

Explication Diagramme

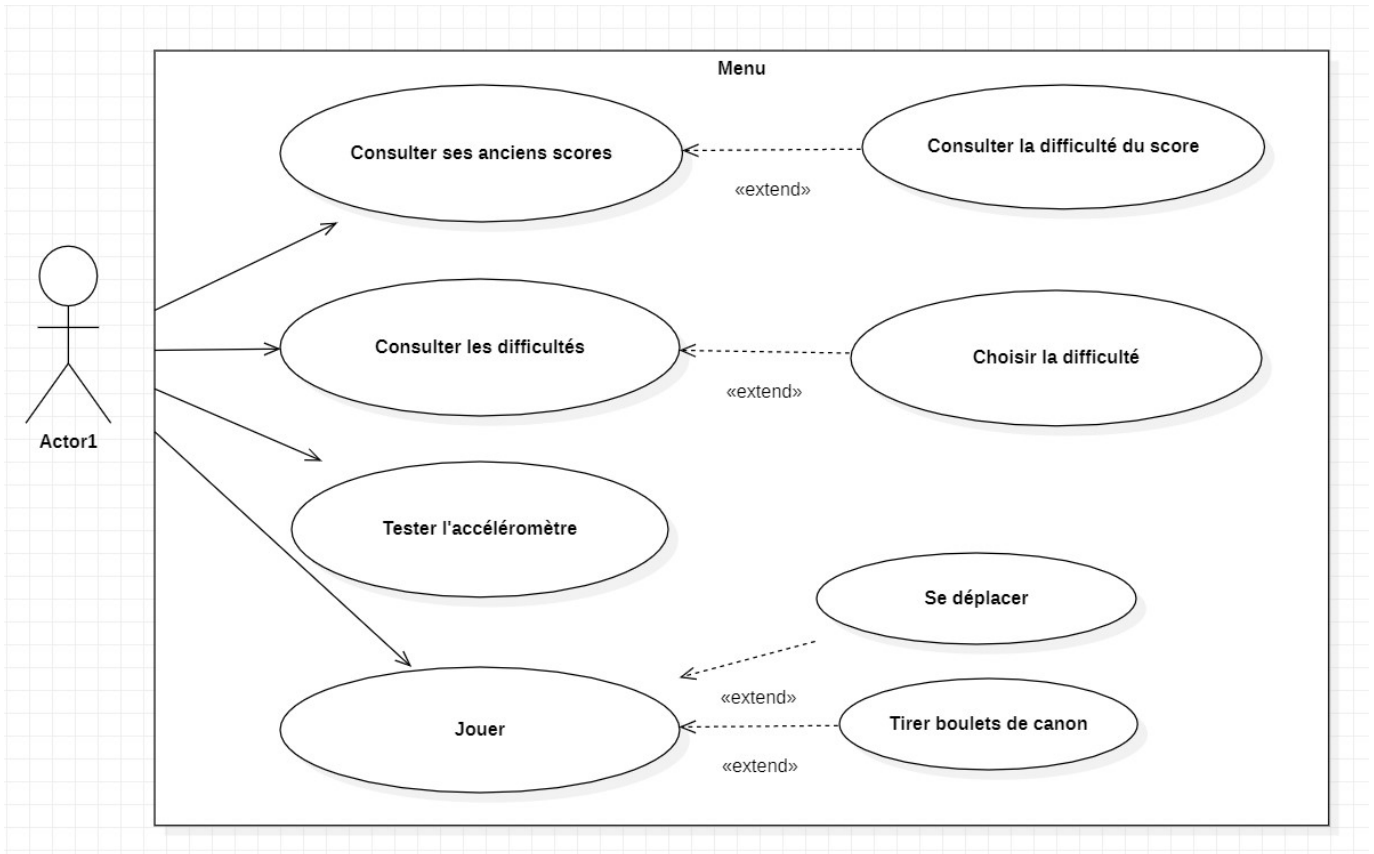


Diagramme de cas d'utilisation de l'application.

L'utilisateur à plusieurs choix, il peut consulter ses anciens scores. Quand il appuie sur un de ses anciens scores, la difficulté apparaît. Il peut choisir la difficulté en appuyant sur Difficulté. Il peut tester l'accéléromètre en appuyant sur Tester l'accéléromètre. Enfin, il peut jouer en appuyant sur Play. Quand il joue il peut se déplacer et tirer des boulets de canon, mais il faut d'abord qu'il joue.

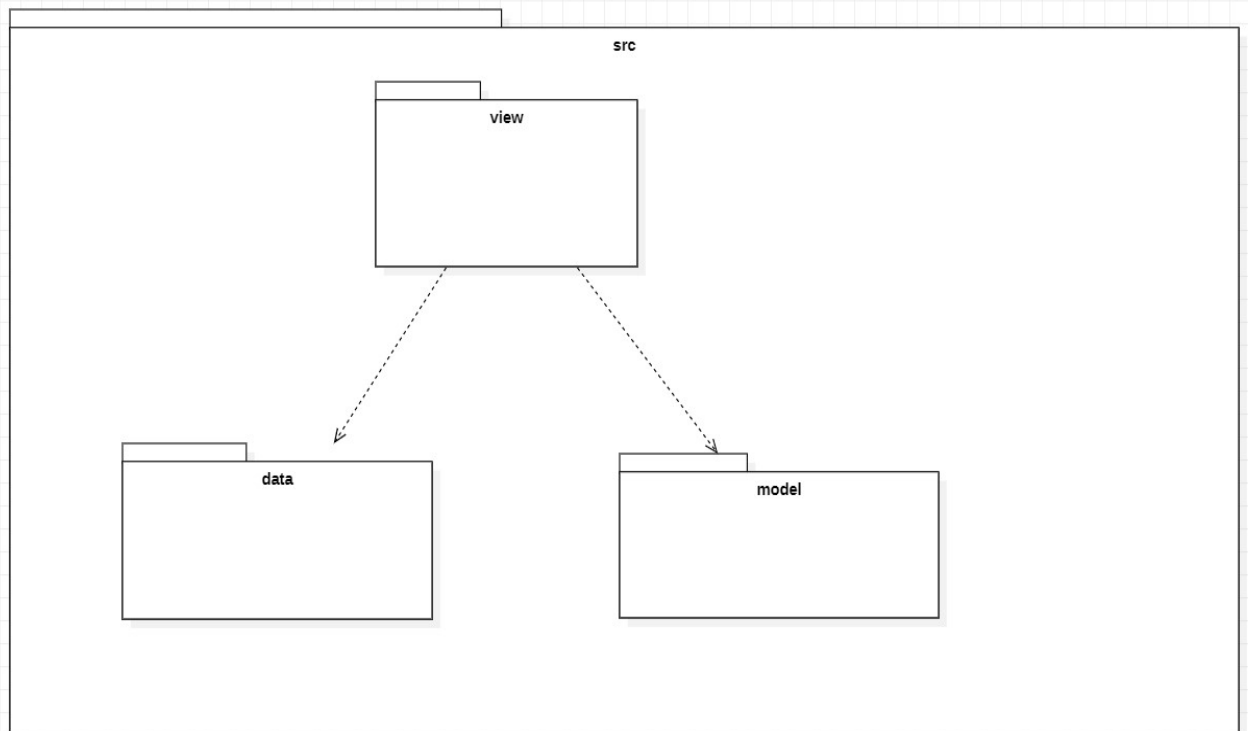


Diagramme de paquetage.

Il y a 3 paquetages :

- View : Pour les éléments de la vue.
- Model : Il y a tout le code, pour le fonctionnement du jeu (controler, metier) et pour le fonctionnement des autres fenêtres (results, difficulty).
- Data : Gère ce qui est données. C'est là où se trouve le Stub des Résultats et la classe permettant de sérialiser et désérialiser des objets.

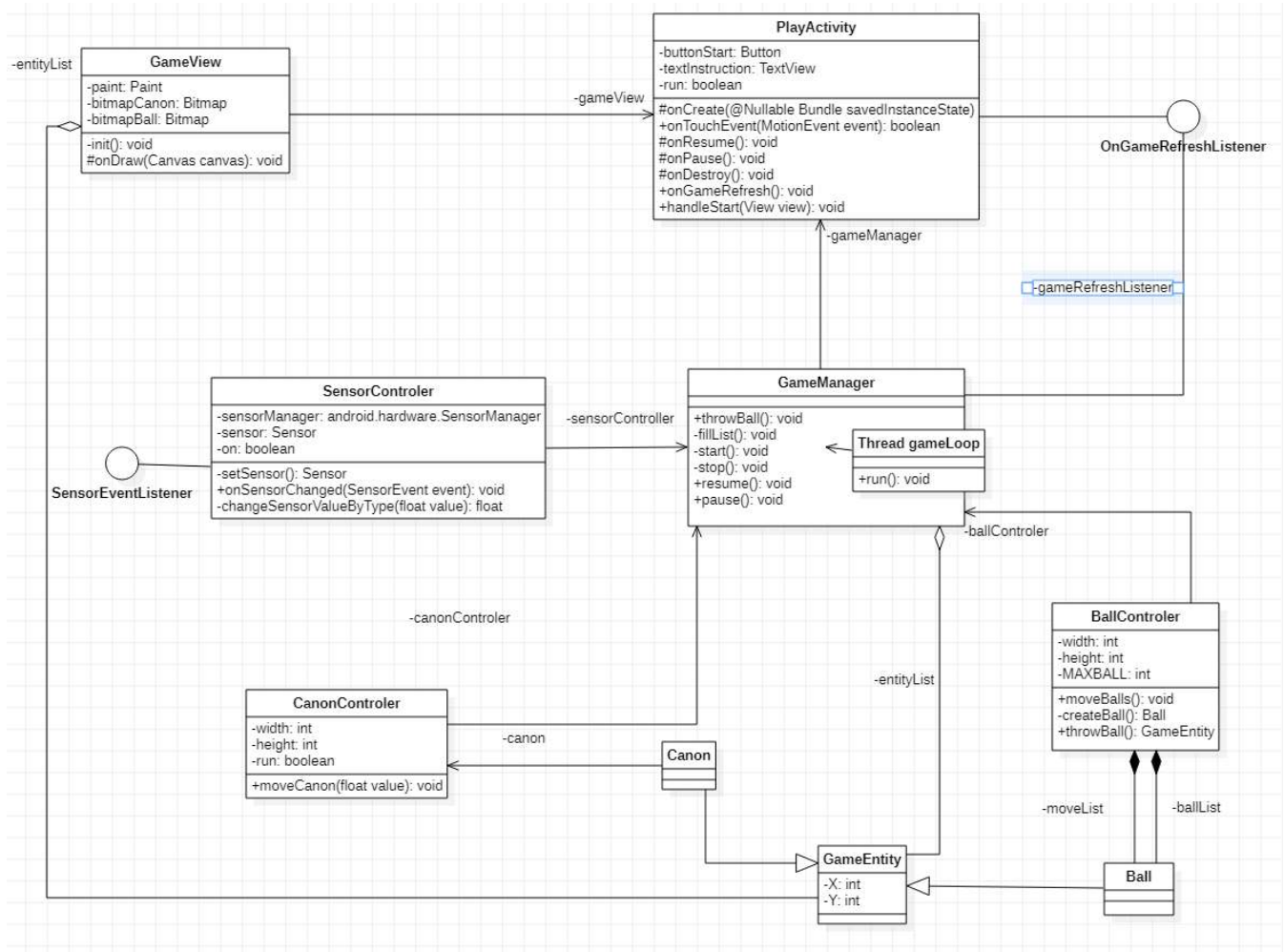


Diagramme de classe de la partie pour Jouer.

La partie Game est toute la partie sur dynamique de l'application « jouable ». Elle est constituée de 4 parties :

1. Les Entités

Une entité est un objet qui sera destiné à être dessiné sur l'écran et à être déplacé grâce à des position. Ces positions sont contenues directement dans la classe GameEntity sous forme de 2 int X et Y. Canon et Ball héritent de GameEntity, ce qui nous permettra de pouvoir les différencier si besoin tout en pouvant les manipuler de la même façon.

2. Les Contrôleurs

D'abord BallControler, comme son nom l'indique, cette classe gère toutes les balles. Elle possède une collection de Ball et les manipule afin d'effectuer leurs mouvements.

La classe `CanonController` gère les mouvements d'un seul et unique `Canon`, mais ce n'est pas elle qui choisit comment déplacer le `Canon`.

Le SensorControler fait le lien entre le métier et le hardware du téléphone. Il gère tous les changements du capteur qu'il envoie au CanonControler. Il possède une méthode qui choisit le capteur adéquat en fonction de ceux disponibles sur le téléphone.

3. Le Manager

Le GameManager est la classe principale du jeu qui fait le lien entre tous nos contrôleurs et notre Vue. Il possède un Thread imbriqué qui nous sert de boucle de jeu. Cette boucle de jeu appelle les méthodes des contrôleurs pour mettre à jour les objets du jeu. Le GameManager possède la collection de tous nos objets. Cette collection est passée à la Vue pour l'affichage.

4. La Vue

La Vue se compose de 2 parties :

La PlayActivity fait le lien direct entre l'utilisateur et la machine. Cette classe permet la gestion des boutons, de l'écran tactile mais aussi du Context.

La GameView est chargée de dessiner tous les objets de notre jeu. Elle contient la même collection de GameEntity que le GameManager par copie et ne fait que la parcourir et dessiner chaque entité dans la méthode onDraw(). Le GameManager passe par une Interface appelée OnGameRefreshListener qui, à chaque tour de la boucle de jeu appelle la méthode onGameRefresh() dans la PlayActivity. Cette méthode appelle le rafraîchissement de la page pour redessiner dessus en appelant le OnDraw() de la GameView (gameView.invalidate()).

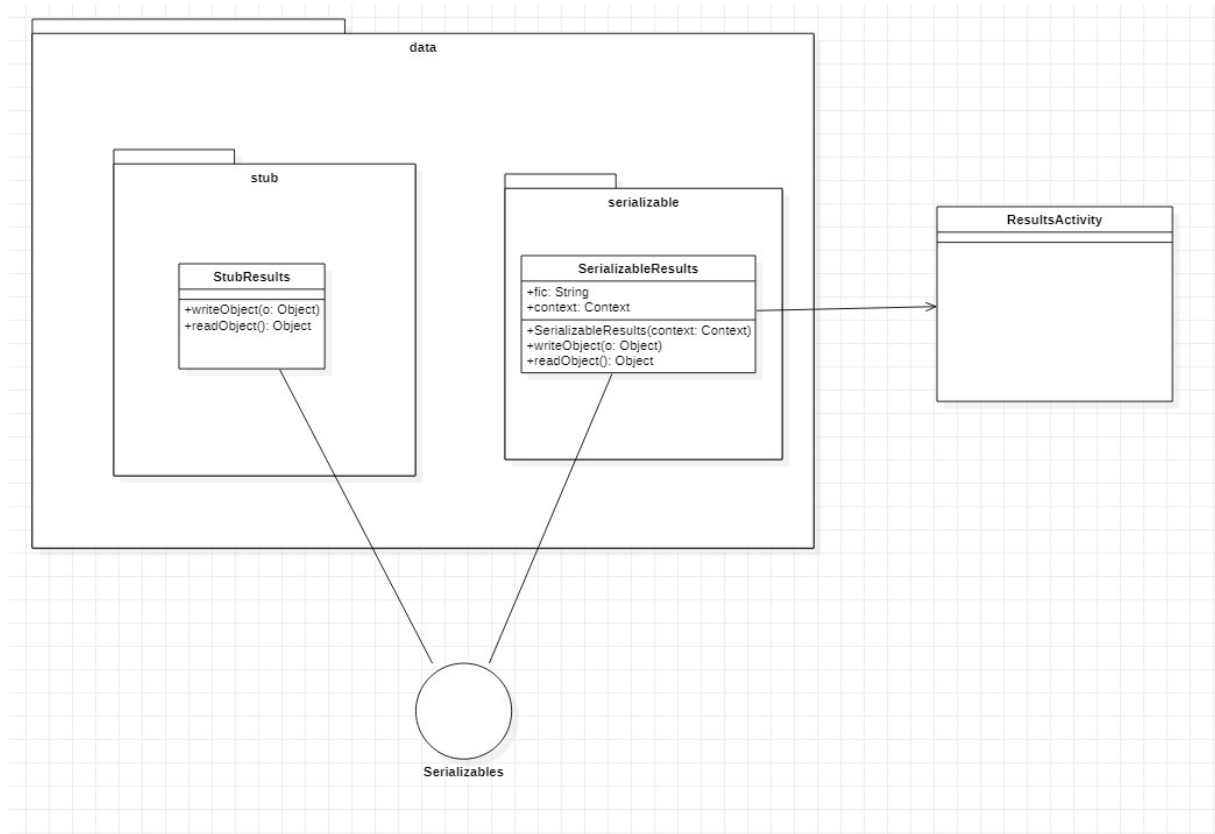


Diagramme de classe du paquet data.

Dans data se trouve 2 autres paquets. Stub et Serializable. Stub est utilisé pour ranger les Stubs, ici il y a StubResultats, qui renvoie un Stats qui contient une liste de résultats remplie. Serializable est utilisé pour ranger les classes permettant la sérialisation d'objets.

Ici SerializableResults est une classe permettant de sérialiser tout type d'objets. Si la méthode readObject de SerializableResults ne trouve pas le fichier à désérialiser, alors c'est readObject du Stub qui est appelé.

Les classes contenues dans Stub et Serializable implémentes l'interfaces Serializables.

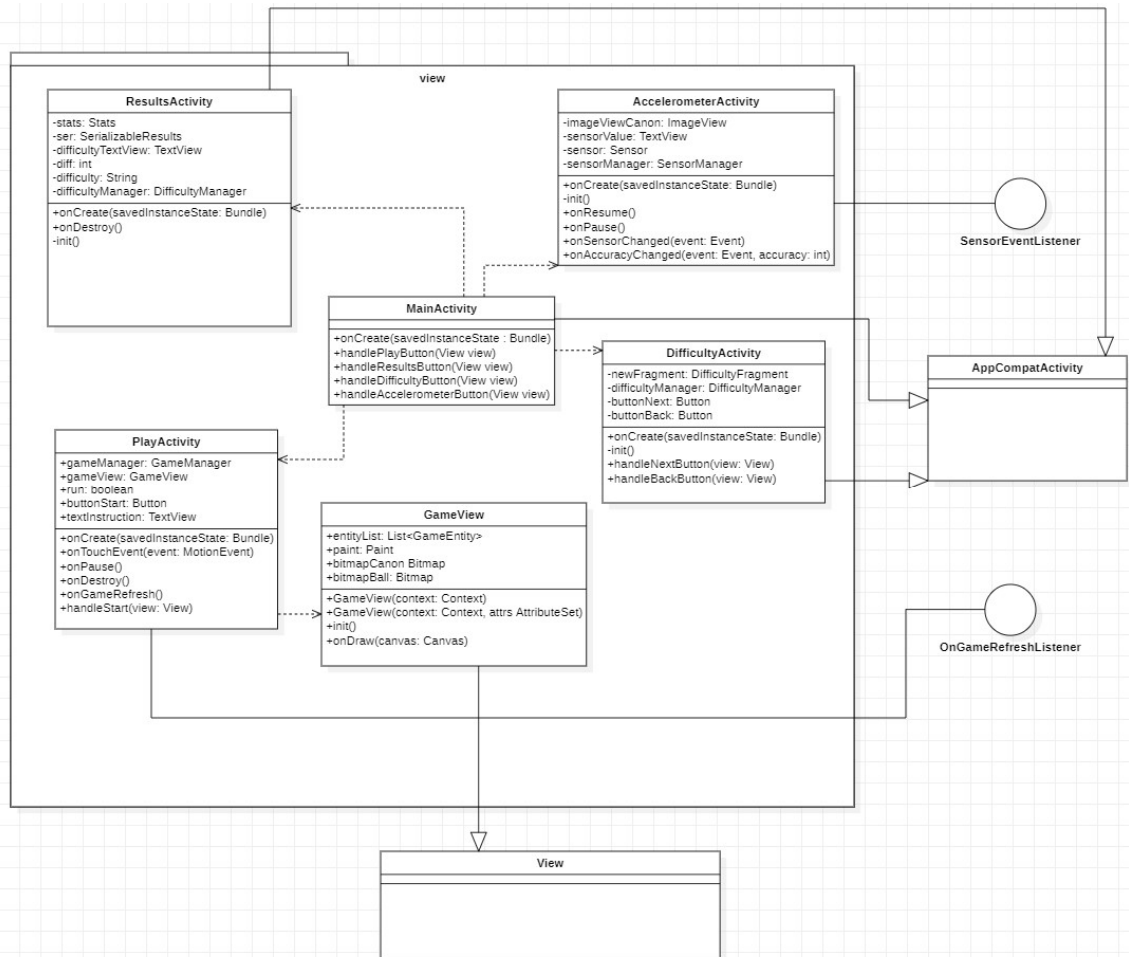


Diagramme de classe du paquet View.

Dans ce paquet se trouve les éléments liés directement à la vue. Chaque classe se finissant par Activity est relié à un fichier .XML comme par exemple DifficultyActivity est reliée à activity_difficulty. Dans ces fichiers, aucun calcul n'est fait mais seulement des appels à des méthodes.

Quand on démarre l'application on arrive sur MainActivity puis on peut se déplacer vers les autres vues.