

UNIVERSIDAD  
**AUSTRAL**



# EventIt

## TPF

Universidad Austral: Ingeniería Biomédica

Materia: Introducción a la programación 2

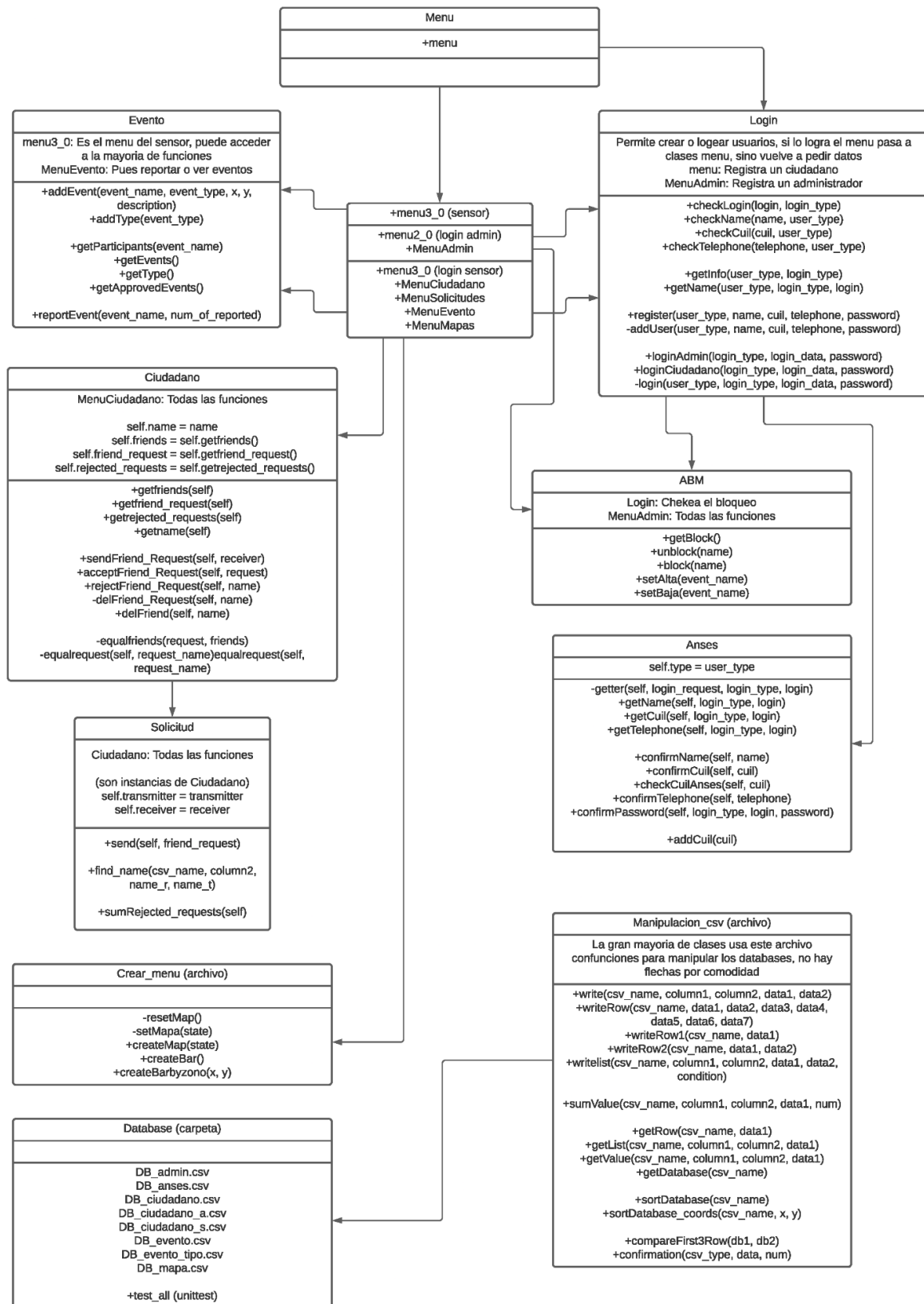
Profesores: Ezequiel Lamonica, Rodrigo Peralta.

Alumnos: Kishkill Ana, López Valentina, Nader Capra Bianca, Sánchez Facundo, Sole Valentina, Viglione Clementina.

Grupo: 13

Fecha de entrega: 30/11/2021

## Diagrama del programa:



## Diseño principal

Una de las principales columnas de este código es la gran dependencia del uso de database, el cual nos permite guardar todo lo que se realice en el programa y permite el acceso a esta información almacenada en ocasiones posteriores. Esto se logra mediante el archivo “Manipulación\_csv”, donde la mayoría o incluso todas las clases acceden con el fin de manipular los datasets, ya sea escribir nuevas líneas, obtener un database completo, modificar valores, comparar, confirmar valores y reordenar los databases. Se utilizaron la librerías pandas, csv y seaborn.

## Inicio

Para comenzar con el código, empezamos con lo último que es el menú, pues después de crear todas las clases, database, etc, el menú fue uno de los últimos pasos. Sin embargo, justamente por ser el paso final es aquel que permite explicar las funcionalidades de cada clase.

## Login

Al inicio se preguntará si se inicia como administrador, ciudadano, sensor<sup>1</sup> o si se registrara un nuevo ciudadano. Para esto se llama a la clase Login el cual puede chequear el login, nombre, teléfono, cuil de un tipo de usuario (administrador o ciudadano). Todos estos datos son cruzados con el dataset del ANSES que son accedidos mediante la clase Anses, encargada de manejar todos los datos de ciudadanos o CUIL individuales, validando la información, obteniendo información de un CSV por tipo de usuario y registra por nombre, contraseña, CUIL, teléfono. Con esto se chequea que estos datos ya no estén ingresados en el CSV y luego los registra. Y por último ingresa tipos de usuario, el cual requiere un tipo de usuario y un dato de ingreso, fijándose que se hayan completado todos los datos (se confirma con el dataset del ANSES) y también verifica que el usuario no esté bloqueado en el ABM entonces se permite el ingreso y si alguno de los datos anteriores falla se lanza una excepción. Existen tres métodos de ingreso, el método “login” es el método base donde son ejecutadas las funciones, mientras que “loginAdmin” e “loginCiudadano” ejecutan las funciones de “ingreso” pasándole el parámetro “user\_type”, ya sea Admin o Ciudadano según corresponda.

---

<sup>1</sup> Ingreso como sensor implementado después de la demo.

## Ciudadano

En caso que se inicie un ciudadano, los mismos tienen que poder indicar contactos de interés con otro ciudadano, aceptando mutuamente dicha relación, se les dio la responsabilidad a estos de enviar solicitudes a otros ciudadanos además de reportar un evento y así vincular a algunos de sus contactos de interés.

## Solicitudes

Para manejar el tema de las solicitudes entre ciudadanos se creó la clase Solicitud que cuenta con un emisor y un receptor, la cual interactúa con el ciudadano al enviar la solicitud. Además para resolver otro problema también se le dio la responsabilidad a esta clase de hacer el conteo de las solicitudes rechazadas. Cuando se llega a 5, un método en el ABM se encarga de cambiar el atributo que está dentro de usuario que indica su estado (bloqueado/desbloqueado).

## Eventos

Para los eventos, únicamente se reservan las funciones de ver los eventos creados y reportar eventos, ya sea uno mismo o la lista de contactos de interés, siempre tomando el número de amigos y sumándolo al evento en cuestión. También se puede acceder a los mapas disponibles y rankings.

## Mapas

Los mapas son creados por el archivo de Crear\_mapas, donde se puede crear un mapa de calor que lo primero que hace es tomar una matriz de 0 donde cada cero está ubicado en una respectiva coordenada que van de 0 a 10 (ya sea x o y), después se hace un loop que mira coordenada a coordenada cada evento, y en caso que haya se sumarán la cantidad de participantes a la matriz.

A la hora de plantear el punto 3 se utilizaron las librerías pandas, csv (ya utilizada anteriormente) y seaborn (la cual al ser una librería que sobrescribe la de matplotlib, tuvo que ser importada). Para resolver el primer ítem se utilizó específicamente la librería heatmap de Seaborn y la matriz creada anteriormente del tipo csv. Heatmap de seaborn permite representar de manera de gráfico rectangular el database o dataset pasado como argumento (el principal de los argumentos que admite la función heatmap, los demás).

Para seguir con el segundo ítem, se hizo uso especialmente de la librería pandas, la cual fue utilizada para filtrar y tabular la información de los eventos en las zonas y lograr visualizar los picos.

## Administradores

Los administradores tienen el poder de acceder a la clase ABM con todas sus funciones, que van de dar de baja y alta distintos eventos hasta bloquear y desbloquear ciudadanos. Para dar de alta un evento, se llama al método “setAlta” lo cual modifica el atributo “State” que se encuentra dentro de evento, que por default es “Baja”; para darlo de baja, se utiliza el método “setBaja” que funciona exactamente igual al otro. Para los ciudadanos es lo mismo que “setAlta” y “setBaja”, pues mediante un nombre se puede cambiar el atributo “Blocked” a “Yes” para bloquear y “No” para desbloquear.

## Sensores

Con respecto a los sensores, lo que se implementó fue una instancia de la clase Eventos, con todos los métodos. Los tres modifican una base de datos CSV, la cual recolecta los datos de cada evento que se cree con su correspondiente sensor. Los métodos son “addEvent” (el cual recibe como atributo nombre, tipo de evento existente, coordenadas en X y en Y y una descripción), “addType” (el cual añade un tipo de evento nuevo) y “reportEvent” (que recibe como atributo un número, el cual representa la cantidad de acompañantes que irán al evento).

## Unittest

Para finalizar, se incluyó un archivo test\_all.py en la carpeta Database el cual es un unittest que permite probar todas las funciones (que no sean mapas). Debido a que se usan mucho los databases, solo funcionará la primera corrida y después en algunos test dará fallo aunque funciones como es debido. Para arreglar esto, hay que entrar en cada csv y quitar (ctrl + z) todos los cambios realizados.