

# GDD

## UNDER PRESSURE

# UNDER PRESSURE



VERGOÑA  
GAMES

Jesús Culebras González  
Aitor Lebrero Barroso  
Marcos Toledo Sánchez  
Juegos en Red, Quintana, 2021-2022

# Índice

<b>1. Introducción</b>	<b>3</b>
1.1. Concepto	3
1.2. Características principales	3
1.3. Género	3
1.4. Propósito	3
1.5. Jugabilidad	4
1.6. Estilo Visual	4
1.7. Expansión	4
<b>2. Mecánicas</b>	<b>5</b>
2.1. Resumen	5
2.2. Tareas	5
2.3. Averías	5
2.4. Salto	5
2.5. Flujo	6
<b>3. Personajes</b>	<b>7</b>
3.1. Personajes Jugables	7
3.2. Personaje No Jugables (NPC)	8
<b>4. Movimiento</b>	<b>9</b>
4.1. Controles	9
4.2. Colisiones	9

# *1. Introducción*

En este documento se va a plasmar toda la información relevante a “Under Pressure”, un videojuego multijugador diseñado por Vergoña Games para navegadores web que será desarrollado utilizando el framework Phaser.

## **1.1. Concepto**

En “Under Pressure” el jugador controlará a un nuevo empleado de la oficina de trámites burocráticos de Ciudad M. Lo que él no sabe es que ha tenido la mala suerte de entrar a trabajar en la oficina del Estado más rocambolesca de su ciudad en la que tendrá que competir no solo contra la burocracia sino contra sus propios compañeros de trabajo que, como él, están trabajando bajo presión para mantener su puesto en la oficina.

## **1.2. Características principales**

El rasgo principal de “Under Pressure” es la dificultad **a la hora de moverse** por el escenario de la partida; debido a la interacción entre jugadores y la constante presión que se siente por el rápido paso del tiempo, para hacer esto más llevadero se ha optado por darle al juego unos **controles simples** que resulten fáciles para cualquier tipo de jugador.

## **1.3. Género**

“Under Pressure” se trata de un juego arcade, casual y multijugador en el que los jugadores deberán completar una sucesión de tareas y minijuegos antes de que se agote el tiempo establecido.

## **1.4. Propósito**

“Under Pressure” nace como una crítica a la absurdez que plantean los trámites burocráticos; muchas veces para hacer el trámite más simple hay que dar muchos papeles, dar muchas vueltas... Gracias a este concepto, “Under Pressure” puede llegar a todo tipo de

jugadores, tanto casuales como hardcore, que quieran jugar una partida rápida con sus amigos o compañeros sin necesidad de instalar nada.

## **1.5. Jugabilidad**

Al comenzar la partida en “Under Pressure”, los jugadores comenzarán en la recepción del edificio donde recibirán su primera misión. A partir de aquí los jugadores navegarán por el edificio, presentado desde una vista frontal, podrán acceder a las distintas plantas del mismo. El jugador debe ir completando las distintas tareas (entregar unos papeles en otro mostrador, rellenar algún formulario...) que le irán poniendo en cada mostrador, simbolizado con flechas del color de cada jugador. Cada jugador tiene un tiempo que disminuye continuamente pero aumenta si completa una tarea y si este llega a 0 el jugador perderá. Ganará el último jugador en pie. Para añadir a la competitividad, los jugadores podrán chocarse entre ellos para evitarse el paso. Según vaya pasando el tiempo, para añadir un poco más de dificultad, varias de las instalaciones se estropearán y los jugadores deberán arreglarlas a través de unos pequeños minijuegos, para ganar al juego los jugadores deberán sobrevivir un día entero de trabajo.

## **1.6. Estilo Visual**

“Under Pressure” tiene una estética pixel-art, muy reminiscente a la que se puede encontrar en juegos de la SNES, con una paleta de color muy amplia y un estilo no muy detallista para no recargar demasiado los entornos y que le resulte claro al jugador.

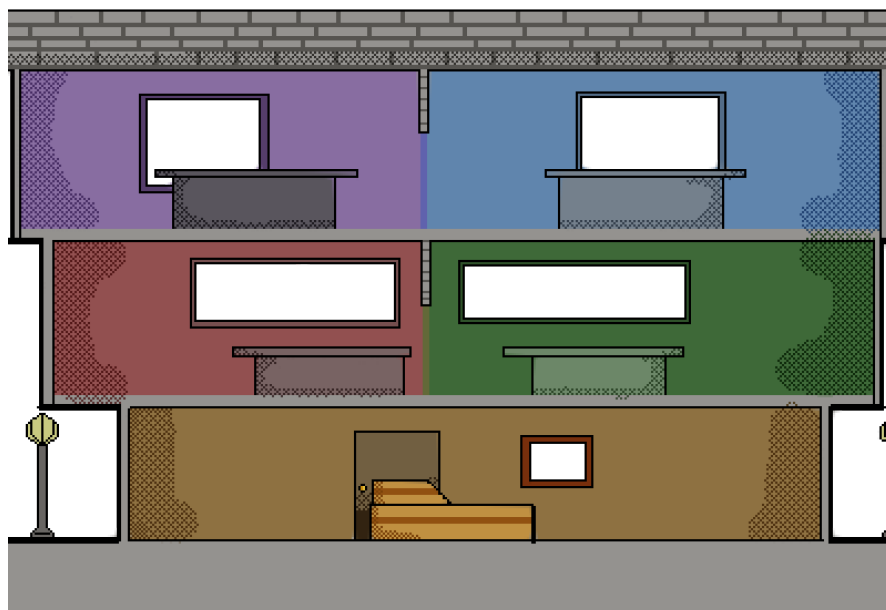


Concept-Art de la recepción

El fondo tiene pocos elementos visibles, destacando sólo las nubes en continuo movimiento en el cielo, para simular ese paso del tiempo rápido y constante.

## 1.7. Expansión

En un principio, “Under Pressure” saldrá con un solo nivel, pero si el título recibe apoyo se podría plantear añadir más niveles en forma de edificios con distintos layouts para diferenciar las distintas partidas.



## 2. Mecánicas

En esta sección se profundizará en las mecánicas que componen “Under Pressure”.

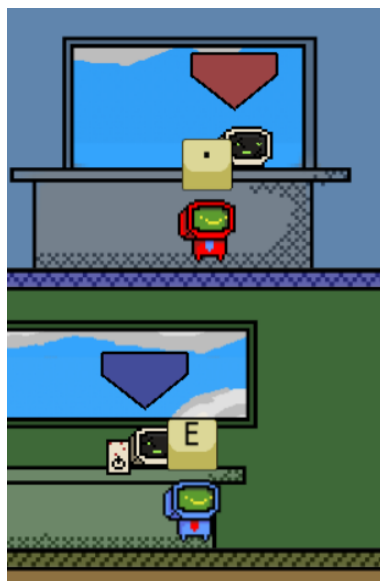
### 2.1. Resumen

El nivel se trata del edificio de trámites burocráticos de Ciudad M, dividido en 3 plantas: la recepción; con un mostrador, y las dos plantas superiores; con dos mostradores cada una. En cada mostrador habrá un empleado que dará tareas a los jugadores. Mientras esto ocurre, en la oficina irán ocurriendo una serie de averías que dificultarán el trabajo de los jugadores

### 2.2. Tareas

Las distintas tareas que pueden ofrecer los funcionarios serán similares entre sí para no liar mucho al jugador, de tal forma que puedan ser procesos mecánicos en los que no haya que pensar mucho. Se usará una sola tecla para interactuar en los mostradores, las tareas consistirán simplemente en interactuar en las distintas ventanillas para poder completarlas. El jugador que complete una tarea, obtendrá una bonificación de tiempo para su contador. Si un jugador completa una tarea, se generarán nuevas tareas para ambos jugadores, ayudando a la competitividad, ya que si un jugador llega antes a completar su tarea, perjudicará al otro.

En la siguiente imagen, se ve como estarán indicadas las tareas y los botones con los que podrán interactuar:



### 2.3. Averías

Durante el transcurso de la partida y de forma esporádica pasará algo en la oficina que perjudicará a los jugadores: Que se caiga una maceta e impida el paso, que la impresora se incendie y haya que apagarla y que se desplome una librería, para arreglar estas averías el jugador deberá usar su botón de interacción. En el caso de que no se arregle ninguna avería durante el curso de la partida, se considerará que ambos jugadores han perdido la partida.



### 2.4. Salto

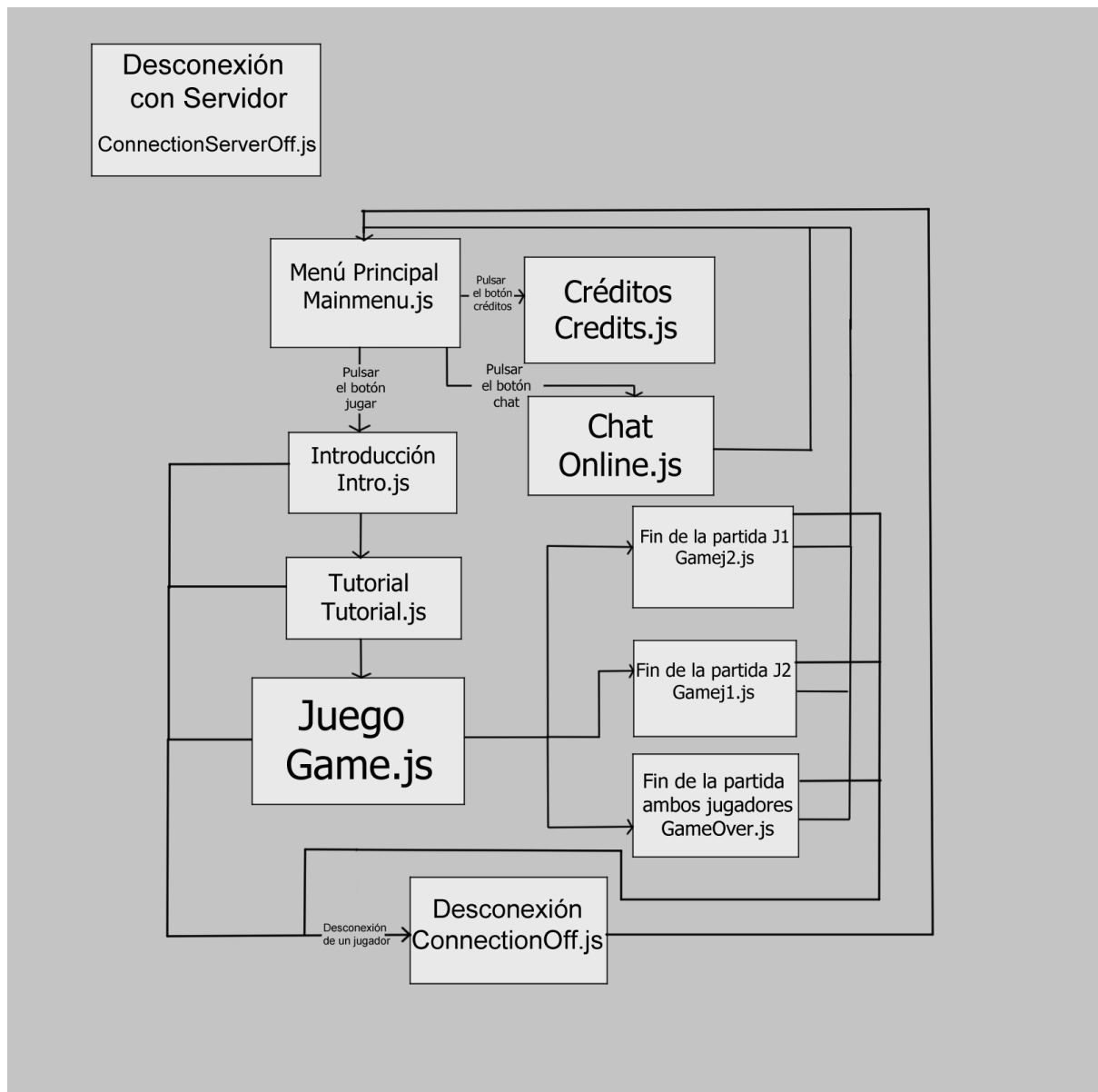
Para esquivar al otro jugador, el personaje tiene la capacidad de ejecutar un salto, para superar al rival, no obstante, si el otro jugador salta a la vez, ambos chocarán en el aire y no habrá servido de nada.

## 2.5. Flujo

Al empezar el juego, los jugadores serán recibidos por el menú principal del juego, donde podrán elegir entre iniciar la partida, ver los créditos del juego o ir a la sala de chat online.

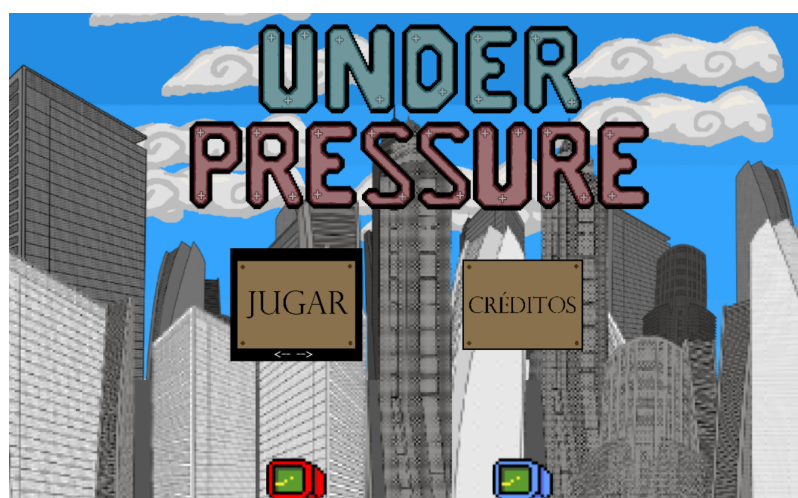
Una vez iniciada la partida, los jugadores recibirán una pequeña introducción y verán el tutorial del juego, tras esto, dará comienzo la partida. En la parte superior de la pantalla verán sendos contadores que indicarán el tiempo restante hasta el despido de cada jugador, cada tarea completado con éxito aumentarán el tiempo del jugador que la haya completado y cambiará la localización de las misiones de los dos jugadores. Cuando se cumplan una de las tres condiciones de derrota (Que el J1 se quede sin tiempo, que el J2 se quede sin tiempo, o que no se arregle ninguna avería) se pasará a las pantallas de Game Over correspondientes, donde se dará la opción de volver al menú principal. Para el modo multijugador, se ha implementado una pantalla de desconexión para cuando un jugador se sale del juego, para que el jugador que se queda se entere. En este diagrama de flujo está todo lo anterior representado:





### 2.5.1. Menú Principal

Se trata de lo primero que verá el jugador al iniciar el juego, da como opciones empezar a jugar y ver los créditos. En ella aparecen los protagonistas andando hacia el trabajo. Para el fondo se ha utilizado el efecto parallax, para incrementar la sensación de movimiento.



### 2.5.2. Escena del juego

En ella se ve el edificio ya con todos los demás assets colocados en su posición (suelos, paredes, averías...). También se puede ver los contadores de tiempo de cada jugador.



### 2.5.3. Game Over J1

Mostrará un fondo azul para indicar que ha ganado el J1, y dará la opción de volver al menú principal.



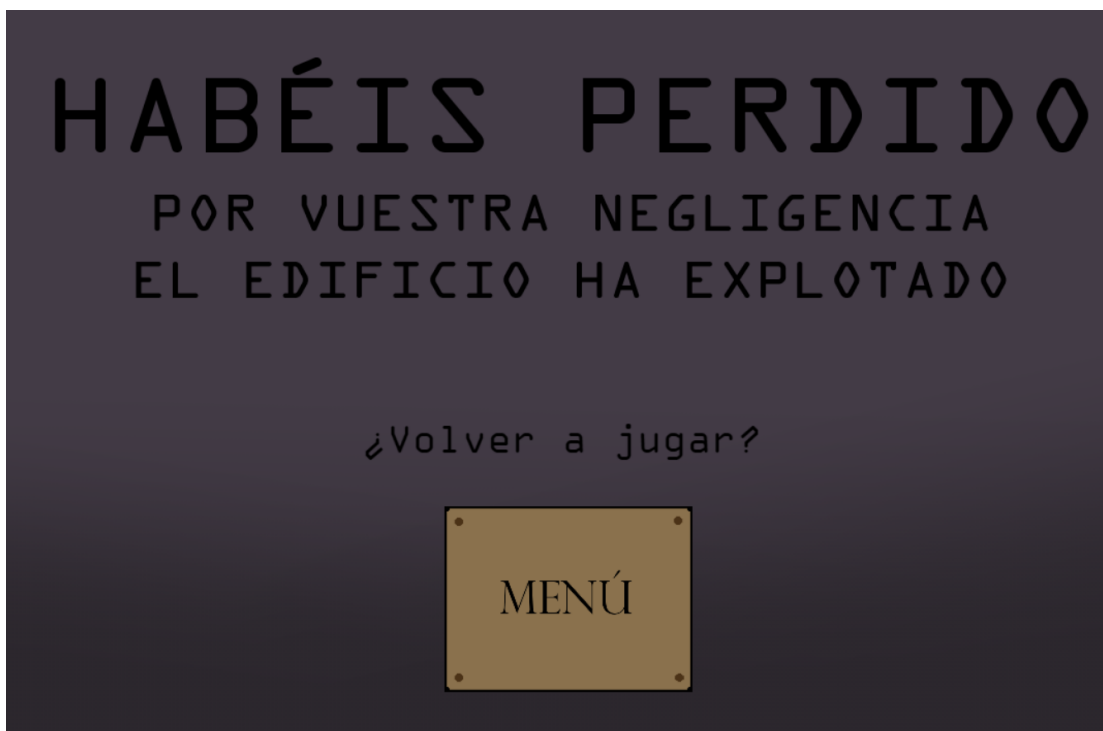
#### 2.5.4. Game Over J2

Mostrará un fondo rojo para indicar que ha ganado el J2, y dará la opción de volver al menú principal.



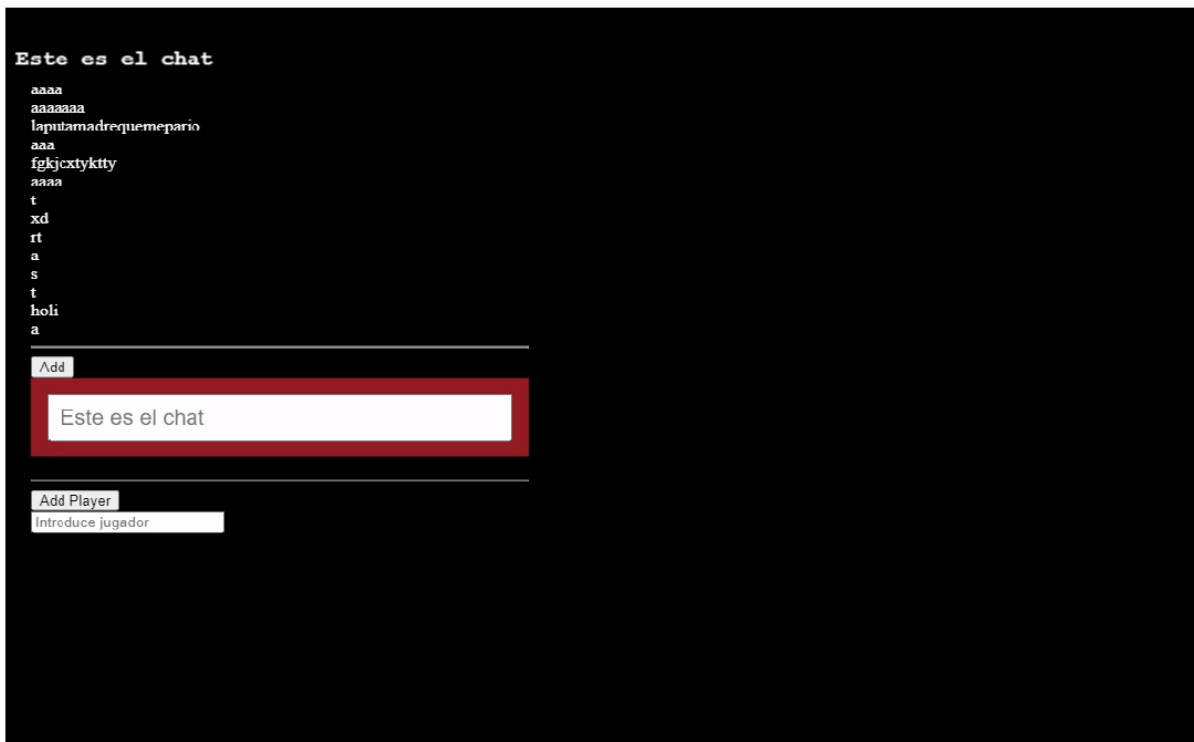
#### 2.5.5. Game Over ambos jugadores

Si los jugadores ignoran las averías durante mucho tiempo, el juego acabará para ambos.



### 2.5.6. Chat online

Escena donde los usuarios van a poder hablar entre ellos



### 2.5.6. Pantalla de desconexión

Escena que indica al usuario que el otro jugador se ha desconectado.

EL OTRO JUGADOR  
SE HA DESCONECTADO

#### 2.5.6. Pantalla de desconexión del servidor

DESCONEXIÓN CON EL  
SERVIDOR

Volver a jugar?

MENÚ

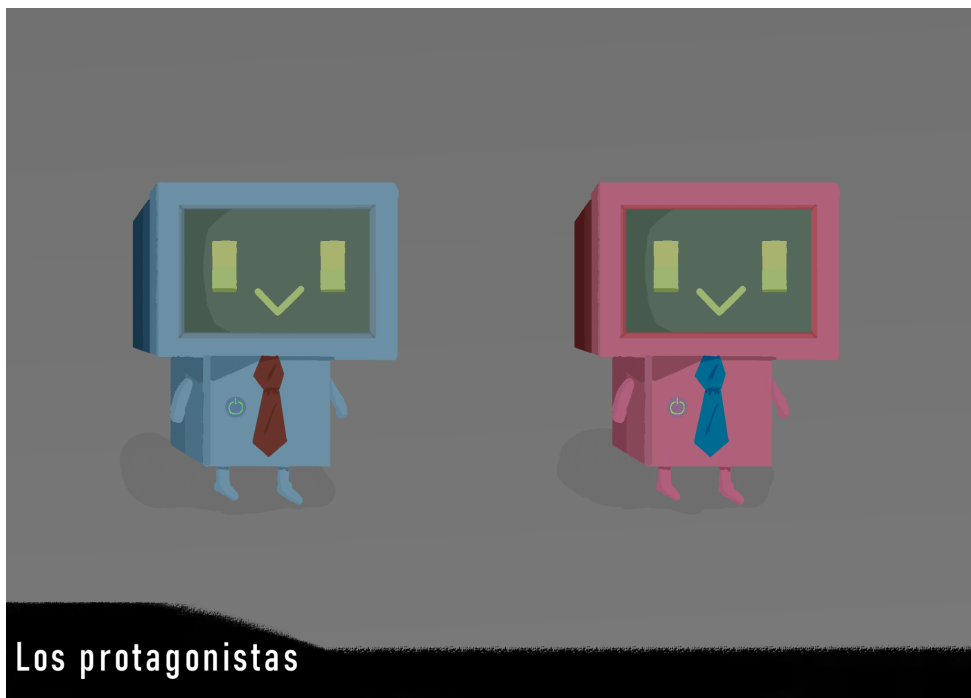
### *3. Personajes*

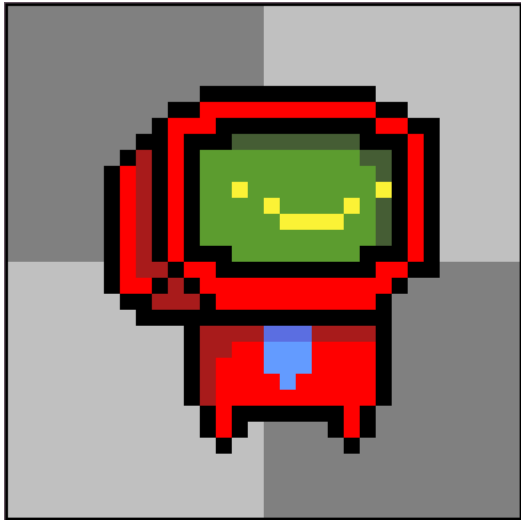
En este apartado se detallará tanto el aspecto como las habilidades de las que dispondrán. En Ciudad M, los ciudadanos son monitores de ordenador antropomórficos, siendo los ancianos monitores CRT y los jóvenes monitores planos.

#### **3.1. Personajes Jugables**

En “Under Pressure” los jugadores encarnan a un simpático monitor de ordenador antropomórfico de mediana edad que se dirige a realizar, lo que él cree, es una jornada más de trabajo.

Se ha optado por un diseño de personaje simple y amigable para que resulte agradable a cualquier tipo de jugador.





Para diferenciar a los jugadores entre sí, cada uno tendrá una paleta de colores distinta siguiendo las paletas que tradicionalmente se han venido usando en la industria (Azul para el J1 y rojo para el J2).

Los jugadores tendrán la capacidad de moverse por el escenario y de interactuar con los elementos del mismo, además de poder ejecutar un salto, para esquivar al otro jugador.

### 3.2. Personaje No Jugables (NPC)

El elenco de NPCs de “Under Pressure” está compuesto de otros funcionarios que trabajan en la oficina del Estado de Ciudad M los cuales les facilitan las tareas a realizar (trasladar papeles, fotocopiarlos, enviar correos...), se tratan de monitores más viejos que los protagonistas para recalcar lo anticuado que está el sistema burocrático actual y la necesidad de actualización del mismo.

Son ellos los encargados de dar las distintas tareas a los jugadores.



## 4. Movimiento

“Under Pressure” se desarrolla sobre un plano bidimensional, los jugadores se moverán sobre todo sobre el eje horizontal y, de vez en cuando, sobre el vertical.

### 4.1. Controles

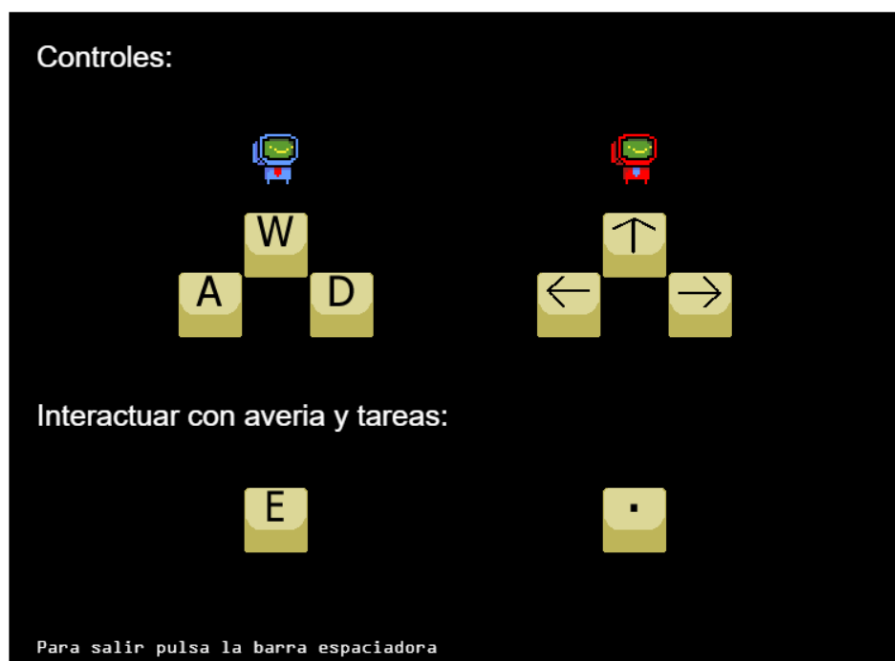
Se van a listar los controles de “Under Pressure”:

-Jugador 1:

- A: Desplazarse a la izquierda
- D: Desplazarse a la derecha
- W: Saltar
- E: Interactuar

-Jugador 2:

- Flecha Izquierda: Desplazarse a la izquierda
- Flecha Derecha: Desplazarse a la derecha
- Flecha Arriba: Saltar
- Botón punto: Interactuar





## 4.2. Colisiones

Como ya se ha mencionado, se implementará un sistema de colisiones entre los jugadores para que se molesten los unos a los otros añadiendo así un nuevo nivel de dificultad. Cuando dos jugadores chocan ambos rebotan hacia atrás, impidiendo el paso y retrasando al jugador.

## 5. Protocolo Websockets

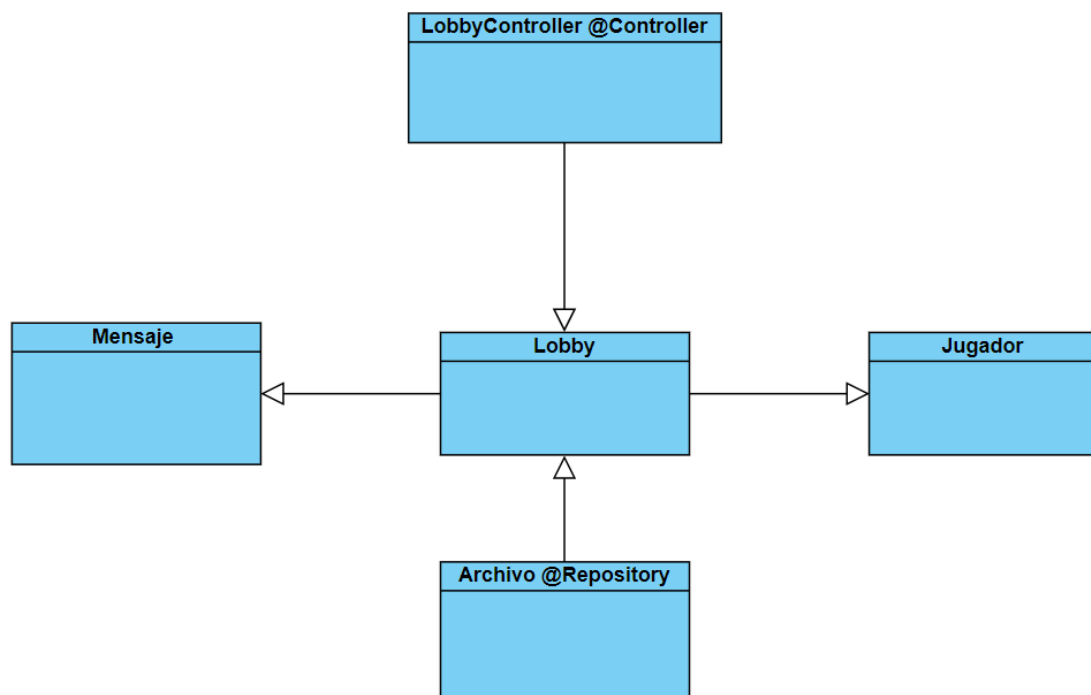
Hemos utilizado el protocolo websocket para crear conexiones entre los usuarios con el servidor y viceversa. Son conexiones que se realizan constantemente en ambos sentidos y según lo queramos, los programadores, a la hora de mandar mensajes.

En la primera pantalla, *"Mainmenu"*, se crean las conexiones en websocket para cada persona que entra en nuestro juego, estas conexiones son gestionadas por la clase *"SceneHandler"*, ya que es la encargada de, dependiendo del mensaje recibido por el servidor, enviarlos al resto de conexiones o a todas incluido el mismo y hacer lo que se necesite con la información enviada. Añadir que esta clase *"SceneHandler"* es creada e inicializada en la clase *"RestEjer1ConUiApplication"* al añadir todas las dependencias de websockets.

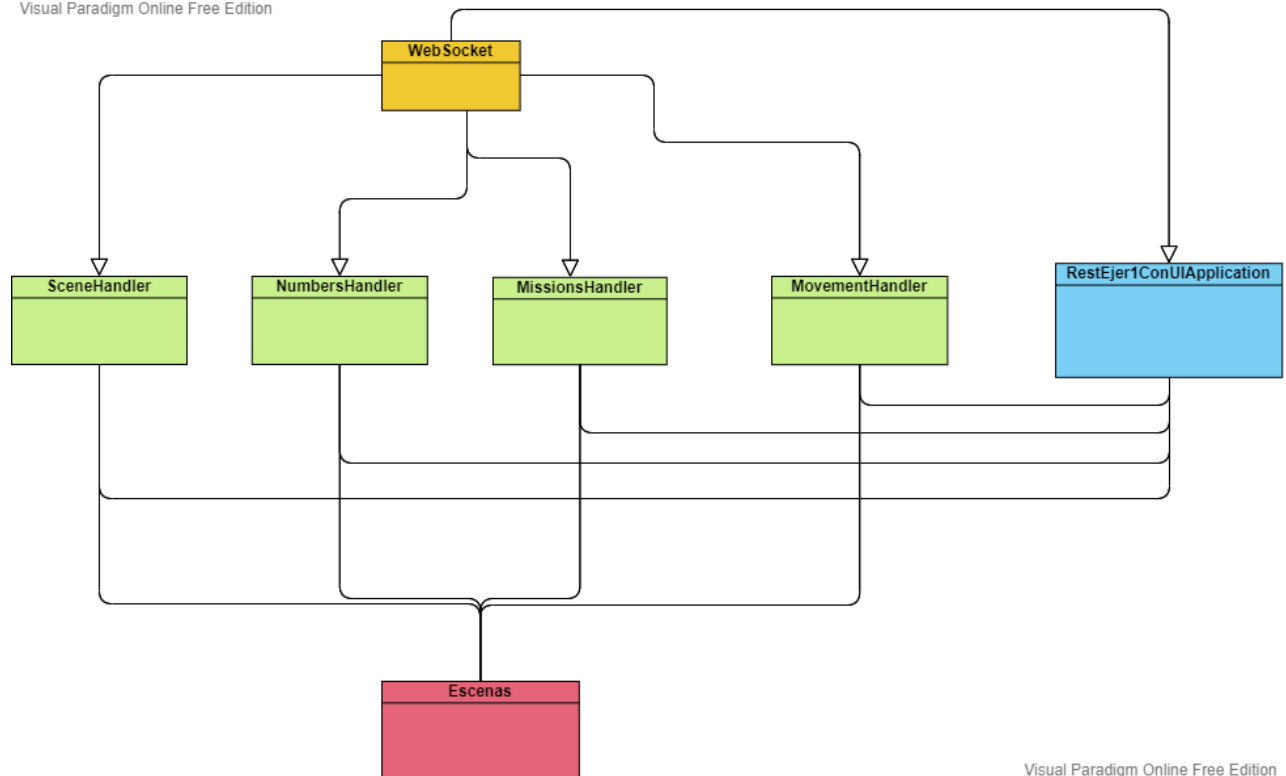
Las conexiones se usan para:

- Cambiar entre escenas, si un jugador quiere avanzar el resto reciben mensajes que les obliga a cambiar de escena. Salvo en el cambio de escena antes del juego en el que ambos jugadores deberán enviar el mensaje de que quieren avanzar para que avancen.
- Controlar al personaje del otro jugador, ambos jugadores llevarán al personaje azul, pero se verá representado por el personaje rojo para el otro jugador, entonces por cada movimiento que hagan se mandará mensajes al servidor para que el otro jugador las reciba y las interprete como el movimiento realizado por el mismo, estos mensajes son continuos y siempre se estan mandando y recibiendo.
- Asignar las misiones y averías, se realizan mediante números aleatorios que se enviarán al servidor y luego todos los jugadores lo recibirán y lo utilizará para mostrar por pantalla, las misiones entre ambos jugadores aparecerán invertidas de manera que visualmente el personaje rojo vaya a las misiones rojas y el azul a las azules. La gestión para la no repetición de números aleatorios generados se da en la clase *"SceneHandler"*, haciendo que el juego sea jugable.

## 6. Diagrama de clases



Hemos creado 5 clases para la gestión del servidor, en este se pueden observar las clases Mensaje, Lobby y Jugador, están son las que se encargan de determinar qué es lo que va a poder guardarse en el propio servidor, con los atributos que indican cómo son las clases, para el control de los datos se creó la clase LobbyController en esta se establece todo o relacionado con REST, colocando métodos Get, Delete y Post, estos método se relacionarán con el javascript, de tal manera que en el javascript (con el uso del lenguaje de jquery), podrás acceder a estos para introducir, eliminar y crear cosas en el propio servidor, por último, se encuentra la clase Archivo, esta se va a encargarse de gestionar la persistencia de datos (por eso hemos creído que se trataría de una clase tipo repository) está tiene 2 métodos de los que 1 se encarga de leer un archivo de texto (para hacer un get que coloque lo que haya en ese fichero en la propia página web) y el otro se encarga de escribir en el archivo de texto (para hacer un post que escribirá en el archivo de texto lo que se escriba en la página web).



Con este nuevo diagrama de clases queríamos mostrar que tanto la clase “*SceneHandler*”, “*MissionsHandler*”, “*NumbersHandler*”, “*MovementHandler*” como la clase “*RestEjer1ConUiApplication*” implementan alguna librería de websocket y juntan trabajan para controlar las conexiones en todas las escenas de nuestro juego.

Dentro de todos los Handlers que sirven para hacer conexiones cliente/servidor con el uso de sockets, podemos encontrarnos:

**SceneHandler:** Este Handler se encarga de gestionar el control de escenas y la sincronización de los jugadores a lo largo del flujo del juego con el fin de que puedan acceder al mismo tiempo a las diferentes pantallas para que estén en igualdad de condiciones, a su vez este va a permitir gestionar las desconexiones simultáneas y controladas en el servidor, de tal manera que al desconectarse un cliente lo notificará a los demás para que vayan al menú principal nuevamente y puedan conectarse a uno nuevo, también, este permitirá inicializar las escenas, generando las coordenadas de los jugadores así como las primeras misiones.

**MovementHandler:** Este Handler se encarga de gestionar el movimiento de los jugadores de tal manera que al moverse un cliente llamará a este para que pueda pasar la información al server y este la traslade al resto de clientes.

MissionsHandler: Este Handler servirá para establecer las nuevas misiones tras completar una, de tal manera que el jugador que llame a este, generará las misiones aleatoriamente y se las enviará al servidor para que este pueda traspasarlas al resto de jugadores.

NumbersHandler: Por último, NumbersHandler se encarga de establecer las averías del edificio aleatoriamente, este contiene ciertas herramientas de sincronización con el fin de que no haya problemas de concurrencia, y enviará el número aleatorio al resto de jugadores.

## 7. Instrucciones de ejecución

Antes de indicar dónde se encuentra el .jar será necesario exponer cómo ejecutar cualquier .jar, se requiere instalar una máquina virtual que traduzca el código para ello habrá que acudir a la web oficial de java e instalarlo (de manera gratuita).

Enlace al instalador: <https://www.java.com/es/download/>

Una vez descargado se podrá ejecutar el .jar del proyecto en cuestión:

Para poder empezar a jugar será preciso ejecutar el .jar, haciendo doble click en la siguiente ruta **F\_3\_GRUPO\_D/target/items-backend-and-frontend-0.0.1-SNAPSHOT.jar**, una vez se haya hecho el doble click, se deberá acudir al navegador para escribir lo siguiente:

Si se quiere conectar localmente (solo lo podrá hacer con el ordenador donde se haya ejecutado el jar) pondrá **localhost:8080**.

Si se encuentra en un ordenador en la misma red pero no es el propio ordenador deberá copiar la ip del ordenador local acudiendo con **cmd** (escrito en la barra de búsqueda de windows) a la pantalla de comandos donde se deberá escribir **ipconfig** para adquirir la ip que se encontrará en **Dirección IPv4** (se tratará de una secuencia de cuatro números cada uno aislado del resto con un punto) una vez copiada se pegará en el buscador y se pondrá a su lado lo siguiente **:8080**.

Para ejecutarlo desde cmd sería colocar la ruta hasta el .jar y al darle al intro con eso ya se prepararía el servidor y se podría acceder desde el buscador colocando **localhost:8080**. o la dirección ip del ordenador en ejecutarlo junto con **:8080**.