```
using SymPy
```

```
using Plots
```

Declaracao de variaveis

(t, s, A, B)

```
@syms t s A B
```

R = 10

```
R = 10
```

L = 4

```
L = 4
```

C = 0.25

```
C = 0.25
```

V0 = 80

```
V0 = 80
```

iL0 = 2

```
iL0 = 2
```

Resolucao da questao

α = 1.25

```
α = R/(2*L)
```

ω0 = 1.0

```
ω0 = 1/sqrt(L*C)
```

s1 = -0.5

```
s1 = -α + sqrt(α^2 - ω0^2)
```

s2 = -2.0

```
s2 = -α - sqrt(α^2 - ω0^2)
```

v_t =

$$Ae^{-0.5t} + Be^{-2.0t}$$

```
v_t = A*exp(s1*t) + B*exp(s2*t)
```

dv_dt =

$$-0.5Ae^{-0.5t} - 2.0Be^{-2.0t}$$

```
dv_dt = diff(v_t, t)
```

eq1 =

$$A + B = 0$$

```
eq1 = subs(v_t, t=>0) ~ 0  # v(0) = 0
```

eq2 =

$$-0.5A - 2.0B = -8$$

```
eq2 = subs(dv_dt, t=>0) ~ -8  # dv(0)/dt = -8
```

Solucionando o problema

sol = Dict($A \Rightarrow -5.33333333333333$, $B \Rightarrow 5.33333333333333$)

```
sol = solve([eq1, eq2], [A, B])
```

v_t_sol =

$$5.33333333333333e^{-2.0t} - 5.33333333333333e^{-0.5t}$$

```
v_t_sol = subs(v_t, sol)
```

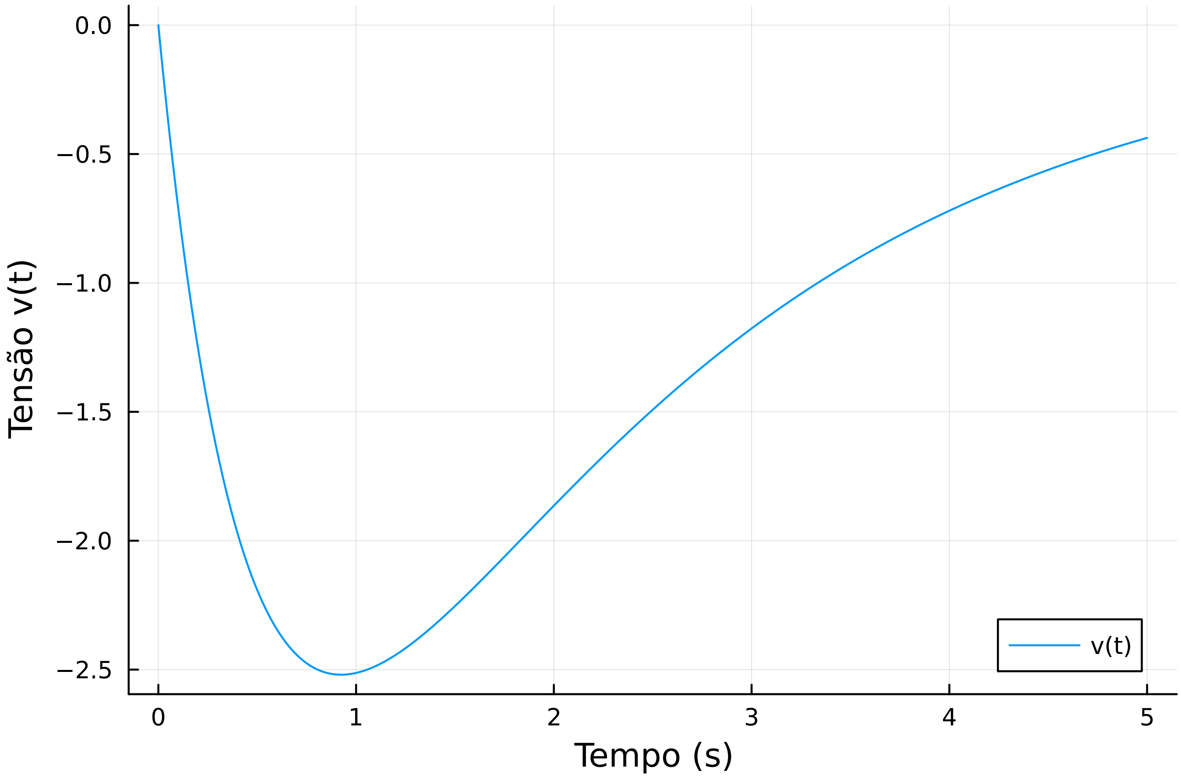
Gráfico:

v (generic function with 1 method)

```
v(t) = 5.333*exp(-2*t) - 5.333*exp(-0.5*t)
```

t_values = 0.0:0.01:5.0

```
t_values = 0:0.01:5
```




```julia
plot(t_values, v.(t_values), xlabel="Tempo (s)", ylabel="Tensão v(t)", label="v(t)")
```