

Audit critique et constructif des objectifs stratégiques L1 d'un développeur Unreal Engine indépendant

Pertinence, réalisme et cohérence chronologique des objectifs

Objectif majeur 1 – Générer des revenus d'ici mai 2026 : Cet objectif de court terme est pertinent car il concrétise l'ambition de **monétiser rapidement les compétences** via la vente d'assets techniques et des missions freelance. Il s'aligne avec l'invariant de « Réutilisabilité et Valeur (Pragmatisme Financier) » du système agentique, qui prône la création d'éléments monétisables (assets **vendables sur Marketplace**) plutôt que de simples projets non exploitables ¹. D'un point de vue **réalisme**, viser des revenus en quelques mois (d'ici mai 2026) est ambitieux mais pas impossible. Les premiers gains risquent d'être modestes (quelques dizaines ou centaines d'euros) avant de monter en puissance. Par exemple, un créateur d'assets Unreal a rapporté que ses **premiers mois de ventes** n'ont généré que ~65 \$ puis ~35 \$ ². Ce n'est qu'après plusieurs assets publiés et une participation à des promotions qu'il a vu ses revenus mensuels dépasser **600 \$ dès le 3^e mois**, confirmant l'existence d'une demande pour ses produits ³. Cela montre qu'il faut s'attendre à une **progression graduelle** – des petits revenus initiaux servant de preuve de concept et de motivation, suivis d'une hausse significative si l'offre s'étoffe et gagne en visibilité.

Chronologiquement, cet objectif de revenus en mai 2026 est cohérent comme **première étape** : il fixe une **date butoir proche** pour obtenir des résultats financiers tangibles, ce qui crée un **sentiment d'urgence constructive**. Cela incite dès maintenant à publier les premiers assets et à décrocher de petites missions freelance pour amorcer le chiffre d'affaires. Ce jalon intermédiaire (mi-2026) s'insère logiquement avant l'objectif final de professionnalisation, fonctionnant comme un **test de viabilité** du modèle économique choisi.

Objectif majeur 2 – Professionnalisation d'ici fin 2027 : Cet objectif de long terme vise à ce que le développeur soit **pleinement professionnel** d'ici fin 2027, c'est-à-dire vivre principalement de son activité indépendante (vente d'assets, missions rémunérées et éventuellement produits issus des MVP). La pertinence est indéniable – il s'agit du **but ultime** de la démarche, en accord total avec la mission L0 de construire une carrière de game developer autonome et monétisable. Sur le plan du **réalisme**, un horizon ~2 ans permet de progresser par itérations et paliers, ce qui est raisonnable. De nombreux développeurs indépendants mettent **plusieurs années** à atteindre des revenus soutenables. Avec une approche méthodique, 2026 servira à bâtir les bases (portfolio d'assets, premiers clients, projets pilotes), et 2027 à **accélérer et consolider**. Par exemple, le créateur d'assets mentionné précédemment a pu atteindre ~15 000 \$ de ventes la première année en combinant **multiplication des assets et opérations marketing** ⁴. Ce niveau de revenu annuel reste à compléter par du freelance pour un salaire équivalent temps plein, mais il démontre qu'en deux ans il est plausible de se rapprocher d'un seuil professionnel. Néanmoins, il faut garder à l'esprit la **compétition croissante** sur les marketplaces : un vendeur expérimenté note une baisse des ventes depuis 2022 due à la saturation du marché (trop d'assets disponibles, visibilité réduite) ⁵. Se professionnaliser exigera donc une **montée en qualité et en notoriété** continue pour se démarquer.

Sur le plan **chronologique**, viser la professionnalisation fin 2027 est cohérent comme **aboutissement** de la trajectoire initiée en 2026. Les deux objectifs majeurs forment une suite logique : **d'abord générer un revenu initial (2026)** pour prouver le concept et bâtir un socle financier, **puis élargir et stabiliser ces revenus (2027)** afin d'en faire une activité professionnelle à part entière. Il sera important que les objectifs intermédiaires (voir suggestions plus bas) rythment ce parcours, afin de mesurer à mi-parcours (par ex. mi-2027) si l'on est sur la bonne voie vers la cible finale. Globalement, les échéances proposées (mi-2026 puis fin-2027) sont **cohérentes** entre elles et avec la vision stratégique : elles traduisent une progression du **court terme vers le long terme**, en laissant environ 18 mois entre les deux pour monter en puissance.

Autres objectifs stratégiques L1 (publication d'assets, missions freelance, projet MVP) : La stratégie décrite repose sur trois **piliers** – vendre des assets techniques, effectuer des missions freelance, et développer un projet fondateur (MVP). On peut s'attendre à ce que la liste d'objectifs L1 inclue des cibles associées à chacun de ces piliers, par exemple : « *Publier [N] assets de qualité d'ici [date]* », « *Accomplir [N] missions freelance en [2026/2027]* », et « *Réaliser un MVP de [projet] d'ici [date]* ». Ces objectifs seraient pertinents car ils sont les **leviers concrets** pour atteindre les deux objectifs majeurs de revenus et de professionnalisation. Il faudra toutefois vérifier leur **réalisme individuel** : publier un asset demande du temps (conception, implémentation, tests, validation sur la marketplace), tout comme trouver des missions freelance (nécessite un réseau ou une présence sur des plateformes). Par exemple, s'engager à publier un asset **tous les 2 mois** en 2026 peut être ambitieux mais tenable si les assets sont de petite à moyenne taille. En revanche, promettre un asset par semaine serait irréaliste et risquerait de sacrifier la qualité (piège du « *Quick & Dirty* » interdit par le système ⁶). De même, se fixer un nombre de missions freelance suppose de pouvoir dégager assez de temps tout en développant des assets – il faudra équilibrer ces activités pour éviter les surcharges. Enfin, l'objectif lié au projet fondateur (MVP) devra être cohérent dans le calendrier : il serait judicieux que le MVP soit amorcé une fois les premières sources de revenus établies, puis abouti en 2027* pour servir de tremplin vers la professionnalisation (par ex. une démo jouable ou un produit minimum viable pouvant être monétisé ou présenté à des investisseurs/éditeurs en fin de parcours).

En somme, la liste d'objectifs L1 semble globalement **pertinente et cohérente** avec la mission et les moyens identifiés. Les deux jalons temporels principaux (mi-2026 et fin-2027) forment une colonne vertébrale logique. Il conviendra de s'assurer que chaque objectif soit **quantifié et réalisable** dans le délai imparti, et de garder de la flexibilité pour s'adapter en cas d'obstacles (marché plus difficile que prévu, retard sur un asset, etc.). La progression devra être régulièrement évaluée afin de valider le réalisme en cours de route et ajuster si besoin.

Reformulation et clarification des objectifs pour les rendre actionnables

Pour rendre les objectifs **actionnables, mesurables et intégrables** dans le système agentique, on propose de les reformuler en termes **SMART** (Spécifiques, Mesurables, Atteignables, Réalistes, Temporellement définis). Voici des reformulations/clarifications des objectifs stratégiques L1 pressentis :

- **Objectif L1 : Atteindre un revenu mensuel régulier d'ici mai 2026.** Au lieu de « revenus en mai 2026 » (formulation vague), on peut spécifier **un montant cible** ou un palier clair. Par exemple : « *Générer au moins 500 € de revenus par mois d'ici fin mai 2026 en combinant la vente d'assets et les missions freelance.* » Cela rend l'objectif mesurable (500 € mensuels récurrents), tout en restant adaptable (peu importe la proportion assets/freelance pourvu que le total y soit). Le choix du montant doit être calibré selon la situation du développeur (500 € n'est qu'un exemple modéré –

on ajustera en fonction des besoins financiers ou de la capacité de production estimée). L'important est d'avoir un **seuil quantitatif** pour que l'agent puisse suivre l'indicateur et évaluer le succès en mai 2026.

- **Objectif L1 : Publier des assets techniques de qualité sur la Marketplace Unreal.** Plutôt que «*publication d'assets techniques*» de façon générale, on clarifie le **nombre et l'échéance**. Par exemple : «*Publier 3 assets techniques originaux sur l'Unreal Marketplace d'ici décembre 2026, dont au moins un avant fin mars 2026.*» Cette formulation décompose la cible : un premier asset très tôt (fin T1 2026) pour entrer sur le marché et apprendre du processus, puis deux autres d'ici fin 2026 pour étoffer le catalogue. Chaque asset devrait être défini (type d'asset, complexité) en L2/L3, mais au niveau stratégique L1 on fixe juste le volume et la deadline. On insiste sur la **qualité** («*assets techniques de qualité*») pour ne pas sacrifier les standards techniques et aligner avec les valeurs du système (pas de compromis sur la cohérence technique et la documentation). Un nombre raisonnable comme 3 assets en ~1 an permet de les planifier sans précipitation, tout en donnant suffisamment de produits sur la Marketplace pour générer des revenus diversifiés.
- **Objectif L1 : Réaliser des missions freelance pour diversifier les revenus.** Au lieu de simplement «missions freelance», on peut cibler un **nombre de projets ou un chiffre d'affaires**. Par exemple : «*Effectuer au moins 2 missions freelance significatives d'ici fin 2026 (ex : développement de fonctionnalité UE5 pour un client, ou prestation technique), générant au total X € de CA cumulé.*» Cela rend l'objectif vérifiable (deux missions minimum complétées) et lie aussi une notion de revenu cumulé cible pour le freelance. X € peut être précisé selon les tarifs pratiqués (ex: viser 5 000 € sur l'année en freelance, ajustable selon temps disponible). On pourrait aussi spécifier **la nature** des missions attendues (par ex. «*une mission en 2026 portant sur un plugin Unreal, une autre sur un level design technique*» selon la spécialisation voulue), mais au niveau L1 on reste généralement plus macro. Cet objectif assure que l'effort ne soit pas que sur les assets (revenus passifs incertains) mais aussi sur du **service actif** apportant de l'expérience client et du cash plus immédiat.
- **Objectif L1 : Développer un projet fondateur (MVP) d'ici 2027.** Plutôt que «*projets fondateurs (type MVP)*» formulé génériquement, il vaut mieux définir quel projet et quelle échéance. Par exemple : «Concevoir et développer un **MVP jouable** d'un projet de jeu ou d'outil original d'ici **juin 2027**, servant de vitrine de savoir-faire et/ou de base à un produit commercialisable.» Cette reformulation précise le *livrable* (un *MVP jouable* ou une *démo fonctionnelle*), la *finalité* (*vitrine de compétences, base d'un produit*) et la *date butoir* (*mi-2027, afin d'avoir le temps de l'itérer avant fin 2027 pour éventuellement le lancer ou le présenter*). On pourrait même nommer le projet s'il est déjà identifié (par ex. «... un MVP du jeu X d'ici juin 2027»). Cela rend l'objectif plus concret pour l'agent : il saura qu'il y a un gros projet à découper en tâches sur 2026-2027. À noter : il peut être utile d'accorder à cet objectif des critères de réussite (par ex. «*MVP testé par 50 utilisateurs ou présenté en concours*»*) pour mesurer son impact, mais cela pourrait être affiné en sous-objectifs.
- **Objectif L1 : Atteindre le statut de développeur indépendant professionnel d'ici décembre 2027.** Au lieu de «*professionnalisation d'ici fin 2027*» un peu abstrait, on peut décrire les critères tangibles qui définiront la réussite. Par exemple : «D'ici fin 2027, dégager un **revenu annuel net équivalent à un salaire à temps plein junior** grâce à l'activité indépendante (assets + freelance + produits), et officialiser le statut professionnel (création d'entreprise ou portage salarial) pour pérenniser l'activité.» Cette formulation inclut deux éléments mesurables : le *niveau de revenu* (on peut préciser un montant, e.g. «~20 000 € annuels» si on vise un SMIC net annuel, ou ajuster selon le contexte de vie du développeur) et un élément qualitatif/réglementaire (*immatriculation de l'activité, ce qui marque la professionnalisation vis-à-vis de l'administration et des clients*). Ainsi, l'objectif n'est pas juste «être pro» dans l'absolu, mais bien gagner sa vie de cette activité et l'avoir structurée

*juridiquement. C'est parfaitement en ligne avec la mission initiale de "production de projets, d'actifs et de systèmes monétisables"**⁷ et la vision d'une carrière pilotée de manière autonome.

En reformulant de la sorte, on obtient des objectifs stratégiques **clairs et actionnables**. Chacun comporte un **verbe d'action**, un **livrable ou indicateur chiffré**, et une **échéance**. Cela facilite leur intégration dans le système agentique : l'agent "Planner" pourra aisément vérifier l'état d'avancement (par exemple, «Asset 1 publié ? oui/non ; combien de revenus générés ce mois ? X € vs objectif Y €»). De plus, ces formulations mettent en lumière les attentes en termes de **qualité** ("de qualité", "significatives", "jouable") pour cadrer l'exécution, ce qui est important dans un système prônant la rigueur technique et la traçabilité. Si certaines de ces clarifications n'étaient pas présentes dans la liste initiale, il est recommandé de les intégrer afin de **lever toute ambiguïté** et d'assurer que les objectifs L1 puissent être déclinés en plans d'action concrets par l'agent.

Suggestions d'objectifs complémentaires ou de points à éclaircir

En examinant la stratégie globale, voici quelques **objectifs manquants ou points flous** qu'il serait bénéfique d'ajouter ou préciser, pour renforcer la feuille de route :

- **Définir des paliers intermédiaires de progression** : Entre les deux grandes échéances (mi-2026 et fin-2027), il serait judicieux d'introduire des **milestones** intermédiaires qui servent de points de contrôle. Par exemple : «*Atteindre 50 % du revenu cible de 2027 d'ici mi-2027*» (soit, si on vise ~20k€ annuels fin 2027, atteindre ~10k€ annualisés à l'été 2027). Ou encore : «*Avoir publié au moins 2 assets et réalisé 1 mission freelance d'ici Q1 2027*». Ces paliers chiffrés fonctionnent comme des **étapes de supervision** : l'agent pourra comparer les réalisations aux prévisions à mi-parcours. S'ils ne sont pas atteints, cela déclenchera des ajustements (par exemple intensifier le marketing ou réviser l'allocation de temps entre dev d'assets et freelance). Sans ces jalons, on risque de découvrir trop tard un écart par rapport à la trajectoire. Ils rendent aussi la progression plus gratifiante, en célébrant chaque étape franchie.
- **Intégrer des objectifs d'amélioration continue du système agentique lui-même** : Puisque le développeur s'appuie fortement sur un **système agentique IA** (Obsidian + Antigravity), on peut formuler des objectifs liés à cet outil pour s'assurer qu'il évolue de pair avec les besoins. Par exemple : «*Doter le système agentique d'indicateurs de performance d'ici [date] pour mesurer la productivité (ex: nombre de tâches accomplies par semaine, vitesse de développement) et la qualité (ex: pourcentage de code documenté, respect des invariants)*». Ou «*Mettre en place d'ici fin 2026 un rapport mensuel automatisé généré par l'agent, résumant les progrès sur chaque objectif L1 et les risques ou dérives éventuels*». Ce type d'objectif explicite le fait que l'on attend de l'agent non seulement qu'il aide ponctuellement, mais qu'il fournit des **livrables concrets** (tableaux de bord, analyses) pour piloter la carrière. Cela rejoint l'idée du système comme «*tableau de bord de progression personnelle et professionnelle*»⁸ et assure que l'agent devienne un véritable **copilote proactif**.
- **Préciser les livrables "agentiques" attendus** : En prolongement du point précédent, il peut être utile de lister **quels outputs concrets l'IA doit produire** dans le cadre du workflow. Par exemple : «*Générer avec l'agent Planner un plan de projet L2/L3 pour chaque nouvel asset avant sa réalisation (design doc, architecture technique validée)*» – cela garantit qu'aucun asset ne démarre sans une phase de conception formalisée, conformément au principe "Architecture First"⁹. Ou «*Pour chaque mission freelance, utiliser l'agent pour documenter les décisions techniques et lessons learned dans le vault Obsidian (post-mortem)*». En inscrivant de tels points dans les objectifs, on **verrouille la discipline** d'utilisation de l'IA comme assistant structurel. Ces

livrables agentiques spécifiques forceront l'adoption complète des outils (pas juste en théorie), et faciliteront la montée en compétence accélérée (puisque l'IA pourra capitaliser la connaissance dans la base documentaire).

- **Ajouter des KPI de supervision de l'IA et de la qualité** : Le système peut bénéficier de **Key Performance Indicators** qui suivent l'efficacité de l'IA et le respect des bonnes pratiques. Par exemple : « *Taux de décisions validées par revue humaine : 100 % des décisions stratégiques (L1) et architecturales (L2) doivent faire l'objet d'une validation humaine explicite* » (c'est déjà un invariant du système ¹⁰ , on peut en faire un indicateur à surveiller pour éviter toute dérive d'automatisation incontrôlée). Ou « *Réduire à <5 % le nombre de tâches marquées "urgentes" contournant le process standard (pas de contournement des lignes rouges)* ». Un autre KPI possible : « *Coverage documentaire : 100 % des assets et projets ont une documentation L2/L3 générée et à jour dans Obsidian* ». Enfin, pour l'IA elle-même : « *Taux d'adoption des suggestions de l'agent : X % des recommandations pertinentes de l'IA sont implémentées* » (ce qui signifierait qu'on utilise effectivement l'IA comme amplificateur, sans pour autant suivre aveuglément tout ce qu'elle propose). Ces KPI devraient être ajoutés en complément des objectifs L1, ou associés à eux, pour **quantifier la qualité du processus** autant que le résultat brut. L'agent pourra en faire le suivi automatique et alerter en cas de déviation (par exemple, si la documentation d'un projet est incomplète, ou si une décision L2 a été prise à la hâte sans validation).
- **Élargir l'objectif “publication d'assets” en intégrant la dimension marketing/visibilité** : Compte tenu des retours de la communauté, la réussite sur les marketplaces dépend aussi de la promotion. Il serait judicieux d'ajouter un sous-objectif du style : « *Construire une présence en ligne pour soutenir la diffusion des assets (par ex. chaîne YouTube avec tutoriels, compte Twitter ou LinkedIn actif, participation à des forums) d'ici fin 2026 afin de fidéliser une communauté autour de ses produits.* » En effet, un créateur souligne que la **qualité seule ne suffit pas, le marketing est clé** (réalisation de vidéos de démonstration, posts communautaires, etc.) ¹¹ . De même, sur Reddit on apprend que seuls ceux qui **investissent du temps en publicité/communauté** ou qui étaient là tôt tirent leur épingle du jeu sur la marketplace ¹² . Intégrer cet aspect dans les objectifs incitera le développeur à **consacrer du temps à la visibilité** (ce qui est souvent négligé mais crucial). Concrètement, cela pourrait être mesuré via un KPI (nombre d'abonnés, de vues, ou taux de conversion des pages assets).
- **Clarifier le périmètre du “projet fondateur (MVP)”** : Si le terme MVP reste flou, on pourrait détailler ce qu'il recouvre. Par exemple, préciser s'il s'agit d'un **jeu vidéo complet (même court)**, d'un **plugin/outil à vendre**, ou d'une **démo technologique**. Également, décider si un seul MVP est visé ou plusieurs petits projets fondateurs (*le pluriel « projets fondateurs » suggère peut-être plusieurs prototypes*). Si plusieurs MVP/projets sont envisagés, on pourrait en faire un objectif à part entière : « *Réaliser 2 projets prototypes (mini-jeux ou outils tech) entre 2026 et 2027, dont au moins un aboutissant à un MVP publiable.* » L'idée est de **cadrer l'effort R&D/créatif** pour qu'il ne dilue pas l'objectif business. Un MVP bien choisi devrait idéalement être **synergique** avec le reste (par ex. réutiliser des assets créés, ou démontrer une compétence qui attire des clients freelance). Il faut donc s'assurer que l'objectif lié au MVP précise ces attentes (réutilisation d'actifs existants, démonstration d'une technologie innovante, etc.), conformément au principe de ne pas créer de **silos isolés** mais des éléments réutilisables et cohérents ¹³ .

En incorporant ces suggestions, la liste des objectifs L1 gagnerait en **exhaustivité et en précision**. On couvrirait ainsi non seulement **le quoi et quand** (revenus, contenus, échéances), mais aussi **le comment** en filigrane (qualité, usage de l'IA, promotion...). Cela fournit un cadre beaucoup plus robuste pour guider l'agent et éviter les “angles morts” dans la stratégie. Naturellement, il faut veiller à ne pas submerger L1 d'objectifs trop détaillés : certains des points ci-dessus peuvent aussi être formulés au

niveau L2 (sous-objectifs stratégiques) ou L3 (objectifs tactiques). L'important est que rien d'important ne soit oublié et que les attentes soient **explicites** à un certain niveau du système, pour que l'IA puisse « tout voir » et analyser les corrélations transversales L1-L6 comme prévu ¹⁴.

Découpage en phases / groupes logiques (Roadmap structurée)

Pour donner une vue d'ensemble temporelle et **structurer la roadmap**, on peut répartir ces objectifs en **phases successives**, chacune avec un focus et des livrables clés. Voici une proposition de découpage logique :

- **Phase 1 : Lancement et premiers revenus (T1-T2 2026)** – *Objectif central : amorcer la monétisation.* Durant cette phase initiale (jusqu'en mai 2026 environ), l'accent est mis sur la **mise en place** et les **premières réalisations tangibles** :
- **Mise en place de l'environnement** : finaliser le setup du système agentique (Obsidian, Antigravity) et s'assurer que l'infrastructure de travail est opérationnelle. Le L0 est verrouillé, on décline maintenant en L1/L2. Parallèlement, préparer le terrain côté business (ouverture de compte vendeur sur l'Unreal Marketplace, profils sur plateformes freelance, etc.).
- **Publication du 1er asset** : concevoir, développer et publier un premier asset technique avant la fin du premier trimestre 2026. Ce premier essai permettra d'apprendre le processus de submission sur la Marketplace et de commencer à se faire connaître. Succès attendu : au moins quelques ventes et retours utilisateurs pour orienter la suite.
- **Acquisition d'une première mission freelance** : en parallèle, prospecter et décrocher au moins un contrat freelance court avant mai 2026. Par exemple, une mission de quelques semaines pour un client (intégration d'une feature, conseil technique...). Livrer cette mission avec satisfaction client afin de bâtir une référence et possiblement un témoignage pour le portfolio.
- **Atteinte des premiers revenus** : grâce à la vente du premier asset et à la mission freelance rémunérée, viser l'atteinte du **palier de revenus de mai 2026**. Même si le montant est modeste, l'objectif est d'enclencher le flux de trésorerie. On s'attend par exemple à encaisser une somme cumulative de l'ordre de quelques centaines d'euros sur la phase, preuve que le modèle commence à générer de la valeur monétaire.
- **Retour d'expérience et ajustements** : clôturer la phase par une **revue critique** (idéalement assistée par l'agent Planner) des résultats obtenus. Identifier ce qui a bien fonctionné (type d'asset apprécié ? client satisfait prêt à recommander ?), et les difficultés rencontrées (délais de validation, imprévus techniques, etc.). Ces leçons alimenteront la phase 2.
- **Phase 2 : Montée en puissance et diversification (T3 2026 – T2 2027)** – *Objectif central : augmenter le catalogue et les revenus, préparer le terrain pour le MVP.* Cette phase intermédiaire s'étend grossièrement du **second semestre 2026 au milieu 2027**. On profite de l'élan initial pour **accélérer et diversifier** les activités :
- **Production régulière d'assets** : fort de l'apprentissage du premier asset, planifier et réaliser la publication de **plusieurs assets additionnels** tout au long de la phase. Par exemple, sortir le 2^e asset d'ici fin 2026, puis un 3^e au premier trimestre 2027. Chacun doit apporter une valeur ajoutée (idéalement dans une thématique connexe pour commencer à établir une **réputation dans un domaine** – par ex. plusieurs packs d'effets visuels, ou plusieurs plugins Blueprints utiles). L'agent veillera à cadencer ces productions pour tenir le rythme sans sacrifier la qualité.
- **Expansion de l'activité freelance** : capitaliser sur la première mission réussie pour en obtenir d'autres, éventuellement plus ambitieuses. Objectif : réaliser **1 à 2 missions freelance supplémentaires** d'ici mi-2027. Cela peut impliquer de monter en gamme (ex. mission plus longue, ou pour un client plus important) afin d'augmenter le revenu service. Pendant cette

phase, le développeur doit aussi **élargir son réseau** (fréquenter des communautés Unreal, LinkedIn, forums) pour gagner en visibilité en tant que prestataire.

• **Community-building & marketing** : c'est durant cette phase qu'il faudra intensifier les **efforts de visibilité**. Par exemple, après avoir accumulé 2-3 assets, lancer une **chaîne YouTube ou un blog** pour présenter les assets en action, publier des tutoriels ou devblogs. Engager la communauté (ex : partager régulièrement des GIF ou vidéos sur Twitter avec les hashtags Unreal, participer aux threads Reddit en mentionnant ses assets quand pertinent, etc.). L'agent peut planifier ces tâches récurrentes (posts mensuels, vidéos trimestrielles) pour structurer la démarche marketing.

• **Début du projet MVP** : une fois quelques assets en vente et des revenus en hausse, la phase 2 est le bon moment pour entamer le **projet fondateur (MVP)**. On démarre par la définition précise du concept (L2 architecture du projet), puis on alloue du temps de développement régulier sur ce MVP sans négliger le reste. L'objectif est d'avancer significativement sur le MVP d'ici mi-2027, par exemple avoir une **première version jouable ou démontrable**. L'agent Planner pourra aider à découper le développement du MVP en tâches incrémentales insérées entre les travaux rémunérateurs, pour garder un équilibre.

• **Suivi des métriques et ajustements** : à la fin de Phase 2 (disons vers **juin 2027**), on effectue un **bilan d'étape majeur**. C'est le moment de vérifier les KPI intermédiaires : nombre d'assets publiés (atteint ou non), revenus mensuels actuels vs. objectif final, missions freelance effectuées, avancement du MVP (est-on proche d'une démo ?). En fonction de ce bilan, on ajuste la dernière phase. Par exemple, si les revenus passifs sont en deçà des attentes, on pourrait prioriser une sortie d'asset supplémentaire avant la fin 2027, ou intensifier la promotion. Ce rétrocontrôle est crucial pour aborder la phase finale en maximisant les chances d'atteindre la professionnalisation.

• **Phase 3 : Consolidation et professionnalisation (T3 – T4 2027) – Objectif central : aboutir à une activité durable et officiellement professionnelle**. La dernière phase, couvrant environ le second semestre 2027, est consacrée à **finaliser les objectifs et formaliser le passage au stade professionnel** :

• **Finalisation du MVP et lancement** : c'est le sprint final pour le projet fondateur. L'objectif est de *terminer le MVP* et de le mettre en valeur. Selon la nature du projet, cela peut être : publication du jeu en early access ou sur itch.io, présentation du prototype à un concours (Epic MegaGrants, game showcase...), ou démarchage d'éditeurs/clients avec cette démo. Le MVP doit jouer son rôle de **vitrine** à ce stade, donc on investira du temps dans une présentation soignée (trailer, page produit, documentation technique si c'est un plugin...). Fin 2027, le développeur devrait pouvoir montrer ce projet comme l'aboutissement de son savoir-faire, ce qui assoit sa crédibilité professionnelle.

• **Atteinte du revenu cible professionnel** : durant cette phase, on vise à **boucler les chiffres**. Grâce aux assets accumulés (qui génèrent désormais un revenu mensuel récurrent plus conséquent) et aux missions freelance de plus grande envergure, l'objectif est d'atteindre ou de dépasser le **seuil de revenu annuel visé**. Par exemple, si on visait ~20k€/an, il faut que les ventes d'assets + contrats réalisés en 2027 permettent d'atteindre cette somme. L'agent aura suivi mois par mois la tendance ; en phase 3 on peut pousser des **actions correctives de dernière minute** si nécessaire (ex : une promo spéciale sur les assets en fin d'année 2027 pour doper un peu les ventes, ou une ultime mission freelance si on a du temps Q4 2027).

• **Création du statut juridique / officialisation** : La professionnalisation passe par des démarches concrètes comme posséder un **statut d'entreprise** (micro-entreprise, SARL, etc.). Durant cette phase, idéalement une fois assuré que les revenus sont là, le développeur peut entamer ces démarches administratives. L'objectif est que pour fin 2027, il soit **officiellement**

enregistré en tant qu'indépendant (si ce n'était pas déjà fait plus tôt) – cela permet de facturer proprement, d'être en règle et de communiquer en tant que professionnel établi.

- **Structuration long-terme** : c'est aussi le moment de formaliser les **processus pérennes**. Par exemple, établir une routine de travail soutenable pour continuer après 2027 (répartition entre création d'assets, travail client, entretien du produit issu du MVP). L'agent peut aider à écrire une **feuille de route 2028+** sur la base des leçons apprises. On vise à ce que la carrière ne s'arrête pas à l'objectif atteint, mais dispose d'un plan de continuité (ex: roadmap de nouveaux assets, version 2 du MVP ou nouveau projet, etc.).
- **Évaluation finale et rétroaction** : en décembre 2027, l'objectif est de réaliser un **audit final** (peut-être avec le concours de l'agent en mode analyste) pour vérifier que tous les objectifs L1 sont accomplis ou en très bonne voie. C'est l'occasion de mesurer l'écart entre la vision initiale et la réalité atteinte, et d'**ancrer la réussite**. Si tout est vert, c'est la confirmation de la professionnalisation. Les quelques objectifs non atteints à 100 % (il y en aura peut-être) pourront soit être replanifiés début 2028, soit abandonnés si devenus non pertinents. L'important est d'acter la **fin de la phase de transition** : le développeur est maintenant un professionnel à part entière, prêt pour de nouvelles aventures avec l'expérience accumulée.

Ce découpage en **trois grandes phases** – **initialisation, montée en puissance, consolidation** – offre une vue claire pour la roadmap. Chaque phase a une **intention dominante** et des livrables clés, ce qui facilite la planification par l'agent. Bien sûr, ce plan n'est pas figé : il doit rester **adaptable**. Par exemple, si les revenus arrivent plus vite que prévu, la phase 2 pourrait être écourtée ; si au contraire un retard survient, on ajustera les objectifs de phase 3 en conséquence. L'essentiel est d'avoir une **structure directrice** pour ne pas se perdre, tout en gardant assez de souplesse pour intégrer les imprévus (opportunité soudaine d'un gros contrat, besoin de pivot si un type d'asset ne vend pas, etc.).

Utilisation de cette structure par l'agent *Planner* pour le plan opérationnel

Avec des objectifs ainsi clarifiés et phasés, l'agent persona "**Planner**" pourra jouer pleinement son rôle de **chef d'orchestre opérationnel**. Voici comment il exploitera cette structure pour décliner un plan d'action efficace :

- **Déclinaison multi-niveaux des objectifs** : À partir des objectifs stratégiques L1 validés (clairs, mesurables et ordonnancés dans le temps), l'agent *Planner* va procéder à une **décomposition hiérarchique**. Chaque objectif L1 sera analysé et traduit en objectifs de niveau inférieur :
- Il identifiera les besoins en décisions **L2 (Architecture/Design)** pour chaque projet découlant des L1. Par exemple, pour l'objectif de publier un asset, l'agent s'assurera qu'un document de conception (L2) soit créé décrivant l'architecture de l'asset, afin de respecter l'approche «*Architecture First*»⁹. De même, pour le MVP, il orchestrera la création d'une architecture logicielle détaillée (peut-être avec l'aide d'un agent "Architecte" spécialisé si le système en prévoit un).
- Ensuite, il va planifier des **objectifs tactiques L3/L4**, typiquement sous forme de tâches ou sprints de développement. Par exemple, l'objectif "Publier Asset #1 d'ici mars 2026" sera décliné en tâches L4 : *concevoir les features de l'asset (User Story 1, 2, 3), coder les fonctionnalités, tester l'asset en conditions réelles, préparer le matériel marketing (captures, vidéo), soumettre à l'Epic Marketplace*, etc., avec pour chacune une date cible interne. L'agent distribuera ces tâches sur le calendrier de Jan-Mars 2026 en calculant la charge de travail et les dépendances.
- Pour les missions freelance, le *Planner* intégrera aussi les **jalons contractuels** (dates de livraison intermédiaires convenues avec le client, par ex.) dans le plan global afin de gérer les éventuels conflits d'agenda entre travail pour client et développement des assets/MVP.

- **Gestion du temps et des priorités** : Grâce au découpage en phases, l'agent sait quelle est la **priorité temporelle** à chaque étape. Il adaptera la planification détaillée en conséquence :
 - En Phase 1, il priorisera les tâches qui contribuent directement aux **premiers revenus** (par ex. finir Asset #1 et exécuter la mission freelance en temps et en heure). Les tâches non critiques pour le court terme (comme le MVP) seront soit programmées plus tard, soit allouées en arrière-plan si le planning le permet.
 - En Phase 2, l'agent pourra incrémenter la charge en parallèle, en lançant par exemple le développement du MVP **en parallèle** de la production d'un Asset #2, tout en laissant des créneaux réservés pour d'éventuelles missions freelance. Il appliquera des techniques de **planification capacitaire** pour s'assurer que la somme des tâches prévues n'excède pas la disponibilité du développeur. Si un conflit apparaît (par ex. deux deadlines rapprochées), l'agent le signalera et proposera des arbitrages (décaler une tâche moins prioritaire, ou sous-traiter si possible).
 - L'agent *Planner* tiendra compte également des **dépendances** entre objectifs : par exemple, il saura qu'une part du revenu de 2027 doit provenir du MVP une fois lancé, donc il fera en sorte que le MVP soit finalisé suffisamment tôt pour générer éventuellement des ventes avant fin 2027. De même, il intégrera les tâches de marketing dans la timeline, conscient que la visibilité conditionne en partie le succès des ventes (comme souligné précédemment).
- **Suivi des KPIs et alertes** : Avec les KPI définis (revenus mensuels, nombre d'assets publiés, taux de doc, etc.), l'agent *Planner* agira comme un **contrôleur de gestion** intelligent. Branché sur les données (ventes effectives depuis la marketplace, comptabilité des missions freelance, état du dépôt de code, etc.), il pourra :
 - Générer des **rapports périodiques** (hebdomadaires, mensuels) affichant l'avancement de chaque objectif par rapport à sa cible. Par exemple, un tableau de bord où l'on voit : « *Revenu actuel mois M : 300 € (objectif 500 €) -> 60 % atteint* », « *Asset #2 : en cours (70 % complété, à finir d'ici 40 jours)* », « *MVP : 30 % réalisé, on est dans les temps par rapport à l'échéance T3 2027* », etc.
 - **Alerter proactivement** en cas de dérive. Grâce aux seuils intermédiaires, l'agent détectera si un indicateur devient rouge. Par exemple, si à l'été 2026 aucun deuxième asset n'est encore entamé, il pourra alerter que l'objectif "3 assets d'ici fin 2026" est compromis et recommander de réévaluer la charge. Ou si les revenus stagnent plusieurs mois de suite en dessous du palier prévu, il pourra suggérer d'intensifier le marketing ou de baisser le prix de certains assets pour doper les ventes.
 - Sur les aspects qualité, l'agent surveillera les **lignes rouges** définies dans le système. Par exemple, il ne permettra pas qu'une tâche de développement commence sans qu'une fiche de conception L2 soit liée (il pourrait même automatiser la création de ce squelette de doc à chaque nouveau projet démarré). S'il repère une « *implémentation rapide non documentée* », il la signalera comme un **écart à corriger** (aligné avec la politique anti-"Quick & Dirty" ⁶). De même, via des hooks dans l'IDE (Antigravity), l'agent peut exécuter des checks sur le code commité pour vérifier qu'il respecte les contraintes L0.5 (performance, absence de régression) et marquer les violations à examiner.
- **Coordination avec l'utilisateur (et validation humaine)** : Bien que l'agent *Planner* automatise la planification et le suivi, il respecte en tout point la philosophie du système : **le contrôle final reste humain**. Ainsi, l'agent fournira des **recommandations** et des plans, mais laissera toujours la décision à l'utilisateur avant d'engager des changements majeurs. Par exemple, si l'agent estime qu'un objectif n'est plus atteignable et propose de le modifier (pivoter vers un autre

marché, repousser une échéance), il soumettra cette suggestion au développeur pour validation explicite. Cela s'inscrit dans la règle de “*Pas d’Hallucination*” – l’agent n’agit pas de sa propre initiative sur les stratégies L1/L2 sans accord humain ¹⁰. Le développeur joue donc un rôle de **superviseur** qui arbitre les décisions finales, en s’appuyant sur l'**analyse synthétique fournie par l’IA** ¹⁴.

- Concrètement, on peut imaginer une réunion hebdomadaire (par exemple chaque lundi matin) où l’agent *Planner* présente un **état d’avancement** et un **plan d’action** pour la semaine à venir, que le développeur ajuste le cas échéant. L’agent prend en compte ces retours puis exécute la mise à jour du planning et la création des tickets de tâches dans l’outil (Obsidian, Jira intégré, etc.).
- De plus, l’agent peut servir d'**assistant de décision**. Par exemple, si deux opportunités se présentent (développer Asset X vs accepter Mission Y), l’agent peut aider à peser le pour et le contre en regard des objectifs stratégiques : il analysera l’impact de chaque choix sur la timeline et les KPIs (gain financier immédiat vs temps pris sur le MVP, etc.), et fournira une recommandation objective. Le choix final reviendra au développeur, éclairé par ces données.
- **Maintien de la cohérence globale** : Enfin, le *Planner* veille à conserver la **cohérence systémique** à mesure que le plan opérationnel évolue. Grâce à sa vision transversale («*voir le tout*» ¹⁴), il s’assure que toute nouvelle initiative s’aligne avec le L0 et les invariants :
 - Par exemple, si en cours de route le développeur envisage d’ajouter un objectif (mettons, participer à une game jam en 2027), l’agent évaluera en quoi cela sert ou non les objectifs L1 existants. Si cela risque de détourner du chemin critique sans bénéfice clair, il le fera remarquer pour éviter une dispersion non justifiée.
 - Inversement, si une opportunité inattendue cohérente avec la mission survient (ex : Epic Games lance un **programme de subventions** pour les créateurs d’assets innovants), l’agent *Planner* pourrait suggérer d’y consacrer du temps car cela pourrait accélérer la réalisation de l’objectif de revenus ou de professionnalisation. Il intégrerait alors cette nouvelle **branche de plan** tout en recalibrant le reste (par exemple, en repoussant légèrement un autre livrable moins prioritaire).

En résumé, l’agent persona “*Planner*” utilisera la structure d’objectifs validée comme un **blueprint** pour organiser l’action. En bon **amplificateur cognitif**, il gèrera la charge de planification détaillée, la surveillance et l’optimisation continue, permettant au développeur de se concentrer sur l’exécution créative et technique en ayant l’assurance qu’aucun aspect stratégique ne tombe dans l’oubli. Ce duo humain-IA, où chacun joue son rôle, maximisera les chances d’atteindre – voire dépasser – les ambitions fixées, tout en respectant les valeurs de cohérence, de traçabilité et de rigueur posées dès le départ. Le résultat espéré en fin de parcours est un développeur **devenu professionnel accompli**, soutenu par un système agentique efficace qui aura prouvé sa valeur tout au long de la route.

Sources : Quelques références pratiques ont été utilisées pour asseoir le réalisme des évaluations : expérience d’un créateur ayant généré ~15 000 \$ en un an sur le Marketplace Unreal en multipliant les assets et les efforts marketing ² ⁴, retour de vendeurs sur la saturation du marché d’assets depuis la plateforme Fab d’Epic ⁵ et importance de se démarquer via la communauté ¹¹ ¹². Ces éléments ont nourri l’audit critique afin d’ancrer les conseils dans la réalité actuelle du secteur. Le plan s’inscrit en outre dans le respect des principes définis dans le vault (invariants du système, rôle de l’IA) pour garantir une cohérence entre la vision et l’exécution ¹ ¹⁴. Toutes ces données convergent vers une même conclusion : avec des objectifs bien calibrés, un usage judicieux de l’IA et une planification méthodique, le développeur dispose d’une feuille de route solide pour passer du statut d’indépendant émergent à celui de **professionnel aguerri d’ici 2027**.

1 6 8 9 10 13 14 Mission.md

file://file_0000000e70471f49e3fdc9c12f94a37

2 3 4 11 How I Earned My First \$15,000 in a Year Selling Assets on UE Marketplace | by Azrael | Medium

<https://medium.com/@mr.hamzik1026/how-i-earned-my-first-15-000-in-a-year-selling-assets-on-ue-marketplace-6421306d2bfc>

5 12 Are you able to make a meaningful income selling assets on Fab? : r/UnrealEngine5

https://www.reddit.com/r/UnrealEngine5/comments/1lrism5/are_you_able_to_make_a_meaningful_income_selling/

7 Mission.md

file://file_000000071ec722f8acee97155f622ce