

# LAB #3: WEB APPLICATION WITH GENIE

Abdelbacet Mhamdi  
Senior-lecturer, Dept. of EE  
ISET Bizerte — Tunisia  
✉ a-mhamdi

## I. EXERCISES

In this lab, you will create a basic web application using **Genie** framework in Julia. The application will allow us to control the behaviour of a sine wave, given some adjustable parameters.

### Exo 1: Sine Wave Control

We provide the Julia and HTML codes to build and run a web app that allows us to control the amplitude and frequency of a sine wave. **Plotly** is used to plot the corresponding graph. We also added a slider to change the number of samples used to draw the figure. The latter setting permits to grasp the influence of sampling frequency on the look of our chart.

```
using GenieFramework
@genietools

@app begin

    @in N::Int32 = 1000
    @in amp::Float32 = 0.25
    @in freq::Int32 = 1

    @out my_sine = PlotData()

    @onchange N, amp, freq begin
        x = range(0, 1, length=N)
        y = amp*sin.(2*π*freq*x)

        my_sine = PlotData(x=x,
                           y=y,

    plot=StipplePlotly.Charts.PLOT_TYPE_LINE)
    end

end

@page("/", "app.jl.html")
```

```
<header class="st-header q-pa-sm">
  <h1 class="st-header__title text-h3" Sinewave
```

```
Dashboard </h1>
</header>

<div class="row">
  <div class="st-col col-12 col-sm st-module">
    <p><b># Samples</b></p>
    <q-slider v-model="N"
      :min="10" :max="1000"
      :step="10" :label="true">
    </q-slider>
  </div>

  <div class="st-col col-12 col-sm st-module">
    <p><b>Amplitude</b></p>
    <q-slider v-model="amp"
      :min="0" :max="3"
      :step=".5" :label="true">
    </q-slider>
  </div>

  <div class="st-col col-12 col-sm st-module">
    <p><b>Frequency</b></p>
    <q-slider v-model="freq"
      :min="0" :max="10"
      :step="1" :label="true">
    </q-slider>
  </div>

  <div class="row">
    <div class="st-col col-12 col-sm st-module">
      <p><b>Sinewave</b></p>
      <plotly :data="my_sine"> </plotly>
    </div>
  </div>
```

```
julia --project
```

```
julia> using GenieFramework
julia> Genie.loadapp() # Load app
julia> up() # Start server
```

We can now open the browser and navigate to the link **localhost:8000**. We will get the graphical interface as in Figure 1.

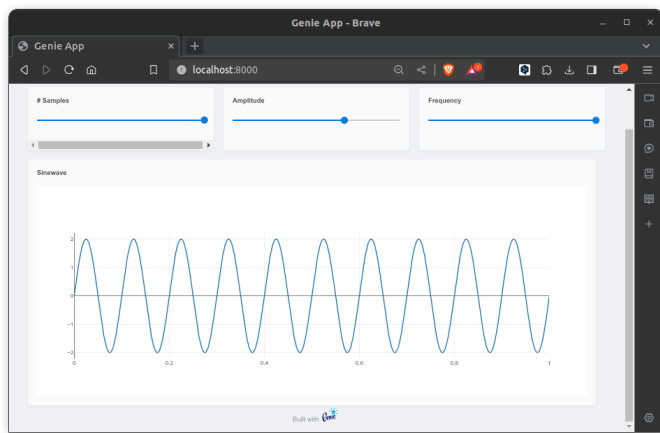


Figure 1: Genie -> Sine Wave



You are asked to add two extra sliders that modify the behaviour of the sine wave graph:

1. *Phase* ranging between  $-\pi$  and  $\pi$ , changes by a step of  $\frac{\pi}{100}$
2. *Offset* varies from  $-0.5$  to  $1$ , by a step of  $0.1$ .