
	M5 - ENTORNS DE DESENVOLUPAMENT		
	MP05/UF1-M16/UF2		
ACTIVITAT Pr4: Refactorización de código			
Apellidos: Tablit Miranda	Nombre: Brian	Fecha:	15/01/2023

## Descripción y Objetivos

El objetivo principal de esta práctica es aprender a refactorizar nuestro código, aplicando las nuevas técnicas y habilidades adquiridas, como los parámetros por referencia.

## Refactorización del Código del Videojuego



1. Usar las variables/parámetros por referencia para crear funciones globales donde creáis oportuno de vuestro código. (5p)

```

40 //Optimizar checkEnemyStatus1() y 2() pasamos todos los atributos del enemigo como parámetro por referencia
41 // a enemyHP, enemyName y enemyIsAlive. Con lo cual con una sola función comprueba el estado de cualquier enemigo.
42
43 bool checkEnemyStatus(int& enemyHP, string& enemyName, bool& enemyIsAlive) {
44     if (enemyHP <= 0) {
45         cout << "Te has cargado al enemigo " << enemyName << " \n";
46         enemyHP = 0;
47         enemyIsAlive = false;
48         return false;
49     }
50     else {
51         cout << "El enemigo " << enemyName << " tiene " << enemyHP << " puntos de vida\n";
52         return true;
53     }
54 }
55
56 //bool checkEnemyStatus1() {
57 //     if (enemyHP1 <= 0) {
58 //         cout << "Te has cargado al enemigo " << enemyName1 << " \n";
59 //         enemyHP1 = 0;
60 //         return false;
61 //     }
62 //     else {
63 //         cout << "El enemigo " << enemyName1 << " tiene " << enemyHP1 << " puntos de vida\n";
64 //         return true;
65 //     }
66 // }
67 //
68 //bool checkEnemyStatus2() {
69 //     if (enemyHP2 <= 0) {
70 //         cout << "Te has cargado al enemigo " << enemyName2 << " \n";
71 //         enemyHP2 = 0;
72 //         return false;
73 //     }
74 //     else {
75 //         cout << "El enemigo " << enemyName2 << " tiene " << enemyHP2 << " puntos de vida\n";
76 //         return true;
77 //     }
78 // }

```

- Para optimizar las funciones checkEnemyStatus1() y checkEnemyStatus2() pasamos todos los atributos del enemigo como parámetros por referencia a enemyHP con un int, el enemyName con string y enemyIsAlive dentro de una variable bool en checkEnemyStatus.

	M5 - ENTORNS DE DESENVOLUPAMENT		
	MP05/UF1-M16/UF2		
ACTIVITAT Pr4: Refactorización de código			
Apellidos: Tablit Miranda	Nombre: Brian	Fecha:	15/01/2023

```

int main()
{
    gameStart();
    while (heroIsAlive && (enemyIsAlive1 || enemyIsAlive2)) {
        //ELEGIMOS ENEMIGO AL QUE ATACAR
        enemyChosed = chooseEnemy();
        //ELEGIMOS ATAQUE Y CAPTURAMOS EL VALOR DEL DANYO QUE HAREMOS
        heroDamage = chooseAttack();

        if (enemyChosed == 1) {
            //ATACAMOS AL ENEMIGO 1
            enemyHP1 = heroAttackEnemy1(heroDamage);
            //llamamos la función
            checkEnemyStatus(enemyHP1, enemyName1, enemyIsAlive1);
        }
        else {
            //ATACAMOS AL ENEMIGO 2
            enemyHP2 = heroAttackEnemy2(heroDamage);
            //llamamos la función
            checkEnemyStatus(enemyHP2, enemyName2, enemyIsAlive2);
        }



        //ATACA EL ENEMIGO 1 SI ESTÁ VIVO Y EL HÉROE TAMBIÉN
        if (enemyIsAlive1 && heroIsAlive) {

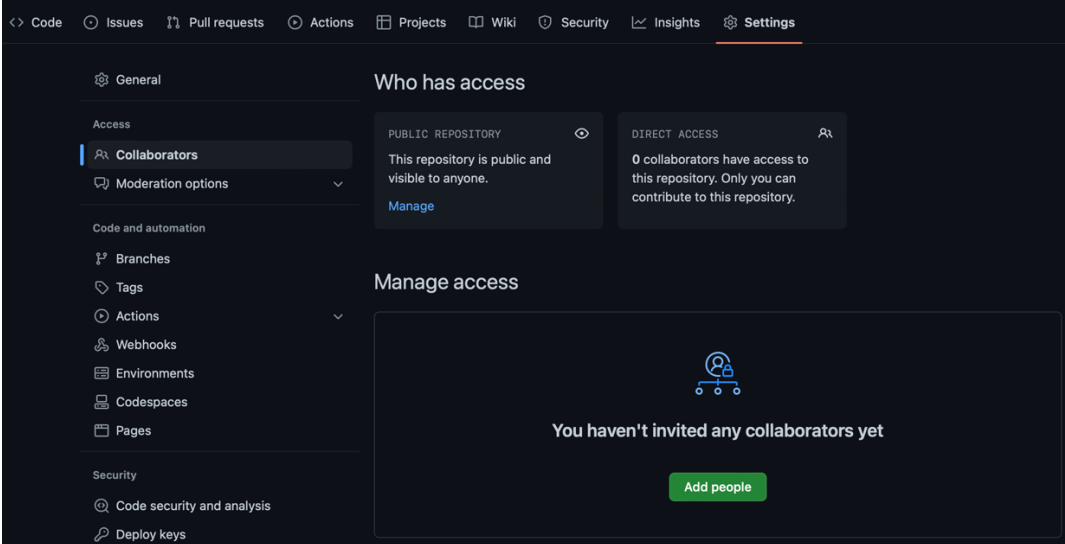
```

- Dentro del main() usaremos la función global que hemos creado de checkEnemyStatus. Junto la función ponemos los atributos del enemigo.
2. Modificar el código para mejorarlo, usando lo aprendido en M3 y M5/M16 desde el desarrollo de la PR3 (3p)
  3. Subir a GIT el juego, teniendo en cuenta los parámetros del apartado de **Entrega de la Actividad PR4. (2p)**

### Entrega de la Actividad Pr4:

- Al Moodle se subirá un fichero con únicamente una URL del proyecto de GIT
- La evaluación se completará con una **comprobación in situ**.
- En el proyecto de GIT, dentro de una carpeta que se llame PR4, tendrá que haber:
  - El código del juego (Sin los archivos de DEBUG)
  - Un documento PDF explicando todos los cambios realizados a nivel de código/funciones
- En caso de que el proyecto GIT sea privado, ir a SETTINGS→COLLABORATORS y añadir al usuario @JAVIZAWA



	M5 - ENTORNS DE DESENVOLUPAMENT		
	MP05/UF1-M16/UF2		
ACTIVITAT Pr4: Refactorización de código			
Apellidos: Tablit Miranda	Nombre: Brian	Fecha:	15/01/2023



The screenshot shows the GitHub repository settings page for a repository. The left sidebar contains navigation links: Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings (selected). The main content area is divided into two sections: 'Who has access' and 'Manage access'.

**Who has access:** This section shows the repository's access settings. It includes a 'General' tab and an 'Access' section. Under 'Access', there are two options: 'Collaborators' (selected) and 'Moderation options'. The 'Collaborators' section shows that the repository is public and visible to anyone. It also indicates that 0 collaborators have access to this repository, and only the user can contribute to this repository. A 'Manage' link is provided.

**Manage access:** This section shows the 'Manage access' tab. It displays a message: 'You haven't invited any collaborators yet'. Below this message is a green button labeled 'Add people'.

	M5 - ENTORNS DE DESENVOLUPAMENT		
	MP05/UF1-M16/UF2		
ACTIVITAT Pr4: Refactorización de código			
Apellidos: Tablit Miranda	Nombre: Brian	Fecha:	15/01/2023