

Julie CAIVEAU
Maïna KAUS SOUDAIN
Adrien DIAZ

Classe Tle NSI

Projet jeu-vidéo: Pyxel Duck

Sommaire



1

Introduction au jeu

2

Gameplay, actions et IHM

3

Schéma et libraires

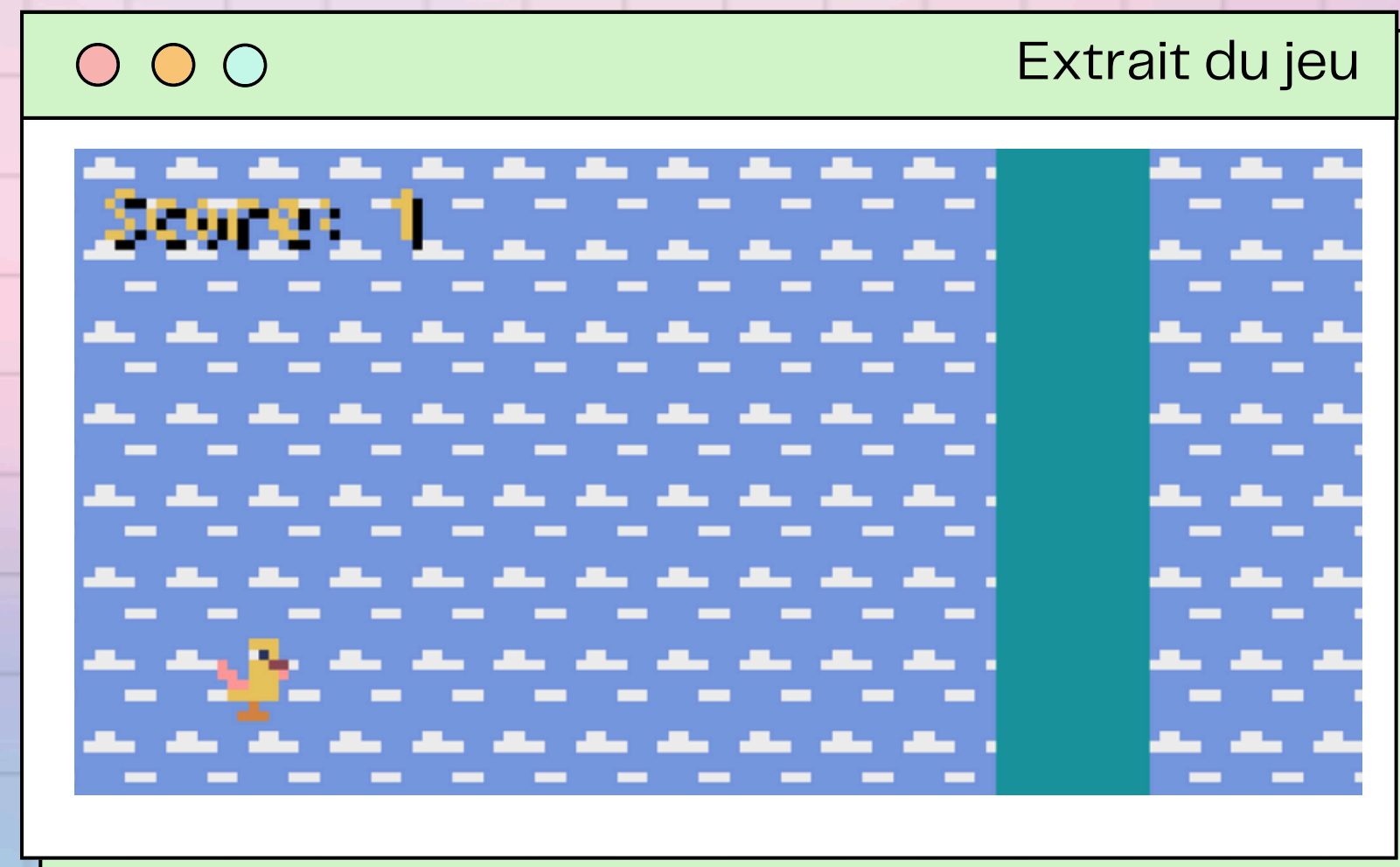
4

Structure du code

1

Introduction au jeu

Après avoir passé un temps dans les pays chauds, un jeune canard se retrouve seul, loin des siens. Déterminé à retrouver sa famille, il entreprend un voyage semé d'embûches, où chaque battement d'ailes le rapproche de son but. Mais le chemin est dangereux, et seul un vol précis et habile lui permettra d'arriver à destination.





Actions

- Sauter : Appuyer sur ESPACE pour donner un coup d'aile à l'oiseau.
- Éviter les obstacles : Gérer la hauteur et le timing des sauts pour ne pas toucher les tuyaux.
- Accumuler des points : Passer entre les tuyaux pour augmenter le score.
- Recommencer après une défaite : Appuyer sur ESPACE pour relancer la partie.



IHM

- Fond d'écran simple avec un défilement en arrière-plan.
- Oiseau en pixel art qui change d'animation lorsqu'il saute.
- Tuyaux verts générés aléatoirement, placés en haut et en bas.
- Score affiché en haut à gauche en pixel art.
- Message "GAME OVER" affiché au centre après une défaite.
- Texte "Press space to continue" pour relancer une partie.
- Texte "Record" mis à jour à chaque nouveau meilleur score.

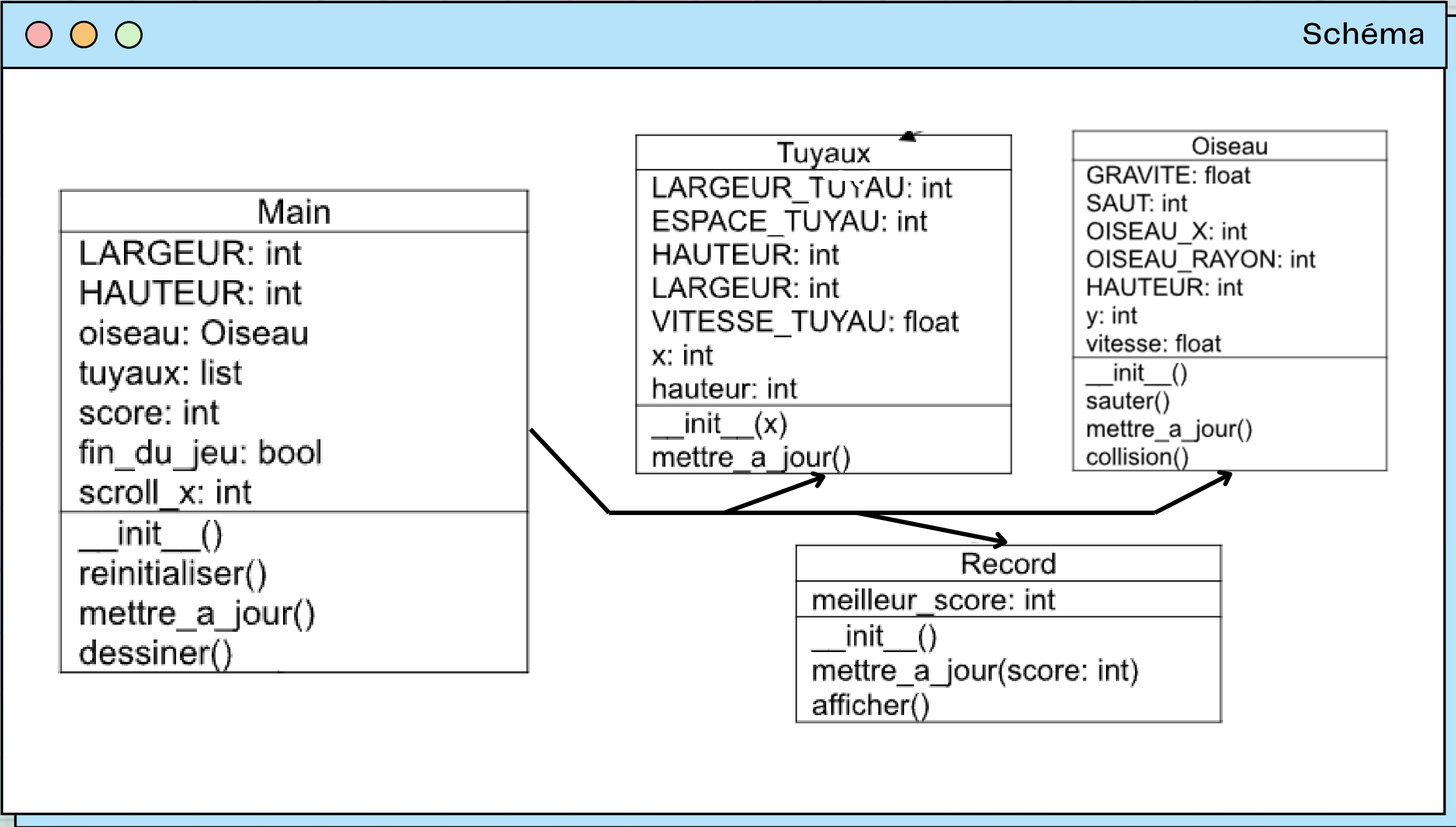


Gameplay

- L'oiseau tombe à cause de la gravité et peut sauter avec ESPACE.
- Les tuyaux défilent vers la gauche pour créer un effet de mouvement.
- Si l'oiseau touche un tuyau ou sort de l'écran, la partie est perdue.
- Le score augmente à chaque passage d'un tuyau.
- L'écran affiche le score et un message "GAME OVER" en cas de défaite.
- Redémarrage du jeu en appuyant sur ESPACE après un Game Over.
- Le meilleur score est affiché à la fin de la partie.

3

Schéma et libraires



Libraries

Librairies utilisées:

pyxel -> Bibliothèque pour créer des jeux rétros en pixel art.

random -> Génère des valeurs aléatoires (utilisé pour la hauteur des tuyaux).

4 Structure du code



Structure

Le jeu est divisé en trois classes principales :

1. Oiseau : Gère le joueur (position, mouvement, collisions).
2. Tuyau : Gère les obstacles (position, déplacement).
3. « Main » : Gère la boucle principale du jeu (initialisation, mise à jour, affichage).
4. Record : Enregistre et met à jour le meilleur score.

Relations entre les classes

- « Main » → Oiseau : Association (la classe Main crée et gère un objet Oiseau).
- « Main » → Tuyaux : Association (la classe Main gère les tuyaux).
- « Main » → Record : Association (la classe Main utilise Record pour suivre le meilleur score).
- Oiseau et Tuyaux : Indépendants (ils n'interagissent pas directement, mais Main gère leur collision).