

```

## quick and dirty simple app for the issue.
# kindly ignore the nonconformity of best practices and right way of doing things

library(shiny)
library(shinydashboard)
eventlog <- patients
ui <- dashboardPage(
  dashboardHeader(),
  dashboardSidebar(
    sidebarMenu(
      menuItem("Process Map", tabName = "process", icon = icon("dashboard")),
      menuItem("Resource Map", tabName = "resource", icon = icon("dashboard")),
      menuItem("Extra", tabName = "extra", icon = icon("dashboard"))
    )
  ),
  dashboardBody(

    tabItems(
      tabItem(tabName = "process",
        fluidRow(
          shinycssloaders::withSpinner(processanimatorOutput("process"))
        )
      ),
      tabItem(tabName = "resource",
        fluidRow(
          shinycssloaders::withSpinner(processanimatorOutput("process2"))
        )
      ),
      tabItem(tabName = "extra",
        fluidRow(
          )
        )
      ),
    )
  )
)

server <- function(input, output) {

  output$process <- renderProcessanimator(expr = {
    graph <- processmapR::process_map(eventlog, render = F)
    model <- DiagrammeR::add_global_graph_attrs(graph, attr = "rankdir", value =
input$orientation, attr_type = "graph")

    animate_process(eventlog, model,
      mode = "absolute",
      legend = "color",

```

```

        mapping = token_aes(color = token_scale("handling", scale = "ordinal", range =
RColorBrewer::brewer.pal(5, "YlOrBr"))),
        duration = 60)

    })

    output$process2 <- renderProcessanimator(expr = {
      graph <- processmapR::process_map(eventlog, render = F)
      model <- DiagrammeR::add_global_graph_attrs(graph, attr = "rankdir", value =
input$orientation, attr_type = "graph")

      animate_process(eventlog, model,
        mode = "absolute",
        legend = "color",
        mapping = token_aes(color = token_scale("handling", scale = "ordinal", range =
RColorBrewer::brewer.pal(5, "YlOrBr"))),
        duration = 60)

    })
  }

shinyApp(ui, server)

```