

PROYECTO FINAL

POTENCIA EN AC Y MEJORAMIENTO FACTOR DE POTENCIA

Profesor

Luis Fernando Vásquez Vera

Estudiantes

Sánchez Maldonado Juan Francisco

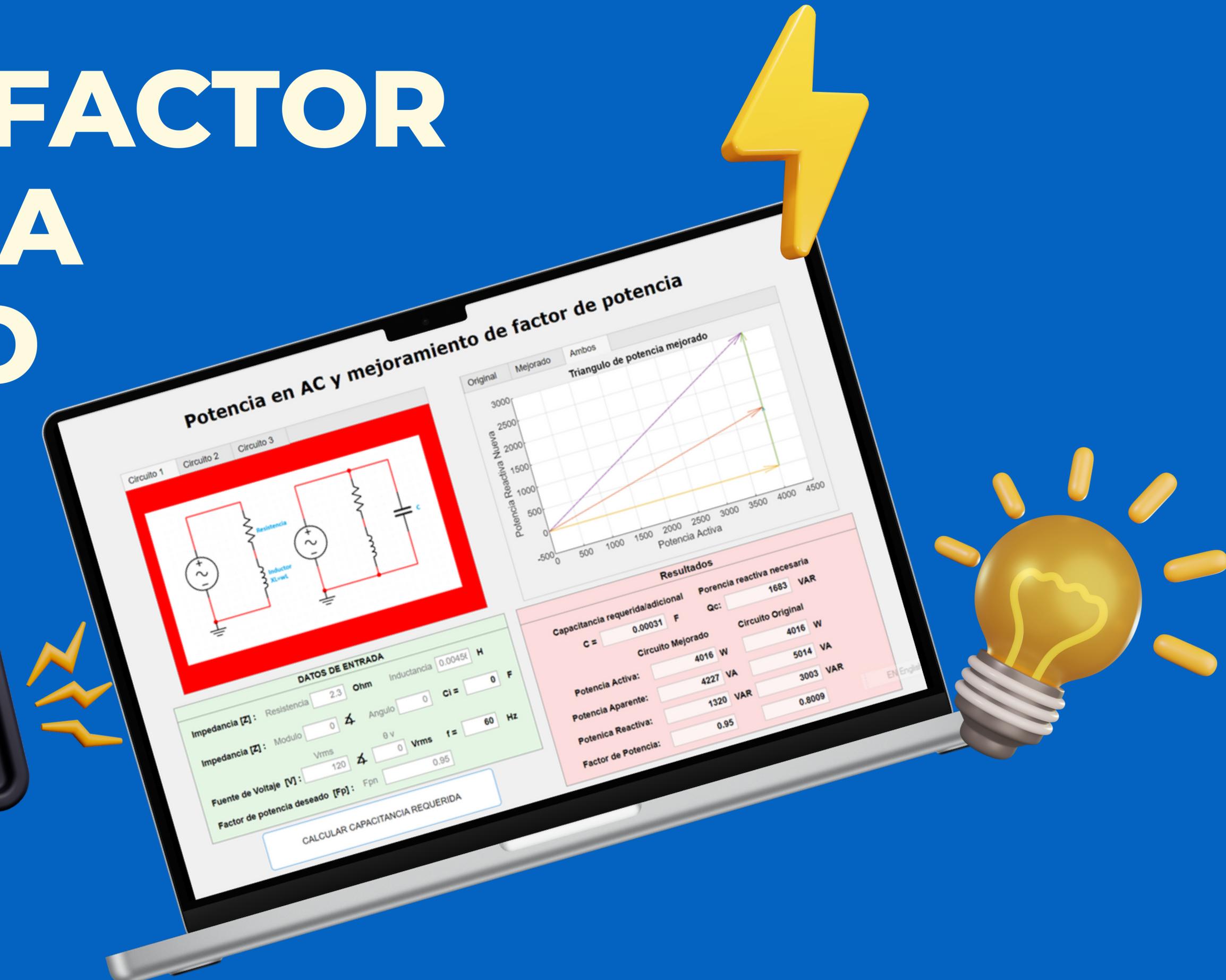
Sarmiento Franco Efrén Josué

Paralelo: 105



MEJORA EL FACTOR DE POTENCIA MÁS RÁPIDO

 DOWNLOAD





Calcula rápidamente la capacitancia necesaria para mejorar el factor de potencia de tus circuitos.



¿Para quién está dirigido?

Profesionales
en el sector
eléctrico

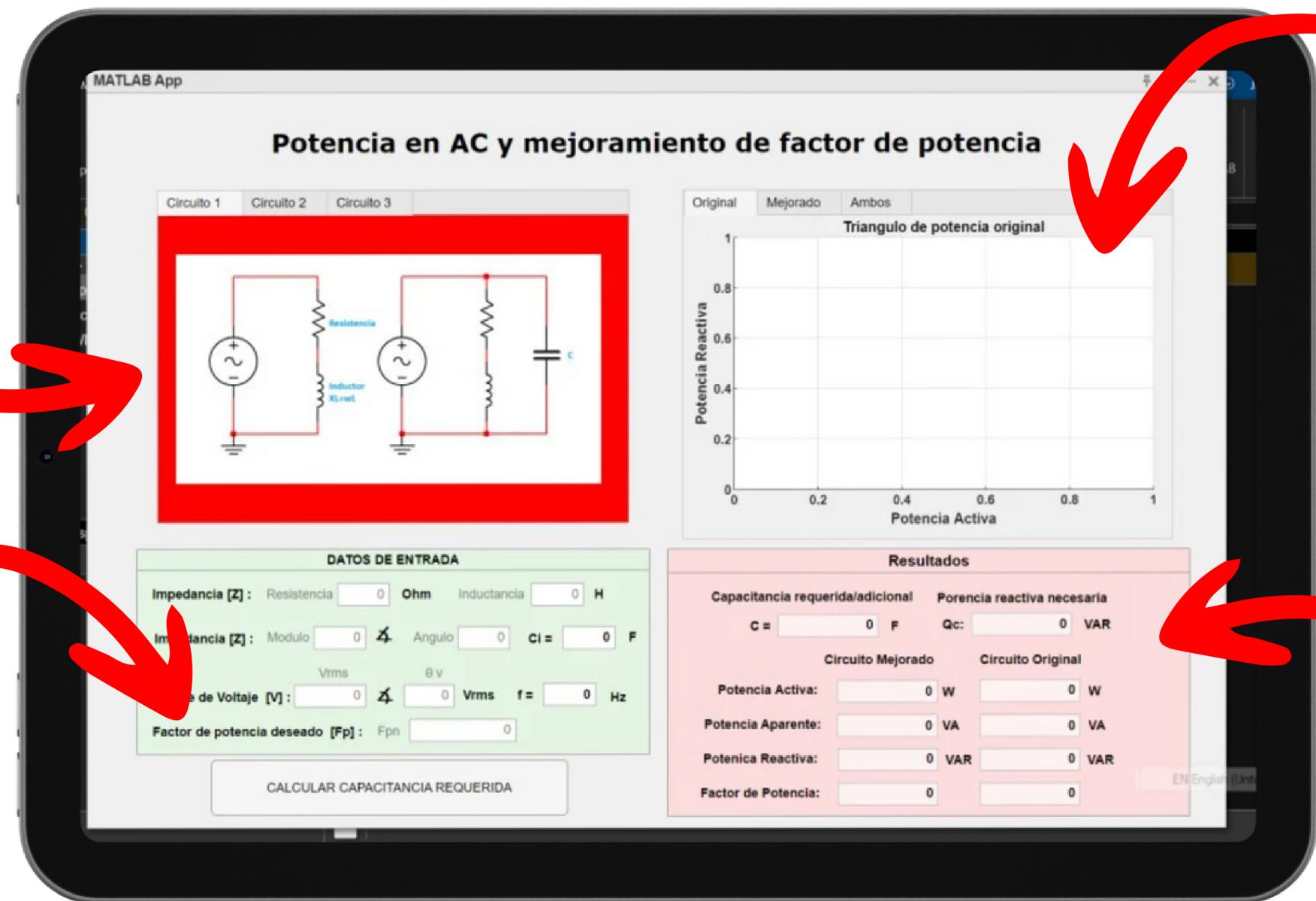


Estudiantes
de ingeniería

¿Cómo funciona?

**Selección
de circuito**

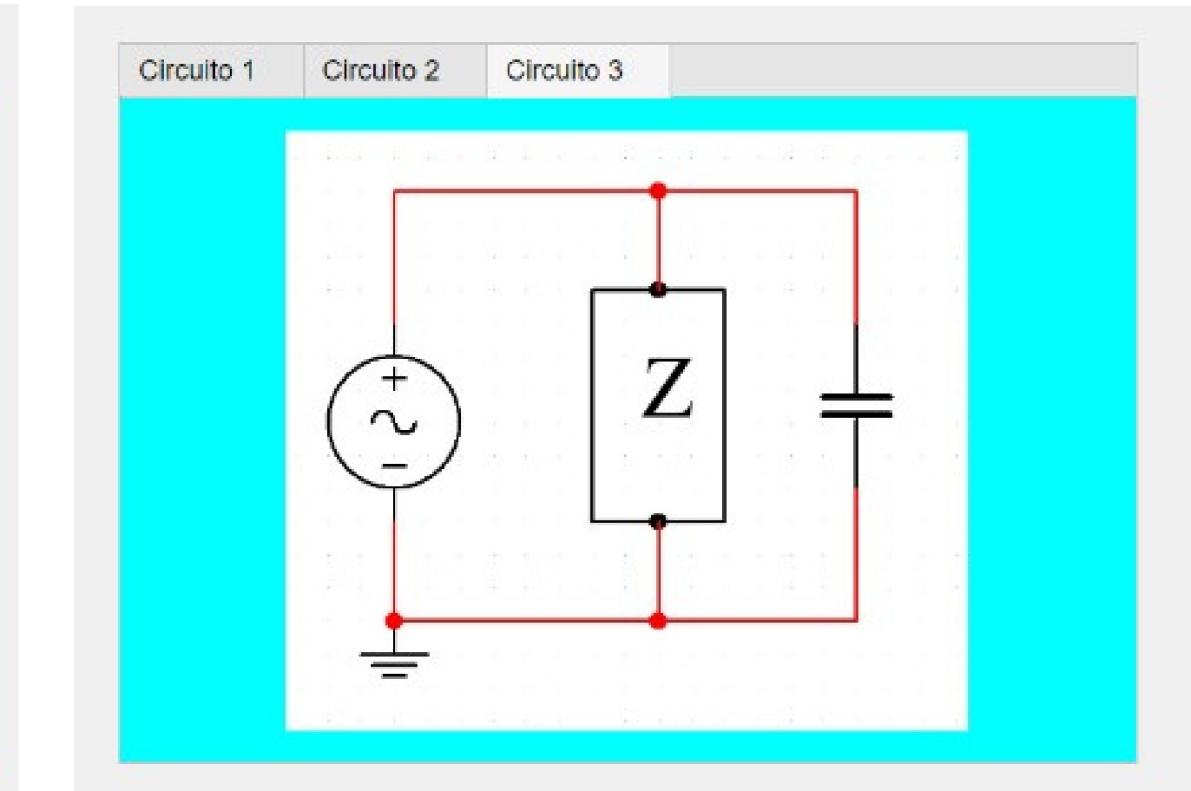
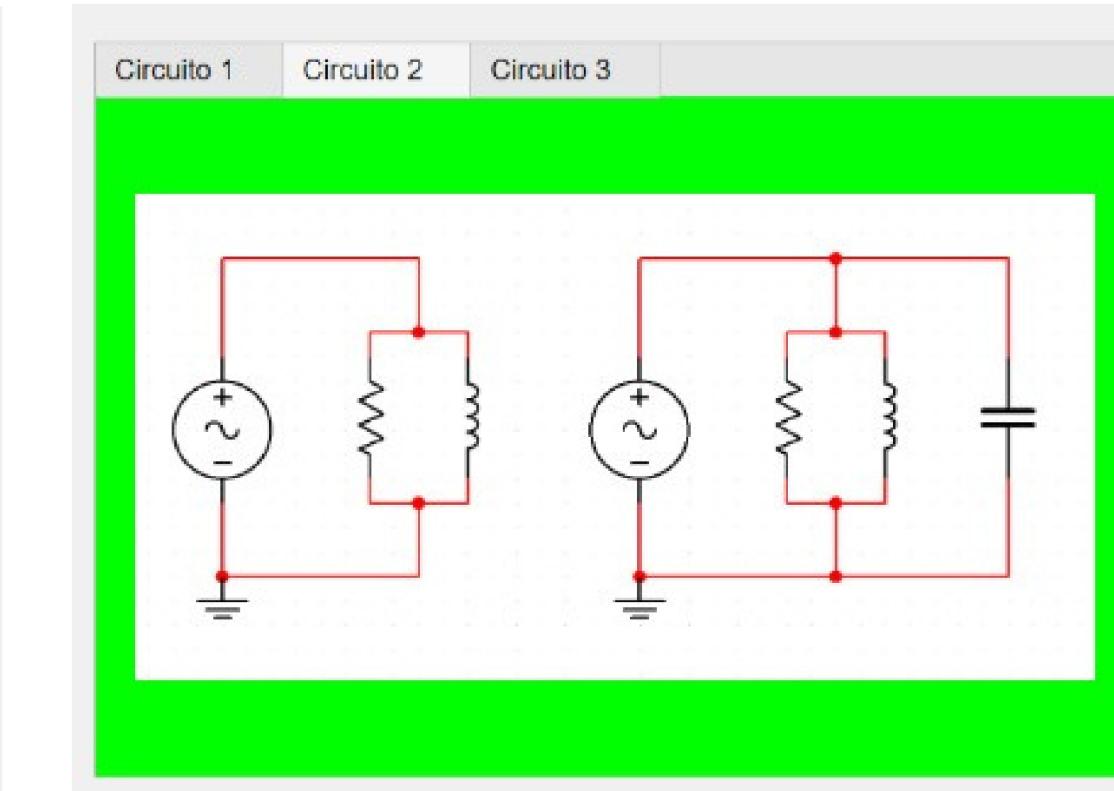
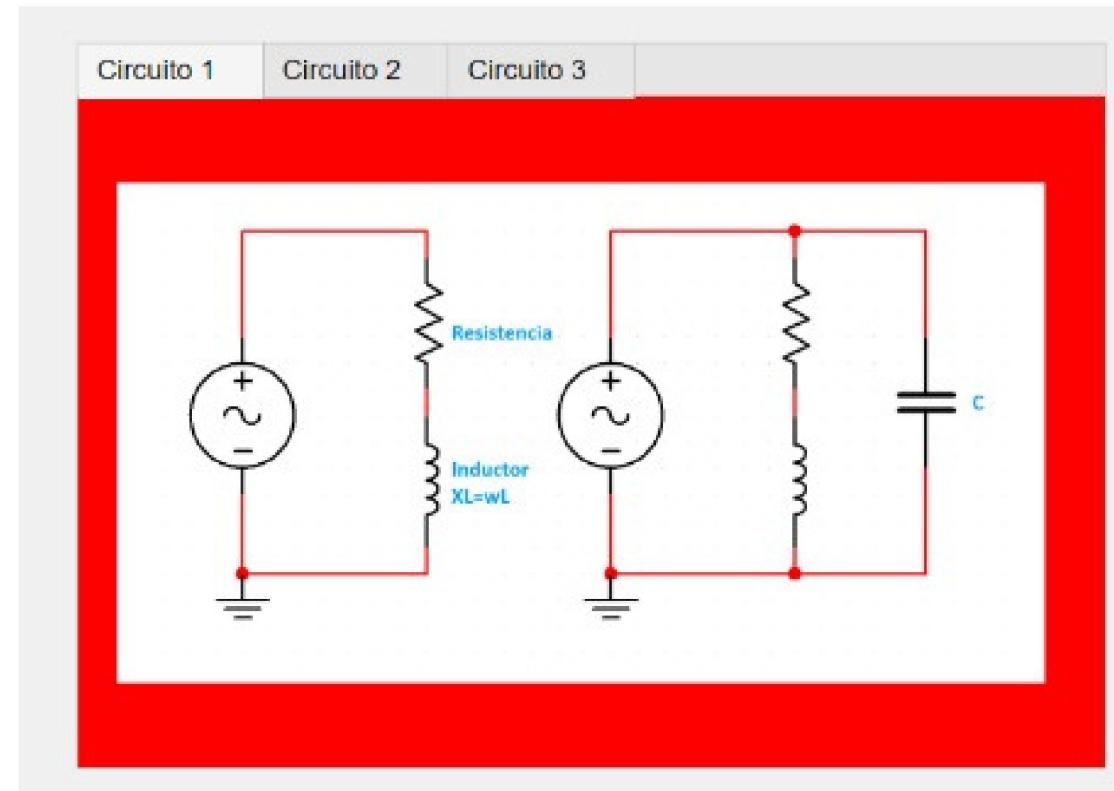
**Ingreso de
datos**



**Gráfico de
triángulos
de potencias**

Resultados

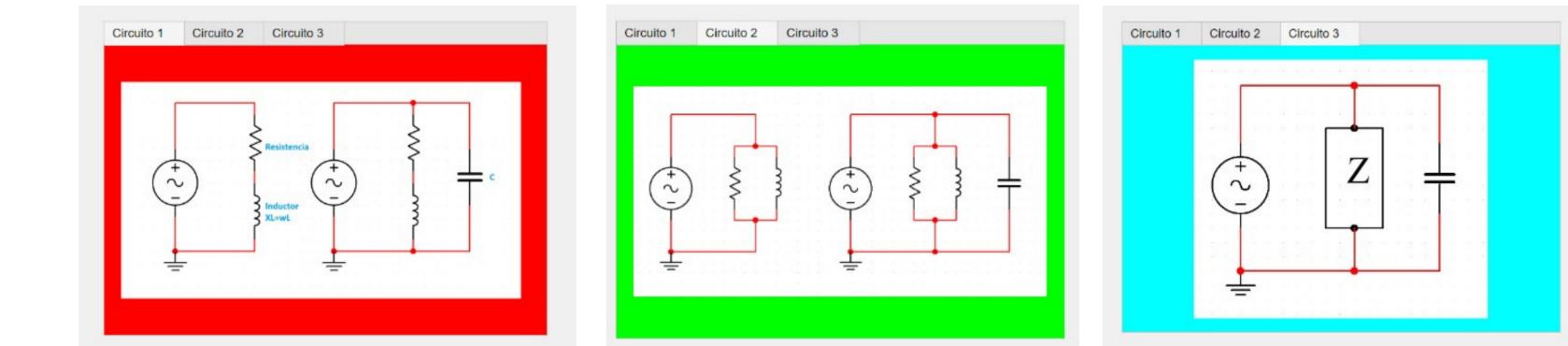
Circuitos disponibles



Datos solicitados

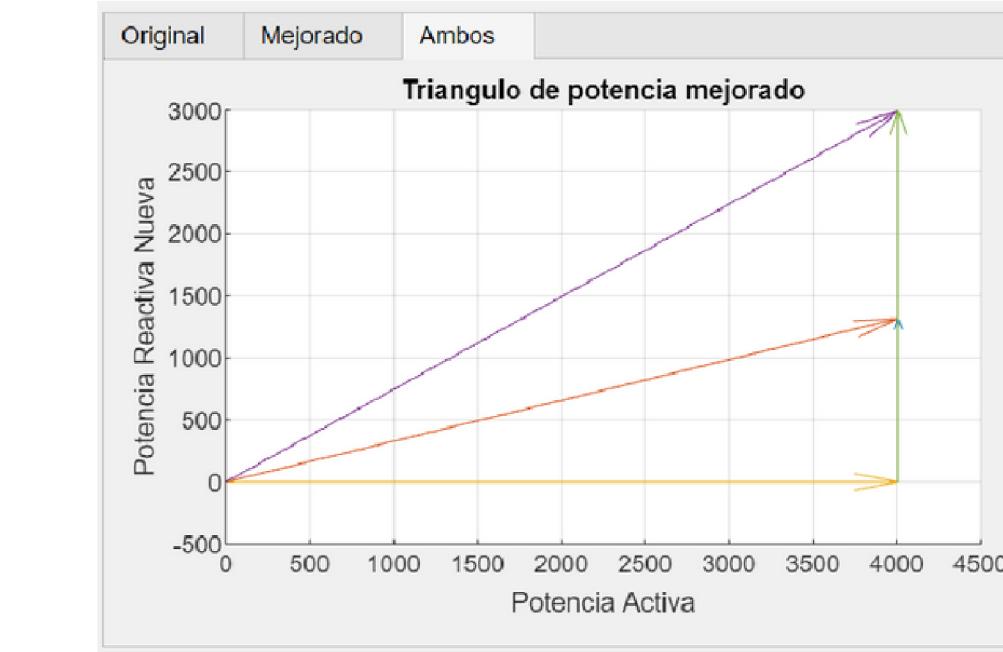
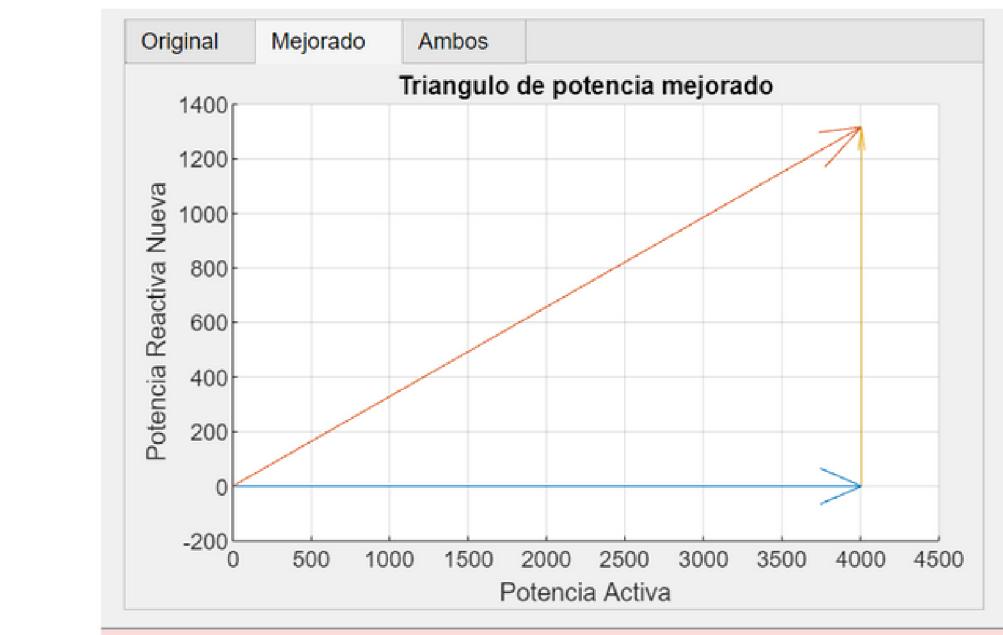
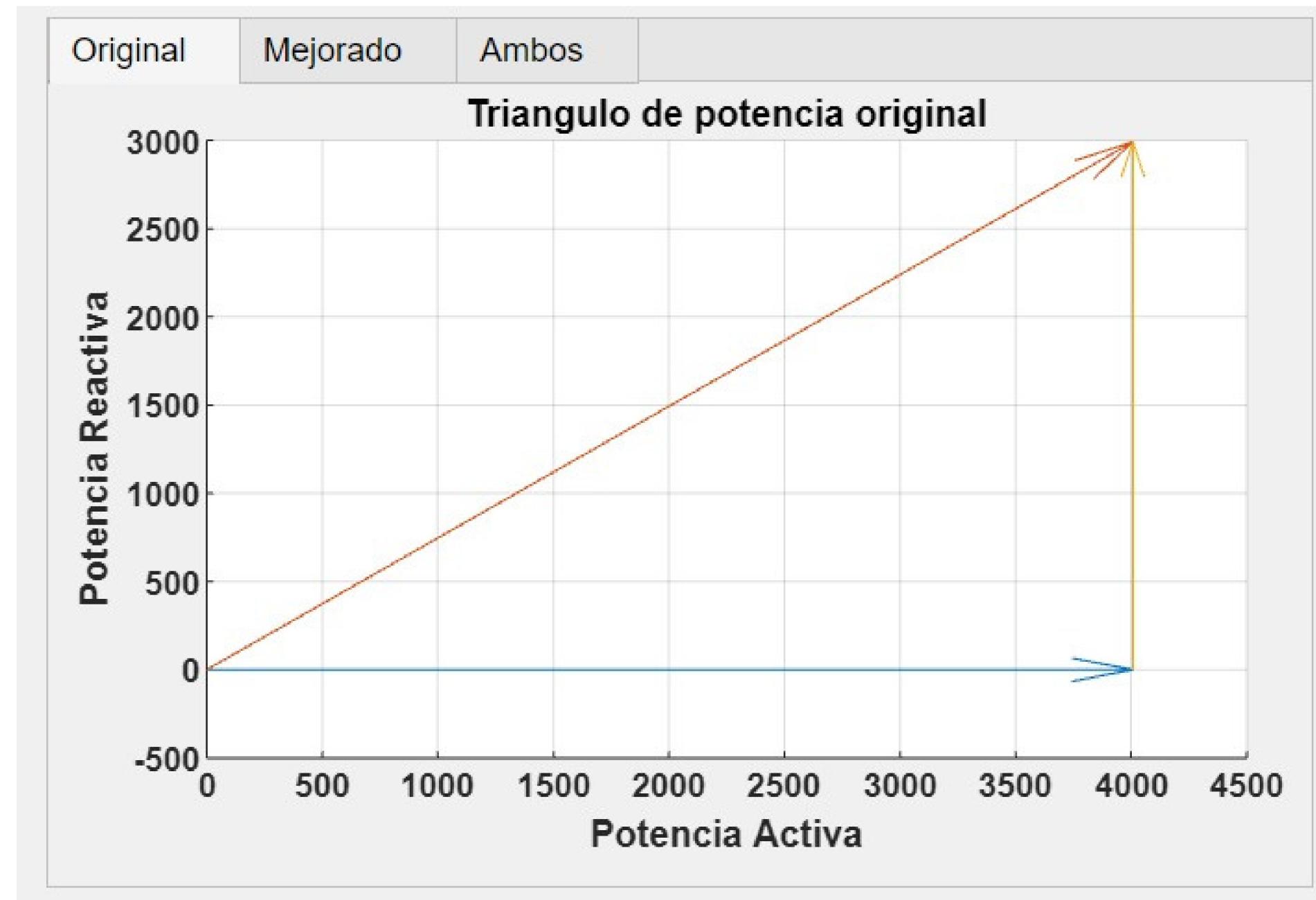
DATOS DE ENTRADA									
Impedancia [Z] :	Resistencia	<input type="text" value="0"/>	Ohm	Inductancia	<input type="text" value="0"/>	H			
Impedancia [Z] :	Modulo	<input type="text" value="0"/>	4	Angulo	<input type="text" value="0"/>	Ci =	<input type="text" value="0"/>	F	
Fuente de Voltaje [V] :	Vrms	<input type="text" value="0"/>	4	θ v	<input type="text" value="0"/>	Vrms	f =	<input type="text" value="0"/>	Hz
Factor de potencia deseado [Fp] :	Fpn	<input type="text" value="0"/>							
CALCULAR CAPACITANCIA REQUERIDA									

Resultados

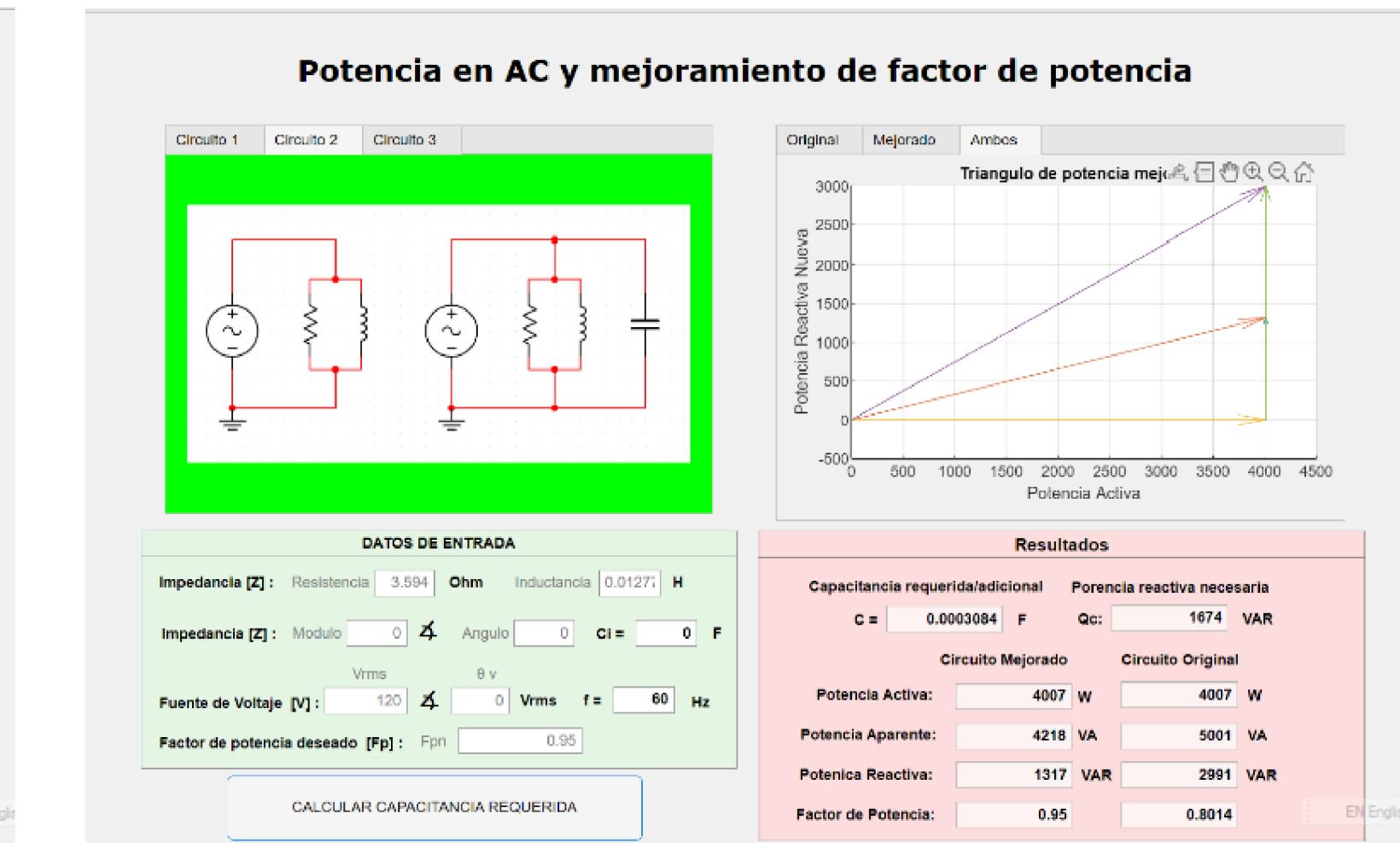
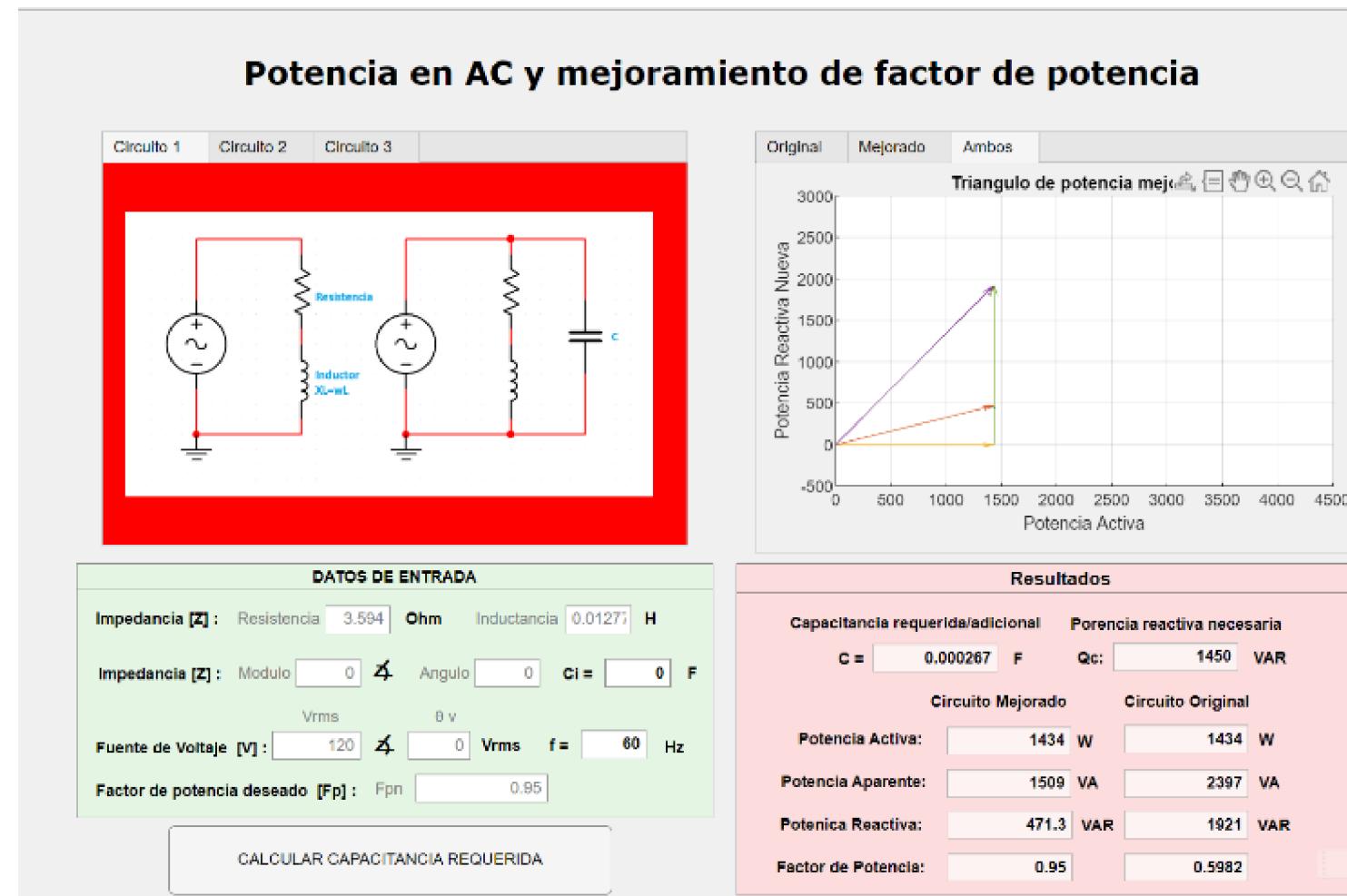


Resultados			
Capacitancia requerida/adicional	Porenzia reactiva necesaria		
$C = 0.0003084 \text{ F}$	$Q_c: 1674 \text{ VAR}$		
Circuito Mejorado		Circuito Original	
Potencia Activa:	4007 W	4007 W	
Potencia Aparente:	4218 VA	5001 VA	
Potenica Reactiva:	1317 VAR	2991 VAR	
Factor de Potencia:	0.95	0.8014	EN ER

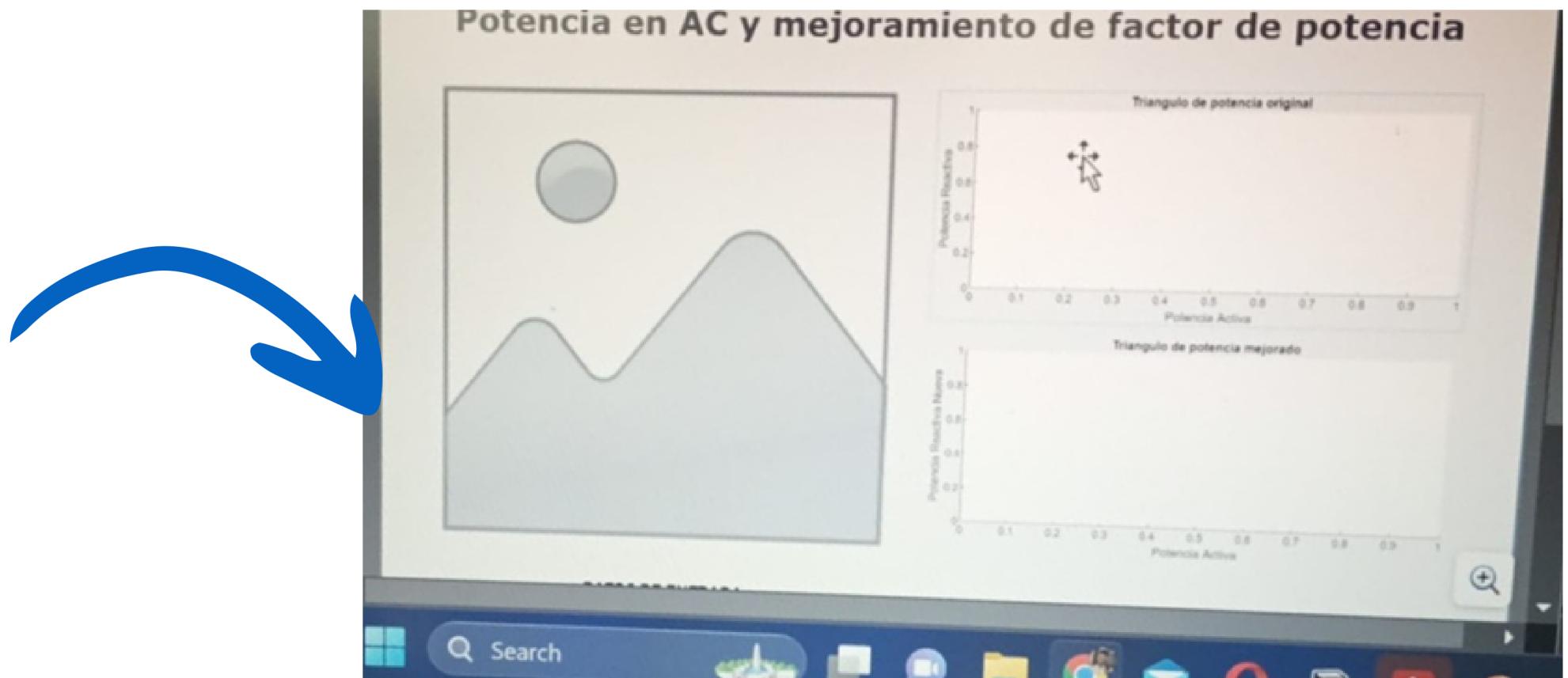
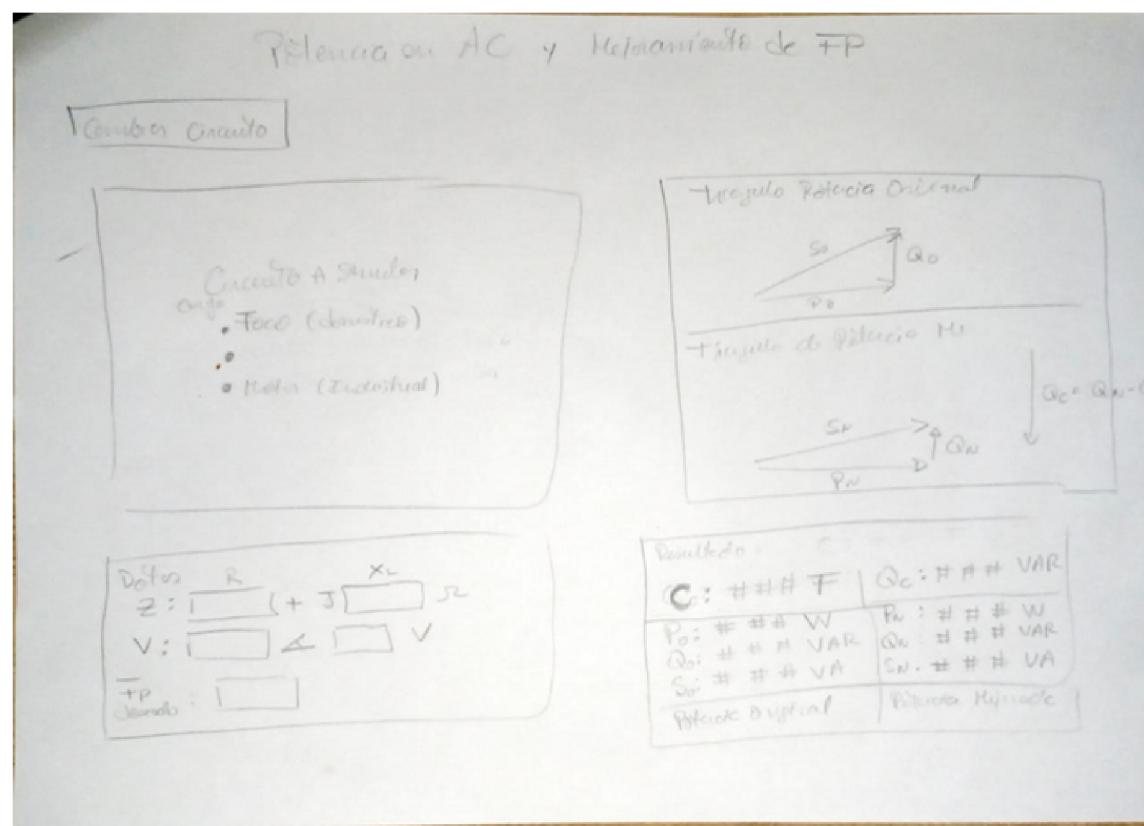
Triangulos de potencia



Comparativa de resultados



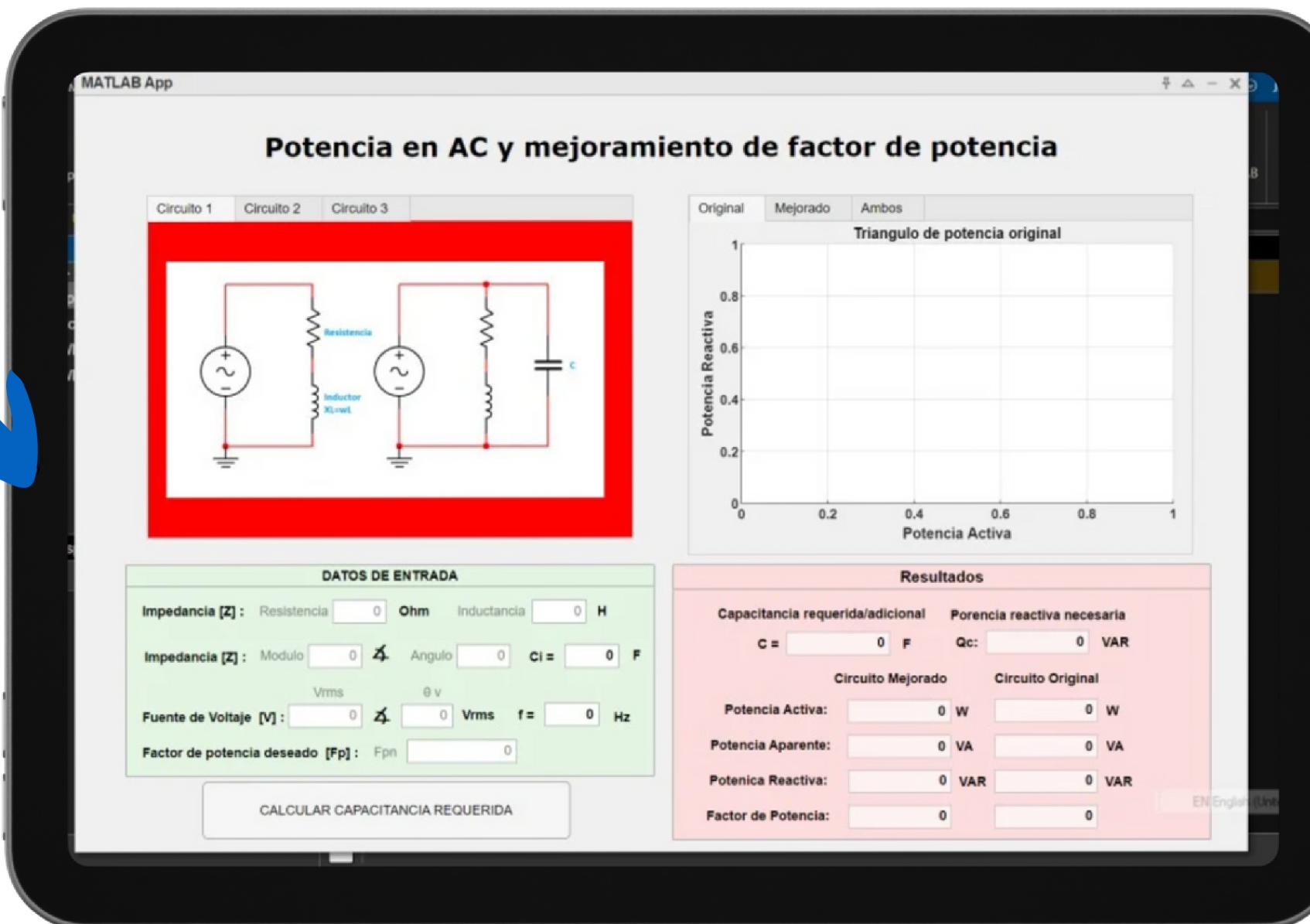
Desarrollo



Boceto

Maquetación

Desarrollo



DESIGNER EDITOR VIEW

Save Print Compare To Go To Bookmark Callback Function Property App Input Arguments App Help Text Comment Indent Run Step Stop

NAVIGATE INSERT CODE RUN

MATLAB App

```
function appStart(app)
    % Inicializar los componentes de la interfaz
    % ...
end

function appUpdate(app)
    % Actualizar los componentes de la interfaz
    % ...
end

function appStop(app)
    % Detener la ejecución si falta algún valor
    if app.TabGroup_circuitos.SelectedTab == app.CircuitoOriginal
        cla(app.graficaPotenciasOriginal);
        cla(app.graficaPotenciasMejorada);
        cla(app.graficaTriangulosPotencias);

        reactanciaInductiva=app.inductancia_in.Value^2*pi*app.frecuencia_in.Value;
        modulZ = sqrt((reactanciaInductiva)^2+(app.resistencia_in.Value)^2);
        anguloz=rad2deg(atan(reactanciaInductiva/app.resistencia_in.Value));
        factPotenciaInicial = cosd(anguloz);

        app.potenciaActivaOriginal_out.Value = ((app.vrms_in.Value^2)/(modulZ))*factPotenciaInicial;
        app.potenciaActivaOriginal_out.Value = ((app.vrms_in.Value^2)/(modulZ))*sin(deg2rad(anguloz));
        app.potenciaAparenteOriginal_out.Value = sqrt((app.potenciaActivaOriginal_out.Value)^2+(app.potenciaReactivaOriginal_out.Value)^2);

        %plot(app.graficaPotenciasOriginal,app.potenciaActivaOriginal_out.Value,0);
        originP = [0, 0]; % punto de inicio de la flecha
        directionP = [app.potenciaActivaOriginal_out.Value, 0]; % dirección de la flecha en el eje x
        quiver(app.graficaPotenciasOriginal, originP(1), originP(2), directionP(1), directionP(2), "off");
        originS = [0, 0]; % punto de inicio de la flecha
        directionS = [app.potenciaActivaOriginal_out.Value, app.potenciaReactivaOriginal_out.Value]; % dirección de la flecha en el eje y
        quiver(app.graficaPotenciasOriginal, originS(1), originS(2), directionS(1), directionS(2), "off");
        originQ = [0, app.potenciaActivaOriginal_out.Value, 0]; % punto de inicio de la flecha
        directionQ = [0, app.potenciaActivaOriginal_out.Value, 1]; % dirección de la flecha en el eje z
        quiver(app.graficaPotenciasOriginal, originQ(1), originQ(2), directionQ(1), directionQ(2), "off");
        hold(app.graficaPotenciasOriginal, "on");

        anguloNuevo=acosd(app.fpn_in.Value);
        app.potenciaReactivaNueva_out.Value=app.potenciaActivaOriginal_out.Value*tand(anguloNuevo);
        app.potenciaReactivaC_out.Value=app.potenciaReactivaOriginal_out.Value-app.potenciaReactivaNueva_out.Value;
        app.potenciaAparenteNueva_out.Value=sqrt((app.potenciaActivaOriginal_out.Value)^2+(app.potenciaReactivaNueva_out.Value)^2);
        app.potenciaActivaNueva_out.Value = app.potenciaActivaOriginal_out.Value;
        w=app.frecuencia_in.Value*2*pi;
        app.capacitanciaRequerida_out.Value=app.potenciaReactivaC_out.Value/(w*(app.vrms_in.Value)^2);
        app.factorOriginal_out.Value = factPotenciaInicial;
        app.factorMejorado_out.Value = app.fpn_in.Value;
        originP2 = [0, 0]; % punto de inicio de la flecha
    end
end
```

Final

Código

GRACIAS...

