
**Escape Game numérique
Laboratoire TIMC**



Sommaire

I.	<i>Présentation du jeu</i>	3
II.	<i>Les énigmes</i>	4
II.1	Le Binaire.....	4
II.2	Les bases de données.....	4
II.3	Les portes logiques	4
II.4	Architecture d'un ordinateur	5
II.5	Algorithmie.....	5
II.6	Réseau	5

I. Présentation du jeu

Intitulé « *Escape Game 2053 – La Résistance du Savoir* », ce jeu propose une immersion dans un univers dystopique où la technologie a été éradiquée par un virus fictif nommé Curare, soutenu par un mouvement anti-numérique, les Antitek. Le joueur incarne un jeune résistant chargé de réactiver les savoirs essentiels pour tenter de restaurer un monde numérique effondré.

Ce scénario sert de fil rouge à un parcours pédagogique structuré autour de plusieurs thématiques clés en informatique : algorithmique, logique binaire, base de données, architecture des ordinateurs, porte logiques, etc. Chaque thématique donne lieu à des épreuves spécifiques, conçues pour être à la fois accessibles, ludiques et pédagogiques. En limitant donc les contraintes matérielles, cette version numérique est adaptée aux usages actuels des jeunes générations.

Le joueur est plongé dans un univers postapocalyptique dès l'écran d'accueil, où un narrateur introduit le contexte de la mission. Après avoir choisi un avatar parmi une sélection de personnages (fictifs), il est invité à explorer une pièce interactive représentant une pièce cachée du laboratoire TIMC.

Le parcours repose sur six énigmes principales :

1. Le binaire – Conversion de nombres décimaux en binaire
2. L'algorithmie – Simulation d'un programme pas à pas.
3. Le routage – Simulation d'un acheminement de données via des routeurs
4. Les Portes logiques – Identification de combinaisons logiques correctes
5. Les Bases de données – Analyse de tables et reconstitution chronologique
6. L'architecture matérielle – Reconnaissance des composants d'un ordinateur

Chaque épreuve est accompagnée d'une mise en situation immersive (texte ou voix IA), d'un système de correction immédiate et d'un suivi de progression. Un temps global de 35 minutes est imposé pour simuler l'urgence de la mission et de reproduire le principe d'un Escape Game.

Lorsque toutes les énigmes sont validées, le joueur déclenche une cinématique de fin, qui ouvre la porte finale et lance une scène symbolique de félicitations, renforçant l'axe narratif et pédagogique du jeu.

II. Les énigmes

II.1 *Le Binaire*

Dans cette énigme, le joueur est invité à convertir plusieurs nombres décimaux en binaire à l'aide d'un tableau de puissance de 2, allant de 1024 à 1.

Le principe est simple : pour chaque case correspondant à une puissance de 2, il faut indiquer 1 si cette valeur est présente dans la décomposition du nombre, sinon 0.

Chaque clic dans une case permet de basculer entre 0 et 1. Une fois la suite complète, il suffit de cliquer sur « Valider votre réponse ! » pour obtenir un retour immédiat.

Cette épreuve permet de comprendre le fonctionnement du système binaire, fondement du codage informatique. Elle développe la logique, la manipulation des puissances de 2, et la décomposition additive d'un entier.

II.2 *Les bases de données*

Cette séquence est composée de trois sous-épreuves successives, toutes basées sur l'analyse et la compréhension de données structurées, à la manière d'une base de données relationnelle. L'objectif est de faire découvrir, de façon progressive, les notions de recherche, tri et validation d'informations dans une table.

Premièrement le joueur accède à un registre ouvert sous forme de livre, affichant plusieurs lignes contenant des identifiants, rôles, dates de disparition et âges. À l'aide d'un énoncé, il doit retrouver les bonnes correspondances dans la base en répondant à plusieurs questions.

Chaque question cible un élément précis (nom, rôle, date, etc.), et le joueur doit trouver le bon champ correspondant à la question.

Une nouvelle consigne s'affiche : le joueur doit trier l'ensemble des lignes du registre dans l'ordre de disparition, du moins récent au plus récent.

Le joueur interagit avec des "slots" pour replacer les lignes selon la bonne chronologie. Une validation permet de corriger les erreurs en temps réel.

Dernièrement, Le joueur accède à un terminal informatique contenant une table plus vaste. Certaines lignes sont volontairement erronées :

- valeurs absentes ou incohérentes
- doublons
- champs absurdes ou contradictoires au niveau des dates

Il doit cliquer sur les lignes incorrectes pour les signaler et obtenir un retour. Le nombre d'erreurs restantes est affiché, comme un compteur.

II.3 *Les portes logiques*

Cette épreuve se divise en deux parties.

Dans la première, le joueur observe les entrées/sorties de plusieurs portes logiques et doit associer la bonne porte (AND, OR, NAND, $=1$, ≥ 1 , etc.) à chaque position A, B, C, D à l'aide d'une table de vérité affichée à droite. Un code final à 4 chiffres est généré selon l'ordre des portes (ABCD).

Dans la seconde partie, les portes sont déjà placées, mais les valeurs d'entrée sont erronées. Le joueur doit tester différentes combinaisons de 0 et 1 pour obtenir les bonnes sorties, avec un feedback visuel via des ampoules qui s'allument en cas de réussite. L'objectif est de découvrir les bases de la logique binaire, par l'expérimentation et l'observation.

II.4 Architecture d'un ordinateur

Dans cette épreuve, le joueur est invité à explorer visuellement les composants internes d'un ordinateur (CPU, GPU, RAM, disque dur, ports, ventilateur, etc.). En cliquant sur chacun, il découvre leur fonction expliquée sous forme de métaphore.

Une fois tous les composants visualisés, un QCM de 6 questions lui est proposé pour associer chaque composant à sa métaphore. Cette épreuve permet une première familiarisation avec les composants physiques d'un PC, en développant une compréhension globale de leur rôle dans le système.

II.5 Algorithmie

Cette épreuve initie le joueur à la lecture et à l'exécution d'un programme simple. Dans la première partie, le joueur **déplace lui-même son personnage (X)** dans une grille, en **drag and drop**, en suivant ligne par ligne un programme affiché à droite. Il applique les instructions (déplacement, test, ajout de variable) pour atteindre la case F.

Dans la seconde partie, c'est un **labyrinthe** : Le joueur doit **simuler mentalement** l'exécution pour déterminer si l'algorithme mène à la sortie. L'ensemble permet de découvrir les bases de l'algorithmie (séquence, boucle, condition) dans un cadre visuel et interactif.

II.6 Réseau

Le joueur doit acheminer un paquet de données du serveur 1 vers le serveur 2, puis du serveur 2 vers le serveur 3, en glissant le drapeau de nœud en nœud à travers les routeurs. Chaque routeur possède une table de routage accessible par clic, indiquant quelle sortie utiliser selon l'adresse IP. Le paquet n'avance que si la sortie est correcte ; sinon, il revient en arrière.

Chaque saut ajoute un poids, et la somme totale des deux trajets fournit un code à 4 chiffres.

L'épreuve introduit les notions de routage, d'adressage IP et de choix d'un chemin réseau optimal.

L'épreuve se termine par la saisie d'un code correspondant à la somme des poids des arcs parcourus entre le serveur 1 et le serveur 2, puis entre le serveur 2 et le serveur 3.