

Développement d'un jeu Bomberman sous Android et iOS

TER

KLOB

K.COUSEIN L.PITIOT O.BONVILA B.TARDIEU
L.DERUELLE

Université Montpellier II

27 mai 2011

Sommaire

- 1 Introduction
- 2 Présentation
- 3 Application
- 4 Réutilisabilité
- 5 Discussion
- 6 Conclusion

Introduction

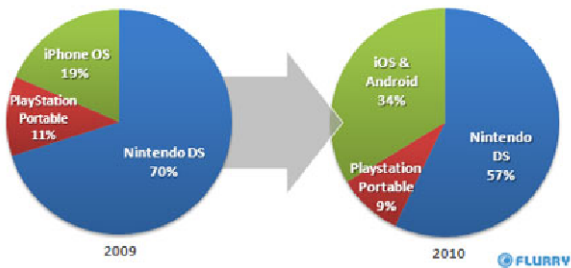


Présentation

Les jeux sur smartphone

Le marché des jeux vidéos sur console portable connaît une réelle expansion :

U.S. Portable Game Software by Revenue



Présentation

Android

Le système d'exploitation possède :

- Noyaux linux pour exploiter le matériel
- Bibliothèques connues et open source (OpenGL ES, SQLite,...)
- Machine virtuelle Java (Dalvik virtual machine)
- API Java riche (package de Java SE, open source ou spécifique au système)



Présentation

Android

Les possibilités de développement sont :

- Langage principal Java, développement en C/C++ possible
- Kit de développement multiplateforme
- Développement sur téléphone ou sur émulateur
- Déploiement des applications peu coûteux



Présentation iOS

Le système d'exploitation possède :

- Noyau dérivé de Mac OS X
- Bibliothèques connues et open source (OpenGL ES, SQLite,...)
- Pas de machine virtuelle. Code compilé en C.
- API Objective-C riche (Core OS, Cocoa Touch,...)



Présentation iOS

Les possibilités de développement sont :

- Langage principal Objective-C, développement en C possible
- Kit de développement disponible sur Mac OS seulement
- Développement sur emulateur seulement.
- Déploiement des applications coûteux

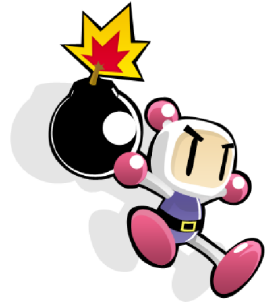


Présentation

Bomberman

Histoire

- Jeu d'action.
- Première apparition en 1987.
- Développé par Hudson Soft.
- Développé sur plusieurs consoles.
- Succès grâce au mode multijoueur sur certaines consoles.



Présentation

Bomberman

Principe :

- Le joueur incarne un poseur de bombes.
- But du jeu : détruire ses ennemis.
- Multiples bonus (Bonus de vie, de bombes, de vitesse,...).
- Multiples malus (Obligation de poser des bombes,...).



Présentation

Rapport avec l'enseignement

Ce TER nous a permis de mettre en application les connaissances acquises dans nos parcours d'enseignements.

Aspect technique

Intelligence artificielle
Communication mobile-serveur
Serveur d'application
Conception de logiciel

Aspect pédagogique

I2A
CASAR
DIWEB
Génie Logiciel

Application

Chargement de l'application

Premier chargement :

- ❶ Création de la base de données
- ❷ Récupération de toutes les ressources dans le fichier XML
- ❸ Instanciation des objets grâce aux ressources récupérées
- ❹ Copie des cartes dans le répertoire d'installation du téléphone
- ❺ Initialisation de la base de données
- ❻ Affichage de la page de création de compte local

Application

Chargement de l'application

Chargement standard :

- 1 Chargement de la base de données
- 2 Récupération de toutes les ressources dans le fichier XML
- 3 Instanciation des objets grâce aux ressources récupérées
- 4 Instanciation du dernier utilisateur

Application

Menus

Diagramme d'activité des menus

Application

Types d'objets

Deux types d'objets :

- Non animés (tile) : une seule et unique image



- Animés (sprite) : une sequence d'images (animation)



Application

Tile mapping

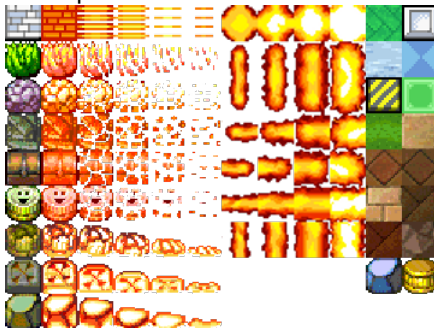
Pourquoi ?

- 1 Tire ses origines des jeux des années 80
- 2 Faible consommation des ressources
- 3 Performances des smartphones limitées

Application

Tile Mapping

Principe :



Application

Gestion des ressources

XML

- Code portable
- Syntaxe extensible (Générique)
- Ajout/suppression d'éléments facile

Exemple de code

Application

Editeur de cartes



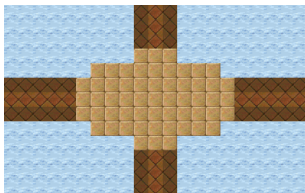
Application

Editeur de cartes

Application

Moteur de rendu (Editeur de cartes)

Deux matrices :



1er niveau

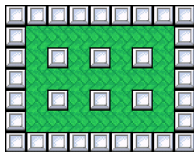


2ème niveau

Application

Moteur de rendu (Jeu)

Bitmap d'objets
inanimés

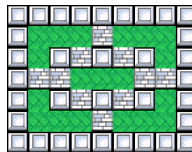


+

Table de hashage
d'objets animés

=

Resultat



Application

Moteurs de rendu

Comparaison de la complexité des deux moteurs

Application

Moteur Physique



Application

Intelligence artificielle

Niveaux de difficulté

- ① Facile
- ② Moyen
- ③ Difficile



Application

Pathfinding

Algorithme A*

- ① Heuristique (de Manatan)
- ② Coût de déplacement
- ③ Premier chemin trouvé
- ④ Rapidité (Dijkstra)

	A	B	C	D	E	F	G
1	94 B2	80 B2	74 C2				
2	24 70	20 60	24 50				
3	74 B3	60 B3	54 B3				
4	14 60	10 50	14 40				
5	60 B3	F Père	40 B3		82 F4	F4	82 F4
6	10 50	G H	10 30		72 10	68 0	72 10
7	74 B3	60 B3	54 B3		74 E5	68 E5	88 F4
8	14 60	10 50	14 40		54 20	58 10	68 20
9	94 B4	80 B4	74 C4	74 C5	74 D5	74 E5	102 F4
10	24 70	20 60	24 50	34 40	44 30	54 20	72 30

Application

Pathfinding

Algorithme de parcours en largeur

- 1 Pas de case d'arrivée nécessaire
- 2 Tous les chemins possibles
- 3 Premier chemin trouvé
- 4 Rapidité

Application

Fonctionnalités

Les joueurs ont accès à :

- la création de comptes multijoueurs
- la page de connexion en ligne
- la liste des parties en lignes

The screenshot shows a 'New Account' screen with a blue header bar. On the left is a 'Back' button and on the right is a 'Connection' button. Below the header, the text 'klob' is displayed. There are three input fields: 'User Name' (with a blue border), 'Password', and 'Verification'. Below these fields are two checkboxes: 'Autologin' and 'Remember password', both of which are checked.

Application

Outils

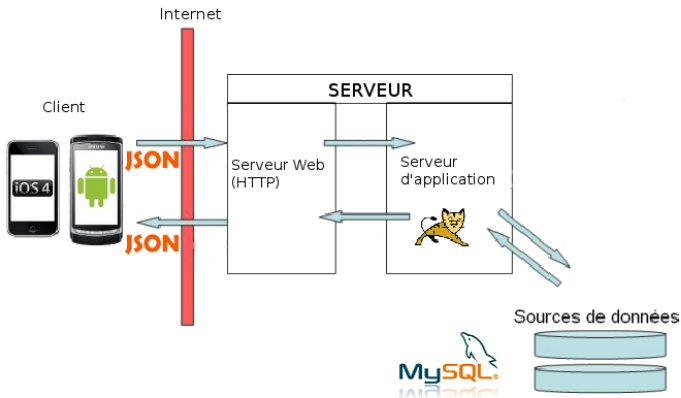
Outils utilisés :

- Servlets
- Serveur d'application
- JSON
- MySQL



Application

Principe



Réutilisabilité

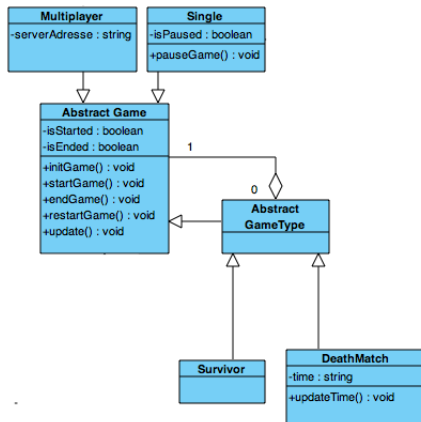
- multilangue
- doc en anglais
- code en anglais

Réutilisabilité

- MVC (gameplay, interface graphique)
- Design pattern décorateur (type de parties)
- XML (objets(bonus,malus,bloc,bombes, etc ...) ,
images(theme), sons)

Réutilisabilité

Nouveaux types de parties



- Design pattern décorateur
- Permet l'extension du modèle

Réutilisabilité

Personalisation

XML L'utilisation du XML permet d'ajouter ou de personnaliser facilement :

- Les objets
- Les images
- Les sons

- Nouvelles servlets = nouvelles fonctionnalités
- même hiérarchie de classes que le mobile = decorator =_i tt le monde doit l'avoir

Discussion

Difficultés

Android

Nouvelle plate-forme

Multi-touch

Ressources limitées

iOS

Nouveau langage
(Objective-C)

Nouvelle plate-forme

Gestion manuelle
de la mémoire

Serveur

Communication avec
la base de données

Servlets

Déploiement

Discussion

Problèmes

Android et iOS :

- Tester l'application
- OpenGL ES

Serveur :

-

Discussion

Améliorations

- Mode histoire
- Ajout de bonus / malus
- Rajout de types de parties
- Gestion des scores
- Nouvelles bombes

Conclusion

Apport en relation avec nos parcours d'enseignement

I2A	CASAR	DIWEB	GL
Recherche opérationnelle	Communication mobile-serveur	Servlet	Conception d'une application
A*	Mise en place d'un serveur	BDD	MVC
Parcours en largeur	Sécurisation du réseau	XML	Design pattern décorateur
Moteur de jeu		IHM ergonomique	

Conclusion

Ce que cela nous a apporté

- Découverte de la programmation mobile (SDK Android, SDK iOS).
- Apprentissage d'un nouveau langage (Objective-C).
- Découverte de la programmation de jeux vidéos.
- Communication mobile-serveur.
- Travail en groupe.