

**Integrantes:** María Alessandra Mejía Navarro

José Ángel Zapata Rosales

Jorge Evaristo Garcia Flores

Abigail Eunice Rocha Osejo

Norman Santiago Manzanares Peña

**Carrera:** Ingeniería en Sistemas de Información

**Facultad:** Ciencia, Tecnología y Ambiente

**Docente:** José Durán

**Práctica de familiarización**

# Índice

[**Índice**](#_jxaxi5nqqe4e) **2**

[**Objetivo general**](#_zdpn7dqxg32y) **3**

[**Planteamiento del problema:**](#_2zko6ee59l5e) **3**

[**Clases y atributos:**](#_cqjnt1tcjxyy) **8**

# Objetivo general

* Diseñar estrategias de solución, las cuales se disponen de una forma secuencial y lógica con el fin de facilitar el proceso de organización en los voluntariados a través de la escritura de un código fuente el cual en principio, será desarrollado en el lenguaje de programación java.

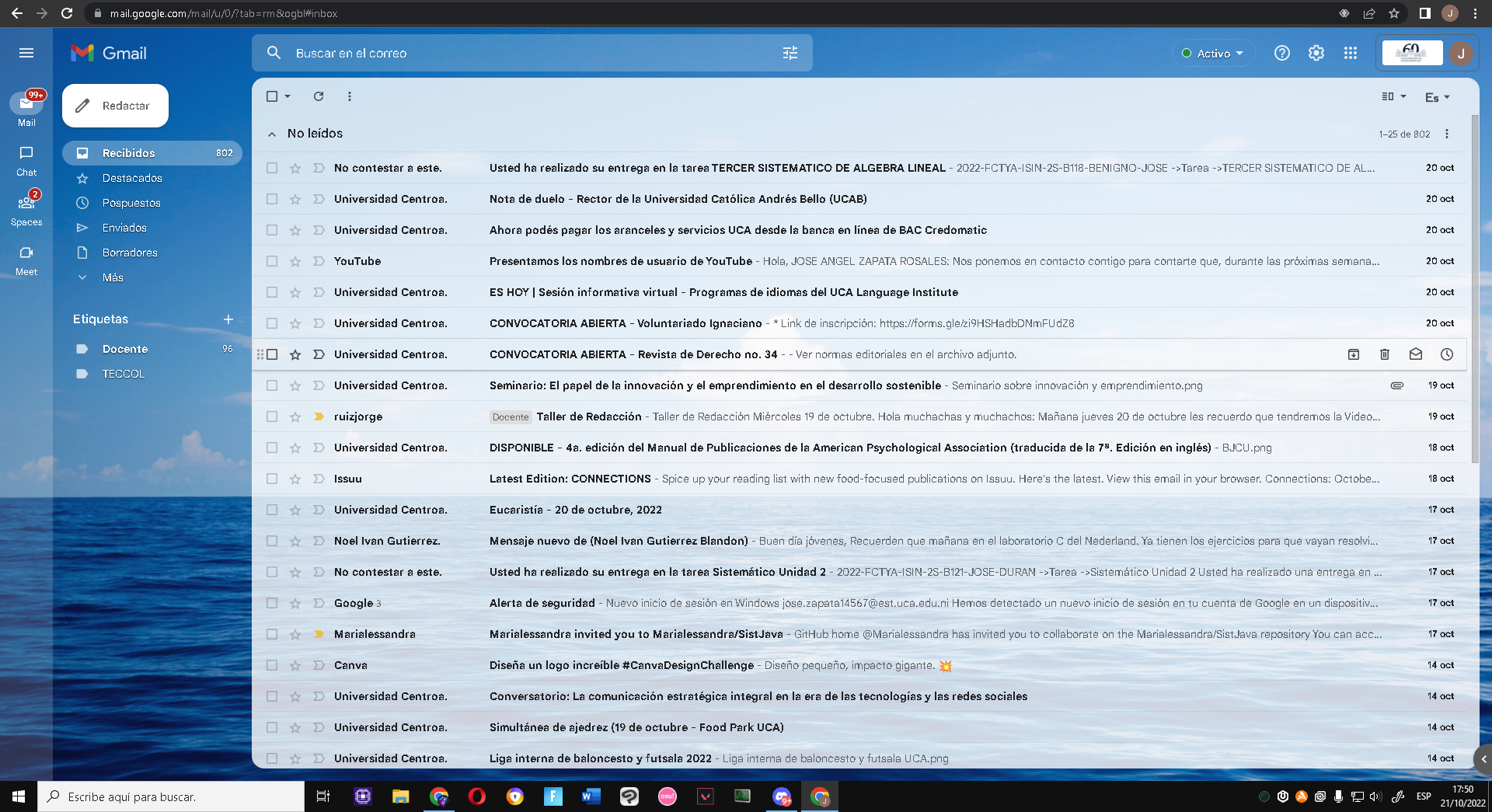
Objetivo específico

* Crear una aplicación de escritorio que sea funcional de acuerdo al planteamiento del problema.
* Desarrollar una aplicación capaz de agregar personas a los voluntariados, eliminar, actualizar las fechas y actividades de los voluntariados.

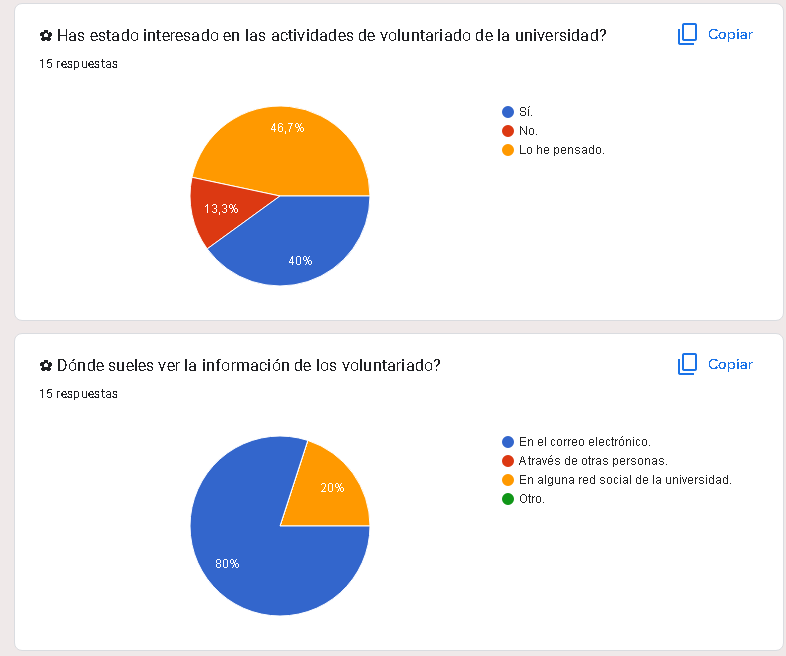
# Planteamiento del problema:

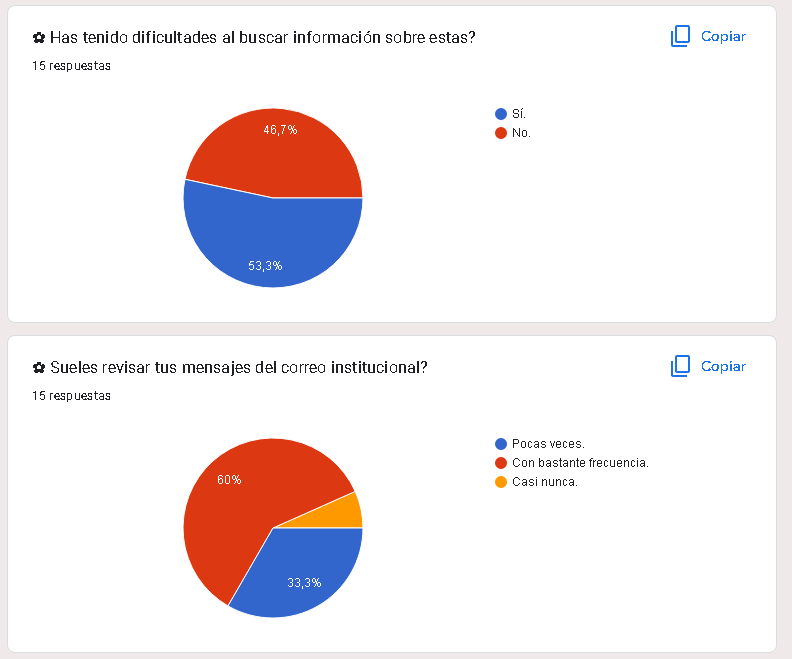
En nuestros meses como estudiantes de la Universidad Centroamericana (UCA), hemos estado haciendo uso de los servicios de los programas y software que nos brinda. En estos, se avisan tanto de las tareas y trabajos que hay que entregar, mensajes de profesores o compañeros, eventos y actividades que la institución organiza, entre muchos otros. Entre las actividades se encuentran los voluntariados, que suelen ser variados y de distintos tipos como lo son el proyecto ambiental, proyecto generando vidas, proyecto San Francisco Xabier, proyecto natras, proyecto colibrí, proyecto Roberto Clemente, Nuestra Señora de Guadalupe, proyecto pajarito azul, proyectos chateles y proyecto scout.

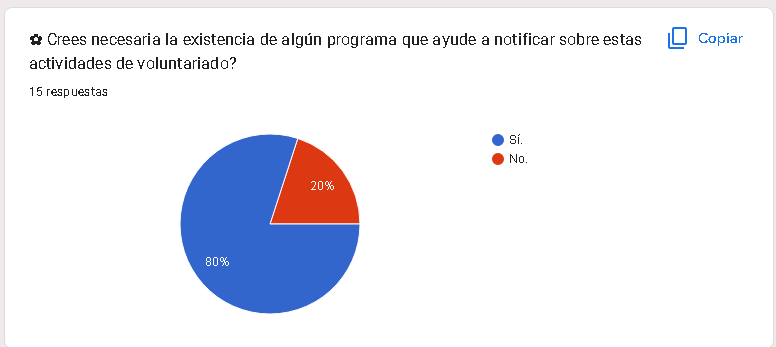
Sabemos que el estar recibiendo respuestas en formularios de google y luego apuntar a los voluntarios suele ser tedioso y muchas veces se pierde la información y organización de estos. Por lo tanto el propósito de nuestro proyecto es facilitarles a los administradores de los voluntariados de la universidad centroamericana (UCA), poder agregar a los voluntarios, eliminar voluntarios, actualizar las fechas y actividades de los voluntariados, etc.



Podemos ver cómo a veces puede ser complicado acceder a estos correos ya que se pueden perder entre todos los mensajes que llegan al correo institucional y de esta manera ser de difícil acceso para los interesados, o incluso esto puede ser motivo de la pérdida del interés, o simplemente no estén al tanto de cuando están activas estas actividades ya que no todos los estudiantes revisan detalladamente su correo electrónico.







Como se muestra en las imágenes, realizamos una encuesta a estudiantes dentro de la Universidad Centroamericana (UCA) de distintas facultades y carreras, en las que observamos cómo los alumnos que están interesados en participar en los voluntariados, debido a diferentes situaciones, no revisan constantemente sus correos que es el medio por el cual se notifica de estos eventos, y que muestran problemas e inconvenientes al hacer búsqueda de la documentación para la debida asistencia; por lo cual concluimos que verdaderamente existe una necesidad de mejorar el sistema de envío, recepción e inscripción en estos.

Es por eso que como grupo, dialogamos y discutimos sobre estos inconvenientes que se presentan y obstaculizan la cooperación y contribución en estas labores; y así llegamos a proponer la elaboración de un software para un programa/aplicación especializado y exclusivo para estas actividades, en las que mediante interfaces daremos las opciones para notificar sobre cualquier información y sus usuarios tengan más facilidad para acceder a cualquier anuncio y notificación para estar al tanto de estas actividades sin tantas complicaciones.

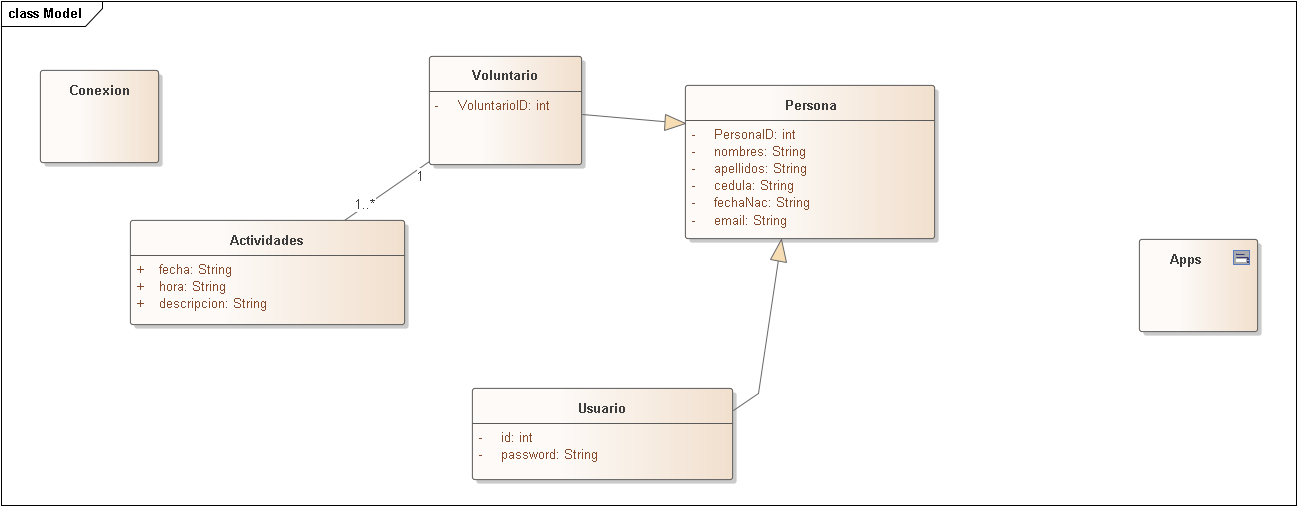
Nuestro programa está pensado que sea algo sencillo, pero con las funciones necesarias e indispensables, empleando distintas herramientas proporcionadas para la programación orientada a objetos en Java, y así crear un software que sea fácil de usar y bastante práctico, así pensamos ayudar a mejorar la comunicación mediante recursos virtuales de la comunidad académica de la Universidad Centroamericana (UCA). Esperamos poder proporcionar un sistema adaptable a las necesidades tanto de los administradores como a los jóvenes que deseen participar, y que sea un material de utilidad versátil y cómodo, con interfaces apropiadas para la circulación de avisos y una mejor percepción de notificaciones.

Resumiendo, esperamos crear un programa efectivo y útil para mejorar la información, inscripción, notificación y comunicación de cualquier noticia, dato, informe, etc; relacionado con el sistema de voluntariado para la comunidad estudiantil, organizado por la Universidad Centroamericana (UCA).

***Preguntas encuesta:***

* ¿Has estado interesado en las actividades de voluntariado de la universidad?
* ¿Dónde sueles ver la información de los voluntariados?
* ¿Has tenido dificultades al buscar información sobre estas?
* ¿Sueles revisar tus mensajes del correo institucional?
* ¿Crees necesaria la existencia de algún programa que ayude a notificar sobre estas actividades de voluntariado?

# Clases,atributos y métodos:



Documento de requerimientos:

La universidad requiere un sistema para los administradores que les permita realizar el registro de nuevas personas en los distintos voluntariados. Las diferentes ramas de los voluntariados son educativos, scouts, ambientales e infantiles.

Las personas interesadas se pueden inscribir mediante un formulario de google, el cual les pide los siguientes datos:

* Nombre y apellidos
* Sexo
* Edad
* Fecha de nacimiento
* Número de teléfono
* Compañía telefónica
* Correo (Institucional o personal)
* Id del carnet o cédula
* Talla de camisa
* Una foto

Nosotros queremos crear un sistema automatizado con una base de datos en SQL server 2019 y una aplicación en Apache netbeans 14, la cual va a permitir llevar un control más eficiente de los voluntarios; así como realizar distintas funciones como:

* Agregar voluntarios nuevos
* Eliminar voluntarios
* Buscar voluntarios
* Ver los eventos de los distintos voluntariados
* Agregar nuevos eventos

Cómo será un sistema orientado a los administradores se le pedirá un usuario y contraseña para acceder al sistema y sus distintas funciones.

Secuencia de eventos:

* La pantalla mostrará el menú de inicio de sesión
* El administrador ingresará el usuario y contraseña
* Sí ambos datos introducidos son correctos se iniciará sesión en el sistema.
* Se mostrará el menú que contiene las siguientes opciones:

- Registrar nuevo voluntario

- Eliminar voluntario

- Buscar voluntario

- Mostrar voluntarios

- Mostrar eventos por voluntariado

* Si selecciona la primera opción se mostrarán los distintos campos a rellenar para registrar una nueva persona
* Si selecciona la segunda opción se le pedirá al administrador que ingrese el código del voluntario a eliminar, se mostrará el voluntario a eliminar para que se verifique si es el que se quiere eliminar y se le preguntará al admin si se desea eliminar el voluntario mostrado, tendrá que digitar SI en caso de que lo quiera eliminar o NO si quiere cancelar la operación.
* Si selecciona la tercera opción se pedirá el código del voluntario que desea buscar, si este existe se mostrará con todos sus datos pertinentes.
* Si selecciona la cuarta opción se mostrará todos los voluntarios registrados hasta la fecha.
* Si selecciona la quinta opción se mostrarán los eventos que posee cada voluntariado.

**Sustantivos y frases nominales**

* voluntariado
* login
* voluntario
* eventos
* Tbl voluntario
* avisos
* Software
* Información

**Clases identificadas**

* **Conexión**
* **Actividades**
* **Voluntario**
* **Usuario**
* **Persona**
* **Apps**

**Identificación de los atributos de las clases**

**Actividades:**

id

actividad

fecha

hora

descripción

**Voluntario**

id: int

**Usuario**

id: int

password: String

**Persona:**

nombre: String

apellidos: String

cedula: String

fechaNac: String

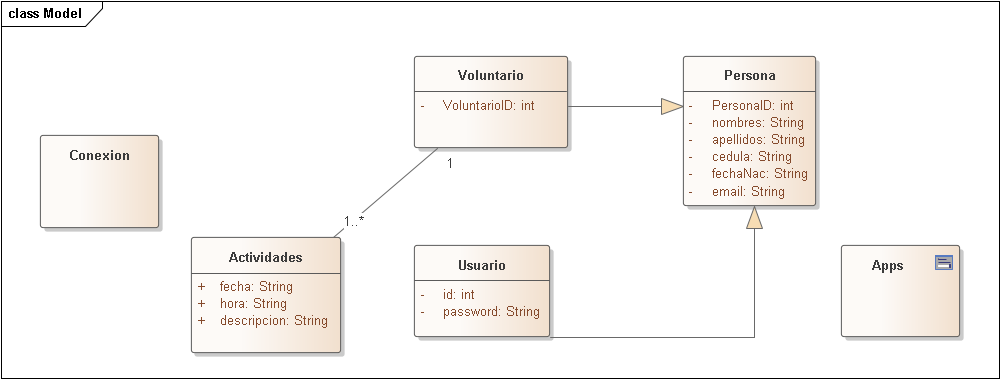
celular: String

email: String

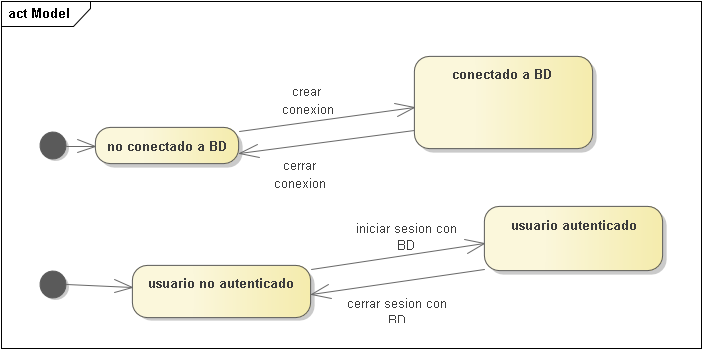
**Conexión**

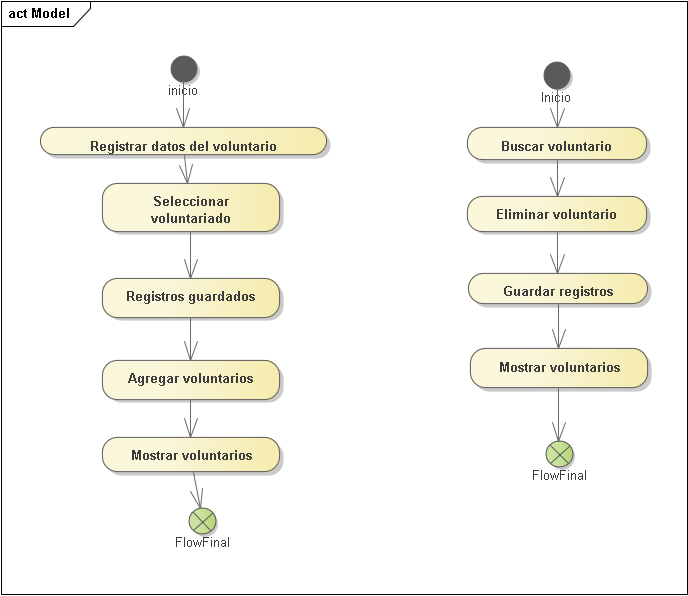
**Apps**

**Diagrama de clases con atributos**

****

**Identificación de los estados y actividades de los objetos**

****

****

**Identificación de las operaciones de las clases**

1. Verbos y Frases verbales para cada clase

Clase Actividades:

1. Agregar id
2. Agregar actividad
3. Agregar fecha
4. Agregar hora
5. Agregar Descripción

Clase Voluntario:

1. Agregar id

Clase Persona:

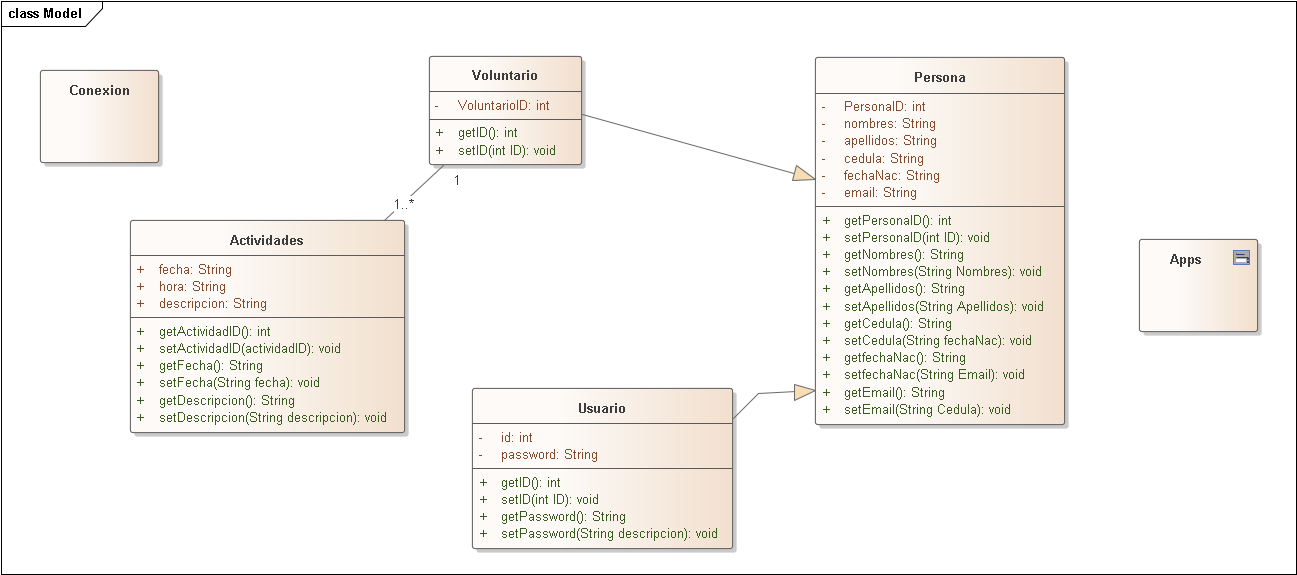
1. Agregar nombres
2. Agregar apellidos
3. Agregar cedula
4. Agregar fechaNac
5. Agregar email

Clase Usuario:

1. Añadir Id
2. Añadir Password
3. Diagrama de clases con operaciones

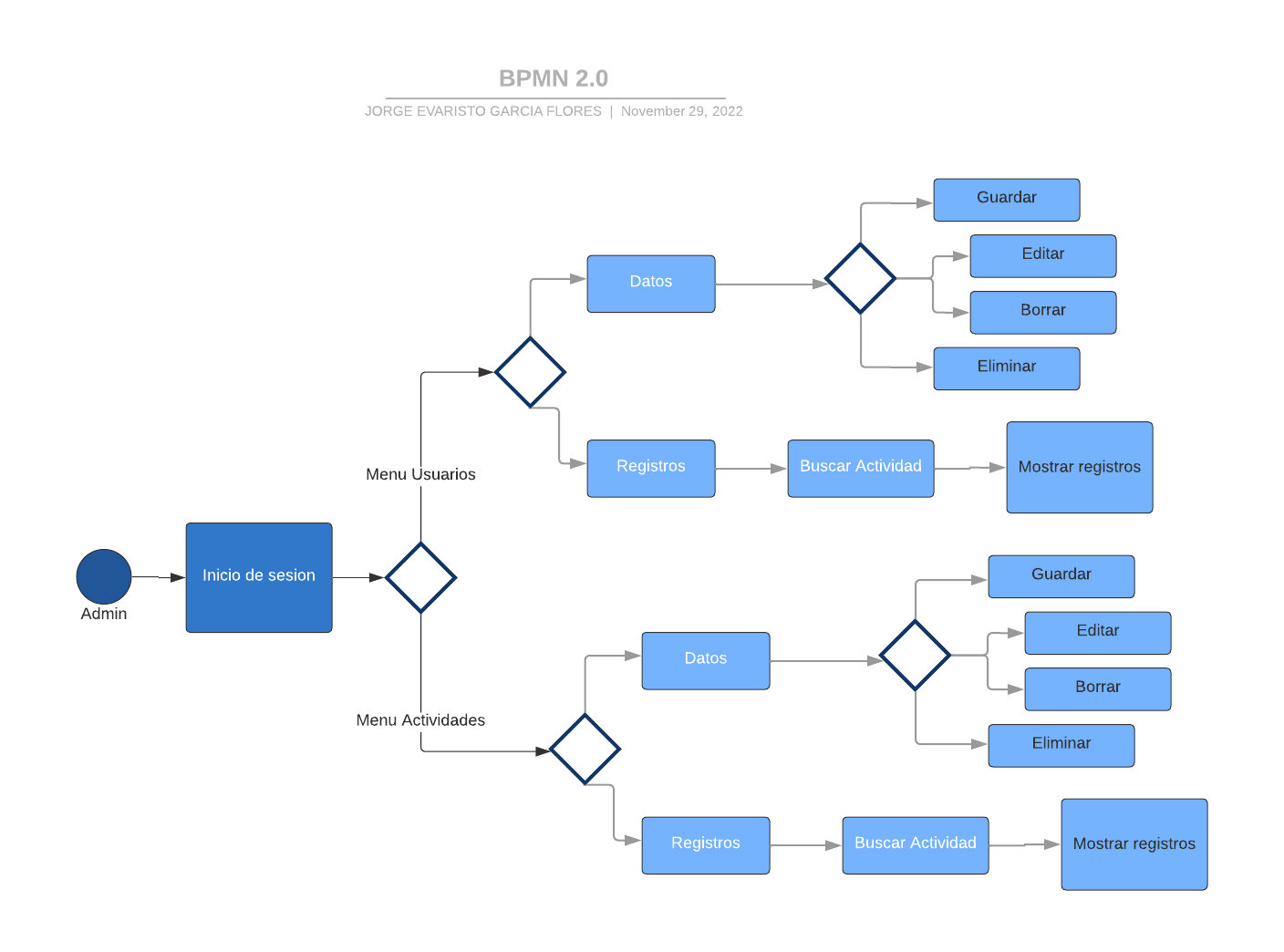
**Identificar y modelar los parámetros de operación**

Diagrama de clases y operaciones con parámetros

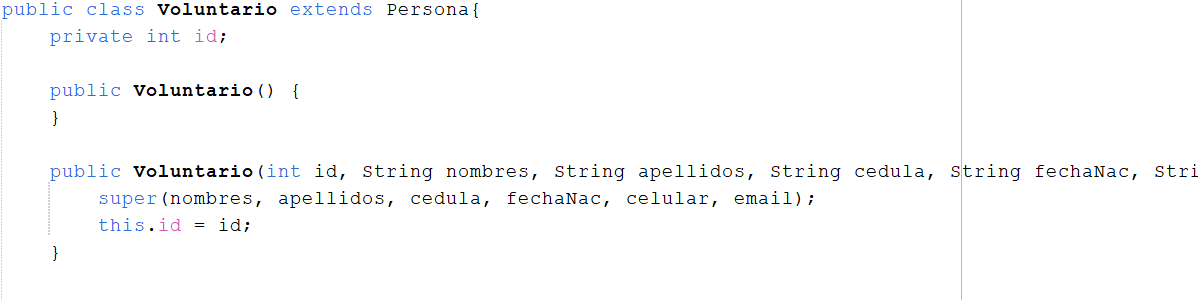


**Identificación de las colaboraciones entre los objetos**

Colaboraciones entre los objetos del sistema



***La clase voluntario es herencia de la clase Persona***



***La clase Usuario es herencia de la clase Persona***

