

IUT de Valence – Université Grenoble Alpes

# Manuel Technique

SAÉ 6.1

BOURGUIGNEAU Ethan, CHAVANEL Yohann, GRAILLE Théo,  
SAPET Alan  
21/06/2024

1. Fonctionnalités réalisées .....	2
1.1. Serveur Thread (Ethan) : .....	Erreur ! Signet non défini.
1.2. Serveur UI (Ethan) : .....	Erreur ! Signet non défini.
1.3. Client Handler (Ethan) : .....	Erreur ! Signet non défini.
1.4. App Defines (Ethan) : .....	Erreur ! Signet non défini.
1.5. Types (Ethan) : .....	Erreur ! Signet non défini.
1.6. Game UI (Yohann) : .....	Erreur ! Signet non défini.
1.6.1 Bevy .....	Erreur ! Signet non défini.
1.6.2 Raylib & Rapier2D .....	Erreur ! Signet non défini.
1.6.3 Egui & Rapier2D .....	Erreur ! Signet non défini.
1.7. Physique du jeu & concepts d'objets (Alan & Théo) : .....	Erreur ! Signet non défini.
1.7.1 Adaptation au choix de librairies (Alan) .....	Erreur ! Signet non défini.
1.7.2 Joueurs et bots (Alan) .....	Erreur ! Signet non défini.
1.7.3 Déplacements du vaisseau (Alan) .....	Erreur ! Signet non défini.
1.7.5 Gestion des limites du terrain (Théo) .....	Erreur ! Signet non défini.
1.7.6 Notions de forces de propulsion et de collision (Théo) ...	Erreur ! Signet non défini.
2. Structuration du projet .....	Erreur ! Signet non défini.
2.1. Organisation du code .....	Erreur ! Signet non défini.
2.2. Choix de conception : .....	Erreur ! Signet non défini.
3. Les difficultés rencontrées .....	Erreur ! Signet non défini.
4. Répartition des tâches .....	Erreur ! Signet non défini.

## 1. Mise en place du serveur

Le serveur de jeu peut être lancé sur tous les systèmes d'exploitation (Windows 10/11, MacOS, Linux) à condition de le compiler sur la machine. Pour cela, il faut que le langage de programmation RUST soit installé

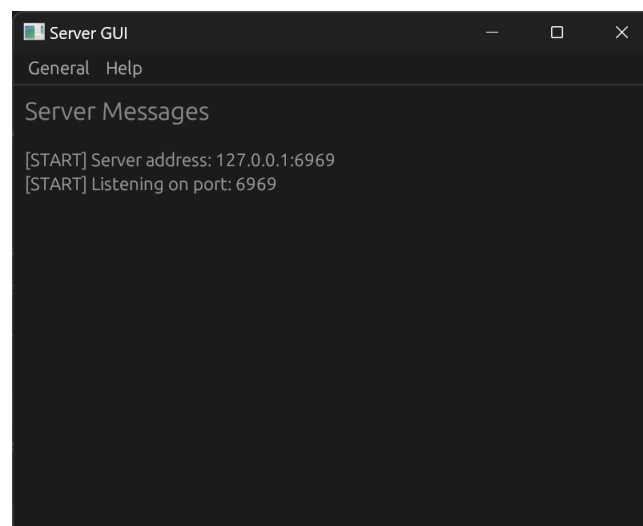
Téléchargement du code :

<https://github.com/VOYOApp/SAE-6.1/>

La compilation du code s'effectue à l'aide de la commande :

```
carbo build
```

Une fois lancé, le serveur est prêt à recevoir des joueurs qui peuvent se connecter à l'adresse IP indiquée.



## 2. Commandes du serveur

L'envoi de commandes pour les utilisateurs se fait de la façon suivante.

```
COMMANDE1=VALEURE1#COMMANDE2=VALEURE2
```

COMMANDE correspond aux commandes ci-dessous.

Le signe = est le séparateur de la commande et de la valeur.

VALEURE est la valeur associée à la commande.

# Permet de chaîner les commandes

La liste de chaque commande disponible est définie dans le tableau suivant

Nom de commande	Description	Valeur nécessaire	Type de valeur
NAME	Définir le nom de l'utilisateur	Oui	Chaine de caractères
COL	Définir la couleur de l'utilisateur	Oui	3 nombres correspondant aux canaux RGB
EXIT	Déconnecte l'utilisateur	Non	
LIVE	Gare l'utilisateur connecté	Non	
MSG	Envoie un message	Oui	Chaine de caractères
CBOT	Renvoie l'entité la plus proche	Non	
CPROJ	Renvoie le projectile le plus proche	Non	
NBOT	Renvoie les détails d'un utilisateur	Oui	Chaine de caractères correspondant au username
NLIST	Renvoie la liste des joueurs	Non	
ORIENT	Renvoie l'orientation du joueur	Non	
USRMSG	Renvoie les messages d'un joueur	Oui	Chaine de caractères correspondant au username
EMPTY	Renvoie une réponse vide	Non	

## 3. Modes de jeux

Il existe uniquement 1 seul mode de jeu, *kill to survive*.