|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Elément/ processus | Fonction | Mode  **Journal des risques**  **Projet Blind**  Gicquel Gregoire,  Poncet Thomas,  Benharara Donia | Cause | Effet | G | F | D | C | Action préventive | Action corrective et/ou ameliorative |
| Intégration Kinect | La Kinect permet de créer un squelette dans le moteur de jeu et ainsi de recréé le mouvement du joueur dans le jeu | Disfonctionnement du matériel.  Le matériel ne s’allume plus, n’est plus utilisable. | Matériel fournis en mauvais état.  Faux contact | Remplacement du matériel.  Remboursement éventuel de la personne qui a fournis le matériel | 3 | 1 | 3 | 9 | Connaissance de fournisseur.  Taches ne dépendant pas du matériel en question pour ne pas ralentir la production.  Simulation remplaçant l’utilisation du matériel. | Rachat du matériel défectueux dans les plus brefs délais. |
|  |  | Intégration du matériel dans Unity ne fonctionne pas.  Pas de retour de données. | Problème de compatibilité.  SDK non valide. | Délais plus long sur cette tâche.  Recherche de solution. | 2 | 2 | 2 | 8 | Recherches préalable sur le matériel et ses possibilités. | Réitéré les recherches et tester un autre sdk et/ou une autre version de unity pour connaitre la source du problème. |
| Intégration Gant CyberGlove | Permet le retour d’informations tactiles depuis le moteur de jeu sur le gant | Disfonctionnement du matériel.  Stimulateurs vibrotactiles  Défectueux.  Câble, batterie ou convertisseur usb défectueux | Matériel ancien, couteux et très fragile. | Fournisseur du matériel mécontent.  Ralentissement du développement | 4 | 1 | 3 | 12 | Attention particulière portée à ce matériel fragile.  Rangement et utilisation adéquat. | Soit le matériel est encore utilisable bien que très altéré.  Soit le matériel n’est plus utilisable et il faudra trouver une solution pour faire sans. |
|  |  | Incompatibilité avec les technologies actuelles. | Code source trop ancien. | Ralentissement du développement, délais accordé à l’amélioration du code source pour le rendre compatible. | 3 | 2 | 2 | 12 | Recherches préalable mais peu d’informations. | Contournement du problème en recréant du code. |
| Développement  Scripting sous Unity | Mise en place du Game Play | Panne d’ordinateur. | Non connue,  Virus, ordinateur ancien. | Délais accordé à la réparation de l’ordinateur.  Développement stoppé pour le membre du groupe. | 3 | 1 | 4 | 12 | Ordinateur de dépannage. | Réparation de l’ordinateur et utilisation de l’ordinateur de dépannage durant le temps de réparation. |
|  | Mise en place de la structure et du Game Play du jeu. | Plantage de l’application. | Code mal effectué | Soit la méthode adoptée n’est pas adapté, soit l’erreur produite est dû à une inattention. | 1 | 4 | 2 | 8 | Prendre soin de connaitre la structure du code que l’on veut produire avant de commencer à coder. | Procéder à une phase de débug. |
|  | Gestion du combat se déroulant à l’aveugle. | Combat injouable à l’aveugle. | Manque de recul. | La méthode utilisé n’est pas adapter et ne répond pas aux objectifs fixés. | 3 | 2 | 2 | 12 | Tester régulièrement à l’aveugle lors du développement.  Travail en amont pour définir le déroulement d’un combat. | Revoir entièrement le système de combat.  Voir ce qui existe en termes de jeux accessibles. |
|  |  | Fatigue, perte de motivation d’un ou plusieurs membres du groupe. | Trop de charge de travail en parallèle. | Baisse de la productivité, la perte de motivation peut être communiquée aux l’autres membre du groupe | 3 | 2 | 1 | 6 | Une bonne organisation préalable permet d’améliorer l’implication. | Faire une pause complète de tous les projets en cours pendant au moins 1 jour. |
|  | Structure du code et mise en commun | Mésentente entre les membres du groupe | Si l’un des membres n’approuve pas les méthodes d’un autre membre. | Tension entre les membres du groupe.  Ralentissement de la productivité. | 2 | 2 | 2 | 8 | Communication des structures que l’on veut adopter au niveau du code lors du développement. | Nettoyage et optimisation du code non approuvé pour qu’il puisse répondre aux attentes de tout le monde et s’adapter correctement aux autres parties. |
|  |  | Abandon d’un membre du groupe. | Abandon du cursus à l’ESGI ou abandon du projet. | Ressource humaine réduite et trop faible.  Augmentation de la charge de travail pour chaque membre. | 4 | 1 | 2 | 8 | Communiquer régulièrement | Redistribution des tâches ou remplacement du membre. |
| Phase d’enregistrement | Equivalent d’un home studio, permet d’enregistrer les voix et de les traiter. | Défaillance du matériel d’enregistrement. | Mauvaise manipulation ou non connue. | On ne peut plus enregistrer de voix. | 3 | 1 | 4 | 12 | Enregistrer d’une autre manière mais de moins bonne qualité |  |
|  | Enregistrement des voix / dialogues | Délais tendu | Membres pas toujours disponible pour enregistrer,  Nous ne sommes pas acteur. | Retard dans le projet | 2 | 2 | 2 | 8 | Avance prise par rapport à l’élaboration des dialogues pour passer à la phase d’enregistrement le plus tôt possible. | On peut modifier les voix de sorte à pouvoir remplacer un membre non disponible pour enregistrer. |
|  |  | Besoin de réécrire les dialogues ou scénario. | Le scénario mis en place ne convient pas ou plus. | Retard dans le projet. | 3 | 2 | 2 | 12 | Faire un point régulier pour valider l’écriture des dialogues | Réécriture des dialogues. |
| Intégration des sons et voix |  | Impossibilité de se repérer à l’aveugle grâce aux sons et voix | Les techniques d’intégration des sons choisie de convienne pas | L’objectif du projet n’est pas atteint | 3 | 2 | 2 | 12 | Recherche préalable sur les différents types de son et leur diffusion | Revoir la méthode appliquer et en changer si celle-ci n’est pas améliorable |
| Livraison du produit Final | Rendu final, ampleur du produit voir commercialisation | Problème de licences |  |  |  |  |  |  |  |  |

**Evaluation (méthode AMDEC)**

**Le calcul de la criticité.**

Avec :

G = Gravité

F = Fréquence

D = Détectabilité

Criticité C = G \* F \* D

**Evaluation des critères.**

**Pour la gravité :**

|  |  |
| --- | --- |
| Valeur de G | Critères de gravité |
| 1 | Défaillance mineure d’un des outils technique utilisé ou du code mis en place provoquant un ralentissement de la production en ne sollicitant qu’un seul membre du groupe jusqu’à résolution estimé entre 1 à 2h. |
| 2 | Défaillance moyenne d’un ou des outils technique utilisé ou bien du code source mis en place. Nécessite une remise en état et/ou du debug sollicitant un ou plusieurs membre du groupes. Gros ralentissement voir arrêt de la production jusqu’à résolution estimé entre 1 à 8h. |
| 3 | Défaillance importante nécessitant l’intervention de tous les membres du groupe et stoppant complétement la production, peux amener à un remplacement matériel ou la suppression de ressources. Résolution estimé entre 1 et 7 jours. |
| 4 | Défaillance catastrophique impliquant une panne irréparable d’un des outils technique utilisé, d’un code source non conforme ou bien des solutions appliqué ne répondant plus aux problématiques du projet. Sollicitation de tous les membres du groupe et arrêt complet de la production pendant plus de 7 jours |

**Pour la fréquence :**

|  |  |
| --- | --- |
| Valeur de F | Critères de fréquence (probabilité d’apparition de la défaillance) |
| 1 | Défaillance inexistante (peut potentiellement arriver une seule fois lors de l’élaboration du projet). |
| 2 | Défaillance occasionnelle, déjà rencontrée, posant le plus souvent problème (moins d’une fois par mois). |
| 3 | Défaillance régulière, permet à la résolution de solidifier les systèmes mis en place et de s’assurer du bon fonctionnement final (1 à 4 fois par mois). |
| 4 | Défaillance systématique, intervient le plus souvent lorsque les tests sont effectués, ou si ces mêmes tests n’ont pas été effectués correctement (au moins une fois par semaine). |

**Pour la détectabilité :**

|  |  |
| --- | --- |
| Valeur de D | Critères de détectabilité |
| 1 | Ce type de défaillance est prévisible est peut-être évité ou donner lieu à une recherche d’information préalable. |
| 2 | Ce type de défaillance peut-être prévisible dans le cadre de test, d’intégration ou d’opération n’ayant jamais été effectué auparavant. |
| 3 | Ce type de défaillance n’est pas facilement décelable notamment si la défaillance concerne les outils techniques utilisé, ou si des tests avait préalablement été effectué au niveau du programme. |
| 4 | Ce type de défaillance n’a montré aucun signe avant-coureur. |

**Identification des actions de prévention et/ou de réduction :**

La finalité de l'analyse AMDEC, après la mise en évidence des défaillances critiques, est de

définir des actions de nature à traiter le problème identifié.

**Les actions sont de 3 types :**

Actions préventives :

On agit pour prévenir la défaillance avant qu'elle ne se produise, pour

l'empêcher de se produire. Ces actions sont planifiées. La période

d'application d'une action résulte de l'évaluation de la fréquence.

Actions correctives :

Lorsque le problème n'est pas considéré comme critique, on agit au moment

où il se présente. L'action doit alors être la plus courte possible pour une

remise aux normes rapide.

Actions amélioratives :

Il s'agit en général de modifications de procédé ou de modifications

technologiques du moyen de production destinées à faire disparaître

totalement le problème. Le coût de ce type d'action n'est pas négligeable

et on le traite comme un investissement.