

Problématique

Les jeux éducatifs peinent souvent à maintenir l'engagement des joueurs, privilégiant le pédagogique au détriment du plaisir. Comment intégrer harmonieusement un QCM de culture générale dans un jeu 2D dynamique, pour rendre l'apprentissage ludique et motivant ?

Objectifs

- Développer un jeu de plateforme 2D de type Mario avec Godot 4.4 en GDScript
- Intégrer un QCM conditionnant la progression case par case (117 questions JSON)
- Implémenter des pièges (recul) et bonus (avance) avec détection par coordonnées
- Concevoir une architecture modulaire et extensible (multijoueur, audio, UI)

Concept de jeu

- Boucle principale : question → réponse → progression → piège/bonus → nouvelle question
- Pièges (cercles rouges) : font reculer ; Bonus (cercles verts) : font avancer
- Alternance gauche/droite par étage, détection de position à 25 px de tolérance
- Objectif : atteindre la fin du niveau en optimisant réponses et zones de risque

Architecture technique

- Scenes/Levels/ : logique de progression, pièges/bonus, orchestration du flow de partie
- Scenes/qcm/ : interface et logique QCM (validation, chaînement, variante multijoueur)
- Scenes/ui/ : lobby, classement, popups, HUD ; Scenes/audio/ : gestionnaire son centralisé
- EventBus centralisé (signaux Godot) pour la communication entre tous les composants
- Séparation stricte Monde/UI : évite les décalages d'alignement et problèmes de clic

Fonctionnalités implémentées

- Système QCM opérationnel avec base de questions JSON éditable sans toucher au code
- Progression joueur liée aux réponses : 3 pièges et 4 bonus positionnés sur la carte
- Fondations multijoueur présentes (lobby, classement, qcm_multiplayer.gd)
- Affichage plein écran optimisé : viewport/stretch ajustés, zéro bande noire

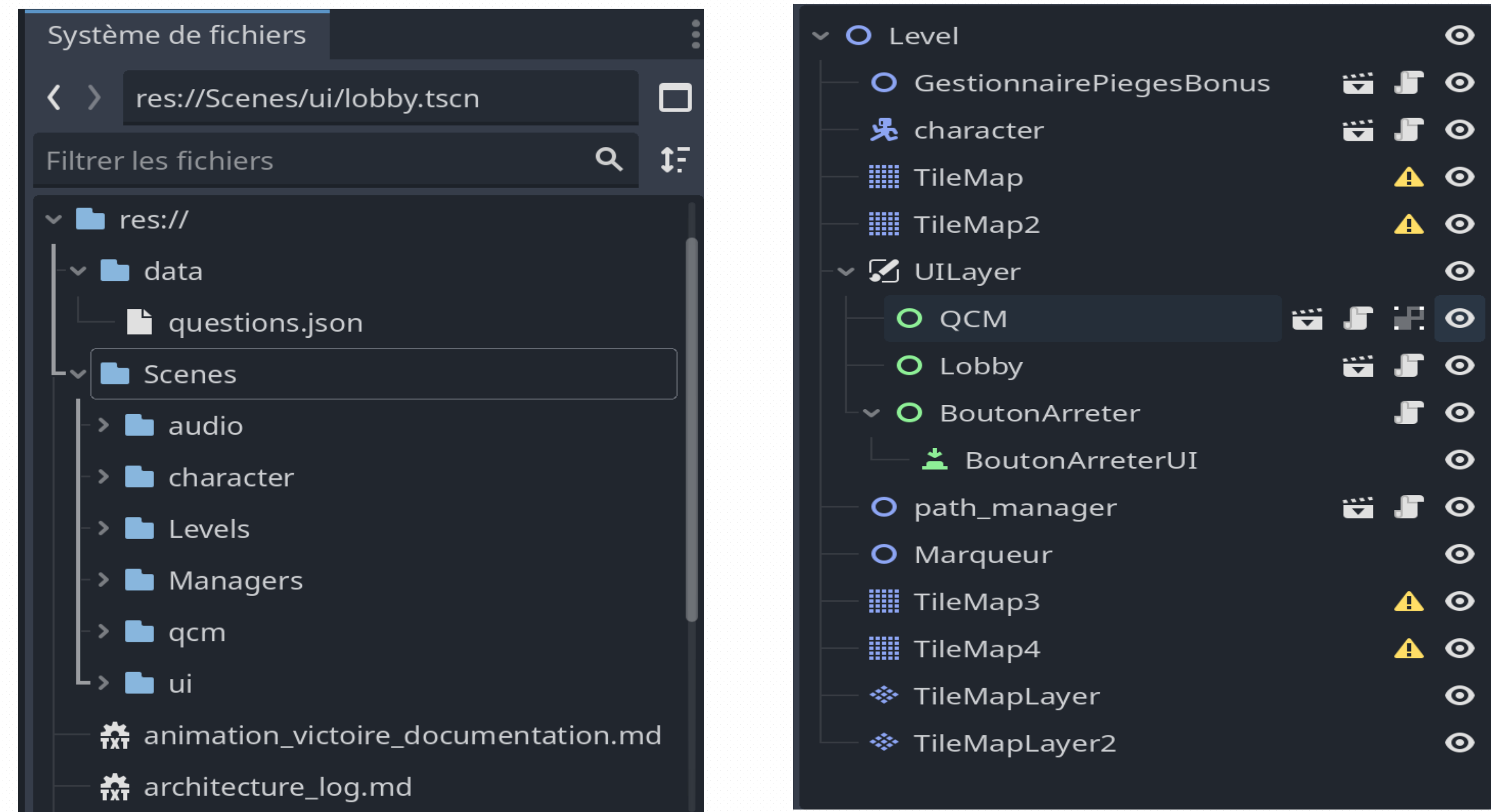


Figure 2 : Architecture modulaire du projet Godot 4.4

État actuel du projet

- Jeu éducatif 2D jouable : boucle QCM → progression → pièges/bonus complètement fonctionnelle
- Affichage revu : meilleure utilisation de l'axe X, disparition des bandes noires gênantes
- UX corrigée : blocages de clic résolus (superposition Control nodes / bouton d'arrêt)
- Stabilité du flow : chemins de nœuds mis à jour après réorganisation des scènes Godot
- Base techniquement crédible pour une démo/pitch ; contenu facilement extensible

Pistes d'évolution

- Gameplay : combos, timers, multiplicateurs de score, difficulté adaptative
- Contenu : catégories thématiques, niveaux progressifs, questions liées aux cours UQAR
- Multijoueur : synchronisation fine, salons privés, rematch, anti-latence réseau
- UI/UX : transitions animées, onboarding, feedback de réponse plus riche
- Publication : build desktop/web, écran options, persistance de score et profil joueur

Bibliographie

- Documentation officielle Godot Engine 4.4 — docs.godotengine.org
- GDScript Reference, Signaux et EventBus pattern — docs.godotengine.org/fr
- Dépôt GitHub — github.com/eliDaniel007/jeu-qcm-2d-godot

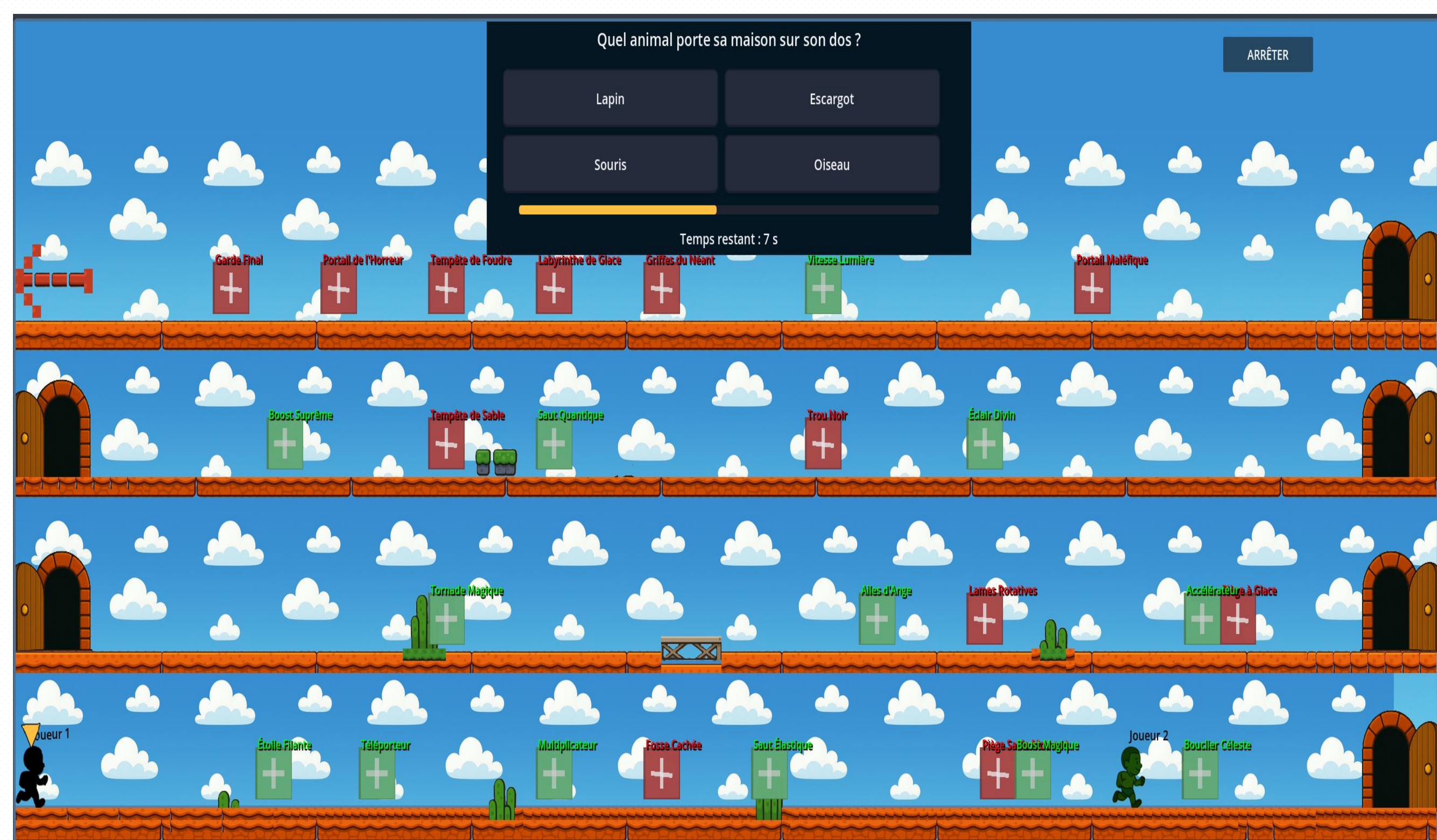


Figure 1 : Carte de jeu (pièges rouges et bonus verts)