Paqueterías Utilizadas

Para el desarrollo y funcionamiento de esta aplicación, se utilizaron las siguientes paqueterías:

* **shiny:** Esta es la librería fundamental para crear aplicaciones web interactivas usando R. Proporciona las funciones y herramientas necesarias para construir y lanzar aplicaciones Shiny, incluyendo una variedad de widgets de entrada y salida. Funciones principales ocupadas dentro del proyecto: ui, server, runApp, shinyApp, entre otros.
* **ggplot2:** Creación de gráficos estáticos y estéticamente agradables. Funciones principales ocupadas dentro del proyecto: ggplot, geom\_line, geom\_bar, facet\_wrap, aes, etc.
* **ggmap:** Extiende ggplot2 para visualizar mapas con imágenes de fondo. Funciones principales ocupadas dentro del proyecto: get\_map, ggmap.
* **ggspatial:** Añade capas espaciales y mapas base a gráficos ggplot2. Funciones principales ocupadas dentro del proyecto: geom\_spatial, annotation\_spatial.
* **ggpubr:** Facilita la creación de gráficos ggplot2 para publicaciones. Funciones principales ocupadas dentro del proyecto: ggarrange, ggboxplot, ggscatter, stat\_compare\_means.
* **leaflet:** Creación de mapas interactivos. Funciones principales ocupadas dentro del proyecto: leaflet, addTiles, addMarkers, addPolygons, addCircleMarkers.
* **tidyverse:** Conjunto de paquetes para la manipulación y visualización de datos. Funciones principales: Funciones principales ocupadas dentro del proyecto: dplyr (filter, mutate, select) y tidyr (pivot\_longer, pivot\_wider).
* **colorspace:** Herramientas para gestionar y manipular colores. Funciones principales ocupadas dentro del proyecto: hcl\_colors, diverge\_hcl, sequential\_hcl.
* **vcd:** Visualización de datos categóricos y modelado. Funciones principales ocupadas dentro del proyecto: mosaic, assoc, doubledecker.
* **htmltools:** Herramientas para trabajar con HTML en R, útiles para personalizar la interfaz de usuario en aplicaciones Shiny. Funciones principales ocupadas dentro del proyecto: tags, HTML, tagList, browsable.
* **DT:** Permite la integración de tablas interactivas en aplicaciones Shiny. Funciones principales ocupadas dentro del proyecto: datatable (para renderizar tablas interactivas), formatPercentage (para formatear columnas como porcentajes).
* **sf:** Proporciona una estructura moderna para manejar datos geoespaciales en R. Funciones principales ocupadas dentro del proyecto: st\_read (leer datos espaciales), st\_write (escribir datos espaciales), st\_transform (transformar coordenadas).
* **sp:** Proporciona clases y métodos para la representación y manipulación de datos espaciales. Funciones principales ocupadas dentro del proyecto: SpatialPoints, SpatialPolygons, spTransform.
* **spdep:** Herramientas para el análisis de dependencia espacial y modelado. Funciones principales ocupadas dentro del proyecto: nb2listw (crear listas de vecindad), lag.listw (crear retrasos espaciales), moran.test (test de Moran para autocorrelación espacial).
* **cowplot:** Mejora y extiende las capacidades de ggplot2, especialmente para combinar múltiples gráficos en una sola figura. Funciones principales ocupadas dentro del proyecto: plot\_grid (combinar múltiples gráficos), get\_legend (extraer leyendas de gráficos).
* **shinycssloaders:** Añade animaciones de carga a los elementos de Shiny, lo que es útil para indicar a los usuarios que un gráfico o tabla está siendo procesado. Funciones principales ocupadas dentro del proyecto: withSpinner (añadir animaciones de carga a elementos Shiny).
* **dygraphs:** Permite interactividad para visualizar series temporales en R. Es especialmente valiosa cuando se trabaja con una gran cantidad de puntos de datos a lo largo del tiempo. Funciones principales ocupadas dentro del proyecto: dygraph(), dyRangeSelector() y dyHighlight().
* **plotly:** Es una biblioteca de gráficos interactivos que permite la creación de una variedad de visualizaciones, desde gráficos de dispersión y de barras hasta mapas y gráficos 3D. Funciones principales ocupadas dentro del proyecto: plot\_ly(), add\_trace() y layout().
* **xts:** Es una biblioteca diseñada específicamente para la manipulación y análisis de series temporales. En el contexto de Shiny, xts puede ser utilizado para preparar y transformar datos antes de visualizarlos con dygraphs o plotly. Funciones principales ocupadas dentro del proyecto: xts(), period.apply(), lag() y to.period().

Bases de datos

*Asentamientos.rds*

* **ID:** Identificador único del asentamiento.
* **cvegeo:** Clave geográfica del asentamiento.
* **entidad:** Código de la entidad.
* **nom\_ent:** Nombre de la entidad (ej. Guanajuato).
* **JUR:** Código de jurisdicción.
* **NOMJUR:** Nombre de la jurisdicción (ej. SALAMANCA).
* **mun:** Código del municipio.
* **nom\_mun:** Nombre del municipio (ej. Yuriria).
* **loc:** Código de la localidad.
* **nom\_loc:** Nombre de la localidad (ej. La Cabaña de los Tepamez).
* **ageb:** Área Geoestadística Básica.
* **mza:** Manzana.
* **pobtot:** Población total.
* **pobfem:** Población femenina.
* **pobmas:** Población masculina.
* **p\_15a49\_f:** Población femenina de 15 a 49 años.
* **p\_18ymas:** Población de 18 años y más.
* **P\_15a19:** Población de 15 a 19 años.
* **Prop\_15a19:** Proporción de población de 15 a 19 años.
* **Prop\_15a19Sder:** Proporción de población de 15 a 19 años con seguro de salud.
* **P20ymasSder:** Población de 20 años y más con seguro de salud.
* **PropP20ymasSder:** Proporción de población de 20 años y más con seguro de salud.
* **P60ymasSder:** Población de 60 años y más con seguro de salud.
* PropP60ymasSder: Proporción de población de 60 años y más con seguro de salud.
* **psinder:** Población sin derechohabiencia.
* **pder\_ss:** Población con derecho a servicios de salud.
* **pder\_imss:** Población con derecho al IMSS.
* **pder\_istee:** Población con derecho al ISSTEE.
* **pafil\_pdom:** Población afiliada a un programa de salud.
* **pder\_segp:** Población con seguro popular.
* **pder\_imssb:** Población con IMSS Bienestar.
* **pafil\_ipriv:** Población con seguro privado.
* **pafil\_otrai:** Población con otro tipo de afiliación.
* **PobResp:** es la población de responsabilidad de la SSA
* **Ai:** Medida de accesibilidad. Un número adimensional que representa la accesibilidad a los servicios de salud utilizando el método de Huff y áreas de capatacion flotantes
* **unidades:** Unidades.
* **obesidad:** Porcentaje de obesidad.
* **hipertension:** Porcentaje de hipertensión.
* **diabetes:** Porcentaje de diabetes.
* **clues:** Códigos de unidades médicas.
* **tviaje:** Tiempos de viaje a las unidades médicas.
* **StdAi:** Desviación estándar de Ai.
* **UMed1:** Unidad médica más cercana.
* **Umed1tv:** Tiempo de viaje a la unidad médica más cercana.
* **Umed1tvcut:** Categoría de tiempo de viaje a la unidad médica más cercana.
* **propPobResp: es la porporción de la población total que es responsabilidad d la SSA**

Por ejemplo, el asentamiento con ID 541027 está ubicado en Guanajuato, en la jurisdicción de SALAMANCA, en el municipio de Yuriria, y se llama "La Cabaña de los Tepamez". Tiene una población total de 24 personas, de las cuales 13 son mujeres y 11 son hombres. Además, proporciona información sobre las condiciones de salud y acceso a servicios de salud de la población de ese asentamiento.

*ECNT.rds*

El dataframe ECNT contiene datos relacionados con la prevalencia de obesidad, hipertensión y diabetes en los municipios de México para el año 2018. A continuación se presenta una descripción general de las columnas:

* **cvegeo:** Clave geográfica del municipio.
* **obesidad:** Porcentaje de prevalencia de obesidad en el municipio.
* **hipertension:** Porcentaje de prevalencia de hipertensión en el municipio.
* **diabetes:** Porcentaje de prevalencia de diabetes en el municipio.
* **Calidad\_Obesidad:** Calidad de la medida de obesidad.
* **Calidad\_Hipertensión:** Calidad de la medida de hipertensión.
* **Calidad\_Diabetes:** Calidad de la medida de diabetes.

Por ejemplo, el municipio con clave 1001 tiene un 31.49% de prevalencia de obesidad, 14.94% de hipertensión y 7.50% de diabetes. Las medidas de calidad para estas prevalencias son todas 1.

*pobcens0020.rds*

El dataframe pobcens0020 contiene datos demográficos extraídos de los censos de población en México, específicamente a nivel municipal. A continuación, se describe el contenido de las columnas:

* **clave:** Clave o identificador de un área o región específica.
* **anio:** Año.
* **edad:** Rango de edad de la población. Por ejemplo, "0a4" representa edades de 0 a 4 años, "05a9" representa edades de 5 a 9 años, y así sucesivamente.
* **sexo:** Género de la población.
* **pob:** Número de individuos o población en ese rango de edad y género específicos para la región o área indicada por la "clave".

*pobreza.rds*

El dataframe pobreza contiene datos relacionados con indicadores de pobreza para los municipios de México. A continuación, un resumen del contenido de las columnas:

* **Clave\_de\_entidad:** Clave o identificador numérico de la entidad federativa.
* **Entidad\_federativa:** Nombre de la entidad federativa (por ejemplo, "Aguascalientes").
* **Clave\_de\_municipio:** Clave o identificador numérico del municipio dentro de la entidad federativa.
* **Municipio:** Nombre del municipio (por ejemplo, "Aguascalientes", "Asientos").
* **año:** Año.
* **Num\_Indicador:** Número o código que identifica un indicador específico.
* **Decripción:** Descripción del indicador.
* **Porcentaje:** Porcentaje asociado al indicador.
* **Población:** Número de individuos o población asociada al indicador en el municipio específico.
* **Carencias\_promedio:** Valor promedio de los diferentes tipos de carencias en el estudio de pobreza multidimensional de CONEVAL
* **Acrónimo:** Abreviatura o acrónimo del indicador.

*servicios.rds*

Contiene datos relacionados con los servicios brindados por las unidades médicas en diferentes entidades y municipios de México.

* **Entidad:** Nombre de la entidad federativa (ejemplo: "Aguascalientes").
* **Jurisdicción:** Jurisdicción sanitaria a la que pertenece la unidad médica.
* **Municipio:** Nombre del municipio en el que se ubica la unidad médica.
* **CLUES:** Código de identificación de la Unidad de Salud.
* **Nombre\_de\_la\_unidad\_médica:** Nombre específico del establecimiento de salud.
* **NOMBRE\_TIPO\_ESTABLECIMIENTO:** Tipo de establecimiento (ejemplo: "DE CONSULTA EXTERNA").
* **CLAVE\_TIPO\_ESTABLECIMIENTO:** Código que representa el tipo de establecimiento.
* **NOMBRE\_DE\_TIPOLOGIA:** Descripción detallada del tipo de establecimiento (ejemplo: "URBANO DE 04 NÚCLEOS BÁSICOS").
* **CLAVE\_DE\_TIPOLOGIA:** Código que representa la tipología del establecimiento.
* **CLAVE\_DE\_SUBTIPOLOGIA:** Código de subtipología del establecimiento.
* **De columna TotalConsulta a DMcntrlTot:** Estadísticas relacionadas con las consultas y la atención de enfermedades específicas en la unidad médica por género y grupos de edad. Incluyen datos sobre:
  + Total de consultas.
  + Consultas de primera vez en el año.
  + Atenciones relacionadas con enfermedades crónicas no transmisibles (ECNT).
  + Atenciones de planificación familiar (PF).
  + Atenciones en unidades médicas especializadas (UNEME).
  + Atenciones relacionadas con embarazos (EMB).
  + Detecciones y manejo de diabetes mellitus (DM) y sus complicaciones.

Por ejemplo, el "CENTRO DE SALUD URBANO LA SALUD" en Aguascalientes tuvo 10,014 consultas totales en el período reportado, de las cuales 4,181 fueron consultas de primera vez en el año. Además, se atendieron 96 casos de ECNT en la primera visita del año, 134 casos de planificación familiar en la primera visita, entre otros datos.

cv\_graf\_mun.rds

Este conjunto de datos contiene información sobre el total de casos de covid-19 por semana, clave de entidad y municipio.

* **Cvegeo:** clave de entidad y municipio
* **Fecha\_semana\_min:** fecha semana epidemiológica
* **Positivos:** número de casos covid-19 positivos.
* **Negativos:** número de casos covid-19 negativos
* **Acumuluado:** número de casos covid-19 acumulados en la serie de tiempo.

cv\_txt\_mun.rds

El dataframe cv\_text\_mun contiene información acerca de casos, defunciones y personas hospitalizadas por Covid-19. Esta información es presentada por semana epidemiológica, clave de entidad y municipio.

* **Semena\_corte:** fecha mínima en semana epidemiológica.
* **Cvegeo:** clave de entidad y municipio.
* **Personas\_estudiadas:** total de personas estudiadas.
* **Positivos:** números de casos covid-19 positivos.
* **Defunciones\_documen:** total de defunciones documentadas.
* **Defunciones\_totales:** total de defunciones.
* **Hospitalizados:** total de hospitalizados.
* **Sem\_cons:** número de mana epidemiológica.

frac\_defmun.rds

En este conjunto de datos se almacena información sobre las 10 princípiales causas de defunción en cada municipio.

* **Cons:** Numero de causa de defunción.
* **Cve\_geo:** clave de entidad y municipio.
* **Sexo:** clasificación de causas por sexo. (Hombres, Mujeres, General)
* **GBD165\_CAUSA\_DEF:** descripción de la causa de la defunción de acuerdo a la clasificación GBD.
* **Total:** total de defunciones.

histo\_mun\_def.rds

El dataframe histo\_mun\_def.rds, contiene información sobre las 10 princípiales causas de defunción en cada municipio. La información se presenta por año en un periodo de 2010 a 2020.

* **Cons:** Numero de causa de defunción.
* **Cve\_geo:** clave de entidad y municipio.
* **Sexo:** clasificación de causas por sexo. (Hombres, Mujeres, General)
* **GBD165\_CAUSA\_DEF:** descripción de la causa de la defunción de acuerdo a la clasificación GBD.
* **Total:** total de defunciones a lo largo del periodo.
* **Tot\_con:** contiene un vector del total de defunciones ordenados por mes año de la defunción.
* **Fechs\_con:** contiene un vector con las fechas agrupadas por mes y año de la defunción.

treemapmunic.rds

Este conjunto de datos contiene información sobre las 10 princípiales causas de defunción en cada municipio, agrupadas por categoría y subcategoría.

* **Cons:** Numero de causa de defunción.
* **Id\_num:** clave de entidad y municipio.
* **GBD165\_CAUSA\_DEF:** descripción de la causa de la defunción de acuerdo a la clasificación GBD. (Categoría)
* **Nom\_causa\_def:** descripción de la causa de la defunción de acuerdo a la clasificación GBD. (Sub categoría)
* **Total:** total de defunciones.

egresos\_no\_obs.csv

En este conjunto de datos se almacena información sobre las 10 princípiales afecciones de egresos hospitalarios en cada municipio.

* **Cons:** Numero de afección en egresos.
* **Cve\_geo:** clave de entidad y municipio.
* **Afec:** descripción de la afección principal del egreso hospitalario.
* **Total:** total de egresos.

anios\_egresos.csv

El conjunto de datos anios\_egresos.csv**,** contiene información sobre las 10 princípiales afecciones de egresos en cada municipio. La información se presenta por año en un periodo de 2008 a 2021.

* **Cons:** Numero de afección en egresos.
* **Cve\_geo:** clave de entidad y municipio.
* **GBD165\_CAUSA\_DEF:** descripción de la afección principal del egreso hospitalario.
* **Anio:** Año de la base de egresos.
* **Total:** total de egresos por año.

treemap\_egre\_c.csv

* **Cons:** Numero de afección en egresos.
* **Id\_num:** clave de entidad y municipio.
* **GBD165\_CAUSA\_DEF:** descripción de la afección principal del egreso hospitalario. (Categoría)
* **Nom\_causa\_def:** descripción de la afección principal del egreso hospitalario. (Sub categoría)
* **Total:** total de egresos.

*entidadesgeo.rds, munisgeosimp.rds, munisogr, munisogrneighb.rds*

Los archivos albergan datos geográficos esenciales para visualizar mapas en la aplicación.

Lineamientos de Diseño y Codificación

El desarrollo de esta aplicación Shiny se ha guiado por principios de diseño y codificación que buscan garantizar un código limpio, eficiente y mantenible. Estos son los pilares que han sostenido el proyecto desde sus inicios y que han permitido que la aplicación sea robusta y confiable. Los aspectos clave de los lineamientos son los siguientes:

* Estructura del Código: Se ha buscado mantener una estructura modular, donde cada función tiene un propósito claro y definido, evitando funciones excesivamente largas o complejas.
* Nomenclatura: Se ha adoptado una convención de nombres consistente para variables, funciones y archivos. Esto facilita la lectura y comprensión del código, especialmente para nuevos colaboradores.
* Documentación Interna: Cada función y módulo cuenta con comentarios descriptivos que explican su propósito, parámetros y retornos. Además, se han incluido comentarios en bloques de código más complejos para clarificar su lógica.
* Organización de Archivos: Los archivos y directorios están organizados de manera lógica, agrupando funcionalidades relacionadas y separando la interfaz de usuario, el servidor y los datos.

Para aquellos que estén interesados en contribuir, es esencial revisar y adherirse a estos lineamientos. Esto asegura que el código mantenga un estándar alto y que la aplicación continúe siendo fácil de modificar y expandir en el futuro.

Colaboraciones

La colaboración es fundamental para mejorar y expandir las capacidades de esta aplicación Shiny. Si identificas oportunidades de mejora, correcciones o incluso nuevas características que podrían enriquecer la experiencia del usuario, ¡estamos abiertos a tus aportaciones!

Pasos para contribuir a la aplicación Shiny de descarga:

* Entender la Funcionalidad: Familiarízate con la aplicación y su principal función de descarga. Esto te ayudará a identificar áreas de mejora o posibles extensiones.
* Clonar el Proyecto: Haz una descarga del proyecto e implementa el mismo en tu máquina local.
* Implementar Cambios: Realiza las modificaciones o añade las características que consideres pertinentes. Si estás añadiendo una nueva funcionalidad, asegúrate de que no interfiera con la función principal de descarga.
* Probar la Aplicación: Antes de enviar tus cambios, prueba la aplicación localmente para asegurarte de que todo funcione correctamente y de que la función de descarga sigue siendo operativa.
* Documentar: Añade comentarios explicativos a cualquier código nuevo o modificado. Si has añadido una nueva funcionalidad, considera actualizar el README o cualquier otra documentación relevante.
* Contacto: Con gusto puedes enviarnos un correo electrónico a uisp@insp.mx y con gusto te asistiremos en tu proceso de colaboración. Tu feedback y aportaciones son valiosos para nosotros, y estamos aquí para ayudarte en cada paso del camino.