

Mobile-DARTS

version 1.1

BTS SNIR LaSalle Avignon 2021

Table des matières

1	Application Android	2
1.1	Table des matières	2
1.2	Informations	2
2	Module de gestion de partie	2
3	Protocole DARTS	3
4	README	4
4.1	Mobile-DARTS	4
4.1.1	Présentation	4
4.1.2	Informations	5
5	A propos	5
6	Licence GPL	5
7	Liste des choses à faire	5
8	Documentation des espaces de nommage	6
8.1	Paquetage com	6
8.2	Paquetage com.lasalle84	6
8.3	Paquetage com.lasalle84.mobile_darts	6
9	Documentation des classes	6
9.1	Référence de la classe com.lasalle84.mobile_darts.ActiviteFinPartie	6
9.1.1	Description détaillée	8
9.1.2	Documentation des fonctions membres	8
9.1.3	Documentation des données membres	12
9.2	Référence de la classe com.lasalle84.mobile_darts.ActiviteJouerPartie	14
9.2.1	Description détaillée	16
9.2.2	Documentation des fonctions membres	17
9.2.3	Documentation des données membres	24
9.3	Référence de la classe com.lasalle84.mobile_darts.ActiviteLancerPartie	27

9.3.1	Description détaillée	30
9.3.2	Documentation des fonctions membres	30
9.3.3	Documentation des données membres	40
9.4	Référence de la classe <code>com.lasalle84.mobile_darts.lhmMobileDarts</code>	43
9.4.1	Description détaillée	44
9.4.2	Documentation des fonctions membres	44
9.4.3	Documentation des données membres	47
9.5	Référence de la classe <code>com.lasalle84.mobile_darts.Joueur</code>	48
9.5.1	Description détaillée	49
9.5.2	Documentation des constructeurs et destructeur	49
9.5.3	Documentation des fonctions membres	50
9.5.4	Documentation des données membres	52
9.6	Référence de la classe <code>com.lasalle84.mobile_darts.Partie</code>	53
9.6.1	Description détaillée	55
9.6.2	Documentation des constructeurs et destructeur	55
9.6.3	Documentation des fonctions membres	56
9.6.4	Documentation des données membres	61
9.7	Référence de la classe <code>com.lasalle84.mobile_darts.Peripherique</code>	65
9.7.1	Description détaillée	68
9.7.2	Documentation des constructeurs et destructeur	68
9.7.3	Documentation des fonctions membres	68
9.7.4	Documentation des données membres	72
9.8	Référence de la classe <code>Thread</code>	76
9.9	Référence de la classe <code>com.lasalle84.mobile_darts.TReception</code>	76
9.9.1	Description détaillée	78
9.9.2	Documentation des fonctions membres	78
9.9.3	Documentation des données membres	80
9.10	Référence de la classe <code>com.lasalle84.mobile_darts.TypeJeu</code>	81
9.10.1	Description détaillée	82
9.10.2	Documentation des constructeurs et destructeur	82
9.10.3	Documentation des fonctions membres	83
9.10.4	Documentation des données membres	85

10 Documentation des fichiers	86
10.1 Référence du fichier ActiviteFinPartie.java	86
10.1.1 Description détaillée	87
10.2 ActiviteFinPartie.java	87
10.3 Référence du fichier ActiviteJouerPartie.java	89
10.3.1 Description détaillée	89
10.4 ActiviteJouerPartie.java	90
10.5 Référence du fichier ActiviteLancerPartie.java	94
10.5.1 Description détaillée	94
10.6 ActiviteLancerPartie.java	94
10.7 Référence du fichier IhmMobileDarts.java	99
10.7.1 Description détaillée	100
10.8 IhmMobileDarts.java	100
10.9 Référence du fichier Joueur.java	101
10.9.1 Description détaillée	101
10.10 Joueur.java	102
10.11 Référence du fichier Partie.java	102
10.11.1 Description détaillée	103
10.12 Partie.java	103
10.13 Référence du fichier Peripherique.java	105
10.13.1 Description détaillée	106
10.14 Peripherique.java	106
10.15 Référence du fichier Planification.md	108
10.16 Planification.md	108
10.17 Référence du fichier Protocole.md	109
10.18 Protocole.md	109
10.19 Référence du fichier README.md	110
10.20 README.md	110
10.21 Référence du fichier TReception.java	111
10.21.1 Description détaillée	112
10.22 TReception.java	112
10.23 Référence du fichier TypeJeu.java	113
10.23.1 Description détaillée	114
10.24 TypeJeu.java	114

1 Application Android

1.1 Table des matières

- [README](#)
- [A propos](#)
- [Licence GPL](#)

1.2 Informations

Auteur

Date

2021

Version

1.1

Voir également

<https://svn.riouxsvn.com/>

2 Module de gestion de partie

Les joueurs paramètrent et lancent la partie à partir d'une application sur un terminal mobile (sous Android).

Fonctionnalités nécessaires au fonctionnement du module

- Saisir les noms des joueurs.
- Paramétrer une partie :
 - le type de jeu :
 - 501 double out
 - 301 double out
 - ...
 - le nombre de joueurs.
 - le nombre de manches gagnantes.
- Lancer la partie.
- Gérer et visualiser le déroulement de la partie :
 - changer automatiquement de joueur
 - affichage des points
- Communiquer avec les modules en Bluetooth

Tâches devant être accomplies au cours du projet

- Création d'une base de données fonctionnelle et complétée
- Création d'un joueur
- Paramétrage d'une partie
- Lancement et gestion d'une partie
- Assurer la liaison bluetooth et la visibilité de la connexion aux modules
- Transmission des informations de la partie aux modules
- Déploiement de l'application mobile

Production attendue au cours du projet

- Une application informatique fonctionnelle
- Un modèle UML complet de la partie à développer
- Un code source commenté de l'application
- Les documentations associés au module

Planification des itérations

Iteration 1	Iteration 2	Iteration 3
Créer le squelette de l'application	Lancer une partie	Consulter l'historique des parties des joueurs
Créer un profil de joueur	Dialoguer avec les modules	Enregistrer les données des manches
	Gérer le déroulement d'une partie	Purger les parties

3 Protocole DARTS

Version : 1.0

Avis : validé le 11/06/2021 par Thierry VAIRA

Format général des trames

Type de contenu : Caractères ASCII

Délimiteurs

- Début : \$DARTS
- Fin : \r\n
- Champ de début : ;

Liste des trames

- Trame de début : START
- Trame de touche : HIT
- Trame de joueur suivant : NEXT
- Trame de fin : END
- Trame de réinitialisation : RESET
- Trame d'acquiescement : OK

Sens des trames

- Émetteur : Application mobile Darts, Module cible Darts.
- Récepteur : ihm Darts, Application mobile Darts, Module cible Darts.

Voir diagramme des échanges

Format détaillé des trames

— Trame début :

Format : `$DARTS;START;TYPE_PARTIE;NB_JOUEURS[;NOMS]\r\n` Exemple : `$DARTS;START;501;2;Billy;Robert`

Types de partie : 301 301_DOUBLE_OUT 501 501_DOUBLE_OUT

Remarque : les champs entre crochets [] sont optionnels.

— Trame de touche :

Format : `$DARTS;HIT;valeurTouche;multiplicateurTouche;ID\r\n`

Les valeurs possibles pour valeurTouche sont :

- Les secteurs 1 à 20;
- Le centre de la cible (la bulle ou Bull's Eye);
- Les valeurs possibles pour multiplicateurTouche sont : 1, 2 (double) et 3 (triple)
- Les valeurs possibles pour ID sont : 0 à 2

Exemples :

```
$DART;HIT;25;2;0\r\n -> Fléchette bullseye (50 points)
$DART;HIT;25;1;0\r\n -> Fléchette demi-bulle (25 points)
```

Les simples :

```
$DART;HIT;1;1;0\r\n -> 1 pts
...
$DART;HIT;20;1;0\r\n -> 20 pts
$DART;HIT;25;1;0\r\n -> 20 pts
```

Les doubles :

```
$DART;HIT;1;2;0\r\n -> 2 pts
...
$DART;HIT;20;2;0\r\n
$DART;HIT;25;2;0\r\n
```

```
$DART;HIT;20;2;0\r\n -> 40 pts
```

Les triples :

```
$DART;HIT;1;3;0\r\n -> 3 pts
...
$DART;HIT;20;3;0\r\n -> 60 pts
```

— Trame d'acquiescement :

Format : `$DARTS;OK\r\n`

Remarque : Les trames de touche sont acquiescées avec les modules cible et mobile.

— Trame de réinitialisation :

Format : `$DARTS;RESET\r\n`

Remarque : La trame RESET permet de réinitialiser le module cible en cas d'erreur ou à la fin d'une partie. Cela permet d'annuler une partie en cours.

— Trame de fin :

Format : `$DARTS;END;(LE GAGNANT)\r\n`

— Trame de joueur suivant :

Format : `$DARTS;NEXT\r\n`

Echanges

4 README

4.1 Mobile-DARTS

4.1.1 Présentation

Réalisé en Java pour Android

4.1.2 Informations

Auteur

Date

2021

Version

1.1

Voir également

<https://svn.riouxsvn.com/>

5 A propos

Auteur

Date

2021

Version

1.1

Voir également

<https://svn.riouxsvn.com/>

6 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

7 Liste des choses à faire

Membre [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler](#) ()

Traiter les messages

8 Documentation des espaces de nommage

8.1 Paquetage com

Paquetages

- package [lasalle84](#)

8.2 Paquetage com.lasalle84

Paquetages

- package [mobile_darts](#)

8.3 Paquetage com.lasalle84.mobile_darts

Classes

- class [ActiviteFinPartie](#)
L'activité [ActiviteFinPartie](#).
- class [ActiviteJouerPartie](#)
L'activité [ActiviteJouerPartie](#).
- class [ActiviteLancerPartie](#)
L'activité [ActiviteLancerPartie](#).
- class [IhmMobileDarts](#)
L'activité principale de l'application DARTS.
- class [Joueur](#)
La classe qui contient toute la gestion des joueurs.
- class [Partie](#)
La classe qui contient toute la gestion de partie.
- class [Peripherique](#)
La classe qui contient toute la gestion des périphériques.
- class [TReception](#)
La classe qui contient toute la gestion des thread de reception et envoi de trames.
- class [TypeJeu](#)
La classe qui contient toute la gestion des types de parties.

9 Documentation des classes

9.1 Référence de la classe com.lasalle84.mobile_darts.ActiviteFinPartie

L'activité [ActiviteFinPartie](#).

Grphe de collaboration de com.lasalle84.mobile_darts.ActiviteFinPartie :

com.lasalle84.mobile _darts.ActiviteFinPartie
<ul style="list-style-type: none"> - adapteur - affichageGagnant - boutonFin - classement - lesJoueur - nomGagnant - TAG
<ul style="list-style-type: none"> + afficherJoueurs() + initialiserWidgets() + onClick() + recupererInfomationPartie() + recupererWidgets() + trierLesJoueurs() # onCreate() # onDestroy() # onPause() # onResume() # onStart() # onStop() - declancherConfettis()

Fonctions membres publiques

- void [afficherJoueurs](#) ()
Méthode appelée à la création de l'activité
- void [initialiserWidgets](#) ()
Méthode permettant d'initialiser les widgets.
- void [onClick](#) (View element)
Méthode appelée à la création de l'activité
- void [recupererInfomationPartie](#) ()
Méthode appelée à la création de l'activité
- void [recupererWidgets](#) ()
Méthode permettant de récupérer les widgets du layout.
- void [trierLesJoueurs](#) ()
Méthode permet de trier les joueur par rapport au score.

Fonctions membres protégées

- void [onCreate](#) (Bundle savedInstanceState)
Méthode appelée à la création de l'activité
- void [onDestroy](#) ()
Méthode appelée à la destruction de l'application (après [onStop\(\)](#)) et détruite par le système Android)
- void [onPause](#) ()
Méthode appelée après qu'une boîte de dialogue s'est affichée (on reprend sur un [onResume\(\)](#)) ou avant [onStop\(\)](#) (activité plus visible)
- void [onResume](#) ()
Méthode appelée après [onStart\(\)](#) ou après [onPause\(\)](#)
- void [onStart](#) ()
Méthode appelée au démarrage après le [onCreate\(\)](#) ou un restart après un [onStop\(\)](#)
- void [onStop](#) ()
Méthode appelée lorsque l'activité n'est plus visible.

Fonctions membres privées

— void [declancherConfettis](#) ()

Attributs privés

— ArrayAdapter< String > [adapteur](#) = null
Adapteur du classement des joueurs.
 — TextView [affichageGagnant](#) = null
Widget texte avec le nom du gagnant.
 — Button [boutonFin](#) = null
Bouton pour fermer la page.
 — ListView [classement](#) = null
Widget avec la liste des joueurs avec leur score.
 — ArrayList< [Joueur](#) > [lesJoueur](#) = null
Conteneur avec des objets [Joueur](#).
 — String [nomGagnant](#) = null
Nom du Gagnant.
 — final String [TAG](#) = "_ActiviteFinPartie_"

9.1.1 Description détaillée

L'activité [ActiviteFinPartie](#).

Définition à la ligne [36](#) du fichier [ActiviteFinPartie.java](#).

9.1.2 Documentation des fonctions membres

9.1.2.1 afficherJoueurs()

```
com.lasalle84.mobile_darts.ActiviteFinPartie.afficherJoueurs ( )
```

Méthode appelée à la création de l'activité

Définition à la ligne [208](#) du fichier [ActiviteFinPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteFinPartie.adapteur](#), [com.lasalle84.mobile_darts.ActiviteFinPartie.affichageGagnant](#), [com.lasalle84.mobile_darts.ActiviteFinPartie.classement](#), [com.lasalle84.mobile_darts.ActiviteFinPartie.lesJoueur](#), [com.lasalle84.mobile_darts.ActiviteFinPartie.nomGagnant](#), et [com.lasalle84.mobile_darts.ActiviteFinPartie.TAG](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.onCreate\(\)](#).

```
00209     {
00210         Log.d(TAG, "afficherJoueurs()");
00211         adapteur = new ArrayAdapter<String>(this, android.R.layout.simple_spinner_item);
00212         for (int i = 0; i < lesJoueur.size(); i++)
00213         {
00214             adapteur.add(lesJoueur.get(i).getNom() + " : " +
00215             lesJoueur.get(i).getScore());
00216         }
00217         classement.setAdapter(adapteur);
00218         String gagnant = nomGagnant + " a gagné la partie";
00219         affichageGagnant.setText(gagnant);
00219     }
```

9.1.2.2 declancherConfettis()

```
void com.lasalle84.mobile_darts.ActiviteFinPartie.declancherConfettis ( ) [private]
```

Définition à la ligne 72 du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.onCreate\(\)](#).

```
00073     {
00074         final KonfettiView konfettiView = findViewById(R.id.viewKonfetti);
00075         konfettiView.build()
00076             .addColors(Color.BLUE, Color.GREEN, Color.RED)
00077             .setDirection(0.0, 359.0)
00078             .setSpeed(1f, 5f)
00079             .setFadeOutEnabled(true)
00080             .setTimeToLive(2000L)
00081             .addShapes(Shape.Square.INSTANCE, Shape.Circle.INSTANCE)
00082             .addSizes(new Size(12, 5f))
00083             .setPosition(-50f, konfettiView.getWidth() + 50f, -50f, -50f)
00084             .streamFor(500, 5000L);
00085     }
```

9.1.2.3 initialiserWidgets()

```
com.lasalle84.mobile_darts.ActiviteFinPartie.initialiserWidgets ( )
```

Méthode permettant d'initialiser les widgets.

Définition à la ligne 171 du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.onCreate\(\)](#).

```
00172     {
00173         Log.d(TAG, "initialiserWidgets()");
00174         boutonFin.setOnClickListener((View.OnClickListener) this);
00175     }
```

9.1.2.4 onClick()

```
com.lasalle84.mobile_darts.ActiviteFinPartie.onClick (
    View element )
```

Méthode appelée à la création de l'activité

Définition à la ligne 92 du fichier [ActiviteFinPartie.java](#).

```
00093     {
00094         Log.d(TAG, "onClick() " + element.toString());
00095
00096         if(element == boutonFin)
00097         {
00098             super.finish();
00099         }
00100     }
```

9.1.2.5 onCreate()

```
com.lasalle84.mobile_darts.ActiviteFinPartie.onCreate (
    Bundle savedInstanceState ) [protected]
```

Méthode appelée à la création de l'activité

Paramètres

<i>savedInstanceState</i>	
---------------------------	--

Définition à la ligne 60 du fichier [ActiviteFinPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteFinPartie.afficherJoueurs\(\)](#), [com.lasalle84.mobile_darts.ActiviteFinPartie.declancherConfettis\(\)](#), [com.lasalle84.mobile_darts.ActiviteFinPartie.initialiserWidgets\(\)](#), [com.lasalle84.mobile_darts.ActiviteFinPartie.recupererInformationPartie\(\)](#), [com.lasalle84.mobile_darts.ActiviteFinPartie.recupererWidgets\(\)](#), et [com.lasalle84.mobile_darts.ActiviteFinPartie.trierLesJoueurs\(\)](#).

```
00061    {
00062        super.onCreate(savedInstanceState);
00063        setContentView(R.layout.activity_activite_fin_partie);
00064        recupererWidgets();
00065        initialiserWidgets();
00066        recupererInformationPartie();
00067        trierLesJoueurs();
00068        afficherJoueurs();
00069        declancherConfettis();
00070    }
```

9.1.2.6 onDestroy()

```
void com.lasalle84.mobile_darts.ActiviteFinPartie.onDestroy () [protected]
```

Méthode appelée à la destruction de l'application (après [onStop\(\)](#) et détruite par le système Android)

Définition à la ligne 146 du fichier [ActiviteFinPartie.java](#).

```
00147    {
00148        super.onDestroy();
00149        Log.d(TAG, "[onDestroy]");
00150    }
```

9.1.2.7 onPause()

```
void com.lasalle84.mobile_darts.ActiviteFinPartie.onPause () [protected]
```

Méthode appelée après qu'une boîte de dialogue s'est affichée (on reprend sur un [onResume\(\)](#)) ou avant [onStop\(\)](#) (activité plus visible)

Définition à la ligne 126 du fichier [ActiviteFinPartie.java](#).

```
00127    {
00128        super.onPause();
00129        Log.d(TAG, "[onPause]");
00130    }
```

9.1.2.8 onResume()

```
void com.lasalle84.mobile_darts.ActiviteFinPartie.onResume ( ) [protected]
```

Méthode appelée après [onStart\(\)](#) ou après [onPause\(\)](#)

Définition à la ligne [116](#) du fichier [ActiviteFinPartie.java](#).

```
00117     {
00118         super.onResume();
00119         Log.d(TAG, "[onResume]");
00120     }
```

9.1.2.9 onStart()

```
void com.lasalle84.mobile_darts.ActiviteFinPartie.onStart ( ) [protected]
```

Méthode appelée au démarrage après le [onCreate\(\)](#) ou un restart après un [onStop\(\)](#)

Définition à la ligne [106](#) du fichier [ActiviteFinPartie.java](#).

```
00107     {
00108         super.onStart();
00109         Log.d(TAG, "[onStart]");
00110     }
```

9.1.2.10 onStop()

```
void com.lasalle84.mobile_darts.ActiviteFinPartie.onStop ( ) [protected]
```

Méthode appelée lorsque l'activité n'est plus visible.

Définition à la ligne [136](#) du fichier [ActiviteFinPartie.java](#).

```
00137     {
00138         super.onStop();
00139         Log.d(TAG, "[onStop]");
00140     }
```

9.1.2.11 recupererInfomationPartie()

```
com.lasalle84.mobile_darts.ActiviteFinPartie.recupererInfomationPartie ( )
```

Méthode appelée à la création de l'activité

Définition à la ligne [182](#) du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.onCreate\(\)](#).

```
00183     {
00184         Log.d(TAG, "recupererInfomationPartie()");
00185         lesJoueur = (ArrayList<Joueur>) getIntent().getSerializableExtra("joueurs");
00186         nomGagnant = getIntent().getStringExtra("gagnant");
00187         Log.d(TAG, "Le nom du gagnant est" + nomGagnant);
00188     }
```

9.1.2.12 recupererWidgets()

```
com.lasalle84.mobile_darts.ActiviteFinPartie.recupererWidgets ( )
```

Méthode permettant de récupérer les widgets du layout.

Définition à la ligne 157 du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.onCreate\(\)](#).

```
00158      {
00159          Log.d(TAG, "recupererWidgets()");
00160          affichageGagnant = (TextView) findViewById(R.id.texteGagnant);
00161          classement = (ListView) findViewById(R.id.ResultaPartie);
00162          boutonFin = (Button) findViewById(R.id.boutonFin);
00163      }
00164  }
```

9.1.2.13 trierLesJoueurs()

```
com.lasalle84.mobile_darts.ActiviteFinPartie.trierLesJoueurs ( )
```

Méthode permet de trier les joueur par rapport au score.

Définition à la ligne 196 du fichier [ActiviteFinPartie.java](#).

Références [com.lasalle84.mobile_darts.Joueur.getScore\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.onCreate\(\)](#).

```
00197      {
00198          Log.d(TAG, "trierLesJoueurs()");
00199          lesJoueur.sort(Comparator.comparingInt(Joueur::getScore));
00200      }
00201  }
```

9.1.3 Documentation des données membres

9.1.3.1 adapteur

```
ArrayAdapter<String> com.lasalle84.mobile_darts.ActiviteFinPartie.adapteur = null [private]
```

Adapteur du classement des joueurs.

Définition à la ligne 50 du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.afficherJoueurs\(\)](#).

9.1.3.2 `affichageGagnant`

```
TextView com.lasalle84.mobile_darts.ActiviteFinPartie.affichageGagnant = null [private]
```

Widget texte avec le nom du gagnant.

Attributs

Définition à la ligne 45 du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.afficherJoueurs\(\)](#).

9.1.3.3 `boutonFin`

```
Button com.lasalle84.mobile_darts.ActiviteFinPartie.boutonFin = null [private]
```

Bouton pour fermer la page.

Définition à la ligne 47 du fichier [ActiviteFinPartie.java](#).

9.1.3.4 `classement`

```
ListView com.lasalle84.mobile_darts.ActiviteFinPartie.classement = null [private]
```

Widget avec la liste des joueurs avec leur score.

Définition à la ligne 46 du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.afficherJoueurs\(\)](#).

9.1.3.5 `lesJoueur`

```
ArrayList<Joueur> com.lasalle84.mobile_darts.ActiviteFinPartie.lesJoueur = null [private]
```

Conteneur avec des objets [Joueur](#).

Définition à la ligne 48 du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.afficherJoueurs\(\)](#).

9.1.3.6 `nomGagnant`

```
String com.lasalle84.mobile_darts.ActiviteFinPartie.nomGagnant = null [private]
```

Nom du Gagnant.

Définition à la ligne 49 du fichier [ActiviteFinPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteFinPartie.afficherJoueurs\(\)](#).

9.1.3.7 TAG

```
final String com.lasalle84.mobile_darts.ActiviteFinPartie.TAG = "_ActiviteFinPartie_" [private]
```

Constantes

Définition à la ligne [41](#) du fichier [ActiviteFinPartie.java](#).

Référéncé par [com.lasalle84.mobile_darts.ActiviteFinPartie.afficherJoueurs\(\)](#).

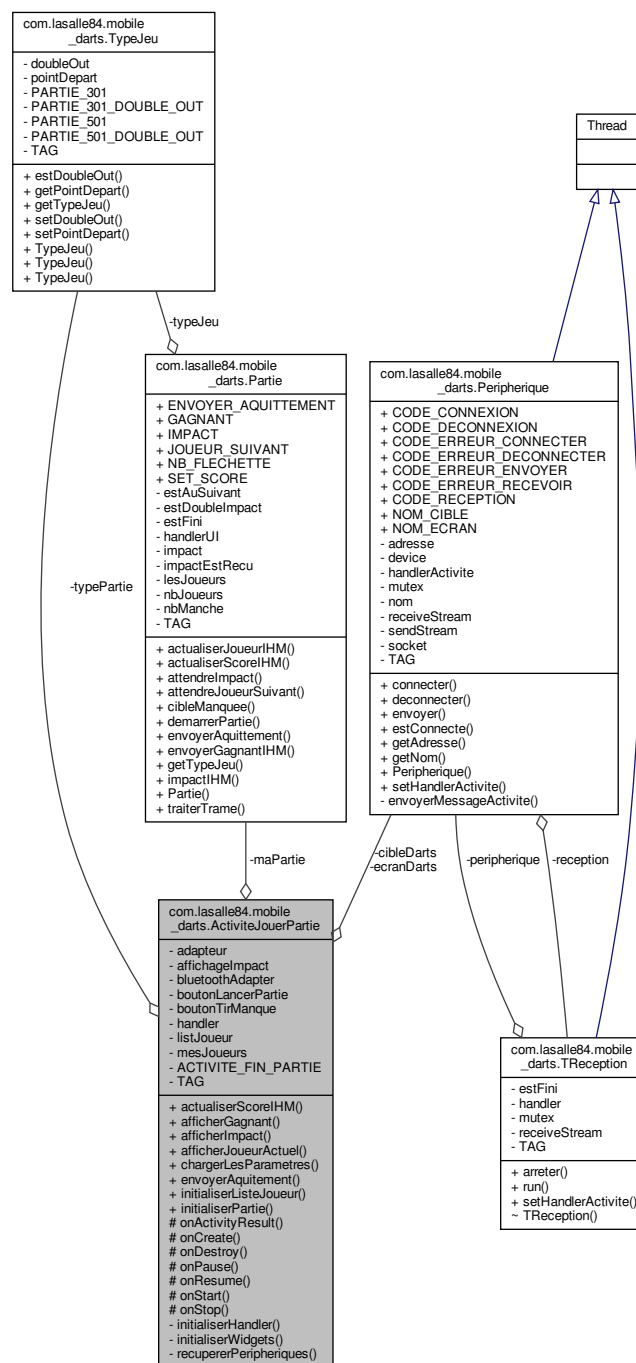
La documentation de cette classe a été générée à partir du fichier suivant :

— [ActiviteFinPartie.java](#)

9.2 Référence de la classe `com.lasalle84.mobile_darts.ActiviteJouerPartie`

L'activité [ActiviteJouerPartie](#).

Graphe de collaboration de com.lasalle84.mobile_darts.ActiviteJouerPartie :



Fonctions membres publiques

- void [actualiserScoreIHM](#) (String joueur, int score)
- void [afficherGagnant](#) (String gagnant)
- void [afficherImpact](#) (int Touche, int multiplicateur)
- void [afficherJoueurActuel](#) (String joueur)
- void [chargerLesParametres](#) ()
- void [envoyerAquittement](#) ()
- void [initialiserListeJoueur](#) ()
- Initialisation des joueurs.
- void [initialiserPartie](#) ()
- Initialisation de la partie.

Fonctions membres protégées

- void `onActivityResult` (int requestCode, int resultCode, Intent data)
Traite le retour des activités.
- void `onCreate` (Bundle savedInstanceState)
Méthode appelée à la création de l'activité
- void `onDestroy` ()
Méthode appelée à la destruction de l'application (après `onStop()` et détruite par le système Android)
- void `onPause` ()
Méthode appelée après qu'une boîte de dialogue s'est affichée (on reprend sur un `onResume()`) ou avant `onStop()` (activité plus visible)
- void `onResume` ()
Méthode appelée après `onStart()` ou après `onPause()`
- void `onStart` ()
Méthode appelée au démarrage après le `onCreate()` ou un restart après un `onStop()`
- void `onStop` ()
Méthode appelée lorsque l'activité n'est plus visible.

Fonctions membres privées

- void `initialiserHandler` ()
Initialisation du handler.
- void `initialiserWidgets` ()
Initialisation des Widgets.
- void `recupererPeripheriques` ()

Attributs privés

- ArrayAdapter< String > `adapteur` = null
Adapteur qui affiche quel joueur doit jouer et son score.
- TextView `affichageImpact` = null
Texte qui affiche la volée actuelle.
- BluetoothAdapter `bluetoothAdapter` = null
Bluetooth Adapteur.
- Button `boutonLancerPartie`
Bouton permettant le lancement de la partie.
- Button `boutonTirManque`
Bouton pour la commande de tir manqué
- Peripherique cibleDarts = null
Le périphérique Bluetooth de la cible.
- Peripherique ecranDarts = null
Le périphérique Bluetooth de l'écran.
- Handler `handler` = null
Le Handler qui permet de communiquer avec le thread.
- ListView `listJoueur` = null
Listview des Joueurs.
- Partie `maPartie` = null
La partie jouée.
- ArrayList< Joueur > `mesJoueurs` = null
liste des Joueurs
- TypeJeu `typePartie` = null

Attributs privés statiques

- static final int `ACTIVITE_FIN_PARTIE` = 0
Code pour l'Activite fin partie.
- static final String `TAG` = "_ActiviteJouerPartie_"

9.2.1 Description détaillée

L'activité `ActiviteJouerPartie`.

Définition à la ligne 35 du fichier `ActiviteJouerPartie.java`.

9.2.2 Documentation des fonctions membres

9.2.2.1 actualiserScoreIHM()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.actualiserScoreIHM (
    String joueur,
    int score )
```

Définition à la ligne 155 du fichier [ActiviteJouerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserPartie\(\)](#).

```
00156     {
00157         Log.d(TAG, "actualiserScoreIHM()");
00158         int idJoueur = 0;
00159
00160         for (int i = 0; i < mesJoueurs.size(); i++)
00161         {
00162             if (mesJoueurs.get(i).getNom().equals(joueur))
00163             {
00164                 idJoueur = i;
00165             }
00166         }
00167         mesJoueurs.get(idJoueur).setScore(score);
00168         String string = mesJoueurs.get(idJoueur).getNom() + ": " +
00169         mesJoueurs.get(idJoueur).getScore();
00169         adapteur.remove(adapteur.getItem(idJoueur));
00170         adapteur.insert(string, idJoueur);
00171     }
```

9.2.2.2 afficherGagnant()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.afficherGagnant (
    String gagnant )
```

Définition à la ligne 205 du fichier [ActiviteJouerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#).

```
00206     {
00207         Log.d(TAG, "afficherGagnant() : Le nom du gagnant est : " + gagnant);
00208         Intent AfficherGagnant = new Intent(this, ActiviteFinPartie.class);
00209         List<Joueur> lesJoueur = new ArrayList<Joueur>();
00210         for (int i = 0; i < mesJoueurs.size(); i++)
00211         {
00212             if (mesJoueurs.get(i).getNom().equals(gagnant))
00213             {
00214                 AfficherGagnant.putExtra("gagnant", mesJoueurs.get(i).getNom());
00215                 lesJoueur.add(mesJoueurs.get(i));
00216             }
00217             else
00218             {
00219                 lesJoueur.add(mesJoueurs.get(i));
00220             }
00221         }
00222         AfficherGagnant.putExtra("joueurs", (Serializable) lesJoueur);
00223     }
00224     startActivityResult(AfficherGagnant, ACTIVITE_FIN_PARTIE);
00225     finish();
00226 }
00227 }
```

9.2.2.3 afficherImpact()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.afficherImpact (
    int Touche,
    int multiplicateur )
```

Définition à la ligne 237 du fichier [ActiviteJouerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#).

```
00238     {
00239         Log.d(TAG, "afficherImpact()");
00240         String multiplication = new String();
00241         switch (multiplicateur)
00242         {
00243             case 0:
00244                 multiplication = "MISS";
00245                 break;
00246             case 1:
00247                 multiplication = "Simple ";
00248                 break;
00249             case 2:
00250                 multiplication = "Double ";
00251                 break;
00252             case 3:
00253                 multiplication = "Triple ";
00254                 break;
00255         }
00256         if (multiplication == "MISS")
00257         {
00258             affichageImpact.setText(affichageImpact.getText().toString().
concat("MISS "));
00259         }
00260         else
00261         {
00262             affichageImpact.setText(affichageImpact.getText().toString().
concat(multiplication + Integer.toString(Touche) + " "));
00263         }
00264     }
```

9.2.2.4 afficherJoueurActuel()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.afficherJoueurActuel (
    String joueur )
```

Définition à la ligne 179 du fichier [ActiviteJouerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#).

```
00180     {
00181         Log.d(TAG, "afficherJoueurActuel()");
00182         String joueurActuel;
00183         for (int i = 0; i < mesJoueurs.size(); i++)
00184         {
00185             if (mesJoueurs.get(i).getNom().equals(joueur))
00186             {
00187                 joueurActuel = "-> " + mesJoueurs.get(i).getNom() + ": " +
mesJoueurs.get(i).getScore();
00188             }
00189             else {
00190                 joueurActuel = mesJoueurs.get(i).getNom() + ": " +
mesJoueurs.get(i).getScore();
00191             }
00192             adapteur.remove(adapteur.getItem(i));
00193             adapteur.insert(joueurActuel, i);
00194         }
00195         affichageImpact.setText("");
00196     }
```

9.2.2.5 chargerLesParametres()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.chargerLesParametres ( )
```

Définition à la ligne 271 du fichier [ActiviteJouerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.onCreate\(\)](#).

```
00272     {
00273         mesJoueurs = new ArrayList<Joueur>();
00274         mesJoueurs = (ArrayList<Joueur>) getIntent().getSerializableExtra("LesJoueurs");
00275         Iterator<Joueur> it = mesJoueurs.iterator();
00276
00277         while(it.hasNext())
00278         {
00279             Log.d(TAG, "le joueur "+ it.next().getNom() + "est chargé");
00280         }
00281         typePartie = new TypeJeu(getIntent().getIntExtra("TypeMode",0));
00282     }
```

9.2.2.6 envoyerAquitement()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.envoyerAquitement ( )
```

Définition à la ligne 284 du fichier [ActiviteJouerPartie.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.envoyer\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#).

```
00285     {
00286         Log.d(TAG, "envoyerAquitement()");
00287         cibleDarts.envoyer("$DARTS;OK;" + "\\r\\n");
00288     }
```

9.2.2.7 initialiserHandler()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler ( ) [private]
```

Initialisation du handler.

Définition à la ligne 325 du fichier [ActiviteJouerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteJouerPartie.actualiserScoreIHM\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.afficherGagnant\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.afficherImpact\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.afficherJoueurActuel\(\)](#), [com.lasalle84.mobile_darts.Peripherique.CODE_RECEPTION](#), [com.lasalle84.mobile_darts.Peripherique.envoyer\(\)](#), [com.lasalle84.mobile_darts.Partie.ENVOYER_AQUITTEMENT](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.envoyerAquitement\(\)](#), [com.lasalle84.mobile_darts.Partie.GAGNANT](#), [com.lasalle84.mobile_darts.Partie.IMPACT](#), [com.lasalle84.mobile_darts.Partie.JOUEUR_SUIVANT](#), [com.lasalle84.mobile_darts.Peripherique.NOM_CIBLE](#), [com.lasalle84.mobile_darts.Peripherique.NOM_ECRAN](#), [com.lasalle84.mobile_darts.Partie.SET_SCORE](#), et [com.lasalle84.mobile_darts.Partie.traiterTrame\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.onCreate\(\)](#).

```

00326     {
00327         Log.d(TAG, "[initialiserHandler]");
00328         this.handler = new Handler(this.getMainLooper())
00329     {
00330         @Override
00331         public void handleMessage(Message msg)
00332         {
00333             super.handleMessage(msg);
00334             Bundle b = msg.getData();
00335             switch(msg.what)
00336             {
00337                 case Peripherique.CODE_RECEPTION:
00338                     if(b.getString("nom").equals(Peripherique.NOM_CIBLE))
00339                     {
00340                         String donnees = b.getString("donnees");
00341                         Log.d(TAG, "[handleMessage] Nom : " + b.getString("nom") + " - Adresse : "
00342                             + b.getString("adresse") + " -> " + b.getString("donnees"));
00343                         maPartie.traiterTrame(donnees);
00344                     }
00345                     else if(b.getString("nom").equals(Peripherique.NOM_ECRAN))
00346                     {
00347                         String donnees = b.getString("donnees");
00348                         Log.d(TAG, "[handleMessage] Nom : " + b.getString("nom") + " - Adresse : "
00349                             + b.getString("adresse") + " -> " + b.getString("donnees"));
00350                         maPartie.traiterTrame(donnees);
00351                     }
00352                     break;
00353                 case Partie.SET_SCORE:
00354                     Log.d(TAG, "[handleMessage] : SET_SCORE " + "Joueur: " + b.getString("joueur")
00355                         + " Score: " + b.getInt("score"));
00356                     actualiserScoreIHM(b.getString("joueur"), b.getInt("score"));
00357                     break;
00358                 case Partie.IMPACT:
00359                     Log.d(TAG, "[handleMessage] : IMPACT " + "Joueur: " + b.getString("joueur")
00360                         + " Touche: " + b.getInt("Touche") + " Multiplicateur Touche: " + b.getInt("MultiplicateurTouche"));
00361                     afficherImpact(b.getInt("Touche"), b.getInt("MultiplicateurTouche"));
00362                     if(ecranDarts != null)
00363                     {
00364                         ecranDarts.envoyer("$DARTS;HIT;" + b.getInt("Touche") + " "
00365                             + b.getInt("MultiplicateurTouche") + ";" + b.getInt("ID") + "\r\n");
00366                     }
00367                     break;
00368                 case Partie.JOUEUR_SUIVANT:
00369                     Log.d(TAG, "[handleMessage] : JOUEUR_SUIVANT " + "Joueur: " + b.getString("joueur"));
00370                     afficherJoueurActuel(b.getString("joueur"));
00371                     break;
00372                 case Partie.ENVOYER_AQUITTEMENT:
00373                     Log.d(TAG, "[handleMessage] : ENVOYER_AQUITTEMENT");
00374                     envoyerAquittement();
00375                     break;
00376                 case Partie.GAGNANT:
00377                     Log.d(TAG, "[handleMessage] : AFFICHER_GAGNANT");
00378                     if(ecranDarts != null)
00379                     {
00380                         ecranDarts.envoyer("$DARTS;END;" + b.getString("gagnant") + " "
00381                             + "\r\n");
00382                     }
00383                     cibleDarts.envoyer("$DARTS;RESET\r\n");
00384                     afficherGagnant(b.getString("gagnant"));
00385                     break;
00386             }
00387         }
00388     };
00389 }

```

9.2.2.8 initialiserListeJoueur()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserListeJoueur ( )
```

Initialisation des joueurs.

Définition à la ligne 411 du fichier [ActiviteJouerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.onCreate\(\)](#).

```

00412     {
00413         Log.d(TAG, "initialiserListeJoueur()");
00414         adapteur = new ArrayAdapter<String>(this, android.R.layout.simple_spinner_item);
00415         for (int i = 0; i < mesJoueurs.size(); i++)
00416         {
00417             adapteur.add(mesJoueurs.get(i).getNom());
00418         }
00419         listJoueur.setAdapter(adapteur);
00420     }

```

9.2.2.9 initialiserPartie()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserPartie ( )
```

Initialisation de la partie.

Définition à la ligne 388 du fichier [ActiviteJouerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteJouerPartie.actualiserScoreIHM\(\)](#), [com.lasalle84.mobile_darts.Peripherique.↵
envoyer\(\)](#), [com.lasalle84.mobile_darts.Joueur.getNom\(\)](#), [com.lasalle84.mobile_darts.TypeJeu.getPointDepart\(\)](#), [com.lasalle84.↵
mobile_darts.TypeJeu.getTypeJeu\(\)](#), et [com.lasalle84.mobile_darts.Joueur.setScore\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserWidgets\(\)](#).

```

00388     {
00389         Log.d(TAG, "initialiserPartie()");
00390         Iterator<Joueur> it = mesJoueurs.iterator();
00391         String trameNomJoueurs = new String();
00392         while(it.hasNext())
00393         {
00394             Joueur monJoueur = it.next();
00395             monJoueur.setScore(typePartie.getPointDepart());
00396             actualiserScoreIHM(monJoueur.getNom(), typePartie.  
getPointDepart());
00397             trameNomJoueurs = trameNomJoueurs.concat("; " + monJoueur.getNom());
00398         }
00399         cibleDarts.envoyer("$DARTS;START;" + typePartie.  
getTypeJeu() + ";" + mesJoueurs.size() + trameNomJoueurs + "\r\n");
00400         //Log.d(TAG, "$DARTS;START;" + typePartie.getTypeJeu() + ";" + mesJoueurs.size() + trameNomJoueurs +  
"\r\n");
00401         if(ecranDarts != null)
00402         {
00403             ecranDarts.envoyer("$DARTS;START;" + typePartie.  
getTypeJeu() + ";" + mesJoueurs.size() + trameNomJoueurs + "\r\n");
00404         }
00405         Log.d(TAG, "Envoi de la trame à la cible et à l'écran");
00406     }

```

9.2.2.10 initialiserWidgets()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserWidgets ( ) [private]
```

Initialisation des Widgets.

Définition à la ligne 293 du fichier [ActiviteJouerPartie.java](#).

Références [com.lasalle84.mobile_darts.Partie.cibleManquee\(\)](#), [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#), et [com.↵
lasalle84.mobile_darts.ActiviteJouerPartie.initialiserPartie\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.onCreate\(\)](#).


```

00294     {
00295         Log.d(TAG, "initialiserWidgets()");
00296         boutonTirManque = (Button) findViewById(R.id.TirManque);
00297         boutonTirManque.setVisibility(View.INVISIBLE);
00298         boutonTirManque.setOnClickListener(view -> {
00299             Log.d(TAG, "[onClick] boutonTirManque");
00300             maPartie.cibleManquee();
00301         });
00302
00303         boutonLancerPartie = (Button) findViewById(R.id.LancerPartie);
00304         boutonLancerPartie.setOnClickListener(view -> {
00305             Log.d(TAG, "[onClick] boutonLancerPartie");
00306             initialiserPartie();
00307             new Thread()
00308             {
00309                 @Override
00310                 public void run() {
00311                     super.run();
00312                     maPartie.demarrerPartie();
00313                 }
00314             }.start();
00315             boutonLancerPartie.setVisibility(View.INVISIBLE);
00316             boutonTirManque.setVisibility(View.VISIBLE);
00317         });
00318         listJoueur = (ListView) findViewById(R.id.MesJoueur);
00319         affichageImpact = (TextView) findViewById(R.id.Impact);
00320     }

```

9.2.2.11 onActivityResult()

```

void com.lasalle84.mobile_darts.ActiviteJouerPartie.onActivityResult (
    int requestCode,
    int resultCode,
    Intent data ) [protected]

```

Traite le retour des activités.

Définition à la ligne 425 du fichier [ActiviteJouerPartie.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.deconnecter\(\)](#).

```

00426     {
00427         super.onActivityResult(requestCode, resultCode, data);
00428         Log.d(TAG, "[onActivityResult] Fin ActiviteFinPartie");
00429         if (ecranDarts != null)
00430         {
00431             écranDarts.deconnecter();
00432         }
00433         cibleDarts.deconnecter();
00434         super.finish();
00435     }

```

9.2.2.12 onCreate()

```

com.lasalle84.mobile_darts.ActiviteJouerPartie.onCreate (
    Bundle savedInstanceState ) [protected]

```

Méthode appelée à la création de l'activité

Paramètres

<i>savedInstanceState</i>	
---------------------------	--

Définition à la ligne 65 du fichier [ActiviteJouerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteJouerPartie.chargerLesParametres\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserListeJoueur\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserWidgets\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.recupererPeripheriques\(\)](#).

```
00066    {
00067        Log.d(TAG, "onCreate()");
00068        super.onCreate(savedInstanceState);
00069        setContentView(R.layout.activity_activite_jouer_partie);
00070
00071        initialiserHandler();
00072        initialiserWidgets();
00073        recupererPeripheriques();
00074        chargerLesParametres();
00075        initialiserListeJoueur();
00076        maPartie = new Partie(mesJoueurs, typePartie,
00077            handler);
00077    }
```

9.2.2.13 onDestroy()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.onDestroy ( ) [protected]
```

Méthode appelée à la destruction de l'application (après [onStop\(\)](#) et détruite par le système Android)

Définition à la ligne 123 du fichier [ActiviteJouerPartie.java](#).

```
00124    {
00125        super.onDestroy();
00126        Log.d(TAG, "[onDestroy]");
00127    }
```

9.2.2.14 onPause()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.onPause ( ) [protected]
```

Méthode appelée après qu'une boîte de dialogue s'est affichée (on reprend sur un [onResume\(\)](#)) ou avant [onStop\(\)](#) (activité plus visible)

Définition à la ligne 103 du fichier [ActiviteJouerPartie.java](#).

```
00104    {
00105        super.onPause();
00106        Log.d(TAG, "[onPause]");
00107    }
```

9.2.2.15 onResume()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.onResume ( ) [protected]
```

Méthode appelée après [onStart\(\)](#) ou après [onPause\(\)](#)

Définition à la ligne 93 du fichier [ActiviteJouerPartie.java](#).

```
00094    {
00095        super.onResume();
00096        Log.d(TAG, "[onResume]");
00097    }
```

9.2.2.16 onStart()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.onStart ( ) [protected]
```

Méthode appelée au démarrage après le [onCreate\(\)](#) ou un restart après un [onStop\(\)](#)

Définition à la ligne [83](#) du fichier [ActiviteJouerPartie.java](#).

```
00084    {
00085        super.onStart ();
00086        Log.d(TAG, "[onStart]");
00087    }
```

9.2.2.17 onStop()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.onStop ( ) [protected]
```

Méthode appelée lorsque l'activité n'est plus visible.

Définition à la ligne [113](#) du fichier [ActiviteJouerPartie.java](#).

```
00114    {
00115        super.onStop ();
00116        Log.d(TAG, "[onStop]");
00117    }
```

9.2.2.18 recupererPeripheriques()

```
void com.lasalle84.mobile_darts.ActiviteJouerPartie.recupererPeripheriques ( ) [private]
```

Définition à la ligne [135](#) du fichier [ActiviteJouerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteLancerPartie.getPeripheriqueCibleDarts\(\)](#), [com.lasalle84.mobile_darts.ActiviteLancerPartie.getPeripheriqueEcranDarts\(\)](#), et [com.lasalle84.mobile_darts.Peripherique.setHandlerActivite\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.onCreate\(\)](#).

```
00136    {
00137        Log.d(TAG, "recupererPeripheriques()");
00138
00139        cibleDarts = ActiviteLancerPartie.getPeripheriqueCibleDarts();
00140        ecranDarts = ActiviteLancerPartie.getPeripheriqueEcranDarts();
00141
00142        if(cibleDarts != null)
00143            cibleDarts.setHandlerActivite(handler);
00144        if(ecranDarts != null)
00145            ecranDarts.setHandlerActivite(handler);
00146    }
```

9.2.3 Documentation des données membres

9.2.3.1 ACTIVITE_FIN_PARTIE

```
final int com.lasalle84.mobile_darts.ActiviteJouerPartie.ACTIVITE_FIN_PARTIE = 0 [static], [private]
```

Code pour l'Activite fin partie.

Définition à la ligne 41 du fichier [ActiviteJouerPartie.java](#).

9.2.3.2 adapteur

```
ArrayAdapter<String> com.lasalle84.mobile_darts.ActiviteJouerPartie.adapteur = null [private]
```

Adapteur qui affiche quel joueur doit jouer et son score.

Définition à la ligne 52 du fichier [ActiviteJouerPartie.java](#).

9.2.3.3 affichageImpact

```
TextView com.lasalle84.mobile_darts.ActiviteJouerPartie.affichageImpact = null [private]
```

Texte qui affiche la vollée actuelle.

Définition à la ligne 54 du fichier [ActiviteJouerPartie.java](#).

9.2.3.4 bluetoothAdapter

```
BluetoothAdapter com.lasalle84.mobile_darts.ActiviteJouerPartie.bluetoothAdapter = null [private]
```

Bluetooth Adapteur.

Définition à la ligne 49 du fichier [ActiviteJouerPartie.java](#).

9.2.3.5 boutonLancerPartie

```
Button com.lasalle84.mobile_darts.ActiviteJouerPartie.boutonLancerPartie [private]
```

Bouton permettant le lancement de la partie.

Définition à la ligne 56 du fichier [ActiviteJouerPartie.java](#).

9.2.3.6 boutonTirManque

```
Button com.lasalle84.mobile_darts.ActiviteJouerPartie.boutonTirManque [private]
```

Bouton pour la commande de tir manqué

Définition à la ligne 55 du fichier [ActiviteJouerPartie.java](#).

9.2.3.7 cibleDarts

```
Peripherique com.lasalle84.mobile_darts.ActiviteJouerPartie.cibleDarts = null [private]
```

Le périphérique Bluetooth de la cible.

Attributs

Définition à la ligne 45 du fichier [ActiviteJouerPartie.java](#).

9.2.3.8 ecranDarts

```
Peripherique com.lasalle84.mobile_darts.ActiviteJouerPartie.ecranDarts = null [private]
```

Le périphérique Bluetooth de l'écran.

Définition à la ligne 46 du fichier [ActiviteJouerPartie.java](#).

9.2.3.9 handler

```
Handler com.lasalle84.mobile_darts.ActiviteJouerPartie.handler = null [private]
```

Le Handler qui permet de communiquer avec le thread.

Définition à la ligne 47 du fichier [ActiviteJouerPartie.java](#).

9.2.3.10 listJoueur

```
ListView com.lasalle84.mobile_darts.ActiviteJouerPartie.listJoueur = null [private]
```

Listview des Joueurs.

Définition à la ligne 53 du fichier [ActiviteJouerPartie.java](#).

9.2.3.11 maPartie

```
Partie com.lasalle84.mobile_darts.ActiviteJouerPartie.maPartie = null [private]
```

La partie jouée.

Définition à la ligne 50 du fichier [ActiviteJouerPartie.java](#).

9.2.3.12 mesJoueurs

```
ArrayList<Joueur> com.lasalle84.mobile_darts.ActiviteJouerPartie.mesJoueurs = null [private]
```

liste des Joueurs

Définition à la ligne 48 du fichier [ActiviteJouerPartie.java](#).

9.2.3.13 TAG

```
final String com.lasalle84.mobile_darts.ActiviteJouerPartie.TAG = "_ActiviteJouerPartie_" [static], [private]
```

Constantes

Définition à la ligne 40 du fichier [ActiviteJouerPartie.java](#).

9.2.3.14 typePartie

```
TypeJeu com.lasalle84.mobile_darts.ActiviteJouerPartie.typePartie = null [private]
```

Définition à la ligne 51 du fichier [ActiviteJouerPartie.java](#).

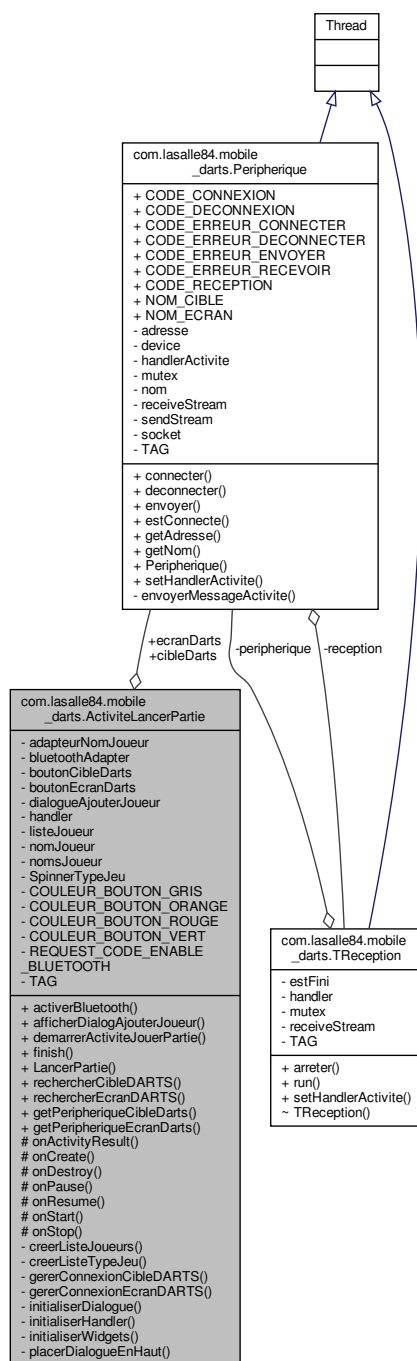
La documentation de cette classe a été générée à partir du fichier suivant :

— [ActiviteJouerPartie.java](#)

9.3 Référence de la classe com.lasalle84.mobile_darts.ActiviteLancerPartie

L'activité [ActiviteLancerPartie](#).

Graphe de collaboration de com.lasalle84.mobile_darts.ActiviteLancerPartie :



Fonctions membres publiques

- void **activerBluetooth** ()
Gère l'activation du Bluetooth.
- void **afficherDialogAjouterJoueur** (View view)
Démarre l'activité pour créer les joueurs.
- void **demarrerActiviteJouerPartie** (View view)
Démarre l'activité pour jouer la partie.
- void **finish** ()
Termine l'activité
- void **LancerPartie** ()

- boolean [rechercherCibleDARTS](#) ()
Démarre la recherche du module Bluetooth cible.
- boolean [rechercherEcranDARTS](#) ()
Démarre la recherche du module Bluetooth écran.

Fonctions membres publiques statiques

- static [Peripherique](#) [getPeripheriqueCibleDarts](#) ()
- static [Peripherique](#) [getPeripheriqueEcranDarts](#) ()

Attributs publics statiques

- static [Peripherique](#) [cibleDarts](#) = null
Le périphérique Bluetooth de la cible.
- static [Peripherique](#) [ecranDarts](#) = null
Le périphérique Bluetooth de l'écran.

Fonctions membres protégées

- void [onActivityResult](#) (int requestCode, int resultCode, Intent data)
Traite le retour des activités.
- void [onCreate](#) (Bundle savedInstanceState)
Méthode appelée à la création de l'activité
- void [onDestroy](#) ()
Méthode appelée à la destruction de l'application (après [onStop\(\)](#) et détruite par le système Android)
- void [onPause](#) ()
Méthode appelée après qu'une boîte de dialogue s'est affichée (on reprend sur un [onResume\(\)](#)) ou avant [onStop\(\)](#) (activité plus visible)
- void [onResume](#) ()
Méthode appelée après [onStart\(\)](#) ou après [onPause\(\)](#)
- void [onStart](#) ()
Méthode appelée au démarrage après le [onCreate\(\)](#) ou un restart après un [onStop\(\)](#)
- void [onStop](#) ()
Méthode appelée lorsque l'activité n'est plus visible.

Fonctions membres privées

- void [creerListeJoueurs](#) ()
Crée la liste des joueurs.
- void [creerListeTypeJeu](#) ()
Crée la liste des types de partie.
- void [gererConnexionCibleDARTS](#) ()
Gère la connexion au module Bluetooth cible.
- void [gererConnexionEcranDARTS](#) ()
Gère la connexion au module Bluetooth écran.
- void [initialiserDialogue](#) ()
- void [initialiserHandler](#) ()
Initialisation du Handler.
- void [initialiserWidgets](#) ()
Initialise les widgets.
- void [placerDialogueEnHaut](#) (Window window)
Place la boîte de dialogue en haut pour laisser l'affichage central visible.

Attributs privés

- ArrayAdapter< String > [adapteurNomJoueur](#)
- BluetoothAdapter [bluetoothAdapter](#) = null
Adaptateur Bluetooth de la tablette.
- ImageButton [boutonCibleDarts](#)
- ImageButton [boutonEcranDarts](#)
- AlertDialog [dialogueAjouterJoueur](#)
- Handler [handler](#) = null
Le Handler qui permet de communiquer avec le thread.
- ListView [listeJoueur](#)
- EditText [nomJoueur](#)
- List< String > [nomsJoueur](#)
Conteneur de noms des Joueurs.
- Spinner [SpinnerTypeJeu](#)

Attributs privés statiques

```

— static final int COULEUR_BOUTON_GRISE = Color.parseColor("#c0c5c6")
— static final int COULEUR_BOUTON_ORANGE = Color.parseColor("#f7bb31")
— static final int COULEUR_BOUTON_ROUGE = Color.parseColor("#ee5e5e")
— static final int COULEUR_BOUTON_VERT = Color.parseColor("#5eed7b")
— static final int REQUEST_CODE_ENABLE_BLUETOOTH = 0
    Code pour activer le Bluetooth.
— static final String TAG = "_ActiviteLancerPartie_"
    TAG pour les logs.

```

9.3.1 Description détaillée

L'activité [ActiviteLancerPartie](#).

Définition à la ligne 45 du fichier [ActiviteLancerPartie.java](#).

9.3.2 Documentation des fonctions membres

9.3.2.1 activerBluetooth()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.activerBluetooth ( )
```

Gère l'activation du Bluetooth.

Définition à la ligne 334 du fichier [ActiviteLancerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.onCreate\(\)](#).

```

00335     {
00336         Log.d(TAG, "activerBluetooth()");
00337
00338         bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
00339         if (bluetoothAdapter == null)
00340         {
00341             Toast.makeText(getApplicationContext(), "Bluetooth non activé !", Toast.LENGTH_SHORT).show();
00342             boutonCibleDarts.setBackgroundColor(
00343                 COULEUR_BOUTON_GRISE);
00344             boutonEcranDarts.setBackgroundColor(
00345                 COULEUR_BOUTON_GRISE);
00346         }
00347         else
00348         {
00349             if (!bluetoothAdapter.isEnabled())
00350             {
00351                 Toast.makeText(getApplicationContext(), "Bluetooth non activé !", Toast.LENGTH_SHORT).show(
00352                     );
00353                 Intent activeBlueTooth = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
00354                 startActivityForResult(activeBlueTooth,
00355                     REQUEST_CODE_ENABLE_BLUETOOTH);
00356             }
00357             else
00358             {
00359                 Toast.makeText(getApplicationContext(), "Bluetooth activé", Toast.LENGTH_SHORT).show();
00360                 boutonCibleDarts.setBackgroundColor(
00361                     COULEUR_BOUTON_ROUGE);
00362                 boutonEcranDarts.setBackgroundColor(
00363                     COULEUR_BOUTON_ROUGE);
00364             }
00365         }
00366     }

```

9.3.2.2 afficherDialogAjouterJoueur()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.afficherDialogAjouterJoueur (
    View view )
```

Démarre l'activité pour créer les joueurs.

Définition à la ligne 365 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteLancerPartie.placerDialogueEnHaut\(\)](#).

```
00366     {
00367         Log.d(TAG, "afficherDialogAjouterJoueur()");
00368         nomJoueur.setText("");
00369         dialogueAjouterJoueur.show();
00370         placerDialogueEnHaut(dialogueAjouterJoueur.getWindow());
00371         getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_ADJUST_NOTHING);
00372     }
```

9.3.2.3 creerListeJoueurs()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.creerListeJoueurs ( ) [private]
```

Crée la liste des joueurs.

Définition à la ligne 251 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteLancerPartie.nomsJoueur](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.onCreate\(\)](#).

```
00252     {
00253         Log.d(TAG, "creerListeJoueurs()");
00254
00255         listeJoueur = (ListView) findViewById(R.id.ListeJoueur);
00256
00257         nomsJoueur = new ArrayList<String>();
00258
00259         adapteurNomJoueur = new ArrayAdapter<String>(this, R.layout.item_joueur, R.id.
    texte_item, nomsJoueur);
00260
00261         listeJoueur.setAdapter(adapteurNomJoueur);
00262     }
```

9.3.2.4 creerListeTypeJeu()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.creerListeTypeJeu ( ) [private]
```

Crée la liste des types de partie.

Définition à la ligne 233 du fichier [ActiviteLancerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserWidgets\(\)](#).

```
00234     {
00235         Log.d(TAG, "creerListeTypeJeu()");
00236
00237         List<String> listeTypeJeu = new ArrayList<String>();
00238         listeTypeJeu.add("301");
00239         listeTypeJeu.add("301 Double Out");
00240         listeTypeJeu.add("501");
00241         listeTypeJeu.add("501 Double Out");
00242
00243         ArrayAdapter<String> adapteurTypeJeu = new ArrayAdapter<String>(this, R.layout.item_type_jeu, R.id.
    texte_type_jeu, listeTypeJeu);
00244         adapteurTypeJeu.setDropDownViewResource(R.layout.spinner_dropdown_item);
00245         SpinnerTypeJeu.setAdapter(adapteurTypeJeu);
00246     }
```

9.3.2.5 demarrerActiviteJouerPartie()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.demarrerActiviteJouerPartie (
    View view )
```

Démarre l'activité pour jouer la partie.

Définition à la ligne 388 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.estConnecte\(\)](#), et [com.lasalle84.mobile_darts.ActiviteLancerPartie.LancerPartie\(\)](#).

```
00389      {
00390          Log.d(TAG, "demarrerActiviteJouerPartie()");
00391          if(cibleDarts.estConnecte())
00392          {
00393              LancerPartie();
00394          }
00395          else
00396          {
00397              Log.d(TAG, "Cible non connectée");
00398              return;
00399          }
00400      }
```

9.3.2.6 finish()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.finish ( )
```

Termine l'activité

Définition à la ligne 541 du fichier [ActiviteLancerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.LancerPartie\(\)](#).

```
00542      {
00543          Log.d(TAG, "[finish]");
00544          Intent intent = new Intent();
00545          setResult(RESULT_OK, intent);
00546          super.finish();
00547      }
```

9.3.2.7 gererConnexionCibleDARTS()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.gererConnexionCibleDARTS ( ) [private]
```

Gère la connexion au module Bluetooth cible.

Définition à la ligne 452 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.connecter\(\)](#), [com.lasalle84.mobile_darts.Peripherique.deconnecter\(\)](#), et [com.lasalle84.mobile_darts.Peripherique.estConnecte\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserWidgets\(\)](#).

```
00453      {
00454          if(cibleDarts != null)
00455          {
00456              Log.d(TAG, "gererConnexionCibleDARTS()");
00457              if(!cibleDarts.estConnecte())
00458                  cibleDarts.connecter();
00459              else
00460                  cibleDarts.deconnecter();
00461          }
00462      }
```

9.3.2.8 `gererConnexionEcranDARTS()`

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.gererConnexionEcranDARTS ( ) [private]
```

Gère la connexion au module Bluetooth écran.

Définition à la ligne 467 du fichier `ActiviteLancerPartie.java`.

Références `com.lasalle84.mobile_darts.Peripherique.connecter()`, `com.lasalle84.mobile_darts.Peripherique.deconnecter()`, et `com.lasalle84.mobile_darts.Peripherique.estConnecte()`.

Référencé par `com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserWidgets()`.

```
00468     {
00469         if(ecranDarts != null)
00470         {
00471             Log.d(TAG, "gererConnexionEcranDARTS ( )");
00472             if(!ecranDarts.estConnecte())
00473                 ecranDarts.connecter();
00474             else
00475                 ecranDarts.deconnecter();
00476         }
00477     }
```

9.3.2.9 `getPeripheriqueCibleDarts()`

```
static Peripherique com.lasalle84.mobile_darts.ActiviteLancerPartie.getPeripheriqueCibleDarts ( ) [static]
```

Définition à la ligne 479 du fichier `ActiviteLancerPartie.java`.

Références `com.lasalle84.mobile_darts.ActiviteLancerPartie.cibleDarts`.

Référencé par `com.lasalle84.mobile_darts.ActiviteJouerPartie.recupererPeripheriques()`.

```
00480     {
00481         return cibleDarts;
00482     }
```

9.3.2.10 `getPeripheriqueEcranDarts()`

```
static Peripherique com.lasalle84.mobile_darts.ActiviteLancerPartie.getPeripheriqueEcranDarts ( ) [static]
```

Définition à la ligne 484 du fichier `ActiviteLancerPartie.java`.

Références `com.lasalle84.mobile_darts.ActiviteLancerPartie.ecranDarts`.

Référencé par `com.lasalle84.mobile_darts.ActiviteJouerPartie.recupererPeripheriques()`.

```
00485     {
00486         return ecranDarts;
00487     }
```

9.3.2.11 initialiserDialogue()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserDialogue ( ) [private]
```

Définition à la ligne 264 du fichier [ActiviteLancerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.onCreate\(\)](#).

```
00265     {
00266         AlertDialog.Builder builder = new AlertDialog.Builder(this);
00267
00268         builder.setTitle(R.string.title);
00269         View view = getLayoutInflater().inflate(R.layout.dialog_nouveau_joueur,null);
00270         nomJoueur = (EditText) view.findViewById(R.id.nomJoueur);
00271
00272         builder.setView(view);
00273
00274         builder.setPositiveButton(R.string.ok, new DialogInterface.OnClickListener() {
00275             public void onClick(DialogInterface dialog, int id) {
00276                 Log.d(TAG, "onClick() nom joueur : " + nomJoueur.getText().toString());
00277                 nomsJoueur.add(nomJoueur.getText().toString());
00278                 adapteurNomJoueur.notifyDataSetChanged();
00279             }
00280         });
00281         builder.setNegativeButton(R.string.annuler, new DialogInterface.OnClickListener() {
00282             public void onClick(DialogInterface dialog, int id) {
00283                 // User cancelled the dialog
00284             }
00285         });
00286
00287         dialogueAjouterJoueur = builder.create();
00288     }
```

9.3.2.12 initialiserHandler()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler ( ) [private]
```

Initialisation du Handler.

A faire Traiter les messages

Définition à la ligne 155 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.CODE_CONNEXION](#), [com.lasalle84.mobile_darts.Peripherique.CODE_DEC↵](#)
[ONNEXION](#), [com.lasalle84.mobile_darts.Peripherique.CODE_ERREUR_CONNECTER](#), [com.lasalle84.mobile_darts.Peripherique.↵](#)
[CODE_ERREUR_DECONNECTER](#), [com.lasalle84.mobile_darts.Peripherique.CODE_ERREUR_ENVOYER](#), [com.lasalle84.↵](#)
[mobile_darts.Peripherique.NOM_CIBLE](#), [com.lasalle84.mobile_darts.Peripherique.NOM_ECRAN](#), et [com.lasalle84.mobile_darts.↵](#)
[Peripherique.setHandlerActivite\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.onCreate\(\)](#).

```
00156     {
00157         if(handler != null)
00158         {
00159             if(cibleDarts != null)
00160             {
00161                 Log.d(TAG,"initialiserHandler() cibleDarts " + handler);
00162                 cibleDarts.setHandlerActivite(
00163                     handler);
00164             }
00165             if(ecranDarts != null)
00166             {
00167                 Log.d(TAG,"initialiserHandler() ecranDarts " + handler);
00168                 ecranDarts.setHandlerActivite(
00169                     handler);
00170             }
00171             return;
00172         }
00173         Log.d(TAG,"initialiserHandler() ");
```

```

00172
00173     this.handler = new Handler(this.getMainLooper())
00174     {
00175         @Override
00176         public void handleMessage(Message msg)
00177         {
00178             super.handleMessage(msg);
00179             Bundle b = msg.getData();
00180             Log.d(TAG, "ActiviteLancerPartie:handleMessage() Nom : " + b.getString("nom") + " -
Adresse : " + b.getString("adresse") + " -> " + msg.what);
00181
00182             switch(msg.what)
00183             {
00184                 case Peripherique.CODE_CONNEXION:
00185                     if(b.getString("nom").equals(Peripherique.NOM_CIBLE))
00186                     {
00187                         boutonCibleDarts.setBackgroundColor(
00188                             COULEUR_BOUTON_VERT);
00189                     }
00190                     else if(b.getString("nom").equals(Peripherique.NOM_ECRAN))
00191                     {
00192                         boutonEcranDarts.setBackgroundColor(
00193                             COULEUR_BOUTON_VERT);
00194                     }
00195                     break;
00196                 case Peripherique.CODE_DECONNEXION:
00197                     if(b.getString("nom").equals(Peripherique.NOM_CIBLE))
00198                     {
00199                         boutonCibleDarts.setBackgroundColor(
00200                             COULEUR_BOUTON_ROUGE);
00201                     }
00202                     else if(b.getString("nom").equals(Peripherique.NOM_ECRAN))
00203                     {
00204                         boutonEcranDarts.setBackgroundColor(
00205                             COULEUR_BOUTON_ROUGE);
00206                     }
00207                     break;
00208                 case Peripherique.CODE_ERREUR_ENVOYER:
00209                     break;
00210                 case Peripherique.CODE_ERREUR_CONNECTER:
00211                     break;
00212                 case Peripherique.CODE_ERREUR_DECONNECTER:
00213                     break;
00214             }
00215         }
00216     };
00217     Log.d(TAG, "initialiserHandler() ecranDarts " + handler);
00218     if(cibleDarts != null)
00219     {
00220         Log.d(TAG, "initialiserHandler() cibleDarts " + handler);
00221         cibleDarts.setHandlerActivite(handler);
00222     }
00223     if(ecranDarts != null)
00224     {
00225         Log.d(TAG, "initialiserHandler() ecranDarts " + handler);
00226         ecranDarts.setHandlerActivite(handler);
00227     }
00228 }

```

9.3.2.13 initialiserWidgets()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserWidgets ( ) [private]
```

Initialise les widgets.

Définition à la ligne 293 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteLancerPartie.creerListeTypeJeu\(\)](#), [com.lasalle84.mobile_darts.ActiviteLancerPartie.gergerConnexionCibleDARTS\(\)](#), [com.lasalle84.mobile_darts.ActiviteLancerPartie.gergerConnexionEcranDARTS\(\)](#), [com.lasalle84.mobile_darts.ActiviteLancerPartie.rechercherCibleDARTS\(\)](#), et [com.lasalle84.mobile_darts.ActiviteLancerPartie.rechercherEcranDARTS\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.onCreate\(\)](#).

```

00294     {
00295         Log.d(TAG, "initialiserWidgets()");
00296
00297         boutonCibleDarts = (ImageButton) findViewById(R.id.boutonCibleDarts);
00298         boutonCibleDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00299         boutonCibleDarts.setOnClickListener(view -> {
00300             Log.d(TAG, "onClick() boutonCibleDarts");
00301             if(rechercherCibleDARTS())
00302             {
00303                 boutonCibleDarts.setBackgroundColor(
00304                     COULEUR_BOUTON_ORANGE);
00305                 gererConnexionCibleDARTS();
00306             }
00307             else
00308             {
00309                 boutonCibleDarts.setBackgroundColor(
00310                     COULEUR_BOUTON_ROUGE);
00311             }
00312         });
00313
00314         boutonEcranDarts = (ImageButton) findViewById(R.id.boutonEcranDarts);
00315         boutonEcranDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00316         boutonEcranDarts.setOnClickListener(v -> {
00317             Log.d(TAG, "onClick() boutonEcranDarts");
00318             if(rechercherEcranDARTS())
00319             {
00320                 boutonEcranDarts.setBackgroundColor(
00321                     COULEUR_BOUTON_ORANGE);
00322                 gererConnexionEcranDARTS();
00323             }
00324             else
00325             {
00326                 boutonEcranDarts.setBackgroundColor(
00327                     COULEUR_BOUTON_ROUGE);
00328             }
00329         });
00330
00331         SpinnerTypeJeu = (Spinner) findViewById(R.id.spinnerTypeJeu);
00332         creerListeTypeJeu();
00333     }

```

9.3.2.14 LancerPartie()

void com.lasalle84.mobile_darts.ActiviteLancerPartie.LancerPartie ()

Définition à la ligne 509 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteLancerPartie.finish\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.demarrerActiviteJouerPartie\(\)](#).

```

00510     {
00511         Log.d(TAG, "LancerPartie()");
00512         Vector<Joueur> mesJoueurs = new Vector<>();
00513         if(nomsJoueur.size() == 0)
00514         {
00515             for (int i = 1; i <= 1; i++)
00516             {
00517                 mesJoueurs.add(new Joueur("Joueur" + i));
00518                 Log.d(TAG, "Joueur" + i + " = " + "Joueur" + i);
00519             }
00520         }
00521         else
00522         {
00523             for (int i = 0; i < nomsJoueur.size(); i++)
00524             {
00525                 mesJoueurs.add(new Joueur(nomsJoueur.get(i)));
00526                 Log.d(TAG, "Joueur" + i + " = " + nomsJoueur.get(i));
00527             }
00528         }
00529
00530         Intent intent = new Intent(ActiviteLancerPartie.this, ActiviteJouerPartie.class);
00531         intent.putExtra("TypeMode", SpinnerTypeJeu.getSelectedItemPosition());
00532         Log.d(TAG, "TypeMode = " + SpinnerTypeJeu.getSelectedItemPosition());
00533         intent.putExtra("LesJoueurs", mesJoueurs);
00534         startActivityForResult(intent, 12);
00535         finish();
00536     }

```

9.3.2.15 onActivityResult()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.onActivityResult (
    int requestCode,
    int resultCode,
    Intent data ) [protected]
```

Traite le retour des activités.

Définition à la ligne 492 du fichier [ActiviteLancerPartie.java](#).

```
00493     {
00494         super.onActivityResult(requestCode, resultCode, data);
00495         Log.d(TAG, "onActivityResult() Fin ActiviteCreerJoueur");
00496         if(resultCode == Activity.RESULT_OK)
00497         {
00498             String nom = data.getStringExtra("retour");
00499             Log.d(TAG, "Nom du Joueur saisi" + nom);
00500             nomsJoueur.add(nom);
00501         }
00502     }
```

9.3.2.16 onCreate()

```
com.lasalle84.mobile_darts.ActiviteLancerPartie.onCreate (
    Bundle savedInstanceState ) [protected]
```

Méthode appelée à la création de l'activité

Paramètres

<i>savedInstanceState</i>	
---------------------------	--

Définition à la ligne 85 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.ActiviteLancerPartie.activerBluetooth\(\)](#), [com.lasalle84.mobile_darts.ActiviteLancerPartie.creerListeJoueurs\(\)](#), [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserDialogue\(\)](#), [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler\(\)](#), et [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserWidgets\(\)](#).

```
00086     {
00087         super.onCreate(savedInstanceState);
00088         setContentView(R.layout.activity_activite_lancer_partie);
00089         Log.d(TAG, "[onCreate]");
00090
00091         initialiserHandler();
00092
00093         creerListeJoueurs();
00094
00095         initialiserWidgets();
00096
00097         initialiserDialogue();
00098
00099         activerBluetooth();
00100     }
```


9.3.2.17 onDestroy()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.onDestroy ( ) [protected]
```

Méthode appelée à la destruction de l'application (après [onStop\(\)](#) et détruite par le système Android)

Définition à la ligne 146 du fichier [ActiviteLancerPartie.java](#).

```
00147    {  
00148        super.onDestroy();  
00149        Log.d(TAG, "[onDestroy]");  
00150    }
```

9.3.2.18 onPause()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.onPause ( ) [protected]
```

Méthode appelée après qu'une boîte de dialogue s'est affichée (on reprend sur un [onResume\(\)](#)) ou avant [onStop\(\)](#) (activité plus visible)

Définition à la ligne 126 du fichier [ActiviteLancerPartie.java](#).

```
00127    {  
00128        super.onPause();  
00129        Log.d(TAG, "[onPause]");  
00130    }
```

9.3.2.19 onResume()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.onResume ( ) [protected]
```

Méthode appelée après [onStart\(\)](#) ou après [onPause\(\)](#)

Définition à la ligne 116 du fichier [ActiviteLancerPartie.java](#).

```
00117    {  
00118        super.onResume();  
00119        Log.d(TAG, "[onResume]");  
00120    }
```

9.3.2.20 onStart()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.onStart ( ) [protected]
```

Méthode appelée au démarrage après le [onCreate\(\)](#) ou un restart après un [onStop\(\)](#)

Définition à la ligne 106 du fichier [ActiviteLancerPartie.java](#).

```
00107    {  
00108        super.onStart();  
00109        Log.d(TAG, "[onStart]");  
00110    }
```

9.3.2.21 onStop()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.onStop ( ) [protected]
```

Méthode appelée lorsque l'activité n'est plus visible.

Définition à la ligne 136 du fichier [ActiviteLancerPartie.java](#).

```
00137     {
00138         super.onStop();
00139         Log.d(TAG, "[onStop]");
00140     }
```

9.3.2.22 placerDialogueEnHaut()

```
void com.lasalle84.mobile_darts.ActiviteLancerPartie.placerDialogueEnHaut (
    Window window ) [private]
```

Place la boîte de dialogue en haut pour laisser l'affichage central visible.

Définition à la ligne 377 du fichier [ActiviteLancerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.afficherDialogAjouterJoueur\(\)](#).

```
00378     {
00379         WindowManager.LayoutParams wlp = window.getAttributes();
00380         wlp.gravity = Gravity.BOTTOM;
00381         wlp.flags &= ~WindowManager.LayoutParams.FLAG_DIM_BEHIND;
00382         window.setAttributes(wlp);
00383     }
```

9.3.2.23 rechercherCibleDARTS()

```
boolean com.lasalle84.mobile_darts.ActiviteLancerPartie.rechercherCibleDARTS ( )
```

Démarre la recherche du module Bluetooth cible.

Définition à la ligne 405 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.NOM_CIBLE](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserWidgets\(\)](#).

```
00406     {
00407         if(cibleDarts != null)
00408             return true;
00409
00410         Log.d(TAG, "rechercherCibleDARTS()");
00411         Set<BluetoothDevice> devices;
00412         devices = bluetoothAdapter.getBondedDevices();
00413         for (BluetoothDevice blueDevice : devices)
00414         {
00415             if(blueDevice.getName().equals(Peripherique.NOM_CIBLE))
00416             {
00417                 cibleDarts = new Peripherique(blueDevice, handler);
00418                 return true;
00419             }
00420         }
00421
00422         return false;
00423     }
```

9.3.2.24 rechercherEcranDARTS()

`boolean com.lasalle84.mobile_darts.ActiviteLancerPartie.rechercherEcranDARTS ()`

Démarre la recherche du module Bluetooth écran.

Définition à la ligne 428 du fichier [ActiviteLancerPartie.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.NOM_ECRAN](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserWidgets\(\)](#).

```

00429     {
00430         if(ecranDarts != null)
00431             return true;
00432
00433         Log.d(TAG, "rechercherEcranDARTS() ");
00434         Set<BluetoothDevice> devices;
00435         devices = bluetoothAdapter.getBondedDevices();
00436         for (BluetoothDevice blueDevice : devices)
00437         {
00438             Log.d(TAG, "Device = " + blueDevice.getName());
00439             if(blueDevice.getName().equals(Peripherique.NOM_ECRAN))
00440             {
00441                 ecranDarts = new Peripherique(blueDevice, handler);
00442                 return true;
00443             }
00444         }
00445
00446         return false;
00447     }

```

9.3.3 Documentation des données membres

9.3.3.1 adaptateurNomJoueur

`ArrayAdapter<String> com.lasalle84.mobile_darts.ActiviteLancerPartie.adaptateurNomJoueur [private]`

Définition à la ligne 66 du fichier [ActiviteLancerPartie.java](#).

9.3.3.2 bluetoothAdapter

`BluetoothAdapter com.lasalle84.mobile_darts.ActiviteLancerPartie.bluetoothAdapter = null [private]`

Adaptateur Bluetooth de la tablette.

Attributs

Définition à la ligne 61 du fichier [ActiviteLancerPartie.java](#).

9.3.3.3 boutonCibleDarts

`ImageButton com.lasalle84.mobile_darts.ActiviteLancerPartie.boutonCibleDarts [private]`

Définition à la ligne 73 du fichier [ActiviteLancerPartie.java](#).

9.3.3.4 boutonEcranDarts

```
ImageButton com.lasalle84.mobile_darts.ActiviteLancerPartie.boutonEcranDarts [private]
```

Définition à la ligne 74 du fichier [ActiviteLancerPartie.java](#).

9.3.3.5 cibleDarts

```
Peripherique com.lasalle84.mobile_darts.ActiviteLancerPartie.cibleDarts = null [static]
```

Le périphérique Bluetooth de la cible.

Définition à la ligne 62 du fichier [ActiviteLancerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.getPeripheriqueCibleDarts\(\)](#).

9.3.3.6 COULEUR_BOUTON_GRIS

```
final int com.lasalle84.mobile_darts.ActiviteLancerPartie.COULEUR_BOUTON_GRIS = Color.parseColor("#c0c5c6")  
[static], [private]
```

Définition à la ligne 55 du fichier [ActiviteLancerPartie.java](#).

9.3.3.7 COULEUR_BOUTON_ORANGE

```
final int com.lasalle84.mobile_darts.ActiviteLancerPartie.COULEUR_BOUTON_ORANGE = Color.parseColor("#f7bb31")  
[static], [private]
```

Définition à la ligne 54 du fichier [ActiviteLancerPartie.java](#).

9.3.3.8 COULEUR_BOUTON_ROUGE

```
final int com.lasalle84.mobile_darts.ActiviteLancerPartie.COULEUR_BOUTON_ROUGE = Color.parseColor("#ee5e5e")  
[static], [private]
```

Définition à la ligne 53 du fichier [ActiviteLancerPartie.java](#).

9.3.3.9 COULEUR_BOUTON_VERT

```
final int com.lasalle84.mobile_darts.ActiviteLancerPartie.COULEUR_BOUTON_VERT = Color.parseColor("#5eed7b")  
[static], [private]
```

Définition à la ligne 56 du fichier [ActiviteLancerPartie.java](#).

9.3.3.10 dialogueAjouterJoueur

```
AlertDialog com.lasalle84.mobile_darts.ActiviteLancerPartie.dialogueAjouterJoueur [private]
```

Définition à la ligne 76 du fichier [ActiviteLancerPartie.java](#).

9.3.3.11 ecranDarts

```
Peripherique com.lasalle84.mobile_darts.ActiviteLancerPartie.ecranDarts = null [static]
```

Le périphérique Bluetooth de l'écran.

Définition à la ligne 63 du fichier [ActiviteLancerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.getPeripheriqueEcranDarts\(\)](#).

9.3.3.12 handler

```
Handler com.lasalle84.mobile_darts.ActiviteLancerPartie.handler = null [private]
```

Le Handler qui permet de communiquer avec le thread.

Définition à la ligne 64 du fichier [ActiviteLancerPartie.java](#).

9.3.3.13 listeJoueur

```
ListView com.lasalle84.mobile_darts.ActiviteLancerPartie.listeJoueur [private]
```

Définition à la ligne 72 du fichier [ActiviteLancerPartie.java](#).

9.3.3.14 nomJoueur

```
EditText com.lasalle84.mobile_darts.ActiviteLancerPartie.nomJoueur [private]
```

Définition à la ligne 75 du fichier [ActiviteLancerPartie.java](#).

9.3.3.15 nomsJoueur

```
List<String> com.lasalle84.mobile_darts.ActiviteLancerPartie.nomsJoueur [private]
```

Conteneur de noms des Joueurs.

Définition à la ligne 65 du fichier [ActiviteLancerPartie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.creerListeJoueurs\(\)](#).

9.3.3.16 REQUEST_CODE_ENABLE_BLUETOOTH

```
final int com.lasalle84.mobile_darts.ActiviteLancerPartie.REQUEST_CODE_ENABLE_BLUETOOTH = 0 [static], [private]
```

Code pour activer le Bluetooth.

Définition à la ligne 51 du fichier [ActiviteLancerPartie.java](#).

9.3.3.17 SpinnerTypeJeu

```
Spinner com.lasalle84.mobile_darts.ActiviteLancerPartie.SpinnerTypeJeu [private]
```

Widgets IHM

Définition à la ligne 71 du fichier [ActiviteLancerPartie.java](#).

9.3.3.18 TAG

```
final String com.lasalle84.mobile_darts.ActiviteLancerPartie.TAG = "_ActiviteLancerPartie_" [static], [private]
```

TAG pour les logs.

Constantes

Définition à la ligne 50 du fichier [ActiviteLancerPartie.java](#).

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

— [ActiviteLancerPartie.java](#)

9.4 Référence de la classe com.lasalle84.mobile_darts.IhmMobileDarts

L'activité principale de l'application DARTS.

Graphe de collaboration de com.lasalle84.mobile_darts.IhmMobileDarts :

com.lasalle84.mobile_darts.IhmMobileDarts
+ EXTRA_MESSAGE - versionDarts - ACTIVITE_LANCER_PARTIE - TAG
+ demarrerActiviteAfficher Historique() + demarrerActiviteLancerPartie() # onActivityResult() # onCreate() # onDestroy() # onPause() # onResume() # onStart() # onStop()

Fonctions membres publiques

- void `demarrerActiviteAfficherHistorique` (View view)
- void `demarrerActiviteLancerPartie` (View view)
Démarrer l'activité `ActiviteLancerPartie`.

Attributs publics statiques

- static final String `EXTRA_MESSAGE` = "com.lasalle84.mobile_darts.MESSAGE"

Fonctions membres protégées

- void `onActivityResult` (int requestCode, int resultCode, Intent data)
Traite le retour des activités.
- void `onCreate` (Bundle savedInstanceState)
Méthode appelée à la création de l'activité
- void `onDestroy` ()
Méthode appelée à la destruction de l'application (après `onStop()`) et détruite par le système Android
- void `onPause` ()
Méthode appelée après qu'une boîte de dialogue s'est affichée (on reprend sur un `onResume()`) ou avant `onStop()` (activité plus visible)
- void `onResume` ()
Méthode appelée après `onStart()` ou après `onPause()`
- void `onStart` ()
Méthode appelée au démarrage après le `onCreate()` ou un restart après un `onStop()`
- void `onStop` ()
Méthode appelée lorsque l'activité n'est plus visible.

Attributs privés

- TextView `versionDarts`

Attributs privés statiques

- static final int `ACTIVITE_LANCER_PARTIE` = 1
Code pour identifier l'activité `ActiviteLancerPartie`.
- static final String `TAG` = "_IhmMobileDarts_"
TAG pour les logs.

9.4.1 Description détaillée

L'activité principale de l'application DARTS.

Définition à la ligne 28 du fichier `IhmMobileDarts.java`.

9.4.2 Documentation des fonctions membres

9.4.2.1 `demarrerActiviteAfficherHistorique()`

```
void com.lasalle84.mobile_darts.IhmMobileDarts.demarrerActiviteAfficherHistorique (
    View view )
```

Définition à la ligne 139 du fichier `IhmMobileDarts.java`.

```
00140    {
00141        Toast.makeText(getApplicationContext(), "Fonction non disponible", Toast.LENGTH_SHORT).show();
00142    }
```

9.4.2.2 demarrerActiviteLancerPartie()

```
void com.lasalle84.mobile_darts.IhmMobileDarts.demarrerActiviteLancerPartie (
    View view )
```

Démarrer l'activité [ActiviteLancerPartie](#).

Définition à la ligne 112 du fichier [IhmMobileDarts.java](#).

```
00113    {
00114        Intent intent = new Intent(this, ActiviteLancerPartie.class);
00115        startActivityForResult(intent, ACTIVITE_LANCER_PARTIE);
00116    }
```

9.4.2.3 onActivityResult()

```
void com.lasalle84.mobile_darts.IhmMobileDarts.onActivityResult (
    int requestCode,
    int resultCode,
    Intent data ) [protected]
```

Traite le retour des activités.

Définition à la ligne 121 du fichier [IhmMobileDarts.java](#).

Références [com.lasalle84.mobile_darts.IhmMobileDarts.ACTIVITE_LANCER_PARTIE](#).

```
00122    {
00123        super.onActivityResult(requestCode, resultCode, data);
00124
00125        switch (requestCode)
00126        {
00127            case ACTIVITE_LANCER_PARTIE:
00128                Log.d(TAG, "[onActivityResult] Fin ActiviteLancerPartie");
00129                if(resultCode == Activity.RESULT_OK)
00130                {
00131                    Log.d(TAG, "[onActivityResult] ActiviteLancerPartie RESULT_OK");
00132                }
00133                break;
00134            default:
00135                Log.d(TAG, "[onActivityResult] requestCode : " + requestCode);
00136        }
00137    }
```

9.4.2.4 onCreate()

```
com.lasalle84.mobile_darts.IhmMobileDarts.onCreate (
    Bundle savedInstanceState ) [protected]
```

Méthode appelée à la création de l'activité

Paramètres

<i>savedInstanceState</i>	
---------------------------	--

Définition à la ligne 49 du fichier [IhmMobileDarts.java](#).


```
00050    {
00051        super.onCreate(savedInstanceState);
00052        setContentView(R.layout.activity_main);
00053        Log.d(TAG, "[onCreate]");
00054
00055        versionDarts = (TextView) findViewById(R.id.versionDarts);
00056        versionDarts.setText("version " + BuildConfig.VERSION_NAME);
00057    }
```

9.4.2.5 onDestroy()

```
void com.lasalle84.mobile_darts.IhmMobileDarts.onDestroy ( ) [protected]
```

Méthode appelée à la destruction de l'application (après [onStop\(\)](#) et détruite par le système Android)

Définition à la ligne 103 du fichier [IhmMobileDarts.java](#).

```
00104    {
00105        super.onDestroy();
00106        Log.d(TAG, "[onDestroy]");
00107    }
```

9.4.2.6 onPause()

```
void com.lasalle84.mobile_darts.IhmMobileDarts.onPause ( ) [protected]
```

Méthode appelée après qu'une boîte de dialogue s'est affichée (on reprend sur un [onResume\(\)](#)) ou avant [onStop\(\)](#) (activité plus visible)

Définition à la ligne 83 du fichier [IhmMobileDarts.java](#).

```
00084    {
00085        super.onPause();
00086        Log.d(TAG, "[onPause]");
00087    }
```

9.4.2.7 onResume()

```
void com.lasalle84.mobile_darts.IhmMobileDarts.onResume ( ) [protected]
```

Méthode appelée après [onStart\(\)](#) ou après [onPause\(\)](#)

Définition à la ligne 73 du fichier [IhmMobileDarts.java](#).

```
00074    {
00075        super.onResume();
00076        Log.d(TAG, "[onResume]");
00077    }
```

9.4.2.8 onStart()

```
void com.lasalle84.mobile_darts.IhmMobileDarts.onStart ( ) [protected]
```

Méthode appelée au démarrage après le `onCreate()` ou un restart après un `onStop()`

Définition à la ligne 63 du fichier `IhmMobileDarts.java`.

```
00064     {  
00065         super.onStart();  
00066         Log.d(TAG, "[onStart]");  
00067     }
```

9.4.2.9 onStop()

```
void com.lasalle84.mobile_darts.IhmMobileDarts.onStop ( ) [protected]
```

Méthode appelée lorsque l'activité n'est plus visible.

Définition à la ligne 93 du fichier `IhmMobileDarts.java`.

```
00094     {  
00095         super.onStop();  
00096         Log.d(TAG, "[onStop]");  
00097     }
```

9.4.3 Documentation des données membres

9.4.3.1 ACTIVITE_LANCER_PARTIE

```
final int com.lasalle84.mobile_darts.IhmMobileDarts.ACTIVITE_LANCER_PARTIE = 1 [static], [private]
```

Code pour identifier l'activité `ActiviteLancerPartie`.

Définition à la ligne 35 du fichier `IhmMobileDarts.java`.

Référencé par `com.lasalle84.mobile_darts.IhmMobileDarts.onActivityResult()`.

9.4.3.2 EXTRA_MESSAGE

```
final String com.lasalle84.mobile_darts.IhmMobileDarts.EXTRA_MESSAGE = "com.lasalle84.mobile_darts.MESSAGE"  
[static]
```

Définition à la ligne 34 du fichier `IhmMobileDarts.java`.

9.4.3.3 TAG

```
final String com.lasalle84.mobile_darts.IhmMobileDarts.TAG = "_IhmMobileDarts_" [static], [private]
```

TAG pour les logs.

Constantes

Définition à la ligne 33 du fichier [IhmMobileDarts.java](#).

9.4.3.4 versionDarts

```
TextView com.lasalle84.mobile_darts.IhmMobileDarts.versionDarts [private]
```

Widgets IHM

Définition à la ligne 40 du fichier [IhmMobileDarts.java](#).

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

— [IhmMobileDarts.java](#)

9.5 Référence de la classe com.lasalle84.mobile_darts.Joueur

La classe qui contient toute la gestion des joueurs.

Graphe de collaboration de com.lasalle84.mobile_darts.Joueur :

com.lasalle84.mobile_darts.Joueur
- aFini - nom - score - TAG
+ getNom() + getScore() + isaFini() + Joueur() + retirerPoint() + setaFini() + setNom() + setScore()

Fonctions membres publiques

- `String` `getNom` ()
Accesseur du nom du joueur.
- `int` `getScore` ()
- `boolean` `isaFini` ()
- `Joueur` (`String` `nom`)
Constructeur de la classe `Joueur`.
- `boolean` `retirerPoint` (`int` `scoreLancer`, `Partie` `maPartie`)
Fonction permettant le retrait ou non de points d'un joueur.
- `void` `setaFini` (`boolean` `aFini`)
Accesseur set si le joueur est eliminer.
- `void` `setNom` (`String` `nom`)
Mutateur du nom du joueur.
- `void` `setScore` (`int` `score`)

Attributs privés

- `boolean` `aFini`
- `String` `nom`
- `int` `score`

Attributs privés statiques

- `static final String` `TAG` = `"_Joueur_"`

9.5.1 Description détaillée

La classe qui contient toute la gestion des joueurs.

Définition à la ligne 19 du fichier `Joueur.java`.

9.5.2 Documentation des constructeurs et destructeur

9.5.2.1 `Joueur()`

```
com.lasalle84.mobile_darts.Joueur.Joueur (
    String nom )
```

Constructeur de la classe `Joueur`.

Paramètres

<code>nom</code>	
------------------	--

Définition à la ligne 39 du fichier `Joueur.java`.

Références `com.lasalle84.mobile_darts.Joueur.nom`.

```
00040    {
00041        this.nom = nom;
00042        this.score = 0;
00043        this.aFini = false;
00044    }
```

9.5.3 Documentation des fonctions membres

9.5.3.1 getNom()

```
com.lasalle84.mobile_darts.Joueur.getNom ( )
```

Accesseur du nom du joueur.

Accesseur du score du joueur.

Renvoie

nom (Le nom du joueur)
score (le score du joueur)

Définition à la ligne 52 du fichier [Joueur.java](#).

Références [com.lasalle84.mobile_darts.Joueur.nom](#).

Référencé par [com.lasalle84.mobile_darts.Partie.actualiserScoreIHM\(\)](#), [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#), [com.lasalle84.mobile_darts.Partie.envoyerGagnantIHM\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserPartie\(\)](#), et [com.lasalle84.mobile_darts.Joueur.retirerPoint\(\)](#).

```
00053     {  
00054         return nom;  
00055     }
```

9.5.3.2 getScore()

```
int com.lasalle84.mobile_darts.Joueur.getScore ( )
```

Définition à la ligne 74 du fichier [Joueur.java](#).

Références [com.lasalle84.mobile_darts.Joueur.score](#).

Référencé par [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#), et [com.lasalle84.mobile_darts.ActiviteFinPartie.trierLesJoueurs\(\)](#).

```
00075     {  
00076         return score;  
00077     }
```

9.5.3.3 isaFini()

```
boolean com.lasalle84.mobile_darts.Joueur.isaFini ( )
```

Définition à la ligne 128 du fichier [Joueur.java](#).

Références [com.lasalle84.mobile_darts.Joueur.aFini](#).

Référencé par [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#).

```
00129     {  
00130         return aFini;  
00131     }
```

9.5.3.4 retirerPoint()

```
com.lasalle84.mobile_darts.Joueur.retirerPoint (  
    int scoreLancer,  
    Partie maPartie )
```

Fonction permettant le retrait ou non de points d'un joueur.

Paramètres

<code>scoreLancer</code>	à retirer
--------------------------	-----------

Définition à la ligne 96 du fichier `Joueur.java`.

Références `com.lasalle84.mobile_darts.TypeJeu.estDoubleOut()`, `com.lasalle84.mobile_darts.Joueur.getNom()`, et `com.lasalle84.mobile_darts.Partie.getTypeJeu()`.

Référencé par `com.lasalle84.mobile_darts.Partie.demarrerPartie()`.

```

00097     {
00098
00099         if (this.score - scoreLancer >= 0)
00100         {
00101             this.score = this.score - scoreLancer;
00102             if (this.score == 1 && maPartie.getTypeJeu().estDoubleOut())
00103             {
00104                 Log.d(TAG, this.getNom() + " est éliminé");
00105                 aFini = true;
00106             }
00107             return true;
00108         }
00109         return false;
00110     }

```

9.5.3.5 `setaFini()`

```

com.lasalle84.mobile_darts.Joueur.setaFini (
    boolean estEliminer )

```

Accesseur set si le joueur est eliminer.

Paramètres

<code>aFini</code>	si le joueur est elimine
--------------------	--------------------------

Définition à la ligne 118 du fichier `Joueur.java`.

Références `com.lasalle84.mobile_darts.Joueur.aFini`.

```

00119     {
00120         this.aFini = aFini;
00121     }

```

9.5.3.6 `setNom()`

```

com.lasalle84.mobile_darts.Joueur.setNom (
    String nom )

```

Mutateur du nom du joueur.

Paramètres

<i>nom</i>	(le nom du joueur)
<i>score</i>	(le score du joueur)

Définition à la ligne 63 du fichier [Joueur.java](#).

Références [com.lasalle84.mobile_darts.Joueur.nom](#).

```
00064    {
00065        this.nom = nom;
00066    }
```

9.5.3.7 setScore()

```
void com.lasalle84.mobile_darts.Joueur.setScore (
    int score )
```

Définition à la ligne 85 du fichier [Joueur.java](#).

Références [com.lasalle84.mobile_darts.Joueur.score](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserPartie\(\)](#).

```
00086    {
00087        this.score = score;
00088    }
```

9.5.4 Documentation des données membres

9.5.4.1 aFini

```
boolean com.lasalle84.mobile_darts.Joueur.aFini [private]
```

Définition à la ligne 31 du fichier [Joueur.java](#).

Référencé par [com.lasalle84.mobile_darts.Joueur.isaFini\(\)](#), et [com.lasalle84.mobile_darts.Joueur.setaFini\(\)](#).

9.5.4.2 nom

```
String com.lasalle84.mobile_darts.Joueur.nom [private]
```

Variables

Définition à la ligne 29 du fichier [Joueur.java](#).

Référencé par [com.lasalle84.mobile_darts.Joueur.getNom\(\)](#), [com.lasalle84.mobile_darts.Joueur.Joueur\(\)](#), et [com.lasalle84.mobile_darts.Joueur.setNom\(\)](#).

9.5.4.3 score

```
int com.lasalle84.mobile_darts.Joueur.score [private]
```

Définition à la ligne 30 du fichier [Joueur.java](#).

Référéncé par [com.lasalle84.mobile_darts.Joueur.getScore\(\)](#), et [com.lasalle84.mobile_darts.Joueur.setScore\(\)](#).

9.5.4.4 TAG

```
final String com.lasalle84.mobile_darts.Joueur.TAG = "_Joueur_" [static], [private]
```

Constantes

Définition à la ligne 24 du fichier [Joueur.java](#).

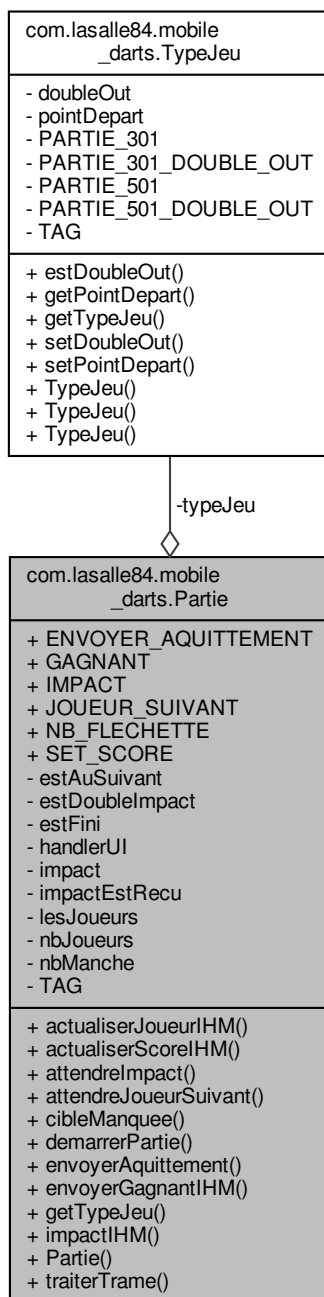
La documentation de cette classe a été générée à partir du fichier suivant :

— [Joueur.java](#)

9.6 Référence de la classe `com.lasalle84.mobile_darts.Partie`

La classe qui contient toute la gestion de partie.

Graphe de collaboration de com.lasalle84.mobile_darts.Partie :



Fonctions membres publiques

- void `actualiserJoueurIHM` (String monJoueur)
Actualiser le joueur à IHM.
- void `actualiserScoreIHM` (Joueur monJoueur, int score)
Méthode permettant d'actualiser le score du joueur à IHM.
- void `attendreImpact` ()
Méthode permettant d'attendre qu'un nouvel impact arrive.
- void `attendreJoueurSuivant` ()
- void `cibleManquee` ()
Méthode envoyant une fléchette manquée.

- void `demarrerPartie` ()
Méthode permettant de démarrer la partie.
- void `envoyerAquittement` ()
Envoyer Aquittement à la cible.
- void `envoyerGagnantIHM` (`Joueur` monJoueur)
Envoyer le gagnant a IHM.
- `TypeJeu` `getTypeJeu` ()
- void `impactIHM` (int Touche, int MultiplicateurTouche, int ID)
Envoyer impact à IHM.
- `Partie` (`ArrayList`< `Joueur` > `lesJoueurs`, `TypeJeu` `typeJeu`, `Handler` `handlerUI`)
- void `traiterTrame` (String trame)

Attributs publics statiques

- static final int `ENVOYER_AQUITTEMENT` = 7
- static final int `GAGNANT` = 6
- static final int `IMPACT` = 5
- static final int `JOUEUR_SUIVANT` = 3
- static final int `NB_FLECHETTE` = 3
Nombre de flechettes par volee.
- static final int `SET_SCORE` = 4

Attributs privés

- boolean `estAuSuivant` = false
Booléen permettant de savoir si il faut passer au joueur suivant.
- boolean `estDoubleImpact` = false
Booléen pour savoir si le dernier impact est du type double.
- boolean `estFini` = false
Booléen pour savoir si la partie est finie.
- `Handler` `handlerUI` = null
Handler pour gérer l'interface.
- int `impact` []
Les impacts.
- boolean `impactEstRecu`
Booléen pour savoir quand on reçoit une trame impact.
- `ArrayList`< `Joueur` > `lesJoueurs`
Les objets Joueur stockés dans un conteneur (Queue)
- int `nbJoueurs`
Nombre de joueurs.
- int `nbManche`
Nombre de manches.
- `TypeJeu` `typeJeu`
Mode de jeu.

Attributs privés statiques

- static final String `TAG` = "_Partie_"
Tag pour Log.

9.6.1 Description détaillée

La classe qui contient toute la gestion de partie.

Définition à la ligne 25 du fichier `Partie.java`.

9.6.2 Documentation des constructeurs et destructeur

9.6.2.1 Partie()

```
com.lasalle84.mobile_darts.Partie.Partie (
    ArrayList< Joueur > lesJoueurs,
    TypeJeu typeJeu,
    Handler handlerUI )
```

Définition à la ligne 56 du fichier [Partie.java](#).

Références [com.lasalle84.mobile_darts.Partie.handlerUI](#), [com.lasalle84.mobile_darts.Partie.lesJoueurs](#), et [com.lasalle84.mobile_darts.Partie.typeJeu](#).

```
00057      {
00058          this.lesJoueurs = lesJoueurs;
00059          this.typeJeu = typeJeu;
00060          this.handlerUI = handlerUI;
00061          this.nbManche = 0;
00062          this.nbJoueurs = this.lesJoueurs.size();
00063          impact = new int[3];
00064      }
```

9.6.3 Documentation des fonctions membres

9.6.3.1 actualiserJoueurIHM()

```
com.lasalle84.mobile_darts.Partie.actualiserJoueurIHM (
    String monJoueur )
```

Actualiser le joueur à IHM.

Paramètres

<i>monJoueur</i>	
------------------	--

Définition à la ligne 186 du fichier [Partie.java](#).

Références [com.lasalle84.mobile_darts.Partie.JOUEUR_SUIVANT](#).

Référencé par [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#).

```
00187      {
00188          Log.d(TAG, "actualiserJoueurIHM()");
00189          Message msg = Message.obtain();
00190          msg.what = JOUEUR_SUIVANT;
00191          Bundle b = new Bundle();
00192          b.putString("joueur", monJoueur);
00193          msg.setData(b);
00194          handlerUI.sendMessage(msg);
00195      }
```

9.6.3.2 actualiserScoreIHM()

```
com.lasalle84.mobile_darts.Partie.actualiserScoreIHM (
    Joueur monJoueur,
    int score )
```

Méthode permettant d'actualiser le score du joueur à IHM.

Paramètres

<i>monJoueur</i>	(Joueur à actualiser)
<i>score</i>	(Score du Joueur)

Définition à la ligne 168 du fichier `Partie.java`.

Références `com.lasalle84.mobile_darts.Joueur.getNom()`, et `com.lasalle84.mobile_darts.Partie.SET_SCORE`.

Référencé par `com.lasalle84.mobile_darts.Partie.demarrerPartie()`.

```

00169      {
00170          Log.d(TAG, "actualiserScoreIHM()");
00171          Message msg = Message.obtain();
00172          msg.what = SET_SCORE;
00173          Bundle b = new Bundle();
00174          b.putString("joueur", monJoueur.getNom());
00175          b.putInt("score", score);
00176          msg.setData(b);
00177          handlerUI.sendMessage(msg);
00178      }

```

9.6.3.3 attendreImpact()

`com.lasalle84.mobile_darts.Partie.attendreImpact ()`

Méthode permettant d'attendre qu'un nouvel impact arrive.

Définition à la ligne 130 du fichier `Partie.java`.

Références `com.lasalle84.mobile_darts.Partie.impactIHM()`.

Référencé par `com.lasalle84.mobile_darts.Partie.demarrerPartie()`.

```

00130      {
00131          Log.d(TAG, "attendreImpact()");
00132          while (!impactEstRecu)
00133          {
00134              sleep(1000);
00135          }
00136          impactIHM(impact[0], impact[1], impact[2]);
00137          if (impact[1] == 2)
00138          {
00139              estDoubleImpact = true;
00140          }
00141          else
00142          {
00143              estDoubleImpact = false;
00144          }
00145      }
00146      Log.d(TAG, "Valeur Touche" + impact[0] + "; Multiplicateur Touche:" +
00147          impact[1] + "; ID Touche : " + impact[2]);
00148  }
00149

```

9.6.3.4 attendreJoueurSuivant()

```
void com.lasalle84.mobile_darts.Partie.attendreJoueurSuivant ( )
```

Définition à la ligne 150 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#).

```
00151      {
00152          Log.d(TAG, "attendreJoueurSuivant()");
00153          while (!estAuSuivant)
00154          {
00155              sleep(1000);
00156          }
00157          Log.d(TAG, "Au suivant" + estAuSuivant);
00158          estAuSuivant = false;
00159      }
```

9.6.3.5 cibleManquee()

```
com.lasalle84.mobile_darts.Partie.cibleManquee ( )
```

Méthode envoyant une fléchette manquée.

Fonction envoyant une fléchette manquée.

Définition à la ligne 254 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserWidgets\(\)](#).

```
00255      {
00256          impact[0] = 0;
00257          impact[1] = 0;
00258          impactEstRecu = true;
00259      }
```

9.6.3.6 demarrerPartie()

```
com.lasalle84.mobile_darts.Partie.demarrerPartie ( )
```

Méthode permettant de démarrer la partie.

Définition à la ligne 72 du fichier [Partie.java](#).

Références [com.lasalle84.mobile_darts.Partie.actualiserJoueurIHM\(\)](#), [com.lasalle84.mobile_darts.Partie.actualiserScoreIHM\(\)](#), [com.lasalle84.mobile_darts.Partie.attendreImpact\(\)](#), [com.lasalle84.mobile_darts.Partie.attendreJoueurSuivant\(\)](#), [com.lasalle84.mobile_darts.Partie.envoyerGagnantIHM\(\)](#), [com.lasalle84.mobile_darts.Partie.estDoubleImpact](#), [com.lasalle84.mobile_darts.TypeJeu.estDoubleOut\(\)](#), [com.lasalle84.mobile_darts.Joueur.getNom\(\)](#), [com.lasalle84.mobile_darts.Joueur.getScore\(\)](#), [com.lasalle84.mobile_darts.Joueur.isaFin\(\)](#), [com.lasalle84.mobile_darts.Partie.NB_FLECHETTE](#), et [com.lasalle84.mobile_darts.Joueur.retirerPoint\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserWidgets\(\)](#).

```

00072         {
00073             Log.d(TAG, "demarrerPartie()");
00074             estAuSuivant = true;
00075             do
00076             {
00077                 nbManche++;
00078                 Iterator<Joueur> it = lesJoueurs.iterator();
00079                 Log.d(TAG, "Manche numéro " + nbManche);
00080                 while (it.hasNext())
00081                 {
00082                     Joueur monJoueur = it.next();
00083                     envoyerGagnantIHM(monJoueur);
00084                     Log.d(TAG, "C'est le tour de " + monJoueur.getNom());
00085                     attendreJoueurSuivant();
00086                     actualiserJoueurIHM(monJoueur.getNom());
00087                     int pointVollee = 0;
00088
00089                     for(int i = 0; i < NB_FLECHETTE; i++)
00090                     {
00091                         if (monJoueur.isaFini())
00092                         {
00093                             envoyerGagnantIHM(monJoueur);
00094                             i = NB_FLECHETTE;
00095                             estFini = true;
00096                         }
00097                         else
00098                         {
00099                             impactEstRecu = false;
00100                             attendreImpact();
00101                             pointVollee += impact[0]*impact[1];
00102                             if(!monJoueur.retirerPoint(pointVollee, this))
00103                             {
00104                                 i = NB_FLECHETTE;
00105                                 pointVollee = 0;
00106                             }
00107                             else if (monJoueur.getScore() == 0 && !typeJeu.
estDoubleOut())
00108                             {
00109                                 envoyerGagnantIHM(monJoueur);
00110                                 i = NB_FLECHETTE;
00111                                 estFini = true;
00112                             }
00113                             else if (monJoueur.getScore() == 0 && typeJeu.
estDoubleOut() && estDoubleImpact)
00114                             {
00115                                 envoyerGagnantIHM(monJoueur);
00116                                 i = NB_FLECHETTE;
00117                                 estFini = true;
00118                             }
00119                         }
00120                     }
00121                     actualiserScoreIHM(monJoueur, monJoueur.getScore());
00122                 }
00123             }while (!estFini);
00124         }

```

9.6.3.7 envoyerAquittement()

`com.lasalle84.mobile_darts.Partie.envoyerAquittement ()`

Envoyer Aquittement à la cible.

Définition à la ligne 224 du fichier `Partie.java`.

Références `com.lasalle84.mobile_darts.Partie.ENVOYER_AQUITTEMENT`.

Référencé par `com.lasalle84.mobile_darts.Partie.traiterTrame()`.

```

00225     {
00226         Log.d(TAG, "envoyerAquittement() ");
00227         Message msg = Message.obtain();
00228         msg.what = ENVOYER_AQUITTEMENT;
00229         handlerUI.sendMessage(msg);
00230     }

```

9.6.3.8 envoyerGagnantIHM()

`com.lasalle84.mobile_darts.Partie.envoyerGagnantIHM (`
`Joueur monJoueur)`

Envoyer le gagnant a IHM.

Paramètres

<i>monJoueur</i>	(Joueur ayant gagné la partie)
------------------	--------------------------------

Définition à la ligne 238 du fichier [Partie.java](#).

Références [com.lasalle84.mobile_darts.Partie.GAGNANT](#), et [com.lasalle84.mobile_darts.Joueur.getNom\(\)](#).

Référencé par [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#).

```

00239      {
00240          Log.d(TAG, "envoyerGagnantIHM() " + monJoueur.getNom());
00241          estFini = true;
00242          Message msg = Message.obtain();
00243          msg.what = GAGNANT;
00244          Bundle b = new Bundle();
00245          b.putString("gagnant", monJoueur.getNom());
00246          msg.setData(b);
00247          handlerUI.sendMessage(msg);
00248      }

```

9.6.3.9 getTypeJeu()

[TypeJeu](#) [com.lasalle84.mobile_darts.Partie.getTypeJeu \(\)](#)

Définition à la ligne 292 du fichier [Partie.java](#).

Références [com.lasalle84.mobile_darts.Partie.typeJeu](#).

Référencé par [com.lasalle84.mobile_darts.Joueur.retirerPoint\(\)](#).

```

00293      {
00294          return typeJeu;
00295      }

```

9.6.3.10 impactIHM()

```

com.lasalle84.mobile_darts.Partie.impactIHM (
    int Touche,
    int MultiplicateurTouche,
    int ID )

```

Envoyer impact à IHM.

Paramètres

<i>Touche</i>	(Valeur de la touche)
<i>MultiplicateurTouche</i>	(Valeur du multiplicateur de la touche)
<i>ID</i>	(Identifiant de la fléchette dans la volée)

Définition à la ligne 205 du fichier [Partie.java](#).

Références [com.lasalle84.mobile_darts.Partie.IMPACT](#).

Référencé par [com.lasalle84.mobile_darts.Partie.attendreImpact\(\)](#).

```

00206    {
00207        Log.d(TAG, "impactIHM() " + Touche + " - " + MultiplicateurTouche);
00208        Message msg = Message.obtain();
00209        msg.what = IMPACT;
00210        Bundle b = new Bundle();
00211        b.putInt("Touche", Touche);
00212        b.putInt("MultiplicateurTouche", MultiplicateurTouche);
00213        b.putInt("ID", ID);
00214        msg.setData(b);
00215        handlerUI.sendMessage(msg);
00216    }

```

9.6.3.11 traiterTrame()

```

void com.lasalle84.mobile_darts.Partie.traiterTrame (
    String trame )

```

Définition à la ligne 265 du fichier [Partie.java](#).

Références [com.lasalle84.mobile_darts.Partie.envoyerAquittement\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#).

```

00266    {
00267        Log.d(TAG, "traiterTrame() " + trame);
00268        trame = trame.replace("\r\n", "");
00269        String[] trameSectionnee = trame.split(";", 5);
00270        for(int i=0; i < trameSectionnee.length; ++i)
00271        {
00272            Log.d(TAG, "traiterTrame() trameSectionnee[" + i + "] = " + trameSectionnee[i]);
00273        }
00274
00275        switch (trameSectionnee[1])
00276        {
00277            case("HIT"):
00278                Log.d(TAG, "traiterTrame : HIT");
00279                impact[0] = Integer.parseInt(trameSectionnee[2]);
00280                impact[1] = Integer.parseInt(trameSectionnee[3]);
00281                impact[2] = Integer.parseInt(trameSectionnee[4]);
00282                envoyerAquittement();
00283                impactEstRecu = true;
00284                break;
00285            case("NEXT"):
00286                Log.d(TAG, "traiterTrame : NEXT");
00287                estAuSuivant = true;
00288                break;
00289        }
00290    }

```

9.6.4 Documentation des données membres

9.6.4.1 ENVOYER_AQUITTEMENT

```

final int com.lasalle84.mobile_darts.Partie.ENVOYER_AQUITTEMENT = 7 [static]

```

Définition à la ligne 40 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.envoyerAquittement\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#).

9.6.4.2 estAuSuivant

```
boolean com.lasalle84.mobile_darts.Partie.estAuSuivant = false [private]
```

Booléen permettant de savoir si il faut passer au joueur suivant.

Définition à la ligne 54 du fichier [Partie.java](#).

9.6.4.3 estDoubleImpact

```
boolean com.lasalle84.mobile_darts.Partie.estDoubleImpact = false [private]
```

Booléen pour savoir si le dernier impact est du type double.

Définition à la ligne 53 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#).

9.6.4.4 estFini

```
boolean com.lasalle84.mobile_darts.Partie.estFini = false [private]
```

Booléen pour savoir si la partie est finie.

Définition à la ligne 51 du fichier [Partie.java](#).

9.6.4.5 GAGNANT

```
final int com.lasalle84.mobile_darts.Partie.GAGNANT = 6 [static]
```

Définition à la ligne 39 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.envoyerGagnantIHM\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.↔initialiserHandler\(\)](#).

9.6.4.6 handlerUI

```
Handler com.lasalle84.mobile_darts.Partie.handlerUI = null [private]
```

Handler pour gérer l'interface.

Définition à la ligne 52 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.Partie\(\)](#).

9.6.4.7 IMPACT

```
final int com.lasalle84.mobile_darts.Partie.IMPACT = 5 [static]
```

Définition à la ligne 38 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.impactIHM\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#).

9.6.4.8 impact

```
int com.lasalle84.mobile_darts.Partie.impact[] [private]
```

Les impacts.

Définition à la ligne 49 du fichier [Partie.java](#).

9.6.4.9 impactEstRecu

```
boolean com.lasalle84.mobile_darts.Partie.impactEstRecu [private]
```

Booléen pour savoir quand on reçoit une trame impact.

Définition à la ligne 50 du fichier [Partie.java](#).

9.6.4.10 JOUEUR_SUIVANT

```
final int com.lasalle84.mobile_darts.Partie.JOUEUR_SUIVANT = 3 [static]
```

HandlerUI

Définition à la ligne 36 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.actualiserJoueurIHM\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.↵initialiserHandler\(\)](#).

9.6.4.11 lesJoueurs

```
ArrayList<Joueur> com.lasalle84.mobile_darts.Partie.lesJoueurs [private]
```

Les objets [Joueur](#) stockés dans un conteneur (Queue)

Définition à la ligne 48 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.Partie\(\)](#).

9.6.4.12 NB_FLECHETTE

```
final int com.lasalle84.mobile_darts.Partie.NB_FLECHETTE = 3 [static]
```

Nombre de flechettes par volée.

Définition à la ligne 31 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#).

9.6.4.13 nbJoueurs

```
int com.lasalle84.mobile_darts.Partie.nbJoueurs [private]
```

Nombre de joueurs.

Attributs

Définition à la ligne 45 du fichier [Partie.java](#).

9.6.4.14 nbManche

```
int com.lasalle84.mobile_darts.Partie.nbManche [private]
```

Nombre de manches.

Définition à la ligne 46 du fichier [Partie.java](#).

9.6.4.15 SET_SCORE

```
final int com.lasalle84.mobile_darts.Partie.SET_SCORE = 4 [static]
```

Définition à la ligne 37 du fichier [Partie.java](#).

Référencé par [com.lasalle84.mobile_darts.Partie.actualiserScoreIHM\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiser↔Handler\(\)](#).

9.6.4.16 TAG

```
final String com.lasalle84.mobile_darts.Partie.TAG = "_Partie_" [static], [private]
```

Tag pour Log.

— Constantes

Définition à la ligne 30 du fichier [Partie.java](#).

9.6.4.17 `typeJeu`

`TypeJeu` `com.lasalle84.mobile_darts.Partie.typeJeu` [private]

Mode de jeu.

Définition à la ligne 47 du fichier [Partie.java](#).

Référéncé par [com.lasalle84.mobile_darts.Partie.getTypeJeu\(\)](#), et [com.lasalle84.mobile_darts.Partie.Partie\(\)](#).

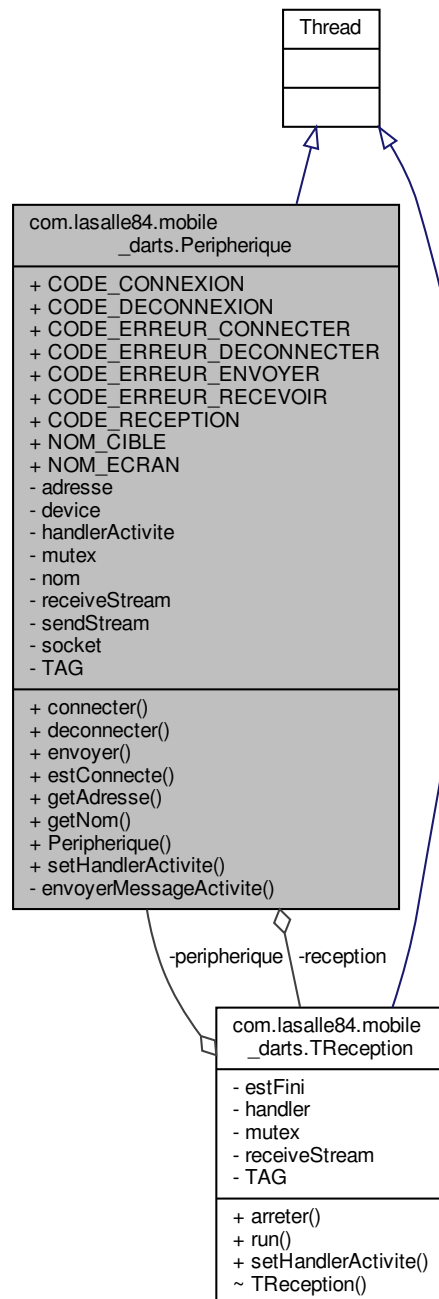
La documentation de cette classe a été générée à partir du fichier suivant :

— [Partie.java](#)

9.7 Référence de la classe `com.lasalle84.mobile_darts.Peripherique`

La classe qui contient toute la gestion des périphériques.

Graphe de collaboration de com.lasalle84.mobile_darts.Peripherique :



Fonctions membres publiques

- void **connecter** ()
Méthode pour se connecter sur le périphérique en Bluetooth.
- void **deconnecter** ()
Méthode pour se deconnecter sur le périphérique en Bluetooth.
- void **envoyer** (final String data)
Méthode pour envoyer une trame en Bluetooth.
- boolean **estConnecte** ()
Méthode qui retourne l'état de connexion'.
- String **getAdresse** ()

- Méthode qui retourne l'adresse du périphérique.
- String `getNom` ()
- Méthode qui retourne le nom du périphérique.
- `Peripherique` (BluetoothDevice `device`, Handler handler)
- Constructeur de la classe `Peripherique`.
- void `setHandlerActivite` (Handler handler)
- Méthode qui permet de changer d'handler.

Attributs publics statiques

- static final int `CODE_CONNEXION` = 0
- Code de Connection.
- static final int `CODE_DECONNEXION` = 2
- Code de Deconnexion.
- static final int `CODE_ERREUR_CONNECTER` = -3
- Code erreur lors de la connexion.
- static final int `CODE_ERREUR_DECONNECTER` = -4
- Code erreur lors de la déconnexion.
- static final int `CODE_ERREUR_ENVOYER` = -1
- Code erreur lors de l'envoi.
- static final int `CODE_ERREUR_RECEVOIR` = -2
- Code erreur lors de la réception.
- static final int `CODE_RECEPTION` = 1
- Code de `TReception`.
- static final String `NOM_CIBLE` ="cible-darts 2021"
- Le nom du module Bluetooth de la cible-darts 2021.
- static final String `NOM_ECRAN` ="ecran-darts"
- Le nom du module Bluetooth de l'écran.

Fonctions membres privées

- void `envoyerMessageActivite` (int what)
- Méthode qui permet d'envoyer un message à l'activité

Attributs privés

- String `adresse`
- Adresse du peripherique bluetooth.
- BluetoothDevice `device`
- Objet BluetoothDevice.
- Handler `handlerActivite` = null
- Handler qui permet de communiquer avec l'activité
- final ReentrantLock `mutex` = new ReentrantLock()
- Mutex pour le changement d'handler.
- String `nom`
- Nom du peripherique bluetooth.
- InputStream `receiveStream` = null
- Input du Bluetooth.
- `TReception` `reception`
- Thread pour recevoir les trames.
- OutputStream `sendStream` = null
- Output du Bluetooth.
- BluetoothSocket `socket`
- Socket Bluetooth.

Attributs privés statiques

- static final String `TAG` = "_Peripherique_"
- TAG pour log.

9.7.1 Description détaillée

La classe qui contient toute la gestion des périphériques.

Définition à la ligne 28 du fichier [Peripherique.java](#).

9.7.2 Documentation des constructeurs et destructeur

9.7.2.1 Peripherique()

```
com.lasalle84.mobile_darts.Peripherique.Peripherique (
    BluetoothDevice device,
    Handler handler )
```

Constructeur de la classe [Peripherique](#).

Paramètres

<i>device,handler</i>	
-----------------------	--

Définition à la ligne 62 du fichier [Peripherique.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.device](#), et [com.lasalle84.mobile_darts.Peripherique.setHandlerActivite\(\)](#).

```
00063     {
00064         Log.d(TAG, "[Peripherique] " + device.getName() + " : " + device.getAddress());
00065         if(device != null)
00066         {
00067             this.device = device;
00068             this.nom = device.getName();
00069             this.adresse = device.getAddress();
00070             setHandlerActivite(handler);
00071         }
00072         else
00073         {
00074             this.device = device;
00075             this.nom = "Aucun";
00076             this.adresse = "";
00077             setHandlerActivite(handler);
00078         }
00079
00080         try
00081         {
00082             assert device != null;
00083             socket = device.createRfcommSocketToServiceRecord(UUID.fromString("
00001101-0000-1000-8000-00805F9B34FB"));
00084             receiveStream = socket.getInputStream();
00085             sendStream = socket.getOutputStream();
00086         }
00087         catch (IOException e)
00088         {
00089             e.printStackTrace();
00090             socket = null;
00091         }
00092
00093         if(socket != null)
00094             reception = new TReception(this, handlerActivite,
00095                                     receiveStream);
00095     }
```

9.7.3 Documentation des fonctions membres

9.7.3.1 connecter()

com.lasalle84.mobile_darts.Peripherique.connecter ()

Méthode pour se connecter sur le périphérique en Bluetooth.

Définition à la ligne 146 du fichier [Peripherique.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.envoyerMessageActivite\(\)](#), et [com.lasalle84.mobile_darts.Peripherique.getNom\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.gererConnexionCibleDARTS\(\)](#), et [com.lasalle84.mobile_darts.ActiviteLancerPartie.gererConnexionEcranDARTS\(\)](#).

```

00147     {
00148         Log.d(TAG,"[connecter] " + device.getName() + " : " + device.getAddress());
00149         new Thread()
00150         {
00151             @Override public void run()
00152             {
00153                 try
00154                 {
00155                     socket.connect();
00156
00157                     envoyerMessageActivite(CODE_CONNEXION);
00158
00159                     // on démarre le thread de réception
00160                     reception.start();
00161                     Log.d(TAG,"connexion reussie " + getNom());
00162
00163                 }
00164                 catch (IOException e)
00165                 {
00166                     e.printStackTrace();
00167                     Log.d(TAG,"erreur connexion " + getNom());
00168                     envoyerMessageActivite(
CODE_ERREUR_CONNECTER);
00169                 }
00170             }
00171         }.start();
00172     }

```

9.7.3.2 deconnecter()

com.lasalle84.mobile_darts.Peripherique.deconnecter ()

Méthode pour se deconnecter sur le périphérique en Bluetooth.

Définition à la ligne 178 du fichier [Peripherique.java](#).

Références [com.lasalle84.mobile_darts.TReception.arreter\(\)](#), [com.lasalle84.mobile_darts.Peripherique.envoyerMessageActivite\(\)](#), et [com.lasalle84.mobile_darts.Peripherique.getNom\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.gererConnexionCibleDARTS\(\)](#), [com.lasalle84.mobile_darts.ActiviteLancerPartie.gererConnexionEcranDARTS\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.onActivityResult\(\)](#).

```

00179     {
00180         Log.d(TAG,"[deconnecter] " + device.getName() + " : " + device.getAddress());
00181         new Thread()
00182         {
00183             @Override public void run()
00184             {
00185                 try
00186                 {
00187                     reception.arreter();
00188                     socket.close();
00189                     envoyerMessageActivite(
CODE_DECONNEXION);
00190                     Log.d(TAG,"déconnexion reussie " + getNom());
00191                 }
00192                 catch (IOException e)
00193                 {
00194                     e.printStackTrace();
00195                     envoyerMessageActivite(
CODE_ERREUR_DECONNECTER);
00196                 }
00197             }
00198         }.start();
00199     }

```


9.7.3.3 envoyer()

```
com.lasalle84.mobile_darts.Peripherique.envoyer (
    final String data )
```

Méthode pour envoyer une trame en Bluetooth.

Définition à la ligne 205 du fichier [Peripherique.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.envoyerMessageActivite\(\)](#), et [com.lasalle84.mobile_darts.Peripherique.getNom\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.envoyerAquitement\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserPartie\(\)](#).

```
00206     {
00207         Log.d(TAG, "[envoyer] " + device.getName() + " : " + device.getAddress());
00208         if(socket == null)
00209         {
00210             return;
00211         }
00212         new Thread()
00213         {
00214             @Override public void run()
00215             {
00216                 try
00217                 {
00218                     if(socket.isConnected())
00219                     {
00220                         sendStream.write(data.getBytes());
00221                         sendStream.flush();
00222                         Log.d(TAG, "[envoyer] trame : " + data);
00223                     }
00224                 }
00225                 catch (IOException e)
00226                 {
00227                     e.printStackTrace();
00228                     Log.d(TAG, "Erreur socket write : " + getNom());
00229                     envoyerMessageActivite(
00230                         CODE_ERREUR_ENVOYER);
00231                 }
00232             }
00233         }.start();
00234     }
```

9.7.3.4 envoyerMessageActivite()

```
void com.lasalle84.mobile_darts.Peripherique.envoyerMessageActivite (
    int what ) [private]
```

Méthode qui permet d'envoyer un message à l'activité

Définition à la ligne 239 du fichier [Peripherique.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.getAdresse\(\)](#), et [com.lasalle84.mobile_darts.Peripherique.getNom\(\)](#).

Référencé par [com.lasalle84.mobile_darts.Peripherique.connecter\(\)](#), [com.lasalle84.mobile_darts.Peripherique.deconnecter\(\)](#), et [com.lasalle84.mobile_darts.Peripherique.envoyer\(\)](#).

```
00240     {
00241         if(handlerActivite == null)
00242             return;
00243         Message msg = Message.obtain();
00244         msg.what = what;
00245         Bundle b = new Bundle();
00246         b.putString("nom", getNom());
00247         b.putString("adresse", getAdresse());
00248         msg.setData(b);
00249         mutex.lock();
00250         handlerActivite.sendMessage(msg);
00251         mutex.unlock();
00252     }
```

9.7.3.5 `estConnecte()`

```
com.lasalle84.mobile_darts.Peripherique.estConnecte ( )
```

Méthode qui retourne l'état de connexion'.

Renvoie

boolean

Définition à la ligne 122 du fichier `Peripherique.java`.

Référencé par `com.lasalle84.mobile_darts.ActiviteLancerPartie.demarrerActiviteJouerPartie()`, `com.lasalle84.mobile_darts.ActiviteLancerPartie.gererConnexionCibleDARTS()`, et `com.lasalle84.mobile_darts.ActiviteLancerPartie.gererConnexionEcranDARTS()`.

```
00123    {  
00124        if(device == null | socket == null)  
00125            return false;  
00126        return socket.isConnected();  
00127    }
```

9.7.3.6 `getAdresse()`

```
com.lasalle84.mobile_darts.Peripherique.getAdresse ( )
```

Méthode qui retourne l'adresse du périphérique.

Renvoie

adresse l'adresse du périphérique

Définition à la ligne 112 du fichier `Peripherique.java`.

Références `com.lasalle84.mobile_darts.Peripherique.adresse`.

Référencé par `com.lasalle84.mobile_darts.TReception.arreter()`, `com.lasalle84.mobile_darts.Peripherique.envoyerMessageActivite()`, et `com.lasalle84.mobile_darts.TReception.run()`.

```
00113    {  
00114        return adresse;  
00115    }
```

9.7.3.7 `getNom()`

```
com.lasalle84.mobile_darts.Peripherique.getNom ( )
```

Méthode qui retourne le nom du périphérique.

Renvoie

nom le nom du périphérique

Définition à la ligne 102 du fichier `Peripherique.java`.

Références `com.lasalle84.mobile_darts.Peripherique.nom`.

Référencé par `com.lasalle84.mobile_darts.TReception.arreter()`, `com.lasalle84.mobile_darts.Peripherique.connecter()`, `com.lasalle84.mobile_darts.Peripherique.deconnecter()`, `com.lasalle84.mobile_darts.Peripherique.envoyer()`, `com.lasalle84.mobile_darts.Peripherique.envoyerMessageActivite()`, et `com.lasalle84.mobile_darts.TReception.run()`.

```
00103    {  
00104        return nom;  
00105    }
```

9.7.3.8 setHandlerActivite()

```
void com.lasalle84.mobile_darts.Peripherique.setHandlerActivite (
    Handler handler )
```

Méthode qui permet de changer d'handler.

Définition à la ligne 132 du fichier [Peripherique.java](#).

Références [com.lasalle84.mobile_darts.TReception.setHandlerActivite\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler\(\)](#), [com.lasalle84.mobile_darts.Peripherique.Peripherique\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.recupererPeripheriques\(\)](#).

```
00133     {
00134         Log.d(TAG, "[setHandlerActivite] " + handler);
00135         mutex.lock();
00136         this.handlerActivite = handler;
00137         if(reception != null)
00138             reception.setHandlerActivite(handler);
00139         mutex.unlock();
00140     }
```

9.7.4 Documentation des données membres

9.7.4.1 adresse

```
String com.lasalle84.mobile_darts.Peripherique.adresse [private]
```

Adresse du peripherique bluetooth.

Définition à la ligne 49 du fichier [Peripherique.java](#).

Référencé par [com.lasalle84.mobile_darts.Peripherique.getAdresse\(\)](#).

9.7.4.2 CODE_CONNEXION

```
final int com.lasalle84.mobile_darts.Peripherique.CODE_CONNEXION = 0 [static]
```

Code de Connection.

Définition à la ligne 36 du fichier [Peripherique.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler\(\)](#).

9.7.4.3 CODE_DECONNEXION

```
final int com.lasalle84.mobile_darts.Peripherique.CODE_DECONNEXION = 2 [static]
```

Code de Deconnexion.

Définition à la ligne 38 du fichier [Peripherique.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler\(\)](#), et [com.lasalle84.mobile_darts.TReception.run\(\)](#).

9.7.4.4 `CODE_ERREUR_CONNECTER`

```
final int com.lasalle84.mobile_darts.Peripherique.CODE_ERREUR_CONNECTER = -3 [static]
```

Code erreur lors de la connexion.

Définition à la ligne 41 du fichier `Peripherique.java`.

Référencé par `com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler()`.

9.7.4.5 `CODE_ERREUR_DECONNECTER`

```
final int com.lasalle84.mobile_darts.Peripherique.CODE_ERREUR_DECONNECTER = -4 [static]
```

Code erreur lors de la déconnexion.

Définition à la ligne 42 du fichier `Peripherique.java`.

Référencé par `com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler()`.

9.7.4.6 `CODE_ERREUR_ENVOYER`

```
final int com.lasalle84.mobile_darts.Peripherique.CODE_ERREUR_ENVOYER = -1 [static]
```

Code erreur lors de l'envoi.

Définition à la ligne 39 du fichier `Peripherique.java`.

Référencé par `com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler()`.

9.7.4.7 `CODE_ERREUR_RECEVOIR`

```
final int com.lasalle84.mobile_darts.Peripherique.CODE_ERREUR_RECEVOIR = -2 [static]
```

Code erreur lors de la réception.

Définition à la ligne 40 du fichier `Peripherique.java`.

Référencé par `com.lasalle84.mobile_darts.TReception.run()`.

9.7.4.8 `CODE_RECEPTION`

```
final int com.lasalle84.mobile_darts.Peripherique.CODE_RECEPTION = 1 [static]
```

Code de `TReception`.

Définition à la ligne 37 du fichier `Peripherique.java`.

Référencé par `com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler()`, et `com.lasalle84.mobile_darts.TReception.run()`.

9.7.4.9 device

```
BluetoothDevice com.lasalle84.mobile_darts.Peripherique.device [private]
```

Objet BluetoothDevice.

Attributs

Définition à la ligne 47 du fichier [Peripherique.java](#).

Référencé par [com.lasalle84.mobile_darts.Peripherique.Peripherique\(\)](#).

9.7.4.10 handlerActivite

```
Handler com.lasalle84.mobile_darts.Peripherique.handlerActivite = null [private]
```

Handler qui permet de communiquer avec l'activité

Définition à la ligne 50 du fichier [Peripherique.java](#).

9.7.4.11 mutex

```
final ReentrantLock com.lasalle84.mobile_darts.Peripherique.mutex = new ReentrantLock() [private]
```

Mutex pour le changement d'handler.

Définition à la ligne 55 du fichier [Peripherique.java](#).

9.7.4.12 nom

```
String com.lasalle84.mobile_darts.Peripherique.nom [private]
```

Nom du peripherique bluetooth.

Définition à la ligne 48 du fichier [Peripherique.java](#).

Référencé par [com.lasalle84.mobile_darts.Peripherique.getNom\(\)](#).

9.7.4.13 NOM_CIBLE

```
final String com.lasalle84.mobile_darts.Peripherique.NOM_CIBLE ="cible-darts 2021" [static]
```

Le nom du module Bluetooth de la cible-darts 2021.

Définition à la ligne 34 du fichier [Peripherique.java](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler\(\)](#), [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler\(\)](#), et [com.lasalle84.mobile_darts.ActiviteLancerPartie.rechercherCibleDARTS\(\)](#).

9.7.4.14 `NOM_ECRAN`

```
final String com.lasalle84.mobile_darts.Peripherique.NOM_ECRAN = "ecran-darts" [static]
```

Le nom du module Bluetooth de l'écran.

Définition à la ligne 35 du fichier `Peripherique.java`.

Référencé par `com.lasalle84.mobile_darts.ActiviteLancerPartie.initialiserHandler()`, `com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserHandler()`, et `com.lasalle84.mobile_darts.ActiviteLancerPartie.rechercherEcranDARTS()`.

9.7.4.15 `receiveStream`

```
InputStream com.lasalle84.mobile_darts.Peripherique.receiveStream = null [private]
```

Input du Bluetooth.

Définition à la ligne 52 du fichier `Peripherique.java`.

9.7.4.16 `reception`

```
TReception com.lasalle84.mobile_darts.Peripherique.reception [private]
```

`Thread` pour recevoir les trames.

Définition à la ligne 54 du fichier `Peripherique.java`.

9.7.4.17 `sendStream`

```
OutputStream com.lasalle84.mobile_darts.Peripherique.sendStream = null [private]
```

Output du Bluetooth.

Définition à la ligne 53 du fichier `Peripherique.java`.

9.7.4.18 `socket`

```
BluetoothSocket com.lasalle84.mobile_darts.Peripherique.socket [private]
```

Socket Bluetooth.

Définition à la ligne 51 du fichier `Peripherique.java`.

9.7.4.19 TAG

```
final String com.lasalle84.mobile_darts.Peripherique.TAG = "_Peripherique_" [static], [private]
```

TAG pour log.

Constantes

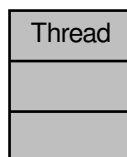
Définition à la ligne 33 du fichier [Peripherique.java](#).

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

— [Peripherique.java](#)

9.8 Référence de la classe Thread

Graphe de collaboration de Thread :



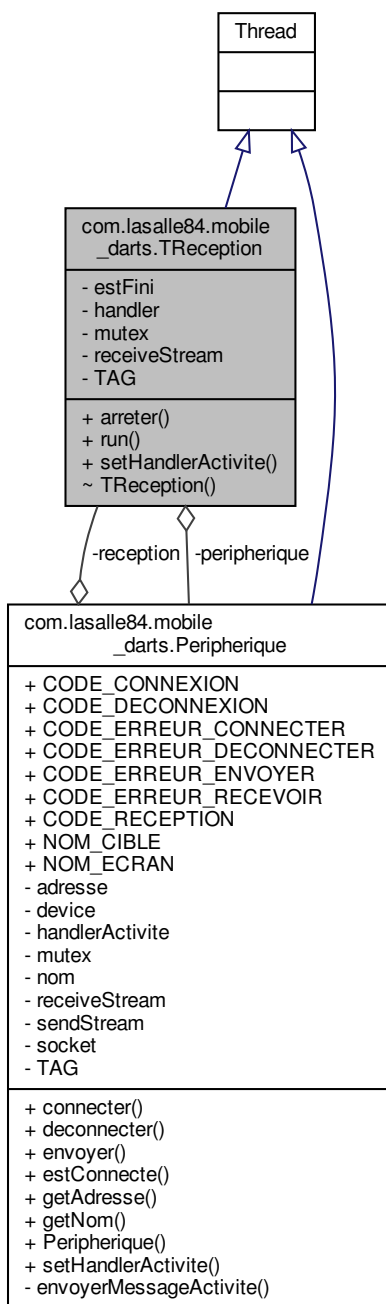
La documentation de cette classe a été générée à partir du fichier suivant :

— [TReception.java](#)

9.9 Référence de la classe com.lasalle84.mobile_darts.TReception

La classe qui contient toute la gestion des thread de reception et envoi de trames.

Graphe de collaboration de com.lasalle84.mobile_darts.TReception :



Fonctions membres publiques

- void `arreter()`
Arrêt du `Thread` permettant l'écoute de trame du périphérique Bluetooth.
- void `run()`
Execution du `Thread` permettant l'écoute de trame du périphérique Bluetooth.
- void `setHandlerActivite` (Handler `handler`)

Attributs privés

- boolean `estFini`


```

    Thread est fini.
— Handler handler
    Handler pour faire passer la trame arrivant.
— final ReentrantLock mutex = new ReentrantLock()
    Mutex pour le changement d'handler.
— Peripherique peripherique
    Peripherique à écouter.
— InputStream receiveStream
    Input du Bluetooth.
— final String TAG = "_TReception_"
    TAG.

```

9.9.1 Description détaillée

La classe qui contient toute la gestion des thread de reception et envoi de trames.

Définition à la ligne 26 du fichier `TReception.java`.

9.9.2 Documentation des fonctions membres

9.9.2.1 arreter()

```
com.lasalle84.mobile_darts.TReception.arreter ( )
```

Arrêt du `Thread` permettant l'écoute de trame du périphérique Bluetooth.

Définition à la ligne 114 du fichier `TReception.java`.

Références `com.lasalle84.mobile_darts.Peripherique.getAdresse()`, et `com.lasalle84.mobile_darts.Peripherique.getNom()`.

Référencé par `com.lasalle84.mobile_darts.Peripherique.deconnecter()`.

```

00115     {
00116         Log.d(TAG, "[arreter] " + peripherique.getNom() + " : " +
peripherique.getAdresse());
00117         if(!estFini)
00118         {
00119             estFini = true;
00120         }
00121         try
00122         {
00123             Thread.sleep(250);
00124         }
00125         catch (InterruptedException e)
00126         {
00127             e.printStackTrace();
00128         }
00129     }

```

9.9.2.2 run()

```
com.lasalle84.mobile_darts.TReception.run ( )
```

Execution du [Thread](#) permettant l'écoute de trame du périphérique Bluetooth.

Définition à la ligne 51 du fichier [TReception.java](#).

Références [com.lasalle84.mobile_darts.Peripherique.CODE_DECONNEXION](#), [com.lasalle84.mobile_darts.Peripherique.CODE_ERREUR_RECEVOIR](#), [com.lasalle84.mobile_darts.Peripherique.CODE_RECEPTION](#), [com.lasalle84.mobile_darts.Peripherique.getAdresse\(\)](#), et [com.lasalle84.mobile_darts.Peripherique.getNom\(\)](#).

```
00052     {
00053         Log.d(TAG, "[run] début " + peripherique.getNom() + " : " +
peripherique.getAdresse());
00054         BufferedReader reception = new BufferedReader(new InputStreamReader(
receiveStream));
00055         while(!estFin)
00056         {
00057             try
00058             {
00059                 String trame = "";
00060                 if(reception.ready())
00061                 {
00062                     trame = reception.readLine();
00063                 }
00064                 if(trame.length() > 0)
00065                 {
00066                     Log.d(TAG, "[run] trame reçue : " + trame);
00067                     Message msg = Message.obtain();
00068                     msg.what = Peripherique.CODE_RECEPTION;
00069                     Bundle b = new Bundle();
00070                     b.putString("nom", peripherique.getNom());
00071                     b.putString("adresse", peripherique.getAdresse());
00072                     b.putString("donnees", trame);
00073                     msg.setData(b);
00074                     handler.sendMessage(msg);
00075                 }
00076             }
00077             catch (IOException e)
00078             {
00079                 e.printStackTrace();
00080                 Log.d(TAG, "Erreur socket read : " + peripherique.
getNom());
00081                 Message msg = Message.obtain();
00082                 msg.what = Peripherique.CODE_ERREUR_RECEVOIR;
00083                 Bundle b = new Bundle();
00084                 b.putString("nom", peripherique.getNom());
00085                 b.putString("adresse", peripherique.getAdresse());
00086                 msg.setData(b);
00087                 handler.sendMessage(msg);
00088             }
00089             try
00090             {
00091                 Thread.sleep(250);
00092             }
00093             catch (InterruptedException e)
00094             {
00095                 e.printStackTrace();
00096             }
00097         }
00098         Message msg = Message.obtain();
00099         Bundle b = new Bundle();
00100         b.putString("nom", peripherique.getNom());
00101         b.putString("adresse", peripherique.getAdresse());
00102         b.putInt("etat", Peripherique.CODE_DECONNEXION);
00103         b.putString("donnees", "");
00104         msg.setData(b);
00105         handler.sendMessage(msg);
00106     }
00107     Log.d(TAG, "[run] fin " + peripherique.getNom() + " : " +
peripherique.getAdresse());
00108 }
```

9.9.2.3 setHandlerActivite()

```
void com.lasalle84.mobile_darts.TReception.setHandlerActivite (
    Handler handler )
```

Définition à la ligne 131 du fichier [TReception.java](#).

Références [com.lasalle84.mobile_darts.TReception.handler](#).

Référencé par [com.lasalle84.mobile_darts.Peripherique.setHandlerActivite\(\)](#).

```
00132     {
00133         Log.d(TAG, "[setHandlerActivite] " + handler);
00134         mutex.lock();
00135         this.handler = handler;
00136         mutex.unlock();
00137     }
```

9.9.3 Documentation des données membres

9.9.3.1 estFini

```
boolean com.lasalle84.mobile_darts.TReception.estFini [private]
```

[Thread](#) est fini.

Définition à la ligne 32 du fichier [TReception.java](#).

9.9.3.2 handler

```
Handler com.lasalle84.mobile_darts.TReception.handler [private]
```

Handler pour faire passer la trame arrivant.

Définition à la ligne 31 du fichier [TReception.java](#).

Référencé par [com.lasalle84.mobile_darts.TReception.setHandlerActivite\(\)](#).

9.9.3.3 mutex

```
final ReentrantLock com.lasalle84.mobile_darts.TReception.mutex = new ReentrantLock() [private]
```

Mutex pour le changement d'handler.

Définition à la ligne 34 du fichier [TReception.java](#).

9.9.3.4 peripherique

```
Peripherique com.lasalle84.mobile_darts.TReception.peripherique [private]
```

[Peripherique](#) à écouter.

Définition à la ligne 30 du fichier [TReception.java](#).

9.9.3.5 receiveStream

```
InputStream com.lasalle84.mobile_darts.TReception.receiveStream [private]
```

Input du Bluetooth.

Définition à la ligne 33 du fichier [TReception.java](#).

9.9.3.6 TAG

```
final String com.lasalle84.mobile_darts.TReception.TAG = "_TReception_" [private]
```

TAG.

Définition à la ligne 28 du fichier [TReception.java](#).

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

— [TReception.java](#)

9.10 Référence de la classe com.lasalle84.mobile_darts.TypeJeu

La classe qui contient toute la gestion des types de parties.

Graphe de collaboration de com.lasalle84.mobile_darts.TypeJeu :

com.lasalle84.mobile_darts.TypeJeu
<ul style="list-style-type: none"> - doubleOut - pointDepart - PARTIE_301 - PARTIE_301_DOUBLE_OUT - PARTIE_501 - PARTIE_501_DOUBLE_OUT - TAG
<ul style="list-style-type: none"> + estDoubleOut() + getPointDepart() + getTypeJeu() + setDoubleOut() + setPointDepart() + TypeJeu() + TypeJeu() + TypeJeu()

Fonctions membres publiques

- boolean `estDoubleOut` ()
- int `getPointDepart` ()
- String `getTypeJeu` ()
- void `setDoubleOut` (boolean `doubleOut`)
- void `setPointDepart` (int `pointDepart`)
- `TypeJeu` ()
- *Constructeur par défaut de la classe `TypeJeu`.*
- `TypeJeu` (int `pointDepart`, boolean `doubleOut`)
- *Constructeur de la classe `TypeJeu`.*
- `TypeJeu` (int `idModeJeu`)

Attributs privés

- boolean `doubleOut`
- int `pointDepart`

Attributs privés statiques

- static final int `PARTIE_301` = 0
- static final int `PARTIE_301_DOUBLE_OUT` = 1
- static final int `PARTIE_501` = 2
- static final int `PARTIE_501_DOUBLE_OUT` = 3
- static final String `TAG` = "_TypeJeu_"

9.10.1 Description détaillée

La classe qui contient toute la gestion des types de parties.

Définition à la ligne 17 du fichier `TypeJeu.java`.

9.10.2 Documentation des constructeurs et destructeur

9.10.2.1 `TypeJeu()` [1/3]

```
com.lasalle84.mobile_darts.TypeJeu.TypeJeu ( )
```

Constructeur par défaut de la classe `TypeJeu`.

Définition à la ligne 39 du fichier `TypeJeu.java`.

```
00040    {
00041        this.pointDepart = 501;
00042        this.doubleOut = true;
00043    }
```

9.10.2.2 `TypeJeu()` [2/3]

```
com.lasalle84.mobile_darts.TypeJeu.TypeJeu (
    int pointDepart,
    boolean doubleOut )
```

Constructeur de la classe [TypeJeu](#).

Définition à la ligne 50 du fichier [TypeJeu.java](#).

Références [com.lasalle84.mobile_darts.TypeJeu.doubleOut](#), et [com.lasalle84.mobile_darts.TypeJeu.pointDepart](#).

```
00051      {
00052          this.pointDepart = pointDepart;
00053          this.doubleOut = doubleOut;
00054      }
```

9.10.2.3 `TypeJeu()` [3/3]

```
com.lasalle84.mobile_darts.TypeJeu.TypeJeu (
    int idModeJeu )
```

Définition à la ligne 56 du fichier [TypeJeu.java](#).

Références [com.lasalle84.mobile_darts.TypeJeu.PARTIE_301](#), [com.lasalle84.mobile_darts.TypeJeu.PARTIE_301_DOUBLE_OUT](#), [com.lasalle84.mobile_darts.TypeJeu.PARTIE_501](#), et [com.lasalle84.mobile_darts.TypeJeu.PARTIE_501_DOUBLE_OUT](#).

```
00057      {
00058          switch (idModeJeu)
00059          {
00060              case PARTIE_301:
00061                  this.pointDepart = 301;
00062                  this.doubleOut = false;
00063                  break;
00064              case PARTIE_301_DOUBLE_OUT:
00065                  this.pointDepart = 301;
00066                  this.doubleOut = true;
00067                  break;
00068              case PARTIE_501:
00069                  this.pointDepart = 501;
00070                  this.doubleOut = false;
00071                  break;
00072              case PARTIE_501_DOUBLE_OUT:
00073                  this.pointDepart = 501;
00074                  this.doubleOut = true;
00075                  break;
00076          }
00077      }
```

9.10.3 Documentation des fonctions membres

9.10.3.1 `estDoubleOut()`

```
boolean com.lasalle84.mobile_darts.TypeJeu.estDoubleOut ( )
```

Définition à la ligne 89 du fichier [TypeJeu.java](#).

Références [com.lasalle84.mobile_darts.TypeJeu.doubleOut](#).

Référencé par [com.lasalle84.mobile_darts.Partie.demarrerPartie\(\)](#), et [com.lasalle84.mobile_darts.Joueur.retirerPoint\(\)](#).

```
00090      {
00091          return doubleOut;
00092      }
```

9.10.3.2 `getPointDepart()`

```
int com.lasalle84.mobile_darts.TypeJeu.getPointDepart ( )
```

Définition à la ligne 79 du fichier [TypeJeu.java](#).

Références [com.lasalle84.mobile_darts.TypeJeu.pointDepart](#).

Référencé par [com.lasalle84.mobile_darts.TypeJeu.getTypeJeu\(\)](#), et [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserPartie\(\)](#).

```
00080    {
00081        return pointDepart;
00082    }
```

9.10.3.3 `getTypeJeu()`

```
String com.lasalle84.mobile_darts.TypeJeu.getTypeJeu ( )
```

Définition à la ligne 99 du fichier [TypeJeu.java](#).

Références [com.lasalle84.mobile_darts.TypeJeu.getPointDepart\(\)](#).

Référencé par [com.lasalle84.mobile_darts.ActiviteJouerPartie.initialiserPartie\(\)](#).

```
00100    {
00101        String typeJeu = Integer.toString(this.getPointDepart());
00102        if (doubleOut)
00103        {
00104            typeJeu = typeJeu.concat("_DOUBLE_OUT");
00105        }
00106        Log.d(TAG, "getTypeJeu: " + typeJeu);
00107        return typeJeu;
00108    }
00109 }
```

9.10.3.4 `setDoubleOut()`

```
void com.lasalle84.mobile_darts.TypeJeu.setDoubleOut (
    boolean doubleOut )
```

Définition à la ligne 94 du fichier [TypeJeu.java](#).

Références [com.lasalle84.mobile_darts.TypeJeu.doubleOut](#).

```
00095    {
00096        this.doubleOut = doubleOut;
00097    }
```

9.10.3.5 `setPointDepart()`

```
void com.lasalle84.mobile_darts.TypeJeu.setPointDepart (
    int pointDepart )
```

Définition à la ligne 84 du fichier [TypeJeu.java](#).

Références [com.lasalle84.mobile_darts.TypeJeu.pointDepart](#).

```
00085    {
00086        this.pointDepart = pointDepart;
00087    }
```

9.10.4 Documentation des données membres

9.10.4.1 `doubleOut`

```
boolean com.lasalle84.mobile_darts.TypeJeu.doubleOut [private]
```

Attributs

Définition à la ligne 31 du fichier [TypeJeu.java](#).

Référéncé par [com.lasalle84.mobile_darts.TypeJeu.estDoubleOut\(\)](#), [com.lasalle84.mobile_darts.TypeJeu.setDoubleOut\(\)](#), et [com.lasalle84.mobile_darts.TypeJeu.TypeJeu\(\)](#).

9.10.4.2 `PARTIE_301`

```
final int com.lasalle84.mobile_darts.TypeJeu.PARTIE_301 = 0 [static], [private]
```

Constantes

Définition à la ligne 22 du fichier [TypeJeu.java](#).

Référéncé par [com.lasalle84.mobile_darts.TypeJeu.TypeJeu\(\)](#).

9.10.4.3 `PARTIE_301_DOUBLE_OUT`

```
final int com.lasalle84.mobile_darts.TypeJeu.PARTIE_301_DOUBLE_OUT = 1 [static], [private]
```

Définition à la ligne 23 du fichier [TypeJeu.java](#).

Référéncé par [com.lasalle84.mobile_darts.TypeJeu.TypeJeu\(\)](#).

9.10.4.4 PARTIE_501

```
final int com.lasalle84.mobile_darts.TypeJeu.PARTIE_501 = 2 [static], [private]
```

Définition à la ligne 24 du fichier [TypeJeu.java](#).

Référencé par [com.lasalle84.mobile_darts.TypeJeu.TypeJeu\(\)](#).

9.10.4.5 PARTIE_501_DOUBLE_OUT

```
final int com.lasalle84.mobile_darts.TypeJeu.PARTIE_501_DOUBLE_OUT = 3 [static], [private]
```

Définition à la ligne 25 du fichier [TypeJeu.java](#).

Référencé par [com.lasalle84.mobile_darts.TypeJeu.TypeJeu\(\)](#).

9.10.4.6 pointDepart

```
int com.lasalle84.mobile_darts.TypeJeu.pointDepart [private]
```

Définition à la ligne 32 du fichier [TypeJeu.java](#).

Référencé par [com.lasalle84.mobile_darts.TypeJeu.getPointDepart\(\)](#), [com.lasalle84.mobile_darts.TypeJeu.setPointDepart\(\)](#), et [com.lasalle84.mobile_darts.TypeJeu.TypeJeu\(\)](#).

9.10.4.7 TAG

```
final String com.lasalle84.mobile_darts.TypeJeu.TAG = "_TypeJeu_" [static], [private]
```

Définition à la ligne 26 du fichier [TypeJeu.java](#).

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

— [TypeJeu.java](#)

10 Documentation des fichiers

10.1 Référence du fichier [ActiviteFinPartie.java](#)

Déclaration de la classe [ActiviteJouerPartie](#).

Classes

— class [com.lasalle84.mobile_darts.ActiviteFinPartie](#)
L'activité [ActiviteFinPartie](#).

Paquetages

— package [com.lasalle84.mobile_darts](#)

10.1.1 Description détaillée

Déclaration de la classe `ActiviteJouerPartie`.

Auteur

LEFORT Théodore

LastChangedRevision

97

LastChangedDate

2021-05-28 11 :55 :20 +0200 (ven. 28 mai 2021)

Définition dans le fichier [ActiviteFinPartie.java](#).

10.2 ActiviteFinPartie.java

```
00001 package com.lasalle84.mobile_darts;
00002
00003 import androidx.annotation.RequiresApi;
00004 import androidx.appcompat.app.AppCompatActivity;
00005
00006 import android.graphics.Color;
00007 import android.os.Build;
00008 import android.os.Bundle;
00009 import android.util.Log;
00010 import android.view.View;
00011 import android.widget.AdapterView;
00012 import android.widget.Button;
00013 import android.widget.ListView;
00014 import android.widget.TextView;
00015
00016 import java.util.ArrayList;
00017 import java.util.Comparator;
00018
00019 import nl.dionsegiijn.konfetti.KonfettiView;
00020 import nl.dionsegiijn.konfetti.models.Shape;
00021 import nl.dionsegiijn.konfetti.models.Size;
00022
00036 public class ActiviteFinPartie extends AppCompatActivity implements View.OnClickListener
00037 {
00041     private final String TAG = "_ActiviteFinPartie_";
00045     private TextView affichageGagnant = null;
00046     private ListView classement = null;
00047     private Button boutonFin = null;
00048     private ArrayList<Joueur> lesJoueur = null;
00049     private String nomGagnant = null;
00050     private ArrayAdapter<String> adapteur = null;
00051
00058     @RequiresApi(api = Build.VERSION_CODES.N)
00059     @Override
00060     protected void onCreate(Bundle savedInstanceState)
00061     {
00062         super.onCreate(savedInstanceState);
00063         setContentView(R.layout.activity_activite_fin_partie);
00064         recupererWidgets();
00065         initialiserWidgets();
00066         recupererInformationPartie();
00067         trierLesJoueurs();
00068         afficherJoueurs();
00069         declancherConfettis();
00070     }
00071
00072     private void declancherConfettis()
00073     {
00074         final KonfettiView konfettiView = findViewById(R.id.viewKonfetti);
00075         konfettiView.build()
00076             .addColors(Color.BLUE, Color.GREEN, Color.RED)
```

```

00077         .setDirection(0.0, 359.0)
00078         .setSpeed(1f, 5f)
00079         .setFadeOutEnabled(true)
00080         .setTimeToLive(2000L)
00081         .addShapes(Shape.Square.INSTANCE, Shape.Circle.INSTANCE)
00082         .addSizes(new Size(12, 5f))
00083         .setPosition(-50f, konfettiView.getWidth() + 50f, -50f, -50f)
00084         .streamFor(500, 5000L);
00085     }
00086
00092     public void onClick(View element)
00093     {
00094         Log.d(TAG, "onClick() " + element.toString());
00095
00096         if(element == boutonFin)
00097         {
00098             super.finish();
00099         }
00100     }
00101
00105     @Override
00106     protected void onStart()
00107     {
00108         super.onStart();
00109         Log.d(TAG, "[onStart]");
00110     }
00111
00115     @Override
00116     protected void onResume()
00117     {
00118         super.onResume();
00119         Log.d(TAG, "[onResume]");
00120     }
00121
00125     @Override
00126     protected void onPause()
00127     {
00128         super.onPause();
00129         Log.d(TAG, "[onPause]");
00130     }
00131
00135     @Override
00136     protected void onStop()
00137     {
00138         super.onStop();
00139         Log.d(TAG, "[onStop]");
00140     }
00141
00145     @Override
00146     protected void onDestroy()
00147     {
00148         super.onDestroy();
00149         Log.d(TAG, "[onDestroy]");
00150     }
00151
00157     public void recupererWidgets()
00158     {
00159         Log.d(TAG, "recupererWidgets()");
00160         affichageGagnant = (TextView) findViewById(R.id.texteGagnant);
00161         classement = (ListView) findViewById(R.id.ResultaPartie);
00162         boutonFin = (Button) findViewById(R.id.boutonFin);
00163     }
00164
00171     public void initialiserWidgets()
00172     {
00173         Log.d(TAG, "initialiserWidgets()");
00174         boutonFin.setOnClickListener((View.OnClickListener) this);
00175     }
00176
00182     public void recupererInfomationPartie()
00183     {
00184         Log.d(TAG, "recupererInfomationPartie()");
00185         lesJoueur = (ArrayList<Joueur>) getIntent().getSerializableExtra("joueurs");
00186         nomGagnant = getIntent().getStringExtra("gagnant");
00187         Log.d(TAG, "Le nom du gagnant est" + nomGagnant);
00188     }
00189
00195     @RequiresApi(api = Build.VERSION_CODES.N)
00196     public void trierLesJoueurs()
00197     {
00198         Log.d(TAG, "trierLesJoueurs()");
00199         lesJoueur.sort(Comparator.comparingInt(Joueur::getScore));
00200     }
00201
00202
00208     public void afficherJoueurs()
00209     {
00210         Log.d(TAG, "afficherJoueurs()");
00211         adapteur = new ArrayAdapter<String>(this, android.R.layout.simple_spinner_item);
00212         for (int i = 0; i < lesJoueur.size(); i++)

```

```
00213     {
00214         adapteur.add(lesJoueur.get(i).getNom() + " : " +
lesJoueur.get(i).getScore());
00215     }
00216     classement.setAdapter(adapteur);
00217     String gagnant = nomGagnant + " a gagné la partie";
00218     affichageGagnant.setText(gagnant);
00219 }
00220
00221 }
```

10.3 Référence du fichier ActiviteJouerPartie.java

Déclaration de la classe ActiviteJouerPartie.

Classes

— class [com.lasalle84.mobile_darts.ActiviteJouerPartie](#)
L'activité [ActiviteJouerPartie](#).

Paquetages

— package [com.lasalle84.mobile_darts](#)

10.3.1 Description détaillée

Déclaration de la classe ActiviteJouerPartie.

Auteur

LEFORT Théodore

LastChangedRevision

123

LastChangedDate

2021-06-11 11 :48 :02 +0200 (ven. 11 juin 2021)

Définition dans le fichier [ActiviteJouerPartie.java](#).

10.4 ActiviteJouerPartie.java

```

00001 package com.lasalle84.mobile_darts;
00002
00003 import android.app.Activity;
00004 import android.bluetooth.BluetoothAdapter;
00005 import android.content.Intent;
00006 import android.os.Bundle;
00007 import android.os.Handler;
00008 import android.os.Message;
00009 import android.util.Log;
00010 import android.view.View;
00011 import android.widget.AdapterView;
00012 import android.widget.Button;
00013 import android.widget.ListView;
00014 import android.widget.TextView;
00015
00016 import androidx.appcompat.app.AppCompatActivity;
00017
00018 import java.io.Serializable;
00019 import java.util.ArrayList;
00020 import java.util.Iterator;
00021 import java.util.List;
00022
00035 public class ActiviteJouerPartie extends AppCompatActivity
00036 {
00040     private final static String TAG = "_ActiviteJouerPartie_";
00041     private final static int ACTIVITE_FIN_PARTIE = 0;
00042
00045     private Peripherique cibleDarts = null;
00046     private Peripherique ecranDarts = null;
00047     private Handler handler = null;
00048     private ArrayList<Joueur> mesJoueurs = null;
00049     private BluetoothAdapter bluetoothAdapter = null;
00050     private Partie maPartie = null;
00051     private TypeJeu typePartie = null;
00052     private ArrayAdapter<String> adapteur = null;
00053     private ListView listJoueur = null;
00054     private TextView affichageImpact = null;
00055     private Button boutonTirManque;
00056     private Button boutonLancerPartie;
00057
00064     @Override
00065     protected void onCreate(Bundle savedInstanceState)
00066     {
00067         Log.d(TAG, "onCreate()");
00068         super.onCreate(savedInstanceState);
00069         setContentView(R.layout.activity_activite_jouer_partie);
00070
00071         initialiserHandler();
00072         initialiserWidgets();
00073         recupererPeripheriques();
00074         chargerLesParametres();
00075         initialiserListeJoueur();
00076         maPartie = new Partie(mesJoueurs, typePartie, handler);
00077     }
00078
00082     @Override
00083     protected void onStart()
00084     {
00085         super.onStart();
00086         Log.d(TAG, "[onStart]");
00087     }
00088
00092     @Override
00093     protected void onResume()
00094     {
00095         super.onResume();
00096         Log.d(TAG, "[onResume]");
00097     }
00098
00102     @Override
00103     protected void onPause()
00104     {
00105         super.onPause();
00106         Log.d(TAG, "[onPause]");
00107     }
00108
00112     @Override
00113     protected void onStop()
00114     {
00115         super.onStop();
00116         Log.d(TAG, "[onStop]");
00117     }
00118
00122     @Override
00123     protected void onDestroy()
00124     {
00125         super.onDestroy();
00126         Log.d(TAG, "[onDestroy]");

```

```

00127     }
00128
00135     private void recupererPeripheriques()
00136     {
00137         Log.d(TAG, "recupererPeripheriques()");
00138
00139         cibleDarts = ActiviteLancerPartie.
00140         getPeripheriqueCibleDarts();
00141         ecranDarts = ActiviteLancerPartie.
00142         getPeripheriqueEcranDarts();
00143
00144         if(cibleDarts != null)
00145             cibleDarts.setHandlerActivite(handler);
00146         if(ecranDarts != null)
00147             ecranDarts.setHandlerActivite(handler);
00148     }
00149
00155     public void actualiserScoreIHM(String joueur, int score)
00156     {
00157         Log.d(TAG, "actualiserScoreIHM()");
00158         int idJoueur = 0;
00159
00160         for (int i = 0; i < mesJoueurs.size(); i++)
00161         {
00162             if (mesJoueurs.get(i).getNom().equals(joueur))
00163             {
00164                 idJoueur = i;
00165             }
00166         }
00167         mesJoueurs.get(idJoueur).setScore(score);
00168         String string = mesJoueurs.get(idJoueur).getNom() + ": " + mesJoueurs.get(idJoueur).getScore();
00169         adapteur.remove(adapteur.getItem(idJoueur));
00170         adapteur.insert(string, idJoueur);
00171     }
00172
00179     public void afficherJoueurActuel(String joueur)
00180     {
00181         Log.d(TAG, "afficherJoueurActuel()");
00182         String joueurActuel;
00183         for (int i = 0; i < mesJoueurs.size(); i++)
00184         {
00185             if (mesJoueurs.get(i).getNom().equals(joueur))
00186             {
00187                 joueurActuel = "-> " + mesJoueurs.get(i).getNom() + ": " + mesJoueurs.get(i).getScore();
00188             }
00189             else {
00190                 joueurActuel = mesJoueurs.get(i).getNom() + ": " + mesJoueurs.get(i).getScore();
00191             }
00192             adapteur.remove(adapteur.getItem(i));
00193             adapteur.insert(joueurActuel, i);
00194         }
00195         affichageImpact.setText("");
00196     }
00197
00205     public void afficherGagnant(String gagnant)
00206     {
00207         Log.d(TAG, "afficherGagnant() : Le nom du gagnant est : " + gagnant);
00208         Intent AfficherGagnant = new Intent(this, ActiviteFinPartie.class);
00209         List<Joueur> lesJoueur = new ArrayList<Joueur>();
00210         for (int i = 0; i < mesJoueurs.size(); i++)
00211         {
00212             if (mesJoueurs.get(i).getNom().equals(gagnant))
00213             {
00214                 AfficherGagnant.putExtra("gagnant", mesJoueurs.get(i).getNom());
00215                 lesJoueur.add(mesJoueurs.get(i));
00216             }
00217             else
00218             {
00219                 lesJoueur.add(mesJoueurs.get(i));
00220             }
00221         }
00222         AfficherGagnant.putExtra("joueurs", (Serializable) lesJoueur);
00223     }
00224     startActivityResult(AfficherGagnant, ACTIVITE_FIN_PARTIE);
00225     finish();
00226 }
00227
00237     public void afficherImpact(int Touche, int multiplicateur)
00238     {
00239         Log.d(TAG, "afficherImpact()");
00240         String multiplication = new String();
00241         switch (multiplicateur)
00242         {
00243             case 0:
00244                 multiplication = "MISS";
00245                 break;
00246             case 1:
00247                 multiplication = "Simple ";
00248                 break;
00249             case 2:

```

```

00250         multiplication = "Double ";
00251         break;
00252     case 3:
00253         multiplication = "Triple ";
00254         break;
00255     }
00256     if (multiplication == "MISS")
00257     {
00258         affichageImpact.setText(affichageImpact.getText().toString().concat("MISS "));
00259     }
00260     else
00261     {
00262         affichageImpact.setText(affichageImpact.getText().toString().concat(multiplication + Integer.
toString(Touche) + " "));
00263     }
00264 }
00265
00271 public void chargerLesParametres()
00272 {
00273     mesJoueurs = new ArrayList<Joueur>();
00274     mesJoueurs = (ArrayList<Joueur>) getIntent().getSerializableExtra("LesJoueurs");
00275     Iterator<Joueur> it = mesJoueurs.iterator();
00276
00277     while(it.hasNext())
00278     {
00279         Log.d(TAG, "le joueur "+ it.next().getNom() + "est chargé");
00280     }
00281     typePartie = new TypeJeu(getIntent().getIntExtra("TypeMode",0));
00282 }
00283
00284 public void envoyerAquittement()
00285 {
00286     Log.d(TAG, "envoyerAquittement()");
00287     cibleDarts.envoyer("$DARTS;OK;" + "\\r\\n");
00288 }
00289
00293 private void initialiserWidgets()
00294 {
00295     Log.d(TAG,"initialiserWidgets()");
00296     boutonTirManque = (Button) findViewById(R.id.TirManque);
00297     boutonTirManque.setVisibility(View.INVISIBLE);
00298     boutonTirManque.setOnClickListener(view -> {
00299         Log.d(TAG, "[onClick] boutonTirManque");
00300         maPartie.cibleManquee();
00301     });
00302
00303     boutonLancerPartie = (Button) findViewById(R.id.LancerPartie);
00304     boutonLancerPartie.setOnClickListener(view -> {
00305         Log.d(TAG, "[onClick] boutonLancerPartie");
00306         initialiserPartie();
00307         new Thread()
00308         {
00309             @Override
00310             public void run() {
00311                 super.run();
00312                 maPartie.demarrerPartie();
00313             }
00314         }.start();
00315     boutonLancerPartie.setVisibility(View.INVISIBLE);
00316     boutonTirManque.setVisibility(View.VISIBLE);
00317 });
00318     listJoueur = (ListView) findViewById(R.id.MesJoueur);
00319     affichageImpact = (TextView) findViewById(R.id.Impact);
00320 }
00321
00325 private void initialiserHandler()
00326 {
00327     Log.d(TAG,"[initialiserHandler]");
00328     this.handler = new Handler(this.getMainLooper())
00329     {
00330         @Override
00331         public void handleMessage(Message msg)
00332         {
00333             super.handleMessage(msg);
00334             Bundle b = msg.getData();
00335             switch(msg.what)
00336             {
00337                 case Peripherique.CODE_RECEPTION:
00338                     if(b.getString("nom").equals(Peripherique.
NOM_CIBLE))
00339                     {
00340                         String donnees = b.getString("donnees");
00341                         Log.d(TAG,"[handleMessage] Nom : " + b.getString("nom") + " - Adresse : " +
b.getString("adresse") + " -> " + b.getString("donnees"));
00342                         maPartie.traiterTrame(donnees);
00343                     }
00344                     else if(b.getString("nom").equals(Peripherique.
NOM_ECRAN))
00345                     {
00346                         String donnees = b.getString("donnees");
00347                         Log.d(TAG,"[handleMessage] Nom : " + b.getString("nom") + " - Adresse : " +

```

```

    b.getString("adresse") + " -> " + b.getString("donnees"));
00348         maPartie.traiterTrame(donnees);
00349     }
00350     break;
00351     case Partie.SET_SCORE:
00352         Log.d(TAG, "[handleMessage] : SET_SCORE " + "Joueur: " + b.getString("joueur")
+ " Score: " + b.getInt("score"));
00353         actualiserScoreIHM(b.getString("joueur"), b.getInt("score"));
00354         break;
00355     case Partie.IMPACT:
00356         Log.d(TAG, "[handleMessage] : IMPACT " + "Joueur: " + b.getString("joueur") + "
Touche: " + b.getInt("Touche") + " Multiplicateur Touche: " + b.getInt("MultiplicateurTouche"));
00357         afficherImpact( b.getInt("Touche"),b.getInt("MultiplicateurTouche
"));
00358         if(ecranDarts != null)
00359         {
00360             ecranDarts.envoyer("$DARTS;HIT;" + b.getInt("Touche") + ";" + b.
getInt("MultiplicateurTouche") + ";" + b.getInt("ID") + "\r\n");
00361         }
00362         break;
00363     case Partie.JOUEUR_SUIVANT:
00364         Log.d(TAG, "[handleMessage] : JOUEUR_SUIVANT " + "Joueur: " + b.getString("
joueur"));
00365         afficherJoueurActuel(b.getString("joueur"));
00366         break;
00367     case Partie.ENVOYER_AQUITTEMENT:
00368         Log.d(TAG, "[handleMessage] : ENVOYER_AQUITTEMENT");
00369         envoyerAquittement();
00370         break;
00371     case Partie.GAGNANT:
00372         Log.d(TAG, "[handleMessage] : AFFICHER_GAGNANT");
00373         if(ecranDarts != null)
00374         {
00375             ecranDarts.envoyer("$DARTS;END;" + b.getString("gagnant") + "\r\n");
00376         }
00377         cibleDarts.envoyer("$DARTS;RESET\r\n");
00378         afficherGagnant(b.getString("gagnant"));
00379         break;
00380     }
00381 }
00382 };
00383 }
00384
00385 public void initialiserPartie() {
00386     Log.d(TAG, "initialiserPartie()");
00387     Iterator<Joueur> it = mesJoueurs.iterator();
00388     String trameNomJoueurs = new String();
00389     while(it.hasNext())
00390     {
00391         Joueur monJoueur = it.next();
00392         monJoueur.setScore(typePartie.getPointDepart());
00393         actualiserScoreIHM(monJoueur.getNom(),typePartie.
getPointDepart());
00394         trameNomJoueurs = trameNomJoueurs.concat(";"+monJoueur.getNom());
00395     }
00396     cibleDarts.envoyer("$DARTS;START;" + typePartie.getTypeJeu() + ";" + mesJoueurs.
size() + trameNomJoueurs + "\r\n");
00397     //Log.d(TAG,"$DARTS;START;" + typePartie.getTypeJeu() + ";" + mesJoueurs.size() + trameNomJoueurs +
00398     "\r\n");
00399     if(ecranDarts != null)
00400     {
00401         ecranDarts.envoyer("$DARTS;START;" + typePartie.getTypeJeu() + ";" +
mesJoueurs.size() + trameNomJoueurs + "\r\n");
00402     }
00403     Log.d(TAG,"Envoi de la trame à la cible et à l'écran");
00404 }
00405
00406 public void initialiserListeJoueur()
00407 {
00408     Log.d(TAG, "initialiserListeJoueur()");
00409     adapteur = new ArrayAdapter<String>(this,android.R.layout.simple_spinner_item);
00410     for (int i = 0; i < mesJoueurs.size(); i++)
00411     {
00412         adapteur.add(mesJoueurs.get(i).getNom());
00413     }
00414     listJoueur.setAdapter(adapteur);
00415 }
00416
00417 protected void onActivityResult(int requestCode, int resultCode, Intent data)
00418 {
00419     super.onActivityResult(requestCode, resultCode, data);
00420     Log.d(TAG, "[onActivityResult] Fin ActiviteFinPartie");
00421     if (ecranDarts != null)
00422     {
00423         ecranDarts.deconnecter();
00424     }
00425     cibleDarts.deconnecter();
00426     super.finish();
00427 }
00428 }
00429 }
00430 }

```


10.5 Référence du fichier `ActiviteLancerPartie.java`

Déclaration de la classe `ActiviteLancerPartie`.

Classes

- class `com.lasalle84.mobile_darts.ActiviteLancerPartie`
L'activité `ActiviteLancerPartie`.

Paquetages

- package `com.lasalle84.mobile_darts`

10.5.1 Description détaillée

Déclaration de la classe `ActiviteLancerPartie`.

Auteur

LEFORT Théodore

LastChangedRevision

123

LastChangedDate

2021-06-11 11 :48 :02 +0200 (ven. 11 juin 2021)

Définition dans le fichier `ActiviteLancerPartie.java`.

10.6 `ActiviteLancerPartie.java`

```
00001 package com.lasalle84.mobile_darts;
00002
00003 import android.app.Activity;
00004 import android.app.AlertDialog;
00005 import android.bluetooth.BluetoothAdapter;
00006 import android.bluetooth.BluetoothDevice;
00007 import android.content.DialogInterface;
00008 import android.content.Intent;
00009 import android.graphics.Color;
00010 import android.os.Bundle;
00011 import android.os.Handler;
00012 import android.os.Message;
00013 import android.util.Log;
00014 import android.view.Gravity;
00015 import android.view.View;
00016 import android.view.Window;
00017 import android.view.WindowManager;
00018 import android.widget.AdapterView;
00019 import android.widget.EditText;
00020 import android.widget.ImageButton;
00021 import android.widget.ListView;
00022 import android.widget.Spinner;
00023 import android.widget.Toast;
00024
00025 import androidx.appcompat.app.AppCompatActivity;
00026
00027 import java.util.ArrayList;
00028 import java.util.Arrays;
00029 import java.util.List;
00030 import java.util.Set;
00031 import java.util.Vector;
00032
00045 public class ActiviteLancerPartie extends AppCompatActivity
```

```

00046 {
00050     private final static String TAG = "_ActiviteLancerPartie_";
00051     private final static int REQUEST_CODE_ENABLE_BLUETOOTH = 0;
00052     // Différentes couleurs des boutons
00053     private final static int COULEUR_BOUTON_ROUGE = Color.parseColor("#ee5e5e");
00054     private final static int COULEUR_BOUTON_ORANGE = Color.parseColor("#f7bb31");
00055     private final static int COULEUR_BOUTON_GRIIS = Color.parseColor("#c0c5c6");
00056     private final static int COULEUR_BOUTON_VERT = Color.parseColor("#5eed7b");
00057
00061     private BluetoothAdapter bluetoothAdapter = null;
00062     public static Peripherique cibleDarts = null;
00063     public static Peripherique ecranDarts = null;
00064     private Handler handler = null;
00065     private List<String> nomsJoueur;
00066     private ArrayAdapter<String> adapteurNomJoueur;
00067
00071     private Spinner spinnerTypeJeu;
00072     private ListView listeJoueur;
00073     private ImageButton boutonCibleDarts;
00074     private ImageButton boutonEcranDarts;
00075     private EditText nomJoueur;
00076     private AlertDialog dialogueAjouterJoueur;
00077
00084     @Override
00085     protected void onCreate(Bundle savedInstanceState)
00086     {
00087         super.onCreate(savedInstanceState);
00088         setContentView(R.layout.activity_activite_lancer_partie);
00089         Log.d(TAG, "[onCreate]");
00090
00091         initialiserHandler();
00092
00093         creerListeJoueurs();
00094
00095         initialiserWidgets();
00096
00097         initialiserDialogue();
00098
00099         activerBluetooth();
00100     }
00101
00105     @Override
00106     protected void onStart()
00107     {
00108         super.onStart();
00109         Log.d(TAG, "[onStart]");
00110     }
00111
00115     @Override
00116     protected void onResume()
00117     {
00118         super.onResume();
00119         Log.d(TAG, "[onResume]");
00120     }
00121
00125     @Override
00126     protected void onPause()
00127     {
00128         super.onPause();
00129         Log.d(TAG, "[onPause]");
00130     }
00131
00135     @Override
00136     protected void onStop()
00137     {
00138         super.onStop();
00139         Log.d(TAG, "[onStop]");
00140     }
00141
00145     @Override
00146     protected void onDestroy()
00147     {
00148         super.onDestroy();
00149         Log.d(TAG, "[onDestroy]");
00150     }
00151
00155     private void initialiserHandler()
00156     {
00157         if(handler != null)
00158         {
00159             if(cibleDarts != null)
00160             {
00161                 Log.d(TAG, "initialiserHandler() cibleDarts " + handler);
00162                 cibleDarts.setHandlerActivite(handler);
00163             }
00164             if(ecranDarts != null)
00165             {
00166                 Log.d(TAG, "initialiserHandler() ecranDarts " + handler);
00167                 ecranDarts.setHandlerActivite(handler);
00168             }
00169             return;

```

```

00170     }
00171     Log.d(TAG, "initialiserHandler()");
00172
00173     this.handler = new Handler(this.getMainLooper())
00174     {
00175         @Override
00176         public void handleMessage(Message msg)
00177         {
00178             super.handleMessage(msg);
00179             Bundle b = msg.getData();
00180             Log.d(TAG, "ActiviteLancerPartie:handleMessage() Nom : " + b.getString("nom") + " - Adresse
: " + b.getString("adresse") + " -> " + msg.what);
00181
00182             switch(msg.what)
00183             {
00184                 case Peripherique.CODE_CONNEXION:
00185                     if(b.getString("nom").equals(Peripherique.
NOM_CIBLE))
00186                     {
00187                         boutonCibleDarts.setBackgroundColor(COULEUR_BOUTON_VERT);
00188                     }
00189                     else if(b.getString("nom").equals(Peripherique.
NOM_ECRAN))
00190                     {
00191                         boutonEcranDarts.setBackgroundColor(COULEUR_BOUTON_VERT);
00192                     }
00193                     break;
00194                 case Peripherique.CODE_DECONNEXION:
00195                     if(b.getString("nom").equals(Peripherique.
NOM_CIBLE))
00196                     {
00197                         boutonCibleDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00198                     }
00199                     else if(b.getString("nom").equals(Peripherique.
NOM_ECRAN))
00200                     {
00201                         boutonEcranDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00202                     }
00203                     break;
00204                 case Peripherique.CODE_ERREUR_ENVOYER:
00205                     break;
00206                 case Peripherique.CODE_ERREUR_CONNECTER:
00207                     break;
00208                 case Peripherique.CODE_ERREUR_DECONNECTER:
00209                     break;
00210             }
00211         }
00212     };
00213     Log.d(TAG, "initialiserHandler() ecranDarts " + handler);
00214
00215     if(cibleDarts != null)
00216     {
00217         Log.d(TAG, "initialiserHandler() cibleDarts " + handler);
00218         cibleDarts.setHandlerActivite(handler);
00219     }
00220     if(ecranDarts != null)
00221     {
00222         Log.d(TAG, "initialiserHandler() ecranDarts " + handler);
00223         ecranDarts.setHandlerActivite(handler);
00224     }
00225 }
00226
00227 private void creerListeTypeJeu()
00228 {
00229     Log.d(TAG, "creerListeTypeJeu()");
00230
00231     List<String> listeTypeJeu = new ArrayList<String>();
00232     listeTypeJeu.add("301");
00233     listeTypeJeu.add("301 Double Out");
00234     listeTypeJeu.add("501");
00235     listeTypeJeu.add("501 Double Out");
00236
00237     ArrayAdapter<String> adapteurTypeJeu = new ArrayAdapter<String>(this, R.layout.item_type_jeu, R.id.
texte_type_jeu, listeTypeJeu);
00238     adapteurTypeJeu.setDropDownViewResource(R.layout.spinner_dropdown_item);
00239     SpinnerTypeJeu.setAdapter(adapteurTypeJeu);
00240 }
00241
00242 private void creerListeJoueurs()
00243 {
00244     Log.d(TAG, "creerListeJoueurs()");
00245
00246     listeJoueur = (ListView) findViewById(R.id.ListeJoueur);
00247
00248     nomsJoueur = new ArrayList<String>();
00249
00250     adapteurNomJoueur = new ArrayAdapter<String>(this, R.layout.item_joueur, R.id.texte_item,
nomsJoueur);
00251
00252     listeJoueur.setAdapter(adapteurNomJoueur);
00253 }

```

```

00263
00264 private void initialiserDialogue()
00265 {
00266     AlertDialog.Builder builder = new AlertDialog.Builder(this);
00267
00268     builder.setTitle(R.string.title);
00269     View view = getLayoutInflater().inflate(R.layout.dialog_nouveau_joueur,null);
00270     nomJoueur = (EditText) view.findViewById(R.id.nomJoueur);
00271
00272     builder.setView(view);
00273
00274     builder.setPositiveButton(R.string.ok, new DialogInterface.OnClickListener() {
00275         public void onClick(DialogInterface dialog, int id) {
00276             Log.d(TAG, "onClick() nom joueur : " + nomJoueur.getText().toString());
00277             nomsJoueur.add(nomJoueur.getText().toString());
00278             adapteurNomJoueur.notifyDataSetChanged();
00279         }
00280     });
00281     builder.setNegativeButton(R.string.annuler, new DialogInterface.OnClickListener() {
00282         public void onClick(DialogInterface dialog, int id) {
00283             // User cancelled the dialog
00284         }
00285     });
00286
00287     dialogueAjouterJoueur = builder.create();
00288 }
00289
00293 private void initialiserWidgets()
00294 {
00295     Log.d(TAG, "initialiserWidgets()");
00296
00297     boutonCibleDarts = (ImageButton) findViewById(R.id.boutonCibleDarts);
00298     boutonCibleDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00299     boutonCibleDarts.setOnClickListener(view -> {
00300         Log.d(TAG, "onClick() boutonCibleDarts");
00301         if(rechercherCibleDARTS())
00302         {
00303             boutonCibleDarts.setBackgroundColor(COULEUR_BOUTON_ORANGE);
00304             gererConnexionCibleDARTS();
00305         }
00306         else
00307         {
00308             boutonCibleDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00309         }
00310     });
00311
00312     boutonEcranDarts = (ImageButton) findViewById(R.id.boutonEcranDarts);
00313     boutonEcranDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00314     boutonEcranDarts.setOnClickListener(v -> {
00315         Log.d(TAG, "onClick() boutonEcranDarts");
00316         if(rechercherEcranDARTS())
00317         {
00318             boutonEcranDarts.setBackgroundColor(COULEUR_BOUTON_ORANGE);
00319             gererConnexionEcranDARTS();
00320         }
00321         else
00322         {
00323             boutonEcranDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00324         }
00325     });
00326
00327     SpinnerTypeJeu = (Spinner) findViewById(R.id.spinnerTypeJeu);
00328     creerListeTypeJeu();
00329 }
00330
00334 public void activerBluetooth()
00335 {
00336     Log.d(TAG, "activerBluetooth()");
00337
00338     bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
00339     if (bluetoothAdapter == null)
00340     {
00341         Toast.makeText(getApplicationContext(), "Bluetooth non activé !", Toast.LENGTH_SHORT).show();
00342         boutonCibleDarts.setBackgroundColor(COULEUR_BOUTON_GRIIS);
00343         boutonEcranDarts.setBackgroundColor(COULEUR_BOUTON_GRIIS);
00344     }
00345     else
00346     {
00347         if (!bluetoothAdapter.isEnabled())
00348         {
00349             Toast.makeText(getApplicationContext(), "Bluetooth non activé !", Toast.LENGTH_SHORT).show(
00350
00351             Intent activeBlueTooth = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
00352             startActivityForResult(activeBlueTooth, REQUEST_CODE_ENABLE_BLUETOOTH);
00353         }
00354         else
00355         {
00356             Toast.makeText(getApplicationContext(), "Bluetooth activé", Toast.LENGTH_SHORT).show();
00357             boutonCibleDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);
00358             boutonEcranDarts.setBackgroundColor(COULEUR_BOUTON_ROUGE);

```

```

00359     }
00360 }
00361
00365 public void afficherDialogAjouterJoueur (View view)
00366 {
00367     Log.d(TAG, "afficherDialogAjouterJoueur()");
00368     nomJoueur.setText("");
00369     dialogueAjouterJoueur.show();
00370     placerDialogueEnHaut(dialogueAjouterJoueur.getWindow());
00371     getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_ADJUST_NOHING);
00372 }
00373
00377 private void placerDialogueEnHaut (Window window)
00378 {
00379     WindowManager.LayoutParams wlp = window.getAttributes();
00380     wlp.gravity = Gravity.BOTTOM;
00381     wlp.flags &= ~WindowManager.LayoutParams.FLAG_DIM_BEHIND;
00382     window.setAttributes(wlp);
00383 }
00384
00388 public void demarrerActiviteJouerPartie (View view)
00389 {
00390     Log.d(TAG, "demarrerActiviteJouerPartie()");
00391     if(cibleDarts.estConnecte())
00392     {
00393         LancerPartie();
00394     }
00395     else
00396     {
00397         Log.d(TAG, "Cible non connectée");
00398         return;
00399     }
00400 }
00401
00405 public boolean rechercherCibleDARTS()
00406 {
00407     if(cibleDarts != null)
00408         return true;
00409
00410     Log.d(TAG, "rechercherCibleDARTS()");
00411     Set<BluetoothDevice> devices;
00412     devices = bluetoothAdapter.getBondedDevices();
00413     for (BluetoothDevice blueDevice : devices)
00414     {
00415         if(blueDevice.getName().equals(Peripherique.NOM_CIBLE))
00416         {
00417             cibleDarts = new Peripherique(blueDevice, handler);
00418             return true;
00419         }
00420     }
00421
00422     return false;
00423 }
00424
00428 public boolean rechercherEcranDARTS()
00429 {
00430     if(ecranDarts != null)
00431         return true;
00432
00433     Log.d(TAG, "rechercherEcranDARTS()");
00434     Set<BluetoothDevice> devices;
00435     devices = bluetoothAdapter.getBondedDevices();
00436     for (BluetoothDevice blueDevice : devices)
00437     {
00438         Log.d(TAG, "Device = " + blueDevice.getName());
00439         if(blueDevice.getName().equals(Peripherique.NOM_ECRAN))
00440         {
00441             ecranDarts = new Peripherique(blueDevice, handler);
00442             return true;
00443         }
00444     }
00445
00446     return false;
00447 }
00448
00452 private void gererConnexionCibleDARTS()
00453 {
00454     if(cibleDarts != null)
00455     {
00456         Log.d(TAG, "gererConnexionCibleDARTS()");
00457         if(!cibleDarts.estConnecte())
00458             cibleDarts.connecter();
00459         else
00460             cibleDarts.deconnecter();
00461     }
00462 }
00463
00467 private void gererConnexionEcranDARTS()
00468 {
00469     if(ecranDarts != null)
00470     {

```

```

00471         Log.d(TAG, "gererConnexionEcranDARTS()");
00472         if(!ecranDarts.estConnecte())
00473             ekranDarts.connecter();
00474         else
00475             ekranDarts.deconnecter();
00476     }
00477 }
00478
00479 public static Peripherique getPeripheriqueCibleDarts()
00480 {
00481     return cibleDarts;
00482 }
00483
00484 public static Peripherique getPeripheriqueEcranDarts()
00485 {
00486     return ekranDarts;
00487 }
00488
00492 protected void onActivityResult(int requestCode, int resultCode, Intent data)
00493 {
00494     super.onActivityResult(requestCode, resultCode, data);
00495     Log.d(TAG, "onActivityResult() Fin ActiviteCreerJoueur");
00496     if(resultCode == Activity.RESULT_OK)
00497     {
00498         String nom = data.getStringExtra("retour");
00499         Log.d(TAG, "Nom du Joueur saisi" + nom);
00500         nomsJoueur.add(nom);
00501     }
00502 }
00503
00504
00509 public void LancerPartie()
00510 {
00511     Log.d(TAG, "LancerPartie()");
00512     Vector<Joueur> mesJoueurs = new Vector<>();
00513     if(nomsJoueur.size() == 0)
00514     {
00515         for (int i = 1; i <= 1; i++)
00516         {
00517             mesJoueurs.add(new Joueur("Joueur" + i));
00518             Log.d(TAG, "Joueur" + i + " = " + "Joueur" + i);
00519         }
00520     }
00521     else
00522     {
00523         for (int i = 0; i < nomsJoueur.size(); i++)
00524         {
00525             mesJoueurs.add(new Joueur(nomsJoueur.get(i)));
00526             Log.d(TAG, "Joueur" + i + " = " + nomsJoueur.get(i));
00527         }
00528     }
00529
00530     Intent intent = new Intent(ActiviteLancerPartie.this,
ActiviteJouerPartie.class);
00531     intent.putExtra("TypeMode", SpinnerTypeJeu.getSelectedItemPosition());
00532     Log.d(TAG, "TypeMode = " + SpinnerTypeJeu.getSelectedItemPosition());
00533     intent.putExtra("LesJoueurs", mesJoueurs);
00534     startActivityForResult(intent, 12);
00535     finish();
00536 }
00540 @Override
00541 public void finish()
00542 {
00543     Log.d(TAG, "[finish]");
00544     Intent intent = new Intent();
00545     setResult(RESULT_OK, intent);
00546     super.finish();
00547 }
00548 }

```

10.7 Référence du fichier IhmMobileDarts.java

Déclaration de la classe IhmMobileDarts.

Classes

— class [com.lasalle84.mobile_darts.IhmMobileDarts](#)
L'activité principale de l'application DARTS.

Paquetages

— package [com.lasalle84.mobile_darts](#)

10.7.1 Description détaillée

Déclaration de la classe `IhmMobileDarts`.

Auteur

LEFORT Théodore

LastChangedRevision

107

LastChangedDate

2021-06-05 20 :49 :53 +0200 (sam. 05 juin 2021)

Définition dans le fichier [IhmMobileDarts.java](#).

10.8 IhmMobileDarts.java

```
00001 package com.lasalle84.mobile_darts;
00002
00003 import android.app.Activity;
00004 import android.content.Intent;
00005 import android.os.Bundle;
00006 import android.util.Log;
00007 import android.view.View;
00008 import android.widget.ImageButton;
00009 import android.widget.TextView;
00010 import android.widget.Toast;
00011
00012 import androidx.appcompat.app.AppCompatActivity;
00013
00014 // Logcat :
00015     _IhmMobileDarts_|_ActiviteLancerPartie_|_ActiviteCreerJoueur_|_ActiviteJouerPartie_|_ActiviteFinPartie_|_Peripherique_|_TRecept
00016
00017 public class IhmMobileDarts extends AppCompatActivity
00018 {
00019     private static final String TAG = "_IhmMobileDarts_";
00020     public static final String EXTRA_MESSAGE = "com.lasalle84.mobile_darts.MESSAGE";
00021     private static final int ACTIVITE_LANCER_PARTIE = 1;
00022
00023     private TextView versionDarts;
00024
00025     @Override
00026     protected void onCreate(Bundle savedInstanceState)
00027     {
00028         super.onCreate(savedInstanceState);
00029         setContentView(R.layout.activity_main);
00030         Log.d(TAG, "[onCreate]");
00031
00032         versionDarts = (TextView) findViewById(R.id.versionDarts);
00033         versionDarts.setText("version " + BuildConfig.VERSION_NAME);
00034     }
00035
00036     @Override
00037     protected void onStart()
00038     {
00039         super.onStart();
00040         Log.d(TAG, "[onStart]");
00041     }
00042
00043     @Override
00044     protected void onResume()
00045     {
00046         super.onResume();
00047         Log.d(TAG, "[onResume]");
00048     }
00049
00050     @Override
00051     protected void onPause()
00052     {
00053         //
00054     }
00055 }
```

```

00084     {
00085         super.onPause();
00086         Log.d(TAG, "[onPause]");
00087     }
00088
00092     @Override
00093     protected void onStop()
00094     {
00095         super.onStop();
00096         Log.d(TAG, "[onStop]");
00097     }
00098
00102     @Override
00103     protected void onDestroy()
00104     {
00105         super.onDestroy();
00106         Log.d(TAG, "[onDestroy]");
00107     }
00108
00112     public void demarrerActiviteLancerPartie(View view)
00113     {
00114         Intent intent = new Intent(this, ActiviteLancerPartie.class);
00115         startActivityForResult(intent, ACTIVITE_LANCER_PARTIE);
00116     }
00117
00121     protected void onActivityResult(int requestCode, int resultCode, Intent data)
00122     {
00123         super.onActivityResult(requestCode, resultCode, data);
00124
00125         switch (requestCode)
00126         {
00127             case ACTIVITE_LANCER_PARTIE:
00128                 Log.d(TAG, "[onActivityResult] Fin ActiviteLancerPartie");
00129                 if(resultCode == Activity.RESULT_OK)
00130                 {
00131                     Log.d(TAG, "[onActivityResult] ActiviteLancerPartie RESULT_OK");
00132                 }
00133                 break;
00134             default:
00135                 Log.d(TAG, "[onActivityResult] requestCode : " + requestCode);
00136         }
00137     }
00138
00139     public void demarrerActiviteAfficherHistorique(View view)
00140     {
00141         Toast.makeText(getApplicationContext(), "Fonction non disponible", Toast.LENGTH_SHORT).show();
00142     }
00143 }

```

10.9 Référence du fichier Joueur.java

Déclaration de la classe Joueur.

Classes

- class [com.lasalle84.mobile_darts.Joueur](#)
La classe qui contient toute la gestion des joueurs.

Paquetages

- package [com.lasalle84.mobile_darts](#)

10.9.1 Description détaillée

Déclaration de la classe Joueur.

Auteur

LEFORT Théodore

LastChangedRevision

98

LastChangedDate

2021-05-28 12:18:19 +0200 (ven. 28 mai 2021)

Définition dans le fichier [Joueur.java](#).

10.10 Joueur.java

```

00001 package com.lasalle84.mobile_darts;
00002
00015 import android.util.Log;
00016
00017 import java.io.Serializable;
00018
00019 public class Joueur implements Serializable
00020 {
00024     private static final String TAG = "_Joueur_";
00025
00029     private String nom;
00030     private int score;
00031     private boolean aFini;
00032
00039     public Joueur(String nom)
00040     {
00041         this.nom = nom;
00042         this.score = 0;
00043         this.aFini = false;
00044     }
00045
00052     public String getNom()
00053     {
00054         return nom;
00055     }
00056
00063     public void setNom(String nom)
00064     {
00065         this.nom = nom;
00066     }
00067
00074     public int getScore()
00075     {
00076         return score;
00077     }
00078
00085     public void setScore(int score)
00086     {
00087         this.score = score;
00088     }
00089
00096     public boolean retirerPoint(int scoreLancer, Partie maPartie)
00097     {
00098
00099         if (this.score - scoreLancer >= 0)
00100         {
00101             this.score = this.score - scoreLancer;
00102             if (this.score == 1 && maPartie.getTypeJeu().estDoubleOut())
00103             {
00104                 Log.d(TAG, this.getNom() + " est éliminé");
00105                 aFini = true;
00106             }
00107             return true;
00108         }
00109         return false;
00110     }
00111
00118     public void setaFini(boolean aFini)
00119     {
00120         this.aFini = aFini;
00121     }
00122
00128     public boolean isaFini()
00129     {
00130         return aFini;
00131     }
00132
00133 }

```

10.11 Référence du fichier Partie.java

Déclaration de la classe Partie.

Classes

- class [com.lasalle84.mobile_darts.Partie](#)
La classe qui contient toute la gestion de partie.

Paquetages

— package [com.lasalle84.mobile_darts](#)

10.11.1 Description détaillée

Déclaration de la classe Partie.

Auteur

LEFORT Théodore

LastChangedRevision

123

LastChangedDate

2021-06-11 11 :48 :02 +0200 (ven. 11 juin 2021)

Définition dans le fichier [Partie.java](#).

10.12 Partie.java

```

00001 package com.lasalle84.mobile_darts;
00002
00003 import android.os.Bundle;
00004 import android.os.Handler;
00005 import android.os.Message;
00006 import android.util.Log;
00007
00008 import java.util.ArrayList;
00009 import java.util.Iterator;
00010
00011 import static android.os.SystemClock.sleep;
00012
00025 public class Partie
00026 {
00030     private final static String TAG = "_Partie_";
00031     public final static int NB_FLECHETTE = 3;
00032
00036     public final static int JOUEUR_SUIVANT = 3;
00037     public final static int SET_SCORE = 4;
00038     public final static int IMPACT = 5;
00039     public final static int GAGNANT = 6;
00040     public final static int ENVOYER_AQUITTEMENT = 7;
00041
00045     private int nbJoueurs;
00046     private int nbManche;
00047     private TypeJeu typeJeu;
00048     private ArrayList<Joueur> lesJoueurs;
00049     private int impact[];
00050     private boolean impactEstRecu;
00051     private boolean estFini = false;
00052     private Handler handlerUI = null;
00053     private boolean estDoubleImpact = false;
00054     private boolean estAuSuivant = false;
00055
00056     public Partie(ArrayList<Joueur> lesJoueurs, TypeJeu typeJeu, Handler handlerUI)
00057     {
00058         this.lesJoueurs = lesJoueurs;
00059         this.typeJeu = typeJeu;
00060         this.handlerUI = handlerUI;
00061         this.nbManche = 0;
00062         this.nbJoueurs = this.lesJoueurs.size();
00063         impact = new int[3];
00064     }
00065
00072     public void demarrerPartie() {
00073         Log.d(TAG, "demarrerPartie()");
00074         estAuSuivant = true;
00075         do
00076         {
00077             nbManche++;
00078             Iterator<Joueur> it = lesJoueurs.iterator();

```

```

00079         Log.d(TAG, "Manche numéro " + nbManche);
00080         while (it.hasNext())
00081         {
00082             Joueur monJoueur = it.next();
00083             envoyerGagnantIHM(monJoueur);
00084             Log.d(TAG, "C'est le tour de " + monJoueur.getNom());
00085             attendreJoueurSuivant();
00086             actualiserJoueurIHM(monJoueur.getNom());
00087             int pointVollee = 0;
00088
00089             for(int i = 0; i < NB_FLECHETTE; i++)
00090             {
00091                 if (monJoueur.isaFin())
00092                 {
00093                     envoyerGagnantIHM(monJoueur);
00094                     i = NB_FLECHETTE;
00095                     estFin = true;
00096                 }
00097                 else
00098                 {
00099                     impactEstRecu = false;
00100                     attendreImpact();
00101                     pointVollee += impact[0]*impact[1];
00102                     if(!monJoueur.retirerPoint(pointVollee, this))
00103                     {
00104                         i = NB_FLECHETTE;
00105                         pointVollee = 0;
00106                     }
00107                     else if (monJoueur.getScore() == 0 && !typeJeu.
estDoubleOut())
00108                     {
00109                         envoyerGagnantIHM(monJoueur);
00110                         i = NB_FLECHETTE;
00111                         estFin = true;
00112                     }
00113                     else if (monJoueur.getScore() == 0 && typeJeu.
estDoubleOut() && estDoubleImpact)
00114                     {
00115                         envoyerGagnantIHM(monJoueur);
00116                         i = NB_FLECHETTE;
00117                         estFin = true;
00118                     }
00119                 }
00120             }
00121             actualiserScoreIHM(monJoueur, monJoueur.
getScore());
00122         }
00123     }while (!estFin);
00124 }
00130 public void attendreImpact() {
00131     Log.d(TAG, "attendreImpact()");
00132     while (!impactEstRecu)
00133     {
00134         sleep(1000);
00135     }
00136     impactIHM(impact[0],impact[1],impact[2]);
00137
00138     if (impact[1] == 2)
00139     {
00140         estDoubleImpact = true;
00141     }
00142     else
00143     {
00144         estDoubleImpact = false;
00145     }
00146
00147     Log.d(TAG, "Valeur Touche" + impact[0]+ "; Multiplicateur Touche:" + impact[1] + "; ID Touche : " +
impact[2]);
00148 }
00149 }
00150 public void attendreJoueurSuivant()
00151 {
00152     Log.d(TAG, "attendreJoueurSuivant()");
00153     while (!estAuSuivant)
00154     {
00155         sleep(1000);
00156     }
00157     Log.d(TAG, "Au suivant" + estAuSuivant);
00158     estAuSuivant = false;
00159 }
00160
00168 public void actualiserScoreIHM(Joueur monJoueur, int score)
00169 {
00170     Log.d(TAG, "actualiserScoreIHM()");
00171     Message msg = Message.obtain();
00172     msg.what = SET_SCORE;
00173     Bundle b = new Bundle();
00174     b.putString("joueur", monJoueur.getNom());
00175     b.putInt("score", score);
00176     msg.setData(b);
00177     handlerUI.sendMessage(msg);

```

```

00178     }
00179
00186     public void actualiserJoueurIHM(String monJoueur)
00187     {
00188         Log.d(TAG, "actualiserJoueurIHM() ");
00189         Message msg = Message.obtain();
00190         msg.what = JOUEUR_SUIVANT;
00191         Bundle b = new Bundle();
00192         b.putString("joueur", monJoueur);
00193         msg.setData(b);
00194         handlerUI.sendMessage(msg);
00195     }
00196
00205     public void impactIHM(int Touche, int MultiplicateurTouche, int ID)
00206     {
00207         Log.d(TAG, "impactIHM() " + Touche + " - " + MultiplicateurTouche);
00208         Message msg = Message.obtain();
00209         msg.what = IMPACT;
00210         Bundle b = new Bundle();
00211         b.putInt("Touche", Touche);
00212         b.putInt("MultiplicateurTouche", MultiplicateurTouche);
00213         b.putInt("ID", ID);
00214         msg.setData(b);
00215         handlerUI.sendMessage(msg);
00216     }
00217
00224     public void envoyerAquittement()
00225     {
00226         Log.d(TAG, "envoyerAquittement() ");
00227         Message msg = Message.obtain();
00228         msg.what = ENVOYER_AQUITTEMENT;
00229         handlerUI.sendMessage(msg);
00230     }
00231
00238     public void envoyerGagnantIHM(Joueur monJoueur)
00239     {
00240         Log.d(TAG, "envoyerGagnantIHM() " + monJoueur.getNom());
00241         estFini = true;
00242         Message msg = Message.obtain();
00243         msg.what = GAGNANT;
00244         Bundle b = new Bundle();
00245         b.putString("gagnant", monJoueur.getNom());
00246         msg.setData(b);
00247         handlerUI.sendMessage(msg);
00248     }
00254     public void cibleManquee()
00255     {
00256         impact[0] = 0;
00257         impact[1] = 0;
00258         impactEstRecu = true;
00259     }
00265     public void traiterTrame(String trame)
00266     {
00267         Log.d(TAG, "traiterTrame() " + trame);
00268         trame = trame.replace("\r\n", "");
00269         String[] trameSectionnee = trame.split(";", 5);
00270         for(int i=0; i < trameSectionnee.length; ++i)
00271         {
00272             Log.d(TAG, "traiterTrame() trameSectionnee[" + i + "] = " + trameSectionnee[i]);
00273         }
00274
00275         switch (trameSectionnee[1])
00276         {
00277             case ("HIT"):
00278                 Log.d(TAG, "traiterTrame : HIT");
00279                 impact[0] = Integer.parseInt(trameSectionnee[2]);
00280                 impact[1] = Integer.parseInt(trameSectionnee[3]);
00281                 impact[2] = Integer.parseInt(trameSectionnee[4]);
00282                 envoyerAquittement();
00283                 impactEstRecu = true;
00284                 break;
00285             case ("NEXT"):
00286                 Log.d(TAG, "traiterTrame : NEXT");
00287                 estAuSuivant = true;
00288                 break;
00289         }
00290     }
00291
00292     public TypeJeu getTypeJeu()
00293     {
00294         return typeJeu;
00295     }
00296 }

```

10.13 Référence du fichier Peripherique.java

Déclaration de la classe Peripherique.

Classes

- class [com.lasalle84.mobile_darts.Peripherique](#)
La classe qui contient toute la gestion des périphériques.

Paquetages

- package [com.lasalle84.mobile_darts](#)

10.13.1 Description détaillée

Déclaration de la classe Peripherique.

Auteur

LEFORT Théodore

LastChangedRevision

123

LastChangedDate

2021-06-11 11 :48 :02 +0200 (ven. 11 juin 2021)

Définition dans le fichier [Peripherique.java](#).

10.14 Peripherique.java

```

00001 package com.lasalle84.mobile_darts;
00002
00003 import android.bluetooth.BluetoothDevice;
00004 import android.bluetooth.BluetoothSocket;
00005 import android.os.Bundle;
00006 import android.os.Handler;
00007 import android.os.Message;
00008 import android.util.Log;
00009
00010 import java.io.IOException;
00011 import java.io.InputStream;
00012 import java.io.OutputStream;
00013 import java.util.UUID;
00014 import java.util.concurrent.locks.ReentrantLock;
00015
00028 public class Peripherique extends Thread
00029 {
00033     private final static String TAG = "_Peripherique_";
00034     public final static String NOM_CIBLE ="cible-darts 2021";
00035     public final static String NOM_ECRAN ="ecran-darts";
00036     public final static int CODE_CONNEXION = 0;
00037     public final static int CODE_RECEPTION = 1;
00038     public final static int CODE_DECONNEXION = 2;
00039     public final static int CODE_ERREUR_ENVOYER = -1;
00040     public final static int CODE_ERREUR_RECEVOIR = -2;
00041     public final static int CODE_ERREUR_CONNECTER = -3;
00042     public final static int CODE_ERREUR_DECONNECTER = -4;
00043
00047     private BluetoothDevice device;
00048     private String nom;
00049     private String adresse;
00050     private Handler handlerActivite = null;
00051     private BluetoothSocket socket;
00052     private InputStream receiveStream = null;
00053     private OutputStream sendStream = null;
00054     private TReception reception;
00055     private final ReentrantLock mutex = new ReentrantLock();
00056
00062     public Peripherique(BluetoothDevice device, Handler handler)
00063     {
00064         Log.d(TAG, "[Peripherique] " + device.getName() + " : " + device.getAddress());

```

```

00065         if(device != null)
00066         {
00067             this.device = device;
00068             this.nom = device.getName();
00069             this.adresse = device.getAddress();
00070             setHandlerActivite(handler);
00071         }
00072         else
00073         {
00074             this.device = device;
00075             this.nom = "Aucun";
00076             this.adresse = "";
00077             setHandlerActivite(handler);
00078         }
00079
00080         try
00081         {
00082             assert device != null;
00083             socket = device.createRfcommSocketToServiceRecord(UUID.fromString("
00001101-0000-1000-8000-00805F9B34FB"));
00084             receiveStream = socket.getInputStream();
00085             sendStream = socket.getOutputStream();
00086         }
00087         catch (IOException e)
00088         {
00089             e.printStackTrace();
00090             socket = null;
00091         }
00092
00093         if(socket != null)
00094             reception = new TReception(this, handlerActivite, receiveStream);
00095     }
00096
00102     public String getNom()
00103     {
00104         return nom;
00105     }
00106
00112     public String getAdresse()
00113     {
00114         return adresse;
00115     }
00116
00122     public boolean estConnecte()
00123     {
00124         if(device == null | socket == null)
00125             return false;
00126         return socket.isConnected();
00127     }
00128
00132     public void setHandlerActivite(Handler handler)
00133     {
00134         Log.d(TAG, "[setHandlerActivite] " + handler);
00135         mutex.lock();
00136         this.handlerActivite = handler;
00137         if(reception != null)
00138             reception.setHandlerActivite(handler);
00139         mutex.unlock();
00140     }
00141
00146     public void connecter()
00147     {
00148         Log.d(TAG, "[connecter] " + device.getName() + " : " + device.getAddress());
00149         new Thread()
00150         {
00151             @Override public void run()
00152             {
00153                 try
00154                 {
00155                     socket.connect();
00156
00157                     envoyerMessageActivite(CODE_CONNEXION);
00158
00159                     // on démarre le thread de réception
00160                     reception.start();
00161                     Log.d(TAG, "connexion reussie " + getNom());
00162                 }
00163                 catch (IOException e)
00164                 {
00165                     e.printStackTrace();
00166                     Log.d(TAG, "erreur connexion " + getNom());
00167                     envoyerMessageActivite(CODE_ERREUR_CONNECTER);
00168                 }
00169             }
00170         }.start();
00171     }
00172
00173
00178     public void deconnecter()
00179     {
00180         Log.d(TAG, "[deconnecter] " + device.getName() + " : " + device.getAddress());

```

```

00181         new Thread()
00182         {
00183             @Override public void run()
00184             {
00185                 try
00186                 {
00187                     reception.arreter();
00188                     socket.close();
00189                     envoyerMessageActivite(CODE_DECONNEXION);
00190                     Log.d(TAG, "déconnexion reussie " + getNom());
00191                 }
00192                 catch (IOException e)
00193                 {
00194                     e.printStackTrace();
00195                     envoyerMessageActivite(CODE_ERREUR_DECONNECTER);
00196                 }
00197             }
00198         }.start();
00199     }
00200
00205     public void envoyer(final String data)
00206     {
00207         Log.d(TAG, "[envoyer] " + device.getName() + " : " + device.getAddress());
00208         if(socket == null)
00209         {
00210             return;
00211         }
00212         new Thread()
00213         {
00214             @Override public void run()
00215             {
00216                 try
00217                 {
00218                     if(socket.isConnected())
00219                     {
00220                         sendStream.write(data.getBytes());
00221                         sendStream.flush();
00222                         Log.d(TAG, "[envoyer] trame : " + data);
00223                     }
00224                 }
00225                 catch (IOException e)
00226                 {
00227                     e.printStackTrace();
00228                     Log.d(TAG, "Erreur socket write : " + getNom());
00229                     envoyerMessageActivite(CODE_ERREUR_ENVOYER);
00230                 }
00231             }
00232         }.start();
00233     }
00234
00235     private void envoyerMessageActivite(int what)
00236     {
00237         if(handlerActivite == null)
00238             return;
00239         Message msg = Message.obtain();
00240         msg.what = what;
00241         Bundle b = new Bundle();
00242         b.putString("nom", getNom());
00243         b.putString("adresse", getAdresse());
00244         msg.setData(b);
00245         mutex.lock();
00246         handlerActivite.sendMessage(msg);
00247         mutex.unlock();
00248     }
00249 }
00250
00253 }

```

10.15 Référence du fichier Planification.md

10.16 Planification.md

```

00001 ## Module de gestion de partie
00002
00003 Les joueurs paramètrent et lancent la partie à partir d'une application sur un terminal mobile (sous
00004     Android).
00005
00006 ### Fonctionnalités nécessaires au fonctionnement du module
00007
00007 - Saisir les noms des joueurs.
00008
00009 - Paramétrer une partie :
00010     - le type de jeu :
00011         - 501 double out
00012         - 301 double out

```

```

00013         - ...
00014         - le nombre de joueurs.
00015         - le nombre de manches gagnantes.
00016
00017 - Lancer la partie.
00018
00019 - Gérer et visualiser le déroulement de la partie :
00020         - changer automatiquement de joueur
00021         - affichage des points
00022
00023 - Communiquer avec les modules en Bluetooth
00024
00025 ### Tâches devant être accomplies au cours du projet
00026
00027 - Création d'une base de données fonctionnelle et complétée
00028 - Création d'un joueur
00029 - Paramétrage d'une partie
00030 - Lancement et gestion d'une partie
00031 - Assurer la liaison bluetooth et la visibilité de la connexion aux modules
00032 - Transmission des informations de la partie aux modules
00033 - Déploiement de l'application mobile
00034
00035 ### Production attendue au cours du projet
00036
00037 - Une application informatique fonctionnelle
00038 - Un modèle UML complet de la partie à développer
00039 - Un code source commenté de l'application
00040 - Les documentations associés au module
00041
00042 ### Planification des itérations
00043
00044 |           Iteration 1           |           Iteration 2           |           Iteration 3
00045 |
00046 |:-----:|:-----:|:-----:|
00047 | Créer le squelette de l'application | Lancer une partie | Consulter l'historique des parties des
    | joueurs |
00048 | Créer un profil de joueur           | Dialoguer avec les modules           | Enregistrer les données des
    | manches |
00049 |                                     | Gérer le déroulement d'une partie   |
    | Purger les parties                 |                                     |

```

10.17 Référence du fichier Protocole.md

10.18 Protocole.md

```

00001 # Protocole DARTS
00002
00003 Version : 1.0
00004
00005 Avis : validé le 11/06/2021 par Thierry VAIRA
00006
00007 ## Format général des trames
00008
00009 Type de contenu : Caractères ASCII
00010
00011 Délimiteurs
00012
00013 - Début : '$DARTS'
00014 - Fin : '\r\n'
00015 - Champ de début : ';'
00016
00017 Liste des trames
00018
00019 - Trame de début : 'START'
00020 - Trame de touche : 'HIT'
00021 - Trame de joueur suivant : 'NEXT'
00022 - Trame de fin : 'END'
00023 - Trame de réinitialisation : 'RESET'
00024 - Trame d'acquiescement : 'OK'
00025
00026 Sens des trames
00027
00028 - Émetteur : Application mobile Darts, Module cible Darts.
00029 - Récepteur : ihm Darts, Application mobile Darts, Module cible Darts.
00030
00031 Voir diagramme des échanges
00032
00033 ## Format détaillé des trames
00034
00035 - Trame début :
00036
00037     Format : '$DARTS;START;TYPE_PARTIE;NB_JOUEURS[;NOMS]\r\n'
00038     Exemple : '$DARTS;START;501;2;Billy;Robert\r\n'

```



```

00039
00040 Types de partie :
00041     '301'
00042     '301_DOUBLE_OUT'
00043     '501'
00044     '501_DOUBLE_OUT'
00045
00046 _Remarque :_ les champs entre crochets '[' sont optionnels.
00047
00048 - Trame de touche :
00049
00050     Format : '$DARTS;HIT;valeurTouche;multiplicateurTouche;ID\r\n'
00051
00052 Les valeurs possibles pour valeurTouche sont :
00053
00054 - Les secteurs 1 à 20 ;
00055 - Le centre de la cible (la bulle ou Bull's Eye) ;
00056 - Les valeurs possibles pour multiplicateurTouche sont : 1, 2 (double) et 3 (triple)
00057 - Les valeurs possibles pour ID sont : 0 à 2
00058
00059 Exemples :
00060
00061 ```
00062 $DART;HIT;25;2;0\r\n -> Fléchette bullseye (50 points)
00063 $DART;HIT;25;1;0\r\n -> Fléchette demi-bulle (25 points)
00064
00065 Les simples :
00066 $DART;HIT;1;1;0\r\n -> 1 pts
00067 ...
00068 $DART;HIT;20;1;0\r\n -> 20 pts
00069 $DART;HIT;25;1;0\r\n -> 20 pts
00070
00071 Les doubles :
00072 $DART;HIT;1;2;0\r\n -> 2 pts
00073 ...
00074 $DART;HIT;20;2;0\r\n
00075 $DART;HIT;25;2;0\r\n
00076
00077 $DART;HIT;20;2;0\r\n -> 40 pts
00078
00079 Les triples :
00080 $DART;HIT;1;3;0\r\n -> 3 pts
00081 ...
00082 $DART;HIT;20;3;0\r\n -> 60 pts
00083 ```
00084
00085 - Trame d'acquiescement :
00086
00087     Format : '$DARTS;OK\r\n'
00088
00089 _Remarque :_ Les trames de touche sont acquittées avec les modules cible et mobile.
00090
00091 - Trame de réinitialisation :
00092
00093     Format : '$DARTS;RESET\r\n'
00094
00095 _Remarque :_ La trame 'RESET' permet de réinitialiser le module cible en cas d'erreur ou à la fin
    d'une partie. Cela permet d'annuler une partie en cours.
00096
00097 - Trame de fin :
00098
00099     Format : '$DARTS;END;(LE GAGNANT)\r\n'
00100
00101 - Trame de joueur suivant :
00102
00103     Format : '$DARTS;NEXT\r\n'
00104
00105
00106 ## Echanges
00107
00108 \image html exchanges.png

```

10.19 Référence du fichier README.md

10.20 README.md

```

00001 \mainpage Application Android
00002
00003 \tableofcontents
00004
00005 \section section_tdm Table des matières
00006 - \ref page_README
00007 - \ref page_about
00008 - \ref page_licence

```

```

00009
00010 \image html Screenshot_DARTS-1.png
00011
00012 \section section_infos Informations
00013
00014 \author
00015 \date 2021
00016 \version 1.1
00017 \see https://svn.riouxsvn.com/
00018
00019
00020 \page page_README README
00021
00022 [TOC]
00023
00024 # Mobile-DARTS {#projet}
00025
00026 ## Présentation {#presentation}
00027
00028 Réalisé en Java pour Android
00029
00030 \image html Screenshot_DARTS-1.png
00031
00032 \image html Screenshot_DARTS-2.png
00033
00034 ## Informations {#informations}
00035
00036 \author
00037 \date 2021
00038 \version 1.1
00039 \see https://svn.riouxsvn.com/
00040
00041
00042 \page page_about A propos
00043
00044 \author
00045 \date 2021
00046 \version 1.1
00047 \see https://svn.riouxsvn.com/
00048
00049
00050 \page page_licence Licence GPL
00051
00052 This program is free software; you can redistribute it and/or modify
00053 it under the terms of the GNU General Public License as published by
00054 the Free Software Foundation; either version 2 of the License, or
00055 (at your option) any later version.
00056
00057 This program is distributed in the hope that it will be useful,
00058 but WITHOUT ANY WARRANTY; without even the implied warranty of
00059 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00060 GNU General Public License for more details.
00061
00062 You should have received a copy of the GNU General Public License
00063 along with this program; if not, write to the Free Software
00064 Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

```

10.21 Référence du fichier TReception.java

Déclaration de la classe TReception.

Classes

- class [com.lasalle84.mobile_darts.TReception](#)
La classe qui contient toute la gestion des thread de reception et envoi de trames.

Paquetages

- package [com.lasalle84.mobile_darts](#)

10.21.1 Description détaillée

Déclaration de la classe TReception.

Auteur

LEFORT Théodore

LastChangedRevision

98

LastChangedDate

2021-05-28 12 :18 :19 +0200 (ven. 28 mai 2021)

Définition dans le fichier [TReception.java](#).

10.22 TReception.java

```

00001 package com.lasalle84.mobile_darts;
00002
00003 import android.os.Bundle;
00004 import android.os.Handler;
00005 import android.os.Message;
00006 import android.util.Log;
00007
00008 import java.io.BufferedReader;
00009 import java.io.IOException;
00010 import java.io.InputStream;
00011 import java.io.InputStreamReader;
00012 import java.util.concurrent.locks.ReentrantLock;
00013
00026 public class TReception extends Thread
00027 {
00028     private final String TAG = "_TReception_";
00029
00030     private Peripherique peripherique;
00031     private Handler handler;
00032     private boolean estFini;
00033     private InputStream receiveStream;
00034     private final ReentrantLock mutex = new ReentrantLock();
00035
00036     TReception(Peripherique peripherique, Handler handler, InputStream flux)
00037     {
00038         Log.d(TAG, "[TReception] " + peripherique.getNom() + " : " + peripherique.
00039             getAdresse());
00040         this.peripherique = peripherique;
00041         setHandlerActivite(handler);
00042         estFini = false;
00043         receiveStream = flux;
00044     }
00045
00046     @Override
00051     public void run()
00052     {
00053         Log.d(TAG, "[run] début " + peripherique.getNom() + " : " + peripherique.
00054             getAdresse());
00055         BufferedReader reception = new BufferedReader(new InputStreamReader(receiveStream));
00056         while(!estFini)
00057         {
00058             try
00059             {
00060                 String trame = "";
00061                 if(reception.ready())
00062                 {
00063                     trame = reception.readLine();
00064                 }
00065                 if(trame.length() > 0)
00066                 {
00067                     Log.d(TAG, "[run] trame reçue : " + trame);
00068                     Message msg = Message.obtain();
00069                     msg.what = Peripherique.CODE_RECEPTION;
00070                     Bundle b = new Bundle();
00071                     b.putString("nom", peripherique.getNom());
00072                     b.putString("adresse", peripherique.getAdresse());
00073                     b.putString("donnees", trame);
00074                     msg.setData(b);

```

```

00074         handler.sendMessage(msg);
00075     }
00076 }
00077 catch (IOException e)
00078 {
00079     e.printStackTrace();
00080     Log.d(TAG, "Erreur socket read : " + peripherique.getNom());
00081     Message msg = Message.obtain();
00082     msg.what = Peripherique.CODE_ERREUR_RECEVOIR;
00083     Bundle b = new Bundle();
00084     b.putString("nom", peripherique.getNom());
00085     b.putString("adresse", peripherique.getAdresse());
00086     msg.setData(b);
00087     handler.sendMessage(msg);
00088 }
00089 try
00090 {
00091     Thread.sleep(250);
00092 }
00093 catch (InterruptedException e)
00094 {
00095     e.printStackTrace();
00096 }
00097 }
00098 Message msg = Message.obtain();
00099 Bundle b = new Bundle();
00100 b.putString("nom", peripherique.getNom());
00101 b.putString("adresse", peripherique.getAdresse());
00102 b.putInt("etat", Peripherique.CODE_DECONNEXION);
00103 b.putString("donnees", "");
00104 msg.setData(b);
00105 handler.sendMessage(msg);
00106
00107 Log.d(TAG, "[run] fin " + peripherique.getNom() + " : " + peripherique.
getAdresse());
00108 }
00109
00114 public void arreter()
00115 {
00116     Log.d(TAG, "[arreter] " + peripherique.getNom() + " : " + peripherique.
getAdresse());
00117     if(!estFini)
00118     {
00119         estFini = true;
00120     }
00121     try
00122     {
00123         Thread.sleep(250);
00124     }
00125     catch (InterruptedException e)
00126     {
00127         e.printStackTrace();
00128     }
00129 }
00130
00131 public void setHandlerActivite(Handler handler)
00132 {
00133     Log.d(TAG, "[setHandlerActivite] " + handler);
00134     mutex.lock();
00135     this.handler = handler;
00136     mutex.unlock();
00137 }
00138 }

```

10.23 Référence du fichier TypeJeu.java

Déclaration de la classe TypeJeu.

Classes

- class [com.lasalle84.mobile_darts.TypeJeu](#)
La classe qui contient toute la gestion des types de parties.

Paquetages

- package [com.lasalle84.mobile_darts](#)

10.23.1 Description détaillée

Déclaration de la classe TypeJeu.

Auteur

LEFORT Théodore

LastChangedRevision

98

LastChangedDate

2021-05-28 12 :18 :19 +0200 (ven. 28 mai 2021)

Définition dans le fichier [TypeJeu.java](#).

10.24 TypeJeu.java

```

00001 package com.lasalle84.mobile_darts;
00002
00003 import android.util.Log;
00004
00017 public class TypeJeu
00018 {
00022     private final static int PARTIE_301 = 0;
00023     private final static int PARTIE_301_DOUBLE_OUT = 1;
00024     private final static int PARTIE_501 = 2;
00025     private final static int PARTIE_501_DOUBLE_OUT = 3;
00026     private final static String TAG = "_TypeJeu_";
00027
00031     private boolean doubleOut;
00032     private int pointDepart;
00033
00039     public TypeJeu()
00040     {
00041         this.pointDepart = 501;
00042         this.doubleOut = true;
00043     }
00044
00050     public TypeJeu(int pointDepart, boolean doubleOut)
00051     {
00052         this.pointDepart = pointDepart;
00053         this.doubleOut = doubleOut;
00054     }
00055
00056     public TypeJeu(int idModeJeu)
00057     {
00058         switch (idModeJeu)
00059         {
00060             case PARTIE_301:
00061                 this.pointDepart = 301;
00062                 this.doubleOut = false;
00063                 break;
00064             case PARTIE_301_DOUBLE_OUT:
00065                 this.pointDepart = 301;
00066                 this.doubleOut = true;
00067                 break;
00068             case PARTIE_501:
00069                 this.pointDepart = 501;
00070                 this.doubleOut = false;
00071                 break;
00072             case PARTIE_501_DOUBLE_OUT:
00073                 this.pointDepart = 501;
00074                 this.doubleOut = true;
00075                 break;
00076         }
00077     }
00078
00079     public int getPointDepart ()
00080     {
00081         return pointDepart;
00082     }
00083
00084     public void setPointDepart(int pointDepart)
00085     {
00086         this.pointDepart = pointDepart;

```

```
00087     }
00088
00089     public boolean estDoubleOut ()
00090     {
00091         return doubleOut;
00092     }
00093
00094     public void setDoubleOut (boolean doubleOut)
00095     {
00096         this.doubleOut = doubleOut;
00097     }
00098
00099     public String getTypeJeu ()
00100     {
00101         String typeJeu = Integer.toString(this.getPointDepart());
00102         if (doubleOut)
00103         {
00104             typeJeu = typeJeu.concat("_DOUBLE_OUT");
00105         }
00106         Log.d(TAG, "getTypeJeu: " + typeJeu);
00107         return typeJeu;
00108     }
00109 }
00110 }
```


Index

ACTIVITE_FIN_PARTIE

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 24

ACTIVITE_LANCER_PARTIE

com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 47

aFini

com : :lasalle84 : :mobile_darts : :Joueur, 52

activerBluetooth

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 30

ActiviteFinPartie.java, 86, 87

ActiviteJouerPartie.java, 89, 90

ActiviteLancerPartie.java, 94

actualiserJoueurIHM

com : :lasalle84 : :mobile_darts : :Partie, 56

actualiserScoreIHM

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 17

com : :lasalle84 : :mobile_darts : :Partie, 56

adapteur

com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 12

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 25

adapteurNomJoueur

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 40

adresse

com : :lasalle84 : :mobile_darts : :Peripherique, 72

affichageGagnant

com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 12

affichageImpact

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 25

afficherDialogAjouterJoueur

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 30

afficherGagnant

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 17

afficherImpact

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 17

afficherJoueurActuel

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 18

afficherJoueurs

com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 8

arreter

com : :lasalle84 : :mobile_darts : :TReception, 78

attendreImpact

com : :lasalle84 : :mobile_darts : :Partie, 57

attendreJoueurSuivant

com : :lasalle84 : :mobile_darts : :Partie, 57

bluetoothAdapter

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 25

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 40

boutonCibleDarts

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 40

boutonEcranDarts

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 40

boutonFin

com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 13

boutonLancerPartie

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 25

boutonTirManque

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 25

CODE_CONNEXION

com : :lasalle84 : :mobile_darts : :Peripherique, 72

CODE_DECONNEXION

com : :lasalle84 : :mobile_darts : :Peripherique, 72

CODE_ERREUR_CONNECTER

com : :lasalle84 : :mobile_darts : :Peripherique, 72

CODE_ERREUR_DECONNECTER

com : :lasalle84 : :mobile_darts : :Peripherique, 73

CODE_ERREUR_ENVOYER

com : :lasalle84 : :mobile_darts : :Peripherique, 73

CODE_ERREUR_RECEVOIR

com : :lasalle84 : :mobile_darts : :Peripherique, 73

CODE_RECEPTION

com : :lasalle84 : :mobile_darts : :Peripherique, 73

COULEUR_BOUTON_GRIS

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 41

COULEUR_BOUTON_ORANGE

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 41

COULEUR_BOUTON_ROUGE

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 41

COULEUR_BOUTON_VERT

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 41

chargerLesParametres

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 18

cibleDarts

com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 25

com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 41

cibleManquee

com : :lasalle84 : :mobile_darts : :Partie, 58

classement

com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 13

com, 6

com.lasalle84, 6

com.lasalle84.mobile_darts, 6

com.lasalle84.mobile_darts.ActiviteFinPartie, 6

com.lasalle84.mobile_darts.ActiviteJouerPartie, 14

com.lasalle84.mobile_darts.ActiviteLancerPartie, 27

com.lasalle84.mobile_darts.IhmMobileDarts, 43

com.lasalle84.mobile_darts.Joueur, 48

com.lasalle84.mobile_darts.Partie, 53

com.lasalle84.mobile_darts.Peripherique, 65

com.lasalle84.mobile_darts.TReception, 76

com.lasalle84.mobile_darts.TypeJeu, 81

com : :lasalle84 : :mobile_darts : :ActiviteFinPartie

adapteur, 12

affichageGagnant, 12

afficherJoueurs, 8

boutonFin, 13

classement, 13

declancherConfettis, 8

initialiserWidgets, 9

lesJoueur, 13

nomGagnant, 13

onClick, 9

onCreate, 9

onDestroy, 10

onPause, 10

onResume, 10

- onStart, 11
- onStop, 11
- recupererInformationPartie, 11
- recupererWidgets, 11
- TAG, 13
- trierLesJoueurs, 12
- com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie
 - ACTIVITE_FIN_PARTIE, 24
 - actualiserScoreIHM, 17
 - adapteur, 25
 - affichagelImpact, 25
 - afficherGagnant, 17
 - afficherImpact, 17
 - afficherJoueurActuel, 18
 - bluetoothAdapter, 25
 - boutonLancerPartie, 25
 - boutonTirManque, 25
 - chargerLesParametres, 18
 - cibleDarts, 25
 - ecranDarts, 26
 - envoyerAquitement, 19
 - handler, 26
 - initialiserHandler, 19
 - initialiserListeJoueur, 20
 - initialiserPartie, 21
 - initialiserWidgets, 21
 - listJoueur, 26
 - maPartie, 26
 - mesJoueurs, 26
 - onActivityResult, 22
 - onCreate, 22
 - onDestroy, 23
 - onPause, 23
 - onResume, 23
 - onStart, 23
 - onStop, 24
 - recupererPeripheriques, 24
 - TAG, 27
 - typePartie, 27
- com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie
 - activerBluetooth, 30
 - adapteurNomJoueur, 40
 - afficherDialogAjouterJoueur, 30
 - bluetoothAdapter, 40
 - boutonCibleDarts, 40
 - boutonEcranDarts, 40
 - COULEUR_BOUTON_GRIIS, 41
 - COULEUR_BOUTON_ORANGE, 41
 - COULEUR_BOUTON_ROUGE, 41
 - COULEUR_BOUTON_VERT, 41
 - cibleDarts, 41
 - creerListeJoueurs, 31
 - creerListeTypeJeu, 31
 - demarrerActiviteJouerPartie, 31
 - dialogueAjouterJoueur, 41
 - ecranDarts, 42
 - finish, 32
 - gererConnexionCibleDARTS, 32
 - gererConnexionEcranDARTS, 32
 - getPeripheriqueCibleDarts, 33
 - getPeripheriqueEcranDarts, 33
 - handler, 42
 - initialiserDialogue, 33
 - initialiserHandler, 34
 - initialiserWidgets, 35
 - LancerPartie, 36
 - listeJoueur, 42
 - nomJoueur, 42
 - nomsJoueur, 42
 - onActivityResult, 36
 - onCreate, 37
 - onDestroy, 37
 - onPause, 38
 - onResume, 38
 - onStart, 38
 - onStop, 38
 - placerDialogueEnHaut, 39
 - REQUEST_CODE_ENABLE_BLUETOOTH, 42
 - rechercherCibleDARTS, 39
 - rechercherEcranDARTS, 39
 - SpinnerTypeJeu, 43
 - TAG, 43
- com : :lasalle84 : :mobile_darts : :IhmMobileDarts
 - ACTIVITE_LANCER_PARTIE, 47
 - demarrerActiviteAfficherHistorique, 44
 - demarrerActiviteLancerPartie, 44
 - EXTRA_MESSAGE, 47
 - onActivityResult, 45
 - onCreate, 45
 - onDestroy, 46
 - onPause, 46
 - onResume, 46
 - onStart, 46
 - onStop, 47
 - TAG, 47
 - versionDarts, 48
- com : :lasalle84 : :mobile_darts : :Joueur
 - aFin, 52
 - getNom, 50
 - getScore, 50
 - isaFin, 50
 - Joueur, 49
 - nom, 52
 - retirerPoint, 50
 - score, 52
 - setNom, 51
 - setScore, 52
 - setaFin, 51
 - TAG, 53
- com : :lasalle84 : :mobile_darts : :Partie
 - actualiserJoueurIHM, 56
 - actualiserScoreIHM, 56
 - attendreImpact, 57
 - attendreJoueurSuivant, 57
 - cibleManquee, 58
 - demarrerPartie, 58
 - ENVOYER_AQUITTEMENT, 61
 - envoyerAquitement, 59
 - envoyerGagnantIHM, 59
 - estAuSuivant, 61
 - estDoubleImpact, 62
 - estFin, 62

GAGNANT, 62
 getTypeJeu, 60
 handlerUI, 62
 IMPACT, 62
 impact, 63
 impactEstRecu, 63
 impactIHM, 60
 JOUEUR_SUIVANT, 63
 lesJoueurs, 63
 NB_FLECHETTE, 63
 nbJoueurs, 64
 nbManche, 64
 Partie, 55
 SET_SCORE, 64
 TAG, 64
 traiterTrame, 61
 typeJeu, 64
 com : :lasalle84 : :mobile_darts : :Peripherique
 adresse, 72
 CODE_CONNEXION, 72
 CODE_DECONNEXION, 72
 CODE_ERREUR_CONNECTER, 72
 CODE_ERREUR_DECONNECTER, 73
 CODE_ERREUR_ENVOYER, 73
 CODE_ERREUR_RECEVOIR, 73
 CODE_RECEPTION, 73
 connecter, 68
 deconnecter, 69
 device, 73
 envoyer, 69
 envoyerMessageActivite, 70
 estConnecte, 70
 getAdresse, 71
 getNom, 71
 handlerActivite, 74
 mutex, 74
 NOM_CIBLE, 74
 NOM_ECRAN, 74
 nom, 74
 Peripherique, 68
 receiveStream, 75
 reception, 75
 sendStream, 75
 setHandlerActivite, 71
 socket, 75
 TAG, 75
 com : :lasalle84 : :mobile_darts : :TReception
 arreter, 78
 estFini, 80
 handler, 80
 mutex, 80
 peripherique, 80
 receiveStream, 81
 run, 78
 setHandlerActivite, 79
 TAG, 81
 com : :lasalle84 : :mobile_darts : :TypeJeu
 doubleOut, 85
 estDoubleOut, 83
 getPointDepart, 83
 getTypeJeu, 84
 PARTIE_301, 85
 PARTIE_301_DOUBLE_OUT, 85
 PARTIE_501, 85
 PARTIE_501_DOUBLE_OUT, 86
 pointDepart, 86
 setDoubleOut, 84
 setPointDepart, 84
 TAG, 86
 TypeJeu, 82, 83
 connecter
 com : :lasalle84 : :mobile_darts : :Peripherique, 68
 creerListeJoueurs
 com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 31
 creerListeTypeJeu
 com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 31

 declancherConfettis
 com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 8
 deconnecter
 com : :lasalle84 : :mobile_darts : :Peripherique, 69
 demarrerActiviteAfficherHistorique
 com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 44
 demarrerActiviteJouerPartie
 com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 31
 demarrerActiviteLancerPartie
 com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 44
 demarrerPartie
 com : :lasalle84 : :mobile_darts : :Partie, 58
 device
 com : :lasalle84 : :mobile_darts : :Peripherique, 73
 dialogueAjouterJoueur
 com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 41
 doubleOut
 com : :lasalle84 : :mobile_darts : :TypeJeu, 85

 ENVOYER_AQUITTEMENT
 com : :lasalle84 : :mobile_darts : :Partie, 61
 EXTRA_MESSAGE
 com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 47
 ecranDarts
 com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 26
 com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 42
 envoyer
 com : :lasalle84 : :mobile_darts : :Peripherique, 69
 envoyerAquittement
 com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 19
 envoyerAquittement
 com : :lasalle84 : :mobile_darts : :Partie, 59
 envoyerGagnantIHM
 com : :lasalle84 : :mobile_darts : :Partie, 59
 envoyerMessageActivite
 com : :lasalle84 : :mobile_darts : :Peripherique, 70
 estAuSuivant
 com : :lasalle84 : :mobile_darts : :Partie, 61
 estConnecte
 com : :lasalle84 : :mobile_darts : :Peripherique, 70
 estDoubleImpact
 com : :lasalle84 : :mobile_darts : :Partie, 62
 estDoubleOut
 com : :lasalle84 : :mobile_darts : :TypeJeu, 83
 estFini

- com : :lasalle84 : :mobile_darts : :Partie, 62
- com : :lasalle84 : :mobile_darts : :TReception, 80
- finish
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 32
- GAGNANT
 - com : :lasalle84 : :mobile_darts : :Partie, 62
- gererConnexionCibleDARTS
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 32
- gererConnexionEcranDARTS
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 32
- getAdresse
 - com : :lasalle84 : :mobile_darts : :Peripherique, 71
- getNom
 - com : :lasalle84 : :mobile_darts : :Joueur, 50
 - com : :lasalle84 : :mobile_darts : :Peripherique, 71
- getPeripheriqueCibleDarts
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 33
- getPeripheriqueEcranDarts
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 33
- getPointDepart
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 83
- getScore
 - com : :lasalle84 : :mobile_darts : :Joueur, 50
- getTypeJeu
 - com : :lasalle84 : :mobile_darts : :Partie, 60
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 84
- handler
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 26
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 42
 - com : :lasalle84 : :mobile_darts : :TReception, 80
- handlerActivite
 - com : :lasalle84 : :mobile_darts : :Peripherique, 74
- handlerUI
 - com : :lasalle84 : :mobile_darts : :Partie, 62
- IMPACT
 - com : :lasalle84 : :mobile_darts : :Partie, 62
- IhmMobileDarts.java, 99, 100
- impact
 - com : :lasalle84 : :mobile_darts : :Partie, 63
- impactEstRecu
 - com : :lasalle84 : :mobile_darts : :Partie, 63
- impactIHM
 - com : :lasalle84 : :mobile_darts : :Partie, 60
- initialiserDialogue
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 33
- initialiserHandler
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 19
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 34
- initialiserListeJoueur
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 20
- initialiserPartie
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 21
- initialiserWidgets
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 9
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 21
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 35
- isaFinis
 - com : :lasalle84 : :mobile_darts : :Joueur, 50
- JOUEUR_SUIVANT
 - com : :lasalle84 : :mobile_darts : :Partie, 63
- Joueur
 - com : :lasalle84 : :mobile_darts : :Joueur, 49
- Joueur.java, 101, 102
- LancerPartie
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 36
- lesJoueur
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 13
- lesJoueurs
 - com : :lasalle84 : :mobile_darts : :Partie, 63
- listJoueur
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 26
- listeJoueur
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 42
- maPartie
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 26
- mesJoueurs
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 26
- mutex
 - com : :lasalle84 : :mobile_darts : :Peripherique, 74
 - com : :lasalle84 : :mobile_darts : :TReception, 80
- NB_FLECHETTE
 - com : :lasalle84 : :mobile_darts : :Partie, 63
- NOM_CIBLE
 - com : :lasalle84 : :mobile_darts : :Peripherique, 74
- NOM_ECRAN
 - com : :lasalle84 : :mobile_darts : :Peripherique, 74
- nbJoueurs
 - com : :lasalle84 : :mobile_darts : :Partie, 64
- nbManche
 - com : :lasalle84 : :mobile_darts : :Partie, 64
- nom
 - com : :lasalle84 : :mobile_darts : :Joueur, 52
 - com : :lasalle84 : :mobile_darts : :Peripherique, 74
- nomGagnant
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 13
- nomJoueur
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 42
- nomsJoueur
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 42
- onActivityResult
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 22
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 36
 - com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 45
- onClick
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 9
- onCreate
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 9
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 22
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 37
 - com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 45
- onDestroy
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 10
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 23
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 37
 - com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 46
- onPause

- com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 10
- com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 23
- com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 38
- com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 46
- onResume
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 10
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 23
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 38
 - com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 46
- onStart
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 11
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 23
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 38
 - com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 46
- onStop
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 11
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 24
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 38
 - com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 47
- PARTIE_301
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 85
- PARTIE_301_DOUBLE_OUT
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 85
- PARTIE_501
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 85
- PARTIE_501_DOUBLE_OUT
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 86
- Partie
 - com : :lasalle84 : :mobile_darts : :Partie, 55
- Partie.java, 102, 103
- Peripherique
 - com : :lasalle84 : :mobile_darts : :Peripherique, 68
- peripherique
 - com : :lasalle84 : :mobile_darts : :TReception, 80
- Peripherique.java, 105, 106
- placerDialogueEnHaut
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 39
- Planification.md, 108
- pointDepart
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 86
- Protocole.md, 109
- README.md, 110
- REQUEST_CODE_ENABLE_BLUETOOTH
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 42
- receiveStream
 - com : :lasalle84 : :mobile_darts : :Peripherique, 75
 - com : :lasalle84 : :mobile_darts : :TReception, 81
- reception
 - com : :lasalle84 : :mobile_darts : :Peripherique, 75
- rechercherCibleDARTS
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 39
- rechercherEcranDARTS
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 39
- recupererInfomationPartie
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 11
- recupererPeripheriques
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 24
- recupererWidgets
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 11
- retirerPoint
 - com : :lasalle84 : :mobile_darts : :Joueur, 50
- run
 - com : :lasalle84 : :mobile_darts : :TReception, 78
- SET_SCORE
 - com : :lasalle84 : :mobile_darts : :Partie, 64
- score
 - com : :lasalle84 : :mobile_darts : :Joueur, 52
- sendStream
 - com : :lasalle84 : :mobile_darts : :Peripherique, 75
- setDoubleOut
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 84
- setHandlerActivite
 - com : :lasalle84 : :mobile_darts : :Peripherique, 71
 - com : :lasalle84 : :mobile_darts : :TReception, 79
- setNom
 - com : :lasalle84 : :mobile_darts : :Joueur, 51
- setPointDepart
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 84
- setScore
 - com : :lasalle84 : :mobile_darts : :Joueur, 52
- setaFini
 - com : :lasalle84 : :mobile_darts : :Joueur, 51
- socket
 - com : :lasalle84 : :mobile_darts : :Peripherique, 75
- SpinnerTypeJeu
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 43
- TAG
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 13
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 27
 - com : :lasalle84 : :mobile_darts : :ActiviteLancerPartie, 43
 - com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 47
 - com : :lasalle84 : :mobile_darts : :Joueur, 53
 - com : :lasalle84 : :mobile_darts : :Partie, 64
 - com : :lasalle84 : :mobile_darts : :Peripherique, 75
 - com : :lasalle84 : :mobile_darts : :TReception, 81
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 86
- TReception.java, 111, 112
- Thread, 76
- traiterTrame
 - com : :lasalle84 : :mobile_darts : :Partie, 61
- trierLesJoueurs
 - com : :lasalle84 : :mobile_darts : :ActiviteFinPartie, 12
- TypeJeu
 - com : :lasalle84 : :mobile_darts : :TypeJeu, 82, 83
- typeJeu
 - com : :lasalle84 : :mobile_darts : :Partie, 64
- TypeJeu.java, 113, 114
- typePartie
 - com : :lasalle84 : :mobile_darts : :ActiviteJouerPartie, 27
- versionDarts
 - com : :lasalle84 : :mobile_darts : :IhmMobileDarts, 48