

Rapport Développement mobile

RolePlay

A.Bento da Silva & G.Griesser

**Abstract :** Ce document contient l’ensemble des recherches et des explications nécessaires à comprendre, la conception et le développement de notre projet orienter android « RolePlay », dans le cadre de nos études à l’He-Arc, durant le cours de développement mobile.

[Introduction 3](#_Toc503527776)

[But 3](#_Toc503527777)

[Prérequis 3](#_Toc503527778)

[Conception 3](#_Toc503527779)

[Uml 3](#_Toc503527780)

[Parcours de l’application 3](#_Toc503527781)

[Diagramme de classe 4](#_Toc503527782)

[Scénario de l’histoire 5](#_Toc503527783)

[Réalisation 5](#_Toc503527784)

[Problème rencontré 5](#_Toc503527785)

[Incompréhension du stockage de données 5](#_Toc503527786)

[Accès bloqué aux données 6](#_Toc503527787)

[Problème lié aux émulateurs 6](#_Toc503527788)

[Conclusion 6](#_Toc503527789)

[Résultat 6](#_Toc503527790)

[Auto-critique 6](#_Toc503527791)

# Introduction

## But

Le principe de base de l’application était de raconter une histoire à l’utilisateur, en le mettant à la place du héros, il doit faire des choix qui modifieront son chemin dans l’histoire.

Ces différents choix permettent d’aborder l’histoire du héro sous un autre angle et permet donc d’augmenter la durée de vie de l’application.

Ce type de jeu narratif est un classique des jeux pc, mais n’étant plus à la mode lors de l’explosion des smartphones, il n’en existe que très peu sur le marché du mobile.

## Prérequis

L’application est écrite en Java spécialisé développement android. Nous avons choisi de nous imposer d’utilisé les moyens de sauvegardes persistantes, ainsi que l’utilisation du capteur l’accéléromètre.

# Conception

## Uml

### Parcours de l’application



Figure , Parcours des Activities

## Scénario de l’histoire

Chaque partie de l’histoire est stocké dans un fichier séparé, ce fichier est défini ainsi :



Figure , Fichier de scénario

Au début du fichier se trouve le texte narratif, puis les lignes débutant par un « / » donne les nœuds vers les fichiers suivants, entre les « / » se trouve le nom du fichier et la fin de la ligne est le choix qui sera présenté à l’utilisateur.

Il existe également des fichiers où le texte du scénario est remplacé par un mot-clé commençant par « \* », ces nœuds définissent des actions du héros comme se battre ou prendre la fuite.

Il existe un dernier type de fichier qui ne possède pas de nœuds vers lesquels se dirigé, puisqu’il s’agit des nœuds dans lesquels le héros meurt ou termine son histoire.

L’ensemble des fichiers de scénario représente le premier chapitre de l’histoire et est représenté par cet arbre :



Une histoire se suis par rapport aux lettres qui compose les fichiers de scénario, en sachant que A1 est le fichier de base de toutes les histoires. Par exemple

A1, A2, A3, A4, A5, A6, A7, A8 composent une histoire, A1, A2, B1, B2, B3, B4 en font une autre.

Les fichiers commençant par D (death) indiquent que le personnage meurt dans ces nœuds.

# Réalisation

## Vue en détail des classes

L’application commence dans **MenuActivity**, ce menu propose de créer une nouvelle partie ce qui appel IntroductionActivity, de charger une partie existante ce qui ouvre loadActivity ou de quitter l’application.

Dans **IntroductionActivity**, on vous invite à entrer un nom pour votre personnage, si celui-ci n’est pas déjà utiliser GameActivity est chargé.

Dans **LoadActivity**, vous pouvez sélectionner une sauvegarde stockée sur votre appareil, ce qui lancera également GameActivity.

**GameActivity** est l’activité principale, elle permet au joueur de suivre l’histoire si la partie se termine, on rappelle MenuActivity.

# Problème rencontré

## Incompréhension du stockage de données

Une erreur que nous avons fait lors de la conception, a été de ne pas spécifié le type de stockage de données, un membre du groupe voulais utiliser l’internal storage et l’autre pensais que l’objectif demandé était de stocker les données sur un support externe (sdCard), après avoir rencontré le problème, nous nous sommes mis d’accord sur l’utilisation de fichier interne.

## Accès bloqué aux données

Un problème rencontré durant la création du FileHandler (gestionnaire de sauvegarde et de lecture des fichiers), est que nous n’avions pas accès au dossier data de l’émulateur à travers « Android Device Monitor », ce qui nous posait des problèmes de débogages, notamment pour savoir si les fichiers de sauvegardes étaient créés.

Après quelques recherches nous avons compris qu’il s’agissait d’un problème de droit de lecture/écriture, que nous avons tenté de modifier, ce qui nécessitait plusieurs opérations notamment le routage de l’appareil, cette direction nous a fait perdre beaucoup de temps, pour au final ne pas apporter de résultats concrets.

Nous avons heureusement trouvé une autre solution temporaire, qui était de ne plus utiliser l’appareil de test fournit par l’école, mais d’utiliser un émulateur avec un niveau d’API supérieur ou égal à 23, qui n’ont pas ces problèmes de droits.

Ce qui nous à mener au problème suivant.

## Problème lié aux émulateurs

Nous avons eu de nombreux problèmes liés aux émulateurs, crash au démarrage, freeze alors que l’application n’est même pas lancée, les error 0x502 que nous n’avons pas pu corriger. Tous ces problèmes ont passablement ralenti, la partie sauvegarde et chargement de fichier.

# Conclusion

## Technique

## Auto-critique