

# オブジェクト指向プログラミングレポート課題

5422007 千本木悠 5422009 本江拓海 5422020 池田悠星

2024 年 8 月 3 日

## 1 システムの概要

プレイヤーから発射される銃弾はキーボードの Z を押すことで発射される。プレイヤーはマウスで動かすことが可能である。敵に銃弾が当たると敵の HP バーが減少する。HP バーが 0 になると敵は消滅する。敵の球がプレイヤーに当たるとプレイヤーの残機が減少する。敵の `zako` は動くようにした。

## 2 クラス設計の指針

`GameObject` クラスですべてのオブジェクトを管理することにし、`Character`、`HPBar`、`Bullet` の 3 つに分けた。`Character` クラスでは `Enemy`、`Player` の 2 つに分け `Enemy` クラスでは `Zako`、`Boss` といったクラス設計を行った。

## 3 クラス設計の詳細

`GameObject` クラスはゲーム内のオブジェクトを表す抽象クラスで、位置、サイズ、速度、色などの属性を持つ。このクラスはオブジェクトの移動や状態の管理を行うメソッドを提供する。

`Bullet` クラスは `GameObject` を継承し、ゲーム内でキャラクターが発射する弾丸を表す。弾丸がどのキャラクターに所有され、どのキャラクターにダメージを与えるかを示す属性を持ち、弾丸の攻撃力も設定できる。`task` メソッドで移動を行い、一定時間後に弾丸をゲームから取り除く。

`HPBar` クラスは `GameObject` を継承し、ゲーム内でキャラクターやオブジェクトの体力を視覚的に表示するための UI 要素を表す。現在の HP と最大 HP を保持し、`draw` メソッドで HP の割合に応じた長さの矩形を描画する。

`TextLabel` クラスは、`GameObject` を継承し、ゲーム内のテキスト表示を扱う UI 要素を表す。文字列、フォントスタイル、文字列の左オフセットを管理し、`draw` メソッドで指定された位置にテキストを描画する。

`Enemy` クラスは、`Character` クラスを継承し、ゲーム内でプレイヤーと対峙する敵キャラクター

を表す。HP を示す health 属性を持ち、collided メソッドで弾丸との衝突を処理、弾丸の攻撃力に応じて敵の体力を減少させる。弾丸が敵に命中した場合、その弾丸をゲームから削除する。

Player クラスは、Character クラスを継承し、ゲーム内のプレイヤーキャラクターを表す。プレイヤーのライフを示す remaining 属性を持ち、shoot メソッドで敵に向けて弾丸を発射する。collided メソッドで弾丸や敵キャラクターとの衝突を処理し、ライフを減少させる。また、task メソッドで一定間隔ごとに弾丸を発射し、ライフがゼロになったときの処理を管理する。

Zako クラスは Enemy クラスを継承し、ゲーム内の雑魚敵キャラクターを表す。HPBar オブジェクトを持ち、敵の現在の体力を視覚化する。shoot メソッドでプレイヤーに向けて弾を発射し、task メソッドで一定間隔での射撃と、時間経過による移動を行う。draw メソッドは敵の姿と HP バーを描画し、HP がゼロになった際に自身と HP バーをゲームから削除する。

Boss クラスは Enemy クラスを継承し、ゲーム内のボスキャラクターを表す。このクラスはボスが特定のパターンで弾丸を発射するメカニズムを提供する。HPBar オブジェクトを使ってボスの体力を表示し、shoot メソッドでは同心円状に弾丸を発射する。task メソッドでは、特定の間隔で射撃を行い、HP がゼロになるとボスと HP バーをゲームから削除する。draw メソッドはボスの姿を描画する。

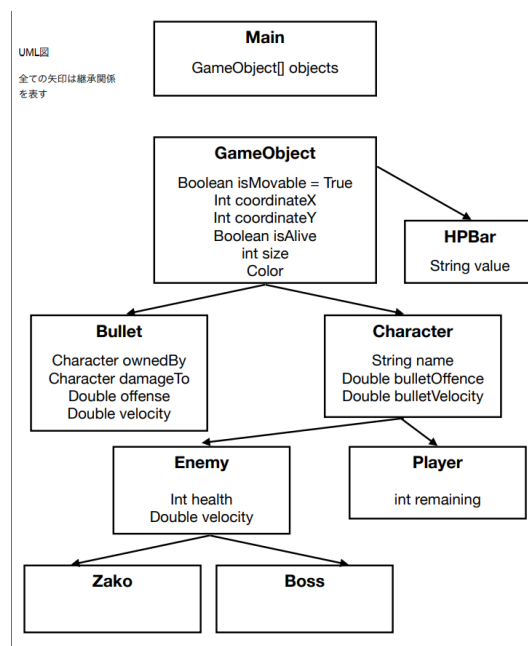


図 1 クラス設計図

## 4 実行結果

図 2 の画面からスタートする。図 3 のように球が発射されダメージを与えた分 HP バーが減少する。図 4 はすべての敵を倒した状態である。

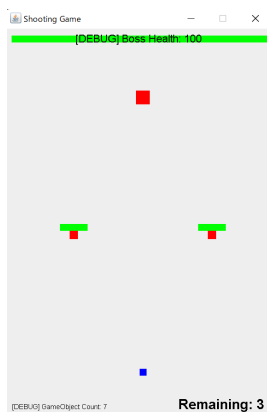


図 2 実行画面 1

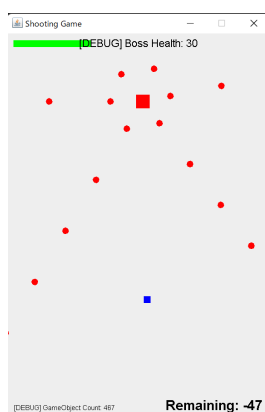


図 3 実行画面 2

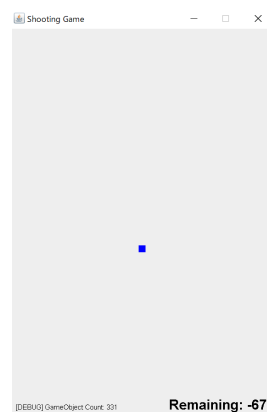


図 4 実行画面 3

## 5 クラス設計に対する考察

今回のクラス設計では保守性，可読性ともに高まったといえる．GameObject クラスから，Character クラス，HPBar クラス，Bullet クラスに分けたことでそれぞれの管理がしやすくなりコードの可読性も高まった．また，新たに中ボスをつくる際も Enemy クラスがあることでコードの追加が容易である．

## 6 ソースコード

コードを GitHub に掲載した．URL を以下に示す．

[https://github.com/Piertotum-Locomotor/kthr\\_shooting](https://github.com/Piertotum-Locomotor/kthr_shooting)

## 7 担当箇所

表 1 執筆者らの寄与

内容	名前
レポート本文作成	千本木，本江
クラス設計	千本木，池田
AI による雛形作成	池田
プログラミング	千本木，本江，池田
LaTeX 化	本江