

SURVIVE IF YOU CAN

Projet Hackathon Junior

Thème : La Purge



MONTÉE EN COMPÉTENCE

LA PROGRAMMATION ORIENTÉE OBJET (POO)

Transition vers une architecture structurée.

- Compréhension des **Classes** et **Objets**.
- Utilisation de l'encapsulation pour protéger les données du jeu.
- Organisation modulaire pour le travail d'équipe.

THÉORIE DES GRAPHES

Application mathématique concrète.

- Modélisation de la ville : Lieux (Nœuds) et Routes (Arêtes).
- Compréhension des graphes pondérés (temps de trajet).
- Complexité algorithmique pour le cheminement.

LE CONTEXTE

OLE THÈME : LA PURGE

Dans un futur dystopique, tout crime est légal pendant 12 heures. L'atmosphère est oppressive, sombre et dangereuse.

Cette contrainte temporelle impose une tension constante au joueur.

OL'OBJECTIF

Survivre à tout prix.

Le joueur doit naviguer intelligemment entre les zones de danger, gérer ses ressources (armes, soins) et atteindre des refuges avant la fin du temps imparti.

ARCHITECTURE TECHNIQUE



BACKEND PHP

Choix stratégique d'utiliser PHP pour sa robustesse en POO, permettant une gestion claire des entités du jeu (Joueur, Inventaire, Carte).



STRUCTURE MVC

Séparation logique entre les Données (Modèle), l'Affichage (Vue) et la Logique (Contrôleur) pour un code maintenable.



FRONTEND LÉGER

HTML5 et CSS3 pour une interface réactive, minimisant le JS pour se concentrer sur la logique serveur complexe.

LE MARCHÉ NOIR

ÉCONOMIE & ALGORITHMES

- **Système de Panier** : Gestion de session pour acheter armes et soins avec une monnaie virtuelle.
- **Défi Technique** : Implémentation d'un algorithme de tri personnalisé (QuickSort) pour classer les objets.
- **Critères de Tri** : Prix (croissant/décroissant) et Puissance de feu.



LA VILLE (GRAPHE)

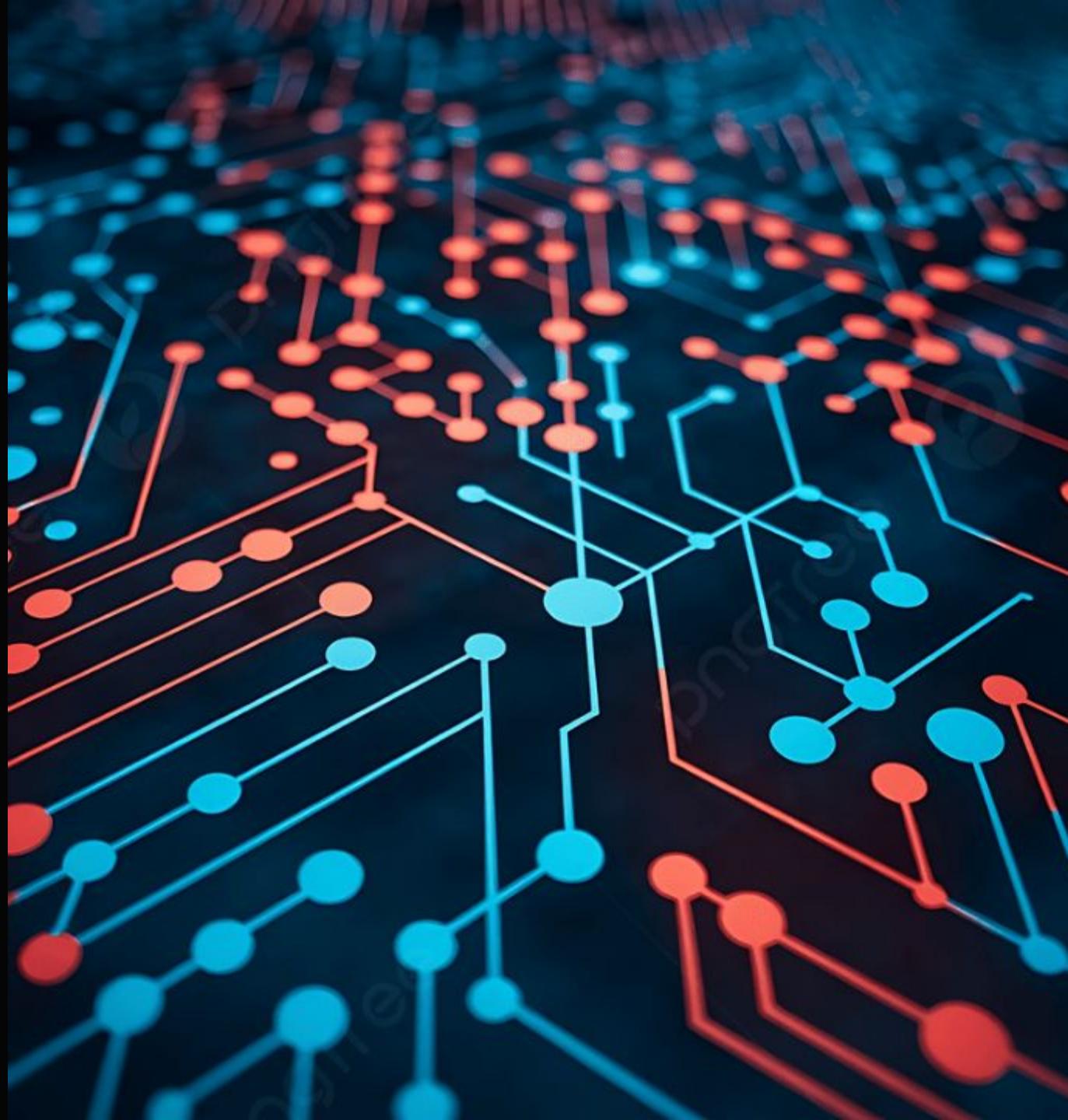
CARTOGRAPHIE COMPLEXE

Le cœur du projet repose sur un réseau de plus de 20 lieux interconnectés.

Nœuds : Quartiers, Refuges, Zones de Loot.

Arêtes : Routes pondérées par la distance.

Transport : Distinction entre itinéraires Voiture (plus lent) et Train (rapide mais limité).



NAVIGATION : DIJKSTRA



1. INITIALISATION

Définir le point de départ et les distances infinies pour les autres noeuds.



2. EXPLORATION

Visiter les voisins, calculer les poids (Distance / Vitesse).



3. OPTIMISATION

Sélectionner le chemin le plus court (coût minimal) vers la destination.



4. CONTRAINTE

Forcer le passage par un 3ème lieu intermédiaire (étape obligatoire).

SYSTÈME DE DANGER

MENACES ALÉATOIRES

Pour simuler l'imprévisibilité de la Purge, un système d'événements aléatoires (RNG) a été mis en place.

Types d'événements :

- ☢ Zones Irradiées
- 💀 Attaques de Gangs
- 🌧 Inondations Subites



DES QUESTIONS ?

Merci de votre attention.

 github.com/bGlmZW/hackatonjr-survive-if-you-can

IMAGE SOURCES



<https://i.redd.it/u0a340lghm2e1.gif>

Source: www.reddit.com



https://png.pngtree.com/background/20250201/original/pngtree-abstract-digital-network-with-glowing-nodes-and-interconnected-circles-picture-image_16137908.jpg

Source: pngtree.com



https://img.freepik.com/premium-vector/hud-red-warning-frames-futuristic-screen-borders_8071-59643.jpg

Source: www.freepik.com



<https://images.unsplash.com/photo-1707999464758-0ad86b14290f?fm=jpg&q=60&w=3000&ixlib=rb-4.1.0&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D>

3D

Source: unsplash.com