



CHECKLIST IMPRIMABLE

Unreal Engine 5 en solo : validez votre projet avant de vous lancer

Cette checklist vous aide à déterminer si Unreal Engine 5 correspond réellement à votre projet solo. Utilisez-la avant de multiplier les assets, les fonctionnalités visuelles ou les systèmes complexes.

Au sommaire

- 1 Le critère décisif
- 2 Mon projet est-il adapté à Unreal Engine 5 en solo ?
- 3 Construire un prototype utile en 5 étapes
- 4 Points de contrôle avant d'ajouter du contenu
- 5 Vérifications pour chaque asset importé

Le critère décisif

Unreal Engine 5 est pertinent pour un projet 3D limité, centré sur une boucle de jeu claire et une direction artistique maîtrisable.

Le moteur fournit des outils de production, mais ne remplace pas le temps nécessaire pour créer les niveaux, l'interface, le son, les tests et l'optimisation.

Un prototype jouable avec une mécanique complète est plus utile qu'un grand environnement sans objectif de jeu validé.

Mon projet est-il adapté à Unreal Engine 5 en solo ?

Je vise une expérience principalement en 3D.

Le moteur est particulièrement orienté vers la création de jeux en trois dimensions.

Je peux décrire la mécanique centrale de mon jeu en une phrase.

Elle doit pouvoir être testée rapidement dans une seule scène.

Mon premier objectif est une zone jouable limitée.

Par exemple : une arène, quelques salles de puzzle ou une courte séquence narrative.

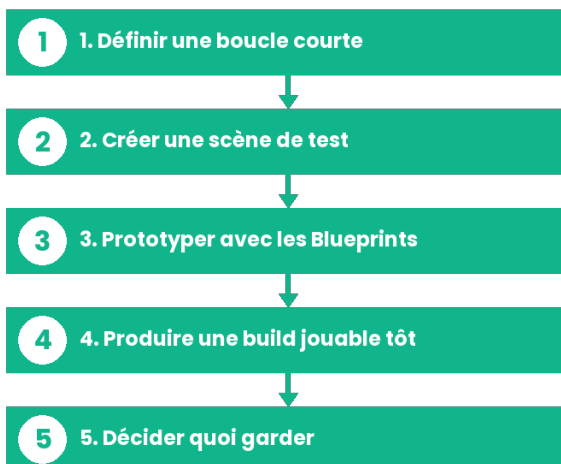
Je suis prêt à apprendre un outil dense et à structurer ma logique de jeu.

Les Blueprints rendent la programmation visuelle, sans supprimer les besoins de conception et d'organisation.

J'accepte de réduire le périmètre si le prototype ne tient pas ses promesses.

Le bon projet solo est celui que vous pouvez terminer, tester et améliorer.

Construire un prototype utile en 5 étapes



1. Définir une boucle courte

Choisissez une action principale, un objectif et un retour clair pour le joueur. Évitez d'ajouter plusieurs mécaniques avant d'en avoir validé une.

2. Créer une scène de test

Montez une zone minimale qui permet de jouer la boucle complète, plutôt que de commencer par une grande carte.

3. Prototyper avec les Blueprints

Reliez événements, conditions, variables et actions pour tester les interactions. Nommez clairement les variables et gardez les graphes lisibles.

4. Produire une build jouable tôt

Testez le jeu hors de l'éditeur : un projet fonctionnel dans Unreal Engine n'est pas automatiquement fluide dans une build destinée aux joueurs.

5. Décider quoi garder

Conservez les fonctions qui améliorent le jeu ou les choix du joueur. Reportez les effets et systèmes qui n'aident pas encore la boucle centrale.

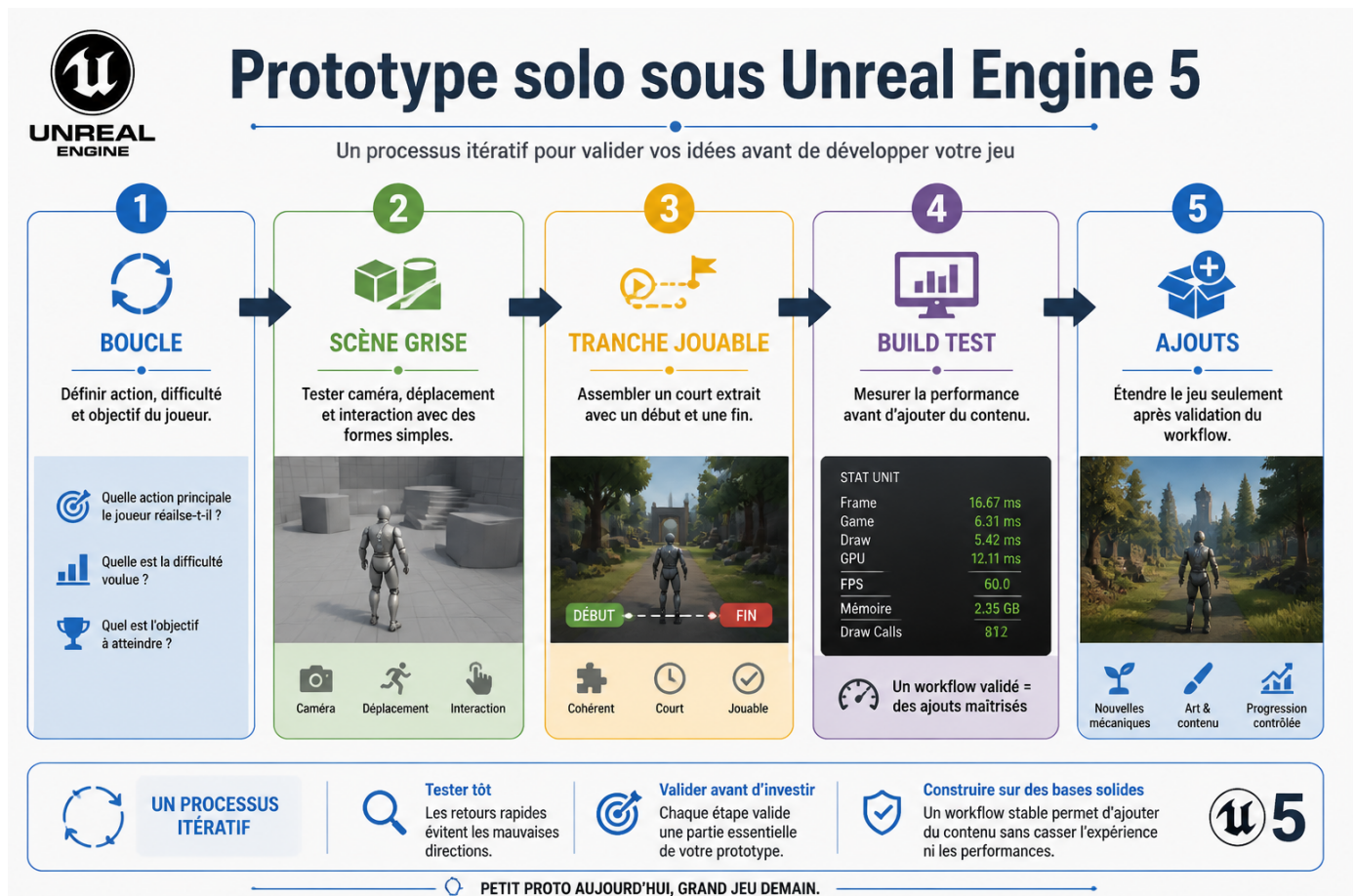


SCHÉMA Un chemin simple : idée de jeu, mécanique testable, prototype jouable, build et ajustements.

Points de contrôle avant d'ajouter du contenu

Élément	Question à se poser	Test concret
Blueprints	Le graphe reste-t-il compréhensible et maintenable ?	Créer une mécanique complète dans une scène.
Rendu et effets	Cette fonction améliore-t-elle réellement l'expérience de jeu ?	Mesurer la fluidité sur une build.
Assets	Cet asset sert-il une scène précise du projet ?	L'importer d'abord dans un projet de test.
Matériel	Ma machine reste-t-elle confortable pendant l'édition ?	Observer les imports, shaders, grandes cartes et builds.
Publication	Le jeu est-il jouable hors de l'éditeur ?	Produire et tester une build le plus tôt possible.

Vérifications pour chaque asset importé

- ☐ Vérifier la compatibilité annoncée avec la version d'Unreal Engine utilisée.
- ☐ Tester l'asset dans un projet séparé avant l'import dans le projet principal.
- ☐ Contrôler la licence et les éventuelles dépendances.
- ☐ Évaluer le poids des textures et des maillages.
- ☐ Prévoir l'harmonisation de l'échelle, des matériaux et du style visuel.
- ☐ Écarter l'asset s'il n'améliore pas une scène ou une mécanique précise.

! À reporter pour un premier projet solo

Un monde ouvert, un jeu de rôle long, un multijoueur compétitif ou une génération procédurale complexe cumulent des systèmes, du contenu et des tests difficiles à assumer seul. Validez d'abord une expérience courte et complète.

Pour les prérequis matériels et les fonctions disponibles, consultez la documentation d'Epic Games correspondant à la version exacte d'Unreal Engine installée.