

# Project proposal

## Roar of steel

### Team members:

MengQi He	100908507
Yaji Hu	101012215
Philip Yuen	100979919

### Game idea:

This game is a (relatively) realistic naval combat game. In this game, the player is an admiral who commands a forward base and needs to build his fleet to protect himself and defeat the enemy.

### Game settings:

The game wants to focus on combat and design ships, so we decide to simplify the resource aspect. There are Iron (mostly used for ship building), manpower (The sailor on ship), and experience (for research). Three resource has different way to produce (Or collect). Iron is produced by building factory at base. Manpower comes from your home land and increase as you control larger area, you can also rescue sailors from lifeboat after sunk a ship. Experience comes from combat, both deal damage and receiving damage gives you experience, destroy enemy ship gives more experience.

What we consider to be the “meat” of this game is the design part. Player need to allocate space of a ship body to different components. And there are many trade-offs in the design. For example, if you make larger cannon and engine, then the cost for armor will increase as you have to cover larger area. And if you increase armor, the ship became heavier and thus slower. Player can save their design and build it in the shipyard at base, and use it to defeat the enemy later.

**Game goal:** Destroy enemy base while protecting your own.

### Game object:

- Buildings: Base, Shipyard, Iron factory.
- Units: Because we have a built-in design part, there isn't a fixed number of units in game. But we planned to have 6 ship bodies (destroyer, light cruiser, heavy cruiser, battlecruiser, battleship, dreadnought), torpedo (consider it as a very small ship with powerful melee range weapon), a lifeboat (hardly consider it as a unit because it doesn't move or do anything), and a collector ship. We will provide some pre-designed ship for new players and AI, but a ship designed by player may fits his play style better.
  - A ship object can be broken down into these components:
    - NavMeshAgent - handles path finding
    - ShipController - handles ship movement using raycasting
    - ShipMotor - moves the ship
    - Ship - main ship component

- And we have planned to add a giant robot as enemy ultimate unit. (and to meet the project requirement.)

**Game Map:** The map is separated into blocks, each block has 4 possible status

1. Sea. Allows movement, doesn't block sight
2. Land. Can't move through, doesn't block sight
3. Hill. Can't move through, block sight.
4. Mountain. Can't move through, block sight and shell.

Created using shaders.

### **Game systems:**

1. Zone control: There will be several "key-points" on the map, you can capture it by send ship close to the point and claim control of that zone. This will increase your manpower income, and allows you to recycle the shipwreck in that zone.
2. Shipwreck recycle: Every ship sunk will transfer part of its iron cost to the zone pool, the player controls that zone can send collector ship to the key-points to collect the iron.
3. Refit: As your technology develops, the old ship doesn't get the new technology bonus, and this is when you need to send them to refit at shipyard. Refitting will cost a little iron and time, but will make your ship more powerful (usually).
4. Realistic firing: The ship can only fire with all turrets when firing at broadsides, and the cannon has certain hit chance related to the speed of both ships, the size of ship and cannon, the distance between two ships and so on. And there will be random accidents happen during combat.
5. Line-Of-Sight system: The ship in this game has a line-of-sight feature. So if there is a hill in between the ship and enemy, it won't be able to detect and fire at the target, unless there is another friendly ship can see the enemy ship. In this case we call it "indirect fire" and will receive penalty at accuracy.

### **Game structure:**

Player class:

- Holds information about the player and his fleet. Including amount of resources, list of ships and buildings, list of designs, list of researches.
- Technology will be stored as an integer array in Player class, each integer indicates the level of one technology.

Pseudo code:

Variables:

```
Int Player_ID;
Player[] Friends;
Player[] Enemies;
Float Iron, man_power, experience;
Ship_design[] designs;
```

```
Ship[] Ships;  
Int[] Researches;
```

Functions:

```
Update();  
UpdateShip();  
BuildShip();  
BuildBuilding();
```

Game class:

- The main control class for the entire game. Enemy AI are controlled in here.

Pseudo Code:

Variables:

```
Player[] Players;  
Int Game_Status;//Controls the status of game (i.e. On menu, paused, in game, or  
ended)
```

Functions:

```
Mainloop();  
AI()
```

Ship class:

- Holds all the information about a ship. Including ship size, position, player, number of cannons, cannon damage, armor piercing value, torpedo, hitpoints, armor, speed, etc.
- Has method for firing, moving, and sunk. Ship AIs are in there too

Pseudo Code: (Don't show the get/set methods as it looks duplicated,)

Variables:

```
Player owner;  
Int Hitpoints,Max_HitPoints;  
Int Main_cal,Sub_cal;  
Int Main_turret,Sub_turret;  
Int Armor,Max_speed;  
Float MainRange,SubRange;  
Float Cur_speed;  
Float[] position;  
Float direction;  
Bool is_torpedo;
```

Functions:

```
Attack(Ship s);  
Move();// Including path finding code  
Sink();  
Update();// Include code for target searching.
```

Ship Design Class:

- The ship blueprint is stored as a ship design object. There it stores the all design parameters of a ship. (pretty much what you can see in the ship design menu preview in the end of the proposal)

UI Class: (**Chose to implement multiple UI classes for specific objects**)

- The user interface for the game, including start menu, ship design menu, research menu and display some information like resource and mini map.

Functions:

```
Reset();//Sort like initialize game,. May take an integer as difficulty settings
StartGame();
OpenResearchMenu();
OpenDesignMenu();
OpenGameMenu();
```

Map Class: (**Chose not to implement**)

- The data related to map, like strategy zones on map and terrains on map. But the collisions are processed by unity engine.

Pseudo Code:

Variables:

```
Int[][] mapTiles;
Float[X][2] Zones;
Float[] Zone_resources;
Float[][] Waypoints;//Some possible variables help for path finding
```

Building(not movable) Class:

- The buildings built on map. Like shipyard and factory. Life boats are also considered as a building for simpler code.

Variables:

```
Int Type;
Float[] position;
```

Functions:

```
Update();//Use a switch or if-statement to do different work based on different type;
```

Stats Class:

- Variables

```
Health
Armor
Speed
```

- Functions:

```
TakeDamage()
Die() - handles when health <= 0 or when the player chooses to destroy the ship
```

**Additional Features:**

1. Main menu/Game over screen
  - Buttons with animations
2. Sound
  - Explosion sound plays when a ship/building is destroyed
  - Sound plays when hovering over a button in the main menu/game over screen
    - Also plays when a button is clicked
  - Game over announcement

**Missing features:**

- Basic AI is missing
  - Enemy is present but does nothing

Ship Design menu preview (the values may get rebalanced)

### Background was changed in final submission

Ship Designer

Ship nameLightingShip classDestroyer

Fire power attribute

Main-gun caliber- 5 +

Main-gun turrets- 3 +

Sub-gun caliber- 0 +

Sub-gun turrets- 0 +

Torpedo tubes- 3 +

Firepower component weight

270 tons

Defensive attribute

Armor rating- 3 +

Speed(knots)- 32 +

Torpedo bulge

Anti-Air (Not available)

Defensive component weight

1451 tons

A Lighting class Destroyer

Equiped with 5 inch Cannon

3 torpedo tube(s)

Armor enough to stop 3 inch shell

Max speed 32 Knots

Total hitpoints: 19

Iron Cost:135

Construction Time: 10.35

Total weight 1921

Max Capacity 2000

Not used weight capacity  
will be used as bulkhead to  
increase the hitpoint of this  
design

Cancel

Confirm Design

Ship Designer

Ship name678abcdefghijklmnShip classDestroyer

Fire power attribute

Main-gun caliber- 5 +

Main-gun turrets- 3 +

Sub-gun caliber- 0 +

Sub-gun turrets- 0 +

Torpedo tubes- 3 +

Firepower component weight

270 tons

Defensive attribute

Armor rating- 4 +

Speed(knots)- 32 +

Torpedo bulge

Anti-Air (Not available)

Defensive component weight

1631 tons

A 12345678abcdefghijklmn class  
Destroyer

Equiped with 5 inch Cannon

3 torpedo tube(s)

Armor enough to stop 4 inch shell

Max speed 32 Knots

Total hitpoints: 10

Iron Cost:135

Construction Time: 11.35

Total weight 2101

Max Capacity 2000

Not used weight capacity  
will be used as bulkhead to  
increase the hitpoint of this  
design

Cancel

Confirm Design

Ship Designer

Ship nameRodneyShip classDreadnought

Fire power attribute

Main-gun caliber- 18 +

Main-gun turrets- 5 +

Sub-gun caliber- 6 +

Sub-gun turrets- 6 +

Torpedo tubes- 0 +

Firepower component weight

12950 tons

Defensive attribute

Armor rating- 18 +

Speed(knots)- 23 +

Torpedo bulge

Anti-Air (Not available)

Defensive component weight

26746 tons

A Rodney class Dreadnought

Equiped with 18 inch Cannon

0 torpedo tube(s)

Armor enough to stop 18 inch  
shell

Max speed 23 Knots

Total hitpoints: 372

Iron Cost:18611

Construction Time: 434.11

Total weight 49696

Max Capacity 50000

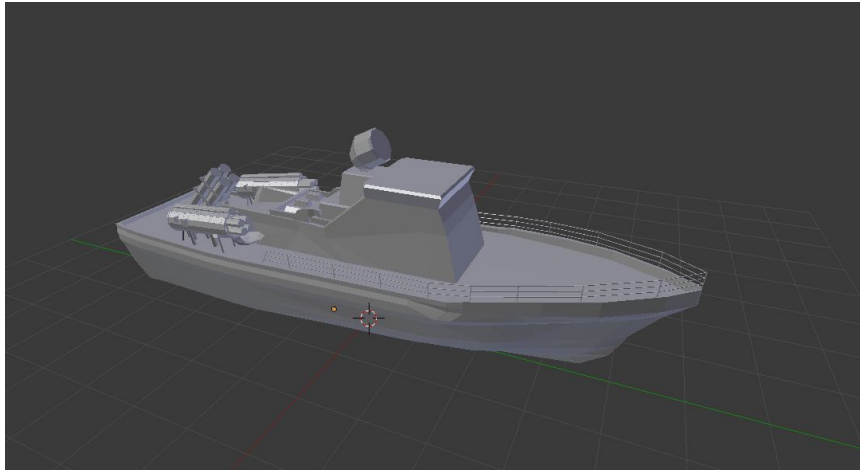
Not used weight capacity  
will be used as bulkhead to  
increase the hitpoint of this  
design

Cancel

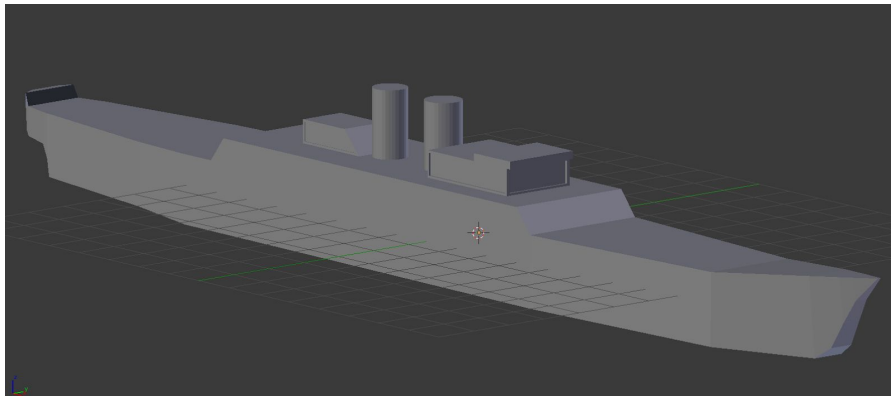
Confirm Design

## SHIP MODELS (Models created using blender)

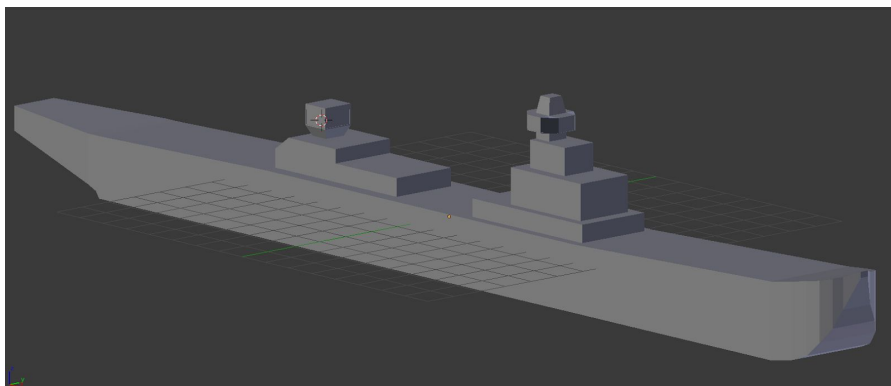
Destroyer



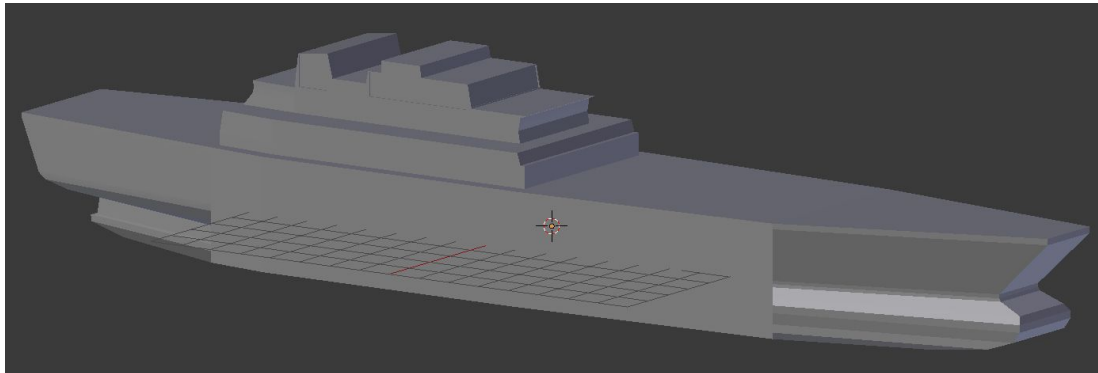
Light Cruiser



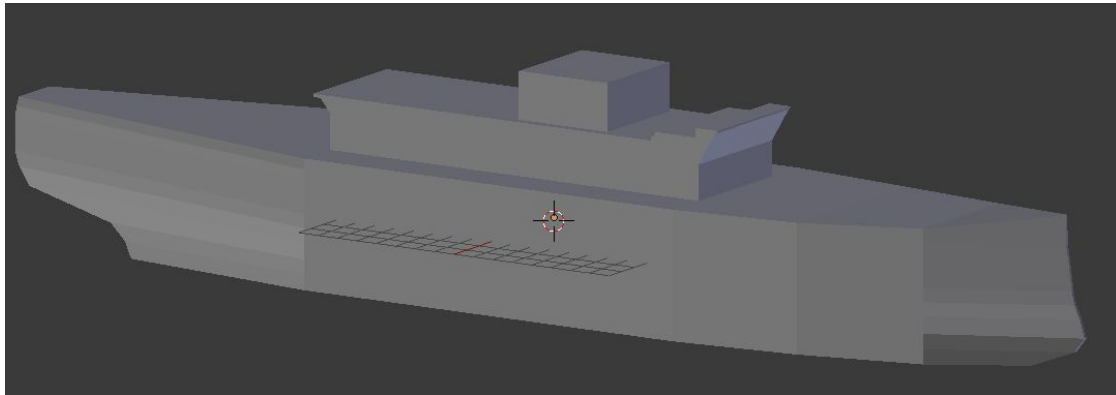
Heavy Cruiser



Battlecruiser



Battleship



Dreadnought

