

# *Bataille Navale*



*Varoso Gomes, David Manuel  
Ch. De l'Union 10  
1008, Prilly  
David-Manuel.VAROSO-GOMES@cpnv.ch*



*SI-c1b*

Début : 06.02.2019  
Mise à jour : 02.04.2019

## Table des matières

1	Analyse préliminaire .....	3
1.1	Introduction .....	3
1.2	Organisation .....	3
1.3	Objectifs .....	3
1.4	Planification initiale .....	4
2	Analyse .....	4
2.1	Cahier des charges détaillé .....	4
2.2	Définition de l'audience .....	4
2.3	Cas d'utilisation .....	5
2.4	Stratégie de test .....	7
2.5	Budget initial .....	7
2.6	Planification détaillée .....	8
3	Conception .....	8
3.1	Analyse de l'environnement .....	8
3.2	Détermination de l'arborescence du site et des rubriques .....	8
3.3	Maquette graphique .....	9
3.4	Conception du Code .....	10
4	Réalisation .....	12
4.1	Dossier de réalisation .....	12
4.2	Description des tests effectués .....	13
4.3	Erreurs restantes .....	13
5	Mise en service .....	13
5.1	Rapport de mise en service .....	13
5.2	Liste des documents fournis .....	13
6	Conclusions .....	14
7	Annexes .....	14
7.1	Sources – Bibliographie .....	14
7.2	Journal de travail .....	15
7.3	Archives du projet .....	16

## **1 Analyse préliminaire**

### **1.1 Introduction**

Ce programme a été fait dans le cadre du module MA-20 pour développer nos compétences en C, il a pour but de permettre à un utilisateur de jouer à la bataille navale.

Évalué pour le 3<sup>ème</sup> trimestre, la limite de temps était de 8 semaines.

### **1.2 Organisation**

*Elève : Varoso Gomes, David Manuel, David-Manuel.VAROSO-GOMES@cpnv.ch, 078 756 98 63*

*Responsable de projet (enseignant : BENZONANA, Pascal, pascal.benzonana@cpnv.ch et 024 / 55 + 76079)*

*Expert 1 : BENZONANA, Pascal, pascal.benzonana@cpnv.ch 024 / 55 + 76079*

*Expert 2 : CARREL, Xavier, xavier.carrel@cpnv.ch 024 / 55 + 76080*

*Intervenants externes : Testeur(s)*

### **1.3 Objectifs**

Général :

Créer un programme qui permettra à un utilisateur de pouvoir jouer à la bataille navale avec un accès à un menu complet.

Spécifique :

L'utilisateur doit avoir accès à une aide complète dans le menu.

L'utilisateur doit pouvoir quitter le programme à partir du menu.

L'utilisateur pourra voir son score à la fin de la partie.

L'utilisateur pourra s'authentifier avant de lancer la partie

Lien GitHub pour plus : <https://github.com/DavidVarosoGomesCPNV/ICT-431-David-Gomes/issues>

## 1.4 Planification initiale

4 Open ✓ 0 Closed		Sort ▼
<b>Planning général</b> Updated 14 days ago	<ul style="list-style-type: none"> <li>Analyse et documentation sur le projet</li> <li>Création de la version 0.1 / 1.0</li> <li>Rendu et finalisation de la version finale</li> </ul>	...
<b>Sprint 3</b> Updated 2 days ago	<ul style="list-style-type: none"> <li>Rendu et finalisation du code de la bataille navale</li> <li>Finalisation de la documentation</li> </ul>	...
<b>Sprint 2</b> Updated 2 days ago	Création de la version 0.1 / 1.0	...
<b>Sprint 1</b> Updated 14 days ago	Analyse et documentation sur le projet	...

## 2 Analyse

### 2.1 Cahier des charges détaillé

Le type d'application C demandé doit offrir les fonctionnalités minimales suivantes :

Version 0.1 (pour le xxx 2018) :

- Jouer à la bataille navale alors que la position des bateaux est codée dans le programme
- Afficher l'aide du jeu

Version 1.0 (pour le xxx 2019)

- S'authentifier en tant que joueur
- Logger les faits importants durant toute la durée de fonctionnement de l'application
- Le programme choisit une grille au hasard dans un répertoire et la charge
- Afficher la liste des scores atteints lors des parties précédentes

### 2.2 Définition de l'audience

*Joueurs tout âge aimant les jeux de stratégie.*

## 2.3 Cas d'utilisation

### Case 1 : Tirer sur une case (Touché, Raté, Coulé)

Identifiant	Tirer sur une case (toucher, rater , couler
En tant que	Utilisateur
Je veux	Tirer
Pour	Toucher, Rater ou couler
Priorité	M

Action	Condition	Réaction
L'utilisateur lance le programme		Le programme ce lance et le menu s'affiche
L'utilisateur rentre son prénom		Le programme enregistre son prénom et un menu ce lance
L'utilisateur sélectionne jouer		Une sélection de grille s'affiche sur l'écran
L'utilisateur sélectionne une grille disponible		Le programme affiche la grille choisie par l'utilisateur
L'utilisateur tire en rentrant une colonne & ligne	Un bateau ce trouvait dans la case visée	Un message s'affiche « Touché »,
L'utilisateur tire en rentrant une colonne & ligne	Aucune bateau ne se trouvait dans le case visée	Un message s'affiche « Raté »,
L'utilisateur tire en rentrant une colonne & ligne	Toutes les cases du bateau ont été touchées	Un message s'affiche « Coulé »,
L'utilisateur tire en rentrant une colonne & ligne	Le tir touche la dernière case du dernier bateau	Une message s'affiche « Gagné »,

### Case 2 : Sélectionner Jouer

Identifiant	Sélectionner Jouer
En tant que	Utilisateur
Je veux	Lancer le programme
Pour	Jouer
Priorité	M

Action	Condition	Réaction
L'utilisateur lance le programme		Le programme ce lance et le menu s'affiche
L'utilisateur rentre son prénom		Le programme enregistre son prénom et un menu ce lance
L'utilisateur sélectionne jouer		Le programme ce lance & affiche une sélection de grilles sur l'écran

### Case 3 : Sélectionner Quitter

Identifiant	Sélectionner quitter
En tant que	Utilisateur
Je veux	Quitter le programme
Pour	Arrêter de jouer
Priorité	M

Action	Condition	Réaction
L'utilisateur lance le programme		Le programme se lance et le menu s'affiche
L'utilisateur rentre son prénom		Le programme enregistre son prénom et une menu se lance
L'utilisateur sélectionne Quitter		Le programme se coupe

#### *Case 4 : Sélectionner Aide*

Identifiant	Sélectionner Aide
En tant que	Utilisateur
Je veux	Afficher le menu aide
Pour	Connaître les règles
Priorité	M

Action	Condition	Réaction
L'utilisateur lance le programme		Le programme se lance et le menu s'affiche
L'utilisateur rentre son prénom		Le programme enregistre son prénom et une menu se lance
L'utilisateur sélectionne Aide		Le programme affiche les règles / commandes

#### *Case 5 : Sélectionner Score*

Identifiant	Sélectionner Score
En tant que	Utilisateur
Je veux	Afficher le score
Pour	Savoir si on a mieux joué à
Priorité	M

Action	Condition	Réaction
L'utilisateur lance le programme		Le programme se lance et le menu s'affiche
L'utilisateur sélectionne jouer		Le jeu se lance
L'utilisateur coule tous les		Il a gagné la partie

bateaux		
L'utilisateur sélectionne le fichier score		Il peut voir le score de tous ceux qui ont joué

## Case 6 : Choix d'une grille

Identifiant	Choix d'une grille
En tant que	Utilisateur
Je veux	Choisir une grille
Pour	Jouer
Priorité	M

Action	Condition	Réaction
L'utilisateur lance le programme		Le programme se lance et le menu s'affiche
L'utilisateur rentre son prénom		Le programme enregistre son prénom et un menu se lance
L'utilisateur sélectionne jouer		Une sélection de grille s'affiche sur l'écran
L'utilisateur sélectionne une grille disponible parmi la liste		Le programme affiche la grille choisie par l'utilisateur

## 2.4 Stratégie de test

Tests à effectuer :

- ❖ L'utilisateur lance le programme
- ❖ L'utilisateur rentre son prénom
- ❖ Le prénom de l'utilisateur est pris en compte
- ❖ L'utilisateur sélectionne jouer
- ❖ L'utilisateur sélectionne une grille disponible
- ❖ L'utilisateur tire en rentrant une colonne & ligne
- ❖ L'utilisateur sait qu'il a touché un bateau
- ❖ L'utilisateur sait qu'il a raté un bateau
- ❖ L'utilisateur sait qu'il a coulé un bateau
- ❖ L'utilisateur sélectionne quitter
- ❖ L'utilisateur sélectionne aide
- ❖ L'aide s'affiche
- ❖ L'utilisateur coule tous les bateaux
- ❖ L'utilisateur sait qu'il a gagné
- ❖ L'utilisateur sélectionne score
- ❖ Le score est dans le fichier score

Résultats sur le fichier Excel Stratégie de test.xlsx

## 2.5 Budget initial

- Licences : CLion étudiant
- Matériel : PC Professionnel
- Temps : 8 semaines

## **2.6 Planification détaillée**

<https://github.com/DavidVarosoGomesCPNV/ICT-431-David-Gomes/issues>

## **3 Conception**

### **3.1 Analyse de l'environnement**

Matériel HW : Système d'exploitation : Windows 10 (X64) 1709

- CPU : Intel Core i7 6700 Cadencé à 3.40 GHz
- Carte mère : Dell Inc. 0X2MKR A00
- Mémoire : 16 Go DDR4
- GPU : Intel(R) HD Graphics 530

Logiciels / IDE : CLion

### **3.2 Détermination de l'arborescence du site et des rubriques**

Arborescence du menu de la bataille navale. C'est-à-dire, par quel chemin doit-on passer pour accéder aux différentes pages. (menu – sous-menu)

#### **Menu**

##### **1 Jouer**

###### **1.1 Lance le jeu**

##### **2 Options**

###### **2.1 Choix de la couleur**

##### **3 Aide**

###### **3.1 Affiche et détaille les règles du jeu**

##### **4 Quitter**

###### **4.1 Ferme le programme**



## 3.3 Maquette graphique

```

-----
      Bataille navale
-----

Le plateau de jeu se présente comme ceci :
  
```

	1	2	3	4	5	6	7	8	9
1	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-
6	-	-	-	-	-	-	-	-	-
7	-	-	-	-	-	-	-	-	-
8	-	-	-	-	-	-	-	-	-
9	-	-	-	-	-	-	-	-	-

Modèle Logique : Valeurs utilisées :

<pre>Raté !  Votre score actuel est de -3    1   2   3   4   5   6   7   8   9 1  R   -   -   -   -   -   -   -   - 2  -   -   -   -   -   -   -   -   - 3  -   -   -   -   -   -   -   -   - 4  -   -   -   -   -   -   -   -   - 5  -   -   -   -   -   -   -   -   - 6  -   -   -   -   -   -   -   -   - 7  -   -   -   -   -   -   -   -   - 8  -   -   -   -   -   -   -   -   - 9  -   -   -   -   -   -   -   -   -</pre>	Si l'utilisateur tire sur une case ou il n'y a pas de bateau, le programme affiche « Raté » et remplace l'endroit où le tir est tombé par un « R »
<pre>Touché !  Votre score actuel est de 0    1   2   3   4   5   6   7   8   9 1  -   -   -   X   -   -   -   -   - 2  -   -   -   -   -   -   -   -   - 3  -   -   -   -   -   -   -   -   - 4  -   -   -   -   -   -   -   -   - 5  -   -   -   -   -   -   -   -   - 6  -   -   -   -   -   -   -   -   - 7  -   -   -   -   -   -   -   -   - 8  -   -   -   -   -   -   -   -   - 9  -   -   -   -   -   -   -   -   -</pre>	Si l'utilisateur tire sur une case ou il y a un bateau, le programme affiche « Touché » et remplace l'endroit où le tir est tombé par un « X »

<pre> Touché !  Votre score actuel est de 9 Vous avez coulé le torpilleur !    1  2  3  4  5  6  7  8  9 1  R  -  R  X  X  -  -  -  - 2  -  -  -  -  -  -  -  -  - 3  -  -  -  -  -  -  -  -  - 4  -  -  -  -  -  -  -  -  - 5  -  -  -  -  -  -  -  -  - 6  -  -  -  -  -  -  -  -  - 7  -  -  -  -  -  -  -  -  - 8  -  -  -  -  -  -  -  -  - 9  -  -  -  -  -  -  -  -  - </pre>	<p>Si l'utilisateur tire sur une case ou il y a un bateau et qu'il s'agit de la dernière partie du bateau, le programme affiche « Touché » et « Vous avez coulé (le bateau en question) ». Les « X » restent affichées sur la grille.</p>
--	---

### 3.4 Conception du Code

#### Point technique 1 : Détection des bateaux coulés

Pour savoir si les bateaux sont coulés j'ai utilisé un système de points de vie.

```

#define TORPIEUR 2 // Constante pour la valeur du torpilleur sur la grille
#define SOUS_MARIN 3 // Constante pour la valeur du sous-marin sur la grille
#define CONTRE_TORPIEUR 4 // Constante pour la valeur du contre-torpilleur sur la grille
#define CROISEUR 5 // Constante pour la valeur du croiseur sur la grille
#define PORTE_AVION 6 // Constante pour la valeur du porte-avion

// Source des noms : https://www.regles-de-jeux.com/regle-de-la-bataille-navale/

// Vie des bateaux , pour savoir quand ils sont coulés
int vie_torpilleur = 0;
int vie_sous_marin = 0;
int vie_contre_torpilleur = 0;
int vie_croiseur = 0;
int vie_porte_avion = 0;

```

Ayant défini des constantes différentes pour chaque bateau, je leur donne à chacun un nombre de points de vie de 0 au début de la partie.

Ensuite je fais une condition pour chaque bateau comme ceci,

```

if (tableau_de_jeu[vertical][horizontal] == TORPILLEUR){
    vie_torpilleur++;

    if (vie_torpilleur == 2) {
        printf("Vous avez coulé le torpilleur !\n");
    }
}

```

Si l'utilisateur a tiré sur une case ou il y avait un bateau, dans cet exemple le torpilleur, il ajoute 1 à ces points de vie.

Les points de vie maximum étant équivalents au nombre de cases que les bateaux prennent sur la grille, dans ce cas le torpilleur étant le bateau avec le

moins de points de vie il ne prend que 2 cases, dans cette même logique le porte-avion aura donc 5 points de vie.

Donc une fois que on a tiré 2 fois sur les cases du torpilleur, ces points de vie sont à 2, le programme affiche alors que ce bateau est coulé.

#### Pseudo-code :

SI tableau\_de\_jeu à la même valeur que TORPIEUR  
Alors vie du torpilleur + 1

SI vie du torpilleur est égale à 2  
Afficher « Vous avez coulé le torpilleur ! »

#### Point technique 2 : Détection de fin de partie

*Pour savoir quand la partie est finie, j'utilise également les points de vie des bateaux  
Avec cette condition*

```
// Condition de victoire, si le nombre de bateaux coulés est égal à 5 (nb de
bateaux max dans la partie) c'est gagné
if(nbbateauxcoule==5) {

    // écrit dans le fichier score.txt que la partie est gagnée
    fprintf(txtscore, "Partie gagnée !! =D\n ");

    // écrit dans le fichier score.txt le score
    fprintf(txtscore, " %s ", "Voici votre score :");
    fprintf(txtscore, " %d " "points !", score);

    // Lignes de séparation
    fprintf(txtscore, "\n-----\n\n");

    // Texte de victoire
    printf("\n\nVotre score final est de %d !", score);
    printf("\nVous pouvez voir le score de tout les joueurs dans le fichier
Score.txt !");
    printf("\nVous avez coulé tout les bateaux bien joué !\n");
    system("pause");
    exit(0);
}
```

Une fois que tous les bateaux ont atteint leur nombre de pts de vie maximum, donc que tout est coulé, le jeu affiche la fin de la partie  
Il commence par écrire le score dans le fichier texte et affiche également le score sur le programme.

## Pseudo code :

SI le nombre de bateaux coulés est égale à 5

Alors écrire dans le fichier texte que la partie est gagnée

Écrire dans le fichier texte le score du joueur

Afficher le score final

Afficher que tous les bateaux ont été coulés

Mettre en pause le système, l'utilisateur doit appuyer sur une touche pour sortir du programme

## 4 Réalisation

### 4.1 Dossier de réalisation

ICT-431-David-Gomes

- *Code – Bataille Navale*
  - *.idea*
  - *Cmake-build-debug*
  - *CMakeLists.txt*
  - *Tableau.h*
  - *Main.c*
- *Documentation – Bataille Navale*
  - *Cahier des charges bataille navale V0.2.docx*
  - *Dossier de projet – Bataille Navale – David Gomes.doc*
  - *Lien GitHub.txt*
  - *Stratégie de test.xlsx*
- *README.md*

Matériel HW : Système d'exploitation : Windows 10 (X64) 1709

- CPU : Intel Core i7 6700 Cadencé à 3.40 GHz
- Carte mère : Dell Inc. 0X2MKR A00
- Mémoire : 16 Go DDR4
- GPU : Intel(R) HD Graphics 530

Logiciels / IDE : CLion

## 4.2 Description des tests effectués

Scénario	David Gomes	Esteban Giorgis	Dylan Berney
L'utilisateur lance le programme	OK	OK	OK
L'utilisateur rentre son prénom	OK	OK	OK
Le prénom de l'utilisateur est pris en compte	OK	OK	OK
L'utilisateur sélectionne jouer	OK	OK	OK
L'utilisateur sélectionne une grille disponible	KO	KO	KO
L'utilisateur tire en rentrant une colonne & ligne	OK	OK	OK
L'utilisateur sait qu'il a touché un bateau	OK	OK	OK
L'utilisateur sait qu'il a raté un bateau	OK	OK	OK
L'utilisateur sait qu'il a coulé un bateau	OK	OK	OK
L'utilisateur sélectionne quitter	OK	OK	OK
L'utilisateur sélectionne aide	OK	OK	OK
L'aide s'affiche	OK	OK	OK
L'utilisateur coule tout les bateaux	OK	OK	OK
L'utilisateur sait qu'il a gagné	OK	OK	OK
L'utilisateur sélectionne score	OK	OK	OK
Le score est dans le fichier score	OK	OK	OK

## 4.3 Erreurs restantes

- Si on met 2 noms (nom + prénom) le programme pense que c'est le default du switch et quitte / plante
- Si on met des valeurs à 2/3 chiffres pour tirer, plante également [Corrigé]

## 5 Mise en service

### 5.1 Rapport de mise en service

Lancer le Code.exe

### 5.2 Liste des documents fournis

- Documentations du projet
- Stratégie de test
- Lien GitHub
- Captures d'écran maquettes

## **6 Conclusions**

- *Objectifs atteints*
  - *Jouer à la bataille navale*
  - *Avoir un système de touché coulé raté pour tous les bateaux*
  - *Avoir un fichier score*
  - *Quelques options*
- *Non-atteints*
  - *Relancer la partie directement après une victoire / défaite*
  - *Remplacer les « X » [Touché] par un « C » [Coulé]*
- *Points positifs*
  - *Grand développement dans nos compétences en C*
- *Points négatifs*
  - *Un seul enseignant n'est vraiment pas assez pour ce genre de module*
  - *Assez perdu en début de module, on ne sait pas par où commencer mais une fois que on a la base du programme c'est plus simple pour continuer / améliorer*
- *Difficultés particulières*
  - *Devoir partir de rien pour faire un programme comme sa.*
- *Suites possibles pour le projet (évolutions & améliorations)*
  - *Amélioration de l'aspect graphique*
  - *Amélioration des options (Taille du plateau etc)*

## **7 Annexes**

### **7.1 Sources – Bibliographie**

- *Aides externes : Esteban, Dylan, Jessy*
- *Sites pour les dessins ASCII :*
  - <http://ascii.co.uk/art/ships>
  - <http://patorjk.com/software/taag/#p=testall&f=Graffiti&t=%0ABattleship>

## 7.2 Journal de travail

Date	Durée	Activité	Remarques
06.02.19	30min	Création des uses-cases	
08.02.19		Finalisation des uses-cases	
19.02	15min	Création du compte GitHub	
06.03	1h	Création du menu de base	
13.03	2h	Création et implémentation d'une grille fixe	
13.03	2h	Implémentation des fonctions toucher, rater, couler du petit bateau	
15.03	1h30	Remplissage du GitHub	
15.03	1h	Léger remplissage du dossier du projet	
19.03	15min	Changement des noms de bateaux avec les vrais noms d'une bataille navale	
19.03	1h	Remplissage du dossier du projet	
21.03	3h	Amélioration des if pour les touché raté, implémentation des points de vie des bateaux	
22.03	2h	Ajout des if touché, raté et coulé de tous les bateaux, tout fonctionnel	
22.03	30min	Amélioration du menu de jeu, ajout des règles de jeu	
27.03	1h	Travail en groupe méthode agile	
27.03	1h	Remplissage du dossier du projet, Stratégie de test, maquette	
29.03	2h	Finalisation de la condition gagner qui quitte le jeu	
29.03	1h30	Remplissage de la doc du projet	
29.03	30min	Tentative de codage pour le choix des cartes / aléatoires	
04.04	3h	Implémentation du fichier score externe	Difficulté à faire apparaitre le score dans le fichier contrairement au pseudo
04.04	1h	Remplissage de la doc	
04.04	25min	Ajout de l'option couleur	

05.04	1h	Ajout de beaucoup de commentaires et modifications légères pour le score.	
05.04	45min	Ajout de sécurités de tir & autres	Pour ne pas pouvoir mettre +de 10 dans la variable ligne & colonne
05.04	20min	Ajout d'un dessin ASCII au menu	
05.04	1h30	Remplissage de la doc	

### **Journal de bord**

Date	Durée	Activité	Remarques
06.02.19	30min	Théorie des uses-cases	
08.02.19	45min	Théorie SMART	
19.02.19	90min	Démo GitHub	
03.04.19	60min	Démo GitHub Desktop	M. Karel car replacement

### **7.3 Archives du projet**

<https://github.com/DavidVarosoGomesCPNV/ICT-431-David-Gomes>