Se agregaron varios atributos a diferentes tablas.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Se agrega el atributo idSede a la tabla Administrador, pues hace más sentido que haya un administrador por sede, que un administrador general, por cierto, además de que el administrador registre vigilantes, ¿debería registrar Puertas y Areas? O investigamos sus nombres y las insertamos nosotros

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Alumno cambio su nombre a Miembro y se agregó el atributo "Tipo Miembro" para incluir a diferentes tipos de actores que forman parte de la universidad, no sólo alumnos, p/e profesores, trabajadores, etc.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Se agregaron las entidades "Sede", "Area" y "Puerta" que ayudarán a aproximar la ubicación en las tablas “Administrador” (por medio de idSede), “Visitante” (por medio de idAsignacion), "Movimiento" (por medio de idAsignacion), "Incidente" (por medio de idArea), “Vigilante\_Puerta” (por medio de idPuerta) y "Miembro" (por medio de idArea).

A la hora de la inserción de datos por el usuario en las tablas “Incidente”, “Vigilante\_Puerta” y “Miembro”, así como por los administradores de sede (si se llegara a implementar mi propuesta del segundo párrafo) en las tablas "Sede", "Area" y "Puerta" serán elementos predefinidos (previamente insertados en la DB) y se mandarán a traer generando un botón dinámico.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Para los atributos que almacenen tipos (alternativas para un mismo dato) serán botones select, ¿cierto? Hay que analizar y ampliar con ideas las posibles opciones, pero temporalmente podrían ser:

“tipoPuerta” en la entidad “Puerta”:

* Entrada
* Salida
* Entrada y Salida

“tipoIncidente” en la entidad “Incidente”:

* Infracción
* Accidente
* Robo
* Daño

“tipoMiembro” en la entidad “Miembro”:

* Alumno
* Profesor
* Trabajador

“tipoRodada” en la entidad “Bicicleta”:

* 12 pulgadas
* …
* …
* 29 pulgadas

“tipoEstilo” en la entidad “Motocicleta”:

* Scooter
* Cuatrimoto?
* Chopper
* Sport
* Cross
* Touring
* Uber eats :V

“tipoMotor” en la entidad “Scooter”:

* Sin motor
* Combustión
* Eléctrico
* Hibrido

Nota: Para “userType” en la tabla Usuario y “vhType” en la tabla Vehículo no aplica pues son especializaciones ya establecidas y se asignaran por el sistema.

Nota2: También podrían almacenarse estos tipos en entidades independientes en la DB y mandarse a traer como botones dinámicos (como he explicado antes), pero siento que es complicarse la vida, aunque se tendría la ventaja de poder agregar más opciones sin tener que modificar el código, ¿Qué sería mejor?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Con respecto a las motocicletas, ¿Será necesario generar un vhCode? Siendo que la matrícula de circulación podría funcionar ya como un identificador y ahorrarnos trabajo a la hora de asignar los códigos, de momento sólo se agrega el atributo “placa” a la tabla “Motocicleta”. PD: Las placas se componen de 7 caracteres, ósea funcionarían como código porque caben perfectamente en la PK.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Olvidé como se “manejan” los visitantes :’v

**Ídea 1 (la original creo):** La primera visita siempre sería registrada pero los visitantes podrían ser invitados a también dar su correo y que se les envíe un código de “visitante frecuente”, en la próxima visita presentarían el identificador y se les buscaría en la DB

Desventajas: La idea necesita el envío de correos automatizados, además de un botón que después de consultar al visitante posibilite actualizar el vehículo y/o el motivo de la visita, también una nueva tabla donde de forma similar a la tabla estacionamiento, se crearía una copia del vehículo utilizado en la visita y los motivos de esta.

**Ídea 2:** Los visitantes siempre se registran, así se puede almacenar un historial de visitas y no es necesaria una identificación mediante códigos de vehículo, ni de ningún tipo en realidad.

Desventajas: Pudiera ser tedioso para el personal de vigilancia registrar cada visita, especialmente a los repartidores, visitantes frecuentes, etc. También, debido a “la naturaleza” de la DB (herencia en las especializaciones), los visitantes son tratados como usuarios, es decir, cada visita crearía un usuario nuevo que no se va a “reutilizar”.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Se crea la tabla “Estacionamiento” es una entidad donde se almacena una copia temporal de registros de la entidad “Movimiento”, dichos registros se guardarán únicamente cuando se trate de un movimiento de entrada, al salir, dicho registro identificado en uno a uno con el atributo idMovimiento será eliminado (eliminado de la tabla Estacionamiento, no se elimina el registro en la tabla Movimiento), esto con la única finalidad de llevar un conteo de vehículos en las diferentes sedes, así como de los datos del vehículo y sus propietarios.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Los 10 caracteres de las llaves (Del 0-9 y repitiendo números) permiten tener 10^10= 10,000,000,000 combinaciones.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Se crea la tabla intermedia “Vigilante\_Puerta” para intentar mantener la integridad de los datos con respecto a ¿qué vigilante autoriza un determinado movimiento? Pues tendría sentido que los vigilantes no permanezcan en una sola puerta todo el tiempo, además de la probable existencia de varios turnos al día. Para ello, necesitamos que los vigilantes cooperen cerrando sus sesiones al terminar su turno, pero siento que no lo van a hacer, esto podría ser automatizado añadiendo los atributos “inicioTurno” y “finTurno” en la tabla “Vigilante”. Los añado temporalmente pues no estoy 100% seguro de si es una buena idea, aiuda.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Sugerencia: El atributo booleano en la tabla miembros indica sí se ha verificado la pertenencia del miembro a la universidad mediante la confirmación de un código enviado al correo institucional, podríamos limitar el registro de vehículos o poder recibir prestamos hasta que este atributo sea true (1) pero primero tendríamos que implementar el envío de mails y verificaciones, pero aún no tengo idea de cómo se hace.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Se elimina el campo de entrada “código de vehículos” en algunos formularios de los canvas (p/e registro de motocicletas, bicicletas y scooters de miembros, también en registro de visitantes), porque los códigos de vehículo son generados por el sistema ya que deben conservar una nomenclatura especifica.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Duda: ¿Será prudente un atributo foto en la tabla incidente para guardar evidencia?, de momento lo agrego al modelo y a los canvas y si no era necesario se elimina.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

En los canvas, se agrega el campo “Finalizar préstamo” para seleccionar la fecha del fin del préstamo, en la aplicación deberíamos programar que no se pueda seleccionar un rango muy largo de tiempo.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Duda: ¿Los vigilantes tienen ya algún tipo de matrícula o identificador? Si es así, sería útil que los registraran y se usara como PK.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

En el servidor, las fotografías deberán estar bien organizadas en carpetas y se deberá programar una función que nombre el archivo de imagen exactamente que las PK de sus tablas correspondientes, es decir Miembro, Vehículo e Incidente.

En los atributos “foto” de las tablas Miembro, Vehículo e Incidente se guardará la dirección (path) de la fotografía alojada en el servidor.