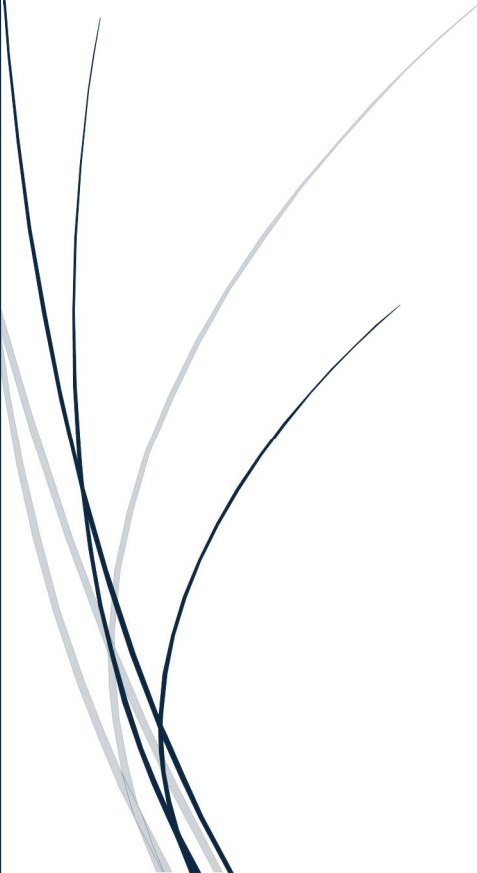


28/08/2024

# Battleship Lite

Cahier de conception



Jérôme Savard & Olivier Bergeron-Houde &  
Samara Boudreault  
CÉGEP DE CHICOUTIMI

## Table des matières

Échéanciers .....	2
Analyse des besoins .....	2
Diagrammes .....	3
Cas d'utilisation.....	3
Séquences .....	4
Classes .....	5
Plan de test .....	6

## Échéanciers

Les remises attendues par le client sont :

Remises	Dates
Plan de développement	2024-09-04
Tests unitaires	2024-09-04
Remise finale	2024-09-15

## Planification

Fonctionnalités	Dates
Documentation & planification	2024-09-04
Sérialiseur	2024-09-04
Créer structure de base	2024-09-04
Créer graphiques	2024-09-06
Ajout connexion P2P	2024-09-11
Ajout fonctionnalités de jeu	2024-09-11
Ajout tests unitaires	2024-09-13

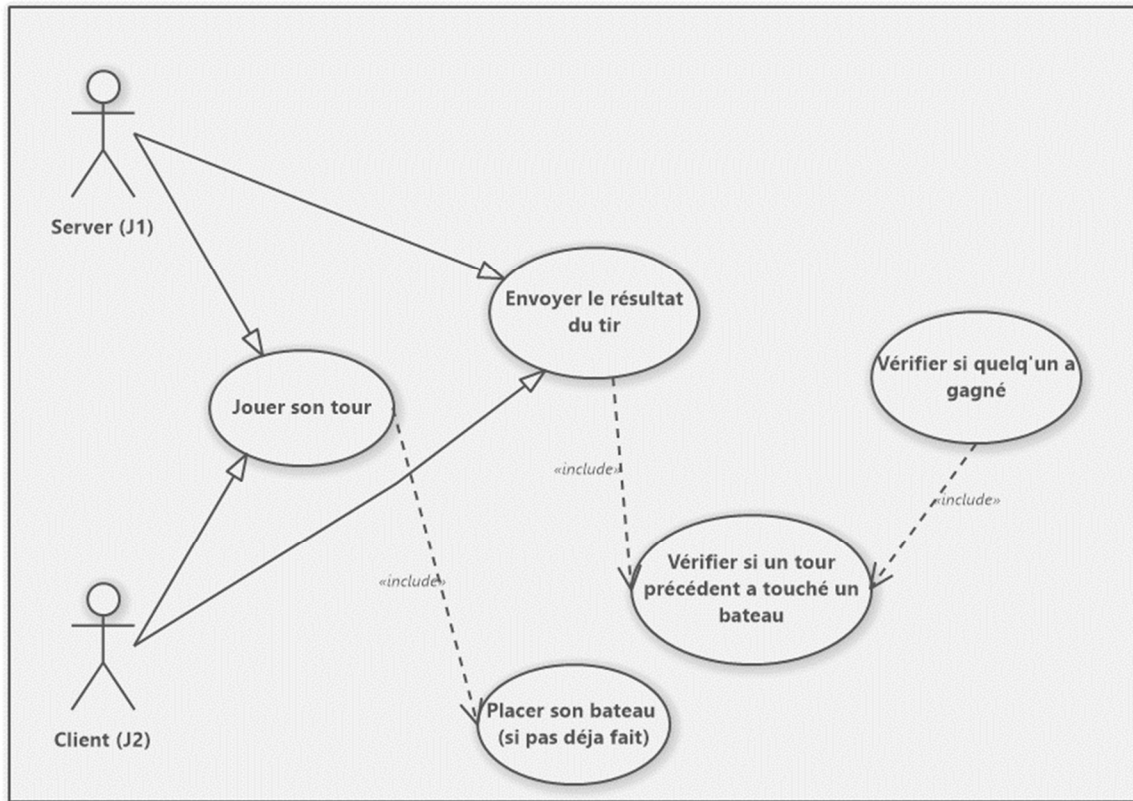
## Analyse des besoins

L'application devra permettre de jouer au jeu de BattleShipLite en réseau (2 joueurs). (à tour de rôle).

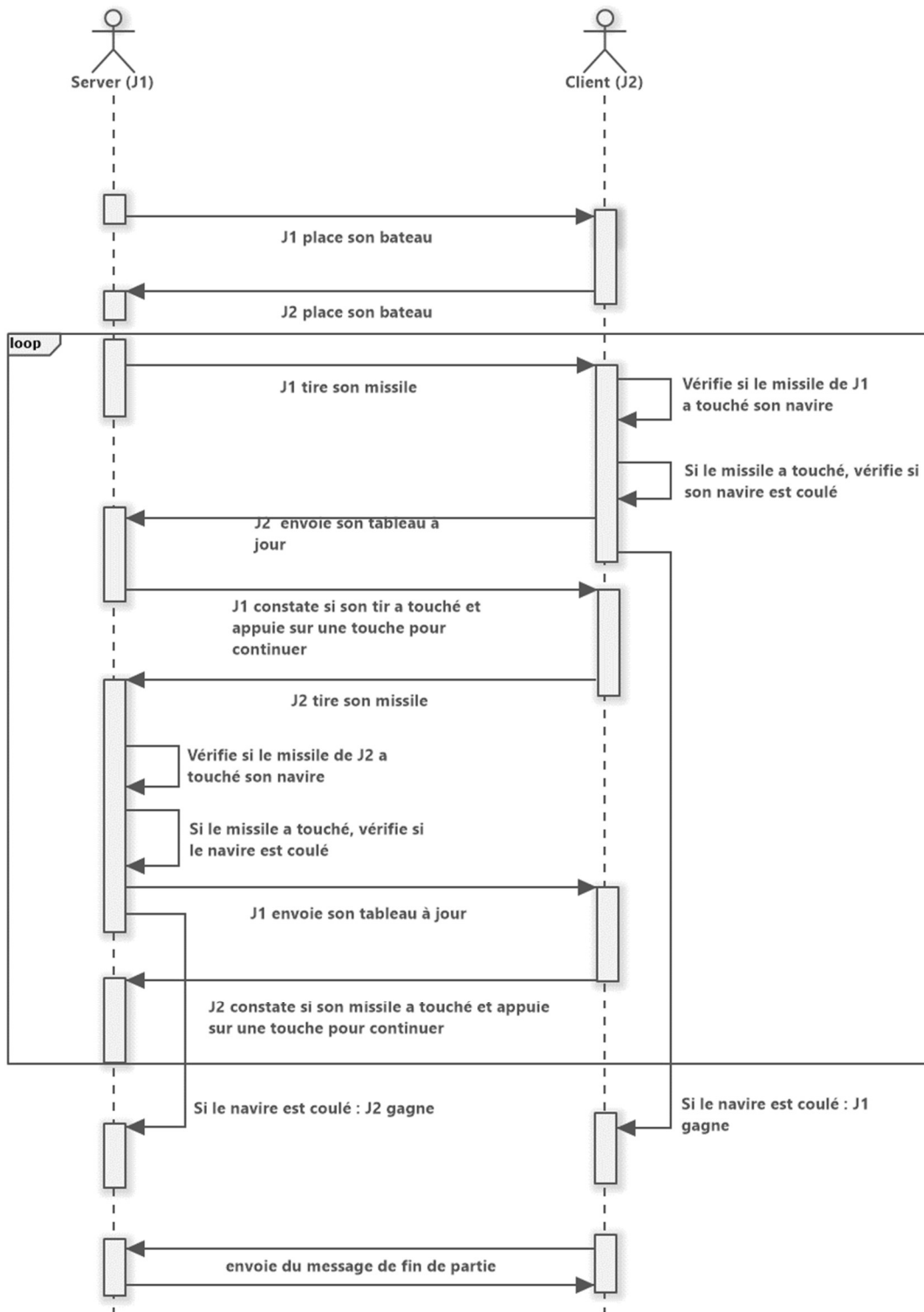
- L'application doit pouvoir jouer sur 2 ordinateurs différents (application serveur sur un, le client sur un autre).
- L'application devra utiliser les mécanismes appropriés afin d'effectuer les communications entre le client et le serveur.
- L'application devra permettre la saisie des informations afin d'établir correctement la connexion entre le client et le serveur (adresse IP du serveur pour la connexion)

# Diagrammes

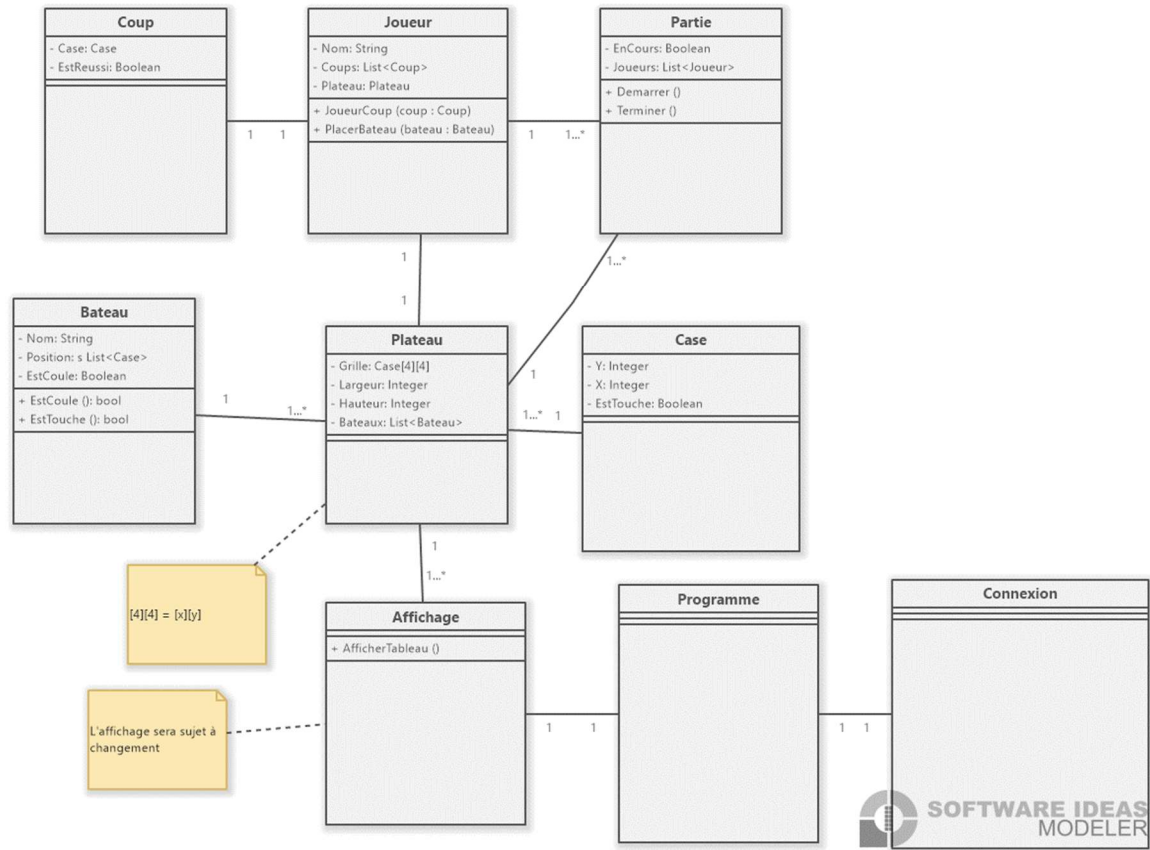
## Diagramme de Cas d'utilisation (durant une partie)



## Diagramme de Séquences



## Classes



\*Changement de Grille : Case[4][4] à Grille List<List<Case>>

## Plan de test

Composant	Scénario de Test	Détails
Client	Adresse IP Non Valide	Tester la réaction du client lorsque l'adresse IP est incorrecte.
	Port Non Valide	Vérifier la réaction du client avec un port incorrect.
	Réponse aux Commandes Invalides	Évaluer comment le client gère les commandes non reconnues ou invalides.
Serveur	Perte de Connexion Réseau	Vérifier comment le serveur gère une perte de connexion réseau.
	Fermeture de l'Application par le Client	Tester la gestion du serveur lorsqu'un client ferme son application.
	Retour en Attente après Déconnexion	Assurer que le serveur passe correctement en attente après la déconnexion d'un client.
	Gestion des Erreurs de Synchronisation	Vérifier la gestion des erreurs de synchronisation des données par le serveur.
Les Deux	Modification Illégale du Tableau	Tester la réaction des deux côtés lorsqu'il y a une modification illégale du tableau.
	Coordonnées Hors Limites	Vérifier la gestion des attaques avec des coordonnées en dehors des limites du tableau.
	Coordonnées Non Valides (ex : "patate" ou espace)	Tester le comportement avec des coordonnées invalides.
	Déclaration de Victoire lorsque le Bateau est Coulé	Vérifier la déclaration correcte de la victoire lorsque le bateau d'un joueur est coulé.
	Limitation du Nombre de Coups à 16	Assurer que le nombre maximal de coups est limité à 16.
	Coup Répété au Même Endroit	Tester la gestion des coups pris au même endroit.
	Placement de Bateau Non Valide	Vérifier la gestion du placement du bateau non valide.
	Placement de Bateau à l'Extérieur	Assurer que les bateaux ne peuvent pas être placés à l'extérieur du tableau.

	Bateau Non Placé	Vérifier la réaction lorsque l'utilisateur essaie de jouer sans ses bateaux.
	Placement de bateau de grandeurs illégal	Tester les règles concernant la distance du bateau.
	Recommencement de Partie	Vérifier la fonctionnalité de recommencement d'une partie après une fin de partie.
	Gagnant	Vérifie s'il est possible d'avoir un gagnant

Les tests unitaires en **surligné** seront les tests qui seront fait dans l'application.