

Proyecto

Sprint semana(2/05/2019-10/05/2019)

El primer sprint que volem fer, és tenir tot lo que es el login,logout,registrarse, con la base de datos(Sólo el apartado de usuarios)

Martes 30-04-2019

Haré el Login, Logout y la base de datos, éste último lo haré después.

Usaré la librería bootstrap para esta vista, junto a los controllers y modelos.

Me ha surgido un problema a la hora de mostrar un error, sí me lo muestra la primera vez pero a la segunda me sale un error de 'Archivo no encontrado'.

Jueves 02-05-2019

Hemos encontrado solución a todos los fallos que teníamos, la detección de errores (La que sólo me dejaba hacer una vez).

La solución fue en el action del form, pasarla por index.php?section=login, de esta manera siempre estoy en el archivo index.php por lo que el algoritmo nunca me mandará a controller dos veces, como me pasaba antes.

La página del registro ya está hecha, nos queda toda la parte lógica de lo que es insertar en la base de datos y enlazarla correctamente.

Martes 07-02-2019

Hasta este día conseguimos meter el mapa y ese mismo día metimos personaje y sus animaciones. El error es que creabamos mal el phaser, ya que hice una pruebas y me di

cuenta que la creación del phaser es lo que fallaba. Así que la solución es crear de esta manera el phaser:

```
var config =
{
  type: Phaser.CANVAS,
  width: 820,
  height: 740,
  physics:
  {
    default: 'arcade',
    arcade: {
      gravity: {y: 300},
      debug: true
    }
  },
  scene: {
    preload: preload,
    create: create,
    update: update
  }
};
var juego = new Phaser.Game(config);
```

Las colisiones sí las insertamos, pero tenemos un problema, que las colisiones son solo por la parte superior, al caminar hacia una pared se atraviesa, esto quiere decir que por los laterales no hay colisiones.

Jueves 09-05-2019

Hoy tengo que ver el porqué no hay colisiones laterales usando el mapa creado en tileMap.

Seguimos sin poder resolver las colisiones laterales.

Pero no “perderemos más tiempo”, dado que somos conscientes que este fallo es grave, no queremos dejar de lado el resto. Encontraremos la solución pero queremos que cuando lo encontremos tengamos más cosas hechas y no nos pase como los demás proyectos.

Viernes 10-05-2019

Hoy es el día de entrega del sprint, así que hoy juntaremos lo que debe de ser el proyecto.

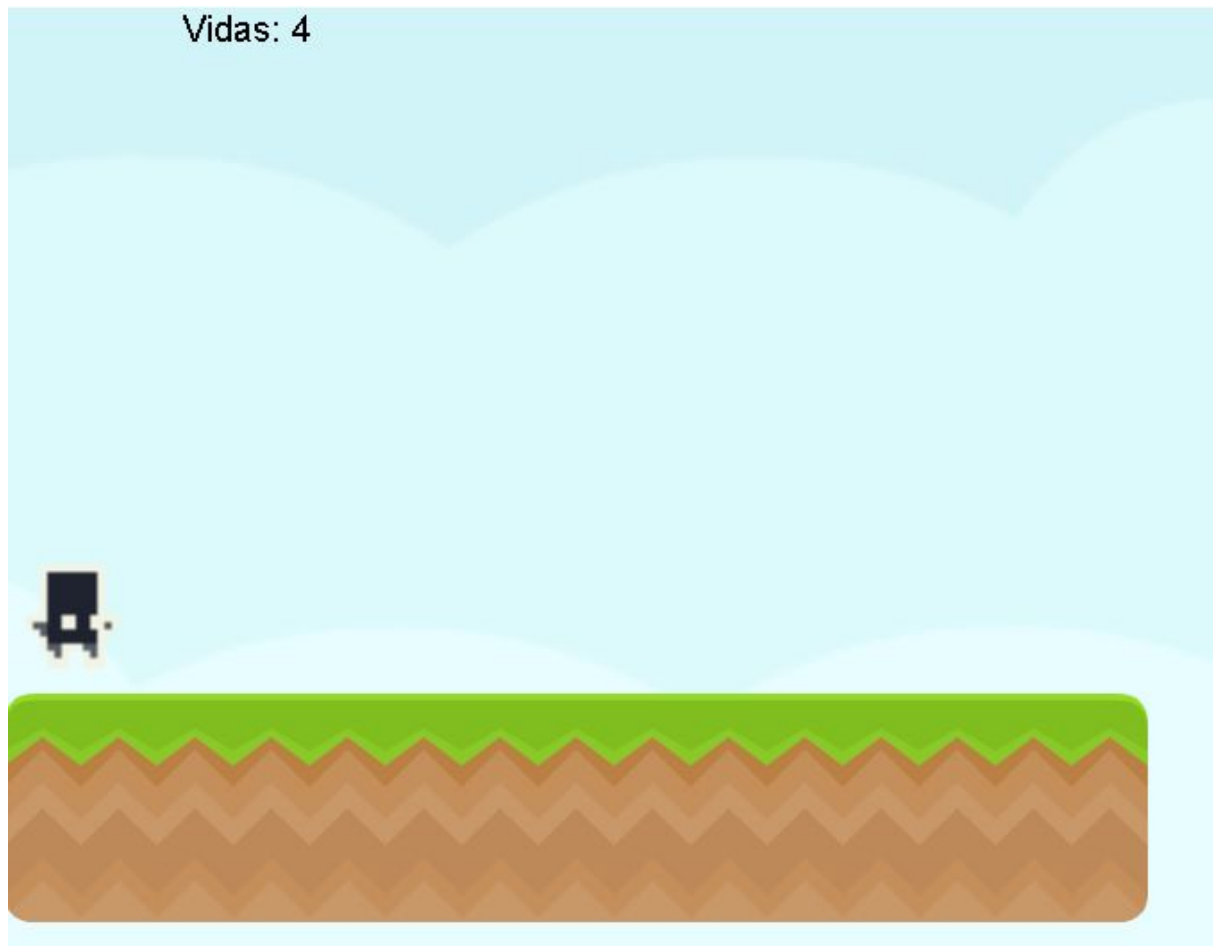
Es decir el poder registrarse y el poder loguearse, al loguearse directamente debe de aparecer la pantalla del juego.

Detalles: No conseguimos colisionar ni lateralmente ni por debajo, es decir, el mapa solo tiene colisiones con la parte superior. Debemos encontrar el fallo pero a la vez avanzar con el proyecto. Así que iremos buscando la solución del fallo mientras seguimos avanzando en el juego. Los mapas a partir de ahora serán algo más serios, es decir que a la hora de entregar sea de modificar poca cosa o entregarlo de una, que sea un mapa decente.

Opiniones: Phaser.js no parece ser un framework demasiado complicado de aprender, pero aparentemente tiene matices, por ejemplo el fallo de la instancia de phaser, ya que instanciando de la manera que lo hacíamos no mostraba el mapa, pero sin embargo al cambiar solamente la instancia ya se mostró el mapa. Y viendo ejemplos en internet de la página oficial vemos que no hacen nada diferente en el código. pero como hemos visto tantos ejemplos pensamos que alomejor hemos aplastado código... O quizás como en los ejemplos no se ve el mapa ni los Tilesets, quizás sea nuestro mapa el que esté mal pero esta última opción no creemos que sea dado que si fuera eso creemos que no habría ningún tipo de colisión, pero nosotros tenemos la colisión con la parte superior.

Lunes 13-05-2019

Agregamos el texto sobre la vida del personaje, al caer al vacío se le restan las vidas. Comienza con 4 vidas que en realidad son 5, por el hecho de que en la vida 0 te permite seguir jugando.



Al morir, quedaría de hacer una nueva imagen de game over para poder mostrarlo. Seguimos sin arreglar las colisiones laterales.

Miércoles 15-05-2019

Tras mucho tiempo de no encontrar el error de las colisiones, lo encontramos.

Era el "movimiento que hacíamos al personaje", éste más bien se teletransportaba píxel a píxel. lo que al encontrar una colisión este traspasaba la pared. al cambiar el código a `setVelocity` se soluciona el error

```

if(cursors.left.isDown)
{
    player.x-=3.300000003;
    //socket.emit("Jugador-Moviendose-Izq");

    player.anims.play('left',true);
    if(cursors.up.isDown ){

        //socket.emit("Jugador-Moviendose-top");
        this.canJump=false;
        player.y += -6.89000000001;
        player.setVelocityY(-11);
        player.anims.play('jump',true);
    }
}
else if(cursors.right.isDown)
{
    player.x +=3.300000003;
    player.anims.play('right',true);
    //socket.emit("Jugador-Moviendose-Der");
    if(cursors.up.isDown){
        //socket.emit("Jugador-Moviendose-top");
        this.canJump=false;
        player.y += -6.89000000001;
        player.setVelocityY(-11);
        player.anims.play('jump',true);
    }
}
else if(cursors.up.isDown){
    //socket.emit("Jugador-Moviendose-top");
    this.canJump=false;
    //player.y += -6.89000000001;
    player.setVelocityY(-11);
    player.anims.play('jump',true);
}
}

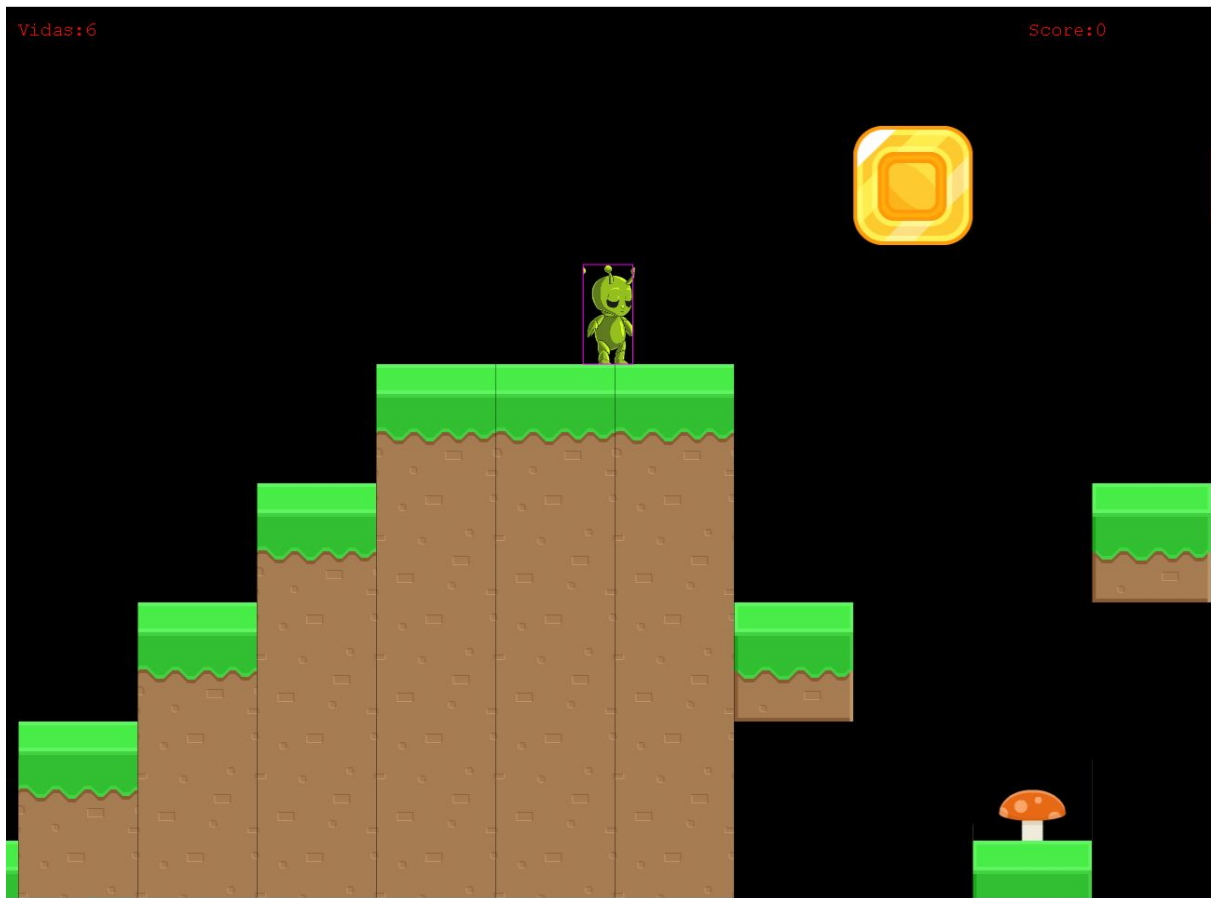
```

Ahora procedemos a crear monedas y su posterior puntuación.

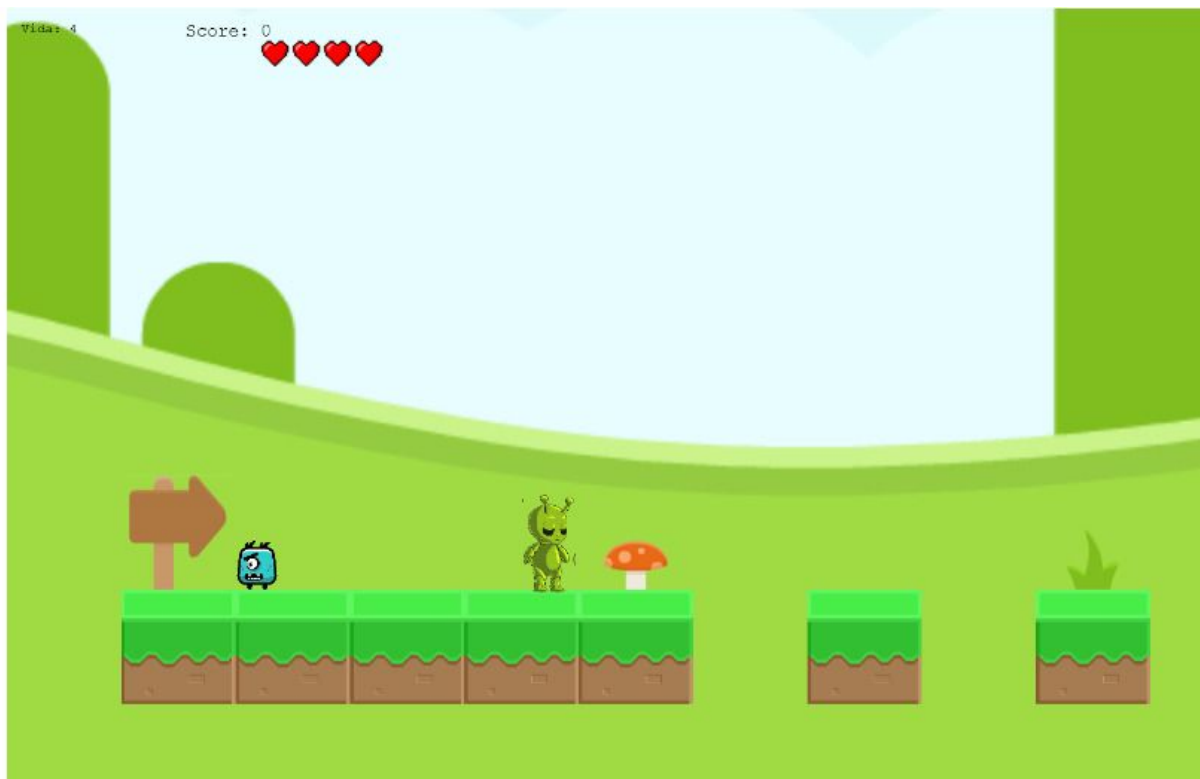
Al buscar meter puntuaciones en este juego, lo que a la hora de meter objetos ayudaría a hacer otras cosas como coger una llave para poder entrar por una puerta...

17-05-2019

A día de hoy ya conseguimos meter texto de puntuación guardarla y cuando muera que se reinicie. Puesto que también tenemos monedas en el mapa y se recogen correctamente

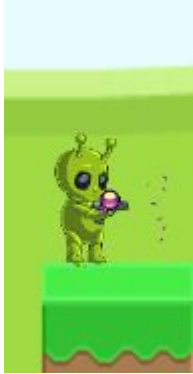


Hemos ingresado enemigos también con movimiento propio.



Como se puede ver en estas fotos, también hemos cambiado el personaje.

Este personaje tiene la propiedad de poder disparar:



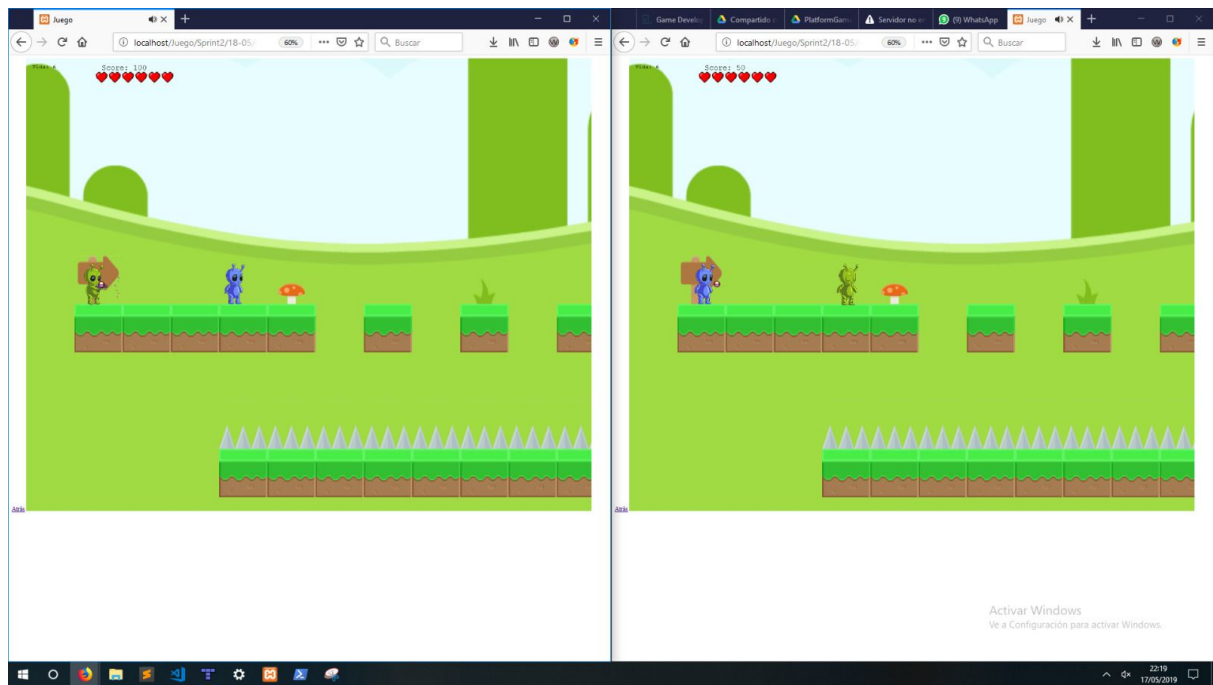
Tenemos datos en un servidor usando sockets. El cual recibe datos de las acciones de los jugadores:

```
izquierda
izquierda
izquierda
izquierda
derecha
derecha
fly
Jugador ha disparado a la Derecha
Jugador ha disparado a la Derecha
Jugador ha disparado a la Derecha
```

Pero esto de vez en cuando nos suelta un warning, que proviene del cdn de sockets. el cual no hemos sabido ni el porqué ni la solución de éste:

⚠ Ha fallado la carga del <script> con origen "<https://cdnjs.cloudflare.com/ajax/libs/socket.io/2.1.1/socket.io.dev.js>".

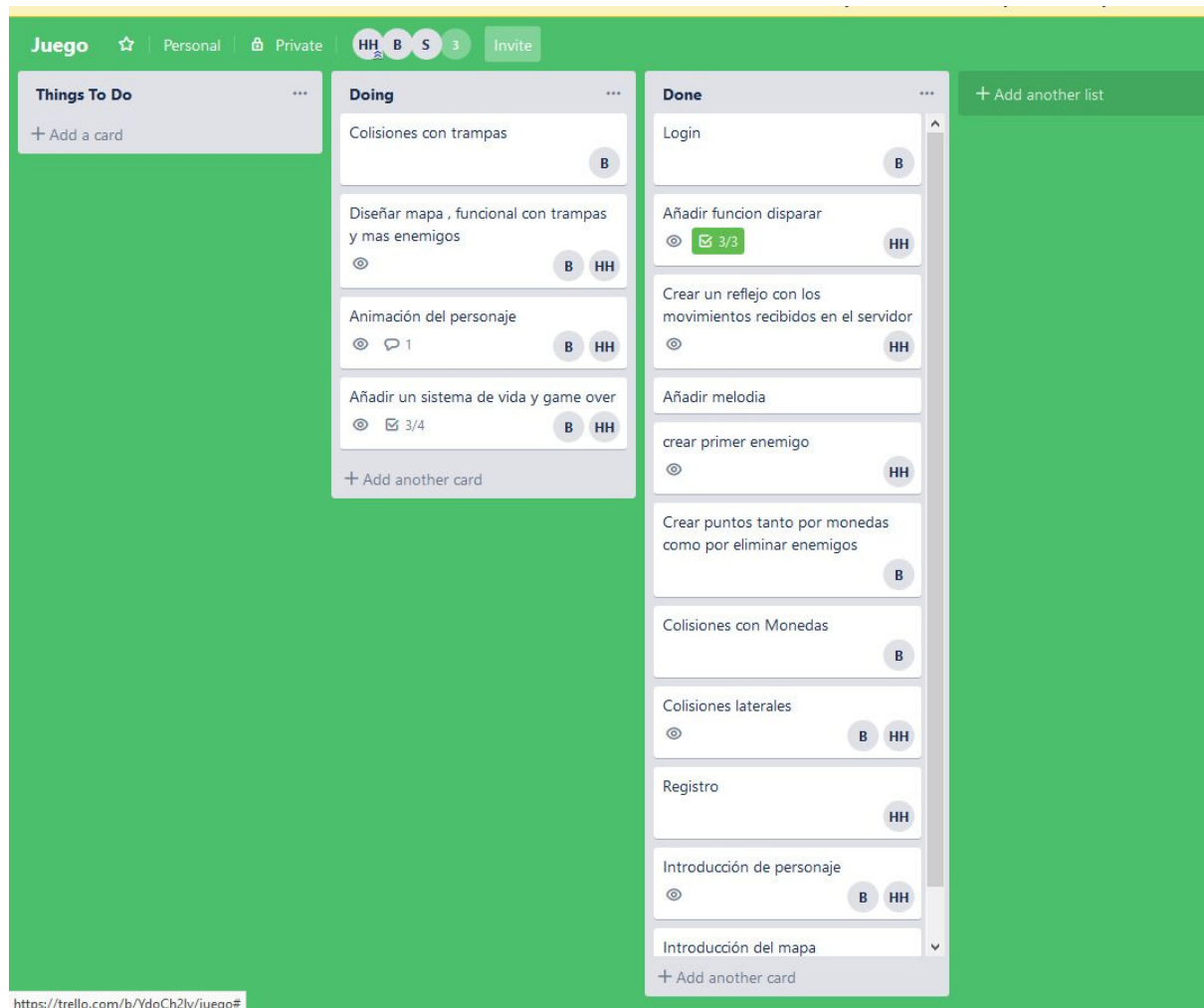
A continuación procedemos a insertar una imagen de los dos jugadores(en un mismo pc)



Nota: Mientras nadie más entre, en nuestra pantalla se verá el segundo jugador pero no tendrá ningún tipo de interacción con el juego, es decir, no podrá ni puntuar ni tendrá colisiones.

Cuando entre un segundo jugador entonces éste podrá tener interacción con el juego.

TRELLO



Nos queda hacer que la colisión se haga bien con las trampas. Funcionar sí funciona pero al ser de 128x128 la muerte se produce antes de poder ver siquiera la colisión entre jugador pincho. Realmente sí hay colisión pero no se llega a apreciar a simple vista ya que con el primer píxel del jugador y de las trampas activa la función de perderVida. Por ello es que seguimos con estas colisiones ya que queremos realizar de una manera más bonita esta colisión. Al igual que con los pinchos, nos pasa lo mismo con el agua y la lava. con el mínimo contacto entre píxeles de jugador y elemento activa la función y no deja apreciar el contacto que buscamos.

Seguimos con la animación del personaje. Porque el SpriteSheet que escogimos era muy grande y tuvimos que hacer escala y al hacer escala cambia los px. Cosa que hace que no cogamos bien frame a frame las fotos para animar al personaje.

La última parte está a medias, nos falta la parte del game over. ya que restamos bien las vidas. pero al llegar a 0 o menor sigue restando y en ese momento queremos cambiar de escena para poder poner una escena de game over.