplication.TrameTTE.getIdTEC ()`

Renvoie

int qui represente l'Id de la TTE

Références [com.example.iris.myapplication.TrameTTE.IdTEC](#).

```
{
    return IdTEC;
}
```

10.4.3.5 `int com.example.iris.myapplication.TrameTTE.getVitesse ()`

Renvoie

`int` qui represente la vitesse de la trottinette

Références [com.example.iris.myapplication.TrameTTE.Vitesse](#).

Référéncé par [com.example.iris.myapplication.MainActivity.afficherDonneesTTE\(\)](#).

```
{  
    return Vitesse;  
}
```

10.4.3.6 `Boolean com.example.iris.myapplication.TrameTTE.TraiterTrame (String trame) throws InterruptedException`

Paramètres

<i>trame</i>	
--------------	--

Renvoie

`bool` indique si la trame est bonne est a ete decoder ou non

Exceptions

<i>InterruptedException</i>	
-----------------------------	--

Références [com.example.iris.myapplication.TrameTTE.DecoderTrame\(\)](#), et [com.example.iris.myapplication.TrameTTE.VerifierTrame\(\)](#).

```
{  
    if(VerifierTrame(trame))  
    {  
        if(DecoderTrame(trame))  
            return true;  
    }  
    return false;  
}
```

10.4.3.7 `Boolean com.example.iris.myapplication.TrameTTE.VerifierTrame (String trame) throws InterruptedException [private]`

Paramètres

<i>trame</i>	
--------------	--

Renvoie

`bool` indiquant si la trame est valide ou non

Exceptions

<i>InterruptedException</i>	
-----------------------------	--

Références [com.example.iris.myapplication.TrameTTE.DEBUT_TRADE](#), [com.example.iris.myapplication.TrameTTE.FIN_TRADE](#), et [com.example.iris.myapplication.TrameTTE.TYPE_PROTOCOLE](#).

Référencé par [com.example.iris.myapplication.TrameTTE.TraiterTrame\(\)](#).

```
{
    /*
    Exemple : (voir dossier)

    $TEC1,25,70,1525*

    - Vitesse : 25 -> 25 km/h
    - Charge : 70 -> 70 %
    - Distance parcourue : 1525 -> 1525 m

    */
    if(trame.length() != 0)
    {
        // est-ce le début valide d'une trame TEC ?
        if(trame.startsWith(DEBUT_TRADE))
        {
            // est-ce le bon protocole ?
            if(trame.startsWith(DEBUT_TRADE + TYPE_PROTOCOLE))
            {
                // est-ce la fin valide d'une trame TEC ?
                if(trame.contains(FIN_TRADE))
                {
                    return true;
                }
            }
            else
            {
                Log.e("<TrameTTE>", "<FiltrerTrame> Trame non valide pas de caractere de Fin");
            }
        }
        else
        {
            Log.e("<TrameTTE>", "<FiltrerTrame> Trame non valide mauvais type de trame");
        }
    }
    else
    {
        Log.e("<TrameTTE>", "<FiltrerTrame> Trame non valide caractere de debut different");
    }
}
else
{
    Log.e("<TrameTTE>", "<FiltrerTrame> Trame vide");
}
return false;
}
```

10.4.4 Documentation des données membres

10.4.4.1 `int com.example.iris.myapplication.TrameTTE.ChargeBatterie` [private]

Référencé par [com.example.iris.myapplication.TrameTTE.DecoderTrame\(\)](#), et [com.example.iris.myapplication.TrameTTE.getChargeBatterie\(\)](#).

10.4.4.2 `final String com.example.iris.myapplication.TrameTTE.DEBUT_TRADE = "$"` [static, private]

Référencé par [com.example.iris.myapplication.TrameTTE.VerifierTrame\(\)](#).

10.5 Référence de la classe

com.example.iris.myapplication.PeripheriqueBluetooth.TReception

47

10.4.4.3 `int com.example.iris.myapplication.TrameTTE.DistanceParcourue`
[private]

Référencé par [com.example.iris.myapplication.TrameTTE.DecoderTrame\(\)](#), et [com.example.iris.myapplication.TrameTTE.getDistanceParcourue\(\)](#).

10.4.4.4 `final String com.example.iris.myapplication.TrameTTE.FIN_TRAME = "*"`
[static, private]

Référencé par [com.example.iris.myapplication.TrameTTE.VerifierTrame\(\)](#).

10.4.4.5 `int com.example.iris.myapplication.TrameTTE.IdTEC` [private]

Référencé par [com.example.iris.myapplication.TrameTTE.DecoderTrame\(\)](#), [com.example.iris.myapplication.TrameTTE.getIdTEC\(\)](#), et [com.example.iris.myapplication.TrameTTE.TrameTTE\(\)](#).

10.4.4.6 `final String com.example.iris.myapplication.TrameTTE.TYPE_PROTOCOLE = "TEC"` [static, private]

Référencé par [com.example.iris.myapplication.TrameTTE.VerifierTrame\(\)](#).

10.4.4.7 `int com.example.iris.myapplication.TrameTTE.Vitesse` [private]

Référencé par [com.example.iris.myapplication.TrameTTE.DecoderTrame\(\)](#), et [com.example.iris.myapplication.TrameTTE.getVitesse\(\)](#).

La documentation de cette classe a été générée à partir du fichier suivant :

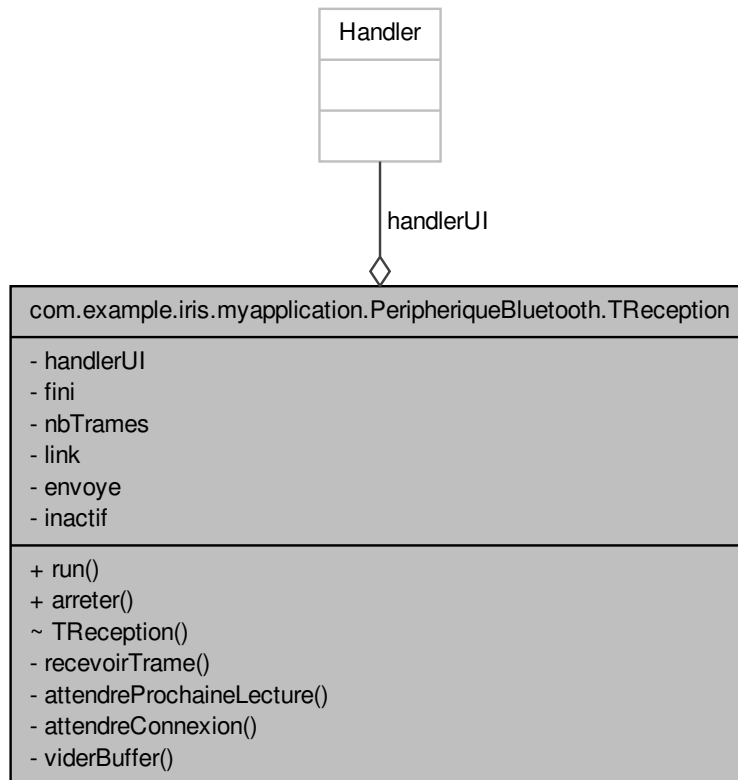
– [TrameTTE.java](#)

10.5 Référence de la classe com.example.iris.myapplication.PeripheriqueBluetooth.TReception

Thread de réception des trames en provenance de la Trottinette via le Bluetooth.

Graphe de collaboration de [com.example.iris.myapplication.PeripheriqueBluetooth.T-](#)

Reception :



Fonctions membres publiques

- void **run** ()
Réceptionne les trames en provenance de l'aquarium via le Bluetooth.
- void **arreter** ()
Permet d'arrêter le thread de réception des trames.

Fonctions de paquetage

- **TReception** (Handler h)
Constructeur de la classe **TReception**.

Fonctions membres privées

- void **recevoirTrame** ()

10.5 Référence de la classe

com.example.iris.myapplication.PeripheriqueBluetooth.TReception

49

- Permet de recevoir les trames Bluetooth.*
 - void [attendreProchaineLecture](#) (int millis)
- Permet d'attendre la prochaine lecture de trames Bluetooth.*
 - void [attendreConnexion](#) ()
- Attend la prochaine connexion Bluetooth.*
 - void [viderBuffer](#) ()
- Permet de vider le buffer.*

Attributs privés

- Handler [handlerUI](#)
- boolean [fini](#)
- long [nbTrames](#) = 0
- long [link](#) = 0
- boolean [envoye](#) = false
- boolean [inactif](#) = false

10.5.1 Description détaillée

Auteur

Thierry Vaira <tvaira@free.fr>
HACHETTE Alexandre

10.5.2 Documentation des constructeurs et destructeur

10.5.2.1 **com.example.iris.myapplication.PeripheriqueBluetooth.TReception.T-Reception (Handler *h*)** [package]

Paramètres

<i>h</i>	
----------	--

Références [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.fini](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.handlerUI](#).

```
{  
    handlerUI = h;  
    fini = false;  
}
```

10.5.3 Documentation des fonctions membres

10.5.3.1 **void com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-arreter ()**

Auteur

Thierry Vaira <tvaira@free.fr>

Références [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.fini](#).

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.deconnecter\(\)](#).

```

{
    if(fini == false)
    {
        fini = true;
    }
    try
    {
        Thread.sleep(250);
    }
    catch (InterruptedException e)
    {
        e.printStackTrace();
    }
}

```

10.5.3.2 void com.example.iris.myapplication.PeripheriqueBluetooth.TReception.attendreConnexion () [private]

Références [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.fini](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.socket](#).

```

{
    /* attente connexion */
    if(!fini)
    {
        while (!socket.isConnected())
        {
            try
            {
                Log.d("Réception run()", "Attente connexion ..."); // d
= debug
                Thread.sleep(150);
            } catch (InterruptedException e)
            {
                e.printStackTrace();
            }
        }
    }
}

```

10.5.3.3 void com.example.iris.myapplication.PeripheriqueBluetooth.TReception.attendreProchaineLecture (int millis) [private]

Paramètres

<i>millis</i>	un int qui représente une durée en millisecondes (ms)
---------------	---

Références [com.example.iris.myapplication.PeripheriqueBluetooth.CODE_ACTIF](#), [com.example.iris.myapplication.PeripheriqueBluetooth.CODE_INACTIF](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.envoye](#), [com.example.iris.myapplication.PeripheriqueBluetooth.gestionnaire](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.inactif](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.link](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.nbTrames](#), [com.example.iris.myapplication.PeripheriqueBluetooth.PERIODE_INACTIVE](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.PROCHAINE_LECTURE](#).

Référéncé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-recevoirTrame\(\)](#).

```

{
    try
    {
        /* attente prochaine lecture */
        Thread.sleep(millis);
        if(envoye)
        {
            envoye = false;
            link = 0;
            if(inactif)
            {
                /* informe l'activité principale */
                Message msg = Message.obtain();
                msg.what = PeripheriqueBluetooth.CODE_ACTIF;
                if (gestionnaire.sendMessage(msg))
                {
                    Log.d("run()", "Socket active : " + nbTrames); // d
                }
            }
            else
            {
                Log.e("run()", "Message non envoyé !");
            }
            inactif = false;
        }
    }
    else
    {
        link += PeripheriqueBluetooth.PROCHAINE_LECTURE;

        /* perte du lien */
        if(link >= PeripheriqueBluetooth.PERIODE_INACTIVE)
        {
            /* informe l'activité principale */
            Message msg = Message.obtain();
            msg.what = PeripheriqueBluetooth.CODE_INACTIF;
            if (gestionnaire.sendMessage(msg))
            {
                Log.d("run()", "Socket inactive : " + nbTrames); //
            }
            else
            {
                Log.e("run()", "Message non envoyé !");
            }
            inactif = true;
            link = 0;
        }
    }
} catch (InterruptedException e)
{
    e.printStackTrace();
}
}

```

10.5.3.4 void com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-recevoirTrame () [private]

Références [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.CODE_RECEPTION](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.fini](#), [com.example.iris.myapplication.PeripheriqueBluetooth.fluxEntree](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getAdresse\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getNom\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.handlerUI](#), [com.example.iris.myapplication.PeripheriqueBluetooth.PROCHAINE_LECTURE](#), [com.example.iris.myapplication.-](#)

[PeripheriqueBluetooth.socket](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TAILLE_BUFFER](#).

```

    {
        try
        {
            if(fluxEntree.available() > 0)
            {
                byte buffer[] = new byte[PeripheriqueBluetooth.
TAILLE_BUFFER];
                if(!fini && socket.isConnected())
                {
                    int k = fluxEntree.read(buffer, 0, PeripheriqueBluetooth
.TAILLE_BUFFER);

                    if (k > 0)
                    {
                        byte rawdata[] = new byte[k];
                        for (int i = 0; i < k; i++)
                            rawdata[i] = buffer[i];

                        String datas = new String(rawdata);
                        System.out.println("<Bluetooth> Reception " + datas

);

                        StringBuffer d = new StringBuffer(datas);
                        Message message = Message.obtain();
                        Bundle paquet = new Bundle();
                        paquet.putString("nom", getNom());
                        paquet.putString("adresse", getAdresse());
                        paquet.putInt("etat", CODE_RECEPTION);
                        paquet.putString("donnees", datas);
                        message.setData(paquet);
                        handlerUI.sendMessage(message);

                    }
                }
            }
            attendreProchaineLecture(PeripheriqueBluetooth.
PROCHAINE_LECTURE);
        }
        catch (IOException e)
        {
            System.out.println("<Socket> Erreur read");
            e.printStackTrace();
        }
    }
}

```

10.5.3.5 void com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-run ()

Auteur

Thierry Vaira <tvaira@free.fr>

Références [com.example.iris.myapplication.PeripheriqueBluetooth.CODE_RECEPTION](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.fini](#), [com.example.iris.myapplication.PeripheriqueBluetooth.fluxEntree](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getAdresse\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.getNom\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.handlerUI](#).

```

{
    System.out.println("<Bluetooth> Attente reception");
    while(!fini)
    {

```

```

try
{
    if(fluxEntree.available() > 0)
    {
        byte buffer[] = new byte[100];
        int k = fluxEntree.read(buffer, 0, 100);

        if(k > 0)
        {
            byte rawdata[] = new byte[k];
            for(int i=0;i<k;i++)
                rawdata[i] = buffer[i];

            String data = new String(rawdata);
            System.out.println("<Bluetooth> Reception " + data)
;

            Message msg = Message.obtain();
            Bundle b = new Bundle();
            b.putString("nom", getNom());
            b.putString("adresse", getAdresse());
            b.putInt("etat", CODE_RECEPTION);
            b.putString("donnees", data);
            msg.setData(b);
            handlerUI.sendMessage(msg);
        }
    }
    try
    {
        Thread.sleep(250);
    }
    catch (InterruptedException e)
    {
        e.printStackTrace();
    }
}
catch (IOException e)
{
    e.printStackTrace();
}
}
}

```

10.5.3.6 void com.example.iris.myapplication.PeripheriqueBluetooth.TReception.viderBuffer () [private]

Références [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.fini](#), [com.example.iris.myapplication.PeripheriqueBluetooth.fluxEntree](#), [com.example.iris.myapplication.PeripheriqueBluetooth.socket](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TAILLE_BUFFER](#).

```

{
    try
    {
        Thread.sleep(1000);
        /* doit-on vider le buffer ? */
        if(!fini && socket.isConnected())
        {
            while (fluxEntree.available() > PeripheriqueBluetooth.
TAILLE_BUFFER)
            {
                Log.d("Réception run()", "Vide le buffer : " +
fluxEntree.skip(fluxEntree.available()) + " octets"); // d = debug
            }
        }
    }
    catch (IOException e)
    {
        e.printStackTrace();
    }
}

```

10.5 Référence de la classe

com.example.iris.myapplication.PeripheriqueBluetooth.TReception

54

```
    }  
    catch (InterruptedException e)  
    {  
        e.printStackTrace();  
    }  
}
```

10.5.4 Documentation des données membres

10.5.4.1 boolean com.example.iris.myapplication.PeripheriqueBluetooth.TReception.envoye = false [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#).

10.5.4.2 boolean com.example.iris.myapplication.PeripheriqueBluetooth.TReception.fini [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-arreter\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.attendreConnexion\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-recevoirTrame\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-run\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.TReception\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.viderBuffer\(\)](#).

10.5.4.3 Handler com.example.iris.myapplication.PeripheriqueBluetooth.TReception.handlerUI [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-recevoirTrame\(\)](#), [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-run\(\)](#), et [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.TReception\(\)](#).

10.5.4.4 boolean com.example.iris.myapplication.PeripheriqueBluetooth.TReception.inactif = false [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#).

10.5.4.5 long com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-link = 0 [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#).

10.5.4.6 long com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-nbTrames = 0 [private]

Référencé par [com.example.iris.myapplication.PeripheriqueBluetooth.TReception.-attendreProchaineLecture\(\)](#).

La documentation de cette classe a été générée à partir du fichier suivant :

– [PeripheriqueBluetooth.java](#)

11 Documentation des fichiers

11.1 Référence du fichier AndroidManifest.xml

11.2 Référence du fichier Changelog.dox

11.3 Référence du fichier MainActivity.java

Classes

- class [com.example.iris.myapplication.MainActivity](#)
Activité principale de l'application (Thread UI)

11.4 Référence du fichier PeripheriqueBluetooth.java

Classes

- class [com.example.iris.myapplication.PeripheriqueBluetooth](#)
Assure le dialogue bluetooth entre la trottinette et le telephone.
- class [com.example.iris.myapplication.PeripheriqueBluetooth.TReception](#)
Thread de réception des trames en provenance de la Trottinette via le Bluetooth.

11.5 Référence du fichier README.dox

11.6 Référence du fichier Trajet.java

Classes

- class [com.example.iris.myapplication.Trajet](#)

11.7 Référence du fichier TrameTTE.java

Classes

- class [com.example.iris.myapplication.TrameTTE](#)
Class [TrameTTE](#) permet de verifier si la trame recue est valide, et la decode dans ce cas la.

Index

AndroidManifest.xml, 54
CODE_ACTIF
 com : :example : :iris : :myapplication-
 : :PeripheriqueBluetooth, 33
CODE_CONNEXION
 com : :example : :iris : :myapplication-
 : :PeripheriqueBluetooth, 33
CODE_INACTIF
 com : :example : :iris : :myapplication-
 : :PeripheriqueBluetooth, 33
CODE_RECEPTION
 com : :example : :iris : :myapplication-
 : :PeripheriqueBluetooth, 33
Changelog.dox, 54
ChargeBatterie
 com : :example : :iris : :myapplication-
 : :TrameTTE, 45
DEBUT_TRAME
 com : :example : :iris : :myapplication-
 : :TrameTTE, 45
DecoderTrame
 com : :example : :iris : :myapplication-
 : :TrameTTE, 42
DistanceParcourue
 com : :example : :iris : :myapplication-
 : :TrameTTE, 45
FIN_TRAME
 com : :example : :iris : :myapplication-
 : :TrameTTE, 46
IdTEC
 com : :example : :iris : :myapplication-
 : :TrameTTE, 46
MainActivity.java, 54
PeripheriqueBluetooth
 com : :example : :iris : :myapplication-
 : :PeripheriqueBluetooth, 27
PeripheriqueBluetooth.java, 54
README.dox, 54
TAG
 com : :example : :iris : :myapplication-
 : :MainActivity, 25
TAILLE_BUFFER
 com : :example : :iris : :myapplication-
 : :PeripheriqueBluetooth, 35
TReception
 com : :example : :iris : :myapplication-
 : :PeripheriqueBluetooth : :T-
 Reception, 48
TYPE_PROTOCOLE
 com : :example : :iris : :myapplication-
 : :TrameTTE, 46
TraiterTrame
 com : :example : :iris : :myapplication-
 : :TrameTTE, 44
Trajet
 com : :example : :iris : :myapplication-
 : :Trajet, 36
Trajet.java, 54
TrameTTE
 com : :example : :iris : :myapplication-
 : :TrameTTE, 42
TrameTTE.java, 54
VerifierTrame
 com : :example : :iris : :myapplication-
 : :TrameTTE, 44
Vitesse
 com : :example : :iris : :myapplication-
 : :TrameTTE, 46
activerBluetooth
 com : :example : :iris : :myapplication-
 : :MainActivity, 9
adresse
 com : :example : :iris : :myapplication-
 : :PeripheriqueBluetooth, 33
affichageChargeBatterie
 com : :example : :iris : :myapplication-
 : :MainActivity, 21
affichageConsommationTrajet
 com : :example : :iris : :myapplication-
 : :MainActivity, 21
affichageDistanceParcourue
 com : :example : :iris : :myapplication-
 : :MainActivity, 21
affichageDistanceParcourueTrajet
 com : :example : :iris : :myapplication-
 : :MainActivity, 21
affichageVitesse
 com : :example : :iris : :myapplication-
 : :MainActivity, 21
affichageVitesseMoyenneTrajet
 com : :example : :iris : :myapplication-
 : :MainActivity, 21

- afficherDonneesTTE
 - com : :example : :iris : :myapplication- : :MainActivity, [10](#)
- afficherPositionTTE
 - com : :example : :iris : :myapplication- : :MainActivity, [12](#)
- arreter
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth : :T-Reception, [48](#)
- arreterBluetooth
 - com : :example : :iris : :myapplication- : :MainActivity, [12](#)
- arreterLocalisation
 - com : :example : :iris : :myapplication- : :MainActivity, [13](#)
- arreterTrajet
 - com : :example : :iris : :myapplication- : :Trajet, [37](#)
- attendreConnexion
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth : :T-Reception, [49](#)
- attendreProchaineLecture
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth : :T-Reception, [49](#)
- bluetoothAdapter
 - com : :example : :iris : :myapplication- : :MainActivity, [21](#)
- bluetoothReceiver
 - com : :example : :iris : :myapplication- : :MainActivity, [22](#)
- boutonCommunicationTTE
 - com : :example : :iris : :myapplication- : :MainActivity, [22](#)
- boutonTrajet
 - com : :example : :iris : :myapplication- : :MainActivity, [22](#)
- calculVitesseMoyenne
 - com : :example : :iris : :myapplication- : :Trajet, [37](#)
- chargeInitialeTTE
 - com : :example : :iris : :myapplication- : :Trajet, [40](#)
- com.example.iris.myapplication.MainActivity, [5](#)
- com.example.iris.myapplication.PeripheriqueBluetooth, [26](#)
- com.example.iris.myapplication.PeripheriqueBluetooth.TReception, [46](#)
- com.example.iris.myapplication.Trajet, [35](#)
- com.example.iris.myapplication.TrameTTE, [41](#)
- com : :example : :iris : :myapplication : :-MainActivity
 - TAG, [25](#)
 - activerBluetooth, [9](#)
 - affichageChargeBatterie, [21](#)
 - affichageConsommationTrajet, [21](#)
 - affichageDistanceParcourue, [21](#)
 - affichageDistanceParcourueTrajet, [21](#)
 - affichageVitesse, [21](#)
 - affichageVitesseMoyenneTrajet, [21](#)
 - afficherDonneesTTE, [10](#)
 - afficherPositionTTE, [12](#)
 - arreterBluetooth, [12](#)
 - arreterLocalisation, [13](#)
 - bluetoothAdapter, [21](#)
 - bluetoothReceiver, [22](#)
 - boutonCommunicationTTE, [22](#)
 - boutonTrajet, [22](#)
 - devices, [22](#)
 - ecouteurGPS, [22](#)
 - etatLocalisation, [22](#)
 - fournisseur, [22](#)
 - geocoder, [23](#)
 - gererBoutons, [13](#)
 - gestionnaire, [23](#)
 - getApplicationName, [14](#)
 - getPosition, [14](#)
 - initialiserActivite, [15](#)
 - initialiserCarte, [15](#)
 - initialiserCommunication, [16](#)
 - initialiserLocalisation, [16](#)
 - journal, [23](#)
 - journaliser, [17](#)
 - locationManager, [23](#)
 - mAzimuthAngleSpeed, [23](#)
 - mCompassOverlay, [23](#)
 - mListLog, [23](#)
 - mListView, [23](#)
 - mRotationGestureOverlay, [23](#)
 - mSpeed, [24](#)
 - myLocationOverlay, [24](#)
 - myOpenMapView, [24](#)

- myScaleBarOverlay, 24
- noms, 24
- onActivityResult, 18
- onClick, 18
- onCreate, 18
- onCreateOptionsMenu, 19
- onDestroy, 19
- onOptionsItemSelected, 19
- onResume, 19
- peripherique, 24
- peripheriques, 24
- remplacerPeripherique, 20
- setBoutons, 20
- spinnerListePeripheriques, 25
- traceTrajet, 25
- trajet, 25
- trajetEncours, 25
- trame, 25
- com : :example : :iris : :myapplication : :-
 - PeripheriqueBluetooth
 - CODE_ACTIF, 33
 - CODE_INACTIF, 33
 - PeripheriqueBluetooth, 27
 - adresse, 33
 - connecter, 28
 - deconnecter, 29
 - envoyer, 30
 - estConnecte, 31
 - fluxEntree, 33
 - fluxEnvoie, 33
 - gestionnaire, 34
 - getAdresse, 31
 - getDevice, 31
 - getNom, 32
 - nom, 34
 - peripherique, 34
 - setNom, 32
 - socket, 34
 - tReception, 35
 - toString, 32
- com : :example : :iris : :myapplication-
 - : :PeripheriqueBluetooth : :T-
 - Reception
 - TReception, 48
 - arreter, 48
 - attendreConnexion, 49
 - attendreProchaineLecture, 49
 - envoye, 53
 - fini, 53
 - handlerUI, 53
 - inactif, 53
 - link, 53
 - nbTrames, 53
 - recevoirTrame, 50
 - run, 51
 - viderBuffer, 52
- com : :example : :iris : :myapplication : :-
 - Trajet
 - Trajet, 36
 - arreterTrajet, 37
 - calculVitesseMoyenne, 37
 - chargeInitialeTTE, 40
 - conversionTemps, 37
 - decoderTemp, 38
 - demarrerTrajet, 38
 - distanceParcourueInitiale, 40
 - distanceParcourueTrajet, 40
 - encours, 40
 - getConsommationTrajet, 39
 - getDistanceParcourue, 39
 - getEtatTrajet, 39
 - getVitesseMoyenneTTE, 39
 - heureDebut, 40
 - tempsTrajet, 40
 - vitesseMoyenneTTE, 41
- com : :example : :iris : :myapplication : :-
 - TrameTTE
 - ChargeBatterie, 45
 - DEBUT_TRAME, 45
 - DecoderTrame, 42
 - DistanceParcourue, 45
 - FIN_TRAME, 46
 - IdTEC, 46
 - TraiterTrame, 44
 - TrameTTE, 42
 - VerifierTrame, 44
 - Vitesse, 46
 - getChargeBatterie, 43
 - getDistanceParcourue, 43
 - getIdTEC, 43
 - getVitesse, 43
- connecter
 - com : :example : :iris : :myapplication-
 - : :PeripheriqueBluetooth, 28
- conversionTemps
 - com : :example : :iris : :myapplication-
 - : :Trajet, 37
- decoderTemp

- com : :example : :iris : :myapplication-
: :Trajet, [38](#)
- deconnecter
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [29](#)
- demarrerTrajet
 - com : :example : :iris : :myapplication-
: :Trajet, [38](#)
- devices
 - com : :example : :iris : :myapplication-
: :MainActivity, [22](#)
- distanceParcourueInitiale
 - com : :example : :iris : :myapplication-
: :Trajet, [40](#)
- distanceParcourueTrajet
 - com : :example : :iris : :myapplication-
: :Trajet, [40](#)
- ecouteurGPS
 - com : :example : :iris : :myapplication-
: :MainActivity, [22](#)
- encours
 - com : :example : :iris : :myapplication-
: :Trajet, [40](#)
- envoye
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth : :T-
Reception, [53](#)
- envoyer
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [30](#)
- estConnecte
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [31](#)
- etatLocalisation
 - com : :example : :iris : :myapplication-
: :MainActivity, [22](#)
- fini
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth : :T-
Reception, [53](#)
- fluxEntree
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [33](#)
- fluxEnvoie
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [33](#)
- fournisseur
 - com : :example : :iris : :myapplication-
: :MainActivity, [22](#)
- geocoder
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
- gererBoutons
 - com : :example : :iris : :myapplication-
: :MainActivity, [13](#)
- gestionnaire
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [34](#)
- getAdresse
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [31](#)
- getApplicationName
 - com : :example : :iris : :myapplication-
: :MainActivity, [14](#)
- getChargeBatterie
 - com : :example : :iris : :myapplication-
: :TrameTTE, [43](#)
- getConsommationTrajet
 - com : :example : :iris : :myapplication-
: :Trajet, [39](#)
- getDevice
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [31](#)
- getDistanceParcourue
 - com : :example : :iris : :myapplication-
: :Trajet, [39](#)
 - com : :example : :iris : :myapplication-
: :TrameTTE, [43](#)
- getEtatTrajet
 - com : :example : :iris : :myapplication-
: :Trajet, [39](#)
- getIdTEC
 - com : :example : :iris : :myapplication-
: :TrameTTE, [43](#)
- getNom
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [32](#)
- getPosition
 - com : :example : :iris : :myapplication-
: :MainActivity, [14](#)
- getVitesse
 - com : :example : :iris : :myapplication-
: :TrameTTE, [43](#)
- getVitesseMoyenneTTE

- com : :example : :iris : :myapplication-
: :Trajet, [39](#)
- handlerUI
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth : :T-
Reception, [53](#)
- heureDebut
 - com : :example : :iris : :myapplication-
: :Trajet, [40](#)
- inactif
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth : :T-
Reception, [53](#)
- initialiserActivite
 - com : :example : :iris : :myapplication-
: :MainActivity, [15](#)
- initialiserCarte
 - com : :example : :iris : :myapplication-
: :MainActivity, [15](#)
- initialiserCommunication
 - com : :example : :iris : :myapplication-
: :MainActivity, [16](#)
- initialiserLocalisation
 - com : :example : :iris : :myapplication-
: :MainActivity, [16](#)
- journal
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
- journaliser
 - com : :example : :iris : :myapplication-
: :MainActivity, [17](#)
- link
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth : :T-
Reception, [53](#)
- locationManager
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
- mAzimuthAngleSpeed
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
- mCompassOverlay
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
- mListLog
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
- mListView
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
- mRotationGestureOverlay
 - com : :example : :iris : :myapplication-
: :MainActivity, [23](#)
- mSpeed
 - com : :example : :iris : :myapplication-
: :MainActivity, [24](#)
- myLocationOverlay
 - com : :example : :iris : :myapplication-
: :MainActivity, [24](#)
- myOpenMapView
 - com : :example : :iris : :myapplication-
: :MainActivity, [24](#)
- myScaleBarOverlay
 - com : :example : :iris : :myapplication-
: :MainActivity, [24](#)
- nbTrames
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth : :T-
Reception, [53](#)
- nom
 - com : :example : :iris : :myapplication-
: :PeripheriqueBluetooth, [34](#)
- noms
 - com : :example : :iris : :myapplication-
: :MainActivity, [24](#)
- onActivityResult
 - com : :example : :iris : :myapplication-
: :MainActivity, [18](#)
- onClick
 - com : :example : :iris : :myapplication-
: :MainActivity, [18](#)
- onCreate
 - com : :example : :iris : :myapplication-
: :MainActivity, [18](#)
- onCreateOptionsMenu
 - com : :example : :iris : :myapplication-
: :MainActivity, [19](#)
- onDestroy
 - com : :example : :iris : :myapplication-
: :MainActivity, [19](#)
- onOptionsItemSelected
 - com : :example : :iris : :myapplication-
: :MainActivity, [19](#)

- onResume
 - com : :example : :iris : :myapplication- : :MainActivity, [19](#)
- peripherique
 - com : :example : :iris : :myapplication- : :MainActivity, [24](#)
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth, [34](#)
- peripheriques
 - com : :example : :iris : :myapplication- : :MainActivity, [24](#)
- recevoirTrame
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth : :T-Reception, [50](#)
- remplacerPeripherique
 - com : :example : :iris : :myapplication- : :MainActivity, [20](#)
- run
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth : :T-Reception, [51](#)
- setBoutons
 - com : :example : :iris : :myapplication- : :MainActivity, [20](#)
- setNom
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth, [32](#)
- socket
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth, [34](#)
- spinnerListePeripheriques
 - com : :example : :iris : :myapplication- : :MainActivity, [25](#)
- tReception
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth, [35](#)
- tempsTrajet
 - com : :example : :iris : :myapplication- : :Trajet, [40](#)
- toString
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth, [32](#)
- traceTrajet
 - com : :example : :iris : :myapplication- : :MainActivity, [25](#)
- trajet
 - com : :example : :iris : :myapplication- : :MainActivity, [25](#)
- trajetEncours
 - com : :example : :iris : :myapplication- : :MainActivity, [25](#)
- trame
 - com : :example : :iris : :myapplication- : :MainActivity, [25](#)
- viderBuffer
 - com : :example : :iris : :myapplication- : :PeripheriqueBluetooth : :T-Reception, [52](#)
- vitesseMoyenneTTE
 - com : :example : :iris : :myapplication- : :Trajet, [41](#)