

Soutenance

Quest for Boaty by Nicolas RUCHE & Maxime VONG

Concept du jeu

- **2 bateaux qui s'envoient des petits bateaux pour couler l'adversaire**
- Tirs de mortiers qui détruisent les petits bateaux
- Bonus : Récupère la vie ou réduit le cooldown du spawn des bateaux
- Les petits bateaux possèdent 3 niveaux :
 - Avance droit devant
 - Fonce sur l'adversaire si proche de lui
 - Tire une fois une balle par spawn
- Pour gagner : envoyez 16 bateaux (4 par niveau) de l'autre côté ou points de vie du joueur adverse à 0

Concept du jeu

- 2 bateaux qui s'envoient des petits bateaux pour couler l'adversaire
- **Tirs de mortiers qui détruisent les petits bateaux**
- Bonus : Récupère la vie ou réduit le cooldown du spawn des bateaux
- Les petits bateaux possèdent 3 niveaux :
 - Avance droit devant
 - Fonce sur l'adversaire si proche de lui
 - Tire une fois une balle par spawn
- Pour gagner : envoyez 16 bateaux (4 par niveau) de l'autre côté ou points de vie du joueur adverse à 0

Concept du jeu

- 2 bateaux qui s'envoient des petits bateaux pour couler l'adversaire
- Tirs de mortiers qui détruisent les petits bateaux
- **Bonus : Récupère la vie ou réduit le cooldown du spawn des bateaux**
- Les petits bateaux possèdent 3 niveaux :
 - Avance droit devant
 - Fonce sur l'adversaire si proche de lui
 - Tire une fois une balle par spawn
- Pour gagner : envoyez 16 bateaux (4 par niveau) de l'autre côté ou points de vie du joueur adverse à 0

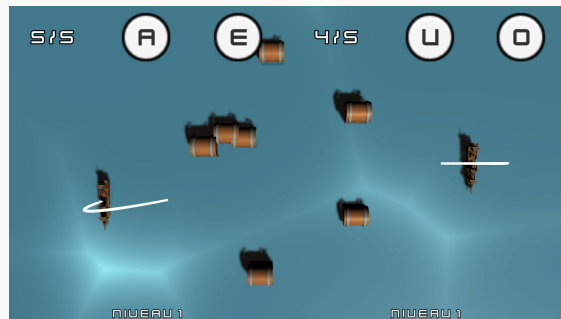
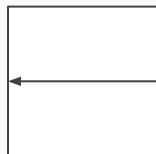
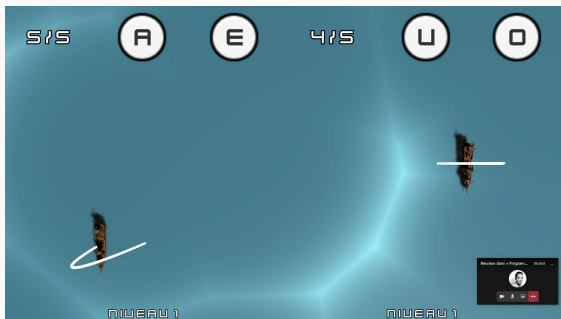
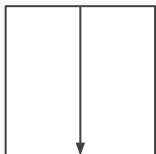
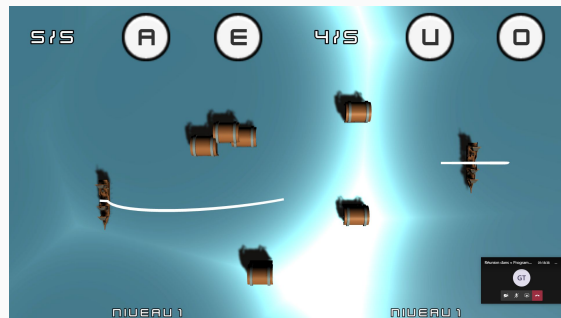
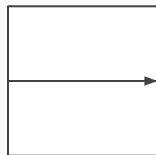
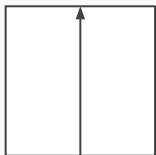
Concept du jeu

- 2 bateaux qui s'envoient des petits bateaux pour couler l'adversaire
- Tirs de mortiers qui détruisent les petits bateaux
- Bonus : Récupère la vie ou réduit le cooldown du spawn des bateaux
- **Les petits bateaux possèdent 3 niveaux :**
 - Avance droit devant
 - Fonce sur l'adversaire si proche de lui
 - Tire une fois une balle par spawn
- Pour gagner : envoyez 16 bateaux (4 par niveau) de l'autre côté ou points de vie du joueur adverse à 0

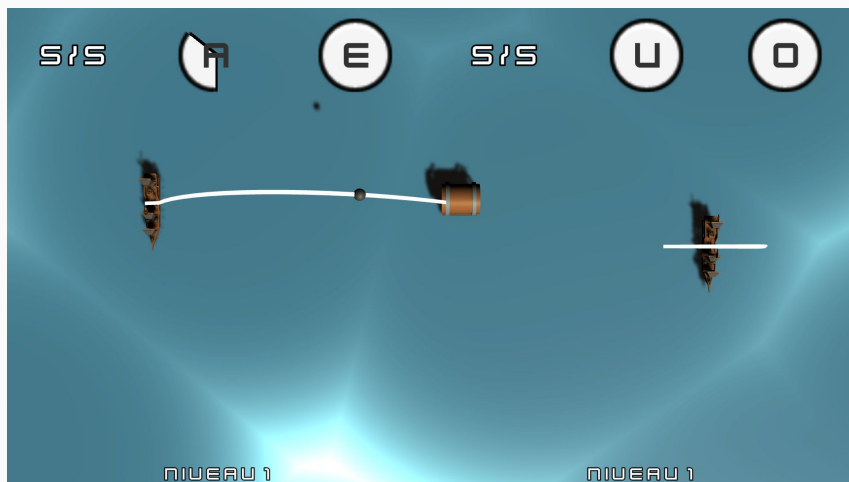
Concept du jeu

- 2 bateaux qui s'envoient des petits bateaux pour couler l'adversaire
- Tirs de mortiers qui détruisent les petits bateaux
- Bonus : Récupère la vie ou réduit le cooldown du spawn des bateaux
- Les petits bateaux possèdent 3 niveaux :
 - Avance droit devant
 - Fonce sur l'adversaire si proche de lui
 - Tire une fois une balle par spawn
- **Pour gagner : envoyez 16 bateaux (4 par niveau) de l'autre côté ou points de vie du joueur adverse à 0**

Gameplay



Gameplay



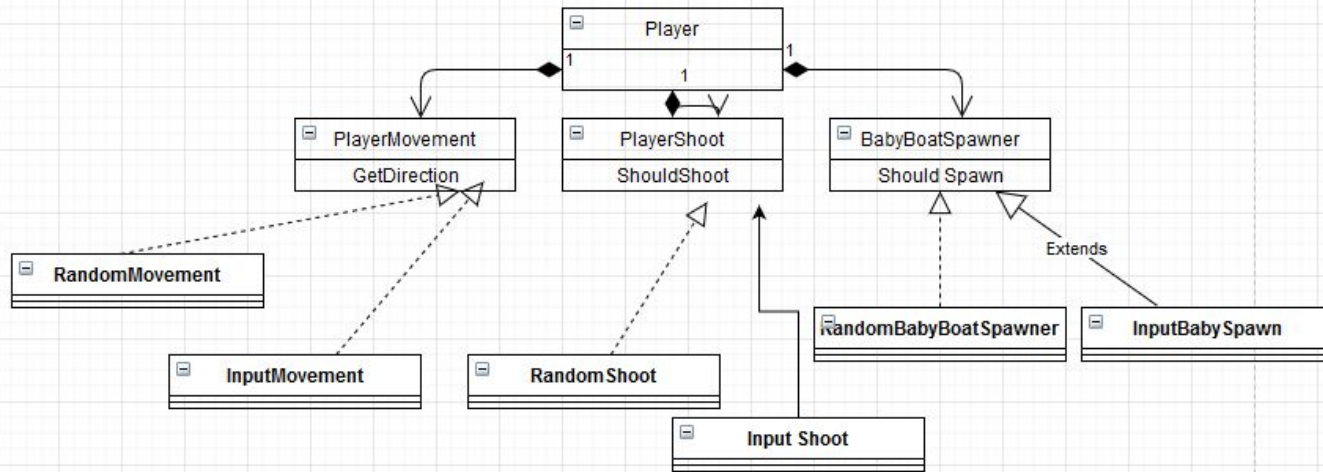
Tir



Spawn

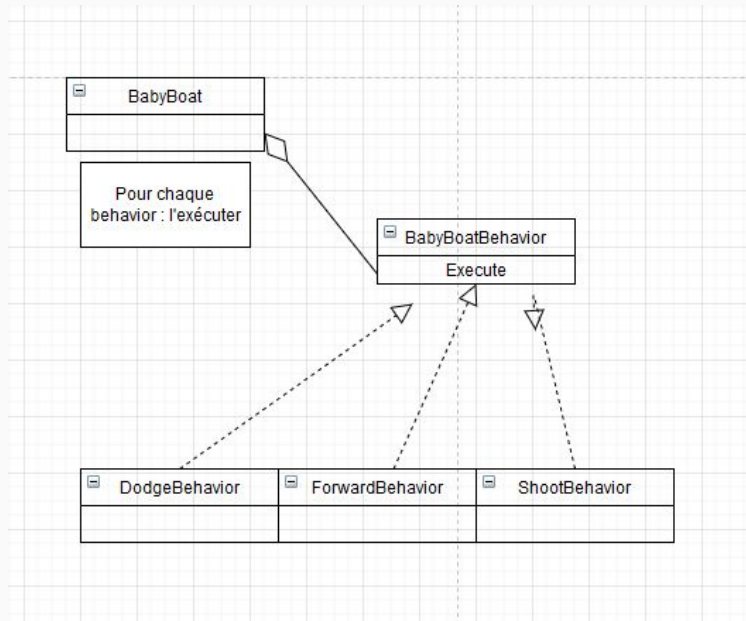
Architecture

Les agents



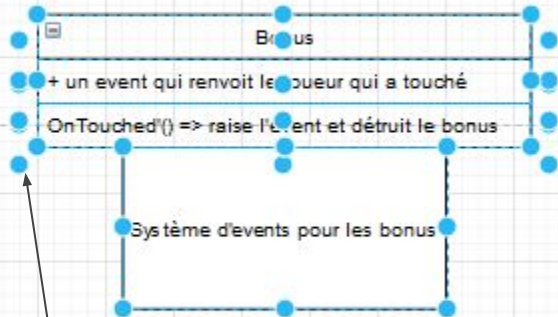
Architecture

Les boats

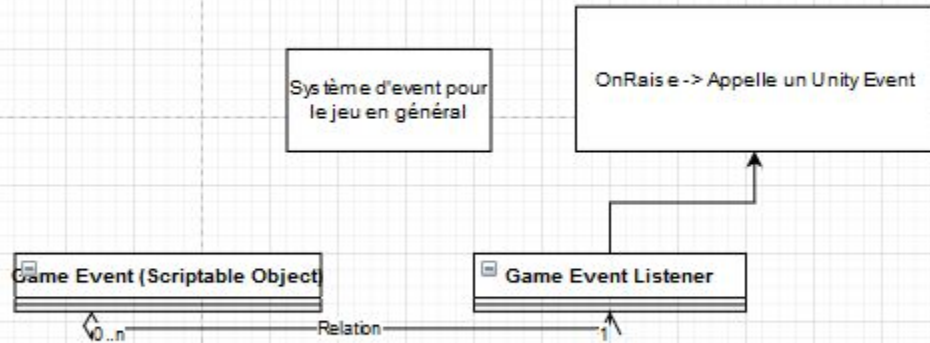


Architecture

Les événements et bonus



Ceci est une feature



Optimisations

- **Implémentation de l'object Pooling pour les balles, bateaux.**
- Scriptable object pour éviter l'appel de méthodes lourdes (Find)
- Utilisation d'événements
- Modèles low poly
- Peu de méthodes coûteuses (pas de Raycast)
- Colliders simples et matrice de collisions optimale
- Mise en cache des GetComponent le plus possible

Optimisations

- Implémentation de l'object Pooling pour les balles, bateaux.
- **Scriptable object pour éviter l'appel de méthodes lourdes (Find)**
- Utilisation d'événements
- Modèles low poly
- Peu de méthodes coûteuses (pas de Raycast)
- Colliders simples et matrice de collisions optimale
- Mise en cache des GetComponent le plus possible

Optimisations

- Implémentation de l'object Pooling pour les balles, bateaux.
- Scriptable object pour éviter l'appel de méthodes lourdes (Find)
- **Utilisation d'événements**
- Modèles low poly
- Peu de méthodes coûteuses (pas de Raycast)
- Colliders simples et matrice de collisions optimale
- Mise en cache des GetComponent le plus possible

Optimisations

- Implémentation de l'object Pooling pour les balles, bateaux.
- Scriptable object pour éviter l'appel de méthodes lourdes (Find)
- Utilisation d'événements
- **Modèles low poly**
- Peu de méthodes coûteuses (pas de Raycast)
- Colliders simples et matrice de collisions optimale
- Mise en cache des GetComponent le plus possible

Optimisations

- Implémentation de l'object Pooling pour les balles, bateaux.
- Scriptable object pour éviter l'appel de méthodes lourdes (Find)
- Utilisation d'events
- Modèles low poly
- **Peu de méthodes coûteuses (pas de Raycast)**
- Colliders simples et matrice de collisions optimale
- Mise en cache des GetComponent le plus possible

Optimisations

- Implémentation de l'object Pooling pour les balles, bateaux.
- Scriptable object pour éviter l'appel de méthodes lourdes (Find)
- Utilisation d'events
- Modèles low poly
- Peu de méthodes coûteuses (pas de Raycast)
- **Colliders simples et matrice de collisions optimale**
- Mise en cache des GetComponent le plus possible

Optimisations

- Implémentation de l'object Pooling pour les balles, bateaux.
- Scriptable object pour éviter l'appel de méthodes lourdes (Find)
- Utilisation d'events
- Modèles low poly
- Peu de méthodes coûteuses (pas de Raycast)
- Colliders simples et matrice de collisions optimale
- **Mise en cache des GetComponent le plus possible**

Ce qui est pas optimisé : Nos diapos

33 slides



Bugs connus

- **Agent random un peu bugué dans le build (bug aléatoire, pas sur Unity)**
- Possibilité de mettre en attente un tir de mortier et un envoi de petit bateau depuis la pause
- Bonus qui spawnent l'un sur les autres (Poisson Disc Sampling)

Bugs connus

- Agent random un peu bugué dans le build (bug aléatoire, pas sur Unity)
- **Possibilité de mettre en attente un tir de mortier et un envoi de petit bateau depuis la pause**
- Bonus qui spawnent l'un sur les autres (Poisson Disc Sampling)

Bugs connus

- Agent random un peu bugué dans le build (bug aléatoire, pas sur Unity)
- Possibilité de mettre en attente un tir de mortier et un envoi de petit bateau depuis la pause
- **Bonus qui spawnent l'un sur les autres (Poisson Disc Sampling)**

Ce qu'on aurait voulu faire en plus

- **Plus de bonus et pouvoir les différencier**
- Meilleur UI in game (cooldowns, level up etc.)
- Un meilleur shader pour l'eau et meilleures particules
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- Meilleur décor in game (des poissons, des mouettes etc...)
- Animation de game over (savoir qui a gagné)

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- **Meilleur UI in game (cooldowns, level up etc.)**
- Un meilleur shader pour l'eau et meilleures particules
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- Meilleur décor in game (des poissons, des mouettes etc...)
- Animation de game over (savoir qui a gagné)

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- Meilleur UI in game (cooldowns, level up etc.)
- **Un meilleur shader pour l'eau et meilleures particules**
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- Meilleur décor in game (des poissons, des mouettes etc...)
- Animation de game over (savoir qui a gagné)

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- Meilleur UI in game (cooldowns, level up etc.)
- Un meilleur shader pour l'eau et meilleures particules
- **Un système de tir de mortier plus généreux en taille au niveau du point d'impact**
- Meilleur décor in game (des poissons, des mouettes etc...)
- Animation de game over (savoir qui a gagné)

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- Meilleur UI in game (cooldowns, level up etc.)
- Un meilleur shader pour l'eau et meilleures particules
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- **Meilleur décor in game (des poissons, des mouettes etc...)**
- Animation de game over (savoir qui a gagné)

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- Meilleur UI in game (cooldowns, level up etc.)
- Un meilleur shader pour l'eau et meilleures particules
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- Meilleur décor in game (des poissons, des mouettes etc...)
- **Animation de game over (savoir qui a gagné)**

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- Meilleur UI in game (cooldowns, level up etc.)
- Un meilleur shader pour l'eau et meilleures particules
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- Meilleur décor in game (des poissons, des mouettes etc...)
- **Animation de game over (savoir qui a gagné)**

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- Meilleur UI in game (cooldowns, level up etc.)
- Un meilleur shader pour l'eau et meilleures particules
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- Meilleur décor in game (des poissons, des mouettes etc...)
- **Animation de game over (savoir qui a gagné)**

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- Meilleur UI in game (cooldowns, level up etc.)
- Un meilleur shader pour l'eau et meilleures particules
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- Meilleur décor in game (des poissons, des mouettes etc...)
- **Animation de game over (savoir qui a gagné)**

Ce qu'on aurait voulu faire en plus

- Plus de bonus et pouvoir les différencier
- Meilleur UI in game (cooldowns, level up etc.)
- Un meilleur shader pour l'eau et meilleures particules
- Un système de tir de mortier plus généreux en taille au niveau du point d'impact
- Meilleur décor in game (des poissons, des mouettes etc...)
- Animation de game over (savoir qui a gagné)
- **Ajouter plus de bugs**

Démonstration