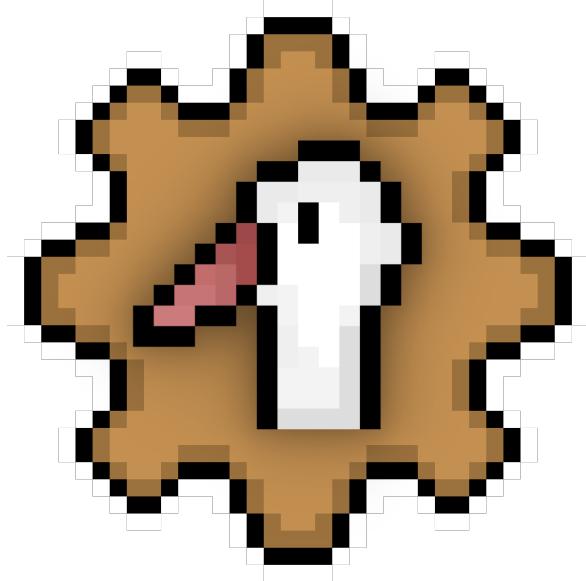


Rapport de Soutenance n°3



STORCKI CORP.

«Plenty Orders»

BEFORT HUGO
GRILLO THOMAS
BEVILACQUA LUC
CHABOISSEAU RAPHAËL

MAI 2025

Sommaire

1	Introduction	4
1.1	Notre entreprise	4
1.1.1	Notre équipe	5
1.2	Présentation du projet	6
1.2.1	Origine et nature du projet	6
1.2.2	Lore du jeu	6
1.3	Logiciels utilisés	7
1.4	Installation du jeu	8
2	Fonctionnalités	9
2.1	Outils / Machines de la Cuisine	9
2.2	Sauvegardes	11
2.3	Scores	12
2.4	Skins / Capacités joueurs	13
2.5	Collisions	14
2.6	Combat joueurs	16
2.7	Système d'inventaire	17
2.8	Effets sonores	19
2.9	Système d'interaction et de tâches	20
2.9.1	Menu principal	20
2.10	Multijoueurs	22
2.11	Évenements	23
2.12	Codes de triche	25
2.13	Dialogues	27
3	Intelligence Artificielle	28
3.1	Clients	28
3.2	Inspecteur	29
3.3	Rat	30
4	Site Web	31
4.1	Accessibilité et Paramètres Utilisateur	31
4.2	Page d'accueil	32
4.3	Section “À propos”	33
4.4	Section Téléchargement	33
4.5	Page de contact	34
5	Niveaux	34
6	Tutoriel	35
6.0.1	Textures	36
6.0.2	Plats	36
6.1	Niveau 1	37
6.1.1	Présentation Niveau 1	37
6.1.2	Plats	37
6.1.3	Textures	39
6.2	Niveau 2	40
6.2.1	Présentation Niveau 2	40
6.2.2	Plats	41

6.2.3	Textures	42
6.3	Niveau 3	43
6.3.1	Présentation Niveau 3	43
6.3.2	Plats	44
6.3.3	Textures	45
6.4	Niveau 4	46
6.4.1	Présentation Niveau 4	46
6.4.2	Plats	47
6.4.3	Textures	48
6.5	Niveau 5	49
6.5.1	Présentation Niveau 5	49
6.5.2	Plats	50
6.5.3	Textures	51
7	La réalisation du projet	51
7.1	Ses joies	51
7.1.1	Ses joies collectives	51
7.1.2	Ses joies personnelles	53
7.2	Ses peines	54
7.2.1	Ses peines collectives	54
7.2.2	Ses peines personnelles	55
8	Conclusion	57

1 Introduction

1.1 Notre entreprise

Storcki Corp est un jeune studio de développement de jeux vidéo, né de la rencontre de ses cinq membres à l'école d'ingénierie informatique EPITA à Strasbourg. Bien que les membres de Storcki Corp proviennent d'horizons divers, c'est désormais la ville de Strasbourg, chef-lieu du Grand Est, qui les relie.

C'est ainsi que le symbole de l'Alsace, la cigogne, trouve naturellement sa place dans le logo du studio. Dans le folklore, les cigognes sont associées à la livraison et à la mobilité. Le logo, et plus particulièrement la cigogne, symbolise la capacité de Storcki Corp à délivrer des jeux vidéo de qualité dans le monde entier. Storcki Corp cultive un équilibre entre originalité, précision technique et une dose d'humour. Les valeurs communes des membres de Storcki Corp, telles que l'entraide, l'inclusivité et un esprit familial, façonnent l'entreprise de l'intérieur et s'expriment à travers leurs réalisations.



Évolution du logo de l'entreprise

1.1.1 Notre équipe



Hugo BEFORT
Chef de projet

Hugo, 19 ans, a fait toute sa scolarité au lycée Sainte-Anne à Strasbourg avant de rejoindre EPITA. Passionné de jeux vidéo et de sports, en particulier des sports nautiques, il a pratiqué divers sports collectifs, ce qui lui a permis de développer une excellente capacité d'adaptation et de travail en équipe.

En tant que chef de projet, il a déjà collaboré avec son ancienne équipe, composée de connaissances du lycée, sur un jeu de survie. Obstinent, il va au bout des choses ; tant qu'il n'a pas réussi à atteindre ses objectifs, il persévère sans abandonner. De plus, il a développé plusieurs sites web.



Thomas GRILLO
Développeur

Thomas, 18 ans, a suivi une éducation à Marseille avant de passer deux ans à Toronto, au Canada, puis d'intégrer EPITA.

Il développe une passion pour la création de jeux vidéo dès l'âge de 12 ans avec la plateforme ROBLOX, avant de fonder son studio Tkium et de sortir jusqu'à présent deux jeux sur Steam avec Unity, ainsi que divers logiciels et bots Discord.



Raphaël CHABOISSEAU
Développeur

Raphaël Chaboisseau, 18 ans, est passionné d'informatique et de jeux vidéo et est un grand amateur de musique. Il a développé son esprit d'équipe en dirigeant plusieurs projets de jeux au lycée.

Habitué à créer des sites web, il automatise également des requêtes internet en utilisant des API (Application Programming Interface).



Luc BEVILACQUA
Développeur

Luc, 18 ans, a effectué sa scolarité en Moselle, plus précisément au lycée Fabert situé au cœur de Metz.

Actuellement étudiant à EPITA Strasbourg, il est passionné de jeux vidéo, de sports et de culture occidentale. Sa curiosité naturelle l'a poussé à acquérir des bases sur de nombreux thèmes, y compris le montage photo et vidéo, qui sont des compétences importantes dans la conceptualisation et la promotion d'un jeu vidéo.

1.2 Présentation du projet

Plenty Orders est un jeu multijoueur en deux dimensions (2D) centré sur la cuisine, développé par Storcki Corp. Il adopte un style arcade et s'inscrit dans la catégorie du « kitchen madness ».

Inspiré notamment par *Witchtastic*, *One-Armed Cook* et *Overcooked*, notre objectif est de recréer, à travers Plenty Orders, une ambiance déjantée et amusante. Ce jeu se veut idéal pour des parties entre amis, offrant une expérience conviviale et sans prise de tête, parfaite pour une soirée détente.

1.2.1 Origine et nature du projet

Notre équipe était initialement partagée entre deux concepts de jeux. Le premier est un jeu de mines en coopération dans un style similaire à Minecraft en 2D. Le deuxième jeu se concentre sur la gestion d'un camp dans un labyrinthe, l'objectif serait de se développer et de créer son camp afin d'être le dernier survivant.

Cependant, aucun de ces jeux ne nous semblait amusant. Ce qui nous intéresse réellement est un jeu où l'on peut s'amuser avec nos amis, un jeu avec une fin qui demeure captivante et qui nous donne envie de le terminer au moins une fois.

Nous avons donc eu l'idée de concevoir un jeu de coopération où l'on doit gagner avec ses amis, tout en étant libres de les ralentir. L'idée de créer un jeu de cuisine a suscité l'enthousiasme de tous, un type de jeu peu exploité, offrant ainsi un large potentiel créatif.

1.2.2 Lore du jeu

Les joueurs incarnent des chefs cuisiniers d'un grand restaurant dont la mission est de reproduire un plat mystérieux hérité de leur grand-mère. Pour y parvenir, ils doivent parcourir le monde à la recherche de cinq ingrédients essentiels, chacun étant caché dans un pays différent.

Afin de réunir ces précieuses ressources, les cuisiniers ferment temporairement leur restaurant et se lancent dans une aventure culinaire à travers cinq destinations, classées par difficulté croissante :

- **Tutoriel** – Initiation aux mécaniques du jeu et mise en avant de l'histoire en France.
- **Égypte** – Premier niveau avec possibilité de perdre.
- **Japon** – Techniques culinaires complexes et rythme plus soutenu.
- **Italie** – Introduction à la gestion avancée des commandes.
- **Mexique** – Environnement dynamique et contraintes imprévisibles.
- **France** – Le défi ultime, où chaque détail compte.

Au fil de leur voyage, les joueurs découvriront des cultures variées ainsi que de nouvelles façons d'aborder la cuisine. Chaque destination les plongera dans un restaurant local où ils devront gérer la préparation des plats, le service aux clients et la satisfaction générale. Toutefois, toute négligence pourrait bien entraîner des conséquences inattendues.

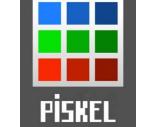
1.3 Logiciels utilisés

Le moteur principal que nous utilisons est **Unity**, un outil puissant et polyvalent largement adopté dans l'industrie vidéoludique pour la création de jeux en 2D et en 3D. Il est couplé au langage de programmation **C#**, reconnu pour sa lisibilité, sa puissance et sa compatibilité native avec Unity. Pour la partie multijoueur, nous intégrons le package officiel **Netcode for GameObjects**, permettant de construire une architecture client-serveur robuste, adaptée aux expériences coopératives et compétitives.

Concernant l'environnement de développement, nous faisons appel à des éditeurs de code performants tels que **Visual Studio Code** pour sa légèreté et sa personnalisation, et **Rider** pour ses fonctionnalités avancées dédiées à Unity.

Pour la partie artistique, notre direction visuelle se base sur le **pixel art**, un style graphique rétro très apprécié pour sa lisibilité et son charme visuel. Nous exploitons pour cela les outils **Piskel** (éditeur en ligne intuitif), **Aseprite** (référence professionnelle pour les animations pixel) et **Photoshop** (incontournable pour les retouches et compositions avancées).

Enfin, pour le design sonore et musical, nous privilégions les ressources *libres de droits*, mais nous utilisons également des logiciels de création comme **Reaper** pour l'édition, **GXSCC** pour générer des sons chiptune au style rétro, et **FL Studio**, l'un des outils les plus puissants pour composer des musiques originales adaptées à notre univers de jeu.

DÉVELOPPEMENT	GRAPHIQUE	SONS
 Unity  Rider  Visual Studio Code	  	 REAPER <small>Digital Audio Workstation</small>  

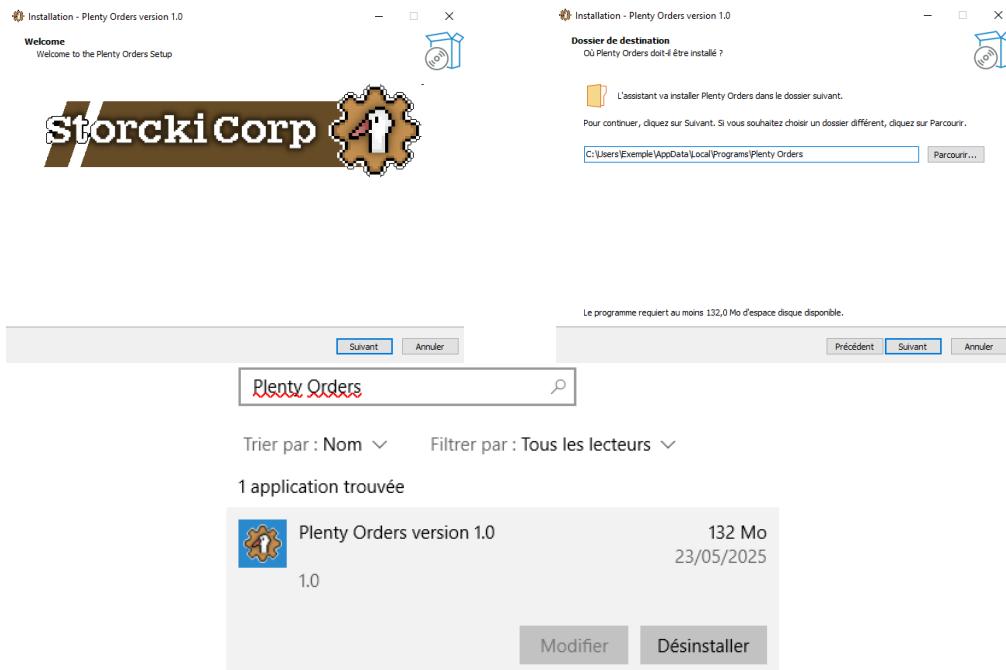
1.4 Installation du jeu

L'installation de Plenty Orders sera facilitée par un installateur Windows classique, conçu pour offrir une expérience utilisateur fluide et intuitive. Ce processus s'appuiera sur des outils standards tels que Inno Setup ou ClickOnce, permettant de générer un fichier exécutable setup.exe qui guidera l'utilisateur à travers les étapes d'installation. Ces outils sont largement utilisés pour créer des installateurs Windows efficaces et personnalisables.

Lors de l'exécution de l'installateur, l'utilisateur pourra choisir le répertoire d'installation, accepter les termes du contrat de licence, et suivre les instructions à l'écran. L'installateur s'assurera que toutes les dépendances nécessaires, telles que le .NET Framework, soient présentes sur le système, et les installera si besoin. Une fois l'installation terminée, un raccourci vers le jeu sera ajouté au menu Démarrer et/ou sur le bureau, selon les préférences de l'utilisateur.

Pour la désinstallation, Plenty Orders sera enregistré dans le panneau de configuration Windows, sous "Programmes et fonctionnalités". L'utilisateur pourra ainsi désinstaller le jeu en toute simplicité, en sélectionnant l'entrée correspondante et en cliquant sur "Désinstaller". Le processus de désinstallation supprimera tous les fichiers installés, les raccourcis créés, ainsi que les entrées de registre associées, assurant une suppression propre du logiciel.

Cette approche standardisée garantit une compatibilité maximale avec les différentes versions de Windows, tout en offrant une expérience utilisateur cohérente et fiable.



2 Fonctionnalités

2.1 Outils / Machines de la Cuisine

Les machines et outils jouent un rôle central dans notre jeu de cuisine, permettant la création des plats tout en renforçant l'immersion du joueur. Le design visuel, l'ergonomie et la fonctionnalité de chaque élément seront soigneusement travaillés pour offrir une expérience fluide et agréable.

Nous opterons pour un design équilibré entre réalisme et stylisation, avec des machines reconnaissables et adaptées à l'identité visuelle du jeu. Les équipements (fours, plaques de cuisson, mixeurs, friteuses, etc.) seront visuellement clairs tout en restant simples pour assurer une bonne lisibilité.

Enfin, les modèles 2D seront optimisés pour garantir des performances fluides, même lors des séquences les plus intenses, en trouvant le bon équilibre entre esthétique et performance technique.

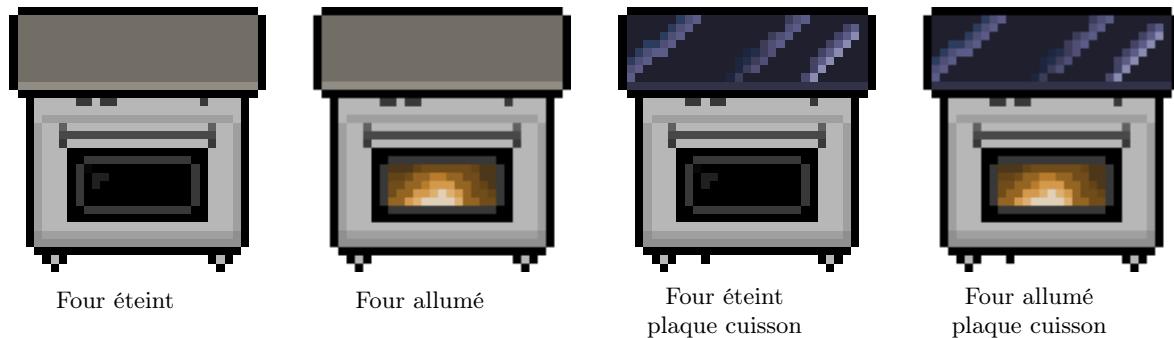
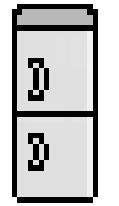


FIGURE 2 – Les différents états du four

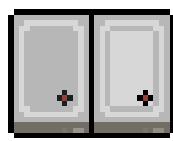
Le choix des fours dans ce projet se base sur leur capacité à simuler différentes étapes de cuisson. Un four standard sera utilisé pour représenter les cuissons classiques, tandis qu'un four avec plaque de cuisson permettra de simuler des préparations nécessitant un espace supplémentaire. Les versions "allumé" et "éteint" permettent de représenter l'état de chaque four pendant le jeu, offrant ainsi une meilleure interaction avec les joueurs.

Dans ce projet, ces quatre éléments sont essentiels pour simuler les différentes étapes de préparation et de gestion des ressources en cuisine. Le frigo est utilisé pour stocker les produits nécessitant une conservation au frais, le placard pour les ingrédients secs ou en conserve, la poubelle pour simuler l'élimination des déchets générés pendant la préparation des plats, et le tiroir pour ranger les petits ustensiles ou accessoires nécessaires à la préparation.

Ensemble, ces équipements ajoutent un aspect de gestion et d'organisation réaliste, permettant aux joueurs d'interagir avec des objets courants de la cuisine tout en renforçant l'immersion et la dimension stratégique du jeu.



Réfrigérateur



Placard



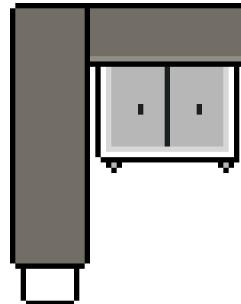
Poubelle



Tiroir



Vue de côté

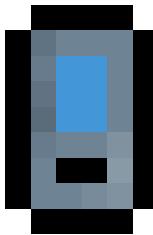


Exemple

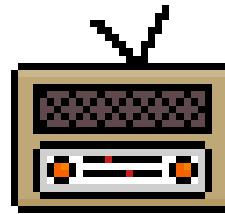


Exemple

Les trois images montrent des vues en side, offrant plus de profondeur qu'une simple vue frontale. La première montre un côté d'un élément, tandis que la deuxième et la troisième ajoutent un placard et un four, illustrant des exemples d'utilisation dans un environnement de cuisine. Ce choix de perspective permet de mieux visualiser l'agencement et l'interaction des objets, rendant la présentation plus dynamique et réaliste.



Téléphone



Radio

Le téléphone et la radio ajoutent une dimension immersive, en diffusant des dialogues ou des extraits audio. Le téléphone permet d'envoyer des messages et des informations, tandis que la radio diffuse de la musique ou des annonces.

Ces éléments renforcent l'atmosphère et apportent des informations supplémentaires de manière subtile, contribuant à la profondeur de l'expérience.



Couverts



Planche à pain



Poêle



Rouleau



Porte-ustensiles

Les outils présentés, tels que les couverts, la planche à pain, la poêle, le rouleau à pâtisserie et le porte-ustensiles, sont essentiels pour préparer et manipuler les aliments dans la cuisine. Chacun de ces éléments joue un rôle spécifique : les couverts servent à couper, couper et servir les ingrédients, la planche à pain est indispensable pour la préparation des pâtes, la poêle est utilisée pour la cuisson, et le rouleau à pâtisserie permet d'étaler la pâte. Le porte-ustensiles permet de garder ces outils à portée de main et organisés. Ensemble, ils permettent d'optimiser les processus culinaires et assurent une expérience de cuisine fluide et efficace.

2.2 Sauvegardes

Les systèmes de sauvegarde permettent aux joueurs de conserver leur progression tout au long de leur aventure. Cela inclut la sauvegarde des niveaux terminés, les scores et les paramètres de la session. L'implémentation vise à garantir une gestion fluide des données sans compromettre la performance du jeu.

Le système de sauvegarde du jeu repose sur l'écriture et la lecture de fichiers binaires stockés localement sur le disque du joueur. Lorsqu'une sauvegarde est créée ou mise à jour, le jeu rassemble toutes les informations importantes à conserver (nom du restaurant, date, niveau actuel, code d'entrée, options de partie, argent du restaurant, etc.) dans un tableau de chaînes de caractères.

Ce tableau est ensuite écrit dans un fichier binaire grâce à la classe SaveSaver, qui utilise un BinaryWriter pour enregistrer chaque donnée ligne par ligne dans un fichier nommé selon le restaurant (par exemple, save_MonRestaurant.bin). À l'inverse, lors du chargement d'une partie, la classe SaveReader lit ce fichier binaire, extrait chaque paramètre dans le bon ordre, et les réinjecte dans le jeu pour restaurer l'état précédent.

*Sauvegardes du joueur*

Ce système permet de gérer plusieurs sauvegardes distinctes, chaque fichier représentant une partie différente. Les sauvegardes sont utilisées aussi bien en solo qu'en multijoueur : lors de l'hébergement d'une partie multijoueurs, l'hôte peut choisir une sauvegarde existante ou en créer une nouvelle, ce qui permet de reprendre une progression ou de démarrer une nouvelle aventure avec ses amis.

2.3 Scores

Le système de scores enregistre les performances des joueurs à chaque niveau, en tenant compte des critères définis (temps, précision, etc.). L'objectif est d'offrir un retour immédiat sur la performance des cuisiniers du restaurant et d'une potentielle faillite.

Le score dans ce projet est principalement géré dans la classe Computer (voir la propriété Score). Le score représente la performance du joueur ou de l'équipe pendant la partie. Il est stocké dans une variable privée et affiché à l'écran via un composant TextMeshProUGUI. À chaque fois que le score change (par exemple, lorsqu'une commande est réussie ou qu'un objectif est atteint), la propriété Score est mise à jour : elle vérifie que la valeur reste dans des bornes raisonnables (jamais négative, maximum 999) et met à jour l'affichage en conséquence.

Le score est donc un indicateur en temps réel de la réussite du joueur. Il peut être augmenté par différentes actions positives (comme servir des clients rapidement ou éviter des erreurs) et éventuellement diminué en cas d'échec ou de mauvaise gestion. Ce système permet aux joueurs de se fixer des objectifs, de comparer leurs performances et d'ajouter un aspect compétitif ou collaboratif à la partie, surtout en multijoueur. L'affichage du score est pensé pour être clair et motivant, encourageant les joueurs à s'améliorer au fil des parties.

*Scores global et personnel*

2.4 Skins / Capacités joueurs

Dans ce projet, chaque joueur peut personnaliser l'apparence de son personnage grâce à un système de skins. Lors de la création ou de la personnalisation du personnage, le joueur choisit différents éléments : les yeux, le chapeau (classe), la couleur de peau et la tenue. Ces choix sont enregistrés sous forme d'un code (par exemple : "2-1-3-4"), qui est ensuite utilisé pour appliquer les bons sprites et objets visuels au personnage en jeu. Le script PlayerOutfit se charge de lire ce code et d'afficher dynamiquement les bons éléments graphiques sur le personnage, en activant ou désactivant les objets nécessaires (par exemple, le bon chapeau ou la bonne couleur de tenue).

Au-delà de l'aspect visuel, chaque classe de personnage (liée au choix du chapeau) confère des capacités ou bonus spécifiques. Par exemple, un « Cuisinier » cuisine plus vite, un « Serveur » gagne plus de pourboires, ou un « DJ » peut changer la musique. Ces capacités sont définies dans le code et appliquées automatiquement en fonction de la classe sélectionnée par le joueur. Ainsi, le choix du skin n'est pas seulement esthétique : il influence aussi le gameplay et la stratégie de chaque partie, permettant aux joueurs d'adapter leur style de jeu à leurs préférences ou au rôle qu'ils souhaitent jouer dans l'équipe.

En résumé, le système de skins et de capacités permet à chaque joueur de se distinguer visuellement et d'apporter une contribution unique à la partie, renforçant à la fois l'aspect collaboratif et la rejouabilité du jeu.



Personnalisation du skin joueur



Personnalisation du skin joueur

Ce système de personnalisation ne se limite pas à l'apparence : chaque skin s'accompagne d'une classe spécifique, dotée de bonus et de malus influençant directement le déroulement de la partie. Cela ajoute une dimension stratégique importante : les joueurs doivent choisir non seulement en fonction de leurs préférences esthétiques, mais aussi selon le rôle qu'ils souhaitent jouer au sein de l'équipe. Le tableau ci-dessous présente les effets de chaque classe, permettant ainsi à chacun d'optimiser sa contribution selon ses forces et le style de jeu de son équipe.

Classe	Bonus	Malus
Cuisinier	Cuisine 15 % plus rapidement	-5 % de vitesse de déplacement
Serveur	+10 % de pourboires	Cuisine 20 % plus lentement
Bunny Girl	+30 % de pourboires	Ne peut pas prendre d'arme
DJ	Peut changer la musique de la boombox	Cuisine 5 % plus lentement
Réparateur	Répare l'ordinateur 50 % plus rapidement	Cuisine 10 % plus lentement
Propre-sur-soi	+30 % de pourboires	Ne peut pas changer la musique de la boombox
Business Man	+20 % de pourboires	Ne sait pas réparer l'ordinateur
Papa-Noël	+5 % de pourboires, cuisine 10 % plus rapidement	Ne peut pas courir

TABLE 1 – Bonus et malus des différentes classes de personnages

2.5 Collisions

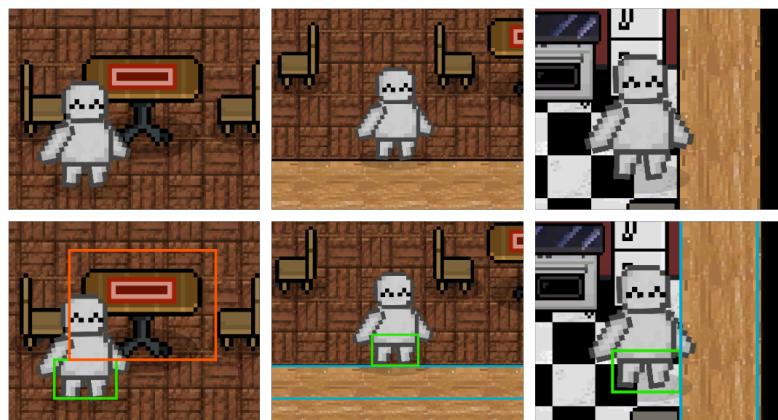
Le jeu intègre un système de collisions permettant des interactions réalistes et fluides entre objets et personnages. L'implémentation des collisions tient compte des différents types d'interactions possibles, telles que les obstacles ou les interactions avec d'autres joueurs. Utilisant un système de Collider, la forme du joueur est compacte et se limite aux jambes pour un effet de superposition au-dessus des objets.

Cette approche donne l'illusion que le personnage passe derrière ou devant certains éléments du décor, renforçant l'impression de profondeur. Elle contribue ainsi à simuler une perspective en fausse 3D, tout en conservant une gestion technique en 2D.

Dans ce projet, la gestion des collisions repose principalement sur le moteur physique intégré de Unity. Chaque objet du jeu qui doit interagir physiquement avec d'autres (personnages, clients, objets du décor, projectiles, etc.) possède un composant Collider (comme BoxCollider2D ou CircleCollider2D) qui définit sa forme de collision.

Lorsque deux objets avec des colliders se rencontrent, Unity détecte automatiquement la collision et peut déclencher des événements ou des réactions selon le code du jeu (par exemple, empêcher un joueur de traverser un mur, déclencher une animation, ou appliquer des effets comme des dégâts).

Les collisions sont utilisées pour rendre le monde du jeu cohérent et interactif : elles empêchent les personnages de traverser les obstacles, permettent de ramasser ou d'utiliser des objets, et gèrent les interactions entre les entités (comme les clients, les rats ou les voitures). Le système de collision fonctionne en temps réel et s'appuie sur les propriétés physiques des objets (masse, vitesse, etc.) pour simuler des comportements naturels. Cela permet d'offrir une expérience de jeu fluide et crédible, où chaque action du joueur ou des éléments du décor a un impact visible et logique sur l'environnement.



Collisions joueur (en vert)

2.6 Combat joueurs

Le système de combat entre joueurs permet des affrontements dynamiques, avec des mécaniques de jeu équilibrées et interactives. L'implémentation se concentre sur la fluidité des combats et l'équilibre des capacités des joueurs pour offrir une expérience compétitive et amusante. Les armes sont disposées dans le restaurant pour être prises et utilisées sans achat au préalable.

Le système de combats entre joueurs dans ce projet s'appuie sur l'utilisation d'armes (comme des objets de type « Melee » ou « Gun ») que les joueurs peuvent ramasser et utiliser pendant la partie. Chaque joueur possède un inventaire qui lui permet de porter un objet à la fois. Lorsqu'un joueur équipe une arme, il peut l'utiliser pour attaquer d'autres joueurs ou des éléments du décor.



Une des armes disposées dans le restaurant

Le script associé à chaque arme (par exemple, Melee.cs pour les armes de corps-à-corps) gère la logique d'attaque : il vérifie si l'arme peut être utilisée (gestion du cooldown entre deux attaques), déclenche l'animation correspondante, et détecte les collisions avec d'autres joueurs à portée.

Lorsqu'une attaque est lancée, le jeu vérifie si un autre joueur se trouve dans la zone d'effet de l'arme (par exemple, à l'aide d'un rayon ou d'une zone de collision). Si c'est le cas, le joueur touché subit les conséquences prévues : cela peut aller d'une simple animation de blessure à la perte de points de vie, voire la mort du personnage (avec apparition de particules de sang, désactivation du contrôle, etc.).



Toutes ces interactions sont synchronisées sur le réseau grâce à Unity Netcode, ce qui garantit que tous les joueurs voient le même résultat en temps réel. Les actions importantes (comme la mort d'un joueur) sont généralement validées par le serveur pour éviter la triche.

Ce système de combat ajoute une dimension compétitive et stratégique au jeu, surtout en mode « versus ». Les joueurs doivent non seulement gérer leur restaurant, mais aussi se défendre ou attaquer les autres pour prendre l'avantage.

Les combats sont pensés pour être dynamiques et accessibles, avec des contrôles simples (clic pour attaquer, touche pour viser ou utiliser une arme) et des retours visuels clairs. L'équilibrage des armes, la gestion des collisions et la synchronisation réseau sont essentiels pour garantir une expérience de jeu fluide et juste pour tous les participants.

2.7 Système d'inventaire

L'inventaire dans ce projet est géré par la classe `Inventory`, qui s'occupe de tout ce qui concerne la gestion des objets portés ou ramassés par le joueur. Chaque joueur possède un inventaire local, représenté par un dictionnaire qui associe le nom de chaque objet à son `GameObject` dans la scène (`LocalPlayerInventory`).

Lorsqu'un joueur ramasse un objet (par exemple, un ingrédient, une arme ou un plat), le script active l'objet correspondant dans l'inventaire du joueur, met à jour l'affichage (sprite de l'objet tenu, animation, etc.) et synchronise cette action avec les autres joueurs via des appels réseau (`ServerRpc` et `ClientRpc`). Si le joueur souhaite déposer ou jeter un objet, l'inventaire désactive l'objet dans la main du joueur et le place dans le décor à l'endroit approprié, en mettant à jour l'état sur tous les clients.

L'inventaire gère aussi des règles spécifiques selon le type d'objet. Par exemple, certains objets ne peuvent pas être jetés mais seulement déposés, d'autres peuvent être cuisinés ou combinés pour créer de nouveaux objets. Le script vérifie également que le joueur ne porte qu'un seul objet à la fois : s'il essaie d'en ramasser un nouveau alors qu'il en a déjà un, l'ancien est automatiquement déposé avant de prendre le nouveau.

Toutes ces opérations sont sécurisées et synchronisées grâce à Unity Netcode, ce qui garantit que l'état de l'inventaire est identique pour tous les joueurs, évitant ainsi les triches ou les désynchronisations.

En résumé, l'inventaire permet au joueur d'interagir de façon fluide et intuitive avec les objets du jeu : ramasser, utiliser, déposer ou jeter des objets se fait en temps réel et de manière cohérente pour tous les participants.

Ce système est essentiel pour le gameplay, car il conditionne la gestion des ressources, la coopération entre joueurs et l'accomplissement des différentes tâches dans le restaurant.

2.8 Effets sonores

Les effets sonores dans ce projet sont générés principalement grâce au système d'audio intégré de Unity, en utilisant des composants AudioSource et des fichiers audio (souvent au format .wav ou .mp3) placés dans le dossier Sounds. Chaque objet ou action du jeu qui doit produire un son possède un ou plusieurs composants AudioSource associés, qui sont configurés dans l'éditeur Unity ou via des scripts.

Par exemple, un objet interactif comme une radio, un four, ou une carafe d'eau possède un champ AudioSource dans son script (voir par exemple Radio_Prop.cs ou WaterJar_Prop.cs). Lorsqu'un joueur interagit avec cet objet (clic, utilisation, réussite ou échec d'une action), le script appelle la méthode Play() sur l' AudioSource correspondant pour déclencher le son approprié.

Le système permet aussi de gérer différents types de sons : bruitages d'interaction, sons d'ambiance, musiques de fond, etc. Certains scripts, comme celui de la radio, utilisent plusieurs AudioSource pour séparer la musique de fond et les effets ponctuels (par exemple, un « clic » quand on allume ou éteint la radio).

Les sons peuvent être joués en boucle, stoppés, ou modifiés en temps réel (volume, pitch, spatialisation) selon les besoins du gameplay. Unity permet également de régler la distance d'audition des sons, ce qui fait que certains effets ne sont audibles que si le joueur est proche de la source, renforçant ainsi l'immersion.

Le système d'effets sonores est conçu pour rendre l'expérience de jeu plus vivante et immersive. Chaque action importante ou événement du jeu est accompagné d'un retour sonore adapté, ce qui aide le joueur à comprendre ce qui se passe et à réagir plus intuitivement.

L'utilisation des outils audio de Unity permet une gestion centralisée, flexible et performante des sons, tout en facilitant l'ajout ou la modification de bruitages au fil du développement.

2.9 Système d'interaction et de tâches

2.9.1 Menu principal

Le menu principal de notre jeu constitue le point d'entrée central de l'expérience utilisateur. Conçu avec une interface intuitive et épurée, il permet aux joueurs de naviguer aisément entre les différentes fonctionnalités du jeu. Parmi les options disponibles, on retrouve :

- **Nouvelle Partie** : Démarrer une session en solo ou en multijoueur.
- **Charger une Partie** : Accéder aux sauvegardes existantes.
- **Options** : Modifier les paramètres graphiques, sonores et les configurations de touches.
- **Crédits** : Découvrir les développeurs et contributeurs du projet.



FIGURE 10 – Interface du menu principal

Une attention particulière a été portée à l'accessibilité, avec des transitions fluides et des animations légères pour une expérience visuelle agréable. Le menu principal joue également un rôle crucial dans l'initialisation des sessions multijoueur, en assurant la gestion des connexions et des sauvegardes.

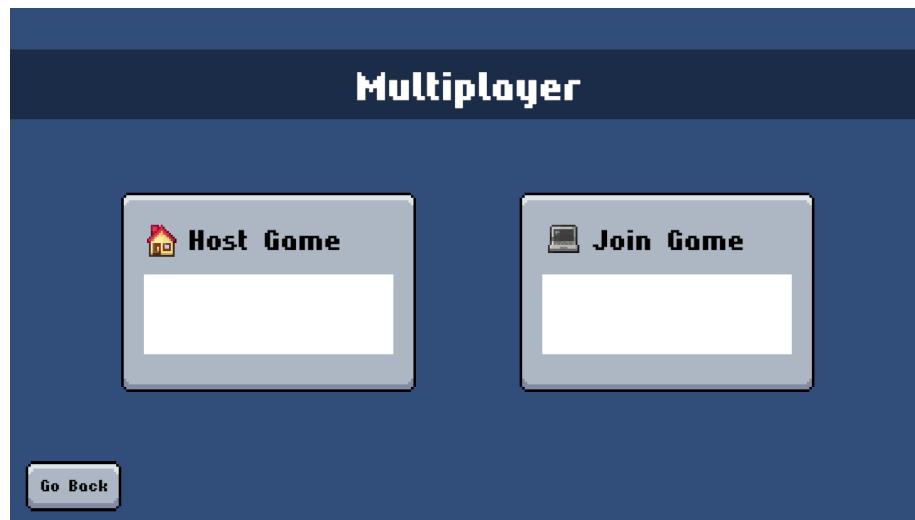


FIGURE 11 – Choix entre héberger ou rejoindre une partie multijoueur

Lorsqu'un joueur choisit d'héberger une partie, il accède à un sous-menu lui permettant de configurer les paramètres de la session, tels que le nom de la partie, le nombre maximum de joueurs et la sélection d'une sauvegarde existante ou la création d'une nouvelle.



FIGURE 12 – Configuration de l'hébergement d'une nouvelle partie

Pour les parties en solo ou les sessions multijoueur, une gestion efficace des sauvegardes est essentielle. Le menu dédié aux sauvegardes offre une vue claire des parties enregistrées, avec des informations telles que la date de la dernière sauvegarde et le temps de jeu total.

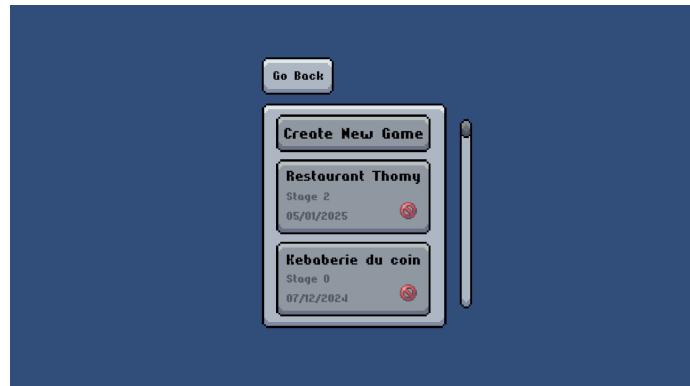


FIGURE 13 – Liste des parties sauvegardées

En résumé, le menu principal est conçu pour offrir une navigation fluide et une gestion efficace des différentes fonctionnalités du jeu, garantissant ainsi une expérience utilisateur optimale dès le lancement du jeu.

2.10 Multijoueurs

Notre projet de jeu Plenty Orders pourra être joué seul ou en multijoueur avec des amis. Dans les deux cas, chaque joueur aura accès à l'ensemble des tâches disponibles dans la cuisine : préparer des plats, nettoyer ou encore commander des ingrédients. Le nombre de clients et les gains générés seront automatiquement ajustés en fonction du nombre de joueurs connectés pour assurer un bon équilibre et éviter que certaines parties ne soient trop faciles.



FIGURE 14 – Menu de création d'une partie multijoueur



FIGURE 15 – Menu pour rejoindre une partie existante via un code

L'expérience multijoueur repose sur une architecture client-serveur utilisant Unity Netcode. Lorsqu'un joueur souhaite créer une partie, il utilise l'interface prévue à cet effet (cf. image ci-dessus) pour héberger la session : il devient alors simultanément serveur et client. Un code unique est généré et peut être transmis aux autres joueurs pour qu'ils rejoignent cette partie.

Les joueurs invités peuvent ensuite entrer ce code via un second menu, pensé pour faciliter la connexion à une partie existante (cf. image ci-dessus). Toutes les communications réseau — synchronisation des objets, mise à jour des variables, gestion des événements — transitent par le serveur, garantissant une expérience cohérente pour tous les participants.

Techniquement, les éléments importants à synchroniser (stocks, argent, commandes, etc.) utilisent des variables ou listes dites **networked**. Toute modification côté serveur est automatiquement propagée aux clients. Des événements comme la connexion, le lancement ou la fin d'une partie sont pilotés à l'aide de **callbacks** et de **Remote Procedure Calls (RPC)**. Cela permet au serveur d'informer efficacement les clients de tout changement significatif.

Cette approche assure que tous les joueurs partagent un état de jeu commun, en temps réel, tout en maintenant une logique centralisée qui évite les triches ou désynchronisations. L'utilisation d'Unity Netcode nous permet ainsi de proposer une infrastructure multijoueur fiable et évolutive.

2.11 Évenements

Dans ce projet, les événements sont des situations spéciales ou des incidents qui peuvent survenir pendant une partie pour dynamiser le gameplay et surprendre les joueurs. Leur activation dépend d'un paramètre choisi dans le menu principal ou lors de la création d'une nouvelle partie : le joueur (ou l'hôte en multijoueur) peut activer ou désactiver les événements via une case à cocher (`MultiplayerHostFrame_EventsEnable`).

Cette option est ensuite transmise à la scène de jeu grâce à la variable statique `Menu.Events`, qui est lue lors de l'initialisation de la partie dans la classe principale `Game`.

Une fois la partie lancée, le système d'événements fonctionne principalement à l'aide de coroutines et de conditions temporelles dans le script principal du jeu (`Game.cs`). À intervalles réguliers, le jeu vérifie si certaines conditions sont réunies pour déclencher un événement : par exemple, l'apparition de l'Inspecteur de l'hygiène, l'arrivée d'un rat dans le restaurant, ou d'autres incidents qui peuvent influencer la partie.

Ces événements sont souvent choisis aléatoirement parmi une liste, et leur déclenchement peut dépendre de la progression du temps, du niveau atteint, ou d'actions spécifiques des joueurs. Lorsqu'un événement est déclenché, il modifie temporairement les règles du jeu, ajoute de nouveaux objectifs ou crée des obstacles inattendus (par exemple, l'Inspecteur peut retirer de l'argent si des infractions sont constatées, ou un rat peut faire fuir les clients).

L'objectif de ce système est d'apporter de la variété et du challenge à chaque partie, en forçant les joueurs à s'adapter à des situations imprévues. Les événements sont pensés pour être suffisamment rares et marquants afin de ne pas perturber le rythme du jeu, mais assez fréquents pour maintenir l'attention et l'engagement des joueurs.

Grâce à leur intégration dans la logique centrale du jeu, ils participent à la rejouabilité et à l'ambiance dynamique du restaurant, tout en restant optionnels pour les joueurs qui préfèrent une expérience plus classique ou prévisible.

2.12 Codes de triche

Dans Plenty Orders, les codes de triche (ou « cheat codes ») ne sont pas accessibles depuis un menu classique ou une interface externe : ils doivent être saisis directement sur un ordinateur en jeu, placé dans certains niveaux du restaurant. Ce PC, intégré de façon diégétique à l'environnement, sert de terminal caché à destination des joueurs les plus curieux ou expérimentés. À première vue décoratif, il devient interactif dès lors qu'un joueur s'en approche et interagit avec son interface texte rudimentaire.



FIGURE 16 – Console de l'ordinateur

L'implémentation de ces cheat codes repose sur la détection de mots-clés spécifiques que le joueur peut taper sur le clavier virtuel de ce terminal. Chaque commande correspond à un effet particulier et est comparée à une base de données définie dans le code du jeu. Si le code est reconnu, l'ordinateur exécute une action particulière dans la partie en cours :

activer un mode « hyper chef » augmentant temporairement la vitesse de cuisson, faire apparaître des ingrédients rares, déclencher une playlist secrète via la boombox, ou encore lancer des événements aléatoires comme une coupure de courant ou un raton-laveur qui vole les commandes.

Ces effets peuvent être purement ludiques ou offrir un avantage stratégique momentané, mais ils sont strictement limités à l'environnement solo ou aux parties privées. Aucun code n'est exploitable en ligne compétitif ou dans les salons publics afin de préserver l'équilibre entre les joueurs.

L'ordinateur en question n'apparaît que dans certains niveaux avancés ou en mode bac à sable, ce qui en fait un outil réservé aux joueurs explorateurs, aux développeurs en phase de test, ou à ceux qui souhaitent casser les règles de façon encadrée.

Le système de triche est modulaire et pensé pour être facilement enrichi de nouveaux codes peuvent être ajoutés dans le script correspondant sans perturber la structure générale du jeu. À terme, il pourrait également intégrer des messages humoristiques, des références cachées à l'équipe de développement, ou des easter eggs à débloquer.

L'objectif de ce système est double : il permet d'une part aux développeurs de valider rapidement certaines fonctionnalités en phase de debug, et d'autre part, il propose aux joueurs une manière alternative de s'amuser, de tester des idées folles, ou de s'immerger davantage dans l'univers du jeu. Le terminal, bien que discret et situé dans un recoin du décor, devient ainsi un espace de liberté ludique et expérimental au cœur du restaurant numérique.

2.13 Dialogues

Le système de dialogues dans ce projet est conçu pour afficher des messages ou des conversations à l'écran, afin de guider le joueur, raconter une histoire, ou fournir des informations contextuelles pendant la partie. Ce système est centralisé dans la classe Dialogue (voir Dialogue.cs), qui gère l'affichage, la file d'attente des messages, et l'animation du texte.

Lorsqu'un script souhaite afficher un dialogue (par exemple, lors d'un tutoriel, d'un événement ou d'une interaction avec un personnage), il appelle la méthode AddDialog(string text). Le texte est alors ajouté à une file d'attente (Queue<string> WaitingDialogs), ce qui permet d'enchaîner plusieurs messages sans qu'ils se superposent.

L'affichage des dialogues est animé : chaque message apparaît lettre par lettre, simulant un effet de « machine à écrire » grâce à une coroutine (DialogueCoroutine). Un son de frappe (Typewriter-Sound) accompagne chaque caractère pour renforcer l'immersion. Le joueur peut choisir d'accélérer ou de passer le dialogue en cliquant sur un bouton « Skip », ce qui affiche immédiatement le texte en entier ou passe au message suivant.

Le système gère aussi la durée d'affichage de chaque message en fonction de sa longueur, pour laisser le temps au joueur de lire confortablement. Une fois tous les dialogues de la file lus, la boîte de dialogue disparaît automatiquement de l'écran.

Ce système de dialogues est utilisé à plusieurs endroits dans le jeu : tutoriels, interactions avec les clients ou les PNJ, notifications importantes, ou encore événements scénarisés. Il permet de structurer la narration et d'offrir une expérience utilisateur claire et interactive, tout en restant flexible et facilement extensible pour ajouter de nouveaux messages ou scénarios au fil du développement.

Grâce à cette architecture, les dialogues contribuent à l'ambiance du jeu et à la compréhension des mécaniques par le joueur, tout en évitant de surcharger l'interface.

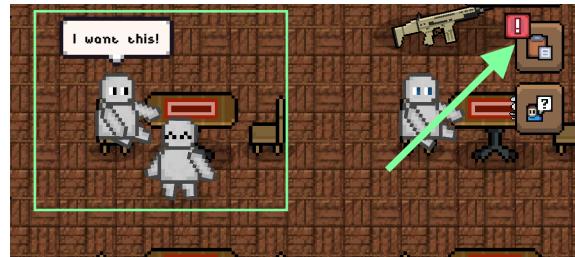
3 Intelligence Artificielle

3.1 Clients

Les clients dans le jeu sont des personnages contrôlés par l'IA qui simulent le comportement de vrais clients dans un restaurant. Lorsqu'un client est généré, il se dirige automatiquement vers une table libre, s'assoit, puis passe une commande (choisie aléatoirement parmi les plats disponibles).

Le joueur doit alors préparer et servir le plat demandé. Si le client reçoit le bon plat dans un délai raisonnable, il mange, paie et quitte le restaurant, libérant la table pour un nouveau client. Si le service est trop lent ou incorrect, le client peut s'impatienter, réduire le pourboire ou partir mécontent.

Ce système permet de créer un flux continu de commandes et de défis pour le joueur, tout en rendant l'expérience de gestion de restaurant vivante et dynamique. Les interactions avec les clients sont gérées à la fois par des scripts d'IA et par des événements réseau pour assurer la synchronisation en multijoueur.



Interaction avec un client et ajout de leur commande dans une liste partagée entre joueurs



Commandes des clients à remplir



Animations aléatoires des clients

3.2 Inspecteur

L'inspecteur est un événement spécial qui peut survenir de façon aléatoire pendant une partie, principalement lorsque l'option « événements » est activée dans le menu de création de partie. Son apparition est gérée par le script principal du jeu (Game.cs).

À intervalles réguliers, le jeu vérifie si les conditions sont réunies pour déclencher l'arrivée de l'inspecteur : cela dépend du temps écoulé, d'un tirage aléatoire, et du fait que l'inspecteur ne soit pas déjà venu récemment. Si toutes les conditions sont remplies, l'inspecteur apparaît dans le restaurant, généralement sous la forme d'un personnage non-joueur (PNJ) contrôlé par l'IA.

Une fois présent dans le restaurant, l'inspecteur se comporte comme un client particulier : il observe l'environnement, vérifie la propreté, la gestion des stocks, et le respect des règles d'hygiène. Le script associé à l'inspecteur va alors analyser différents paramètres du restaurant : présence de rats, tables sales, clients mécontents, ou autres infractions.

Si des problèmes sont détectés, l'inspecteur peut infliger des pénalités au joueur, comme une amende (retrait d'argent), une baisse de la réputation, ou d'autres conséquences négatives. À l'inverse, si tout est en ordre, il peut féliciter le joueur ou simplement repartir sans incident.

L'objectif de l'inspecteur est d'ajouter une dimension de pression et de gestion de la qualité au jeu. Les joueurs doivent non seulement s'occuper des clients et de la cuisine, mais aussi veiller en permanence à la propreté et au bon fonctionnement du restaurant pour éviter les sanctions.

L'arrivée de l'inspecteur est annoncée par un message ou une animation spécifique, afin que les joueurs puissent réagir rapidement (nettoyer, chasser les rats, etc.). Ce système encourage une gestion rigoureuse et ajoute du réalisme, tout en rendant chaque partie plus dynamique et imprévisible.

En multijoueur, il favorise la coopération : les joueurs peuvent se répartir les tâches pour préparer au mieux l'inspection et limiter les risques de pénalités.

3.3 Rat

Dans ce projet, le rat est un événement spécial qui peut survenir aléatoirement pendant une partie, principalement pour ajouter du challenge et du dynamisme à la gestion du restaurant. Le rat est géré par un script dédié (souvent nommé Rat.cs ou similaire) et son apparition est contrôlée par le script principal du jeu (Game.cs).

À intervalles réguliers, le jeu vérifie si les conditions sont réunies pour faire apparaître un rat : cela dépend généralement de l'activation des événements dans les options de partie, du temps écoulé, et d'un tirage aléatoire. Si toutes les conditions sont remplies, le rat est instancié à un point d'apparition prédéfini dans le restaurant.

Une fois apparu, le rat se déplace de façon autonome dans la salle, en suivant un comportement d'IA simple : il peut se diriger vers des zones spécifiques (comme la cuisine ou la salle principale), chercher de la nourriture ou simplement se déplacer de manière erratique pour gêner les joueurs.

Sa présence a des conséquences directes sur le déroulement de la partie : il peut effrayer les clients, réduire leur satisfaction, ou même provoquer leur départ si le rat n'est pas rapidement éliminé. Les joueurs doivent donc réagir vite, soit en chassant le rat à la main, soit en utilisant des objets spécifiques (comme un balai ou un piège) pour l'attraper ou le faire fuir.

Le système de rat est pensé pour surprendre les joueurs et les forcer à adapter leur stratégie en temps réel. Il ajoute une dimension supplémentaire à la gestion du restaurant, en introduisant un élément de stress et d'imprévu.

En multijoueur, il encourage la coopération : un joueur peut se charger de la cuisine pendant qu'un autre s'occupe de la chasse au rat. L'apparition et la disparition du rat sont synchronisées entre tous les joueurs grâce au système réseau, garantissant que tout le monde voit le même événement au même moment. Ce mécanisme contribue à rendre chaque partie unique et à renforcer l'immersion dans l'univers du jeu.



4 Site Web

Le site web de *Plenty Orders*, développé par l'équipe de Storcki Corporation, constitue une vitrine numérique du projet, pensée pour offrir une expérience claire, accessible et immersive à tous les visiteurs, qu'ils soient joueurs, développeurs curieux ou partenaires potentiels.

Chaque élément a été conçu pour refléter l'identité visuelle du jeu tout en assurant ergonomie, accessibilité et simplicité d'utilisation.

4.1 Accessibilité et Paramètres Utilisateur

Dès l'arrivée sur le site, l'utilisateur est accueilli par une interface épurée comprenant une barre de paramètres située en haut de page. Cette barre permet un accès immédiat aux fonctionnalités essentielles : activation du mode daltonien et changement de langue.

Le mode daltonien est une fonctionnalité d'accessibilité pensée pour les utilisateurs souffrant de troubles de la vision des couleurs ou ayant besoin d'un contraste plus marqué pour des raisons médicales ou de confort visuel.

Un simple clic sur le bouton dédié transforme instantanément les couleurs de l'interface en une version en noir et blanc à haut contraste, sans altérer l'organisation des contenus.

Le site propose également une bascule de langue dynamique, permettant aux visiteurs de choisir entre le français et l'anglais. Dès que la langue est modifiée, tous les textes visibles, titres, boutons et menus sont immédiatement mis à jour sans rechargement de page. Cette flexibilité permet une expérience fluide et inclusive.

Ces paramètres sont conservés tout au long de la navigation, à l'exception de la page de contact, qui redirige directement vers notre profil Instagram.



FIGURE 19 – Barre de paramètres en haut de page (mode daltonien et sélection de langue)

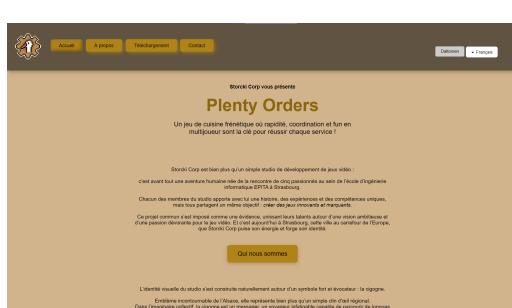


FIGURE 20 – Interface de sélection de langue

4.2 Page d'accueil

La page d'accueil est pensée comme une introduction chaleureuse à l'univers du jeu. Elle reprend les couleurs dominantes du logo et de l'interface de *Plenty Orders*, ainsi que des éléments visuels évoquant l'esthétique du jeu de cuisine : illustrations de plats, textures rappelant les matériaux de cuisine, typographie ludique.

On y trouve une présentation brève de Storck Corporation, une équipe jeune et passionnée formée dans le cadre de notre projet à EPITA. Des appels à l'action guident naturellement le visiteur vers les autres sections du site : découvrir l'équipe, télécharger le jeu, ou en apprendre plus sur son développement.



(a) Introduction visuelle de la page d'accueil

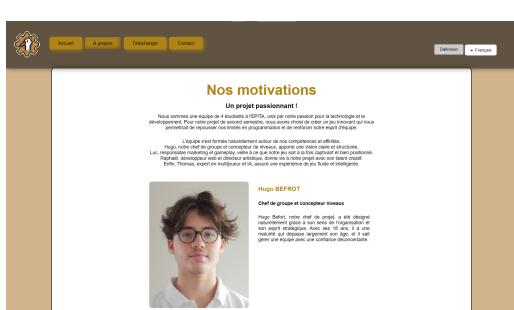


(b) Vue intégrée dans la continuité du design de jeu

4.3 Section “À propos”

La section “À propos” met en lumière les membres de l’équipe derrière *Plenty Orders*. Chaque développeur est présenté avec une photo, son rôle précis (programmation, design, marketing, etc.), et une brève description. Cette page adopte un fond inspiré de l’ambiance du restaurant fictif du jeu, créant un lien visuel fort entre le site et l’univers du gameplay.

Ce design immersif renforce la cohérence de la communication du projet et permet de souligner les personnalités qui ont contribué à sa création.



(a) Présentation d'un membre de l'équipe



(b) Description des rôles et ambiance de page personnalisée

4.4 Section Téléchargement

La section téléchargement est structurée pour guider l'utilisateur avec efficacité. En haut de la page se trouve un bouton principal permettant d'obtenir la dernière version du jeu, toujours à jour, dans un format d'installation optimisé pour une compatibilité maximale sur les ordinateurs Windows.

Plus bas, des liens additionnels permettent de récupérer des documents clés pour les curieux ou partenaires techniques : cahier des charges, rapports de soutenance, ou encore une version exécutable autonome pour tester le jeu rapidement sans installation.



FIGURE 23 – Bouton de téléchargement principal



FIGURE 24 – Accès aux documents techniques et version autonome

4.5 Page de contact

La page de contact propose une redirection simple vers le compte Instagram officiel de Storcki Corporation. Cette décision vise à centraliser la communication de manière moderne et interactive, en profitant des avantages des réseaux sociaux : accès rapide aux actualités, partage de visuels du développement, et échanges directs avec notre communauté.

5 Niveaux

Dans *Plenty Orders*, l’expérience du joueur repose sur une progression à travers plusieurs niveaux aux mécaniques évolutives. Cette structure permet de stimuler l’engagement des joueurs tout en introduisant progressivement de nouveaux éléments de gameplay. Chaque niveau est conçu pour apporter un défi supplémentaire, non seulement en augmentant la difficulté, mais aussi en enrichissant l’environnement, les interactions, et la coordination entre joueurs.

Le développement des niveaux a été pensé pour rester fidèle à l’univers du jeu, inspiré de la restauration rapide dans un contexte coopératif. Le joueur y incarne un membre d’une brigade de cuisine devant réaliser un maximum de commandes dans un temps imparti. L’objectif n’est pas seulement de réussir les recettes, mais aussi d’apprendre à communiquer efficacement, répartir les tâches, et s’adapter rapidement aux imprévus. Ces éléments prennent de l’ampleur à mesure que le joueur progresse dans les différents niveaux.

Chaque niveau a été soigneusement prototypé, testé, puis ajusté à l’aide de retours utilisateurs recueillis pendant la phase de bêta-test. Le rythme, la complexité des recettes, la disposition des éléments de cuisine, ou encore les événements aléatoires ont été calibrés pour offrir une courbe de progression cohérente et gratifiante. Un soin particulier a été apporté à l’équilibrage entre tension et plaisir, afin de maintenir une dynamique collaborative sans frustrer les joueurs.

Enfin, le style graphique et sonore évolue également légèrement au fil des niveaux, afin de créer une sensation de découverte continue. Cette dimension immersive contribue à renforcer l'implication des joueurs et leur désir d'atteindre les niveaux suivants. Le jeu étant multijoueur, la coopération est d'autant plus valorisée que certains niveaux nécessitent une parfaite coordination entre les membres de l'équipe pour être réussis.

La section suivante détaillera les caractéristiques et les intentions de design de chaque niveau actuellement intégré au jeu, en mettant en lumière les choix techniques et artistiques qui les soutiennent.

6 Tutoriel

Avant de plonger dans les niveaux plus complexes, les joueurs de *Plenty Orders* passent par un niveau tutoriel conçu pour introduire les mécaniques fondamentales du jeu. Ce tutoriel a pour but de familiariser les utilisateurs avec les commandes de base, l'interface, ainsi que le déroulement d'une commande en cuisine.

L'objectif principal est de proposer une prise en main fluide, accessible aussi bien aux débutants qu'aux joueurs plus expérimentés. Il permet d'appréhender les principes de déplacement, d'interaction avec les éléments de cuisine, ainsi que les premières recettes. Un soin particulier a été apporté à la clarté des instructions visuelles et textuelles pour guider le joueur sans l'assister excessivement.

Le tutoriel se déroule dans un environnement simplifié et sans pression de temps, afin de permettre l'expérimentation. Il joue un rôle clé dans l'immersion initiale et contribue à la satisfaction des premiers instants de jeu, tout en posant les bases d'une progression cohérente vers les niveaux suivants.



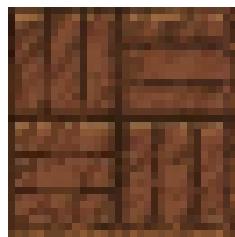
FIGURE 25 – Aperçu du tutoriel

6.0.1 Textures

Les textures du niveau tutoriel ont été conçues pour instaurer un environnement accueillant et lisible, facilitant l'apprentissage des mécaniques de jeu. Simples et épurées, elles permettent au joueur de se concentrer sur les interactions essentielles sans être distrait par des détails superflus. L'objectif est de créer un espace clair et fonctionnel, évoquant une cuisine moderne et accessible, avec des matériaux reconnaissables tels que le carrelage, l'inox ou les surfaces de travail en bois clair.

Chaque texture est pensée pour favoriser une lisibilité maximale : les zones interactives sont bien délimitées, et les objets de jeu — ustensiles, plans de travail, appareils — se distinguent facilement du décor. Ce choix permet de guider le regard du joueur et d'éviter toute confusion lors de ses premières actions dans le jeu.

Visuellement, le tutoriel sert également de base esthétique à l'ensemble du jeu. Les textures utilisées respectent la direction artistique globale, avec un style semi-réaliste légèrement stylisé. Elles sont volontairement sobres afin de ne pas surcharger l'espace et de permettre une transition naturelle vers les environnements plus thématiques des niveaux suivants.



Sol Tutoriel



Mur Tutoriel

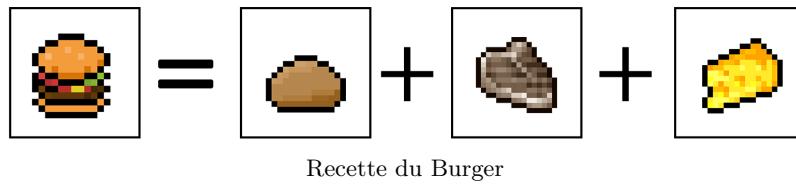
6.0.2 Plats

Le burger est le tout premier plat que les joueurs apprennent à cuisiner dans le niveau tutoriel. Simple, emblématique et immédiatement reconnaissable, il sert d'introduction idéale aux mécaniques de base du jeu.

La recette est volontairement accessible : elle nécessite trois ingrédients à assembler dans un ordre logique après cuisson, ce qui permet aux joueurs de se familiariser avec les différentes étapes de préparation :

- **Pain cuit** – à toaster légèrement pour révéler une texture dorée.
- **Steak cuit** – à cuire jusqu'à obtenir une belle coloration.
- **Fromage** – à poser directement sur le steak, apportant une touche fondante et gourmande.

Ce plat sert d'exercice d'entraînement pratique à la manipulation des aliments et des machines de la cuisine.



6.1 Niveau 1

6.1.1 Présentation Niveau 1

Le premier niveau de *Plenty Orders* se déroule en Égypte, dans l'intérieur d'une maison. C'est le premier vrai niveau après le tutoriel. Le joueur y évolue de manière plus autonome, avec moins d'indications à l'écran.

Le style graphique reste le même que dans le tutoriel, avec des textures adaptées au thème égyptien : murs en pierre, sols sableux, portes en bois sculpté. L'interface et les mécaniques sont identiques, ce qui permet une continuité fluide dans l'apprentissage du jeu.



FIGURE 28 – Aperçu du Niveau 1

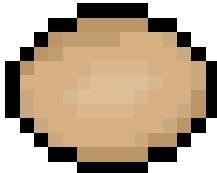
6.1.2 Plats

Les plats du Niveau 1 sont conçus pour offrir une expérience simple et claire tout en introduisant les mécaniques de base du jeu. L'objectif est de permettre aux joueurs de se familiariser facilement avec la préparation des plats tout en maintenant une esthétique agréable.

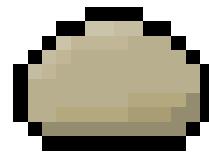
Chaque plat est assez simple visuellement, avec des couleurs et des formes distinctes afin de faciliter l'identification des ingrédients. Le design est fonctionnel, avec un équilibre entre lisibilité et attractivité. Cela permet aux joueurs de comprendre rapidement comment chaque composant interagit.



Thé noir



Crêpes



Pâte à pain crue

La réalisation de la crêpe dans le niveau 1 repose sur une préparation simple mais essentielle, permettant d'introduire les mécanismes de cuisson dans le jeu. La pâte à pain crue est versée dans une poêle chauffée, où elle prend progressivement la forme d'une crêpe. Ce processus de cuisson met en avant les bases de la gestion des ressources et des outils. L'image ci-dessous illustre cette étape clé, montrant la crêpe en pleine cuisson.



FIGURE 30 – Recette de Crêpes

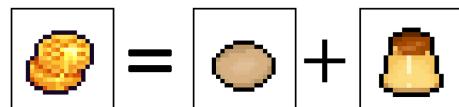


FIGURE 31 – Recette waffle

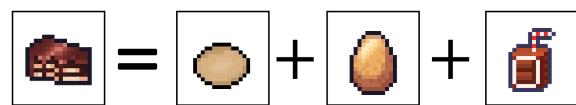


FIGURE 32 – Recette gateau au chocolat



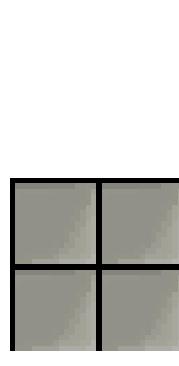
FIGURE 33 – Recette pancake aux pommes

6.1.3 Textures

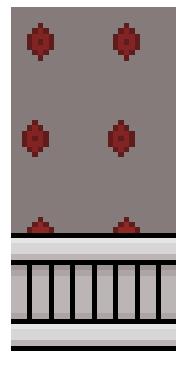
Les textures des éléments du Niveau 1, qui se déroule en Égypte, sont simples et claires, facilitant l'identification des objets et matériaux caractéristiques de l'époque. L'objectif est de créer une atmosphère visuelle en adéquation avec le thème égyptien, en intégrant des éléments reconnaissables comme le sable, les murs en pierre et les objets de la vie quotidienne. Ces textures seront à la fois réalistes et stylisées, suffisamment détaillées pour être intéressantes visuellement tout en restant facilement distinguables.

Chaque texture est soigneusement pensée pour s'harmoniser avec l'univers graphique global du jeu, tout en offrant une variété de matériaux typiques de l'Égypte ancienne, comme le calcaire, les tissus ou les objets artisanaux. L'accent sera mis sur des détails subtils, tels que les motifs sur les murs ou les textures de la pierre, qui enrichissent l'expérience sans encombrer l'image.

Bien que les textures du Niveau 1 restent simples et épurées, elles serviront de base pour des évolutions futures, en se complexifiant à mesure que le joueur progresse dans le jeu, et en introduisant des éléments plus raffinés et détaillés dans les niveaux suivants.



Sol Égypte



Mur Égypte

Mur extérieur
Egypte

Porte Égypte

6.2 Niveau 2

6.2.1 Présentation Niveau 2

Le troisième niveau se déroule au Japon, dans un cadre traditionnel rappelant les cuisines de rue ou les izakayas. Les plats à réaliser sont plus nombreux et les commandes s'enchaînent plus vite, augmentant ainsi la pression sur les joueurs.

Le style graphique reste fidèle à l'identité du jeu, avec des textures évoquant le bois clair, les cloisons en papier et des éléments décoratifs typiques japonais.



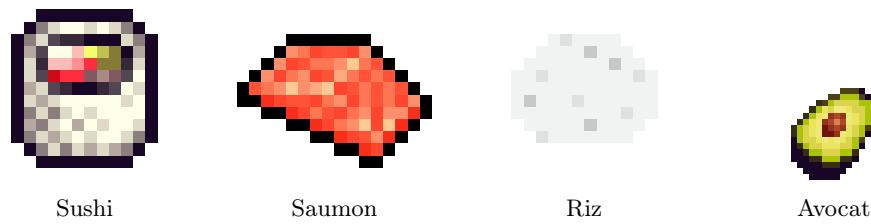
FIGURE 35 – Aperçu du Niveau 2 : salle à manger



FIGURE 36 – Aperçu du Niveau 2 : cuisine

6.2.2 Plats

Les plats du Niveau 2, centré sur le Japon, doivent introduire les mécaniques de jeu tout en restant visuellement clairs et cohérents avec le thème japonais. L'ajout de sushis au menu permet de diversifier les plats tout en conservant des formes et couleurs distinctes, facilitant leur identification pour le joueur.



Sushi

Saumon

Riz

Avocat

Les sushis dans le jeu sont réalisés avec quatre ingrédients principaux : le saumon, le riz et de l'avocat. Le saumon est coupé en fines tranches, l'avocat en lamelles, et le riz est préparé pour être utilisé dans la recette. Ces ingrédients sont ensuite combinés pour former un sushi, qui sera visuellement représenté dans le jeu. L'objectif est de permettre au joueur de suivre les étapes de préparation de manière simple, avec une interface claire et intuitive.

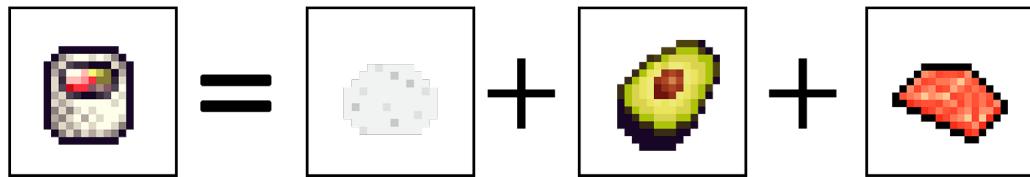


FIGURE 38 – Recette de Sushi

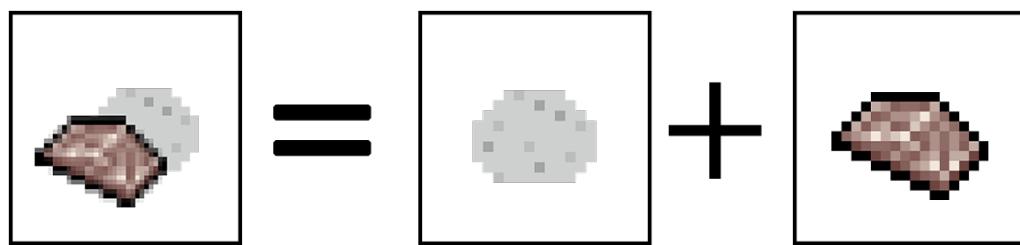
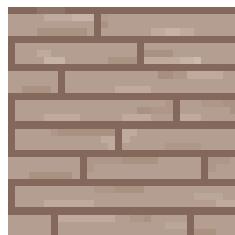


FIGURE 39 – Recette riz au saumon

6.2.3 Textures

Les textures des plats du Niveau 2, inspiré du Japon, devront être plus détaillées que celles du Niveau 1, ajoutant une dimension réaliste tout en restant lisibles. Chaque texture devra refléter les spécificités culinaires japonaises, comme la délicatesse du riz, la brillance du poisson cru ou les détails des légumes, avec des nuances et des textures fines pour offrir une expérience immersive.

Les textures évolueront au fur et à mesure que les plats se diversifient, en intégrant des éléments visuels propres à la cuisine japonaise, comme les motifs subtils des sushis ou l'aspect lisse et élégant de la soupe miso. Les textures devront s'adapter à la complexité croissante des recettes tout en restant fidèles à l'esthétique japonaise.



Sol Japon



Mur Japon



Porte Japon

6.3 Niveau 3

6.3.1 Présentation Niveau 3

Le troisième niveau se déroule en Italie, dans une cuisine moderne inspirée des trattorias. La difficulté augmente légèrement, avec des recettes plus longues à exécuter et une disposition plus complexe des postes de cuisine.

Comme dans les niveaux précédents, le style graphique reste cohérent. Les textures ont été adaptées au thème italien avec des matériaux comme le carrelage, le bois sombre et des éléments décoratifs typiques.



FIGURE 41 – Aperçu du Niveau 3 : salle à manger



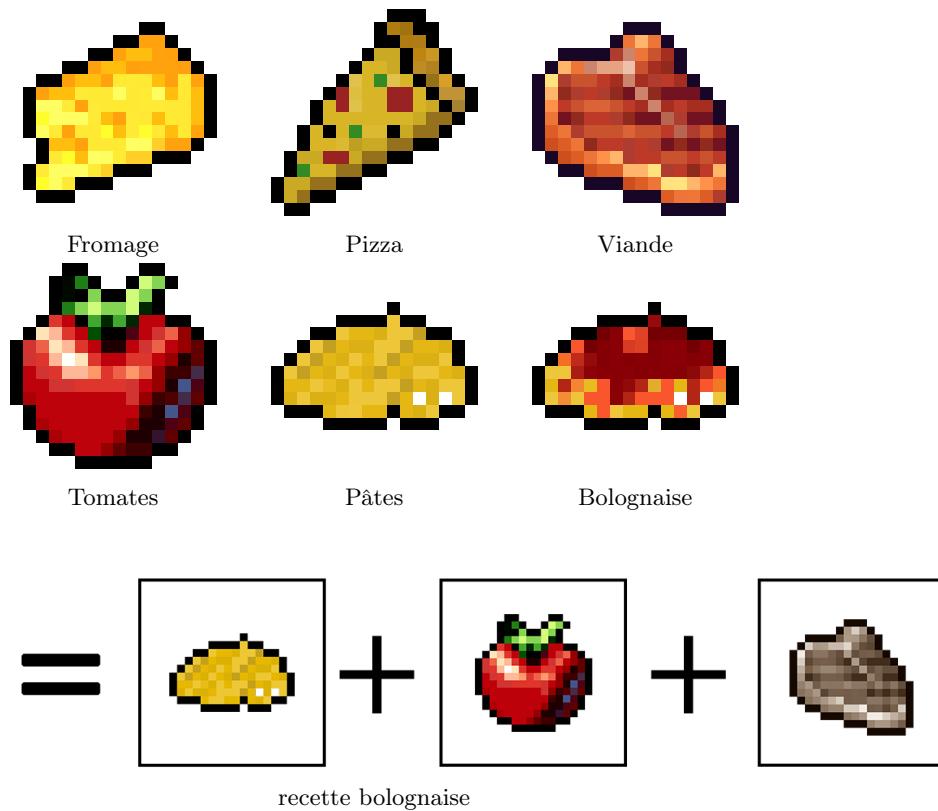
FIGURE 42 – Aperçu du Niveau 3 : cuisine

6.3.2 Plats

Les textures des plats du Niveau 3, inspiré de l'Italie, doivent évoquer la richesse et la chaleur de cette cuisine méditerranéenne tout en conservant une lisibilité claire. Chaque ingrédient est représenté de manière stylisée mais reconnaissable, qu'il s'agisse de la pâte, du fromage, des tomates, de la viande ou de la pizza elle-même.

Les textures visent à retranscrire les particularités visuelles de chaque élément : les pâtes affichent un ton beige doré avec des formes courtes et ondulées, les tomates brillent d'un rouge vif, et le fromage fond légèrement sur les surfaces chaudes. La viande se distingue par des nuances brun foncé et une texture grillée, tandis que la pizza combine ces ingrédients dans un ensemble cohérent.

Ces textures s'intègrent à l'univers visuel global du jeu, en apportant une cohérence avec le style coloré et accessible de Plenty Orders. Comme dans les niveaux précédents, l'objectif reste de maintenir un bon équilibre entre esthétique et clarté, en mettant en valeur la cuisine italienne tout en assurant une reconnaissance rapide des ingrédients en jeu.



6.3.3 Textures

Les textures des plats du Niveau 3, inspiré de l'Italie, conserveront une direction artistique lisible tout en introduisant plus de détails visuels. Les ingrédients comme la pâte à pizza, le fromage fondu, la sauce tomate ou encore la viande seront représentés de manière stylisée mais reconnaissable, pour renforcer l'immersion sans nuire à la clarté du gameplay.

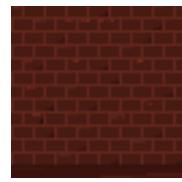
Chaque texture sera pensée pour évoquer les codes visuels de la cuisine italienne, en jouant sur les couleurs chaudes, les contrastes et les formes. Les plats emblématiques comme les pizzas ou les pâtes auront un aspect appétissant et identifiable au premier coup d'œil.



Sol Italie



Mur Italie

Mur extérieur
Italie

Porte Italie

6.4 Niveau 4

6.4.1 Présentation Niveau 4

Ce niveau se déroule au Mexique, dans une ambiance chaleureuse et vivante, fidèle à l'imaginaire collectif lié à la culture mexicaine. Le joueur évolue dans un environnement inspiré des maisons traditionnelles mexicaines, où les couleurs vives et les matériaux typiques comme la terre cuite ou le bois brut renforcent l'immersion.

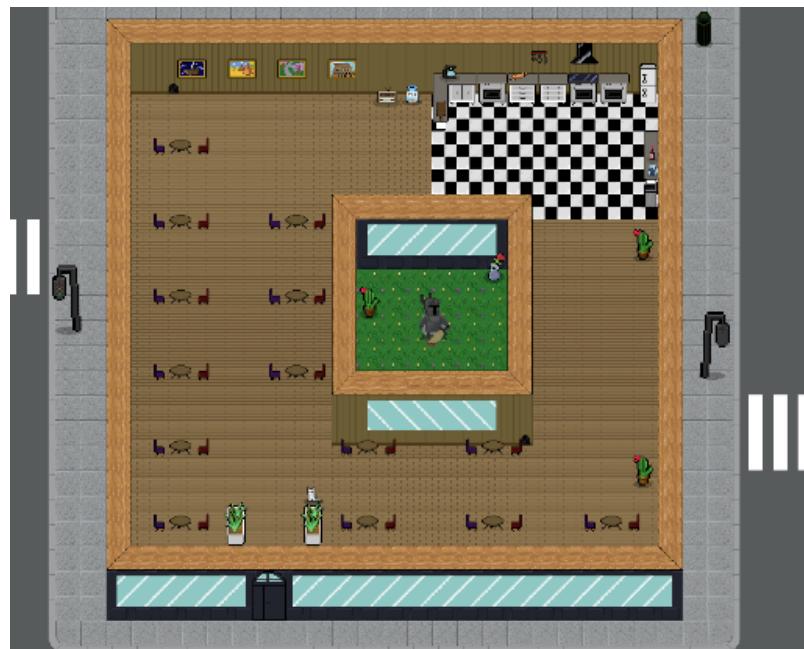


FIGURE 46 – Aperçu du Niveau 4

6.4.2 Plats

Comme pour les niveaux précédents, le style graphique reste cohérent avec l'univers général du jeu, mais intègre des éléments visuels propres au Mexique. La progression du joueur est toujours plus libre, avec moins d'indications directes, l'invitant à développer ses réflexes et son autonomie.

Les textures des plats du Niveau 4, inspiré du Mexique, continueront d'allier lisibilité et détails. Les aliments typiques comme le burrito, les tacos et la boisson Coca-Cola seront représentés avec des textures claires et colorées, afin d'évoquer l'ambiance mexicaine tout en facilitant leur reconnaissance rapide.

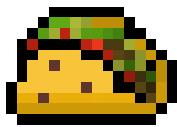
L'accent sera mis sur des textures contrastées et des formes simples, pour rendre chaque plat visuellement distinct et cohérent avec l'univers graphique du jeu.



Coca-Cola



Burrito



Tacos



Salade



=



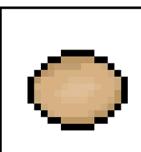
+



+



+



recette burrito



=



+



+



+



recette tacos

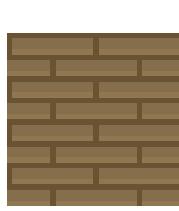


6.4.3 Textures

Les textures des plats du Niveau 4, inspiré du Mexique, combinent simplicité et clarté pour permettre une identification rapide des aliments typiques tels que le burrito, les tacos, la salade fraîche et la boisson Coca-Cola. Chaque texture est pensée pour évoquer l'ambiance chaleureuse et colorée du Mexique tout en restant lisible et cohérente avec l'esthétique globale du jeu.

Ces textures mettent en avant des couleurs vives et des formes distinctes, reflétant les caractéristiques visuelles propres aux plats mexicains. Les détails, comme les feuilles croquantes de la salade ou les motifs sur les tacos, sont subtils mais suffisants pour enrichir l'expérience visuelle sans surcharger l'image.

Comme pour les niveaux précédents, ces textures simples servent de base solide et pourront être enrichies dans de futurs niveaux, suivant la progression du joueur et l'évolution du jeu.



6.5 Niveau 5

6.5.1 Présentation Niveau 5

Le Niveau 5 marque le retour en France, offrant un cadre familier et chaleureux aux joueurs. Le joueur devra gérer efficacement les commandes pour réussir ce niveau, en s'adaptant aux recettes et aux ingrédients typiques de la région.

Le style graphique reste cohérent avec les niveaux précédents, garantissant une expérience visuelle homogène tout au long du jeu.



FIGURE 53 – Aperçu du Niveau 5



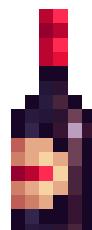
FIGURE 54 – Aperçu du Niveau 5

6.5.2 Plats

Les textures des plats du Niveau 5, de retour en France, doivent refléter l'authenticité et la richesse de la cuisine française traditionnelle. Chaque texture est conçue pour montrer les particularités des aliments, comme la croûte dorée et croustillante du croque-monsieur, la finesse du jambon et la couleur profonde du vin.

Les textures doivent rester claires et lisibles, tout en apportant une touche réaliste qui invite à l'immersion. La variété des matériaux, du pain grillé au liquide du vin, nécessite une attention particulière aux détails, afin d'équilibrer réalisme et style graphique cohérent avec le reste du jeu.

Cette approche garantit que les plats du niveau 5 s'intègrent parfaitement à l'univers graphique global, tout en valorisant la gastronomie française présente dans ce niveau.



Vin

6.5.3 Textures

Les textures des plats du Niveau 5, qui marque le retour en France, doivent représenter fidèlement la cuisine traditionnelle française avec simplicité et clarté. Chaque texture met en valeur les caractéristiques propres à chaque aliment, comme le croustillant du croque-monsieur, la finesse du jambon tranché, ou encore la transparence et la profondeur de la couleur du vin.

L'objectif est de conserver une lisibilité optimale tout en apportant un certain réalisme qui enrichit l'expérience visuelle. Les détails subtils, tels que les grains du pain grillé ou les reflets dans le verre de vin, contribuent à l'immersion sans surcharger l'image.

Ces textures respectent le style graphique global du jeu, sobre et efficace, et s'adaptent parfaitement à l'ambiance du niveau.



7 La réalisation du projet

7.1 Ses joies

7.1.1 Ses joies collectives

Assister à la transformation du concept initial en produit abouti a été une grande source de fierté pour toute l'équipe *Storcki Corp.* Chaque aspect désormais fonctionnel, de la jouabilité intégrale du Niveau 1 aux niveaux supérieurs plus élaborés, témoigne du travail assidu et de l'évolution constante du projet.

Un des moments les plus mémorables a été la première session multijoueur jouée dans des conditions quasi identiques à celles de la version finie : observer les joueurs interagir, se défier ou collaborer dans un univers que nous avons imaginé et développé de A à Z a été extrêmement gratifiant.

L'incorporation réussie de mécaniques sophistiquées, telles que l'intelligence artificielle des clients et les éléments *PvP*, a véritablement enrichi le gameplay.

Le retour des bêta-testeurs, très enthousiastes face à l'évolution du jeu, a consolidé notre confiance dans la qualité de l'expérience que nous proposons. Les éloges reçus sur l'univers graphique homogène, les animations fluides, l'ergonomie des interfaces ou encore l'ambiance sonore ont renforcé notre sentiment d'accomplissement.

Ce projet nous a permis non seulement de développer nos compétences techniques et créatives, mais aussi de vivre une véritable aventure humaine.

Une des grandes joies a été de pouvoir enfin se déplacer et effectuer des actions basiques dans le jeu. Cela a marqué un tournant majeur, car l'équipe a pu véritablement voir ses idées prendre forme de manière tangible. Le projet commence réellement à prendre forme et chaque nouvelle fonctionnalité implémentée donne à l'équipe un sentiment de satisfaction et de progression. De plus, les retours des bêta-testeurs ont été extrêmement positifs, ce qui a renforcé notre motivation et nous a permis d'avoir une vision plus claire de ce que les joueurs attendent du jeu. Leur enthousiasme a renforcé notre conviction que nous sommes sur la bonne voie.

Nous avons particulièrement apprécié l'intégration des armes dans le jeu, un ajout qui a non seulement enrichi l'expérience mais aussi ouvert la porte à de nouvelles mécaniques de gameplay. Nous avons hâte de développer le côté PvP, qui, selon nous, apportera une dimension encore plus excitante et compétitive à l'expérience du joueur.

Le style graphique choisi, l'homogénéité visuelle entre les différents éléments du jeu (boutons, plats, restaurants) et la cohérence des textures contribuent à créer une expérience immersive et engageante. Les succès rencontrés sur la fluidité des systèmes de sauvegarde, de scores ou encore sur l'intégration des collisions témoignent de notre capacité à transformer des idées en fonctionnalités concrètes et fonctionnelles.

Chaque étape franchie représente un pas de plus vers la finalisation du projet, et chaque problème résolu ajoute à la satisfaction de voir les pièces du puzzle se mettre en place.

Nous avons hâte de voir comment les joueurs réagiront face à ces nouvelles mécaniques et de recueillir leurs retours pour continuer à améliorer l'expérience de

7.1.2 Ses joies personnelles



Hugo BEFORT
Chef de projet

Participer à ce projet a été une source de grande satisfaction sur plusieurs plans. En tant que chef de projet, j'ai eu la chance de suivre le déroulement de la conception, de voir les progrès concrets et d'accompagner l'équipe dans la réalisation de ce projet.

L'une de mes plus grandes joies a été d'observer comment les idées, parfois floues ou ambitieuses au départ, ont progressivement pris forme jusqu'à devenir un jeu complet, cohérent et fonctionnel.

La dynamique d'équipe, l'entraide collective et les nombreuses petites victoires, qu'il s'agisse d'une fonction qui fonctionne enfin, d'un retour enthousiaste d'un testeur ou simplement d'un moment de travail fluide, ont rendu cette aventure à la fois exigeante et gratifiante.



Thomas GRILLO
Développeur

Ce projet m'a permis d'explorer des aspects techniques que je n'avais jamais vraiment approfondis auparavant, notamment en gestion de sauvegardes et logiques de gameplay. Voir certaines de mes idées être intégrées dans le jeu final a été très gratifiant.

Ce que j'ai le plus apprécié, c'est le sentiment de construire quelque chose de tangible en équipe, avec des résultats visibles et testables. Même si tout n'était pas parfait, la satisfaction d'avoir concrétisé un projet complet reste forte.



Raphaël CHABOISSEAU
Développeur

Ce projet m'a permis de renforcer mes compétences techniques, notamment sur la partie interface et interactions. Même si certaines tâches étaient longues et parfois rébarbatives, j'ai pris du plaisir à voir le jeu devenir de plus en plus jouable et visuellement agréable.

Travailler dans un cadre concret, avec des objectifs clairs et une équipe soudée, a été une expérience très formatrice. Le fait de voir les joueurs tester ce que j'avais contribué à créer a été l'un des meilleurs moments du projet.



Malgré les difficultés techniques rencontrées, j'ai vraiment apprécié le sentiment de progression tout au long du développement. Pouvoir dire « ça, c'est moi qui l'ai fait » sur certaines parties du jeu, c'est une vraie fierté.

J'ai aussi beaucoup aimé les moments de travail collectif, quand on avançait ensemble vers un objectif commun. Ces instants ont donné du sens à l'effort fourni, au-delà du code lui-même.

Luc BEVILACQUA
Développeur

7.2 Ses peines

7.2.1 Ses peines collectives

Malgré les progrès réalisés, le développement de *Plenty Orders* a rencontré plusieurs défis techniques. Des problèmes d'optimisation des sauvegardes ont ralenti certains aspects du développement, affectant la fluidité du jeu pendant un temps. L'équipe a dû travailler dur pour résoudre ces problèmes et assurer une expérience de jeu fluide.

Les collisions ont pris plus de temps que prévu pour être totalement fonctionnelles, mais ces problèmes ont été résolus juste avant la deadline. Cela a retardé l'intégration de certaines mécaniques importantes, mais tout est désormais en ordre.

Une recette identique pour deux aliments différents a causé des confusions, mais nous avons corrigé cela lors de la conception des niveaux pour clarifier les choses.

La période du projet PFA a été la plus difficile, car nous n'avons eu que peu de temps pour nous consacrer au projet. Cependant, cela n'a pas provoqué de conflits dans l'équipe et nous avons su rester productifs malgré la pression.

Nous aurions aimé avoir plus de temps pour intégrer certaines fonctionnalités non prévues initialement, comme un mode de jeu compétitif local, un système de progression plus étoffé ou encore un menu principal plus complet.

La gestion du multijoueur a également posé des défis, notamment en termes de synchronisation et de réactivité entre les clients.

7.2.2 Ses peines personnelles



Hugo BEFORT
Chef de projet

L'un des aspects les plus complexes pour moi a été de composer avec les contraintes de temps et la charge de travail fluctuante. Il aurait été appréciable d'avoir davantage de latitude pour tester et améliorer certaines fonctionnalités.

Gérer le suivi de chaque tâche tout en prenant des décisions rapidement n'a pas toujours été évident, et j'ai parfois eu le sentiment de devoir faire des compromis trop tôt. Cela dit, cela m'a permis de mieux appréhender les réalités d'un projet en conditions limitées.



Thomas GRILLO
Développeur

J'ai parfois été frustré par le manque de temps pour peaufiner certains systèmes, comme les sauvegardes. Ces fonctionnalités ont nécessité des ajustements de dernière minute, ce qui a pu générer du stress.

Même si le résultat final est fonctionnel, je garde l'impression qu'avec un peu plus de temps, certains éléments auraient pu être mieux optimisés ou enrichis.



Raphaël CHABOISSEAU
Développeur

Le travail sur les visuels et les interactions m'a demandé plus de temps que prévu. Certaines tâches m'ont paru répétitives et chronophages, notamment en fin de projet.

Il a parfois été difficile de garder une vision d'ensemble avec la pression des deadlines, et j'aurais aimé avoir plus de recul pour ajuster certaines décisions prises un peu vite.



Les mécaniques liées aux collisions m'ont donné du fil à retordre, et leur intégration a pris plus de temps que je ne l'imaginais. Certaines fonctionnalités ont dû être simplifiées ou reportées pour respecter le planning.

Ce n'était pas toujours satisfaisant sur le moment, mais cela m'a permis de mieux comprendre les compromis nécessaires dans un projet collectif.

Luc BEVILACQUA
Développeur

8 Conclusion

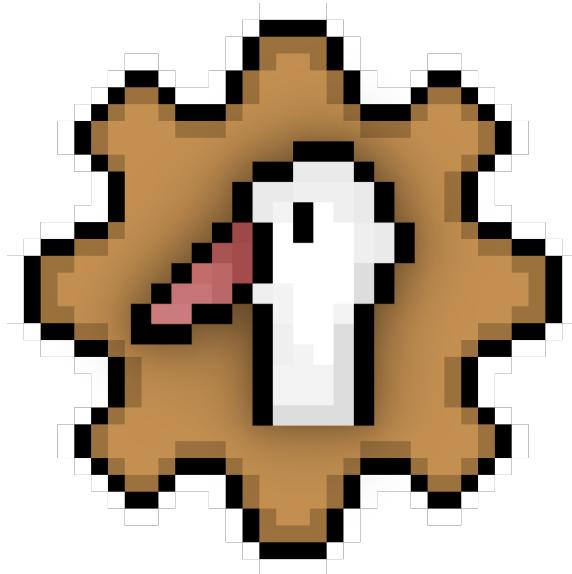
Le développement de Plenty Orders a représenté une expérience riche, mêlant créativité, contraintes techniques et travail d'équipe. Malgré les difficultés rencontrées, chaque membre a su s'adapter, apprendre et progresser au fil du projet. Le jeu final, bien que perfectible sur certains aspects, constitue un aboutissement concret de plusieurs mois d'efforts, de réflexion et de collaboration.

Au-delà du résultat, cette aventure a été l'occasion de mettre en pratique des compétences variées : conception de mécaniques de jeu, gestion de projet, design graphique, développement technique, et surtout, coordination entre profils aux rôles complémentaires.

Si nous avions davantage de temps, certaines fonctionnalités auraient pu être approfondies ou ajoutées, mais nous sommes fiers du chemin parcouru et de la cohérence du produit livré. Plenty Orders restera pour nous un projet marquant, aussi bien sur le plan humain que technique.

C'est sur ces quelques phrases que se concluent ce rapport de soutenance et cette aventure.

Storcki Corp, là où la créativité et la technologie s'envoient



STORCKI CORP.