

JEDIMAC TOWER DEFENSE

MATHILDE DUMOULIN ✨ JULIEN HAUDEGOND

I. ARCHITECTURE LOGICIELLE

```
.
├── bin
├── data
├── doc
├── fonts
├── images
│   └── sprites
├── include
│   ├── animations
│   ├── app
│   ├── constructions
│   ├── gui
│   ├── monsters
│   ├── nodes
│   └── structures
├── lib
│   ├── include
│   └── lib
├── obj
├── sounds
└── src
    ├── animations
    ├── app
    ├── constructions
    ├── gui
    ├── monsters
    └── nodes
```

ROOT

```
├── animations
│   ├── end_game.c
│   ├── explosion.c
│   └── intro.c
├── app
│   ├── game.c
│   └── menu.c
├── constructions
│   ├── building.c
│   ├── building_list.c
│   ├── tower.c
│   └── tower_list.c
├── gui
│   ├── graphic.c
│   ├── image.c
│   ├── map_draw.c
│   ├── sprite.c
│   ├── text.c
│   └── window.c
├── main.c
├── monsters
│   ├── monster.c
│   └── wave.c
├── nodes
│   ├── itd.c
│   ├── link.c
│   └── node.c
└── save.c
```

SOURCES

```
├── animations
│   ├── end_game.h
│   ├── explosion.h
│   └── intro.h
├── app
│   ├── game.h
│   └── menu.h
├── const.h
├── constructions
│   ├── building.h
│   ├── building_list.h
│   ├── tower.h
│   └── tower_list.h
├── gui
│   ├── graphic.h
│   ├── image.h
│   ├── map_draw.h
│   ├── sprite.h
│   ├── text.h
│   └── window.h
├── monsters
│   ├── monster.h
│   └── wave.h
├── nodes
│   ├── itd.h
│   ├── link.h
│   └── node.h
├── save.h
└── structures
    ├── str_building.h
    ├── str_building_list.h
    ├── str_end_game.h
    ├── str_explosion.h
    ├── str_image.h
    ├── str_itd.h
    ├── str_link.h
    ├── str_monster.h
    ├── str_node.h
    ├── str_pixel.h
    ├── str_sprite.h
    ├── str_text.h
    ├── str_tower.h
    ├── str_tower_list.h
    └── str_wave.h
```

HEADERS

II. FONCTIONNALITÉS SUPPLÉMENTAIRES

INTRODUCTION

PAUSE

MUSIQUE

SAUVEGARDE AUTOMATIQUE

CONSTRUCTIBILITÉ D'UNE CASE

ÉCRANS DE FIN

**CHOIX DE LA CARTE DEPUIS LE
MENU**

EXPLOSIONS



II. FONCTIONNALITÉS SUPPLÉMENTAIRES

SAUVEGARDE AUTOMATIQUE

```
int saveGame(const char* itdPath, int money, int wave, TowerList* tl, BuildingList* bl, int status) {
    const char* fichier = "data/save.txt";

    //Open a file
    FILE* F = fopen(fichier, "w");
    if (!F) {
        fprintf(stderr, "Unable to open the file.\n");
        exit(EXIT_FAILURE);
    }

    //if the player did not finish the game
    if (status != 0 && status != 1) {

        fprintf(F, "# MAP\n");
        fprintf(F, "%s\n", itdPath);

        fprintf(F, "# MONEY\n");
        fprintf(F, "%d\n", money);

        fprintf(F, "# WAVE\n");
        fprintf(F, "%d\n", wave);

        fprintf(F, "# TOWERS & BUILDINGS\n");
        if (tl->tower) {
            TowerList* tmp = tl;
            fprintf(F, "%d %d %d %d\n", 1, tmp->tower->type, tmp->tower->win_x, tmp->tower->win_y);

            while (tmp->nextTower) {
                tmp = tmp->nextTower;
                if (tmp->tower) {
                    fprintf(F, "%d %d %d %d\n", 1, tmp->tower->type, tmp->tower->win_x, tmp->tower->win_y);
                }
            }
        }
        if (bl->build) {
            BuildingList* tmp = bl;
            fprintf(F, "%d %d %d %d\n", 2, tmp->build->type, tmp->build->win_x, tmp->build->win_y);

            while (tmp->nextBuild) {
                tmp = tmp->nextBuild;
                if (tmp->build) {
                    fprintf(F, "%d %d %d %d\n", 2, tmp->build->type, tmp->build->win_x, tmp->build->win_y);
                }
            }
        }
    }

    //else, the file is overwrite with empty data.
    fclose(F);
    return EXIT_SUCCESS;
}
```

MAP

data/Map_02.itd

MONEY

0

WAVE

2

TOWERS & BUILDINGS

1 1 335 396

1 3 396 335

2 2 396 396

2 2 396 457

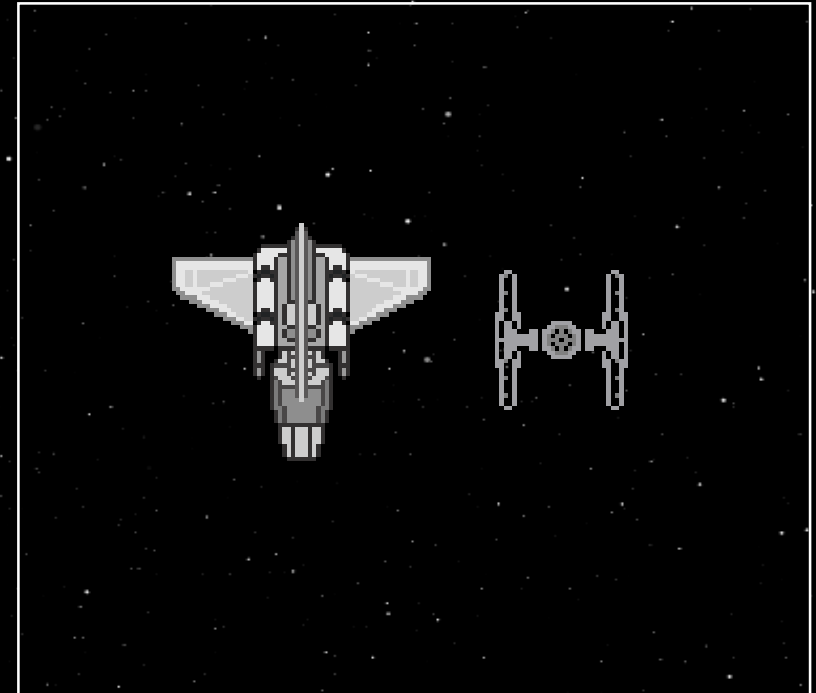
2 2 335 457

III. IDENTITÉ VISUELLE

CHARTE GRAPHIQUE DE STAR WARS : ÉCRITURE ET COULEURS



REBEL ALLIANCE



**GALACTIC EMPIRE
(BAD GUYS)**

III. IDENTITÉ VISUELLE



**LAMBDA CLASS
IMPERIAL SHUTTLE**

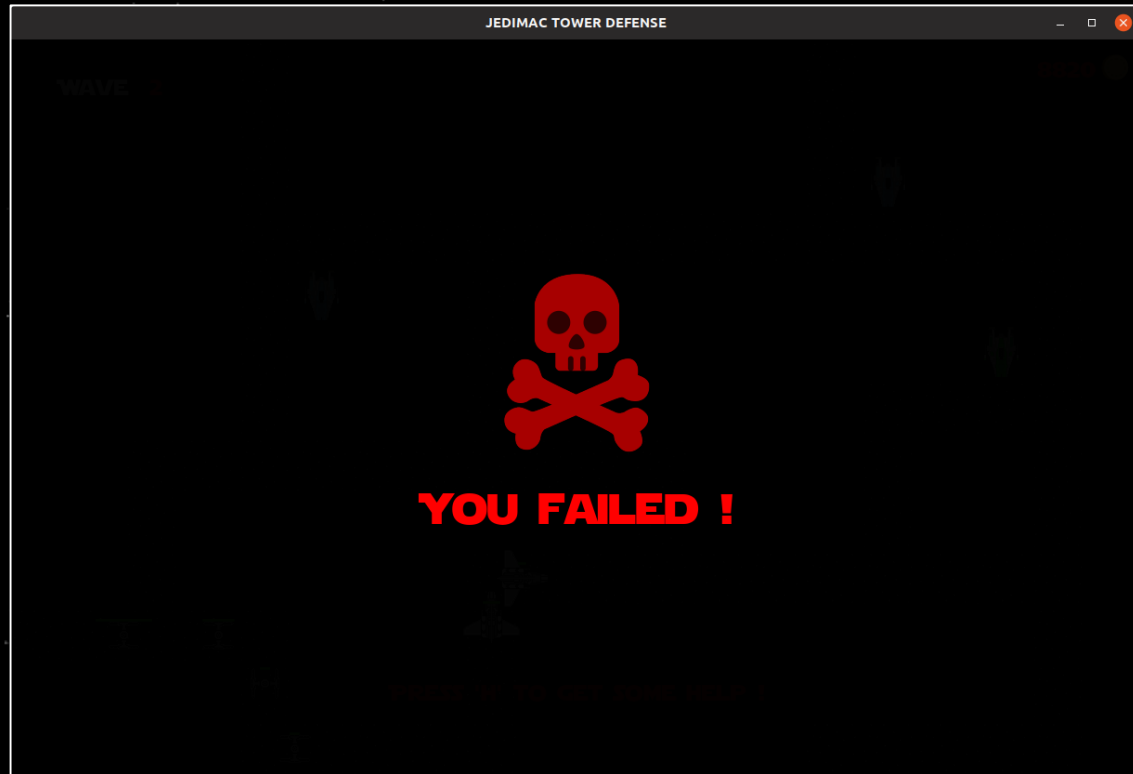


TIE-FIGHTER



A-WING

III. IDENTITÉ VISUELLE



LES 2 ÉCRANS DE FIN POSSIBLES

CONCLUSION

EXPÉRIENCE TRÈS ENRICHISANTE

APPRENDRE À STRUCTURER UN PROJET

APPRENDRE À SÉPARER LES TÂCHES ENTRE NOUS

HÂTE DE FAIRE LE PROJET DE 2^{ÈME} ANNÉE !