

Projet Bataille Navale



Introduction

Chaque joueur possède une flotte de 10 bateaux :

- 1 étage (une seule case) – 4 bateaux ;
- 2 étage (deux cases) – 3 bateaux ;
- 3 étage (trois cases) – 2 bateaux ;
- 4 étage (quatre cases) – 1 bateau.

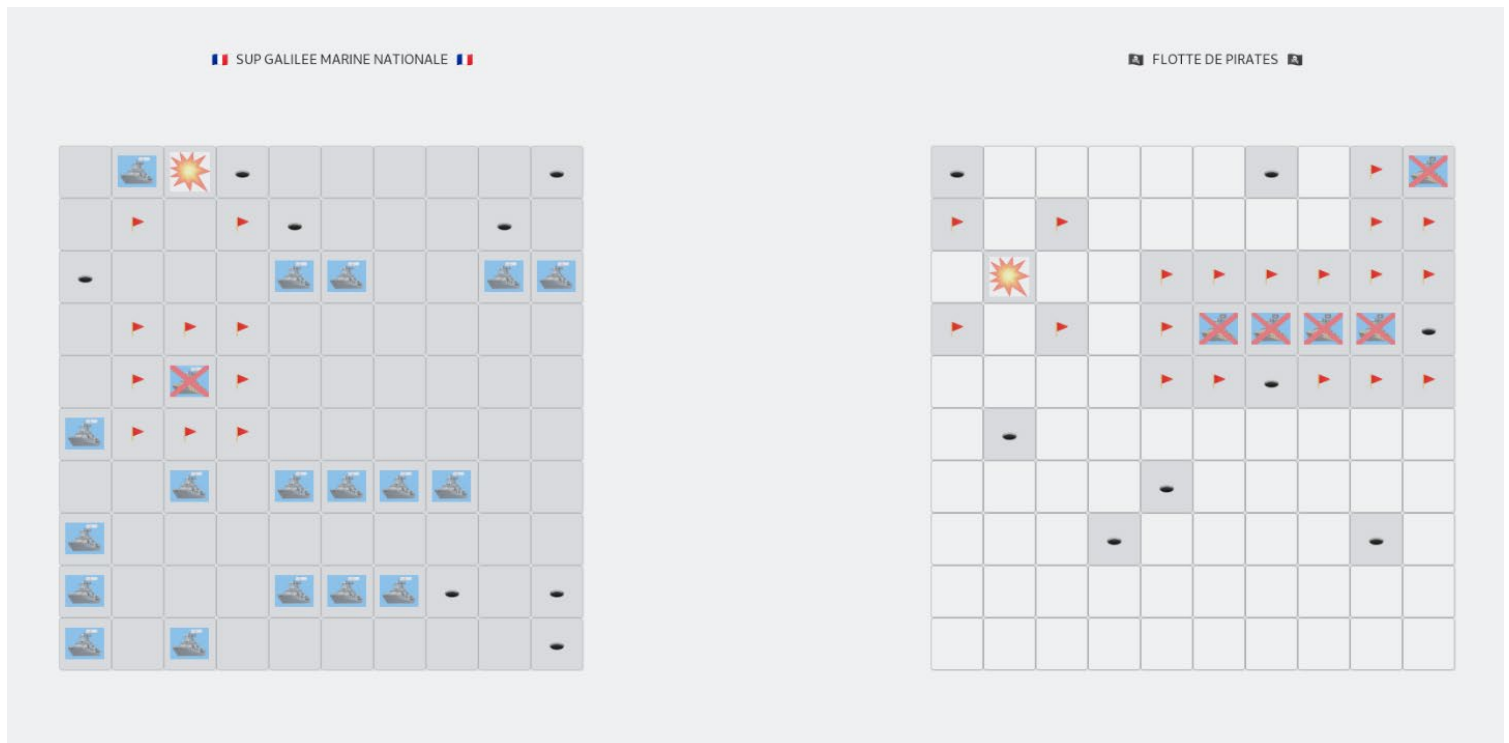
Tous les navires répertoriés doivent être placés sur un champ carré de 10 par 10, et les navires ne peuvent toucher ni les coins ni les côtés.

Le but de ce jeu est de détruire tous les navires ennemis. Celui qui le fait le premier gagne.

En cas de tir réussi sur le navire ennemi, une croix est placée sur la case correspondante du champ ennemi et un deuxième coup est tiré. **Le jeu se termine** lorsque l'un de ses participants perd tous ses navires.

1. Aspect du terrain de jeu

La capture d'écran ci-dessous montre à quoi ressemble le terrain de jeu, combien de types de navires, en fonction du nombre de ponts, il y aura dans l'escadron, à quoi ressemble un échec, un coup et une marque où le navire ne peut pas être exactement.



Pont de navire



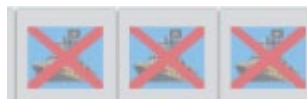
Bévue



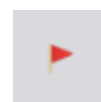
L'un des ponts du navire a été touché



Tout le navire touché

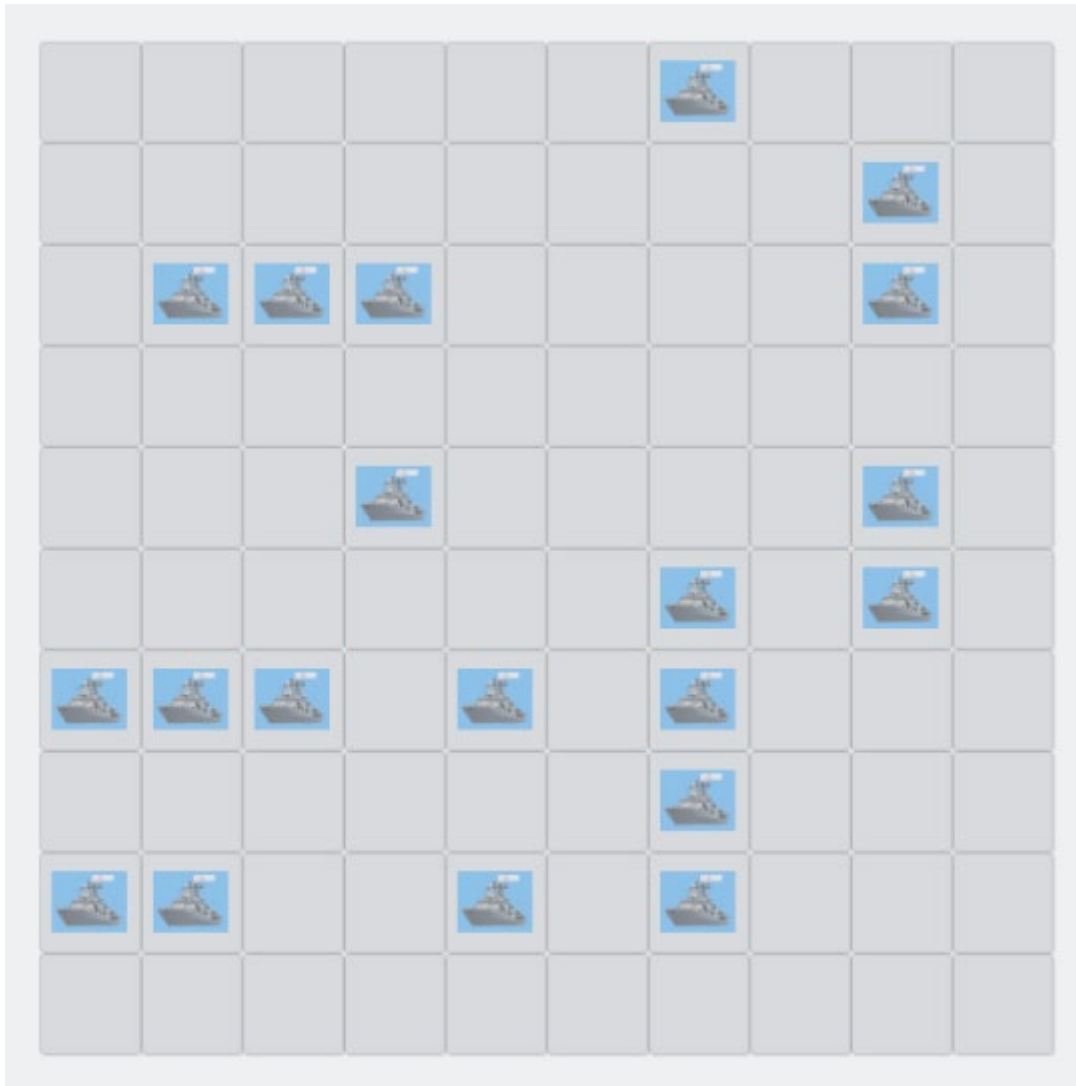


Une marque où un navire ne peut pas être placé



2. Disposition des navires dans le jeu

Les navires peuvent être placés verticalement et horizontalement, mais il doit y avoir au moins une cellule vide entre les navires, y compris en diagonale. Les navires ne peuvent pas être en forme de L.



Toutes ces exigences sont mises en œuvre dans le programme. Dans ce cas, les navires des joueurs sont placés selon une loi aléatoire. L'algorithme est donné ci-dessous :

- 1) Un terrain de jeu vide de taille GRID_SIZE x GRID_SIZE est créé ;
- 2) Pour chaque type de navire, en commençant par le plus grand ;
 - 2.1) Des coordonnées aléatoires (X, Y) sont générées sur le terrain de jeu ;
 - 2.2) Une direction aléatoire est générée (horizontale ou verticale) ;
 - 2.3) Nous vérifions si le navire avec la direction et la taille données tient sur le terrain de jeu, à partir des coordonnées (X, Y) ;
 - 2.4) Si cela convient, placez le navire sur le plateau de jeu ;
 - 2.5) Si cela ne rentre pas, passez à l'étape 2.1. pour déterminer de nouvelles coordonnées et directions aléatoires ;
- 3) Renvoyez les navires placés sur le terrain de jeu.

Les restrictions suivantes sont imposées aux coordonnées générées (X,Y):

- a) le champ a un marqueur FALSE, ce qui signifie que la cellule n'est pas occupée.
- b) Nous utilisons la dépendance de la coordonnée maximale sur le nombre de ponts du navire. Pour la coordonnée "Y", avec le navire placé horizontalement, cela ressemblera à ceci :

$$y = (9 - \text{decks}) + 1$$

où decks est le nombre de ponts dont dispose le navire.

Dans ce cas, la coordonnée "X" peut prendre n'importe quelle valeur comprise entre 0 et 9. Des restrictions similaires sont imposées avec l'emplacement vertical du navire.

Un exemple de placement d'un navire à 3 ponts :

- Nous pouvons commencer par les cellules 0, 1, 2, ..., 7, puis il y aura de la place pour placer le navire dans trois cellules.



- Vous ne pouvez pas commencer à partir des cellules 8 et 9, car le navire sortira du champ



3. [Tir du joueur](#)


Chacun des joueurs a le droit de tirer, à condition que la valeur de la variable de transition de virage soit inversée

is_player1_turn == TRUE ou is_player2_turn != TRUE

Le programme traite les informations en fonction du résultat du tir :

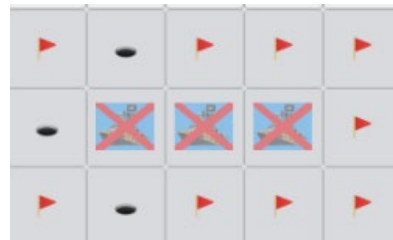
- manquer ;
- monter sur le pont du navire ;
- frapper tout le navire.

3.1) En cas d'échec, nous effectuons les actions suivantes :

- afficher visuellement un raté 
- marquez la cellule comme inaccessible pour les prises de vue suivantes ;
- modifier la valeur d'une variable (`is_player1_turn = FALSE`), passant ainsi le déménagement à l'ordinateur.

3.2) L'algorithme de traitement du résultat de la frappe d'un navire ennemi :

- afficher visuellement le coup et marquer la cellule comme inaccessible pour les tirs suivants ;
- afficher un message sur le hit ;
- selon les coordonnées du tir, nous déterminons le navire qui a été touché ;
- augmenter le compteur de coups sur le navire donné par un ;
- si le nombre de coups est égal au nombre de ponts du navire, alors affichez visuellement que tout le navire est coulé



- nous marquons les cellules le long du périmètre sous la forme d'un drapeau, car les cellules sont disponibles pour le tir, car, selon les règles du jeu, les navires ne peuvent pas toucher les ponts.
- vérifier si toutes les sources sont coulées. Selon le résultat, soit nous continuons (faites le coup suivant), soit nous terminons la rotation du jeu en déclarant la victoire ;
- on passe le coup au joueur, puisqu'une touche est enregistrée pendant le tir.

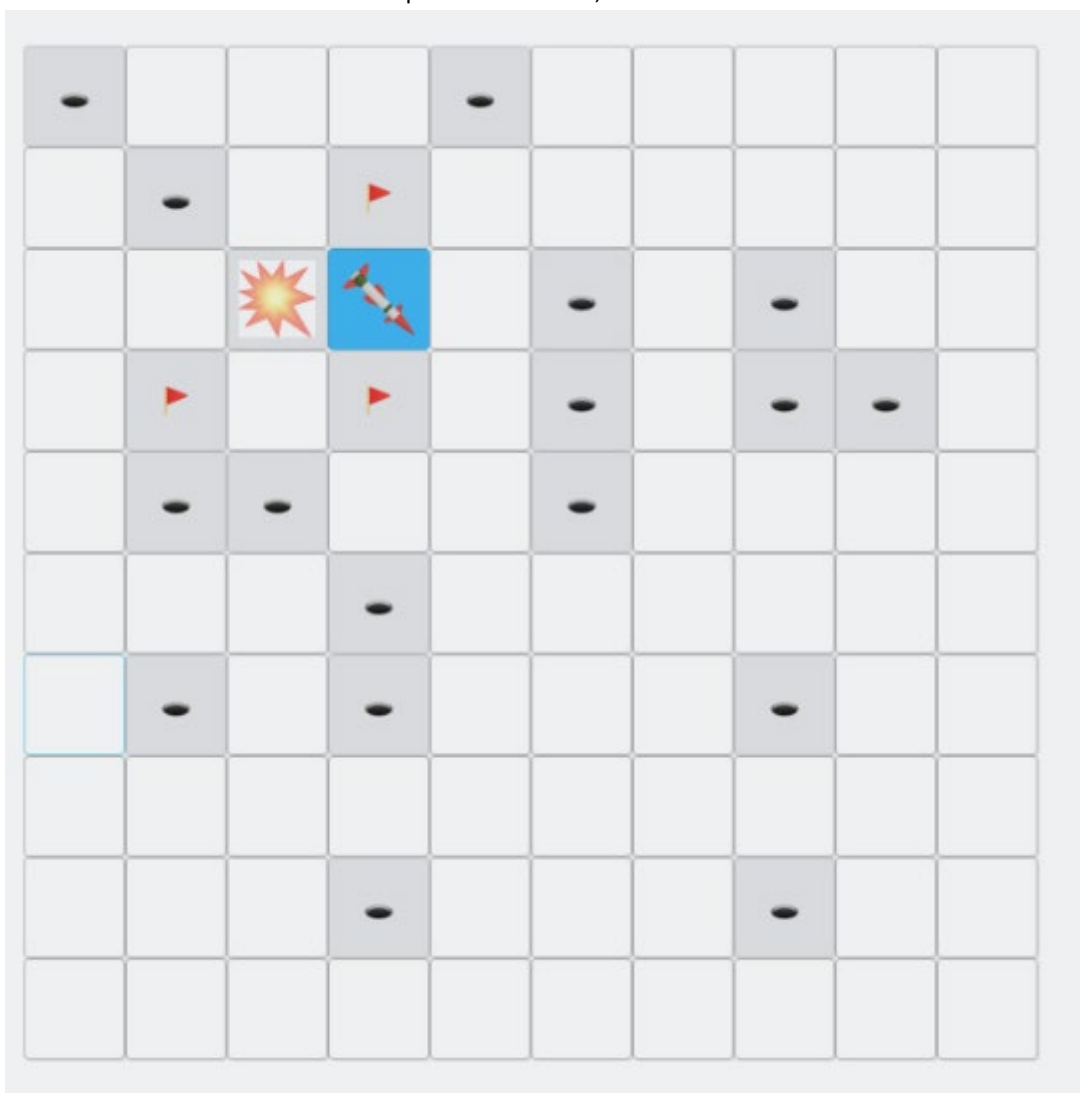
4. Coup d'ordinateur

Pour qu'un ordinateur joue au moins sur un pied d'égalité avec une personne, il (l'ordinateur) doit avoir une sorte de tactique de jeu. De plus, cette tactique devrait être optimale par rapport au jeu "Battleship" et offrir de grandes chances de vaincre un homme.

L'algorithme de traitement des coups coïncide en grande partie avec l'algorithme de traitement des coups du navire du joueur, mais il y a des ajouts importants concernant l'intelligence artificielle de l'ordinateur, qui est un algorithme d'action qui change en fonction de la situation sur le terrain de jeu.

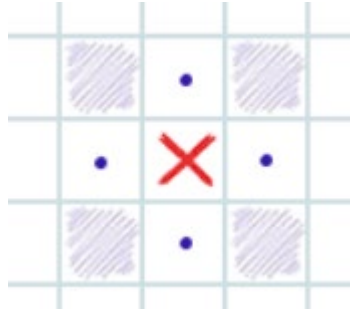
a) Nous marquons les cellules vides garanties autour du hit.

Après chaque tir réussi, l'ordinateur détermine les coordonnées des cellules situées en diagonale de la cellule touchée. Selon les règles du jeu, il n'y a définitivement aucun navire de joueur à l'intérieur. Il peut y avoir de une à quatre cellules de ce type, en fonction des coordonnées du coup - au centre, au bord ou au coin du terrain de jeu.



b) Bombarder le terrain de jeu autour du coup.

Après avoir touché le vaisseau du joueur, l'ordinateur doit continuer à tirer dessus jusqu'à ce qu'il soit complètement détruit. Il est logique de supposer que le prochain pont sera dans une cellule adjacente. Après le premier coup, on ne sait toujours pas comment le navire est situé - horizontalement ou verticalement. Par conséquent, il est nécessaire de tirer sur la cellule avec un coup de tous les côtés. Le nombre de coordonnées de tir peut varier de deux à quatre, selon les coordonnées du coup :



La validation des coordonnées de décortilage s'effectue en deux étapes :

- a. un. Dans un premier temps, on vérifie que la coordonnée ne dépasse pas le terrain de jeu. Ensuite, nous vérifions la valeur de la matrice par les coordonnées restantes du bombardement.
- b. Après cela, nous vérifions si le navire est coulé.
- c. En cas de coup sur un navire ou sur le pont d'un navire, nous plaçons des marqueurs autour du périmètre du navire afin qu'il soit impossible de trouver un autre navire ennemi dans ces cellules.

Ainsi, en appliquant les tactiques optimales pour traiter un ordinateur frappant un navire ennemi, nous réduisons le nombre de cellules disponibles pour un tir, augmentant ainsi la probabilité d'un coup ultérieur sur d'autres navires.

5. Destruction de la flotte ennemie et fin de partie.

La flotte ennemie est considérée comme détruite si la valeur de la variable

`player1_ships_left == 0` ou `player2_ships_left == 0`

Dans ce cas, si le joueur a gagné, on lui montre une félicitation pour la victoire. Sinon, un message de perte s'affiche.

Conclusion

Le programme de jeu Bataille Navale implémente toutes les règles du jeu de bataille navale. La logique des mouvements informatiques optimaux est mise en œuvre.

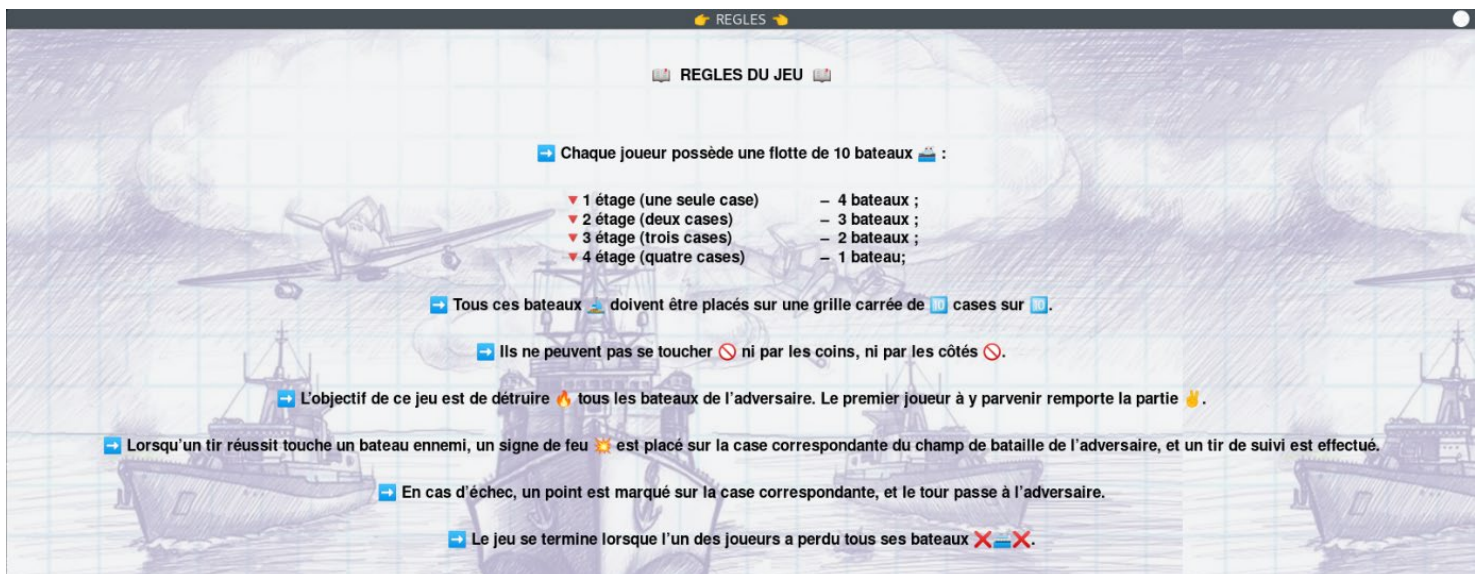
En complément, nous pouvons proposer la mise en place d'une autre façon de jouer : non seulement un joueur avec un ordinateur, mais aussi deux joueurs. On voudra également ajouter des effets audio et vidéo.

De plus, dans un élan de recherche et d'apprentissage nous avons décidé de coder l'entièreté du jeu. Ainsi les fonctions utilisées ne sont pas celles qui ont été fournies. Cela nous a permis de mieux appréhender la bibliothèque gtk et de nous exercer.

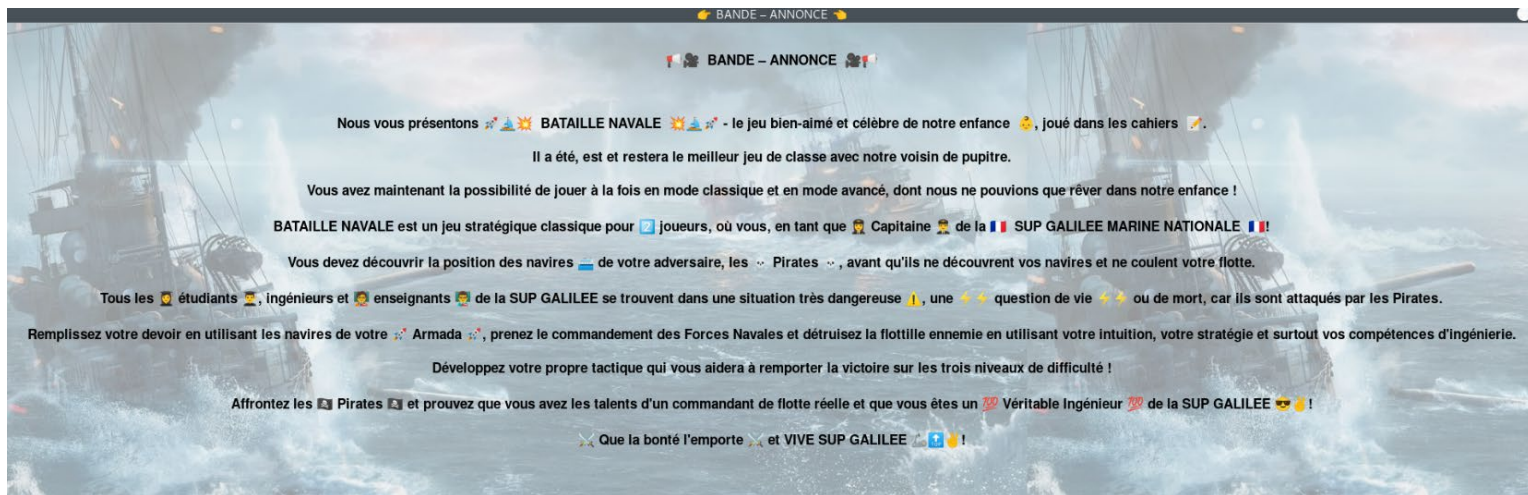
Annexes



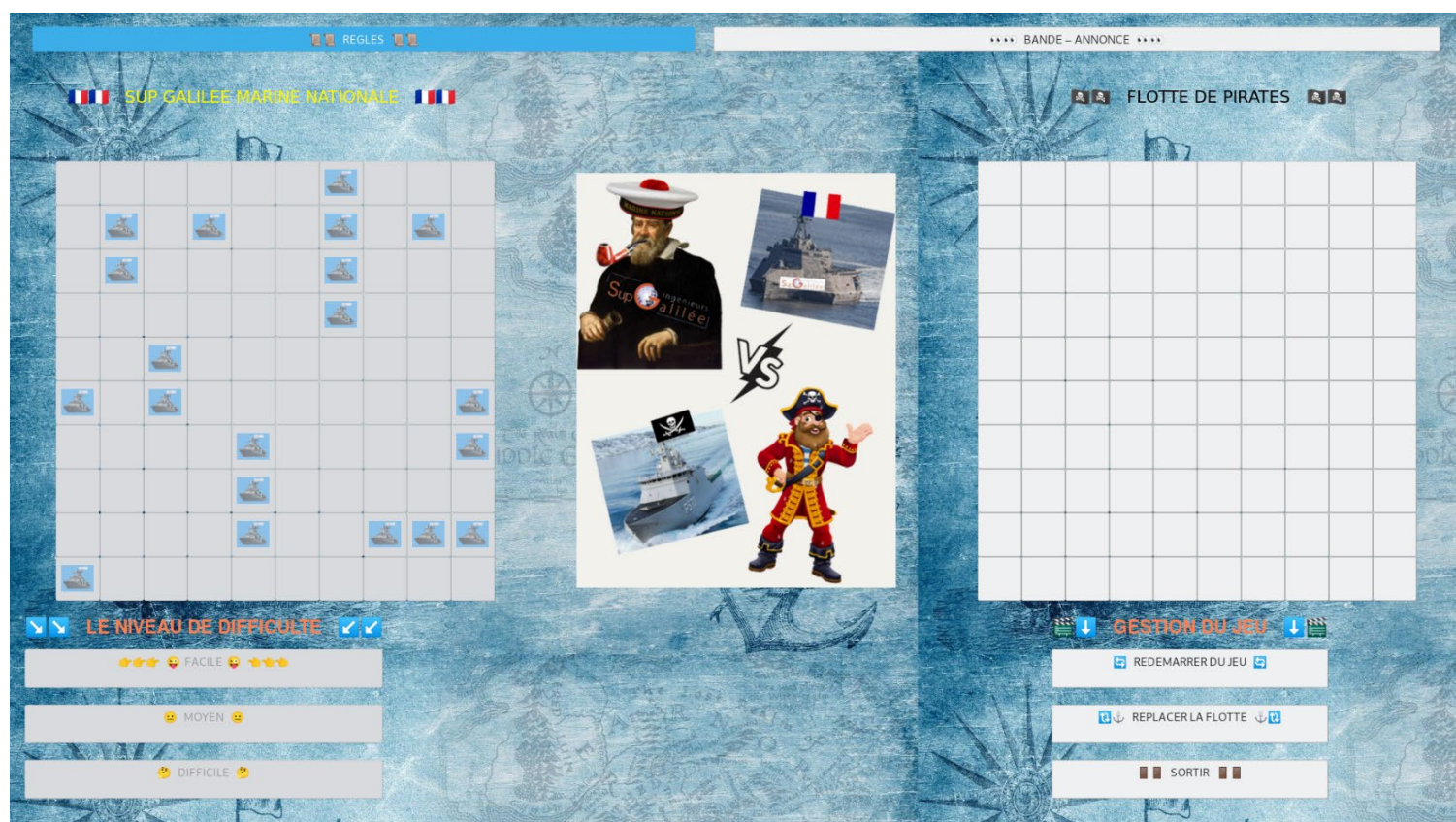
Preview



Regles



Bande Annonce



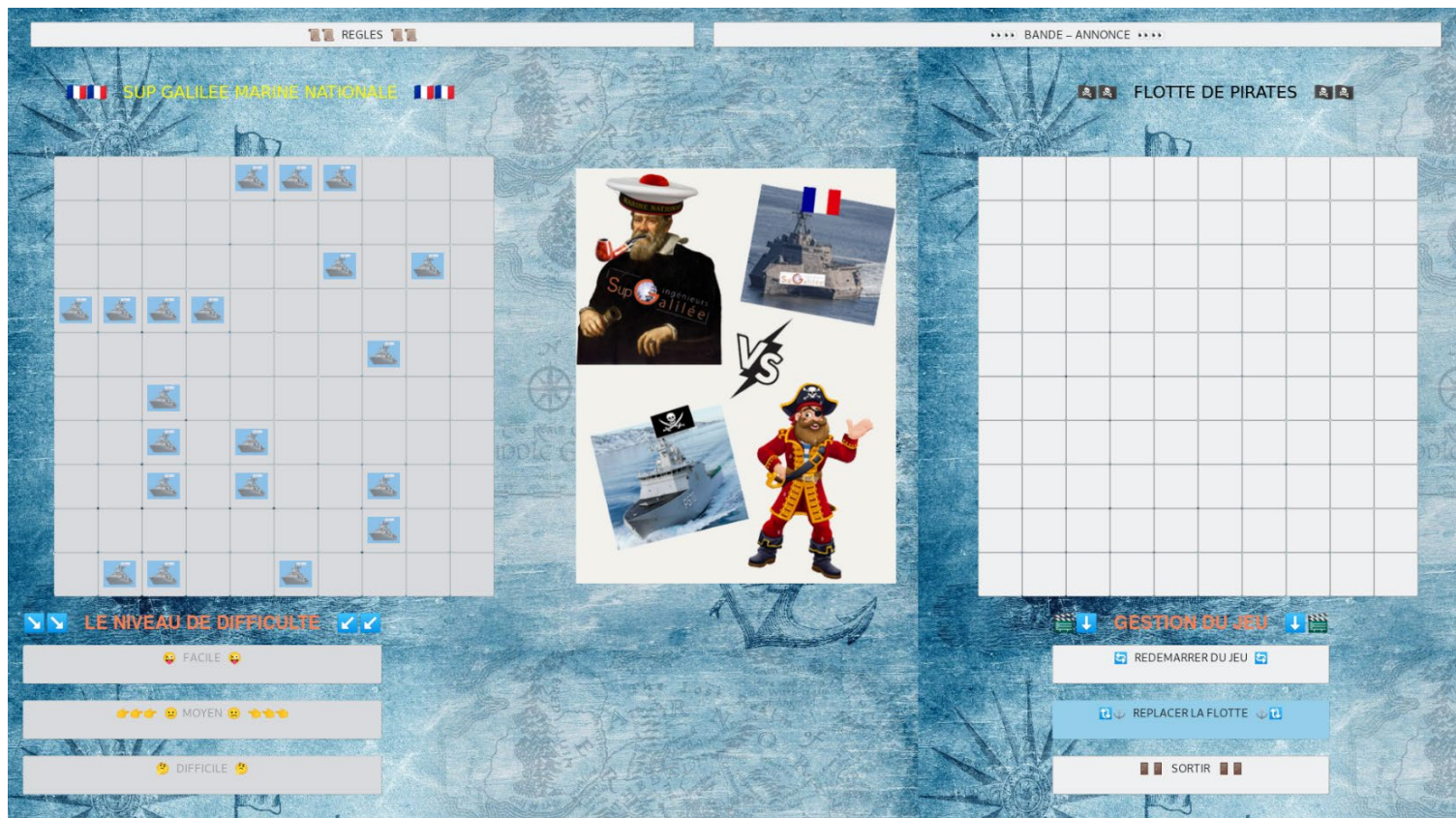
Jeu Facile



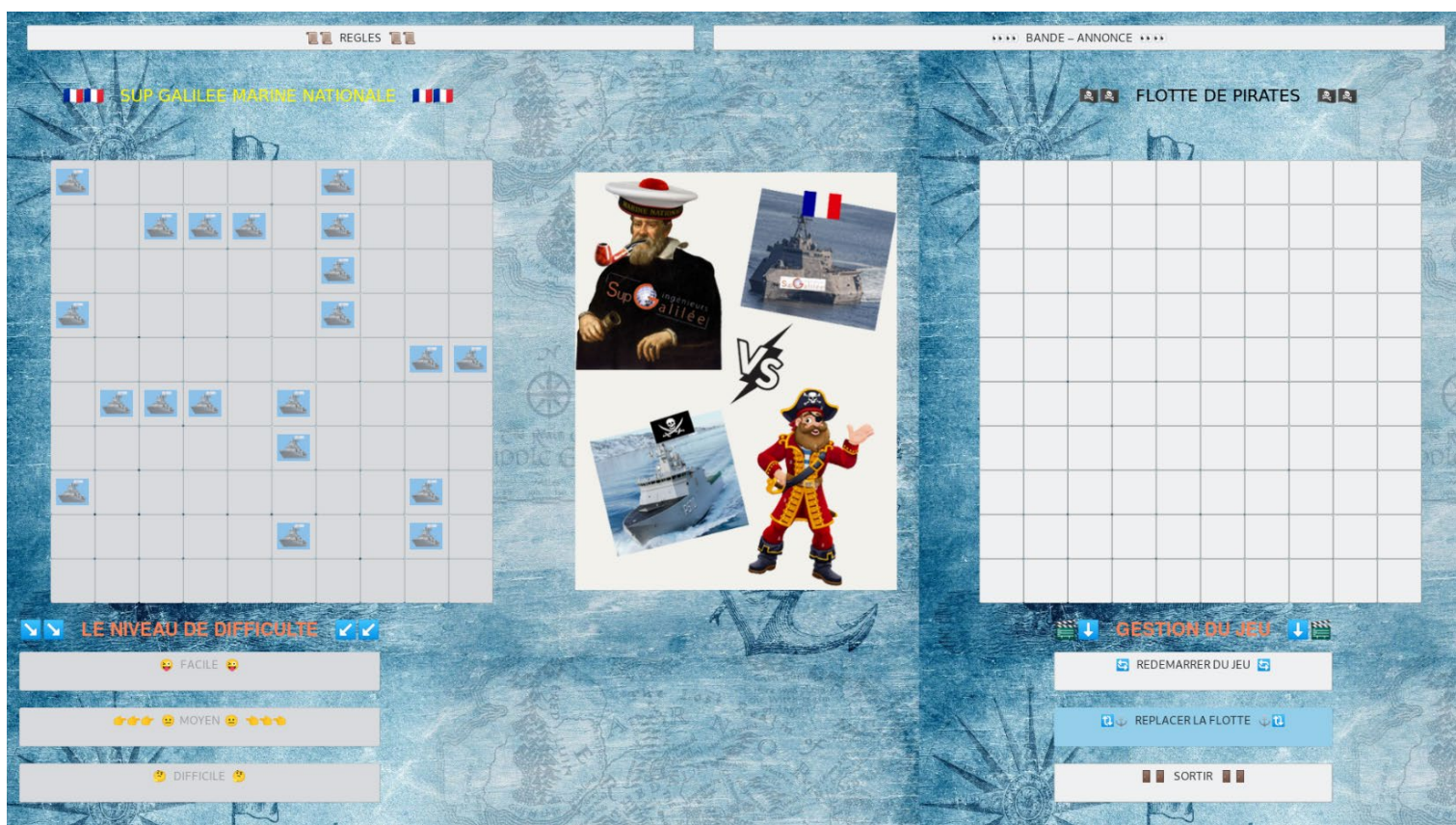
Positionnement de la Flotte No 1



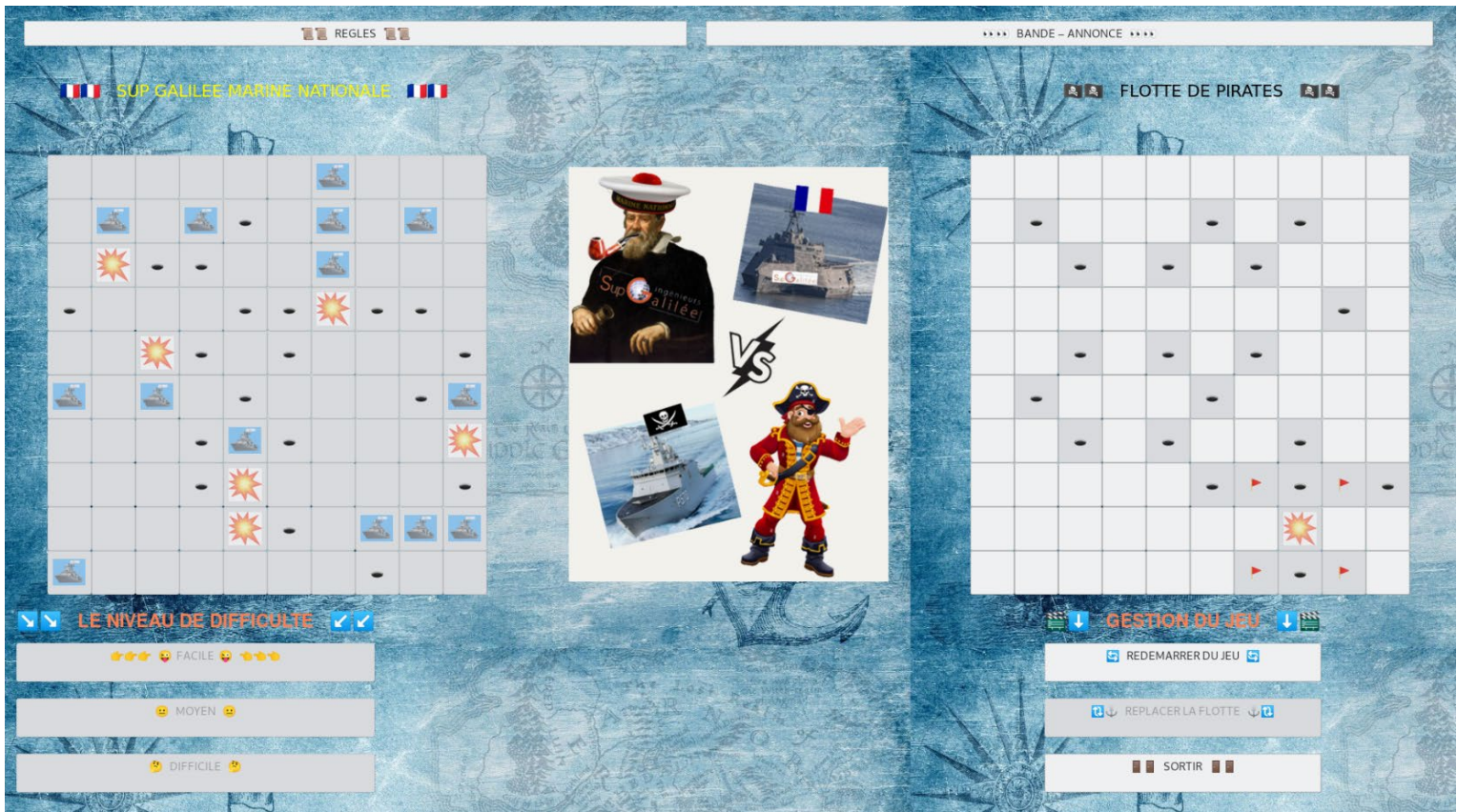
Positionnement de la Flotte No 2



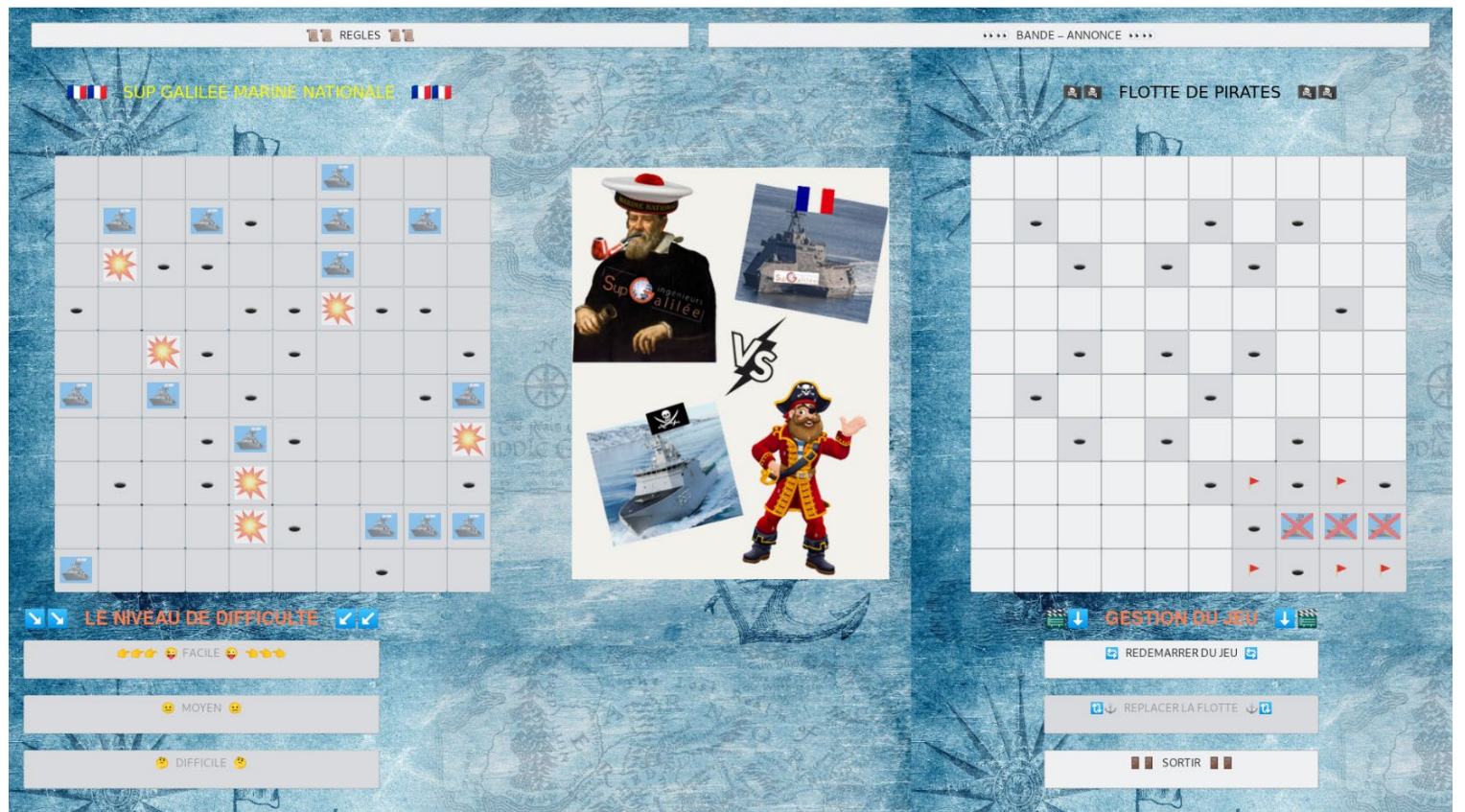
Positionnement de la Flotte No 3



Positionnement de la Flotte No 4



Jeu No 1



Jeu No 2



Jeu No 3



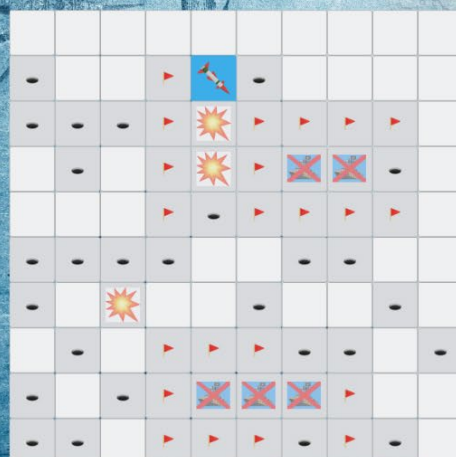
Jeu No 4

REGLES

BANDE - ANNONCE

SUP GALILEE MARINE NATIONALE

FLOTTE DE PIRATES



LE NIVEAU DE DIFFICULTE

FACILE

MOYEN

DIFFICILE

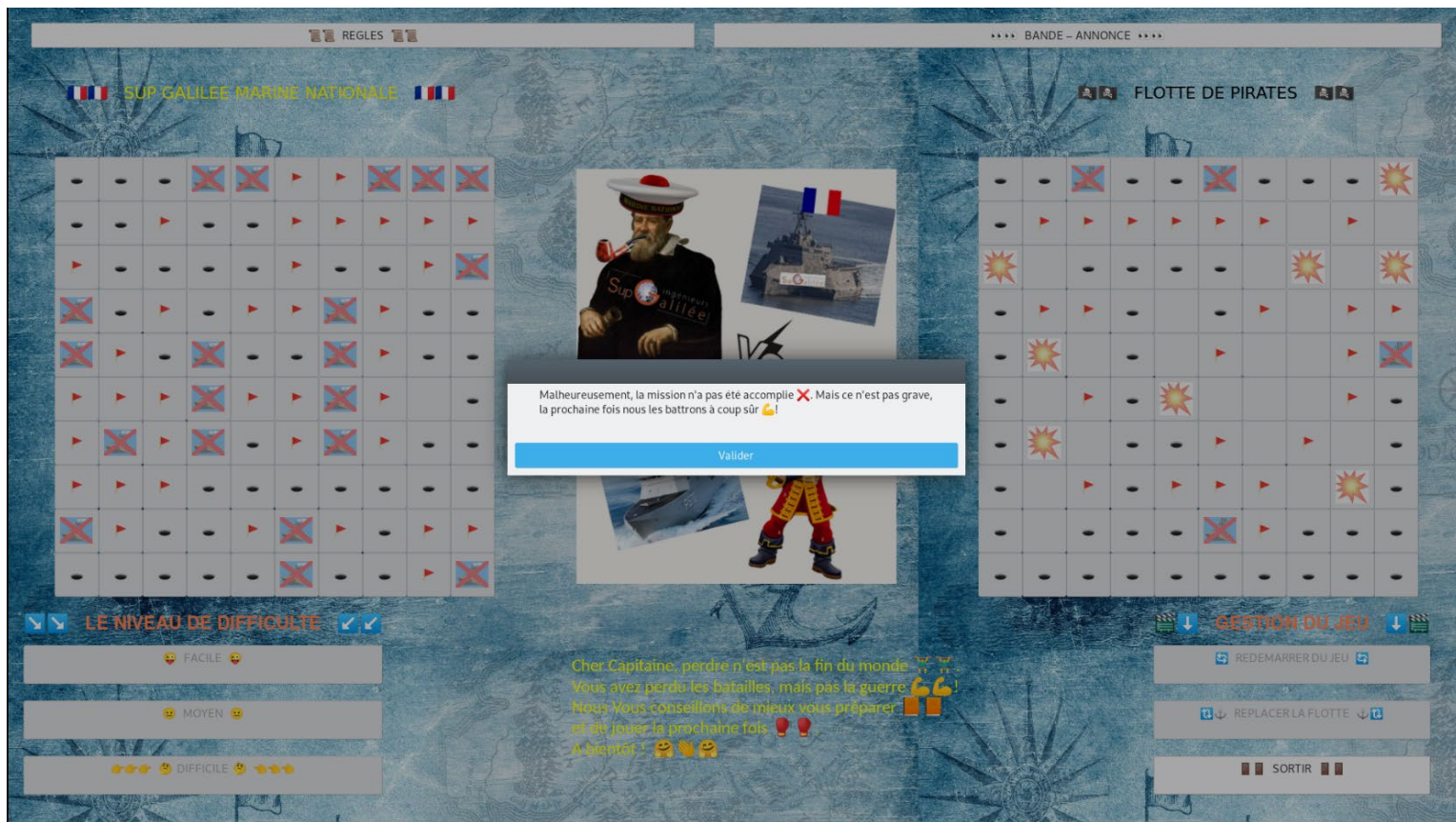
GESTION DU JEU

REDEMARRER DU JEU

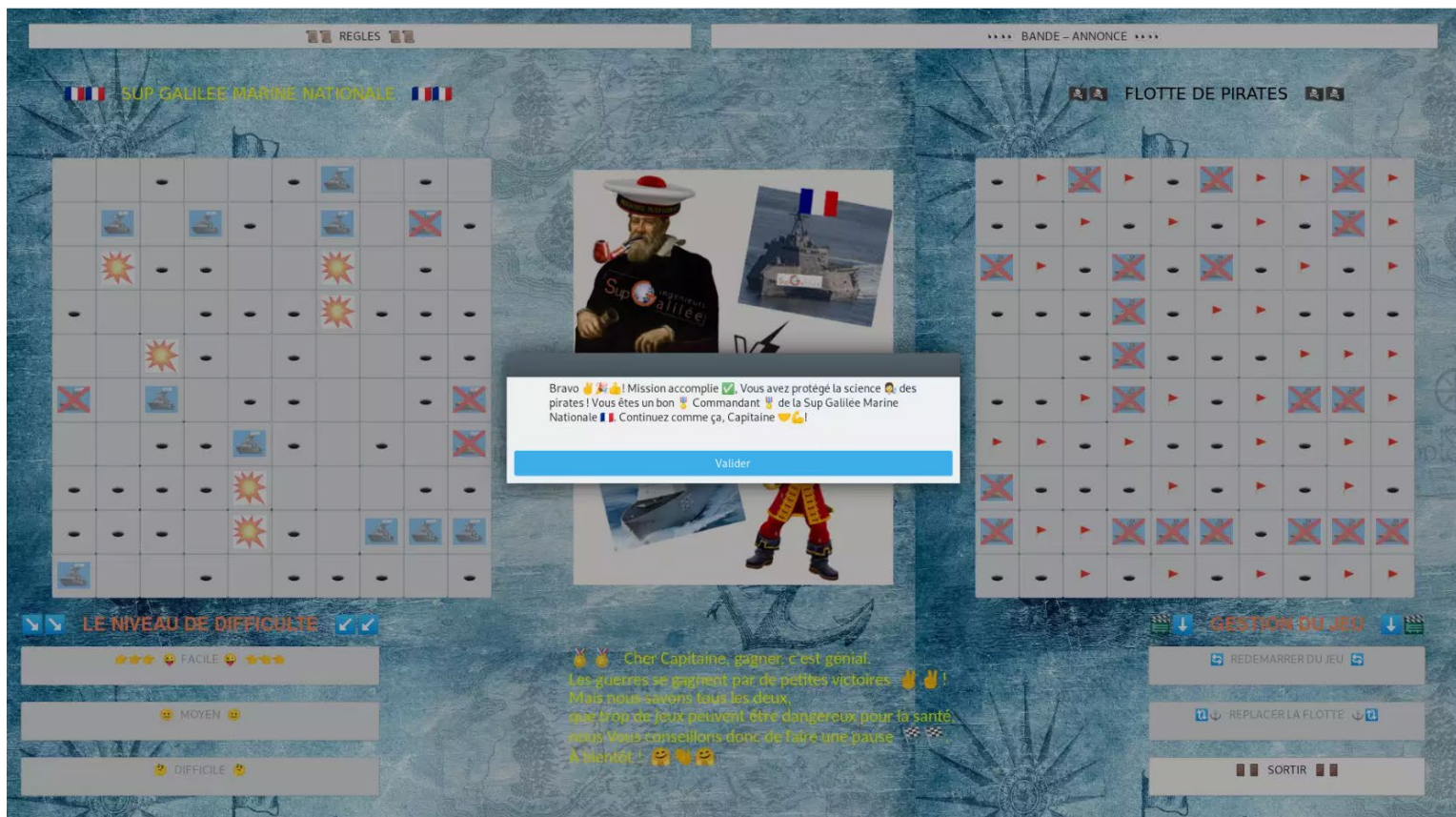
REPLACER LA FLOTTE

SORTIR

Jeu No 5



Fin du Jeu – Perte



Fin du Jeu – Victoire