



# Manual de Programador de FEAR & HUNGER

Creado y Diseñado por : Fernando Romero Gil.  
Curso 1 DAM B

La creación de clases de mi juego se ha basado en Clases se basa en 3 tipos :

HTML :

- Fear & Hunger.htm : Pagina principal del videojuego en la cual podemos hacer la creación , eliminación y continuación de la partida además de dirigirnos al repositorio.
- Crear Personaje.html : Es la pantalla la cual seleccionamos al personaje que deseamos jugar.
- D\_arce.html, Cahara.html, Enki.html, Ragnvald.html : Son las páginas en las cuales te enseña su mini historia y te deja seleccionarlo finalmente o volver atrás.
- Lobby.html : Lugar en el cual se encuentran las diferentes opciones para jugar.
- Tienda.html : Aqui es donde tu puedes seleccionar el vendedor que quieres asignar a tu personaje.
- TiendaObjeto.html : Lugar en el cual compramos los objetos.
- Inventario.html : Lugar en el cual se encuentran los objetos comprados por el usuario y también donde se pueden equipar
- Combate.html : Lugar en donde usted puede realmente jugar se le asignan un grupo de amigos que van saliendo en rondas.

CSS :

- Fear & Hunger.css : le da el estilo a Fear & Hunger.html
- Personaje.css : le da el estilo a la creación de personaje
- Tienda.css : le da el estilo a Tienda.html
- TiendaObjeto.css : le da el estilo a TiendaObjeto.html
- Inventario.css : le da el estilo al Inventario.html
- Lobby.css : le da el estilo a Lobby.html
- Combate.css : le da estilo a Combate.html
- CrearPersonaje.css : le da estilo a la selección de personajes
- Combate.css : le da el estilo a Combate.html

JS :

- Script.js : le da la funcionalidad a Fear & Hunger.html
- Personaje.js: le da la funcionalidad a creación de personaje
- Tienda.js : le da la funcionalidad a Tienda.html
- TiendaObjeto.js : le da la funcionalidad a TiendaObjeto.html
- Inventario.js : le da la funcionalidad a Inventario.html
- Lobby.js : le da la funcionalidad a Lobby.html
- Combate.js : le da la funcionalidad a Combate.html
- Arma.js : Es la que construye las armas de los personajes
- Combate.js : le da la funcionalidad a Combate.html
- Enemigos.js : es el constructor de los enemigos

## 2. Funciones Principales :

Explicaremos las funcionalidades de algunos trozos del código

```
export class Personaje {  
  constructor(nombre, nivel, vidaMaxima, ataque, defensa, velocidad, dinero) {  
    this.nombre = nombre;  
    this.nivel = nivel;  
    this.vidaMaxima = vidaMaxima;  
    this.vida = this.vidaMaxima;  
    this.ataque = ataque;  
    this.defensa = defensa;  
    this.velocidad = velocidad;  
    this.dinero = dinero;  
    this.experiencia = 0;  
    this.estado = "Normal";  
    this.inventario = new Inventario();  
    this.equipo = { arma: null};  
  }  
}
```

1º Este es el constructor de los personajes que son jugables en el videojuego para poder ingresar las variables al personaje.

```

const comprarArma = (indice) => {
  const arma = tienda.armasDisponibles[indice];

  if (personaje.dinero >= arma.costos) {
    personaje.dinero -= arma.costos;

    if (vendedor === "Isayah") {
      const estafa = {
        nombre: "Basura de Isayah",
        tipo: "Basura",
        poder: 0,
        costos: 0,
        descripcion: "Un regalo de Isayah. No sirve para nada.",
        imagen: "imagenes/Basura.png"
      };
      guardarEnInventario(estafa);
      alert("¡Isayah te ha estafado! Has recibido Basura de Isayah.");
    } else {
      guardarEnInventario(arma);
      armasCompradas.push(arma.nombre);
      guardarEstadoArmas();
      alert(`Has comprado ${arma.nombre} por ${arma.costos} monedas.`);
    }

    localStorage.setItem('personaje', JSON.stringify(personaje));

    actualizarDinero();
  } else {
    alert("Dinero insuficiente.");
  }
};

```

2º Esta es la función mediante la cual la lógica de la pagina actual en el cual hace dos cosas quitar el dinero del personaje al comprar un objeto y dentro de él reconoce si el vendedor es Isayah y si lo es procede a ingresar en el inventario la basura y guarda los cambios en el localStorage.

```
function cargarPartida() {
  const personajeGuardado = localStorage.getItem('personaje');
  const rondaGuardada = localStorage.getItem('rondaActual');

  console.log("Personaje guardado:", personajeGuardado);
  console.log("Ronda guardada:", rondaGuardada);

  if (personajeGuardado && rondaGuardada !== null) {
    try {
      personaje = Personaje.fromJSON(JSON.parse(personajeGuardado));
      rondaActual = parseInt(rondaGuardada, 10);
      console.log(`Partida cargada. Personaje: ${personaje.nombre}, Ronda: ${rondaActual}`);
      alert("Partida cargada. Redirigiendo al lobby...");
      window.location.href = "Lobby.html";
    } catch (error) {
      console.error("Error al cargar la partida:", error);
      alert("Hubo un error al cargar la partida. Por favor, intenta de nuevo.");
    }
  } else {
    console.log("No hay partida guardada.");
    alert("No hay partida guardada.");
  }
}
```

3° Así es como la Carga de partida en la página principal del videojuego guarda la ronda del personaje y el personaje guardado y a continuación y te redirige a la página de lobby de la partida pasada.

```
export class Enemigo {
  constructor(nombre, nivel, vida, ataque) {
    this.nombre = nombre;
    this.nivel = nivel;
    this.vida = vida;
    this.ataque = ataque;
    this.defensa = 3 + (nivel);
  }
}
```

4 ° Este es el constructor de enemigos con sus variables y los datos correspondientes.

## Modificaciones y Ampliaciones :

```
function iniciarRagnvaldr() {  
  const personaje = {  
    nombre: "Ragnvaldr",  
    nivel: 1,  
    vida: 120,  
    ataque: 10,  
    defensa: 8,  
    velocidad: 4,  
    dinero: 200  
  };  
  guardarPersonaje(personaje);  
}  
  
function iniciarDarce() {  
  const personaje = {  
    nombre: "D'arce",  
    nivel: 1,  
    vida: 100,  
    ataque: 8,  
    defensa: 10,  
    velocidad: 6,  
    dinero: 200  
  };  
  guardarPersonaje(personaje);  
}  
  
function iniciarEnki() {  
  const personaje = {  
    nombre: "Enki",  
    nivel: 1,  
    vida: 100,  
    ataque: 12,  
    defensa: 6,  
    velocidad: 8,  
    dinero: 200  
  };  
  guardarPersonaje(personaje);  
}  
  
function iniciarCahara() {  
  const personaje = {  
    nombre: "Cahara",  
    nivel: 1,  
    vida: 110,  
    ataque: 9,  
    defensa: 7,  
    velocidad: 9,  
    dinero: 2000  
  };  
  guardarPersonaje(personaje);  
}
```

5 ° Aquí se encuentra la creación de los 4 personajes y según la descripción del usuario uno de los 4 será seleccionado y aquí tmb se puede modificar sus estadísticas como el oro del sahara que lo he cambiado para hacer pruebas

```

this.armasDisponibles = [
  { nombre: "Hoz", poder: 10, tipo: "Hoz", costo: 100, imagen: "imagenes/hoz.png", comprado: false },
  { nombre: "Tubería oxidada", poder: 7, tipo: "Tubería", costo: 70, imagen: "imagenes/tuberia.png", comprado: false },
  { nombre: "Picadora de carne", poder: 20, tipo: "Sierra", costo: 200, imagen: "imagenes/picadora.png", comprado: false },
  { nombre: "Mazo de carne", poder: 22, tipo: "Mazo de carne", costo: 220, imagen: "imagenes/mazo.png", comprado: false },
  { nombre: "Rastrillo del segador", poder: 25, tipo: "Rastrillo del segador", costo: 250, imagen: "imagenes/rastrillo.png", comprado: false },
  { nombre: "Linterna cabeza de Guerrero", poder: 30, tipo: "Linterna cabeza de Guerrero", costo: 300, imagen: "imagenes/linterna.png", comprado: false },
  { nombre: "Espada de sangre", poder: 30, tipo: "Espada de sangre", costo: 300, imagen: "imagenes/espada_sangre.png", comprado: false }
];

```

6 ° Dentro de la clase tienda yo he creado los diferentes tipos de armas y ahí podremos ingresar un nuevo arma siguiendo los pasos del constructor.

```

const enemigos = [
  { nombre: "La Niña", vida: 30, maxVida: 30, ataque: 5, defensa: 2, imagen: "imagenes/la_nina.png" },
  { nombre: "Guarda", vida: 50, maxVida: 50, ataque: 10, defensa: 5, imagen: "imagenes/guarda.png" },
  { nombre: "Manuba", vida: 70, maxVida: 70, ataque: 15, defensa: 8, imagen: "imagenes/manuba.png" },
  { nombre: "Guardia de Elite", vida: 150, maxVida: 150, ataque: 12, defensa: 10, imagen: "imagenes/guardia_elite.png" },
  { nombre: "Guarda sin Mandibula", vida: 110, maxVida: 110, ataque: 20, defensa: 8, imagen: "imagenes/guarda_sin_mandibula.png" },
  { nombre: "Prisionero Negro", vida: 80, maxVida: 80, ataque: 20, defensa: 5, imagen: "imagenes/prisionero_negro.png" },
  { nombre: "Hombre Lagarto", vida: 150, maxVida: 150, ataque: 25, defensa: 15, imagen: "imagenes/hombre_lagarto.png" },
  { nombre: "Estacada Nocturna", vida: 90, maxVida: 90, ataque: 25, defensa: 12, imagen: "imagenes/estacada_nocturna.png" },
  { nombre: "Ladrón de Cuerpos", vida: 100, maxVida: 100, ataque: 20, defensa: 10, imagen: "imagenes/ladron_de_cuerpos.png" },
  { nombre: "Señor de las Moscas", vida: 200, maxVida: 200, ataque: 15, defensa: 20, imagen: "imagenes/senor_de_las_moscas.png" },
  { nombre: "El Cuervo", vida: 250, maxVida: 250, ataque: 33, defensa: 25, imagen: "imagenes/el_cuervo.png" }
];

```

7° Este es donde se han creado los enemigos contra los que el usuario tiene que luchar.