## Cahier des Charges: "OPERATION: we have a problem."

### I. Présentation:

"OPERATION: we have a problem." (abrégé en OWHAP) est un prototype de jeu vidéo fortement inspiré des jeux vidéo Faster Than Light et Among Us. On y dirige un robot immortel dont le travail est d'assurer la maintenance d'un vaisseau de colon humain en cryostase, jusqu'à ce qu'il arrive à destination, en survivant aux nombreux événements qui mettent en danger le vaisseau.

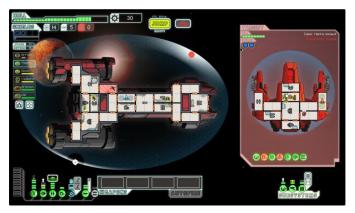




Figure 1: Faster Than Light (FTL)

Figure 2: Among Us

Il est réalisé par les étudiants de Lyon1 en L2 d'info : Pierre MANOEL (p2101842) et Oumayma JAMALI (p1810913) en projet pour l'UE LIFAPCDA. Nous disposons d'environ 2 mois pour la réalisation et il sera codé en C++, en utilisant les bibliothèques SDL2 et QT.

### II. Description:

OWHAP se présente comme un jeu en vue de dessus en 2D. On y dirige un personnage qui peut se déplacer (avec les touches ZQSD) dans les différentes salles d'un vaisseau spatial. Il peut y réaliser une multitude de tâches liées à la salle dans laquelle il se trouve (exemple : réparation du moteur dans la salle des moteurs). L'interaction se fait via une touche du clavier et déclenche une action à réaliser sous la forme d'un mini-jeu classique (exemple : Space Invader, Pong...). Ces tâches à sont liées à une série d'événements qui arrivent à un intervalle régulier (exemples : avarie moteur, attaque pirate ...) dont on peut suivre la progression avec une barre. Le but du jeu est d'arriver à la dernière étape en survivant aux différents événements sans mourir/exploser/avant la fin du temps imparti.

En clair, les fonctionnalités seraient :

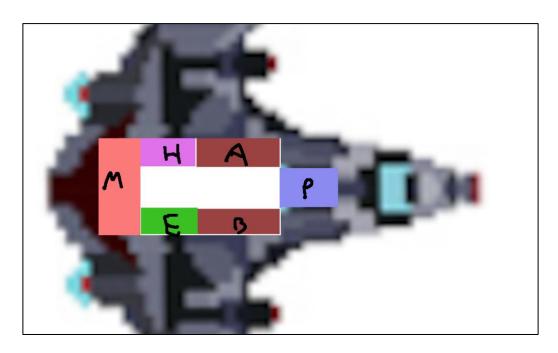
- Décors spatiaux (inspirés de FTL)
- Vaisseau en vue de dessus en 2D avec 6 salles (Moteur, Humains, Energie, Arme, Bouclier et Pilotage, toutes visibles sur le même plan) ainsi que des couloirs pour naviguer.
- Personnage en 2D (inspiré de Among us).
- Le personnage peut se déplacer dans les salles du vaisseau avec les touches ZQSD.

- Événements qui provoquent des possibilités d'interaction avec des salles.
- Le personnage peut interagir avec les objets de la salle où il se trouve (réparation, armement...) avec une touche (probablement I).
- Interaction sous la forme de mini-jeu basique.
- Déclenchement des évènements à intervalles de temps périodiques, que l'on peut suivre grâce à une barre de progression. Fin du jeu lorsque la barre atteint la fin.
- Menu de pause qui peut s'ouvrir avec une touche qui permet de : reprendre, recommencer ou quitter le jeu.

#### Contraintes:

Nous disposons seulement de 8 semaines de développement et de 2 développeurs (non professionnels en plus (et non rémunérés)). Le jeu nécessite un clavier également.

#### Croquis:



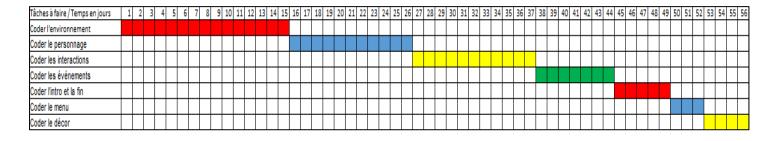
M=moteur, H=humain, E=énergie, A=armes, B=bouclier, P=pilotage.

## III. Déroulement du projet

- Coder l'environnement et le vaisseau (2 semaines)
  - -Salles sous forme de blocs.
  - -Création de l'image du vaisseau en vue de dessus.
  - -Livrable : on cherche à obtenir une image de vaisseau et des salles (en mode texte/en mode normal)
- Coder le personnage (10 jours)
  - -Déplacement avec ZQSD.
  - -Création de l'image du personnage.

- -Livrable: on cherche à obtenir un personnage qui peut bouger dans le vaisseau (mode texte/normal).
- Coder les interactions (10 jours) :
  - -Lien entre Personnage et Environnement (salles)
  - -Spécification chaque interaction dédiée à chaque salle
  - -Livrable : on veut que le personnage puisse interagir complètement avec les salles (texte/normal)
- Coder les événements (7 jours) :
  - -Barre de progression temporelle
  - -Lien Événement et Interaction
  - -Création d'actions/mini-jeux pour les interactions entre le personnage et les objets d'une salle.
  - -Livrable : on veut une barre de progression efficace et des interactions claires et réalisables.
- Coder l'intro et la fin (5 jours) :
  - -Création d'un tuto pour démarrer le jeu
  - -Création de la notion d'objectif et de fin du jeu
  - -Création de la possibilité de perdre (quand l'un des événements n'est pas résolu dans les temps)
  - -Livrable : on veut un début de jeu clair, une fin du jeu et la possibilité de perdre
- Coder le menu de pause (3 jours) :
  - -Création d'un menu de pause du jeu qui peut : reprendre, recommencer ou quitter le jeu.
  - -Livrable : on veut un menu de pause fonctionnel.
- Coder le décor (4 jours) :
  - -Amélioration de l'aspect esthétique via des images et des sprites dédiés.
  - -Amélioration visuelle de la fin, des événements, de la barre de progression avec plus de détails.

### IV. Diagramme de Gantt:



# V. Diagramme des classes :

