



# LAB #3: WEB APPLICATION WITH GENIE

farah jemouai  
Dept. of EE  
ISET Bizerte — Tunisia  
 abc

Chahd ezzine  
Dept. of EE  
ISET Bizerte — Tunisia  
 abc

## I. WORK

In this lab, we created 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. we carried out this lab using the REPL as in Figure 1.

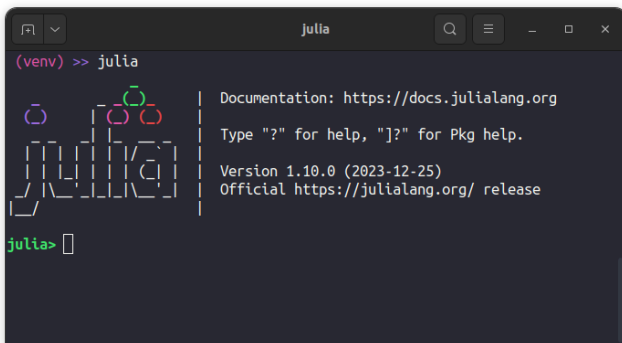


Figure 1: Julia REPL

### 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, frequency, phase and offset of a sine wave. **Plotly** is used to plot the corresponding graph. we also added a slider to replace the number of samples, slider to control the phase, slider of the offset used to draw the figure. The latter setting permits to grasp the influence of sampling frequency on the look of our chart.

#### Sine Wave Control

```
using GenieFramework
@genietools

@app begin

    @in N::Int32 = 1000
    @in amp::Float32 = 0.25
    @in ph::Float32 = 0.0
    @in offs::Float32 = 0.0
    @in freq::Int32 = 1
```

```
@out my_sine = PlotData()

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

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

plot=StipplePlotly.Charts.PLOT_TYPE_LINE)
end

end

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

#### html program

```
<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>Phase</b></p>
    <q-slider v-model="ph"
:min="-3.141" :max="3.141"
:step="1" :label="true">
```

```

</q-slider>
</div>

<div class="st-col col-12 col-sm st-module">
  <p><b>Offset</b></p>
  <q-slider v-model="offs"
    :min="-.5" :max="1"
    :step=".1" :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>

<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>
</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](http://localhost:8000). We will get the graphical interface as in .

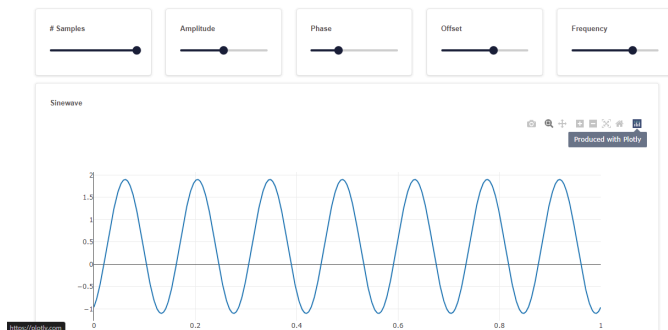


Figure 2: Genie -> Sine Wave