

Rapport de Soutenance

Carré Magique

Gabriel Croci, Chloé Lambert, Olivier Cesari, Vincent Die, Sofia Aïd

March 8, 2025

1 Introduction

Le projet High Stakes, un jeu vidéo innovant inspiré du jeu de cartes "Le Président". Depuis la validation du cahier des charges, notre équipe s'est consacrée à développer un jeu unique combinant des mécaniques classiques avec une immersion narrative et visuelle captivante. Ce projet représente une opportunité de mettre en œuvre des compétences techniques et artistiques, tout en favorisant une collaboration polyvalente. Ce document fera le bilan des travaux réalisés, identifiera les retards et défis rencontrés, et présentera les objectifs futurs pour garantir la réussite de ce projet. Le projet High Stakes se distingue par sa combinaison du jeu en 3D pour les duels et de graphismes en 2D au style rétro pour les environnements. Cette approche hybride vise à offrir une expérience immersive et intéressante. Les ambitions de notre équipe vont au-delà de la simple réalisation technique : nous voulons créer une aventure divertissante et mémorable pour les joueurs.

Ce rapport s'inscrit dans cette vision et reflète le dévouement de l'équipe Carré Magique à atteindre cet objectif. Notre objectif est de réaliser un jeu unique et original tout en adoptant une approche classique du jeu vidéo. High Stakes, Enjeux élevés, se construira avec un lore, une histoire qui se développe au fil du jeu, en plus des parties de Président classiques. Le jeu se déroulera sous une forme narrative où le joueur sera accompagné d'un narrateur tout le long de sa partie. Il explorera un environnement en 2D avec un style médiéval revisité, et sera invité à réaliser des duels, cette fois en 3D, plus précisément des parties de Président contre des IA. Selon la difficulté du jeu et la progression, les IA seront plus au moins fortes et intelligentes, évoluant dans leurs techniques de jeu.

Le joueur apprendra au fur et à mesure à jouer, en même temps que la difficulté de l'IA évolue. En avançant, le joueur découvrira l'histoire du jeu, qui sera conclue à la fin, lors d'un duel final contre le boss, l'ennemi principal, du jeu. Nous avons opté pour ce modèle de jeu qui est classique, mais s'adapte à beaucoup de types de jeux et qui plaie généralement aux joueurs. En revanche, le fait de combiner ce modèle et le Président est complètement novateur, bien qu'il existe des jeux de cartes interactifs, il n'existe aucun jeu du Président qui adopte ce modèle. C'est ce qui rend notre jeu et nous démarque des jeux de carte classiques que l'on peut retrouver en ligne. En plus de l'approche classique du Président et de l'aspect historique, nous rajoutons un ensemble de règles et de variantes de jeu en plus pour rendre notre jeu plus jouable et plus dynamique. L'objectif est également de rendre la partie moins monotone et répétitive au fil des tours. Une partie de Président entre amis devient rapidement lassante si on n'y inclut pas de variante ou de règles en plus. Enfin, les parties de Président seront exclusivement en 3D. Assis autour d'une table, entre joueurs ou IA, une partie se déroulera dans un environnement en 3 dimensions, avec un ensemble d'animations, d'effets visuels et de sons qui rendront le jeu plus divertissant. La map du jeu, les 2 paysages, bâtiments etc, seront réalisés en 2D, avec des animations et effets qui ressemblent au style "8-bits" pixélisé. Ce document

explorera en premier lieu les origines de notre projet, sa nature, puis son objet d'étude. L'état de l'art, l'entreprise et son équipe seront développés. Enfin, la répartition, l'avancement et la planification des tâches seront expliqués à la fin de ce document.

Finalisation du développement initial

- Jeu général du président terminé
- Ajout des mécaniques de base du président (Règle du 2 qui coupe le jeu, jeu en double, en triple et hiérarchisation des gagnants/perdant : assignation de rôles)
- Mise en place des différents assets réalisés par Chloé, animation des cartes et des boutons, de l'interface graphique générale du jeu.
- Mise en place d'une interface graphique : au lancement du jeu, l'utilisateur a l'écran d'accueil avec le bouton jouer. Ensuite il aura une interface lui demandant de lancer une partie. Pour le moment, les parties sont en mode Facile, comme spécifié dans le CDC, le mode Facile favorise le joueur en dépit des IA. Une fois le jeu lancé, la distribution des cartes se réalisera. Chaque joueur aura 13 cartes données aléatoirement avec le système de RNG et d'IA réalisé par Vincent. Ensuite, la partie débute par un premier mouvement du joueur principal, puis du 2e, 3e et enfin 4e joueur.

Le joueur peut sélectionner 1, 2 ou 3 cartes s'il décide de jouer en simple, en double ou en triple. Les IA sont capables de jouer également en simple, double et triple. Lorsqu'un 2 est posé, la pile se "coupe", c'est-à-dire que toutes les cartes vont dans la pile et que le joueur qui a posé ce 2 rejoue. La partie se termine lorsque tous les joueurs se seront débarrassés de leurs cartes. Une fois la partie terminée, le joueur verra un classement avec les différents rôles attribués. Pour le moment, ils ne servent à rien.
- Les adversaires sont uniquement des IA, le mode multijoueur est actuellement en cours de réflexion par la personne qui s'en charge.
- Le mode de jeu en 2D, en mode exploration n'a pas été changé, cependant il sera inclus dans la prochaine version avec le jeu du Président. Le joueur pourra lancer des parties en lançant des combats contre des adversaires non joueurs dans le mode exploration.

Voici un exemple avec une image :

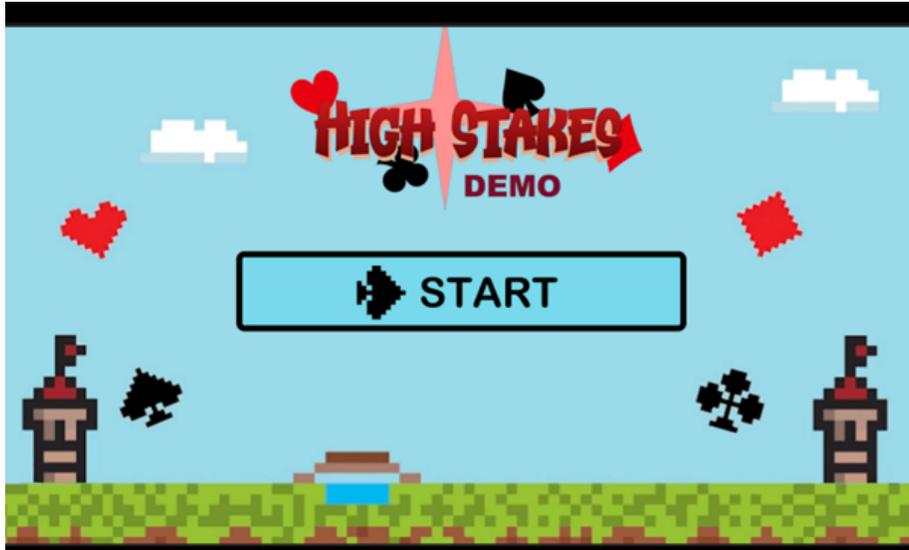


Figure 1: L'entrée



La Fonction RNG

Notre fonction RNG, que nous avons conçue, est divisée en plusieurs étapes clés qui assurent un comportement cohérent et prévisible tout en offrant une certaine aléa en fonction du niveau de difficulté sélectionné par le joueur. L'une des premières décisions que prend le joueur est de choisir le niveau de difficulté, ce qui influencera directement le comportement du générateur de nombres aléatoires. Le niveau de difficulté choisi par le joueur va influencer sur la probabilité d'apparition des différents indices de cartes, permettant ainsi un ajustement dynamique en fonction des préférences du joueur. En effet, plus le niveau est élevé, plus les probabilités de tirer une carte puissante augmentent. Cette variation des probabilités suit une logique basée sur la loi des grands nombres, ce qui garantit que sur un grand nombre d'essais, les résultats se rapprochent d'une distribution équilibrée mais influencée par le niveau choisi.

La fonction RNG repose sur l'heure actuelle du joueur, en la prenant comme base pour générer des nombres aléatoires. En d'autres termes, l'heure actuelle du système est utilisée comme semence pour calculer un nombre de départ. Cette valeur est ensuite multipliée par un facteur qui dépend du niveau de difficulté. Par exemple, pour un niveau facile, ce facteur peut être relativement faible, tandis que pour un niveau difficile, le facteur sera plus élevé, augmentant ainsi la portée des résultats possibles. Une fois ce calcul effectué, nous utilisons la fonction `Random` de C# pour réduire cette valeur et obtenir un nombre compris dans une plage spécifique. Ce nombre retourné par la fonction représente l'indice d'une carte, où 1 correspond à la carte la plus faible et 13 à la carte la plus forte. Ce processus assure que chaque tirage de carte est influencé à la fois par l'heure actuelle (ajoutant un aspect dynamique et temporel) et par le niveau de difficulté, garantissant ainsi une certaine variabilité tout en respectant les règles définies par le joueur.

Au final, la carte qui sera tirée dépend à la fois de l'heure (qui est une valeur unique à chaque exécution du programme) et du niveau de difficulté, ce qui permet au jeu de s'adapter à la fois à la situation du joueur et aux exigences spécifiques qu'il a choisies. En résumé, notre RNG fonctionne en combinant des éléments temporels (l'heure actuelle) et des facteurs définis par l'utilisateur (le niveau de difficulté), ce qui permet d'aboutir à un générateur de nombres aléatoires à la fois unique et dynamique, et qui varie de manière prévisible en fonction du contexte de jeu. L'IA : Pour l'IA nous avons utilisé des fonctionnalités de Unity.

```
public static int suffle(int x)
{
    Random random = new Random();
    string seed = DateTime.Now.ToString();
    int r = seed[^1] + 1;
    r += seed[^2] + 1;
    r *= x;
    while (r > 15)
    {
        r -= random.Next(5,x);
    }
    r -= random.Next(2,x);
    return Math.Abs(r);
}
```

```
using unity.MLAgentusing UnityEngine;
using unity.ml;
```

Introduction

Dans ce projet de modélisation 3D, j'ai conçu une table, des chaises et une chope de bière en utilisant Blender. L'objectif était de créer un mobilier agréable aux yeux des joueurs avec une touche médiévale, tout en optimisant la modélisation pour une utilisation dans un jeu vidéo. J'ai utilisé différentes techniques de modélisation, de texturage et d'optimisation afin d'obtenir un rendu fidèle et performant.

Modélisation de la table

La table est composée d'un plateau rectangulaire et de quatre pieds en bois. J'ai commencé par créer un cube de base (**Shift + A > Mesh > Cube**), que j'ai redimensionné en utilisant l'outil de mise à l'échelle (**S**) pour obtenir une forme allongée et fine, correspondant à un plateau de table. Ensuite, j'ai ajouté quatre autres cubes pour former les pieds, que j'ai placés aux coins du plateau en les ajustant avec l'outil de déplacement (**G**).

Afin de donner un aspect plus réaliste, j'ai appliqué un léger bevel (**Ctrl + B**) aux arêtes du plateau pour adoucir les bords. J'ai également sculpté légèrement la surface avec l'outil de sculpture pour donner un effet de bois usé. Après avoir terminé la modélisation, j'ai déplié les UV (**U > Smart UV Project**) pour préparer la table à la texturation.

Modélisation des chaises

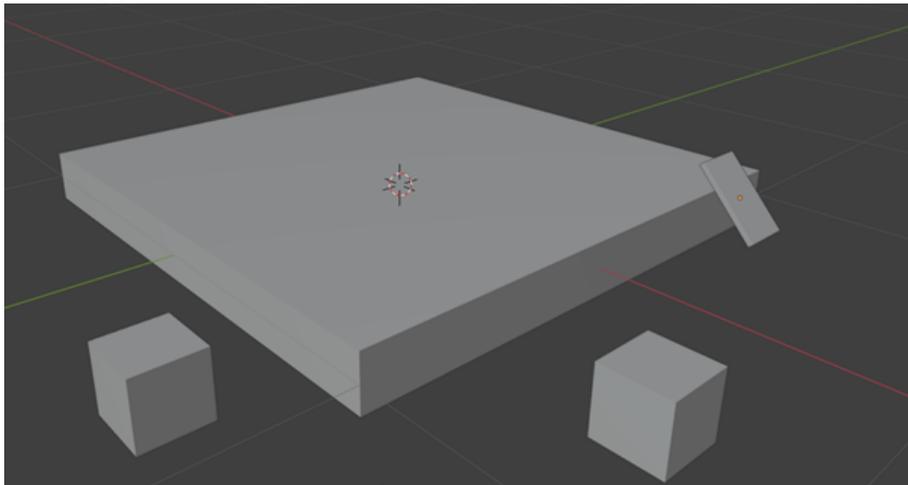
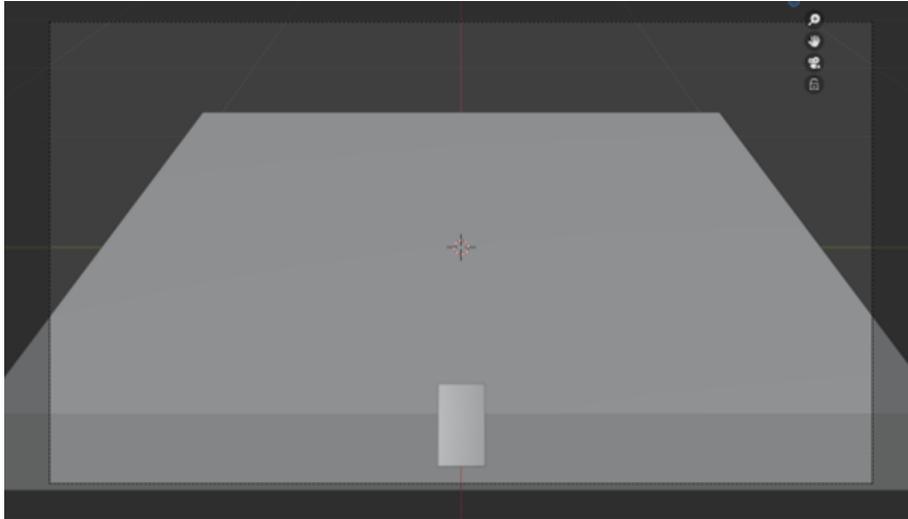
Les chaises ont été conçues sur le même principe que la table, avec un cube redimensionné pour l'assise et quatre pieds placés de manière équidistante. Pour le dossier, j'ai ajouté un rectangle vertical légèrement courbé à l'aide de l'outil de modification (**Modifier > Subdivision Surface**), ce qui permet d'obtenir une structure plus organique.

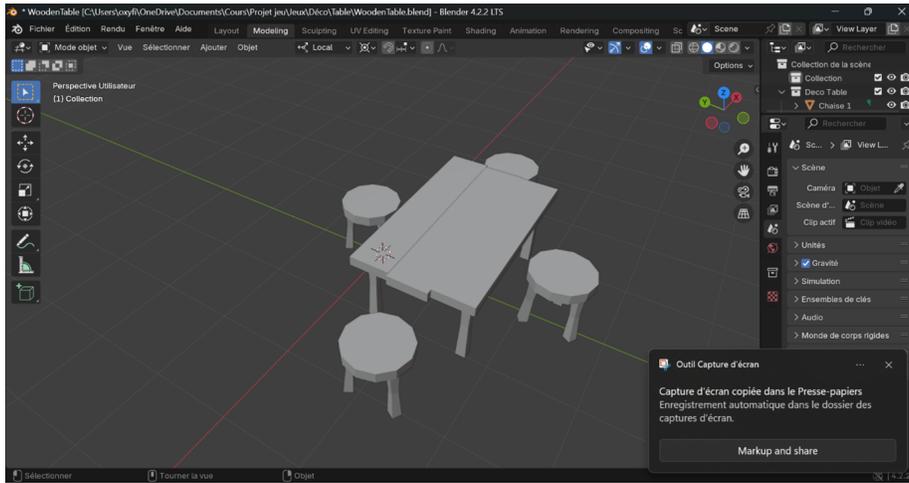
Ensuite, j'ai ajouté des traverses horizontales reliant les pieds pour plus de stabilité et un effet plus authentique. [13:20] En ajustant les proportions et en affinant les détails, j'ai obtenu une chaise à l'apparence robuste et crédible. J'ai appliqué une texture de bois similaire à celle de la table pour assurer une cohérence visuelle.

Modélisation de la chope de bière

Pour la chope, j'ai utilisé un cylindre de base (**Shift + A > Mesh > Cylinder**) que j'ai légèrement étiré sur l'axe Z pour obtenir une hauteur correcte. J'ai ensuite extrudé le dessus vers l'intérieur (**E > Scale**) pour créer l'ouverture du récipient. Pour les détails, j'ai ajouté des anneaux horizontaux en dupliquant des cercles extrudés afin d'imiter les renforts en métal des chopes médiévales. La poignée a été créée à partir d'un torus (**Shift + A > Mesh > Torus**), que

j'ai redimensionné et ajusté pour qu'il s'attache naturellement sur le côté de la chope. Enfin, j'ai appliqué une texture métallique pour les parties en acier et une texture boisée pour le corps de la chope. J'ai également ajouté une légère réflexion pour donner un effet plus réaliste. Il faut savoir aussi que nous nous sommes aidés de patrons qui nous ont inspirés pour faire nos dessins.

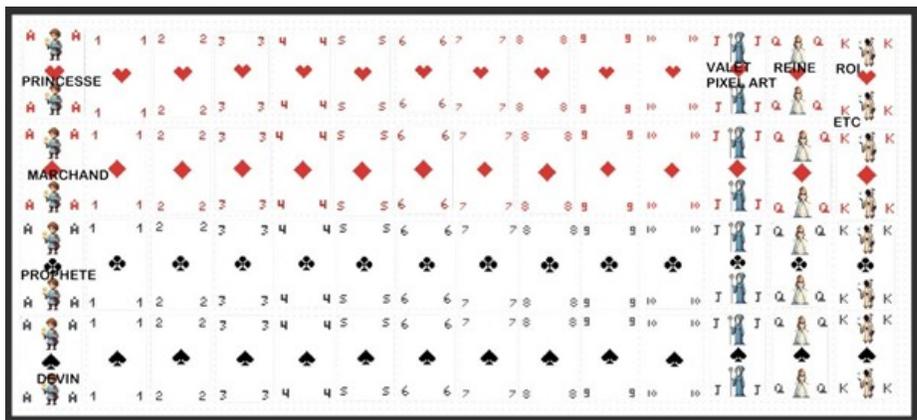
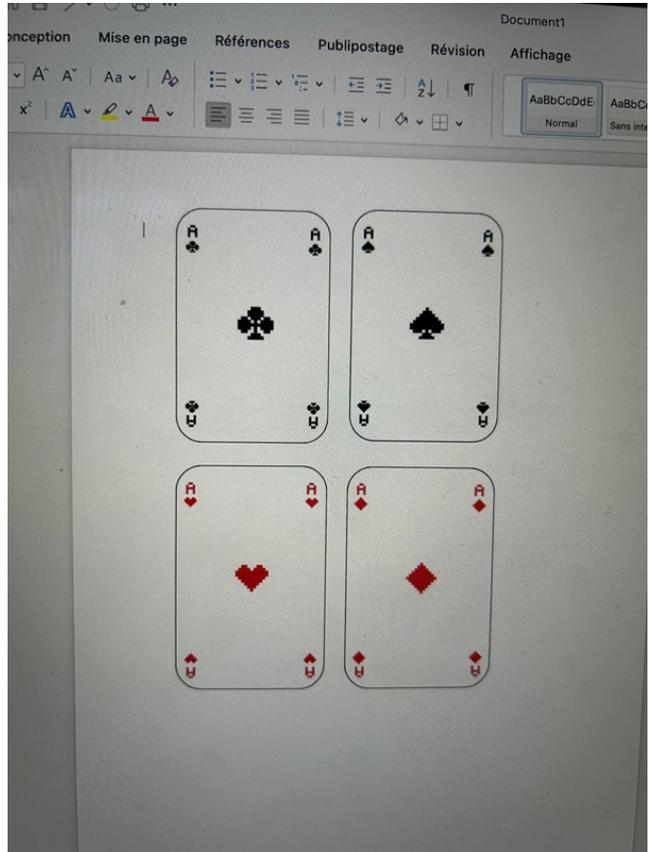


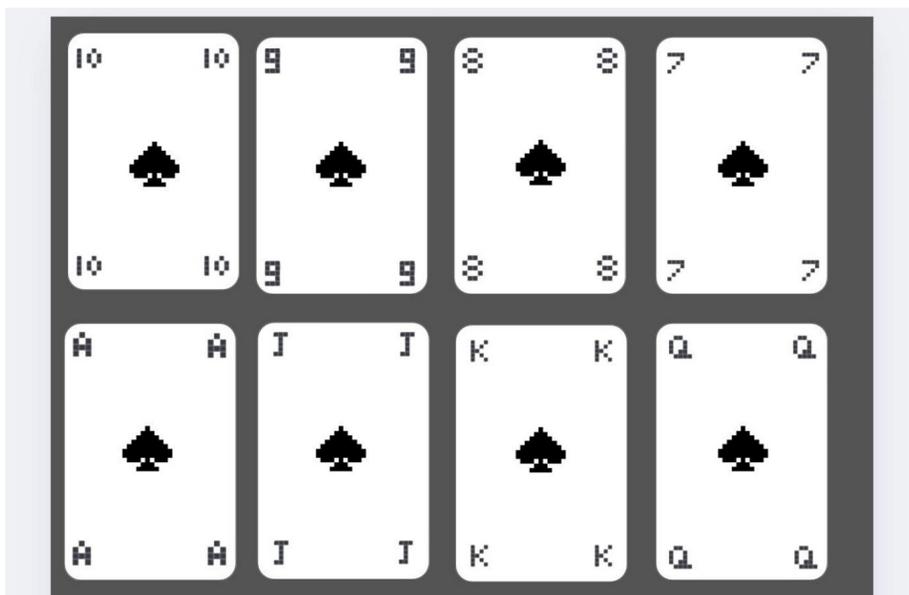
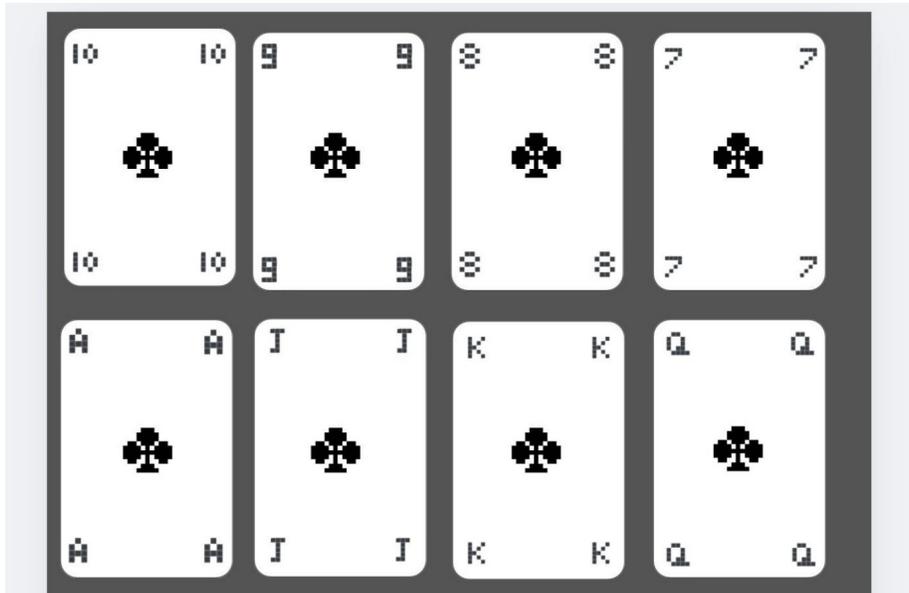


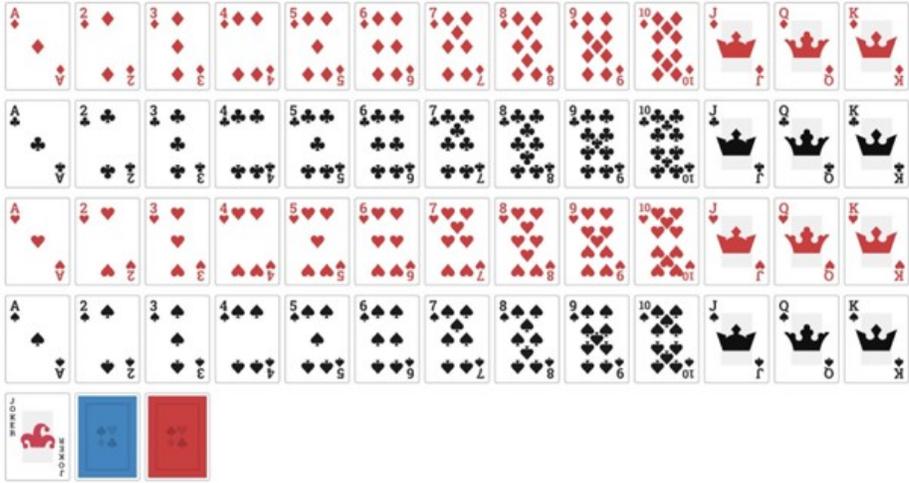


Les Cartes du jeu en Pixel

Maintenant expliquons comment faire la création de carte en pixel. Pour les cartes en pixel art, on choisit une taille comme 64x88 pixels ou 128x192 pixels. Puis on utilise une palette de couleurs limitée pour garder l'aspect simple et lisible. Garde les bords nets et définis avec des couleurs contrastées. On dessine un rectangle pour représenter la carte. Ajoute une couleur de fond simple (par exemple, un vert pour imiter un tapis de casino) et définis les coins où les symboles et numéros apparaîtront. Ensuite la création de l'As : Numéro "A" : Place un "A" dans le coin supérieur gauche (et parfois dans le coin inférieur droit). Utilise une police simple et lisible, contrastée par rapport au fond. Symboles : Utilise un symbole de cœur, carreau, trèfle ou pique dans chaque coin de la carte. Le symbole doit être clair et simple, en pixel art. Puis les détails : Ajoute des ombres subtiles ou des motifs dans le fond pour un look plus raffiné, mais évite les détails excessifs pour que le texte et les symboles restent lisibles. Création des autres cartes. Et les numéros, pour les cartes de 2 à 10, place simplement le numéro dans les coins. Les cartes de figures (Roi, Dame, Valet) afficheront "R", "D", et "V". Symboles : Chaque carte affichera un symbole correspondant (par exemple, cœur pour un As de cœur). Les cartes de figures peuvent avoir un fond plus orné pour les différencier. On a utilisé pixie qui est parfait pour le pixel art. Photoshop et GIMP peuvent aussi être utilisés en définissant la grille de pixels pour un rendu plus précis. Voici quelques design en exemple que nous avons.







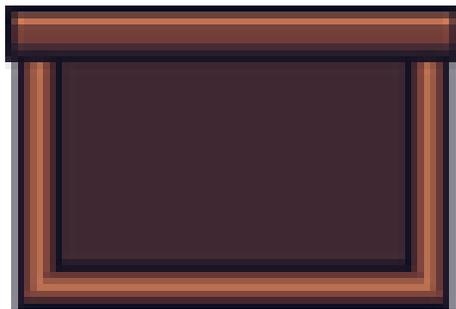
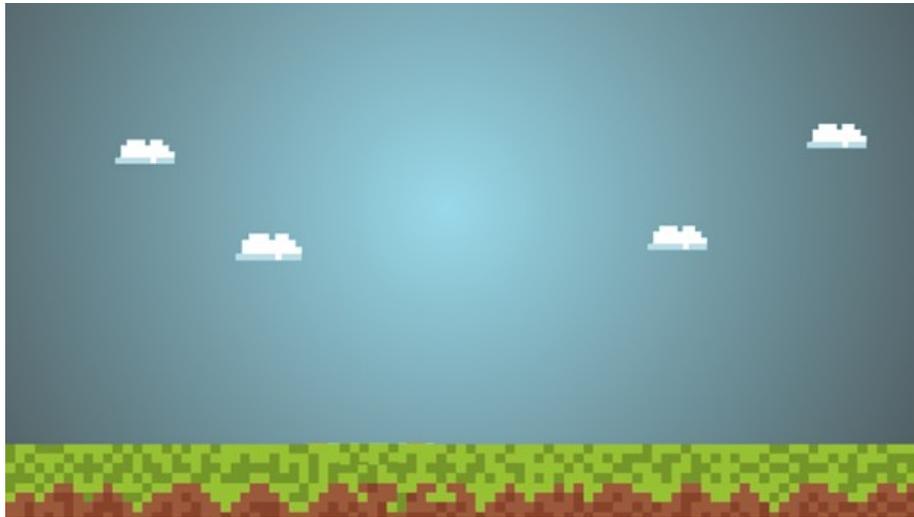
2 Les Assets

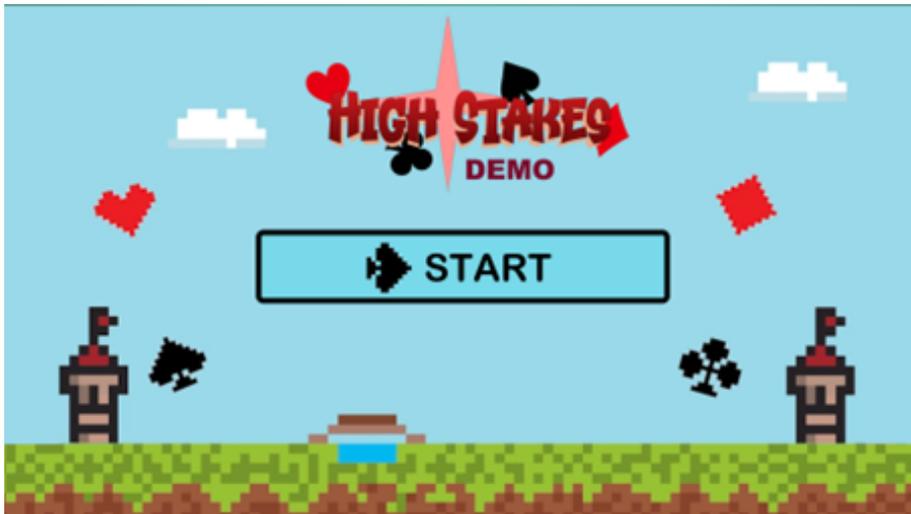
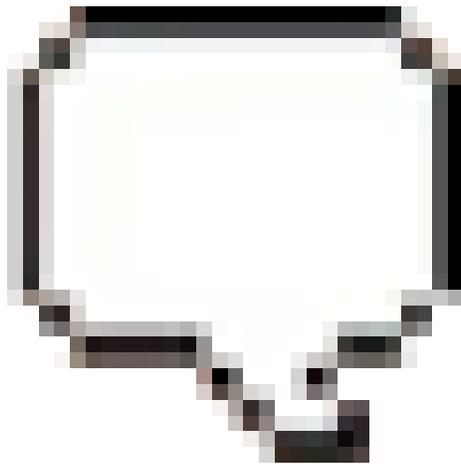
La création des assets graphiques pour un jeu vidéo implique la conception d'éléments visuels essentiels à l'interface, comme les boutons, menus, barres de progression, et autres éléments interactifs. Pour cela, des logiciels de création graphique tels que Photoshop ou Illustrator sont utilisés afin de concevoir des éléments visuels attractifs et adaptés au style du jeu. Chaque asset doit être créé en tenant compte de son aspect fonctionnel et esthétique. Par exemple, un bouton aura plusieurs états (normal, survolé, cliqué) afin d'offrir une expérience interactive fluide.

Une fois les éléments créés, ils sont exportés dans des formats adaptés, comme PNG pour la transparence ou JPEG pour des images de fond. Les animations sont également cruciales pour rendre l'interface dynamique, ce qui implique de créer des images supplémentaires pour chaque transition ou interaction.

Ces assets sont ensuite intégrés dans le moteur de jeu, où les développeurs configurent des scripts pour gérer les interactions entre l'utilisateur et l'interface. Il est également important de veiller à ce que les assets soient optimisés pour ne pas nuire aux performances du jeu, en réduisant leur taille tout en conservant une bonne qualité visuelle.







Conclusion

Ce projet, intitulé "High Stakes", a permis de combiner des mécaniques de jeu classiques avec des éléments modernes et immersifs, tels qu'une interface 3D pour les duels et un environnement 2D rétro pour l'exploration. Grâce à une approche hybride, le jeu se distingue par son concept novateur de l'intégration du jeu de cartes "Le Président" dans un univers narratif et interactif. L'équipe a réussi à mettre en œuvre des mécanismes de jeu solides et fonctionnels, tout en offrant une expérience visuelle et sonore captivante. Les défis rencontrés ont permis d'améliorer la conception du jeu, en optimisant la jouabilité et l'intelligence artificielle des adversaires. Le développement du générateur de nombres aléatoires (RNG) et de l'IA a renforcé la dynamique du jeu, tout en garantissant une expérience fluide et agréable pour les joueurs. Les prochaines étapes se concentreront sur l'amélioration de l'interface multijoueur et l'intégration de nouvelles mécaniques de jeu pour enrichir l'expérience. En conclusion, "High Stakes" représente un projet abouti qui illustre l'innovation et la collaboration au sein de l'équipe Carré Magique, avec un potentiel pour devenir une référence dans son genre.