Lab #3: Web Application with Genie

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

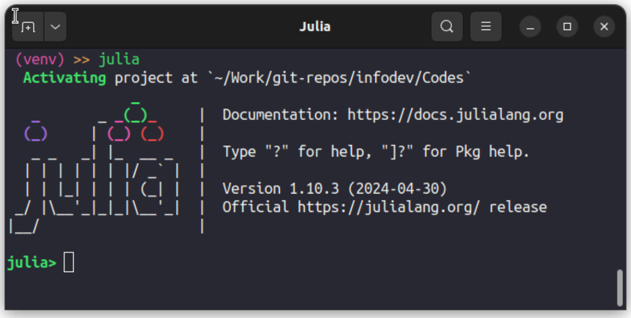
Sta dhia
Dept. of EE
ISET Bizerte — Tunisia
dhiasta

jeridi sadok mahdi
Dept. of EE
ISET Bizerte — Tunisia
sadokmahdi

***Abstract* — Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do..**

I. Introduction

In this lab, we 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. You are required to carry out this lab using the REPL as in Figure 1.



```
(venv) >> julia
 Activating project at `~/Work/git-repos/infodev/Codes`

               _
   _       _ _(_)_     |  Documentation: https://docs.julialang.org
  (_)     | (_) (_)    |
   _ _   _| |_  __ _   |  Type "?" for help, "]?" for Pkg help.
  | | | | | | |/ _` |  |
  | | |_| | | | (_| |  |  Version 1.10.3 (2024-04-30)
 _/ |\__'_|_|_|\__'_|  |  Official https://julialang.org/ release
|__/                   |

julia> 
```

Figure 1: Julia REPL

II. Application

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.

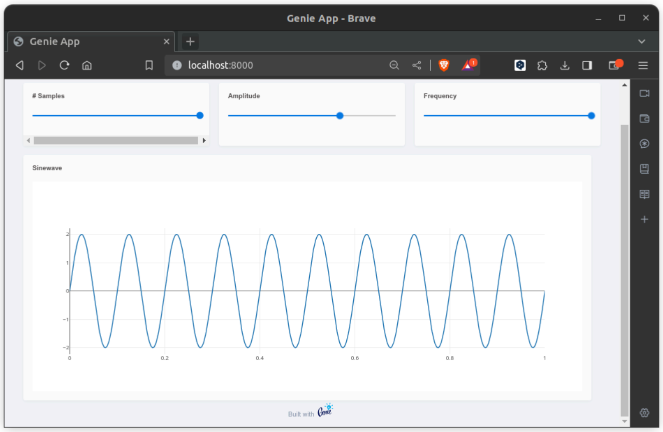
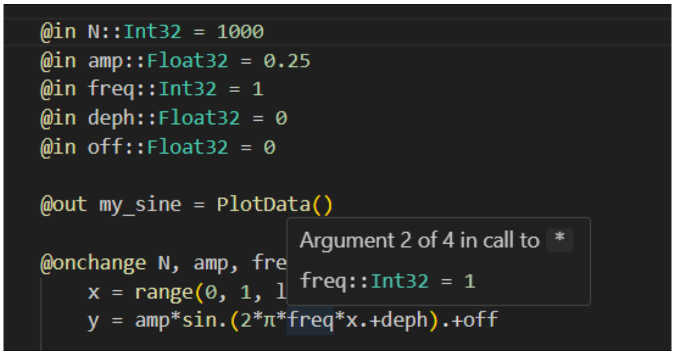


Figure 2: Genie -> Sine Wave

we have added to inputs phse and offset , their type is float and default value is 0. Also , we have added their names after onchange so we can control them.This work is shown in this code below:

- **The first programme is "app.jl"**



```
@in N::Int32 = 1000
@in amp::Float32 = 0.25
@in freq::Int32 = 1
@in deph::Float32 = 0
@in off::Float32 = 0

@out my_sine = PlotData()

@onchange N, amp, fre
    x = range(0, 1, l
    y = amp*sin.(2*π*freq*x.+deph).+off
```

```
Argument 2 of 4 in call to *
freq::Int32 = 1
```

Figure 3: Juliaj1

```
using GenieFramework
@genietools

@app begin

    @in N::Int32 = 1000
    @in amp::Float32 = 0.25
    @in freq::Int32 = 1
    @in deph::Float32 = 0
    @in off::Float32 = 0

    @out my_sine = PlotData()

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

        my_sine = PlotData(x=x,
                           y=y,
plot=StipplePlotly.Charts.PLOT_TYPE_LINE)
    end

end

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

- **The second programme is “app.jl.html”**

the second part we have added the slides of dephasage and offset in “app.jl.html” to control them on the web.

```
<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>dephansage</b></p>
        <q-slider v-model="deph"
    :min="-3.14" :max="3.14"
    :step=".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="off"
    :min="-0.5" :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>
```

```
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 http://127.0.0.1:8000. We will get the graphical interface as in Figure 2.

in the end we have a new web with the new slides of dephasage and offset

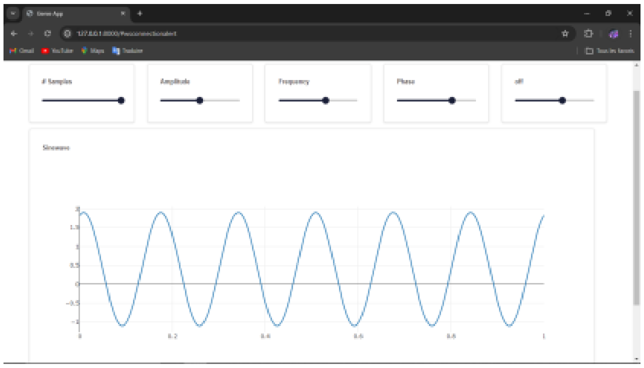


Figure 4: new web