

Créez des jeux de A à Z avec Unity

Jeu mobile et FPS 3D

par
Anthony Cardinale

Créez des jeux de A à Z avec Unity • Jeu mobile et FPS 3D
par Anthony Cardinale

ISBN (imprimé) : 978-2-8227-1150-0

Copyright © 2025 Éditions D-BookeR
Tous droits réservés

Conformément au Code de la propriété intellectuelle, seules les copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective ainsi que les analyses et les courtes citations dans un but d'exemple et d'illustration sont autorisées. Tout autre représentation ou reproduction, qu'elle soit intégrale ou partielle, requiert expressément le consentement de l'éditeur (art L 122-4, L 122-5 2 et 3a).

Publié par les Éditions D-BookeR, 229 rue Solférino, 59000 Lille
www.d-booker.fr
contact@d-booker.fr

Les exemples (téléchargeables ou non), sauf indication contraire, sont propriété des auteurs.

Mise en page : générée sous Calenco avec des XSLT développées par la société NeoDoc (www.neodoc.biz)
Diffusion : AFPU-D (<https://www.afpu-diffusion.fr>)
Distribution : DILISCO (<https://dilisco-diffusion.centprod.com>)

Dépôt légal : Février 2025
Édition : 4
Version : 4.0

Table des matières

Avant-propos	ix
1. Sources des exemples	x
2. URL raccourcies	x
Trouver une idée de jeu	1
1. Huit pistes pour stimuler vos idées	1
2. Le brainstorming	8
3. Les ateliers créatifs	9
Récolter des idées	9
Prototyper une idée	9
Tester une idée	11
Pour une bonne conception	13
1. Le game design	13
2. Le gameplay	14
3. Le level design	17
4. Personnage et joueur	18
Analyse du jeu Trine	23
1. Présentation	23
2. Système de jeu	24
3. En quoi révolutionne-t-il son genre ?	25
4. Les personnages	26
5. La boucle de jeu	28
6. La rejouabilité	29
7. Level design	30
8. Making of du level design	31
9. Conclusion	34
I. Votre premier jeu mobile	35
Présentation du projet	37
1. Découverte de Unity et mise en place du projet	39
1.1. Téléchargement du Unity HUB	40
1.2. Création d'un projet 3D	44
1.3. Découverte de l'interface de Unity	46
La fenêtre de projet	46

La hiérarchie	47
La scène	47
La fenêtre de jeu	47
L'inspecteur	48
La barre d'outils	49
La barre de menus	50
1.4. Assets	51
L'Asset Store	52
Importation de nos assets	53
Le gestionnaire de paquets (Package Manager)	54
2. Création des niveaux	57
2.1. Utilisation des modèles 3D	57
2.2. Playtest	60
2.3. La gravité	61
2.4. Les composants de collision	63
2.5. Création du premier niveau	66
2.6. Un peu de pratique (requis)	69
3. Un peu de couleur	71
3.1. Les lumières	71
3.2. Le ciel (Skybox)	73
3.3. Remplissage du vide	76
3.4. Coloration des éléments	81
3.5. Un peu de pratique (recommandé)	84
4. Création de la balle	85
4.1. La forme	85
4.2. La physique	86
4.3. Création du champ de force	88
4.4. Trail Renderer	92
4.5. Un peu de pratique (recommandé)	94
5. Découverte de la programmation sous Unity	95
5.1. Structure d'un script et variables	95
5.2. Les fonctions	101
5.3. Les conditions	102
5.4. Un petit exemple de script	103
5.5. Utilisation de la documentation	105
5.6. Un exemple concret	106
6. Le script de la caméra	109

6.1. Mise en place	109
6.2. Les variables	110
6.3. Le cœur du script	112
6.4. Tester le jeu sur mobile	117
6.5. Un peu de pratique (recommandé)	123
7. Début du script de tir	125
7.1. Création de l'interface de test	125
7.2. Propulser la balle	128
7.3. Appeler la fonction	129
7.4. Un peu de pratique (recommandé)	131
8. L'interface utilisateur	133
8.1. Préparation des visuels	133
8.2. Création de l'interface	135
8.3. Coloration de l'interface	138
8.4. Lier l'interface au script	139
8.5. Un peu de pratique (requis)	143
9. Menu Pause et fin du script de tir	145
9.1. Menu Pause	145
9.2. Continuons le script de tir	148
9.3. Bloquer le tir pendant que la balle roule	150
9.4. Un peu de pratique (recommandé)	153
10. Animations, rebonds et obstacles	155
10.1. Création et animation d'obstacles	155
À l'aide du système d'animation de Unity	157
Par script	160
10.2. Une balle rebondissante	161
10.3. Un peu de pratique (recommandé)	163
11. Déblocage du niveau suivant	165
11.1. Les déclencheurs	165
Relancement du niveau	165
Déclencheur de fin	169
11.2. Les PlayerPrefs	172
11.3. Un peu de pratique (recommandé)	174
12. Création du menu principal	175
12.1. Création de la scène du menu	175
12.2. Script du menu	178

12.3. Bloquer les niveaux	179
12.4. Un peu de pratique (recommandé)	181
13. Amélioration des graphismes	183
13.1. Post-traitement	183
13.2. Mise en place du post-traitement	184
13.3. Les filtres de post-traitement	187
13.4. Anti-aliasing	191
13.5. Un peu de pratique (recommandé)	191
14. Publier son jeu	193
14.1. Compilation du projet	193
Propriétés du projet	193
Génération de l'exécutable et tests	195
14.2. Publier son jeu sur Google Play	197
14.3. Un peu de marketing	202
II. Un FPS 3D pour PC	205
Présentation du projet	207
15. Importation des ressources et personnage contrôlable	209
15.1. Récupération des ressources	209
15.2. Mise en place du personnage joueur	214
15.3. Un peu de pratique (recommandé)	216
16. Découverte de ProBuilder	217
16.1. Installation et présentation de ProBuilder	217
16.2. Utilisation de ProBuilder	219
Créer une nouvelle forme	220
Modes d'édition	220
Extrusion	222
Découper une forme	225
Ajouter de la couleur	228
Créer un escalier	228
16.3. Un peu de pratique (recommandé)	229
17. Level design avec ProBuilder	231
17.1. Modélisation 3D du niveau	231
17.2. Un peu de pratique (recommandé)	247
18. Éclairage et décoration	249

18.1. Les lumières	249
18.2. Un peu de décoration	252
18.3. Amélioration des graphismes	255
18.4. Un peu de pratique (recommandé)	259
19. Création du script de tir	261
19.1. Mise en place du pistolet	261
19.2. Création du projectile	266
19.3. Le script de tir de projectiles	268
19.4. Un peu de pratique (recommandé)	272
20. Pièges, tourelles automatiques et interface graphique	273
20.1. Création d'un piège	273
20.2. Tourelles automatiques	275
20.3. Script de la tourelle	278
20.4. Création d'une interface	281
20.5. Un peu de pratique (recommandé)	285
21. Collisions et gestion de la vie	287
21.1. Perdre des points de vie	287
21.2. Tomber dans un piège	291
21.3. Détruire un ennemi	292
21.4. Gagner de la vie	294
21.5. Un peu de pratique (requis)	296
22. Système d'ouverture de porte	297
22.1. Création de la clé	297
22.2. Création de la porte	299
22.3. Le portail magique	306
22.4. Un peu de pratique (requis)	307
23. Minimap et cinématique	309
23.1. Création de la minimap	309
23.2. Création d'une cinématique d'introduction	314
23.3. Un peu de pratique (recommandé)	322
24. Des ennemis qui se déplacent	323
24.1. Configuration de l'ennemi	323
24.2. Configuration d'une entité autonome	326
24.3. Le script de déplacement autonome	329
24.4. Un peu de pratique (recommandé)	332
25. Détails et finitions	333

25.1. Effets sonores	333
25.2. Une barre de vie pour les robots	336
25.3. Changement d'arme	339
25.4. Un peu de pratique (recommandé)	345
26. Compilation et distribution du jeu	347
26.1. Quelques notions d'optimisation	347
26.2. Compilation du jeu	350
26.3. Publier son jeu sur itch.io	353
26.4. Un peu de pratique	360
Crédits photos	361
Index	363
À propos de l'auteur	371

Avant-propos

Écrire un livre informatique, notamment sur Unity, est une tâche complexe car les outils évoluent en permanence. Malgré les rééditions du livre, il est difficile de toujours coller au logiciel, qui évolue plusieurs fois par an. Par ailleurs, depuis 2014 (date à laquelle j'ai commencé à écrire la première édition de ce livre), notre façon de développer des jeux a changé. Le langage de programmation a évolué, nous disposons de nouveaux outils d'animation, de conception d'interfaces, de création d'entités autonomes (IA), etc. Il existe également de nouveaux outils pour améliorer le rendu visuel des jeux et s'approcher d'un rendu professionnel. Toutes ces transformations m'ont conduit à repenser le livre et à le restructurer en deux parties complémentaires basées sur deux nouveaux projets d'exemple plus dans l'air du temps.

Au final, cette évolution est une très bonne nouvelle pour vous car cette nouvelle édition toute neuve prend en compte les retours des précédentes éditions et propose un contenu toujours plus riche et plus poussé, qui vous permettra d'apprendre à développer des jeux de qualité professionnelle.

Il s'adresse à tous ceux qui souhaitent découvrir la création de jeux avec un outil moderne et puissant. Il s'adresse aussi aux développeurs souhaitant migrer vers Unity.

Le livre est structuré en deux parties relativement autonomes :

- *I. Votre premier jeu mobile* vise à vous fournir toutes les bases dont vous aurez besoin pour réaliser un jeu fonctionnel à destination de smartphones et tablettes. De la conception des niveaux à la programmation en passant par l'optimisation, vous ferez un tour complet des fonctionnalités et techniques indispensables au développement de jeux avec Unity et le langage C#. Ce module constitue une bonne introduction pour ceux qui découvrent le logiciel.
- *II. Un FPS 3D pour PC* va vous permettre d'approfondir votre connaissance de Unity tout en concevant un jeu plus poussé à destination des PC. Vous verrez comment introduire des ennemis dynamiques et des pièges, mettre en place un système de points de vie, créer un dialogue, proposer plusieurs armes, ouvrir des portes, etc. Les PC disposant d'une bonne puissance de calcul, vous pourrez vous autoriser des graphismes, des éclairages et des rendus plus subtiles.

Du fait de leur construction indépendante, certains outils pourront être expliqués dans chacune des parties. Le contexte étant différent, la lecture des deux approches peut vous aider à mieux les assimiler. N'hésitez donc pas à naviguer de l'une à l'autre.

I. Votre premier jeu mobile

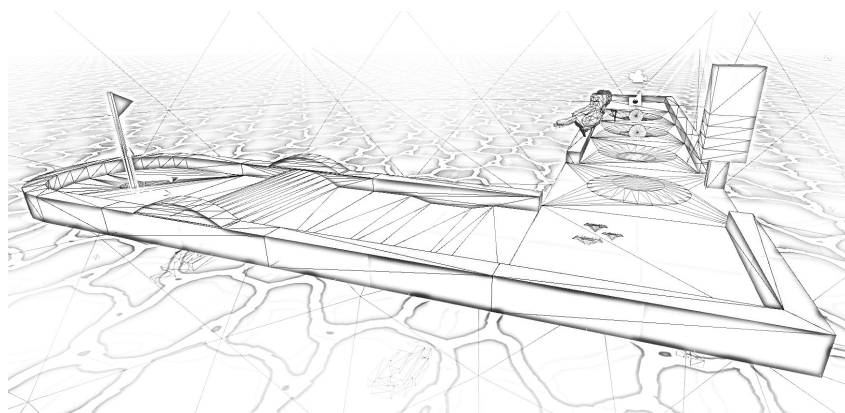
Présentation du projet

Notre premier jeu sera un jeu de golf 3D pour mobile. J'ai fait ce choix car c'est un type de jeu intéressant et qui plaira au plus grand nombre tout en étant réalisable sans trop de difficultés. Il est important pour moi que chaque lecteur puisse mener à bien ce projet. Un jeu de golf est relativement simple à développer tout en offrant la possibilité aux lecteurs les plus motivés de le faire grandement évoluer et de prendre des libertés pour aboutir à un projet professionnel.

Nous commencerons par concevoir quelques niveaux. La première étape sera de créer les parcours sur lesquels la balle de golf pourra se déplacer. Une fois que la structure des niveaux sera prête, je vous montrerai comment les décorer pour pousser un peu plus loin le level design. Pour certains niveaux, nous serons amenés à ajouter des obstacles animés afin d'ajouter de la difficulté.

Voici un croquis d'un exemple de niveau que nous pourrions réaliser :

Figure 34 : Croquis d'un niveau du jeu



Sur la Figure 34, vous pouvez voir le parcours avec quelques éléments décoratifs. Nous utiliserons des modèles 3D simples, des formes cubiques et des couleurs unies. Le sol sera vert et la bordure en bois. Vous pourrez bien entendu personnaliser votre jeu si vous souhaitez un visuel complètement différent.

Notre interface sera également simple. Un viseur pour s'orienter, une barre de puissance pour tirer la balle et un bouton PAUSE pour accéder au menu. Notre projet sera conçu

2

Création des niveaux

Dans ce chapitre, nous attaquons la création de notre jeu avec la conception d'un premier niveau. Nous ne nous occuperons pour l'instant que du visuel ; nous n'écrirons pas de code. Ce sera également l'occasion d'aborder les notions de solidité, de gravité et de collision.

Figure 2.1 : Exemple de niveau que nous allons réaliser



2.1. Utilisation des modèles 3D

Pour que vous puissiez créer un jeu avec plusieurs niveaux fonctionnels, je vous ai fourni dix modélisations de niveaux que vous trouverez dans le dossier `Assets/Modelles3D_Golf/Levels`. Pour utiliser l'un de ces modèles, il vous suffit de le glisser/déposer de la fenêtre de projet vers la scène comme sur la Figure 2.2.

7

Début du script de tir

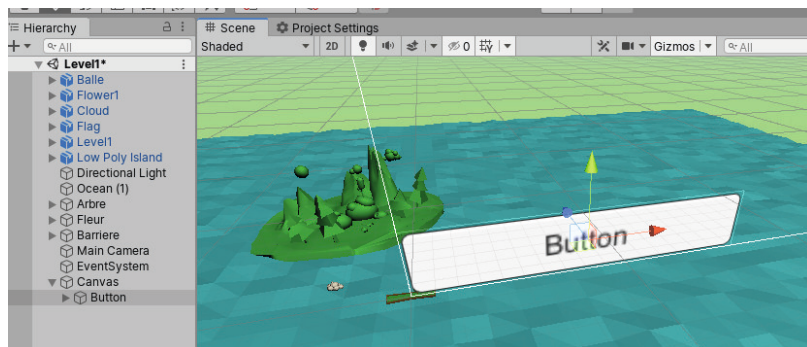
Nous allons maintenant écrire le script qui permettra de propulser la balle et commencer à nous faire une idée de ce que sera le gameplay. Le script sera lié à une interface utilisateur : son déclenchement et le réglage de la puissance du tir en dépendront. Afin d'avancer progressivement, étape par étape, nous nous concentrerons d'abord sur un tir unique (puissance non réglable). Une fois que nous aurons la base du script, nous créerons l'interface au chapitre *L'interface utilisateur* puis nous terminerons l'écriture de ce script avec toutes ses fonctionnalités au chapitre *Menu Pause et fin du script de tir*.

Dans ce chapitre, nous allons voir comment tirer la balle en face de la caméra à une puissance fixée. Nous ajouterons un bouton d'interface temporaire que nous utiliserons pour tirer la balle. Cela nous permettra de voir comment lui appliquer une force de propulsion.

7.1. Création de l'interface de test

Ce que j'appelle l'interface de test, c'est une interface utilisateur qui nous permettra de tester notre script mais que nous supprimerons après nos essais. Nous construirons une meilleure interface au chapitre suivant. Notre interface de test sera extrêmement basique puisqu'elle sera constituée d'un seul bouton. Pour créer ce bouton, cliquez sur le menu **GAMEOBJECT > UI > BUTTON**. Un bouton apparaît alors sur la scène.

Figure 7.1 : Création d'un bouton



II. Un FPS 3D pour PC

Présentation du projet

Dans ce module, nous allons développer un jeu de type FPS (jeu de tir à la première personne) comprenant les éléments indispensables de ce type de jeux : un niveau 3D, un système de déplacement, des armes et un inventaire, des objets à ramasser comme des trousse de soin, des ennemis à combattre et un objectif à atteindre.

Nous allons commencer par la création du monde 3D et la mise en place de l'éclairage. Ensuite nous ajouterons le personnage et nous écrirons le script de tir qui permettra de propulser des projectiles. Nous ajouterons des ennemis, nous mettrons en place une petite intelligence artificielle ainsi que la gestion de la vie et des collisions. Nous continuerons le développement pour ajouter des détails puis nous compilerons le projet afin de pouvoir le partager.

Figure 13 : Croquis (2D vue de haut) du niveau que nous allons créer



Sur ce croquis, vous pouvez voir les salles et couloirs du niveau ainsi que la position de quelques objets décoratifs. J'ai également indiqué la position des pièges, des portes, de

16

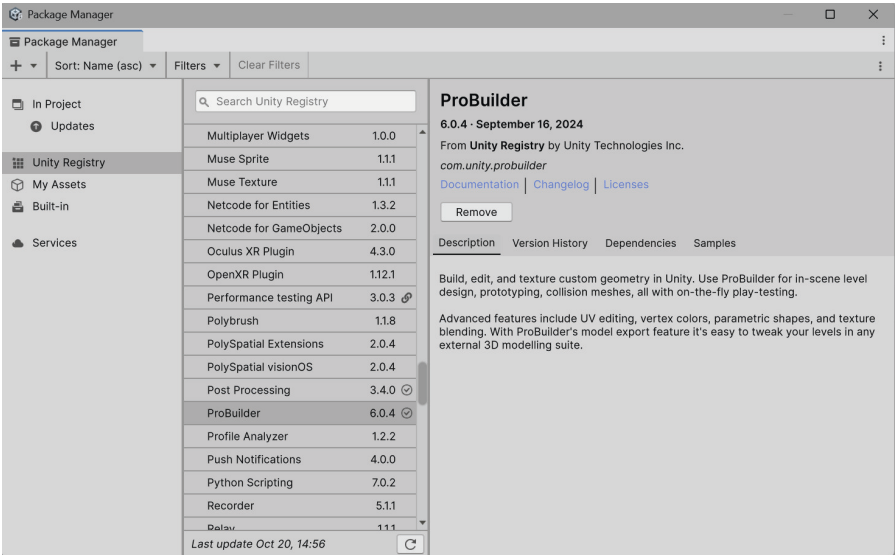
Découverte de ProBuilder

À l'origine, Unity ne permettait pas de faire de la modélisation 3D ; vous ne pouviez que positionner des modèles importés pour concevoir vos niveaux et la modélisation devait se faire sous un logiciel dédié de type Blender. Si je parle au passé c'est qu'aujourd'hui il existe une solution pour modéliser directement dans Unity : ProBuilder. L'outil a été développé de manière indépendante il y a quelques années et il a rencontré un tel succès que Unity Technologies l'a racheté et intégré de façon officielle au moteur de jeu.

16.1. Installation et présentation de ProBuilder

ProBuilder n'est pas directement utilisable dans un projet Unity. Vous devez au préalable l'installer en passant par le Package Manager [p. 54]. Cliquez donc sur le menu WINDOW > PACKAGE MANAGER pour ouvrir celui-ci, et dans la rubrique UNITY REGISTRY, recherchez ProBuilder et sélectionnez-le. Cliquez ensuite sur le bouton INSTALL.

Figure 16.1 : Installation de ProBuilder



22

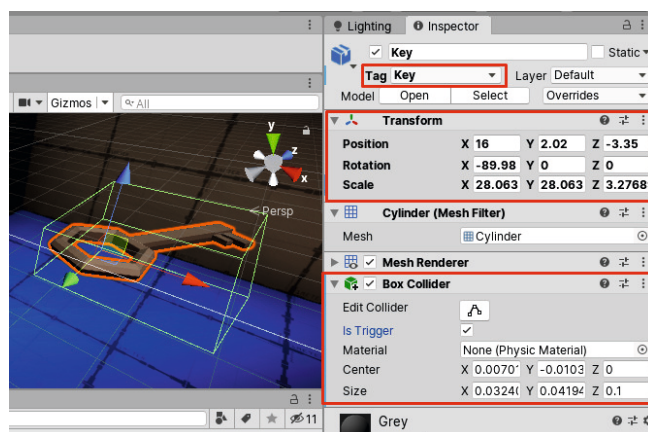
Système d'ouverture de porte

L'objectif du joueur sera de récupérer une clé afin d'ouvrir une porte derrière laquelle un portail magique se trouve. Ce portail est la fin du niveau. Pour freiner le joueur, des ennemis seront répartis sur le niveau. Dans le chapitre précédent, nous avons vu comment travailler avec les collisions, nous allons utiliser ces connaissances pour coder ce système de clé/porte.

22.1. Création de la clé

Ici c'est du classique ! La manipulation est exactement la même que pour le MediKit. Importez la clé dans la scène, ajoutez un Collider, cochez **ISTRIGGER**, ajustez la taille, créez un **TAG KEY** et appliquez ce tag à la clé... Voilà le résultat :

Figure 22.1 : Configuration de la clé



C'est tout ce que vous avez à faire pour créer votre clé. Dans le script de collision, nous pouvons la détecter et la ramasser. Afin de varier un peu les manipulations et de vous apporter de nouvelles notions, nous allons animer la clé par script. Créez donc un nouveau script que vous appellerez `RotateScript` et ajoutez-le à la clé.

24

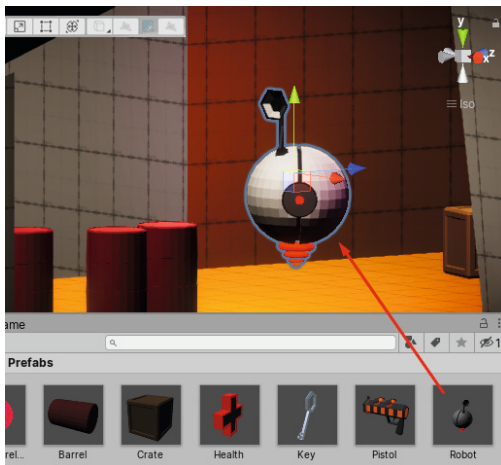
Des ennemis qui se déplacent

Dans ce chapitre, nous allons ajouter un peu de difficulté au jeu en créant des ennemis capables de se déplacer dans le niveau. Ils pourront ainsi croiser le joueur et le prendre par surprise car leur position ne peut pas être connue à l'avance.

24.1. Configuration de l'ennemi

Nous allons commencer par configurer notre ennemi en ajoutant les composants nécessaires. Dans cette section, je vais aller un peu vite sur les explications car cela sera très similaire à la configuration des tourelles automatiques. Commencez par ajouter un robot ennemi. Vous trouverez le modèle 3D avec les autres ressources.

Figure 24.1 : Ajout du robot volant



Ajoutez ensuite un SphereCollider qui doit englober tout le robot puis créez un objet vide, enfant du robot, que vous placerez juste devant l'œil avec l'axe bleu (Z) pointant devant le robot. Cet objet vide que vous pouvez appeler `Tir` sera la position d'instanciation des projectiles.