Для корабля игрока применен шаблон синглтон. Идея состоит в том чтобы создать объект корабль игрока только один раз. Последующие вызовы возвращают ссылку на ранее созданный объект.

/// <summary>Поле хранит единственный экземпляр </summary>

private static PlayerShip \_instance;

/// <summary>Статический метод возвращает экземпляр</summary>

public static PlayerShip GetShip()

{

if (\_instance == null)

{

PlayerShip PlayerShip = new PlayerShip();

\_instance = PlayerShip;

return PlayerShip;

}

return \_instance;

}

//Конструктор сделан приватным для применения синглтона

private PlayerShip()

{

//нельзя напрямую создать обьект применен синглтон

//PlayerShip = new PlayerShip();

PlayerShip = PlayerShip.GetShip();

Для кораблей противника был применен патерн стратегия для выбора траектории движения.

//общий интерфейс для реализации движения

public interface IUpdate

{

Thickness Update(Thickness Position);

}

public class EnemyShipMk1 : GameObject

{

//интерфейсная ссылка для использования стратегии

private IUpdate UpdateStrategy;

public override Thickness Update(Thickness p)

{

//используем ту стратегию на которую есть ссылка

Grid.Margin = UpdateStrategy.Update(Grid.Margin);