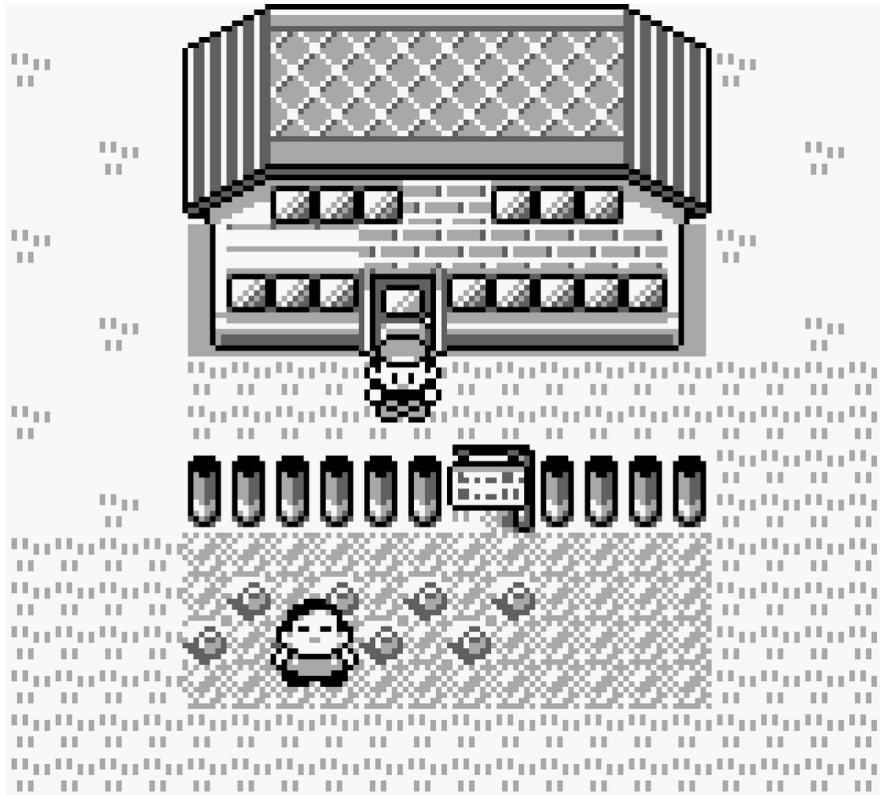


Po-C-mon



Pokémon est un jeu vidéo emblématique où les joueurs capturent et entraînent des créatures fantastiques appelées Pokémon pour devenir un Maître Pokémon et explorer un monde rempli d'aventures.

Les combats stratégiques et la collecte de Pokémon sont au cœur de l'expérience, offrant une aventure passionnante dans un univers riche en mystères.

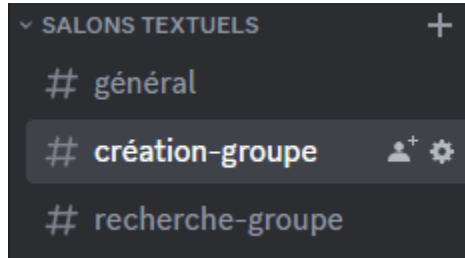
Cette nuit, nous allons implémenter une version simplifiée de ce magnifique jeu. 😊

Création des groupes

Vous pouvez choisir les personnes avec qui travailler, la seule contrainte est que vous devez être un groupe de 3 ou 4 personnes.

Rendez-vous sûr : <https://discord.gg/xSAfn8yBpY>

Vous y trouverez un salon nommé “création-groupe”.



Dans ce salon, vous envoyez un message formaté de cette manière :

Nom 1 Prénom 1 / Nom 2 Prénom 2 / Nom 3 Prénom 3 / Nom 4 Prénom 4

Une fois votre message envoyé, nous vous affecterons un rôle avec votre numéro de groupe.

Nous vous informerons via ce rôle pour le déroulement des soutenances demain matin



Configuration de l'environnement

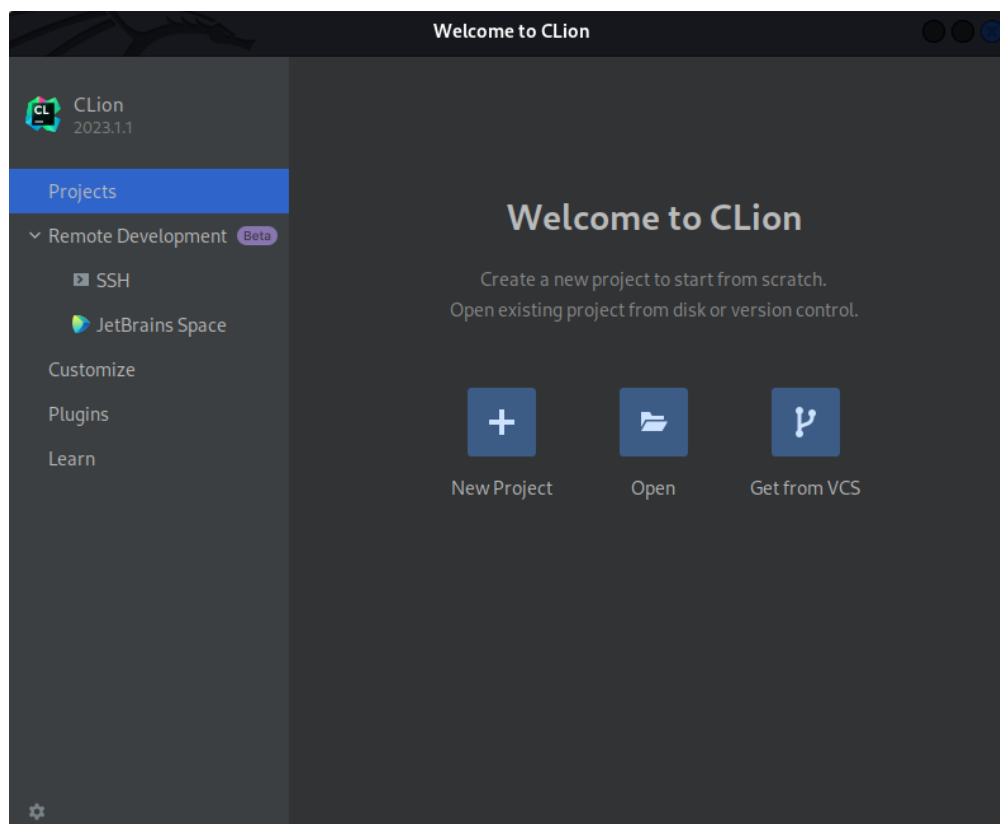
Installation de CLion

Rendez vous sûr : <https://www.jetbrains.com/fr-fr/clion/>

CLion est un IDE JetBrains, vous avez une licence disponible pour tous les IDE de JetBrains grâce à votre email "...@myges.fr".

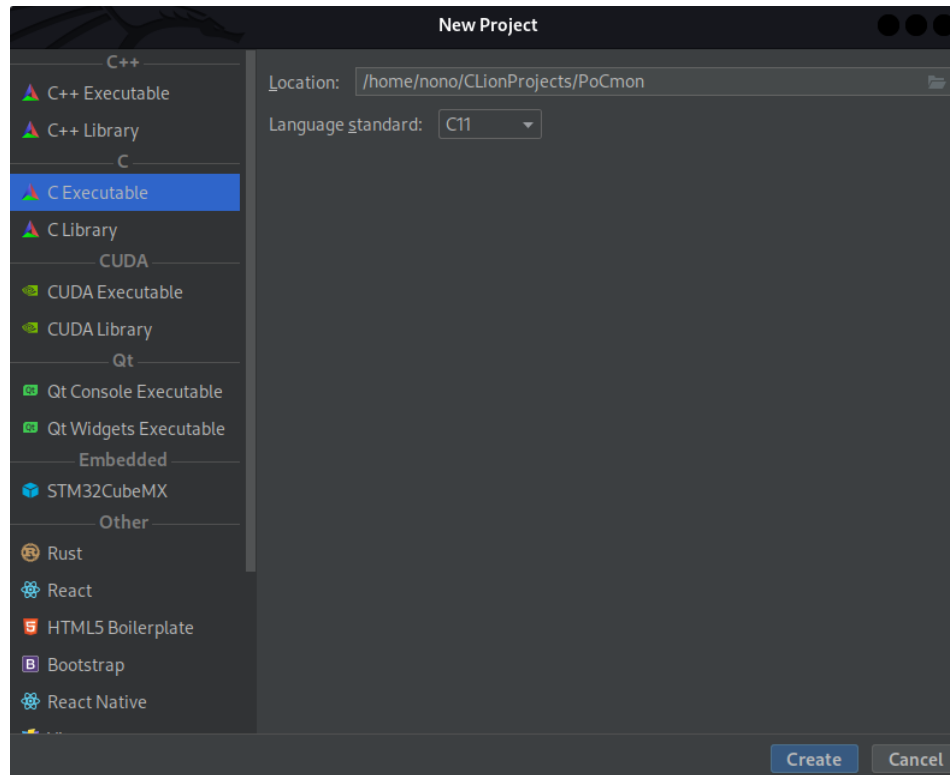
(ou utilisez l'essai de 30 jours, ça suffira pour cette nuit 😊)

Une fois CLion installé, vous pouvez le lancer et vous arriverez sur cette fenêtre:



D'ici vous pouvez créer un nouveau projet, pensez bien à:

- Sélectionner "C Executable" sur la gauche
- Vous mettre en C11
- Et donner un nom à votre projet



Les Fonctionnalités

Génération de la carte

La première étape va être de générer une carte aléatoire de **25 colonnes et 30 lignes**.

Cette carte doit être composée de 3 types de cases différentes:

- **Chemin** -> Le joueur a le droit de se déplacer
- **Hautes herbes** -> Le joueur peut se déplacer dessus et y rencontrer des Pokémon sauvages
- **Obstacle** -> Le joueur n'a pas le droit de se déplacer dessus

Il vous est demandé de créer cette carte de manière **aléatoire**.

N'oubliez pas il faudra une case sur laquelle l'utilisateur commence la partie, et qu'il ne soit pas bloqué entre 4 obstacles ! 😊

Bonus : Pour réaliser cette génération de carte, vous pouvez mettre en place un **Bruit de Perlin** pour les hautes herbes et les obstacles, et un **Perlin Worms** pour les chemins. (vous pouvez agrandir la carte si vous avez besoin 😊)

Pour vous aider : https://fr.wikipedia.org/wiki/Bruit_de_Perlin
[github Perlin en C](#)

<https://stackoverflow.com/questions/70477601/perlin-worms-for-2d-cave-generation>

Liste des pokémon disponible

Pour obtenir la liste des Pokémon disponible dans le jeu, vous devez lire un fichier **CSV**. Les données contenues dans le fichier sont sous la forme:

nom_du_pokemon;pv;attaque;défense;vitesse;type

Le fichier est téléchargeable via l'url :

<https://github.com/arnaud111/PoCmon/blob/master/poCmon.csv>

L'équipe

Lors du lancement du jeu, vous devez laisser le choix à l'utilisateur entre 3 pokémon, le pokémon sélectionné sera ajouté à son équipe.

Le joueur aura le droit d'avoir jusqu'à 6 pokémons dans son équipe.

Combat

Maintenant qu'on a une carte et une équipe de pokémons, c'est l'heure du dudududuel !

Lorsque le joueur marche dans les Hautes herbes, il a une petite probabilité de croiser un pokémon sauvage (10%)

Lorsqu'un combat se lance, par défaut on va prendre le 1er pokémon que le joueur a dans son équipe.

Entre notre pokémon et le pokémon sauvage, c'est celui avec le plus de vitesse qui attaque en premier.

Tour du joueur

Lors du tour du joueur, on lui laisse un choix entre 4 actions :

1. Attaquer le Pokémon adverse

Pour calculer les pv perdus voici la formule:

A = Pokémon attaquant

B = Pokémon attaqué

$$pv_perdus = (A.attaque - B.défense) * bonus_types$$

https://upload.wikimedia.org/wikipedia/commons/thumb/c/c5/Pok%C3%A9mon_Table_des_types_2G.svg/1200px-Pok%C3%A9mon_Table_des_types_2G.svg.png

2. Changer de Pokémon

Le joueur a le droit de changer de Pokémon en plein combat.

Pour ce faire vous devez lui afficher la liste des Pokémon qu'il a dans son équipe et le laisser choisir le Pokémon à mettre dans le combat.

3. Sac

Dans le sac, l'utilisateur a 2 choix :

- Pokéball :

Lorsqu'il lance une Pokéball, il a 1 chance sur 3 de capturer le Pokémon adverse.

S'il réussit et qu'il a moins de 6 Pokémon dans son équipe, le Pokémon y est ajouté. En revanche s'il a déjà 6 Pokémon, vous devrez lui afficher un menu lui demandant de remplacer un de ses Pokémon actuel.

- Soins :

Lorsqu'il utilise un soin, son Pokémon en combat est soigné de 50% de ses PV max.

4. Fuir

Si le joueur décide de s'enfuir il a 1 chance sur 2 de sortir du combat.

Tour de l'adversaire

Pour le tour de l'adversaire on va faire très simple, il attaque forcément votre Pokémon, avec la formule décrite au-dessus.

Évidemment, si un Pokémon du joueur meurt, il faudra lui afficher le menu avec la liste des Pokémon de son équipe afin qu'il puisse changer 😊

Dans le cas où tous les Pokémon du joueur sont morts, alors il a perdu le combat

Pokédex

Lorsque l'utilisateur est sur la carte, il devra avoir une touche lui permettant d'ouvrir le Pokédex.

Dans ce pokédex, il doit voir la liste des pokémon qu'il a déjà croisée lors de son aventure.

Et un conteur du nombre de pokémon qu'il lui reste à découvrir.

Sauvegarde

Que serait Pokémon si on ne pouvait même pas sauvegarder son avancement ?

Promis ça ne sera pas très compliqué 😊

Lorsque l'utilisateur est sur la carte principale, vous devez lui laisser une touche lui permettant de sauvegarder sa partie.

Pour sauvegarder la partie vous devez créer un fichier au format de votre choix, qui stockera la liste des Pokémon présents dans l'équipe du joueur, ainsi que l'avancement du Pokédex.

Lors du lancement du jeu vous devrez essayer de lire le fichier de sauvegarde et récupérer les données stockées afin de reprendre la partie actuelle.

Bonus : SQLite 🐼

Maintenant qu'on a réussi tout ça, on va améliorer le système de sauvegarde. 😊

On est sympa on vous fournit un exemple pour l'initialisation de la database et de la bibliothèque: <https://github.com/arnaud111/PoCmon>

Dans un premier temps, vous allez prendre le fichier CSV contenant la liste des Pokémon, et stocker les données dans la table.

Ensuite, au lieu de faire la sauvegarde dans un fichier, vous allez devoir créer des tables afin de stocker l'équipe de pokémon ainsi que l'avancement dans le Pokédex.

Lors du chargement du jeu, vous devez récupérer toutes les données depuis la database.

De plus, vous allez mettre en place un système de sauvegarde automatique. À chaque modification de l'équipe ou du Pokédex, vous devez mettre à jour la database.

Bonus : Réseau 🥰

Déjà si vous êtes arrivé jusque-là, franchement, bravo. 🙌
Vous mangez des cartes graphiques.

Mais ce serait dommage qu'on vous laisse aller dormir, donc vous avez le droit d'aller chercher un café et de revenir travailler, on va faire un peu de réseau maintenant. 😊

On va pas vous demander de refaire tout le jeu en réseau, on n'est pas des monstres après tout. 😊

Vous devrez simplement implémenter la partie combat 1v1 en réseau. En utilisant la bibliothèque: `<sys/socket.h>`

Pas de système d'équipe, pas de sauvegarde, pas de Pokéball, pas de fuite.

Un joueur lance le jeu en mode "serveur", l'autre le lance en mode "client".
Le "serveur" gèrera l'état de la partie et le "client" se contentera d'afficher l'état du jeu et d'envoyer les actions faite par le joueur.

Les 2 joueurs sélectionnent un pokémon dans la liste et le combat se lance, ils peuvent seulement attaquer l'autre pokémon ou se soigner.

Pour vous aider : https://www.tala-informatique.fr/wiki/index.php/C_socket

Vous trouverez ci-dessous un schéma explicatif de la communication réseau. 😊

