



Portafolio 5to semestre

Harware – Arduino

Nombre de los integrantes:

Huerta Estaragués Eduardo Damián

Fecha del Documento:
08/09/2024



Índice

Investigación ----- pag. 3 a 13

Progreso ----- pag. 14 a 19

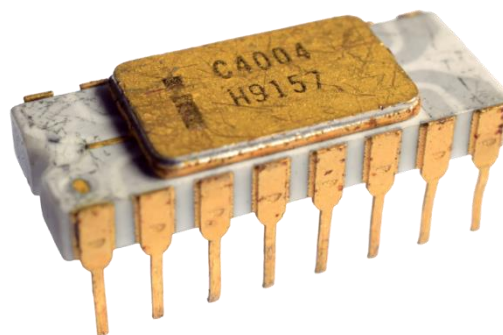
GDD ----- pag 20 a 22

APA ----- pag 23 a 24



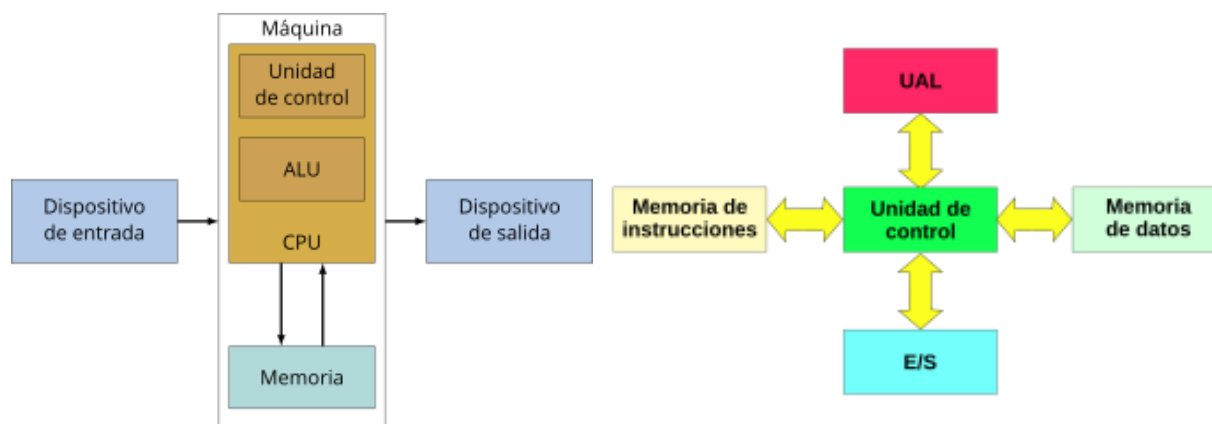
Los Microcontroladores

Estos son Circuitos integrados, los cuales son programables. Siendo capaces de ejecutar programas grabados en una memoria que se encuentra integrada en su cuerpo. Lo especial de estos componentes, es que cuentan con múltiples bloques que cumplen tareas específicas. Siendo la mayor parte de una computadora dependiendo de que tan avanzado sea el controlador (CPU, memoria y periféricos); Estos componentes se suelen usar para el manejo de actividades repetitivas, cíclicas y simples. Pues no cuentan con gran potencia, pero mantienen un bajo consumo energético.



Toral, A., & Toral, A. (2024, 23 febrero). *Primer microprocesador: Intel 4004*. PCBRAPIDO.COM.
<https://www.pcbapido.com/primer-microprocesador-intel->

Los primeros microcontroladores surgieron en 1971, siendo capaces de procesar 4bits. Siendo un gran avance, el cual fue rápidamente por grandes como IBM e Intel. Con un rápido avance en memorias y fabricación, reduciendo el costo de aparatos electrónicos y llevándolo al público general. A la vez que generando arquitecturas y desarrollos como la arquitectura Von Neumann y Arquitectura Harvard.





Arduino

Es un hardware de sistema abierto, que surgió en 2005 en Italia, el cual está conformado por una variedad de “single board microcontrollers” y microcontroladores. Que permiten la construcción de diferentes aparatos y sistemas electrónicos. A la vez que una variedad de periféricos y sensores, para cumplir con cualquier proyecto que se busque cumplir.

Siendo libres, todos tienen acceso al software y hardware que lo conforman. A la vez que permite un amplio y constante desarrollo en ambos ámbitos. Poniendo a disposición múltiples variantes del sistema, para cumplir con múltiples usos y funciones.

Para la programación de estos sistemas se usa una variedad de lenguajes, desde C, C++ a la vez que una API standard (Arduino programming Language, basado en Processing IDE)



Componentes Arduino a emplear

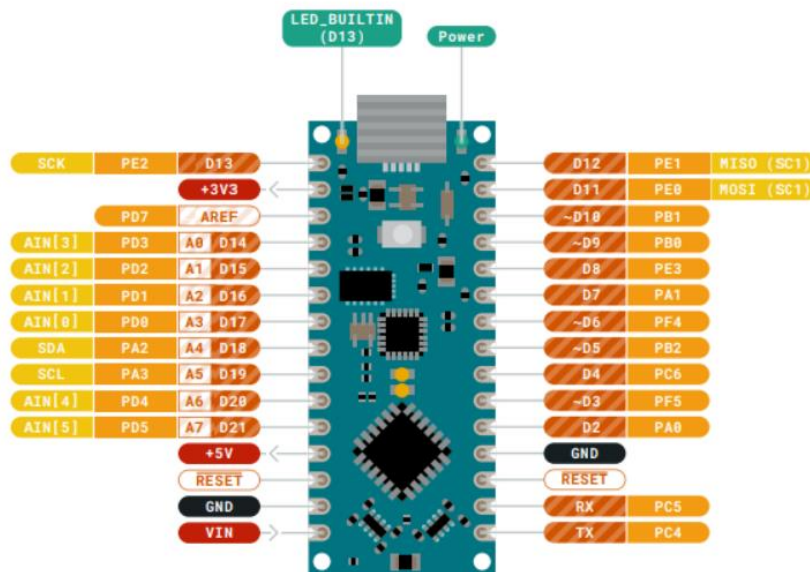
- Nano Every
 - ATmega4809
 - Processor
 - AVR CPU at up to 20 MHz
 - 48 kB Flash
 - 6 kB SRAM
 - 256B EEPROM
 - Power On Reset (POR) and Brown Out Detection (BOD)



- Peripherals
 - 1x 16-bit Timer/Counter with a dedicated period register and 3x compare channels
 - 4x 16-bit Timer/Counter with input capture
 - 1x 16-bit Real-Time Counter (RTC) running from an external crystal or an internal RC oscillator
 - 4x USART with fractional baud rate generator, auto-baud, and start-of-frame detection
 - 1x Master/slave Serial Peripheral Interface (SPI)
 - 1x Dual mode Master/Slave TWI with dual address match 6x 16 bit Timers (1 dedicated to RTC)
 - Event System for CPU independent and predictable inter-peripheral signaling
 - Configurable Custom Logic (CCL) with up to four programmable Look-up Tables (LUT)
 - 1x Analog Comparator (AC) with a scalable reference input
 - Watchdog Timer with Window mode, with separate on-chip oscillator
 - External interrupt on all general-purpose pins
- ATSAM11D14A
 - Processor
 - Arm® Cortex®-M0+ at up to 48 MHz
 - 16KB Flash
 - 4KB SRAM
 - Power On Reset (POR) and Brown Out Detection (BOD)
 - One full-speed (12Mbps) Universal Serial Bus (USB) 2.0 interface
 - Embedded device function



- Eight endpoints
 - Standard firmware for USB-UART bridge
 - Can be reprogrammed for other USB classes
 - UART connection to ATmega4809
 - Field upgradeable through USB Bootloader
- MPM3610 (DC-DC)
 - Regulates input voltage from up to 21V with a minimum of 65% efficiency @minimum load
 - More than 85% efficiency @12V
 - AP2112K-3,3 (LDO)
 - Regulates input voltage from 5V to 3.3V
 - Up to 550mA output current for user application (recommended max 200mA)



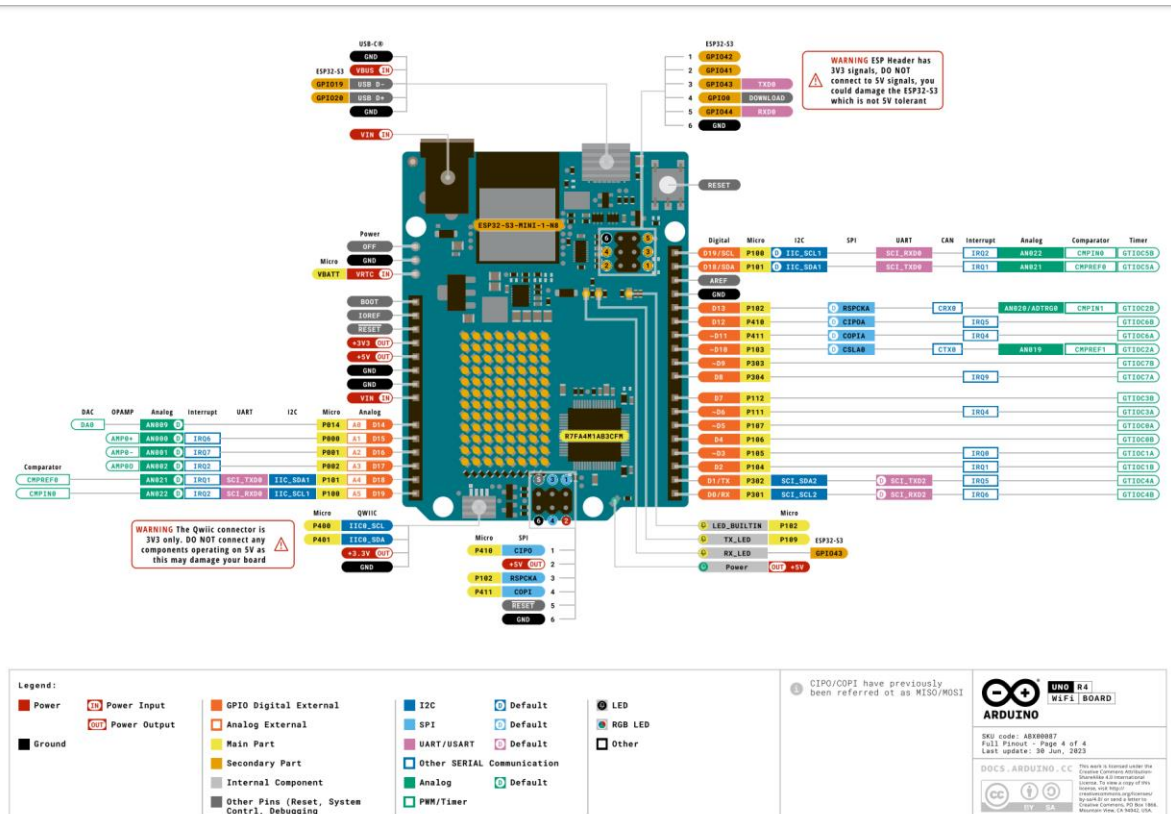
Pinout



- Arduino UNO R4 WIFI
 - R7FA4M1AB3CFM#AA0
 - Overview
 - 48 MHz Arm® Cortex®-M4 microprocessor with a floating point unit (FPU)
 - 5 V operating voltage
 - Real-time Clock (RTC)
 - Memory Protection Unit (MPU)
 - Digital-to-analog Converter (DAC)
 - Memory
 - 256 kB Flash Memory
 - 32 kB SRAM
 - 8 kB Data Memory (EEPROM)
 - Peripherals
 - Capacitive Touch Sensing Unit (CTSU)
 - USB 2.0 Full-Speed Module (USBFS)
 - 14-bit ADC
 - Up to 12-bit DAC
 - Operational Amplifier (OPAMP)
 - Power
 - Operating voltage for RA4M1 is 5 V
 - Recommended input voltage (VIN) is 6-24 V
 - Barrel jack connected to VIN pin (6-24 V)
 - Power via USB-C® at 5 V
 - Communication
 - 1x UART (pin D0, D1)
 - 1x SPI (pin D10-D13, ICSP header)



- 1x I2C (pin A4, A5, SDA, SCL)
- 1x CAN (pin D4, D5, external transceiver is required)
-
- ESP32-S3-MINI-1-N8
 - Overview
 - Xtensa® dual-core 32-bit LX7 microprocessor
 - 3.3 V operating voltage
 - 40 MHz crystal oscillator
 - Wi-Fi®
 - Wi-Fi® support with 802.11 b/g/n standard (Wi-Fi® 4)
 - Bit rate at up to 150 Mbps
 - 2.4 GHz band
 - Bluetooth®
 - Bluetooth® 5



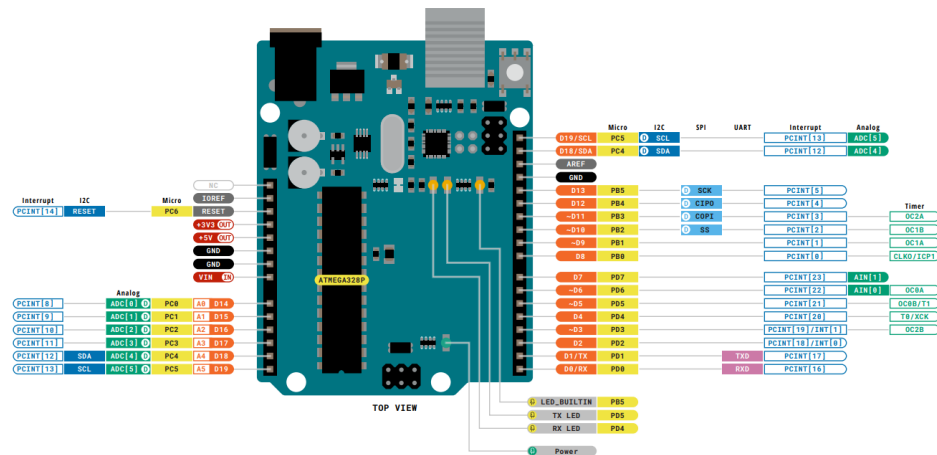


Conexiones:

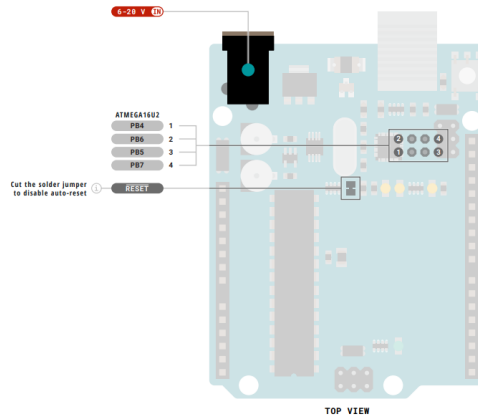
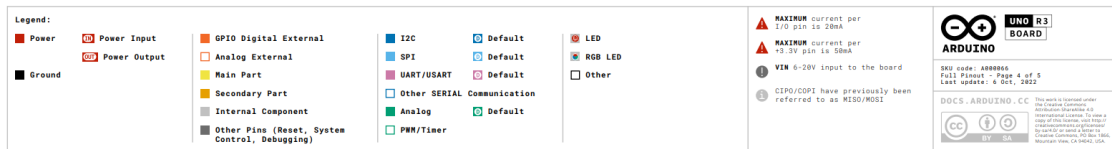
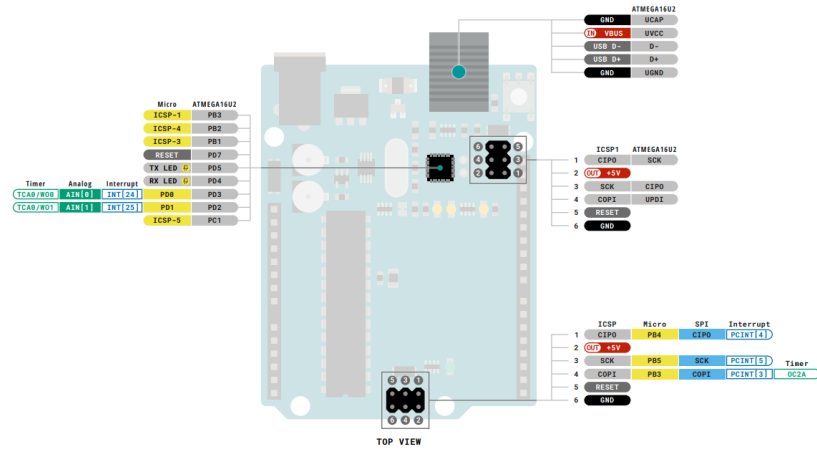
- USB
 - Conexión de entrada, para datos y cargas. Pudiendo ser de tipo A, C, micro USB u otros.
- Conexión Jack
 - Conector para corriente (Preferible 7V)
- Pines VIN y GND
 - Entrada o salida de tensión.
- Pin 5v
 - Alimentación regulada de 5V
- Pin 3,3V
 - Alimentación regulada de 3,3V
- Pin Reset / botón Reset
 - Pin para reiniciar la placa
- IOREF
 - Este es un pin que da la tensión de referencia a la que trabaja la placa
- Entradas Analógicas
 - Permite recibir valores Flotantes mediante convertidor analógico digital con 1024 valores (0-1023).
- Entradas Digitales
 - Permiten recibir valores binarios.
- Pin AREF
 - Introducir una tensión de referencia menor o igual a 5v para sensores que trabajen a tensiones más pequeñas y así lograríamos tener una lectura correcta de los pines de entrada analógicos.
- PIN SDA y SCL
 - Estos conectan dispositivos I2C (Transferir datos).



- Pines 13 a 10
 - función secundaria, comunicación con SPI (SCK, MISO, MOSI, SS)
- GND
 - Masa o tierra
- ISCP ATMEGA328
 - Permiten programar Arduino o cargarle el bootloader sin necesidad de sacar el chip de la placa, es un método alternativo a la programación clásica desde el USB.
- ISCP ATMEGA16U2
 - Permite programar mediante ICSP el chip Atmega16U2, este chip hace de interface entre el puerto USB y el microcontrolador principal Atmega328.



| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|----|--|
| Legend: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | </ | |
|----------------|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|----|--|



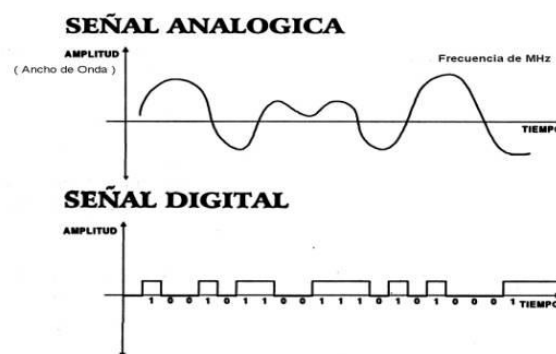
Tipos de señales

Digitales

Estas son un tipo de señales eléctricas, las cuales forman una honda fluida. La cual puede tomar cualquier valor. Permitiendo medición de luz, sonido y temperatura.

Analógicas

Estas son un tipo de señales binarias, las cuales permite dar un valor de 1 o 0 (verdadero o falso). Contando con 3 parámetros, Altura (Nivel eléctrico), Duración y frecuencia (Velocidad de cambios por segundo). Estas se emplean como valores booleanos, encendidos y apagados, etc.



Programación y más. (2015, 21 mayo). *Señales analógicas y digitales - Dominio del tiempo* [Video]. YouTube. <https://www.youtube.com/watch?v=Jp41luUgnyA>

NFC Mundo real

- Identificación
 - Tarjetas de acceso
- Servicios
 - Transporte
 - Pago
- Datos
 - Intercambio de datos
 - Sincronización de dispositivos



Techvolt. (2024, 26 febrero). ¿Qué es la tecnología NFC? - Techvolt - Medium. Medium. <https://medium.com/@techvolt/qu%C3%A9-es-la-tecnolog%C3%ADa-nfc-0343a99d6e02>



- Entretenimiento
 - Figuras interactivas
 - Juguetes reactivos
- Datos
 - Almacenamiento de datos pequeños



Gameplays de PC y Emuladores 4ldoZR@. (2022, 21 marzo).
 GAMEPLAY EMULADOR YUZU (YA SE PUEDE UTILIZAR LOS AMIIBOS EN
 EL JUEGO DE SUPER SMAS BROS ULTIMATE) [Video]. YouTube.
https://www.youtube.com/watch?v=Dpq_v__AVHg



Proyecto hasta este punto

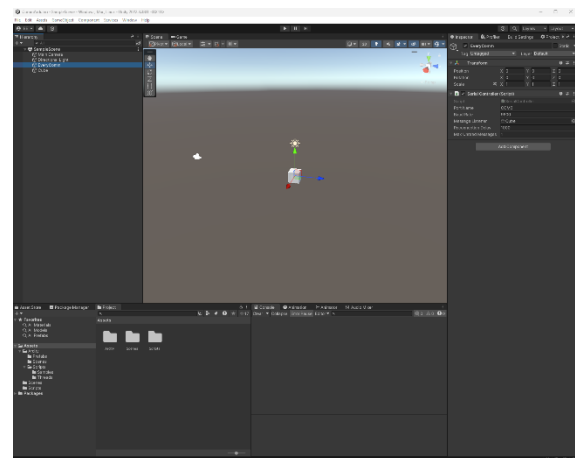
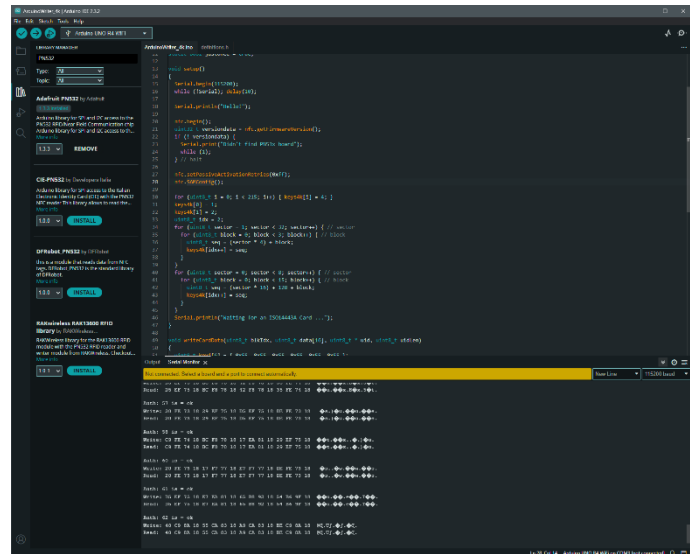
Lectura de NFC y conexión con Unity (Sem-2)

Empleando un módulo PN532 NFC y un Arduino UNO R4 WIFI, se agregó la librería de “Adafruit PN532” permitiendo la lectura de elementos NFC.

A la vez que se logró conectar el programa de Arduino con el paquete “Ardity.untypackage”. El cual nos permitirá introducir nuestros modelos en tarjetas NFC y activas animaciones con la misma estrategia.

En clase se acordó un modelo de tamaño 12*12*12, para permitir la integración de colores en los modelos. Sin embargo, el planteamiento del proyecto sigue siendo

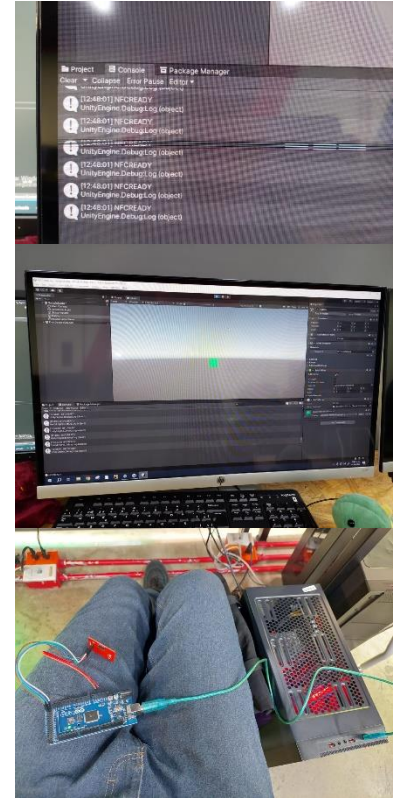
acordado. En lo personal se propone la idea de tener 2 lectores, para poder tener uno que sea el personaje y sin removerla poder generar ataques, en un juego de combate por turnos. Inspirado por juegos como Dragon City o Pokémon.



Datos NFC (Sem-4)

Empleando el programa “Magica” y “Goxel”, generamos modelos los cuales se exportaron en formato de texto, sin embargo, para poder leer y escribir estos datos ocupamos darles un nuevo formato. Esto se realizó con un script que nos proporcionó el profe, en el que se agrega el nombre del modelo, a la vez que los 3 valores de las posiciones del cubo y su color.

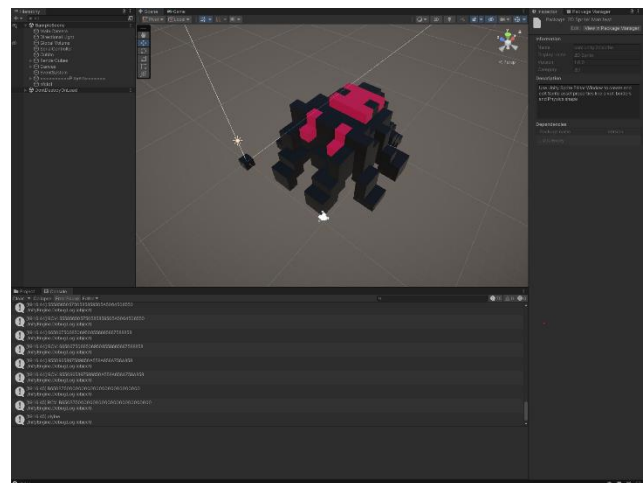
Con el nuevo formato, lo podemos escribir en la tarjeta gracias al script “ArduinoWriter_4k”, por lo que usando un sistema similar debemos leer y pasar los valores a Unity. Por lo que usamos “Ardity.unityEngine” y “ArduinoSender_4kBYH”.



Modelos y Color NFC (Sem-7)

Si integro un código el cual estará recibiendo los datos que estará mandando Arduino por el COM, y posteriormente lo pasa a otro código que los descifra para poder instanciar los cubos de nuestro modelo en el mundo.

Descifrar tanto la posición, como el color de estos cubos. A la vez que se implementó el principio de la lógica, para poder posicionar y rotar los modelos del jugador uno y jugador 2.



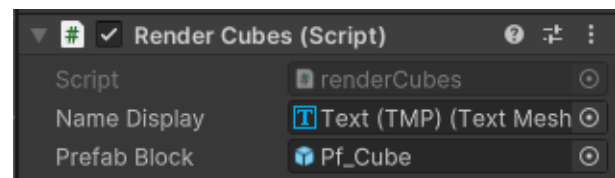


```

13 ^ public class renderCubes : MonoBehaviour
14 {
15     [SerializeField] TextMeshProUGUI nameDisplay; // Text (TMP) (TextMeshProUGUI)
16     [SerializeField] GameObject prefabBlock; // Pf_Cube
17
18     private List<SPersonaje> mObjetos = new List<SPersonaje>();
19
20     private Color[] paleta = new Color[] {...};
38
39     // Frequently called 1 usage
40     public List<SPersonaje> getCreatedObjects()
41     {
42         return mObjetos.Count > 0 ? mObjetos : null;
43     }
44
45     // Event function
46     void Start()
47     {
48         nameDisplay.text = "-*-";
49
50         // 1 usage
51         public void setModelToRender(sModel cubes)
52         {
53             string namme = "nfcio" + (mObjetos.Count + 1);
54             GameObject objToSpawn = new GameObject(namme);
55
56             nameDisplay.text = cubes.name;
57             mObjetos.Add(new SPersonaje(){ mModel = objToSpawn, mName = cubes.name});
58
59             for (int i = 0; i < cubes.cubeCount; i++)
60             {
61                 sCubeData cbdata = cubes.cubes[i];
62                 Vector3 cbpos = new Vector3(cbdata.x, cbdata.z, -cbdata.y);
63                 GameObject voxelcube = Instantiate(prefabBlock, cbpos, Quaternion.identity);
64                 voxelcube.GetComponent<MeshRenderer>().material.color = paleta[cbdata.C];
65                 voxelcube.transform.SetParent(transform);
66             }
67         }
68     }

```

Este es el código que recibe los datos de los cubos y genero el cubo en el mundo, con la posición y color acorde a los datos de la tarjeta escaneada. A la vez que se le usa para mostrar el nombre del modelo.





```

C# renderCubes.cs  C# modelRecv.cs  C# SerialController.cs  C# Players.cs x
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  2 asset usages
6  public class Players : MonoBehaviour
7  {
8      [SerializeField] renderCubes cubesScript;  RenderCubes (renderCubes)
9
10     private List<Personaje> mPersonajes;
11     [SerializeField] List<Vector3> positions;  Serializable
12     [SerializeField] List<Vector3> rotations;  Serializable
13
14     private int mCuantos = 0;
15
16     Event function
17     void Update()
18     {
19         mPersonajes = cubesScript.getCreatedObjects();
20         if (mPersonajes == null) { return; }
21         if(mPersonajes.Count > mCuantos)
22         {
23             mCuantos = mPersonajes.Count;
24             for (int i = 0; i < mPersonajes.Count; ++i) {
25                 Transform egoTr = mPersonajes[i].mModel.transform;
26                 egoTr.position = positions[i];
27                 egoTr.eulerAngles = rotations[i];
28             }
29
30             if (mPersonajes.Count > 0) {
31                 Transform egoTr = mPersonajes[0].mModel.transform;
32                 if (Input.GetKey(KeyCode.X)) { egoTr.position += new Vector3(1, 0, 0); }
33                 if (Input.GetKey(KeyCode.Z)) { egoTr.position -= new Vector3(1, 0, 0); }
34             }
35             if (mPersonajes.Count > 1) {
36                 Transform egoTr = mPersonajes[1].mModel.transform;
37                 if (Input.GetKey(KeyCode.M)) { egoTr.position += new Vector3(1, 0, 0); }
38                 if (Input.GetKey(KeyCode.N)) { egoTr.position -= new Vector3(1, 0, 0); }
39             }
40         }
41     }

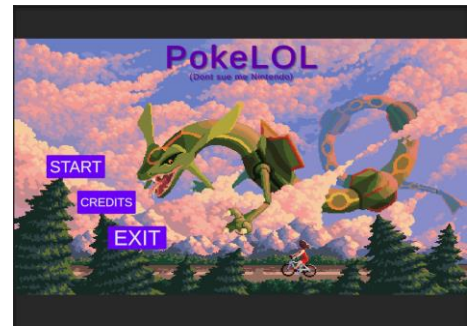
```



“Player” es el controlador de los usuarios y sus parámetros. Desde donde se encuentran en el mundo, sus estadísticas, estado, tipo de bala y maneja el llamado de victoria.

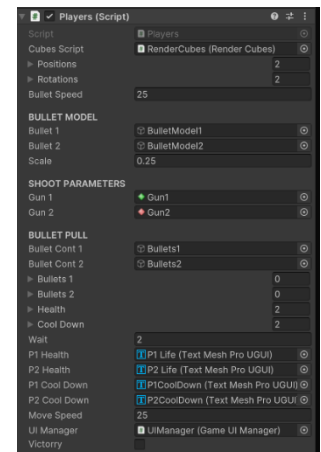
Sem 9

Esta es la entrega final, y se verán los resultados obtenidos. Se implementó una interfaz a la vez que integrando un sistema para poder guardar múltiples modelos a la vez que se generó un sistema de pull de objetos para un sistema de disparos y se han sentado las bases para poder generar múltiples ideas y un juego mucho más grande y ahora con ver lo que se y lo que es posible con estos sistemas, tengo ideas más grandes y más claras para mejores juegos y sistemas más divertidos. A la vez que me quedo con la curiosidad y querer mejorar los sistemas actuales para poder integrar los statos en las mismas tarjetas y generar un juego completo.



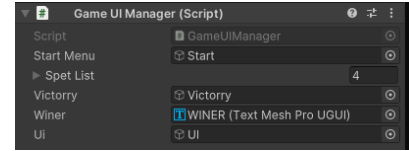
Algunas de las ideas que me surgen ahora son RPG, o un juego en 3ra persona. Pudiendo escoger un personaje y al morir tener que cambiarlo, a la vez que cargar un objeto para equipar. Pero para lograr esto, tendría que pulir sistemas y generar nuevos para poder integrar nuevos parámetros en las tarjetas, como lo es su tipo, sus propiedades, entre otros.

Si bien es algo que se buscará integrar fuera de esta última entrega, ya no tendrá acceso en este proyecto. En este caso se generó un sistema simple, para demostrar los aprendizajes pasados y de la integración de Hardware. Por muy simple que sea, el valor principal que encuentro en este proyecto es haber logrado solucionar y afrontar los problemas presentados.





Se integro, un player sistema. Que en este caso maneja ambos players desde un solo objeto. A la vez que se integró sistemas para manejar la interfaz, mostrando el estado de carga de los objetos, aunque noto falta un sistema que muestre el estado o % de carga directo en la ventana. Lo demás ha permanecido mayormente igual.



GDD

Este escrito planteara el desarrollo de un juego con el título "...", el cual tendrá como mecánica principal el uso e integración de recursos hardware externos para la interacción con el programa, esto incluye tarjetas(Personajes y ataques) y un lector de estas tarjetas (Portal).

Este será un juego semejante juegos como Skylanders, Disney Infinity o Amibos. Pero se diferenciará pues, los modelos estar en la tarjeta, permitiendo agregar nuevos personajes y modelos sin requerir descargas externas y con mayor desarrollo se podrían implementar para poder cambiar mecánicas entre otras acciones posibles al leer datos de estas tarjetas.



(Skylanders, s. f.)



(Abarca, 2022)

Se cuenta con Arduino como plataforma de hardware.

Tarjetas de 4kb, para los personajes y tarjetas de 1kb para modelos más pequeños. Por lo que se tendrán personajes de 16*16*16 cuadros. Y modelos de

12*12*12 en tarjetas de 1K. Pudiendo también llegar a

generar sistemas para introducir animaciones simples, datos de personaje, llaves para interactuar o desbloquear áreas o juegos.

Para la interfaz se generará una vista lateral a 10°, para brindar profundidad a la escena. La escena encuadra a los modelos en la porción superior de la pantalla, a la vez que se deja 1/3 de la pantalla disponible para colocar opciones de ataque, parámetros de los jugadores, curas, etc.



Se tendrá un área de combate, donde se podrán enfrentar 2 usuarios u contra la máquina. A la vez que se integraría un área de mejoras donde el usuario puede comprar algunas mejoras para su personaje, desde defensas, hasta incrementadores de daño.



El ciclo de juego se basaría en pelear, ganando dinero dependiendo si gana el usuario u el 2P (maquina o usuario). Con ese dinero se puede comprar mejoras y poder volver a jugar.

El ciclo de pelea tendría una página donde se puede escoger 1 o 3 mejoras para ese combate. Si es un segundo usuario este podrá escoger una mejora que haya comprado el usuario 1, en caso de ser la maquina se asignará uno de forma aleatoria dentro de los que tengan valores más altos.

La compra de mejoras tendrá múltiples disponibles que afectaran el valor de los parámetros al aplicar multiplicadores. Y potenciar ataques con posibilidades de ataques críticos o ataques dobles.

El enfoque principal seria la venta y colección de las tarjetas, las cuales cuentan con artes de los personajes. Semejante a cartas de Magic, Pokémon, entre otros. Teniendo cartas de personajes, ataques, efectos, etc.



(Wyche, 2023)



Tiempos

| Semana | Actividades |
|--------|---|
| 1ra | Planteamiento de proyecto y problemas |
| 2da | Generar lectura de tarjetas NFC Conexión Arduino – Unity |
| 3ra | Escritura de Modelos 3D y animaciones |
| 4ta | Generación de Sistema de lectura de modelos en Unity |
| 5ta | Generación de sistemas básicos de juego |
| 6ta | Refinamiento de sistemas de juego |
| 7ma | Integración de animaciones |
| 8va | Generación de Interfaz |
| 9na | Generación de Hardware |
| 10ma | Pulido de proyecto y Muestra |
| 11va | Entrega y análisis |



Apa

1.3.3 Tipos de señales (Analógica-Digital). (s. f.).

http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro27/133_tipos_de_seales_analgicadigital.html

Abarca, L. G. (2022, 11 marzo). *Royal Frontier, estrategia por turnos para PC y consolas • Consola y Tablero*. Consola y Tablero. <https://consolaytablero.com/2022/03/11/royal-frontier-pc-consolas/>

Arduino. (2024). *Arduino*. Arduino DOCS. Recuperado 18 de julio de 2024, de <https://docs.arduino.cc>

Carla-Lp. (2023, 12 julio). ¿Cómo utilizar la tecnología NFC para mejorar el engagement de tus clientes? *Logopost*. <https://logopost.es/como-utilizar-la-tecnologia-nfc/>

colaboradores de Wikipedia. (2024b, abril 19). *Arquitectura de von Neumann*. Wikipedia, la Enciclopedia Libre. https://es.wikipedia.org/wiki/Arquitectura_de_Von_Neumann

colaboradores de Wikipedia. (2024c, mayo 13). *Arquitectura Harvard*. Wikipedia, la Enciclopedia Libre. https://es.wikipedia.org/wiki/Arquitectura_Harvard

colaboradores de Wikipedia. (2024d, junio 5). *Microcontrolador*. Wikipedia, la Enciclopedia Libre. [https://es.wikipedia.org/wiki/Microcontrolador#:~:text=Un%20microcontrolador%20\(abreviado%20%C2%B5C%2C%20UC,que%20cumplen%20una%20tarea%20espec%C3%ADfica.](https://es.wikipedia.org/wiki/Microcontrolador#:~:text=Un%20microcontrolador%20(abreviado%20%C2%B5C%2C%20UC,que%20cumplen%20una%20tarea%20espec%C3%ADfica.)

Computer, K., & Computer, K. (2023, 23 mayo). *El surgimiento revolucionario: el primer microprocesador de la historia | Kitsune Computer / Fabricamos servidores NAS*. Kitsune Computer. <https://www.kitsunecomputer.com/el-surgimiento-revolucionario-el-primer-microprocesador-de-la-historia/>

Cto. (2021, 27 enero). ¿Que es Arduino? Capacitación Técnica. <https://capacitaciontecnica.com.ar/2019/11/11/que-es-arduino/>

Etapé, J. A. P. (2018, 25 febrero). ¿Qué amiibos desbloquean más extras en Nintendo Switch? *Hobby Consolas*. <https://www.hobbyconsolas.com/reportajes/que-amiibos-desbloquean-mas-extras-nintendo-switch-189988>



Morales, J. A. R. (2021, 15 mayo). *Pines y conectores de Arduino UNO*. PASIÓN ELECTRÓNICA.

<https://pasionelectronica.com/pines-y-conectores-de-arduino-uno/>

Penalva, J. (2024, 17 mayo). *NFC: qué es y para qué sirve en este 2024*. Xataka.

<https://www.xataka.com/basics/nfc-que-es-y-para-que-sirve>

Skylanders, C. T. W. (s. f.). *Skylanders: Giants*. Wiki Skylanders.

https://skylanders.fandom.com/es/wiki/Skylanders:_Giants

Wikipedia contributors. (2024b, julio 10). *Arduino*. Wikipedia. <https://en.wikipedia.org/wiki/Arduino>

Wyche, J. (2023, July 28). *The most valuable Gen 1 Pokemon cards*. TheGamer.

<https://www.thegamer.com/pokemon-tcg-most-expensive-generation-1-cards/>