**Name-Gurnoor Arora**

**Registration Number-22BDS0020**

**LAB ASSESSMENT 5 & 6- BUS TRACKING SYSTEM**

bus\_tracking\_api.R

library(plumber)

## Warning: package 'plumber' was built under R version 4.4.3

library(jsonlite)

## Warning: package 'jsonlite' was built under R version 4.4.3

library(httr)

## Warning: package 'httr' was built under R version 4.4.3

library(geosphere)

## Warning: package 'geosphere' was built under R version 4.4.  
api\_key <- "hidden"  
origin <- "28.6139,77.2090" # Connaught Place, Delhi  
destination <- "28.6400,77.2500" # Some location in Delhi  
#origin <- "12.9692,79.1559" # VIT Vellore  
#destination <- "12.9165,79.1325" # Vellore Fort  
url <- paste0(  
 "https://maps.googleapis.com/maps/api/directions/json?",  
 "origin=", origin, "&destination=", destination,  
 "&key=", api\_key,  
 "&mode=driving"   
   
)  
response <- GET(url)  
data <- content(response, "parsed")  
route\_steps <- data$routes[[1]]$legs[[1]]$steps  
interpolate\_points <- function(start, end, n = 10) {  
 lats <- seq(start[1], end[1], length.out = n)  
 lons <- seq(start[2], end[2], length.out = n)  
 return(mapply(function(lat, lon) list(lat = lat, lon = lon), lats, lons, SIMPLIFY = FALSE))  
}  
  
# Generate more points along the route  
route\_coords <- list()  
for (step in route\_steps) {  
 start <- c(step$start\_location$lat, step$start\_location$lng)  
 end <- c(step$end\_location$lat, step$end\_location$lng)  
 route\_coords <- append(route\_coords, interpolate\_points(start, end, n = 15))  
}  
  
bus\_index <- 1  
  
#\* @apiTitle bus\_tracking\_api  
  
#\* Get Live Bus Location  
#\* @get /bus-location  
function() {  
 if (bus\_index > length(route\_coords)) {  
 bus\_index <<- 1   
 }  
 location <- route\_coords[[bus\_index]]  
 bus\_index <<- bus\_index + 1   
   
 return((list(lat = location$lat, lon = location$lon)))  
}

## function ()   
## {  
## if (bus\_index > length(route\_coords)) {  
## bus\_index <<- 1  
## }  
## location <- route\_coords[[bus\_index]]  
## bus\_index <<- bus\_index + 1  
## return((list(lat = location$lat, lon = location$lon)))  
## }

frontend.R

Gurnoor

2025-03-31

library(shiny)

## Warning: package 'shiny' was built under R version 4.4.3

library(leaflet)

## Warning: package 'leaflet' was built under R version 4.4.3

library(httr)

## Warning: package 'httr' was built under R version 4.4.3

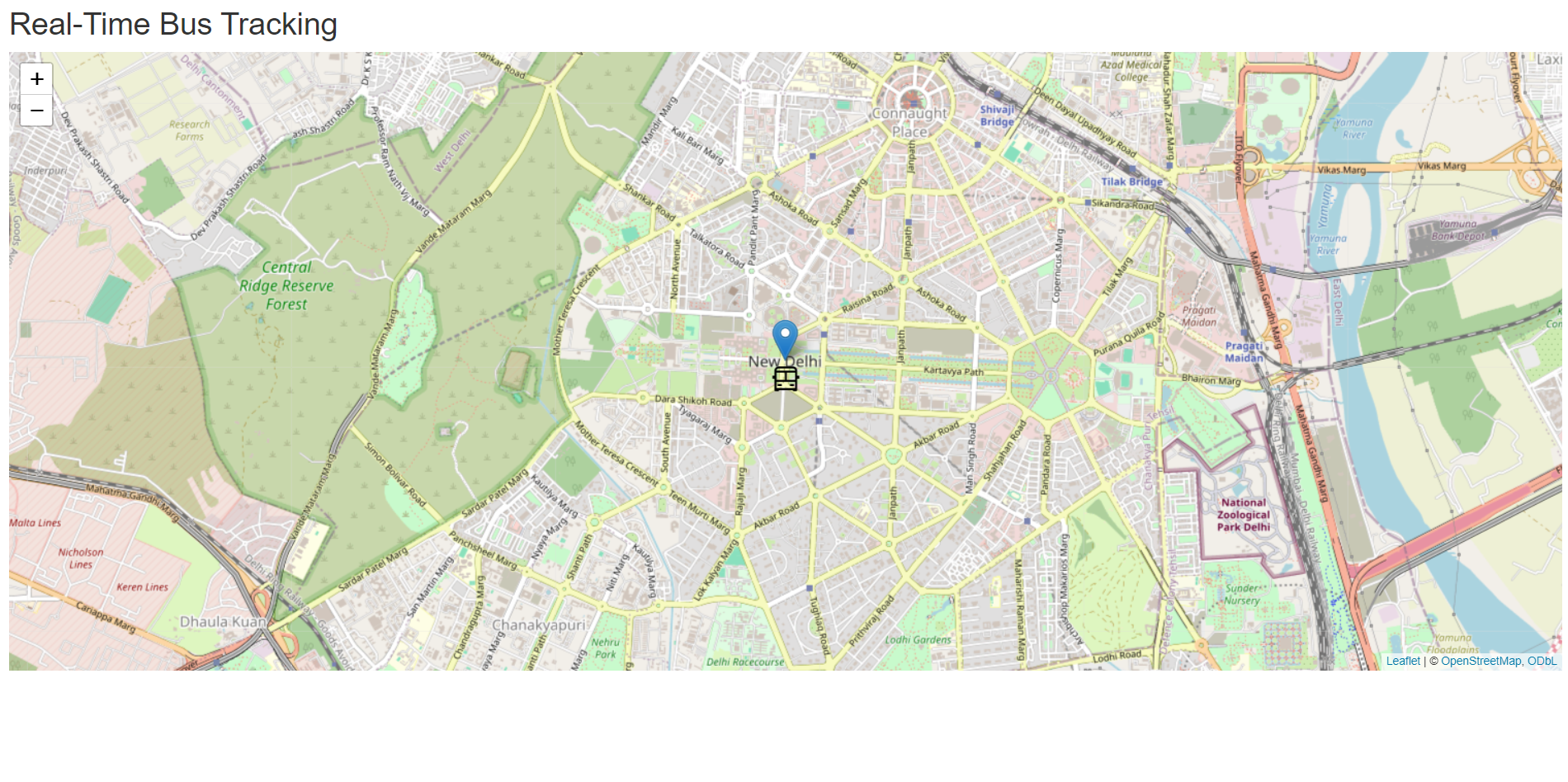
library(jsonlite)

## Warning: package 'jsonlite' was built under R version 4.4.3

##   
## Attaching package: 'jsonlite'

## The following object is masked from 'package:shiny':  
##   
## validate  
API\_URL <- "http://127.0.0.1:8000/bus-location"  
start\_location <- list(lat = 28.6139, lon = 77.2090) # Connaught Place, Delhi  
end\_location <- list(lat = 28.6400, lon = 77.2500) # Destination in Delhi  
ui <- fluidPage(  
 titlePanel("Real-Time Bus Tracking"),  
 leafletOutput("bus\_map", height = "600px"),  
 tags$script(HTML("  
 setInterval(function() {  
 Shiny.onInputChange('update\_bus', Math.random());  
 }, 1000);  
 "))  
)  
server <- function(input, output, session) {  
   
 bus\_location <- reactiveVal(NULL)  
   
  
 updateBusLocation <- function() {  
 res <- try(GET(API\_URL), silent = TRUE)  
 if (!inherits(res, "try-error") && http\_status(res)$category == "Success") {  
 data <- fromJSON(content(res, as = "text"))  
 bus\_location(data)  
 }  
 }  
   
 updateBusLocation()  
 observeEvent(input$update\_bus, {  
 updateBusLocation()  
 })  
   
 output$bus\_map <- renderLeaflet({  
 leaflet() %>%  
 addTiles() %>%  
 setView(lng = start\_location$lon, lat = start\_location$lat, zoom = 14) %>%  
   
 addMarkers(lng = start\_location$lon, lat = start\_location$lat,   
 popup = "Start Location", icon = makeAwesomeIcon(icon = "flag", markerColor = "green")) %>%  
 addMarkers(lng = end\_location$lon, lat = end\_location$lat,   
 popup = "End Location", icon = makeAwesomeIcon(icon = "flag", markerColor = "red"))  
 })  
   
 observe({  
 loc <- bus\_location()  
 if (!is.null(loc)) {  
 leafletProxy("bus\_map") %>%  
 clearGroup("bus") %>%  
 addMarkers(lng = loc$lon, lat = loc$lat, popup = "Bus is here!",   
 icon = makeIcon(  
 iconUrl = "https://img.icons8.com/?size=100&id=16195&format=png&color=000000",  
 iconWidth = 32, iconHeight = 32  
 ),  
 group = "bus")  
 }  
 })  
}  
  
shinyApp(ui, server)

DEMO-



A map of a city

AI-generated content may be incorrect.

A map of a city

AI-generated content may be incorrect.