FORMACIÓN PROFESIONAL

CURSO DE PRÁCTICA INTENSIVA

CUADERNO DE INFORMES

FORMACIÓN PROFESIONAL

CFP/UCP/ESCUELA: SENATI CHINCHA

ESTUDIANTE: ABURTO ACEVEDO JHOSTYN

ALBERTO ID: 001444450 BLOQUE: 40PIADS601

CARRERA: INGENIERÍA DE SOFTWARE CON IA

INSTRUCTOR: JHON EDWARD FRANCIA MINAYA

SEMESTRE: VI DEL: 15/06/24 AL: 29/11/24

INSTRUCCIONES PARA EL USO DEL CUADERNO DE INFORMES

1. PRESENTACIÓN.

El Cuaderno de Informes es un documento de auto control, en el cual el estudiante, registra diariamente, durante la semana, las tareas, operaciones que ejecuta en su aprendizaje, es un medio para desarrollar la Competencia de Redactar Informes.

2. INSTRUCCIONES PARA EL USO DEL CUADERNO DE INFORMES.

2.1 En la hoja de informe semanal, el estudiante registrará diariamente los trabajos que ejecuta, indicando el tiempo correspondiente. El día de asistencia registrará los contenidos que desarrolla. Al término de la semana totalizará las horas.
De las tareas ejecutadas durante la semana, el ESTUDIANTE seleccionará la tarea más significativa (1) y él hará una descripción del proceso de ejecución con esquemas, diagramas y dibujos correspondientes que aclaren dicho proceso.

2.2 Semanalmente, el Instructor revisará y calificará el Cuaderno de Informes haciendo las observaciones y recomendaciones que considere convenientes, en los aspectos relacionados a la elaboración de un Informe Técnico (letra normalizada, dibujo técnico, descripción de la tarea y su procedimiento, normas técnicas, seguridad, etc.

2.3 Escala de calificación vigesimal:

CUANTITATIVA	CUALITATIVA	CONDICIÓN
16,8 – 20,0	Excelente	Aprobado
13,7 – 16,7	Bueno	
10,5 – 13,6	Aceptable	
00 – 10,4	Deficiente	Desaprobado

PLAN ESPECÍFICO DE APRENDIZAJE (PEA) SEGUIMIENTO Y EVALUACIÓN

Llenar según avance

Nº	OPERACIONES/TAREAS	OPERACIONES EJECUTADAS*				OPERACIONES POR EJECUTAR	OPERACIONES PARA SEMINARIO
		1	2	3	4		
01	**Desarrollo de modelos IA**						
02	**Desarrollo de modelos Machine Learning**						
03	**Desarrollo de modelos IA con WATSON**						
04	Desarrollo de modelos con WATSON Studio						
05	Muestra información usando Master-detail and Drawer Navigation						
06	**Personaliza ListView**						
07	**Utiliza MVVM**						
08	**Consume servicios REST**						
09	**Publica aplicaciones**						
10	**Calcula costes de uso de servicios en la nube**						
11	**Diseña infraestructura de red en la nube**						
12	**Asegura la conectividad de red en Microsoft Azure**						
13	**Diseña Data Warehouse y Arquitectura BI**						
14	Diseña Transacciones – Auditorias – Power View						
15	**Diseña Dashboard: SQL Server Reporting Services**						
16	**Aplicar en un caso práctico la configuración y uso de almacenamiento de datos con BI**						
17	**Implementa paquete Neuralnet**						
18	**Implementa paquete H20**						

INFORME SEMANAL

VI SEMESTRE **SEMANA N° 10**

	Día	Mes	Año
Del			
Al			

DÍA	TAREAS EFECTUADAS	HORAS
LUNES		
MARTES		
MIÉRCOLES		
JUEVES		
VIERNES	**APRENDEMOS ARDUINO Y LO PROGRAMAMOS CON C++**	6 HORAS
SÁBADO		
	TOTAL	6 HORAS

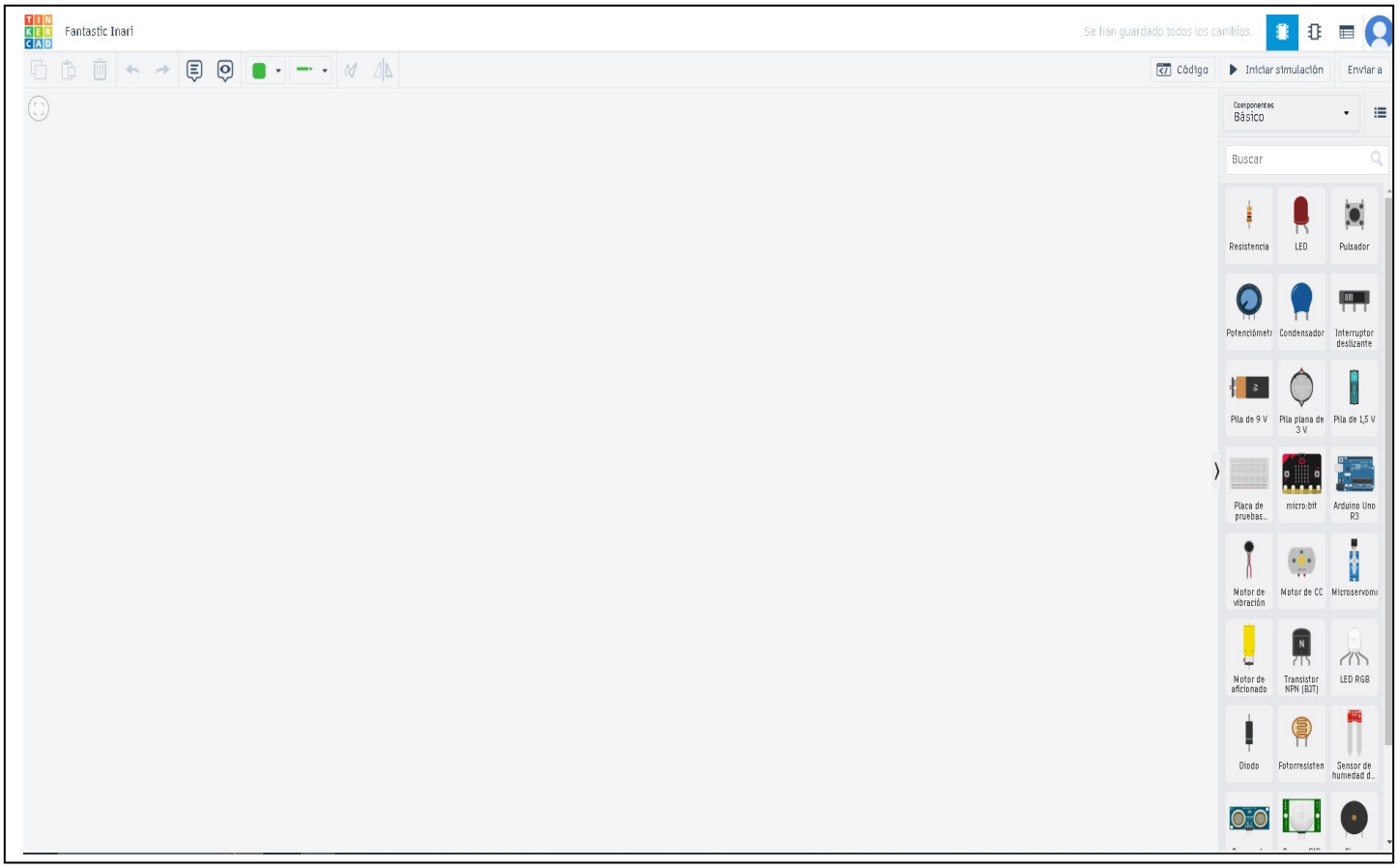
INFORME DE TAREA MÁS SIGNIFICATIVA

Tarea:

SEMAFORO CON CONTROL DE SEGURIDAD

Descripción del proceso:

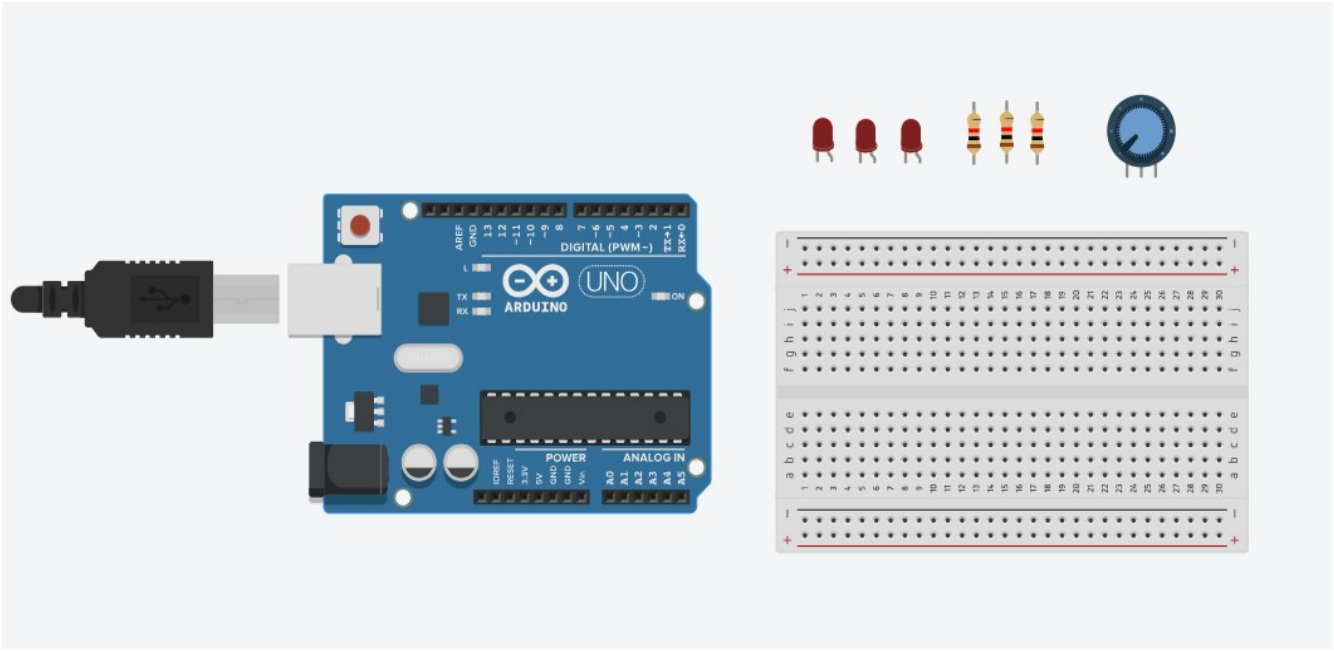
Usando la página TINKERCAD, que nos dará un entorno para usar Arduino.



-

En Arduino realizaremos un semáforo que los tiempos de duración de las luces lo controlaremos con un Potenciómetro.

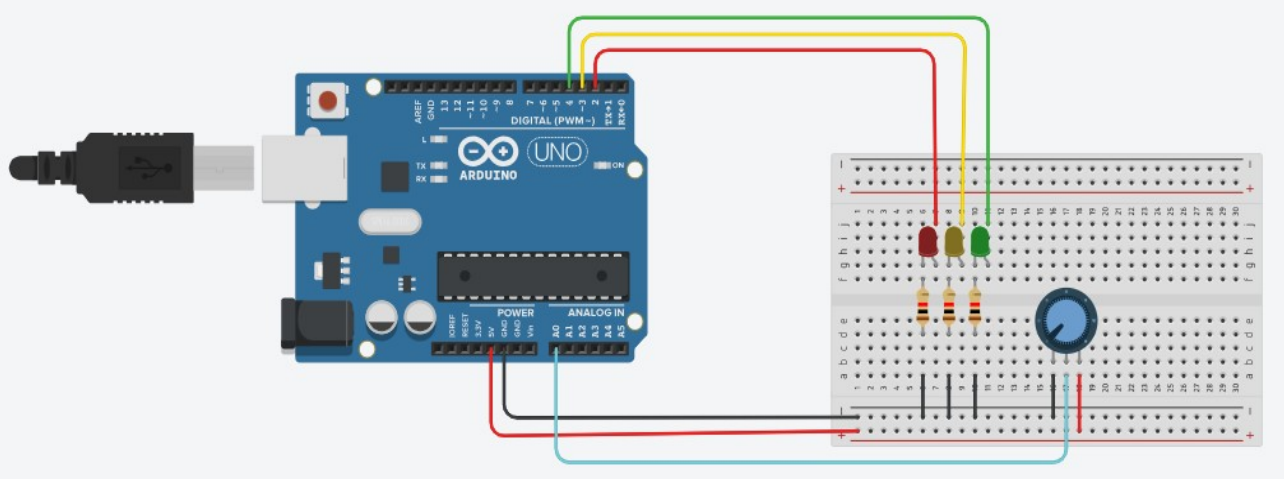
Primero veremos los componentes que usaremos, como el Arduino, placa de prueba, 3 leds, 3 resistencias y un potenciómetro.



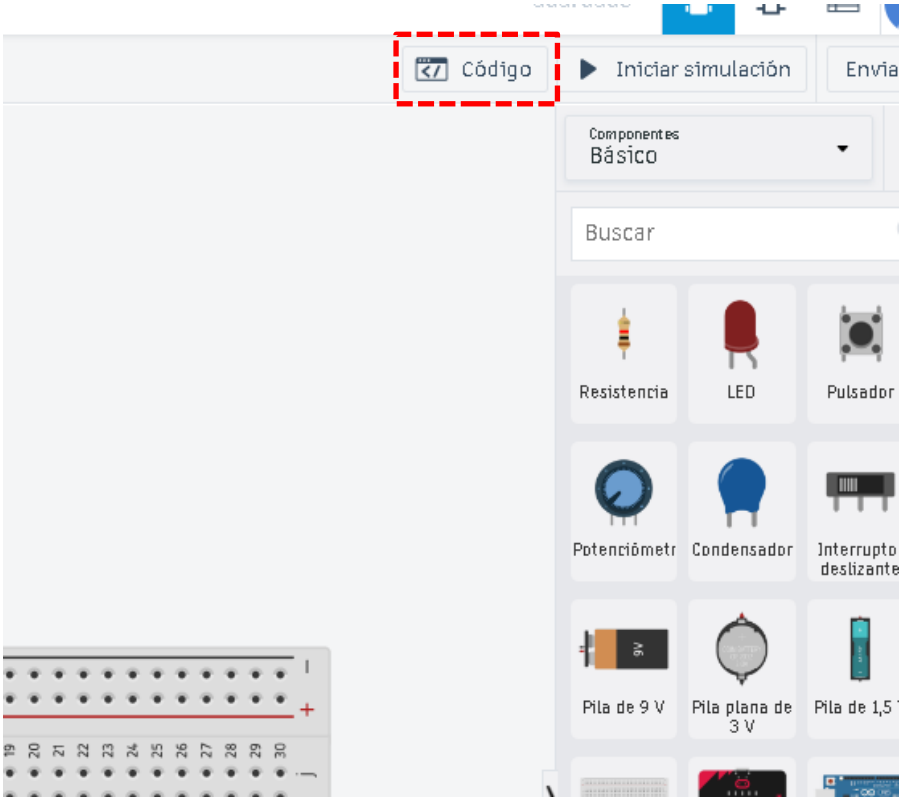
Ahora pondremos los 3 leds en la caja de prueba, le cambiamos de color para simular el semáforo y cada led estará conectado a una resistencia.

Conectamos el Arduino a la placa pasándole energía y tierra.
Conectamos la resistencia a la parte de la tierra y los leds lo conectamos a los pines digitales.
El Potenciómetro también lo conectamos.

Quedando de esta manera.

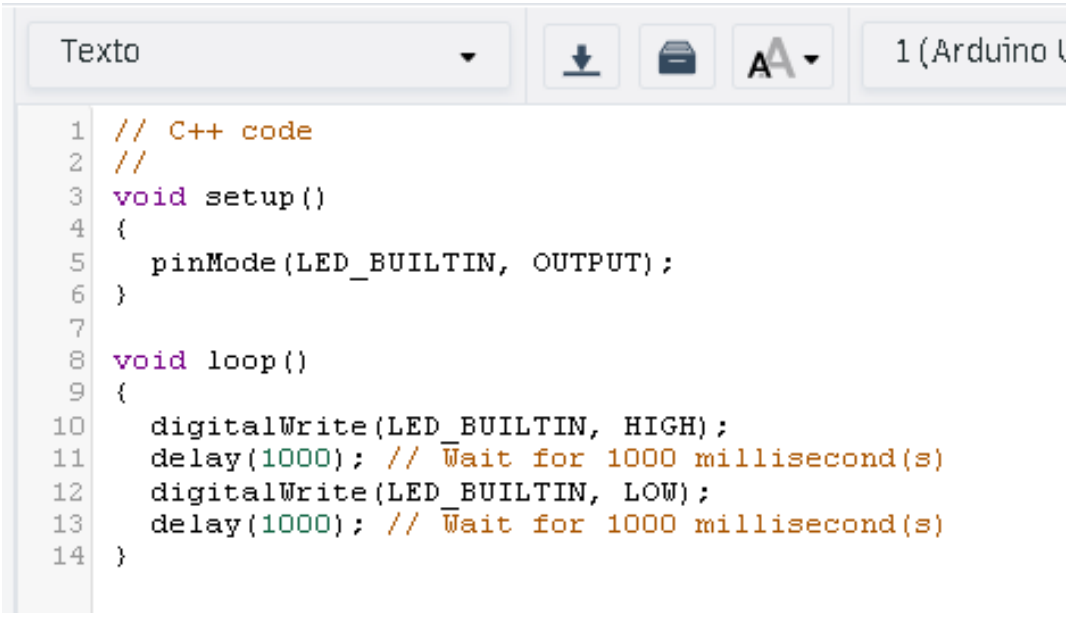


Uno ves que ya conectamos los componentes, en la parte derecha hay una opción que dice código.



Por defecto viene en bloques, pero lo cambiamos a texto.

Y vemos como ya podemos programar nuestro Arduino.



Primero empezamos declarando 2 variables, uno es valorPot que tomara el valor del potenciómetro y el otro es velocidad.

```
1  int valorPot = 0;
2  int velocidad = 0;
```

También declaramos nuestros leds, cada uno en la ubicación de su pin digital y creamos una array llamado vLed[], en este almacenamos el tiempo de encendido de cada led.

```
4  byte ledVerde = 4;
5  byte ledAmarillo = 3;
6  byte ledRojo = 2;
7  int  vLed[] = {6000,3000,7000};
```

En la función setup instanciamos los leds, cada uno con su respectiva variable que creamos previamente.

```
void setup()
{
  pinMode (ledVerde, OUTPUT);
  pinMode (ledAmarillo, OUTPUT);
  pinMode (ledRojo, OUTPUT);
}
```

Acá creamos un método que nos permitirá actualizar los datos del array vLed, así podremos actualizar el tiempo para cada led.

```
void actualizarDatos(int v1, int v2, int v3){
  vLed[0] = v1;
  vLed[1] = v2;
  vLed[2] = v3;
}
```

Y en el método loop, lo que hacemos primero es a la variable valorPot pasarle el valor del Potenciómetro y luego mapeamos este valor entre 1 y 3 y estos nuevos valores lo pasamos a velocidad.

```
void loop()
{
  valorPot = analogRead(A0);
  velocidad = map (valorPot, 0, 1023, 1,3);
```

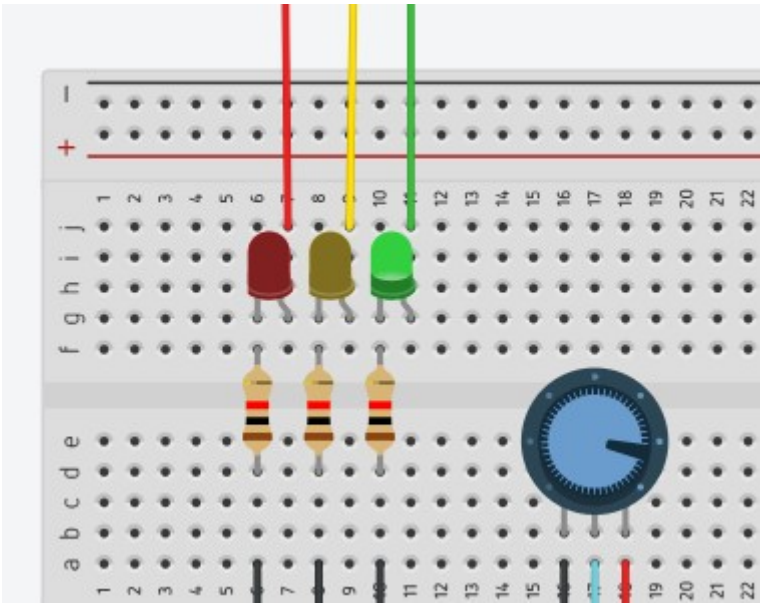
Luego hacemos 3 condiciones, donde si el valor es 1, 2 o 3, se llamará al método actualizarDatos y le pasaremos los nuevos valores de tiempo.

```
if(velocidad == 1){actualizarDatos(6000, 3000, 7000);}
if(velocidad == 2){actualizarDatos(5000, 2000, 6000);}
if(velocidad == 3){actualizarDatos(4000, 1000, 5000);}
```

Y, por último, a cada led lo encendemos y que el tiempo que este encendido dependerá de la velocidad y luego se apagará y se encenderá el siguiente led.

```
digitalWrite(ledVerde, HIGH);
delay(vLed[2]);
digitalWrite(ledVerde, LOW);
digitalWrite(ledAmarillo, HIGH);
delay(vLed[1]);
digitalWrite(ledAmarillo, LOW);
digitalWrite(ledRojo, HIGH);
delay(vLed[0]);
digitalWrite(ledRojo, LOW);
```

Y dependiendo de la velocidad del potenciómetro, se encenderán las luces de distinta duración.



Logramos realizar estos y otro más Arduino en el salón.