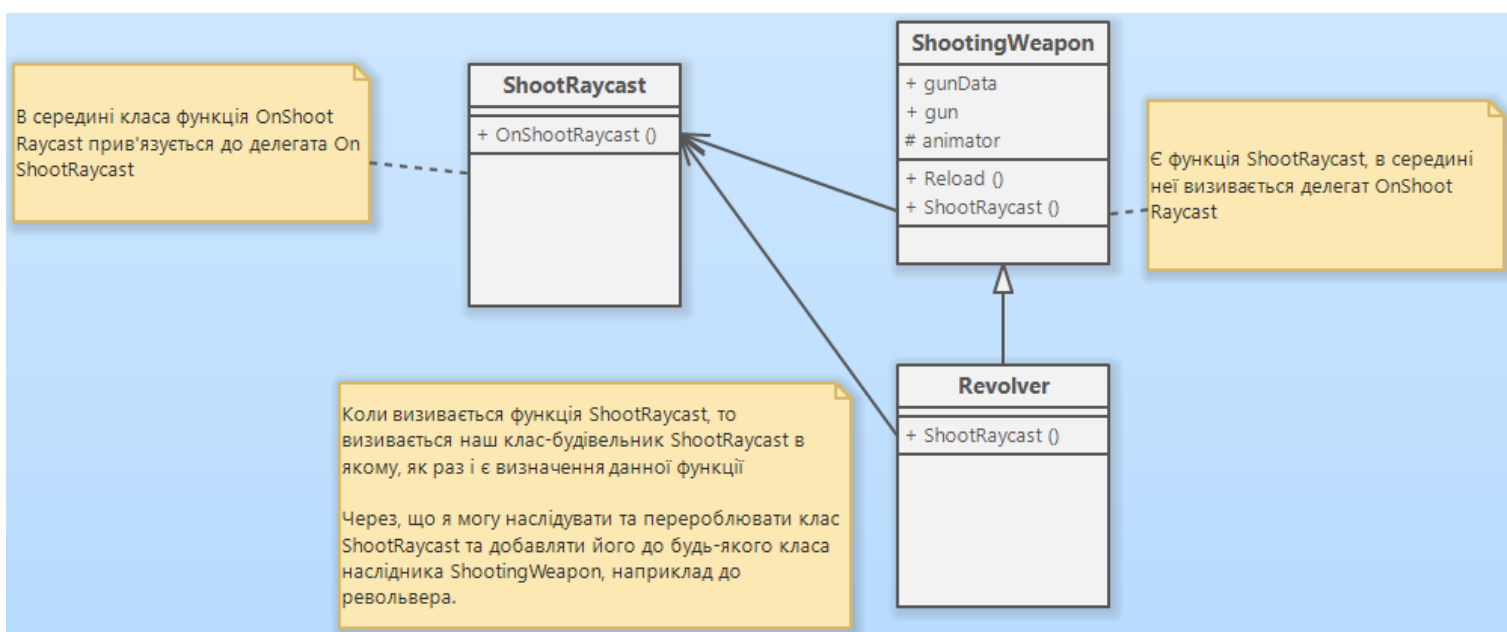


Будівельник - це породжувальний патерн проектування, який дає змогу створювати складні об'єкти покроково. Будівельник дає можливість використовувати один і той самий код будівництва для отримання різних уявлень об'єктів.

В моєму проекті цей патерн наявний, хоч і не в чистому вигляді. Наприклад, в мене є зброя, а саме вогнепальна зброя. Звісно вона не одного вигляду та звісно в мене є клас-база, в якому виділені всі основні методи та поля, але наприклад, пістолет та рушниця стріляють по різному, пістолет – пулею, рушниця картечкою. І що мені потрібно робити, щоб реалізувати різний тип вистрілу? Успадковувати клас-базу та реалізовувати для кожного підкласу метод по своєму. Так, добре, в цьому випадку мені це підходить. Але якщо в мене будуть 2 пістолета і вони мають різницю тільки в один метод. Реалізовувати два класа не має сенсу, тому для цього я використовую делегати.

Вони, як функція, яку можна доповнювати та перероблювати в будь якому іншому класі. Тобто мені не потрібно в класі пістолета реалізовувати метод вогню, потрібно лише прив'язати метод до делегата.

Для цього я використовую клас ShootRaycast, в мене він як раз і є будівельником. Всередині нього є функція, яка прив'язується до делегату. Цей клас можна успадковувати та перероблювати функціонал методу, як завгодно, але клас буде підходити для будь якого підкласу від ShootingWeapon.



В класі ShootRaycast в методі Start (що виконується коли гра стартує), до делегату класу ShootingWeapon прив'язується функція OnShootRaycast (це функція класу ShootRaycast).

```
protected virtual void Start()
{
    if (!CameraOrigin)
    {
        CameraOrigin = Camera.main;
    }
    GetComponent<ShootingWeapon>().OnShootRaycast += OnShootRaycast;
    if (!bulletHolesDataBase)
    {
        bulletHolesDataBase = GameObject.Find(ShootingWeaponConstants.BULLET_HOLES_DATA_BASE).GetComponent<BulletHolesDataBase>();
    }
}
```

Скрипти можете знайти за шляхом:

Assets/Scripts/Weapon/ShootingWeapon/MainScripts/ShootingWeapon.cs

Assets/Scripts/Weapon/ShootingWeapon/MainScripts/Revolver/Revolver.cs

Assets/Scripts/Weapon/ShootingWeapon/MainScripts/ShootRaycast.cs