# Lab #3: Web Application with Genie

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

ADEM NEMRI
*Dept. of EE*
*ISET Bizerte — Tunisia*
*ademnmr*

HOUSSNI MARNISI
*Dept. of EE*
*ISET Bizerte — Tunisia*
*hosnimarnisi*

## I. Introduction

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 adjustble parameters.

## II. Application : Opening Genie

in this application we gonna open genie web to use it changing some sinewave parameter

- write down this code to load the up genie app and had the link to the genie space

```
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

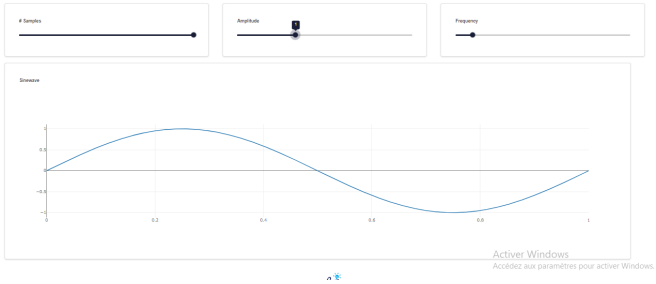

Figure 1: Genie web

## III. Application : Adding phase

in this application we gonna add a phase parametere to GenieFramework

- adding the phase to app.jl

```
using GenieFramework
@genietools

@app begin

    @in N::Int32 = 1000
    @in amp::Float32 = 0.25
    @in freq::Int32 = 1
    @in ph::Float32 = 0
    @out my_sine = PlotData()

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

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

plot=StipplePlotly.Charts.PLOT_TYPE_LINE)
    end

end

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

- adding the phase to 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="st-col col-12 col-sm st-module">
        <p><b>phase</b></p>
  <q-slider v-model="freq"
    :min="-3.14" :max="3.14"
    :step="0.314" :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>
```

- the result in genie :

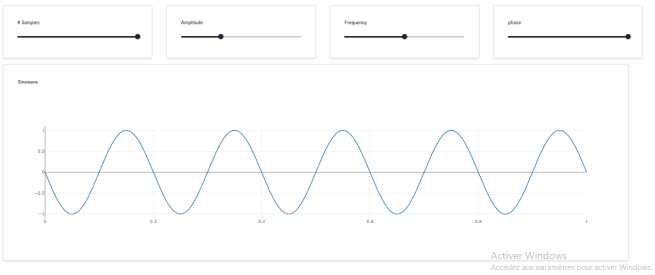

Figure 2: adding phase

## IV. Application : Adding the offset

in this app we gonna add another paramatere the offset

- adding the phase to app.jl

```
using GenieFramework
@genietools

@app begin

    @in N::Int32 = 1000
    @in amp::Float32 = 0.25
    @in freq::Int32 = 1
    @in ph::Float32 = 0
    @in off::Float32 = 0
    @out my_sine = PlotData()

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

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

plot=StipplePlotly.Charts.PLOT_TYPE_LINE)
```

```
    end

end

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

- adding the phase to 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="st-col col-12 col-sm st-module">
        <p><b>phase</b></p>
  <q-slider v-model="freq"
    :min="-3.14" :max="3.14"
    :step="0.314" :label="true">
  </q-slider>
    </div>
    <div class="st-col col-12 col-sm st-module">
        <p><b>offset </b></p>
  <q-slider v-model="freq"
    :min="-0.5" :max="1"
    :step="0.1" :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>
```

- the result in genie :

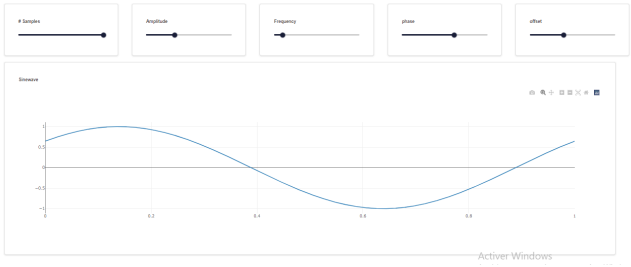


Figure 3: adding offset