AE 22: Building a climate risk dashboard - UI

Suggested answers

APPLICATION EXERCISE ANSWERS

MODIFIED
April 23, 2025

Communicating climate risk

FEMA has asked us to build an improved dashboard that visualizes the risk of climate change in the United States.

Based on your submitted designs and our skill level, we will work to implement this Shiny dashboard.

Construct a user interface

Structure the dashboard layout

Useful resources

- Shiny layouts

- [x] Implement a standard navigation bar layout
- [x] Define the dashboard title
- [x] Implement the three page layout
- [x] Include separate sidebars for the national and county views
- [x] Define separate rows for the county tab (map and plot on top, value boxes on the bottom)

```
library(tidyverse)
library(scales)
library(shiny)
library(sf)
library(janitor)
library(ggthemes)
library(colorspace)
library(bslib)
library(bsicons)
library(gt)

# Import data -------------------------------------------------------------
```

```r
# climate risk
climate_risk <- read_rds(file = "data/climate-risk.rds")

# import state and county boundaries
state_sf <- st_read(dsn = "data/states.geojson")
county_sf <- st_read(dsn = "data/counties.geojson")

# combine climate risk with county_sf
climate_sf <- left_join(
  x = county_sf,
  y = climate_risk,
  by = join_by(GEOID == state_county_fips_code)
) |>
  as_tibble() |>
  st_as_sf()

# Define UI ----------------------------------------------------------------
ui <- page_navbar(
  title = "National Risk Index Counties",

  # National Risk Index page
  nav_panel(
    title = "National Risk Index",
    layout_sidebar()
  ),

  # County Details page
  nav_panel(
    title = "County Details",
    layout_sidebar(
      sidebar = sidebar(),
      # state map + plot
      layout_column_wrap(),
      # value boxes
      layout_column_wrap()
    )
  ),

  # Data page
  nav_panel(
    title = "Data"
  )
)
```

```
# Server function
server <- function(input, output) {}

# Run the app
shinyApp(ui = ui, server = server)
```

Position the outputs

> **Useful resources**
>
> - Shiny outputs
> - Make sure to organize your outputs using cards

- [x] First tab
  - [x] National map
- [x] Second tab
  - [x] State map with highlighted county
  - [x] Dot plot for hazard types
  - [x] Value boxes for 4 overall risk measures (overall risk score, expected annual loss, social vulnerability, community resilience)
- [x] Third tab
  - [x] Overall data set - does not require Shiny to implement. Use

```
climate_risk |>
  gt() |>
  opt_interactive()
```

```
library(tidyverse)
library(scales)
library(shiny)
library(sf)
library(janitor)
library(ggthemes)
library(colorspace)
library(bslib)
library(bsicons)
library(gt)

# Import data ------------------------------------------------------------------
# climate risk
climate_risk <- read_rds(file = "data/climate-risk.rds")
```

```r
# import state and county boundaries
state_sf <- st_read(dsn = "data/states.geojson")
county_sf <- st_read(dsn = "data/counties.geojson")

# combine climate risk with county_sf
climate_sf <- left_join(
  x = county_sf,
  y = climate_risk,
  by = join_by(GEOID == state_county_fips_code)
) |>
  as_tibble() |>
  st_as_sf()

# Define UI ----------------------------------------------------------------
ui <- page_navbar(
  title = "National Risk Index Counties",

  # National Risk Index page
  nav_panel(
    title = "National Risk Index",
    layout_sidebar(),
    # Main content
    card(
      card_header("National Risk Map"),
      plotOutput(outputId = "national_map")
    )
  ),

  # County Details page
  nav_panel(
    title = "County Details",
    layout_sidebar(
      sidebar = sidebar(),
      # state map + plot
      layout_column_wrap(
        card(
          card_header("County Map"),
          plotOutput(outputId = "county_map")
        ),
        card(
          card_header("County Hazards"),
          plotOutput(outputId = "county_hazards")
        )
      ),
```

```
      # value boxes
      layout_column_wrap(
        value_box(
          title = "Overall risk score",
          value = textOutput("county_risk"),
          showcase = bs_icon("radioactive")
        ),
        value_box(
          title = "Expected annual loss",
          value = textOutput("county_loss"),
          showcase = bs_icon("trash")
        ),
        value_box(
          title = "Social vulnerability",
          value = textOutput("county_vulnerability"),
          showcase = bs_icon("cone-striped")
        ),
        value_box(
          title = "Community resilience",
          value = textOutput("county_resilience"),
          showcase = bs_icon("emoji-sunglasses")
        )
      )
    )
  ),

  # Data page
  nav_panel(
    title = "Data",
    card(
      card_header("National Risk Index Data"),
      climate_risk |>
        gt() |>
        opt_interactive()
    )
  )
)

# Server function
server <- function(input, output) {}

# Run the app
shinyApp(ui = ui, server = server)
```

Define the inputs

Utilize appropriate sidebars to define all the inputs for the app.

Useful resources

- Shiny inputs
- Shiny reference sheet

National tab

- [x] Dropdown for hazard type

County tab

- [x] Dropdown for county

Get all county names

```
climate_sf |>
  arrange(STATEFP) |>
  pull(county)
```

- [x] Checkboxes for overall risk measures

Get all risk measures

```
# hazard types
hazard_types <- climate_risk |>
  select(contains("hazard")) |>
  colnames()

# human-readable labels
hazard_types_labels <- hazard_types |>
  str_remove(pattern = "_hazard_type_risk_index_score") |>
  make_clean_names(case = "title")
```

```
library(tidyverse)
library(scales)
library(shiny)
library(sf)
library(janitor)
library(ggthemes)
library(colorspace)
library(bslib)
library(bsicons)
```

```r
library(gt)

# Import data -------------------------------------------------------------
# climate risk
climate_risk <- read_rds(file = "data/climate-risk.rds")

# import state and county boundaries
state_sf <- st_read(dsn = "data/states.geojson")
county_sf <- st_read(dsn = "data/counties.geojson")

# combine climate risk with county_sf
climate_sf <- left_join(
  x = county_sf,
  y = climate_risk,
  by = join_by(GEOID == state_county_fips_code)
) |>
  as_tibble() |>
  st_as_sf()

county_names <- climate_sf |>
  arrange(STATEFP) |>
  pull(county)

# define hazard types
hazard_types <- climate_risk |>
  select(contains("hazard")) |>
  colnames()

# create human-readable labels
hazard_types_labels <- hazard_types |>
  str_remove(pattern = "_hazard_type_risk_index_score") |>
  make_clean_names(case = "title")

# create a named character vector for the input
names(hazard_types) <- hazard_types_labels

# Define UI ---------------------------------------------------------------
ui <- page_navbar(
  title = "National Risk Index Counties",

  # National Risk Index page
  nav_panel(
    title = "National Risk Index",
    layout_sidebar(
```

```
      # select between the four risk ratings
      varSelectInput(
        inputId = "risk_var",
        label = "Risk index",
        # select specific columns of data to populate select options
        data = climate_risk |>
          select(`National Risk Index`, `Expected Annual Loss`, `Social Vulnerability`,
          `Community Resilience`)
      )
    ),
    # Main content
    card(
      card_header("National Risk Map"),
      plotOutput(outputId = "national_map")
    )
  ),

  # County Details page
  nav_panel(
    title = "County Details",
    layout_sidebar(
      sidebar = sidebar(
        # extract county/state labels as character vector
        selectizeInput(
          inputId = "county",
          label = "Selected county",
          choices = county_names,
          selected = NULL,
          # custom selectize.js options
          options = list(
            # placeholder text
            placeholder = "Select a county",
            # limit to one county at a time
            maxItems = 1
          )
        ),

        # identify columns with hazard risks and extract column names
        checkboxGroupInput(
          inputId = "hazard_types",
          label = "Hazard types",
          # all possible choices
          choices = hazard_types,
          # initialize plot with all individual hazards
```

```
        selected = hazard_types
      )
    ),
    # state map + plot
    layout_column_wrap(
      card(
        card_header("County Map"),
        plotOutput(outputId = "county_map")
      ),
      card(
        card_header("County Hazards"),
        plotOutput(outputId = "county_hazards")
      )
    ),
    # value boxes
    layout_column_wrap(
      value_box(
        title = "Overall risk score",
        value = textOutput("county_risk"),
        showcase = bs_icon("radioactive")
      ),
      value_box(
        title = "Expected annual loss",
        value = textOutput("county_loss"),
        showcase = bs_icon("trash")
      ),
      value_box(
        title = "Social vulnerability",
        value = textOutput("county_vulnerability"),
        showcase = bs_icon("cone-striped")
      ),
      value_box(
        title = "Community resilience",
        value = textOutput("county_resilience"),
        showcase = bs_icon("emoji-sunglasses")
      )
    )
  )
),

# Data page
nav_panel(
  title = "Data",
  card(
```

```
      card_header("National Risk Index Data"),
      climate_risk |>
        gt() |>
        opt_interactive()
    )
  )
)

# Server function
server <- function(input, output) {}

# Run the app
shinyApp(ui = ui, server = server)
```