


LAB #3: WEB APPLICATION WITH GENIE

Belkhamza Ayhem

Dept. of EE

ISET Bizerte — Tunisia

 Ayhem99

I. EXERCISE

In this lab, we 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 Sinewave Dashboard. This dashboard allows you to manipulate various parameters of a sine wave and visualize the results.

Here's a brief explanation of each component:

Samples: This slider adjusts the number of samples used to generate the sine wave. The range is from 10 to 1000, with steps of 10.

Amplitude: This slider adjusts the amplitude of the sine wave. The range is from 0 to 3, with steps of 0.5.

Frequency: This slider adjusts the frequency of the sine wave. The range is from 0 to 10, with steps of 1.

Offset: This slider adjusts the offset of the sine wave. The range is from -3.141 to 3.141, with steps of 1.

Phase: This slider adjusts the phase of the sine wave. The range is from -0.5 to 1, with steps of 0.1.

The Sinewave section at the bottom displays the generated sine wave based on the parameters set above.

The plotly component is used for this visualization..

```
using GenieFramework
@genietools

@app begin

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

    @out my_sine = PlotData()

    @onchange N, amp, freq, off, ph begin
        x = range(0, 1, length=N)
```

```
y = amp*sin.(2*pi*freq*x.+ph).+off

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="st-col col-12 col-sm st-module">
        <p><b>Offset</b></p>
```

```

<q-slider v-model="off"
  :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>Phase</b></p>
  <q-slider v-model="ph"
    :min="-0.5" :max="1"

    :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>

```

we add two extra sliders that modify the behaviour of the sine wave graph:

1. *Phase* ranging between $-\pi$ and π , changes by a step of $\frac{\pi}{100}$

HTML:

```

<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=".0314"
    :label="true">
  </q-slider>
</div>

```

Julia:

```

@in ph::Float32 = 0.0
...
@onchange N, amp, freq , ph begin
...
y = amp*sin.(2*π*freq*x.+ph)

```

1. *Offset* varies from -0.5 to 1 , by a step of 0.1 .

HTML:

```

<div class="st-col col-12 col-sm st-module">
  <p><b>Offset</b></p>
  <q-slider v-model="off"
    :min="-0.5" :max="1" :step=".1" :label="true">
  </q-slider>
</div>

```

Julia:

ISSET Bizerte

```

@in off::Float32 = 0.0
...
@onchange N, amp, freq ,off, ph begin
...
y = amp*sin.(2*π*freq*x.+ph).+off

```

now we open **cmd** (command prompt) to launch the julia lancer

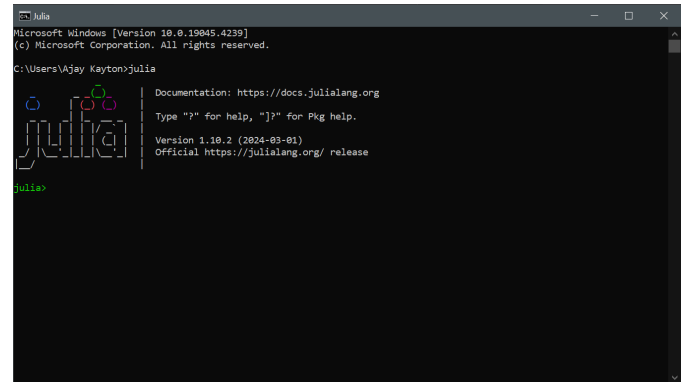


Figure 1: Cmd

```

cd("(location of the folder )/infodev-main/Codes/
web-app")
julia> using GenieFramework
julia> Genie.loadapp() # we use this command to
load the app
julia> up() # To start the server

```

We can now open the browser and navigate to the link **localhost:8000**. We will get the graphical interface as in Figure 2.

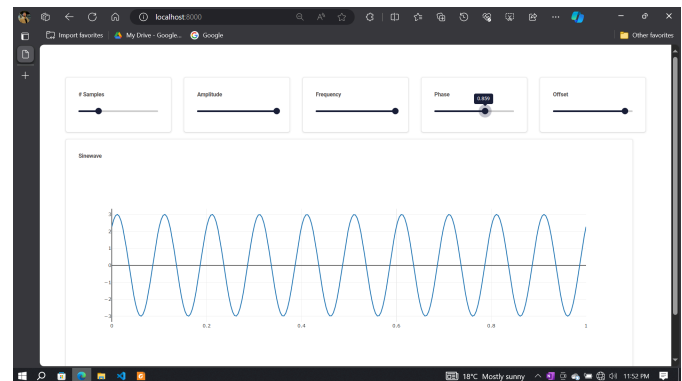


Figure 2: Genie -> Sine Wave