# Informations générales

*Je vous encourage à faire ce projet en équipe de 2 pour pouvoir vous entraider avec les nouvelles technologies à apprendre, mais les projets individuels seront quand même acceptés.*

*Une seule remise par équipe, pour toutes les 3 parties du projet.*

**Nom(s) du ou des coéquipiers** : Louis-Philippe Forget, Samuel Gignac

**Repository GitHub :** <https://github.com/SamuelGignac/chernobyl>

Table des matières

[Informations générales 1](#_Toc148018142)

[Description du projet *complet* 2](#_Toc148018143)

[Technologies candidates 3](#_Toc148018144)

[Plateformes de déploiement (matériel, logiciels, services cloud, etc.) 4](#_Toc148018145)

[Sources de données (bases de données, API web ou locale, etc.) 5](#_Toc148018146)

[Langages, librairies et *frameworks* 5](#_Toc148018147)

[Environnement de développement 7](#_Toc148018148)

[Autres technologies 8](#_Toc148018149)

[Choix et justification des technologies 9](#_Toc148018150)

[Proposition de prototype 10](#_Toc148018151)

# Description du projet *complet*

*Inscrivez ci-dessous une courte description (environ 5 lignes) de votre idée de projet « complet » (celui que vous aimeriez réaliser si le temps n’était pas une contrainte – nous ferons seulement un prototype dans le cadre du cours).*

*N’oubliez pas d’indiquer quelle est la « nouvelle » technologie que vous aller utiliser pour ce projet.*

Le "Biome Radioactif" est une proposition de mod pour Minecraft qui introduirait un biome extrêmement dangereux et contaminé par la radioactivité. Ce biome apporterait un nouveau niveau de défi aux joueurs intrépides qui oseraient s'y aventurer.

Caractéristiques principales du biome radioactif :

* Effet de Brouillard : Le biome serait enveloppé d'un épais brouillard, réduisant la visibilité des joueurs et créant une atmosphère sinistre.
* Radioactivité (Poison) : Les joueurs qui pénètrent dans ce biome seraient exposés à une radioactivité nocive. Cela entraînerait une forme de poison progressive, drainant la santé du joueur tant qu'il reste dans la zone contaminée.
* Arbres Morts : Le biome serait peuplé de sombres arbres morts, dépourvus de feuillage et ayant un aspect sinistre. Les joueurs pourraient récolter des ressources uniques de ces arbres.

En plus du biome radioactif, le mod inclurait d'autres éléments :

* Bombes : Les joueurs pourraient fabriquer des bombes en utilisant des matériaux tels que le fer, la redstone, le TNT, l'uranium et le plutonium. Les bombes auraient une texture et un effet sonore distinctifs, ainsi qu'un effet d'explosion dévastateur.
* Réacteur Nucléaire : Un réacteur nucléaire pourrait être construit pour générer de l'énergie redstone, mais il nécessiterait une gestion appropriée pour éviter les accidents.
* Chapeau d'Oppenheimer : Les joueurs pourraient obtenir un "Chapeau d'Oppenheimer" spécial en explorant le biome radioactif. Ce chapeau unique serait un objet de collection avec des propriétés spéciales.
* Enchantement "Plomb" sur l'Armure : Les joueurs pourraient améliorer leur armure avec l'enchantement "Plomb" qui fournirait une protection supplémentaire contre la radioactivité et les effets nocifs du biome.
* Succès Débloqué : Lorsqu'un joueur fabrique et utilise sa première bombe nucléaire, il débloquerait un succès intitulé "Now I have become death, the destroyer of worlds," rendant hommage à la célèbre citation de J. Robert Oppenheimer.

Ce prototype de projet vise à ajouter un élément de science-fiction et de survie à Minecraft, tout en intégrant des éléments historiques liés à J. Robert Oppenheimer et au développement des armes nucléaires. Il encouragerait les joueurs à explorer ce biome dangereux tout en gérant les risques associés à la radioactivité*. La description du prototype est à la fin de ce document (une fois les technologies choisies).*

*RAPPEL : Venez discuter avec moi de votre idée de projet (si ce n’est pas déjà fait) pour valider qu’il est suffisant, réaliste, et qu’il contient bien au moins une nouvelle technologie.*

# Technologies candidates

Ci-dessous, on veut lister les technologies qui pourraient convenir au besoin précédent, organisées selon leur catégorie.  
Vous pouvez indiquer « aucun » s’il n’est pas nécessaire de préciser le choix d’une technologie pour une catégorie donnée.

Pour chaque catégorie, vous devez remplir le tableau suivant en cherchant et documentant quelles technologies seraient des candidates potentielles afin de répondre au besoin du client.  
*Conseil : Mieux vaut proposer plus de technologies que moins, on fera le choix des technologies les plus adaptées plus tard, une fois toutes les options sur la table.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nom de la technologie | Description sommaire | Avantages | Limitations ou incertitudes | Références et exemples | Analyse (votre jugement) |
| Le nom de la technologie | Un court résumé de ce que ça fait (dans le cadre du besoin ci-haut). | Quelles sont les forces de cette technologie?   * Facilité d’apprentissage * Bonne documentation * Très populaire, établie depuis longtemps * Etc… | Quelles sont les faiblesses de cette technologie?   * Faible performance * Coût ou licence * Courbe d’apprentissage * Peu d’exemples connus * Etc… | Listez les sites, articles ou vidéos qui vous ont aidés à comprendre la technologie. Des exemples de projets similaires sont aussi très utiles. | Écrivez une ou deux phrases qui indiquent si vous pensez que cette technologie serait adéquate ou non au besoin, et pourquoi. |
| Ce tableau n’est qu’un exemple, n’inscrivez rien ici, utilisez plutôt les tableaux des catégories spécifiques dans les prochaines pages :  [Plateformes de déploiement (matériel, logiciels, services cloud, etc.) 2](#_Toc144118370)  [Sources de données (bases de données, API web ou locale, etc.) 3](#_Toc144118371)  [Langages, librairies et *frameworks* 3](#_Toc144118372)  [Environnement de développement 3](#_Toc144118373)  [Autres technologies 3](#_Toc144118374) | | | | | |

## Plateformes de déploiement (matériel, logiciels, services cloud, etc.)

Quelle sorte de matériel ou quelles alternatives (services web, virtualisation, etc.) sont nécessaires pour répondre au besoin?

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nom de la technologie | Description sommaire | Avantages | Limitations ou incertitudes | Références et exemples | Analyse (votre jugement) |
| Minecraft Launcher (Java) | Il s'agit du logiciel officiel qui permet aux joueurs de démarrer la version Java de "Minecraft". Ce lanceur donne accès aux différentes versions du jeu, permet de gérer les profils de joueur, d'installer des mises à jour et de personnaliser les paramètres avant de lancer le jeu. | * Accès à différentes versions * Intégration facile avec des mods * Mises à jour automatiques | * Seul Launcher Java officiel * Moins optimisé que, par exemple, la version Bedrock * Peut être complexe | <https://minecraft.fandom.com/wiki/Minecraft_Launcher>  <https://minecraft.miraheze.org/wiki/Minecraft_Launcher> | Je pense que ce serait la meilleure option pour nous, car nous l’avons déjà utilisé. C'est aussi le seul launcher Java officiel et le plus utilisé. Finalement, il supporte Forge qui est le mod loader que nous espérons utiliser. |
| Minecraft Launcher (Bedrock) | Il s'agit du logiciel officiel qui permet aux joueurs de démarrer la version Bedrock de "Minecraft". Contrairement à la version Java, l'édition Bedrock est optimisée pour un jeu multiplateforme, incluant Windows 10, consoles, mobiles et d'autres dispositifs. Il présente des performances souvent améliorées par rapport à l'édition Java. | * Crossplay * Meilleure performance que la version Java * Mises à jour unifiées | * Modding plus limité mais toutefois possible * Force l’utilisation de certains services indésirables ex: Xbox Live * Force l’utilisation de C++ (pas mauvais mais nous préférons Java) | <https://minecraft.fandom.com/wiki/Bedrock_Edition> | Il pourrait être intéressant d’un jour tenter de modder la version Bedrock, mais nous préférons l’utilisation de Java. C’est aussi avec cette version que nous avons toujours joué au jeu. |

## Sources de données (bases de données, API web ou locale, etc.)

Si l’application aura besoin de conserver ou d’obtenir des données d’une DB ou d’une API, quelles technologies ou services pourraient les fournir?

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nom de la technologie | Description sommaire | Avantages | Limitations ou incertitudes | Références et exemples | Analyse (votre jugement) |
| AUCUN | AUCUN | AUCUN | AUCUN | AUCUN | AUCUN |

## Langages, librairies et *frameworks*

Quels langages de programmation et quelles librairies ou *frameworks* de ces langages pourraient répondre au besoin?  
*Conseil : Le choix des librairies ou frameworks est souvent plus important que le choix du langage lui-même, donc mieux vaut commencer par là.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nom de la technologie | Description sommaire | Avantages | Limitations ou incertitudes | Références et exemples | Analyse (votre jugement) |
| Java 17 | Java est un langage de programmation polyvalent et orienté objet qui a été développé par Sun Microsystems (acquis par Oracle Corporation) dans les années 1990. Il est conçu pour être portable, sécurisé, et robuste, ce qui signifie qu'il peut être utilisé pour développer des applications sur une variété de plates-formes matérielles et logicielles tout en minimisant les problèmes de sécurité et de stabilité. | * Implémenté par Minecraft. * Documentation forte * Bonne portabilité * Orienté objet | * Performance * Garbage collector peut ralentir le programme * Grande consommation de mémoire dû à la JVM | [JavaDoc](https://docs.oracle.com/en/java/javase/17/) | Excellent langage que nous connaissons bien alors pour ça se sera plus simple. |
| C++ | C++ est un langage de programmation puissant et polyvalent, adapté à une grande variété de domaines et d'applications. Il offre aux développeurs un contrôle précis sur la mémoire et les ressources, tout en leur permettant de tirer parti de la programmation orientée objet. Cependant, le fait de gérer manuellement la mémoire peut être plus complexe et nécessiter une gestion minutieuse pour éviter les problèmes. | * La performance est la plus grande force de C++ * Langage de programmation polyvalent * Compatible avec le langage C * Utilisé par Minecraft BedRock | * Langage complexe * Gestion manuelle de la mémoire * Courbe d’apprentissage raide | * [Wiki C++](https://fr.wikipedia.org/wiki/C%2B%2B#:~:text=C%2B%2B%20est%20un%20langage,objet%20et%20la%20programmation%20g%C3%A9n%C3%A9rique) * [C++ officiel](https://cplusplus.com/doc/tutorial/) * [En apprendre plus](https://www.w3schools.com/cpp/) | Ce langage de programmation est excellent pour du bas niveau. Dans notre cas à nous puisque le modificateur se fera dans la version Java le langage Java sera utilisé. |
| Kotlin | Kotlin est un langage de programmation moderne conçu pour fonctionner de manière transparente avec Java, ce qui facilite la transition pour les développeurs Java. Il est reconnu pour sa concision, sa lisibilité et son expressivité, réduisant la quantité de code nécessaire pour accomplir des tâches courantes. Kotlin prend en charge la programmation orientée objet et fonctionnelle, permettant aux développeurs de choisir la meilleure approche pour un problème donné. Il offre également une sécurité accrue en éliminant les exceptions de pointeur nul courantes en Java grâce à son système de types nullable. Enfin, Kotlin est utilisé pour le développement d'applications Android, ainsi que pour une variété d'autres types de projets, en raison de sa polyvalence et de son efficacité. | * Simple et très lisible * Interopérable avec Java * Réduction des NPE | * Taille de la documentation et de la communauté est faible * Apprentissage initial peut être difficile * Interopérabilité est parfois imparfaite | * [Kotlin.org](https://kotlinlang.org/) * [En apprendre plus](https://www.w3schools.com/KOTLIN/index.php) * [Wiki Kotlin](https://en.wikipedia.org/wiki/Kotlin_(programming_language)) | Ce langage de programmation est en pleine croissance, il n’est pas encore assez connu pour qu’on puisse dédier trop de ressource et que finalement on trouve une faille dans le système. |
| Forge | Minecraft Forge est un outil qui permet la création et l'intégration de mods pour Minecraft. Il facilite la compatibilité entre différents mods, offre des outils aux développeurs et simplifie l'installation des mods pour les utilisateurs. | * Environnement standardisé * De nombreux outils pour les développeurs * Communauté active | * Pas le plus performant * Les mises à jour peuvent être lentes * Pas le plus sécure | <https://docs.minecraftforge.net/en/latest/>  <https://forge.gemwire.uk/wiki/Main_Page>  <https://www.youtube.com/@ModdingByKaupenjoe/videos> | Je pense que Forge serait la meilleure option pour nous, car très populaire et bien documenté. C’est aussi celui dont nous avions entendu parler. |
| Fabric | Fabric est un mod loader léger pour Minecraft, permettant aux joueurs d'installer et aux développeurs de créer des mods. Contrairement à d'autres systèmes, il est réputé pour sa rapidité de mises à jour avec les nouvelles versions de Minecraft et sa modularité, offrant ainsi une grande flexibilité pour le modding. | * Léger et rapide comparé à ses pairs * Mises à jour rapides * Utilise Mixin qui est reconnu pour rendre le développement plus flexible | * Moins établi que Forge * Moins stable * Courbe d’apprentissage très élevée | <https://fabricmc.net/wiki/start>  <https://www.youtube.com/@ModdingByKaupenjoe/videos> | Fabric semble être une très bonne alternative à Forge. Nous nous inquiétons toutefois du fait qu’il soit plus récent et moins établi. |

## Environnement de développement

Pour développer le logiciel répondant au besoin, quels outils ou processus de développement seraient utiles ou nécessaires?  
Certaines plateformes nécessitent parfois l’utilisation de logiciels spécifiques pour le développement d’applications; Si c’est le cas, il faut les identifier ici.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nom de la technologie | Description sommaire | Avantages | Limitations ou incertitudes | Références et exemples | Analyse (votre jugement) |
| IntelliJ | IntelliJ IDEA est un environnement de développement intégré (IDE) développé par JetBrains, largement utilisé pour la programmation en Java et d'autres langages. Il offre une gamme de fonctionnalités avancées, telles que l'achèvement automatique de code, la détection d'erreurs en temps réel et des outils de refactorisation puissants, qui facilitent le développement d'applications de manière efficace. IntelliJ IDEA est apprécié pour son interface utilisateur conviviale et ses nombreuses extensions et plugins qui étendent ses fonctionnalités pour divers langages de programmation et cadres de travail. | * + - * Productivité améliorée (Surtout en Java)       * Polyvalent pour plusieurs langages       * Écosystème riche en extensions | * + - * L’IDE est couteux       * Programme lourd pour l’ordinateur       * Apprentissage de base difficile | [IntelliJ](https://www.jetbrains.com/idea/) | Je pense que cet éditeur de texte sera parfait pour nos besoins en Java. Le cout ne sera pas un problème puisque nous avons une licence éducative grâce au cégep |
| Visual Studio Code | Visual Studio Code (VS Code) est un éditeur de code source léger et extrêmement populaire développé par Microsoft. Il se distingue par sa polyvalence et sa prise en charge de nombreux langages de programmation, avec des extensions pour presque tous les besoins de développement. Il offre des fonctionnalités telles que l'achèvement automatique du code, la détection d'erreurs en temps réel, l'intégration Git, et une communauté active qui fournit une multitude d'extensions et de thèmes personnalisés. | * + - * Gratuit       * Open source       * Large éventail d’extensions | * + - * Lorsqu’on est dans un projet volumineux la performance diminue drastiquement       * Moins d’intégration profonde       * Moins de fonctionnalités d’entreprise | [VSC](https://code.visualstudio.com/) | Je pense que ceci est un excellent éditeur de texte. Par contre, la compatibilité avec Java est moins bonne qu’avec IntelliJ, donc nous n’utiliserons pas cet IDE. |

## Autres technologies

Y a-t-il d’autres catégories de technologies qui sont nécessaires ou utiles pour répondre au besoin?  
Par exemple, des logiciels ou applications existantes qui couvrent déjà la majorité des fonctionnalités répondant au besoin.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nom de la technologie | Description sommaire | Avantages | Limitations ou incertitudes | Références et exemples | Analyse (votre jugement) |
| AUCUN | AUCUN | AUCUN | AUCUN | AUCUN | AUCUN |

# Choix et justification des technologies

Indiquez ci-dessous quelles technologies vous avez décidé d’utiliser pour réaliser le projet décrit en haut.  
*Note : Il est possible que vous changiez d’avis lors de la mise à l’essai, et c’est tout à fait correct; Les choix fait ici sont seulement provisoires.*

Minecraft Launcher (Java) :

Il s'agit du programme exécutable qui permet aux joueurs de lancer la version Java de Minecraft, un populaire jeu de construction de mondes. Le lanceur donne accès aux différentes versions du jeu, permet de gérer les profils de joueur et de modder le jeu à l'aide de mods Java.

Java 17 :

Java 17 est une version majeure du langage de programmation Java, sortie en 2021. C'est une version LTS (Long-Term Support) qui signifie qu'elle bénéficie d'un support à long terme de la part de son éditeur, Oracle. Java 17 introduit plusieurs nouvelles fonctionnalités, améliorations et correctifs par rapport aux versions précédentes.

Forge :

Forge est un outil de modding pour Minecraft qui permet aux développeurs de créer et d'installer des mods pour le jeu. Il fournit un environnement unifié pour le développement de mods, facilitant la compatibilité entre les différents mods et le jeu de base. Les joueurs peuvent utiliser Forge pour installer et gérer facilement ces mods.

IntelliJ :

IntelliJ IDEA est un environnement de développement intégré (IDE) conçu pour le développement Java. Créé par JetBrains, cet IDE est populaire pour ses puissantes fonctionnalités, son interface utilisateur intuitive et sa prise en charge d'un large éventail de langages de programmation en dehors de Java, tels que Kotlin, Groovy et Scala.

# Proposition de prototype

Afin de tester si ces technologies seront adaptées au développement du projet décrit en haut, proposez un petit sous-ensemble du projet qui permettrait de les mettre à l’essai. Vous pouvez recommander de faire plus d’un prototype indépendant s’il y a plusieurs nouvelles technologies très différentes à intégrer.

*Pour vous aider, demandez-vous quels sont les aspects clés à expérimenter pour estimer si le choix de la technologie est adéquat. Le prototype ne devrait prendre que quelques semaines à développer tout au plus (ou moins, si certaines technologies sont déjà maitrisées).*

Pour le prototype simplifié de ce projet, nous nous concentrerons sur la création de blocs de plutonium et d'uranium, la fabrication de bombes nucléaires, et la mise en œuvre d'une explosion nucléaire avec un effet radioactif de type poison. Les blocs de plutonium et d'uranium seront introduits dans le biome radioactif en tant que ressources clés. Les joueurs pourront collecter ces matériaux pour ensuite les utiliser dans la fabrication de bombes nucléaires selon une recette spécifique. Une fois la bombe fabriquée, les joueurs pourront la déployer dans le jeu, provoquant une explosion dévastatrice qui détruira les blocs environnants et créera un cratère. Après l'explosion, la zone environnante sera contaminée par la radioactivité, infligeant un effet de poison progressif aux joueurs qui s'y aventureront, drainant leur santé tant qu'ils restent dans la zone contaminée. Pour éviter de perdre de la vie sans s’en rendre compte, il sera possible pour le joueur de créer un compteur Geiger qui indique la radiation émise dans le bloc présent. Ce prototype simplifié démontre les éléments clés du concept du projet, en mettant en évidence les mécanismes de collecte, de fabrication et les conséquences de l'utilisation d'armes nucléaires dans le jeu. Les technologies utilisées demeurent Minecraft Launcher (Java), Java 17, Forge et IntelliJ IDEA pour le développement du mod.

*RAPPEL : Si ce n’est pas déjà fait, venez discuter avec moi de votre proposition de prototype pour valider qu’il est suffisant et réaliste (et donc permettra d’évaluer votre travail de manière juste).*