



1. Instalación

Descargue Simulide de su página oficial simulide.com haciendo clic en **Downloads**. Puede escoger versiones oficiales o de prueba. Ud. decide. Aquí se está usando una versión de Prueba 1.1.0 (2023/12/21 : Updated to SR2).



Si puede done pues este es un muy buen Simulador e incentivará a su autor a hacer mejoras significativas, sino puede por ahora, ponga en precio 0. le pedirá luego su correo electrónico. luego presione en **Obtener**.

☐ Windows 64
 ☐ Windows 32
 ☐ Linux 64

☐ MacOS

Designa un precio justo:

30-day money-back guarantee.

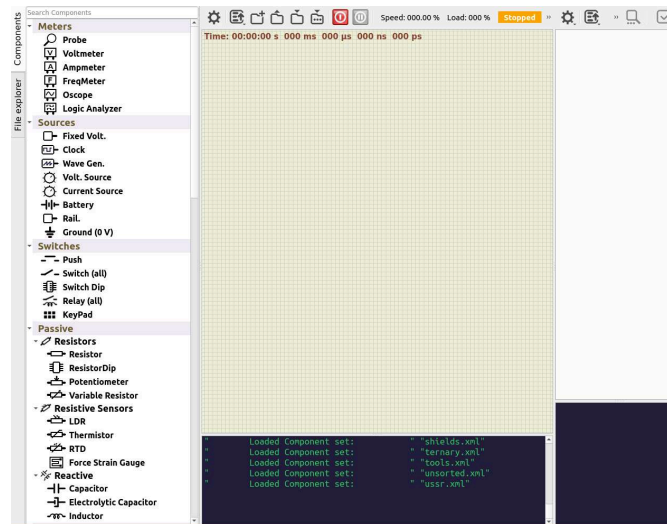
Información del contacto
 Dirección de correo electrónico

(SimulIDE, s.f.).

Luego le aparecerá un enlace de Descarga (**Download**) y también recibirá un mensaje al correo que uso.

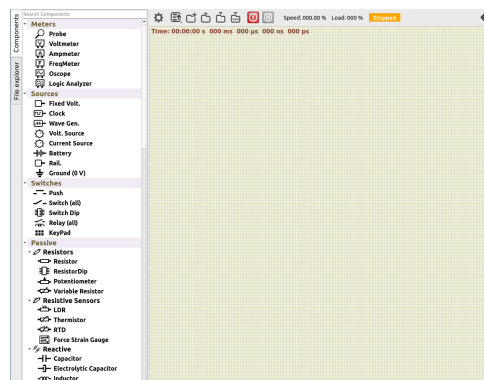
2. Entorno de Desarrollo

La **parte izquierda** es el **Área de Componentes Electrónicos** y también **Explorador de Archivos** , la **parte Central** es el **Área de diseño Electrónico** y la **parte Izquierda** es el **Área de Edición** para escribir el código si es que se usa una Placa como Arduino. La **parte baja central** es el **Área de Mensajes**.



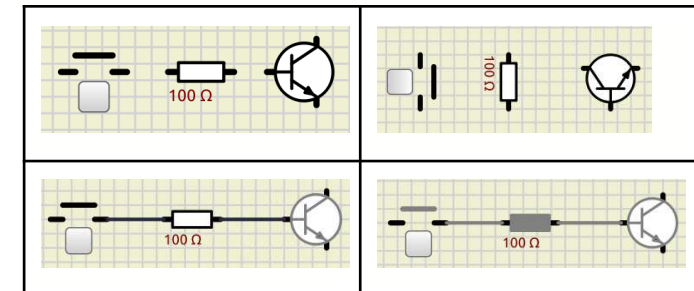
(SimulIDE, s.f.).

Usando el botón izquierdo del mouse y presionando en los bordes de las áreas podemos esconderlas y dejar solamente dos áreas, el Área de Componentes y el Área de Diseño. Así:

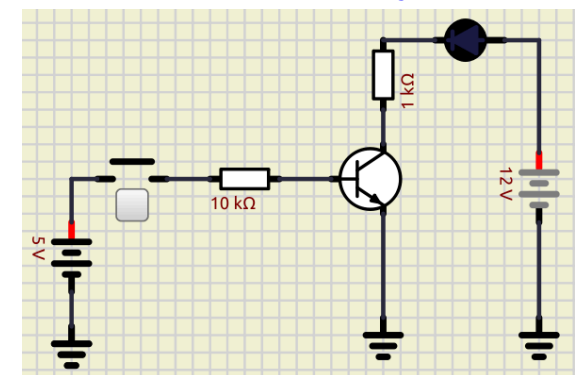


3. Entorno de Desarrollo

Para diseñar se coje un componente del área de componentes electrónicos y **mientras se mantiene presionado el botón izquierdo del mouse se lo arrastra** al área de diseño. **Para rotar** es solo presionar en el componente con el **botón derecho** y le aparecerá un pequeño menú de opciones. **Para generar el cableado** ubíquese en los terminales del componente y el cursor cambiará a (+) y manteniendo el botón izquierdo del mouse podrá crear el cableado, soltando el botón podrá cambiar la ruta del cableado hasta que encuentre el otro contacto del otro componente a unir presionará por última vez el botón izquierdo del mouse y listo. **Finalmente para borrar simplemente presionado el botón izquierdo del mouse arrastre y seleccione el área que quiere borrar** sean componentes o cables y estos cambiarán a un color gris y con la tecla **Suprimir o Delete de su teclado** podra borrarlos. Si se equivoco puede con presionando el botón derecho le saldrá un menú donde podrá deshacer (**Undo**) y le aparecerá de nuevo lo anterior.

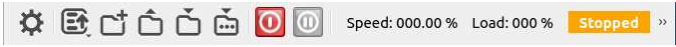


4. Caso 1. Pulsador, Transistor y Led.



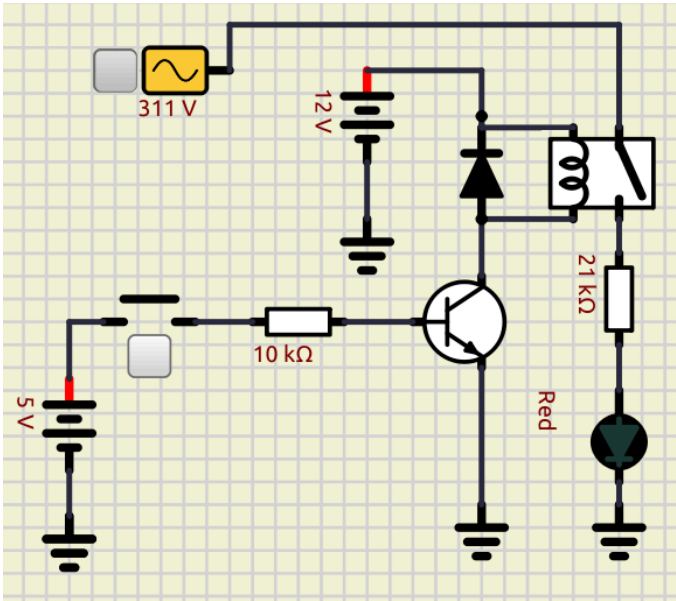
(Oscar Núñez Mori, 2024)

Para ejecutar o correr el código simplemente presionar en el **botón Rojo**. Para el archivo hacerlo en la carpeta que aparece con (...) puntos suspensivos.



Con la **Tecla Windows** y seleccionando los componentes podemos mover todo el circuito en el Área de Diseño.

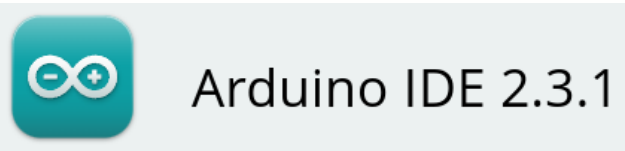
4. Caso 2. Transistor, Relay 12v a 220v.
(No hacer este experimento sin supervisión de un Ingeniero)



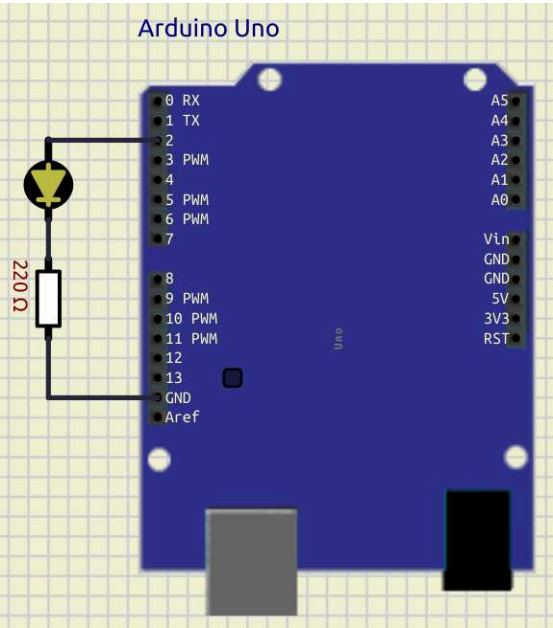
(Oscar Núñez Mori, 2024a)

5. Caso 3. Arduino Uno R3. Parpadeo.

La forma más fácil de programar Arduino Uno R3 en SimulIDE es correr el Programa en [Arduino IDE](#) y luego cargar el archivo (.hex) generado haciendo clic en la Tarjeta Arduino Uno con el botón derecho del mouse o ratón escogiendo la opción **Load Firmware** en SimulIDE.



(Arduino, s.f.)



(Oscar Núñez Mori, 2024b)

5.1 Código Parpadeo (Blink.ino)

```
// Arduino.cc. blink.ino.
// https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
// Modificado por Oscar NÚÑEZ MORI

int led = 2;

void setup() {
  pinMode( led, OUTPUT );
}

void loop() {
  digitalWrite( led, HIGH );
  delay( 1000 );
  digitalWrite( led, LOW );
  delay( 1000 );
}
```

6. Caso 4. Arduino Uno R3, Pulsador y Led

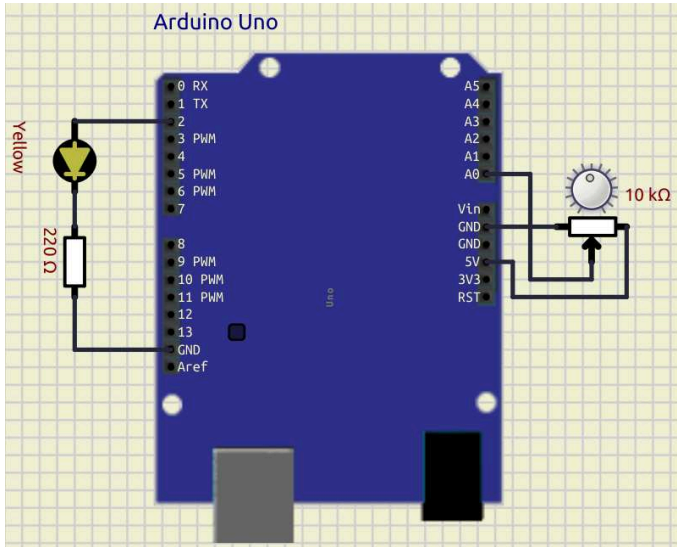
6.1 Código. Entrada Analógica con Potenciómetro.

```
// Arduino. AnalogInput.ino
// https://www.arduino.cc/en/Tutorial/BuiltInExamples/AnalogInput
// Modificado por Oscar NÚÑEZ MORI.
```

```
int pot = 14; // Pin A0. Potenciómetro.
int led = 2; // select the pin for the LED
int valorPot = 0; // Variable almacenada del Pot.

void setup() {
  pinMode( led, OUTPUT ); // led como salida (OUTPUT)
}

void loop() {
  valorPot = analogRead( pot ); // Lee valor del Pot.
  digitalWrite( led, HIGH ); // Prende el led
  delay( valorPot ); // Retardo en milisegundos.
  digitalWrite( led, LOW ); // Apaga el led.
  delay( valorPot );
}
```



(Oscar Núñez Mori, 2024c)

REFERENCIAS

SimulIDE (s.f.). *SimulIDE Circuit Simulator*. <https://simulide.com/>
Arduino (s.f.). *Arduino IDE*. <https://www.arduino.cc/en/software>
Oscar Núñez Mori (2024, Feb 18). *PULSADOR-TRANSISTOR-LED*. <https://github.com/oscar-uni/SIMULIDE/tree/main/PULSADOR-TRANSISTOR-LED>
Oscar Núñez Mori (2024a, Feb 18). *TRANSISTOR-RELAY-12V-220V*. <https://github.com/oscar-uni/SIMULIDE/tree/main/TRANSISTOR-RELAY-12V-220V>
Oscar Núñez Mori (2024b, Feb 18). *ARDUINO-UNO-R3-BLINK*. <https://github.com/oscar-uni/SIMULIDE/tree/main/ARDUINO-UNO-R3-BLINK>
Oscar Núñez Mori (2024c, Feb 18). *ARDUINO-UNO-R3-POTENCIOMETRO*. <https://github.com/oscar-uni/SIMULIDE/tree/main/ARDUINO-UNO-R3-POTENCIOMETRO>