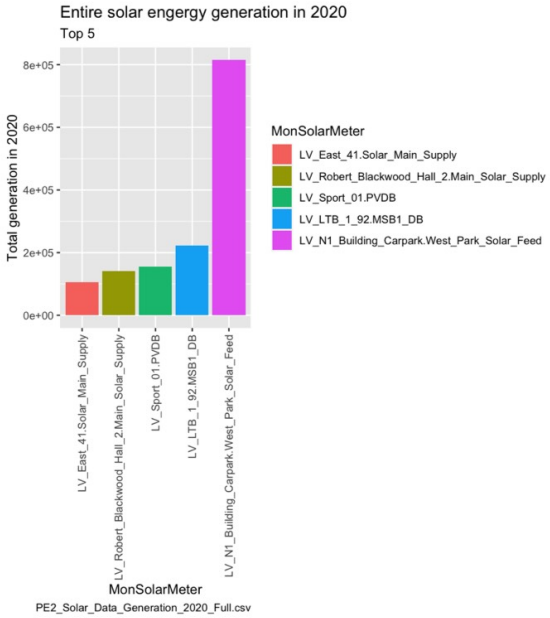
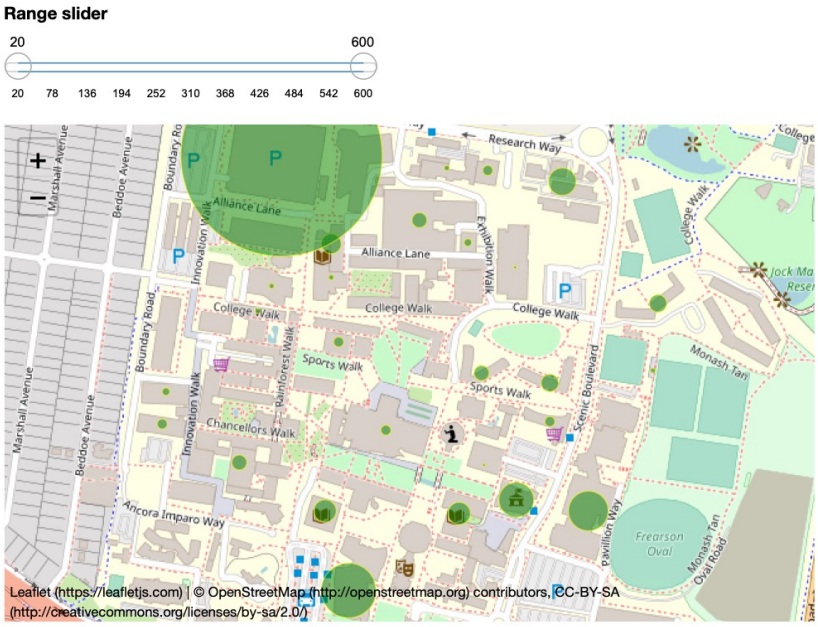
Shiny Web

Solar Energy Generation in Monash University Clayton Campus

Top 5 energy generating solar panels on buildings

This graph plots the total annual solar production of the five solar buildings with the highest total production capacity in 2020, and the total annual solar production of the five solar buildings with the highest annual production capacity. Each solar building is distinguished by color.





Energy generation throughout 2020

This graph is based on the total daily solar production of the five solar buildings with the highest annual total production capacity and the five solar buildings with the highest annual production capacity. Each solar building is distinguished by color, and the line graph can celebrate the daily production capacity trend of each solar building.

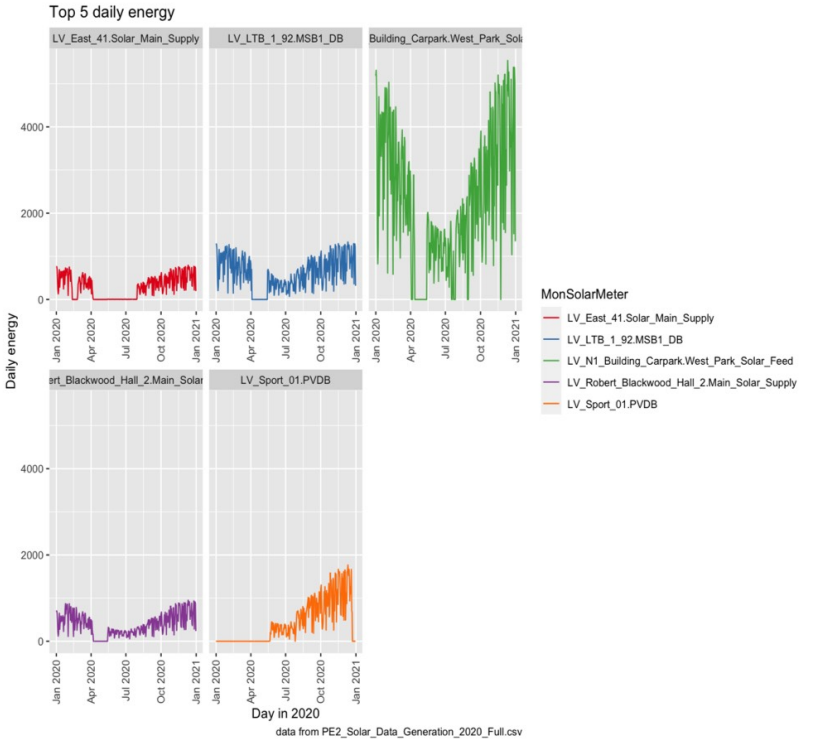


Figure 1： Final layout

Source Code

```
# import libraries
library(ggplot2)
library(leaflet)
library(dplyr)
library(RColorBrewer)
library(maps)
library(shiny)

# data input/ read files
solar_panels <- read.csv('./PE2_Solar_Panels.csv')
solar_generation <-
    read.csv('./PE2_Solar_Data_Generation_2020_Full.csv')

#2. the entirety of 2020 for only the top 5 buildings (VIS 1).
# calculate all year generations of different solar panels
# Reference: remove null value: https://www.statmethods.net/input/missingdata.html
year_generation <-
    solar_generation %>%
    group_by(MonSolarMeter) %>%
    summarise(total_generation = sum(Real.Energy.Into.the.Load..kWh., na.rm = TRUE))
year_generation

# order total generations of different solar panels
# Reference: Order:https://stackoverflow.com/questions/12289224/rank-and-order-in-r
order_generation <-
    year_generation[order(year_generation$total_generation, decreasing = TRUE), ]
order_generation

# select top 5 generations
top5_generation <- head(order_generation, 5)
top5_generation

# draw bar chart of top 5 generations
# Reference: ggplot mutate reorder:https://www.r-graph-gallery.com/267-reorder-a-variable-in-ggplot2.html
year_top5 <-
    top5_generation %>% mutate(MonSolarMeter = reorder(MonSolarMeter, total_generation)) %>%
    ggplot(aes(x = MonSolarMeter,
                        y = total_generation, fill = MonSolarMeter)) + geom_bar(stat = 'identity') + labs(
                            y = 'Total generation in 2020',
                            title = 'Entire solar engergy generation in 2020',
                            subtitle = 'Top 5',
                            caption = 'PE2_Solar_Data_Generation_2020_Full.csv'
                        )

# Reference: modify label angle:https://stackoverflow.com/questions/1330989/rotating-and-spacing-axis-labels-in-ggplot2
year_top5 <-
    year_top5 + theme(axis.text.x    = element_text(
        angle = 90,
        hjust = 1,
        vjust = 0.5
    ))

year_top5

#3. total energy generation during 2020 for only the top 5 buildings (VIS 2).
# Reference: convert date:https://www.statmethods.net/input/dates.html
solar_generation$Timestamp <-
    solar_generation$Timestamp %>% as.Date()

# Reference: facet_wrap:http://www.zevross.com/blog/2019/04/02/easy-multi-panel-plots-in-r-using-facet_wrap-and-facet_grid-from-ggplot2/
energy_trend <-
    solar_generation %>% filter(solar_generation$MonSolarMeter
                                %in% top5_generation$MonSolarMeter) %>%
    group_by(MonSolarMeter, Timestamp) %>%
    summarise(trend_energy = sum(Real.Energy.Into.the.Load..kWh., na.rm = T)) %>%
    ggplot(aes(
        x = Timestamp,
        y = trend_energy,
        group = MonSolarMeter,
```

```
        color = MonSolarMeter
    )) + geom_line(stat = 'identity') + scale_color_brewer(palette = 'Set1') +
    labs(
        x = 'Period in 2020',
        y = 'Energy trend',
        caption =    'data from PE2_Solar_Data_Generation_2020_Full.csv'
    ) + facet_wrap( ~ MonSolarMeter, scales = "free_x")

energy_trend <- energy_trend + theme(axis.text.x    = element_text(
    angle = 90,
    hjust = 1,
    vjust = 0.5
))

energy_trend
# interactive proportional symbol map,spatial positions of all 27 buildings
# join two tables
year_energy <-
    solar_generation %>%
    group_by(BuildingNum) %>%
    summarise(total_generation = sum(Real.Energy.Into.the.Load..kWh., na.rm = TRUE))
year_energy

# join two tables
new_solarPanels <-
    left_join(solar_panels, year_energy, by = "BuildingNum")
new_solarPanels

# Reference: popup multiple columns:https://stackoverflow.com/questions/32523314/r-how-to-paste-multiple-column-of-same-variable-in-popup-leaflet
location_diagram <- new_solarPanels %>%
    filter(Capacity..kW. >= min(Capacity..kW.) &
               Capacity..kW. <= max(Capacity..kW.)) %>%
    leaflet() %>%
    addTiles() %>%
    addCircleMarkers(
        lng = ~ Longitude,
        lat = ~ Latitude,
        weight = 1,
        color = "yellow",
        opacity = 0.5,
        stroke = T,
        fillOpacity = 0.5,
        fillColor = "green",
        radius = ~ total_generation / 10000,
        label = ~ as.character(BuildingName),
        popup = paste("BuildingName:",new_solarPanels$BuildingName,"<br>",
                      "Total Generation:",new_solarPanels$total_generation,"<br>")
    )

location_diagram

#Shiny
library("shiny")

order_generation <-
    year_generation[order(year_generation$total_generation, decreasing = TRUE),]
order_generation

c_min <-
    head(new_solarPanels[order(new_solarPanels$Capacity..kW., decreasing = FALSE),], 1)$Capacity..kW.
c_max <-
    head(new_solarPanels[order(new_solarPanels$Capacity..kW., decreasing = TRUE),], 1)$Capacity..kW.

c_min <- 20
c_max <- 600

ui <- fixedPage(
    h1("Solar Energy Generation in Monash University Clayton Campus"),
    hr(),
    fixedRow(
        column(
            5,
            h3("Top 5 energy generating solar panels on buildings"),
```

```
        p(
          "This graph plots the total annual solar production of the five solar buildings with the highest total production capacity in 2020, and the total annual solar production of the five solar buildings with the highest annual production capacity. Each solar building is distinguished by color."
        ),
        plotOutput("year_top5", width = "100%")
      ),
      column(
        7,
        div(style = "height:50px"),
        sliderInput(
          inputId = "range_slider",
          label = "Range slider",
          min = c_min,
          max = c_max,
          value = c(c_min, c_max)
        ),
        leafletOutput("solar_map", width = "100%")
      ),
    ),
    fixedRow(column(
      5,
      div(style = "height:150px"),
      h3("Energy generation throughout 2020"),
      p(
        "This graph is based on the total daily solar production of the five solar buildings with the highest annual total production capacity and the five solar buildings with the highest annual production capacity. Each solar building is distinguished by color, and the line graph can celebrate the daily production capacity trend of each solar building."
      )
    ),
    column(
      7,
      div(style = "height:180px"),
      plotOutput("trend_energy", width = "100%")
    ))
)



server <- function(input, output, session) {
  output$year_top5 <- renderPlot(year_top5, height = 500, width = 450)
  output$trend_energy <-
    renderPlot(energy_trend, height = 600, width = 700)
  output$solar_map <- renderLeaflet({
    capacity_min <- input$range_slider[1]
    capacity_max <- input$range_slider[2]
    new_solarPanels %>%
      filter(Capacity..kW. >= capacity_min &
               Capacity..kW. <= capacity_max) %>%
      leaflet() %>%
      addTiles() %>%
      addCircleMarkers(
        lng = ~ Longitude,
        lat = ~ Latitude,
        weight = 1,
        color = "yellow",
        opacity = 0.5,
        stroke = T,
        fillOpacity = 0.5,
        fillColor = "green",
        radius = ~ total_generation / 10000,
        label = ~ as.character(MonBuildNum),
        popup = paste("BuildingName:",new_solarPanels$BuildingName,"<br>",
                      "Total Generation:",new_solarPanels$total_generation,"<br>")
      )
  })
}


shinyApp(ui = ui, server = server)
```

Reference:

ggplot_theme: https://stackoverflow.com/questions/1330989/rotating-and-spacing-axis-labels-in-ggplot2
ggplot_facet_wrap:
http://www.zevross.com/blog/2019/04/02/easy-multi-panel-plots-in-r-using-facet_wrap-and-facet_grid-from-ggplot2/
leaflet: https://cran.r-project.org/web/packages/leaflet/leaflet.pdf
UI: https://shiny.rstudio.com/reference/shiny/0.14/fixedPage.html