

# Diseño de Sistemas

SeguriPoint – Segundo Parcial 2024

# SeguriPoint - Segundo parcial 2024

## ¡IMPORTANTE!

***El presente documento representa solamente una serie de consideraciones generales del parcial, no es una propuesta de solución, ya que el mismo carece de modelos y justificaciones, fundamentales para la asignatura.***

# SeguriPoint - Segundo parcial 2024

## Contexto general

*Una empresa de seguridad ha solicitado el diseño y desarrollo de un Sistema que permita, por un lado, monitorear a sus vigiladores (empleados) y, por el otro, que los clientes puedan personalizar sus preferencias de seguridad.*

*Actualmente, la empresa ofrece los siguientes servicios:*

- 1. Vigilancia presencial.*
- 2. Vigilancia a través de tótems.*
- 3. Monitoreo mediante un Sistema cerrado de cámaras y sensores de movimiento y humo.*

*En esta primera etapa nos focalizaremos en el servicio de Vigilancia a través de Tótems. En el diseño del Sistema deben considerarse tres tipos de usuarios: Administradores de la empresa, Vigiladores y Usuarios finales de los servicios contratados (clientes).*

# Modelado de Datos

A continuación, listaremos las consideraciones más relevantes por cada requerimiento.

**1. Que los vigiladores se identifiquen en sus puestos de trabajo, teniendo especial cuidado de guardar los horarios de inicio y finalización de sus jornadas laborales, así como también sus momentos de descanso.**

- Surgen las entidades “**Vigilador**” y “**Trabajo**” (o “**Sesión**”), cada una con sus respectivos atributos.
- Cada trabajo refiere a una jornada laboral del vigilador y tiene que estar relacionada con el **Tótem** (entidad que aún no aparece en este requerimiento).

# Modelado de Datos

***2. Que los vigiladores puedan interactuar por voz mediante los tótems cada vez que el botón de pánico sea presionado.***

- Es necesario guardar las interacciones que existen con el botón de pánico de un tótem, con fecha y hora exacta, y tiempo de respuesta.
- Lo anterior es necesario para la estadística n°1 (*“el empleado que cuenta con el promedio de respuesta más rápido de botón de pánico”*)

# Modelado de Datos

## ***3. Que los vigiladores puedan activar las distintas alertas cada vez que lo requieran.***

- Surge la entidad “**AlertaGenerada**” relacionada con la entidad “**Trabajo**” (o Sesión), y con la entidad “**Alerta**” (o “PosibleAlerta”).
- Cada Alerta debe estar relacionada con “**Acción**”
- Es necesario guardar la fecha y hora.
- Esta entidad es necesaria para cumplir con la estadística n°2 (“*El empleado que activó la mayor cantidad de alertas dentro de su horario laboral*”)

# Modelado de Datos

## 4. Que los administradores gestionen los tótems instalados en cada uno de los establecimientos.

- Surge la entidad “**Tótem**”, además de “**Establecimiento**” y “**Cliente**”.
- Del Establecimiento:
  - Dirección PRECISA: ubicación (lat/long) y dirección normalizada.
  - Cant. metros cuadrados
  - Demografía de personas: relación con una entidad “**Persona**”. De cada persona:
    - Edad: Fecha de nacimiento (no la edad como número)
    - Género, nivel educativo, estado civil y ocupación: valores normalizados. En todo caso, constraint de CHECK en la base de datos.
    - Cant. de accesos al establecimiento: es necesario guardar, por cada día y semana, la cantidad de personas que ingresaron (histórico).
  - Fotografías: relación contra una entidad “**Fotografía**”, la cual debe tener referencia a un sector.
  - Relación con entidad “**Cliente**”.
  - ¿Costo? (podría pensarse de guardar acá el valor acumulado anual desnormalizado).

# Modelado de Datos

## ***4. Que los administradores gestionen los tótems instalados en cada uno de los establecimientos.***

- Del Tótem:
  - Sector en el que está instalado
  - Relación contra la entidad “**Trabajo**” o “**Sesión**”
  - Relación contra entidad que guarda las interacciones con el botón de pánico.
- Del Cliente:
  - Datos mínimos a considerar.
  - Recordar que necesitamos la estadística “los clientes con mayores valores de facturaciones”.



# Modelado de Datos

**5. Que los administradores definan el costo de cada evento personalizado solicitado por los clientes.**

- Surge la entidad “**Evento**” o “**Alerta personalizada**”
- De cada Evento:
  - Nombre
  - Descripción
  - Flujo de acción
  - Acciones: relación contra entidad “**Acción**”
  - Precio

# Modelado de Datos

## ***6. Que los clientes puedan gestionar alertas y acciones.***

- Relación entre “***Establecimiento***” y “***Alerta***” (o “PosibleAlerta”), considerando que un cliente puede tener varios Establecimientos y que por cada uno configura las alertas que desea.

# Modelado de Datos

## ***7. Que los clientes puedan gestionar sus eventos personalizados.***

- Reafirma requerimiento número 5.

# Modelado de Datos

## ***8. Que los clientes puedan gestionar reclamos técnicos relacionados al uso del servicio.***

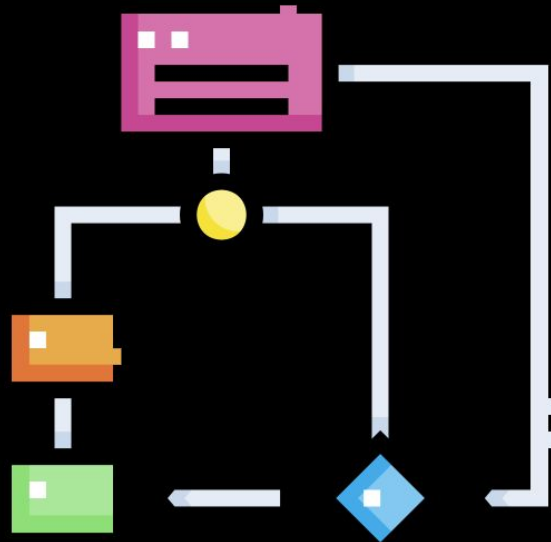
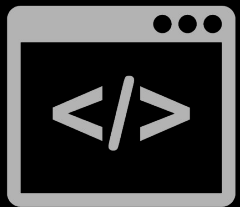
- Surge entidad “***Reclamo***”
- De cada reclamo:
  - Relación contra entidad Tótem
  - Fecha y hora
  - Relación contra entidad “***Tipo de problema***” (normalizado)
  - Persona que reclama (puede ser un cliente, o alguna persona del establecimiento)
  - Descripción
  - Relación contra entidad “***Fotos***”

# Modelado de Datos

## **9. Visualizar todas las estadísticas solicitadas.**

- Empleado más comprometido del mes: el empleado que cuenta con el promedio de respuesta más rápido de botón de pánico.
  - Puede resolverse a partir del Requerimiento 1
- Empleado más atento de la semana: el empleado que activó la mayor cantidad de alertas dentro de su horario laboral
  - Puede resolverse a partir del Requerimiento 3
- Top 10 clientes que más ingresos generan: los clientes con mayores valores de facturaciones.
  - Debe ser posible sumarizar todos los costos de los clientes.
- Top 3 problemas más frecuentes: los 3 problemas más frecuentes que se presentan en los tótems.
  - Puede resolverse a partir del Requerimiento 8
- La alerta más escogida en el último año: la alerta más escogida entre los clientes de entre todas las existentes.
  - Puede resolverse a partir del Requerimiento 6.

# Arquitectura



# Arquitectura

**1. Considerando que la solución se apoya en un Estilo de Microservicios y que se quiere escalar a 3 instancias el Microservicio de Alertas.**

- a) ¿Qué componentes son necesarios que interactúen para soportar las nuevas instancias?**
- b) Genere el Diagrama de Despliegue dónde muestra cómo quedaría el Microservicios escalado.**

- a) Para soportar el escalado deben intervenir las nuevas instancias, el Service Registry y el API Gateway (considerando que el API Gateway hace el balanceo de cargas).
- b) Se deben mostrar 3 instancias del Microservicio Alerta. Es importante que esté presente el API Gateway y el Service Registry.

# Arquitectura

## **2. Considerando el tipo de Sistema que estamos desarrollando.**

- a) ¿Qué tipo de cliente utilizará y por qué? Justifique utilizando 2 atributos de calidad.**
- b) ¿Considera viable una Aplicación Móvil? ¿De qué tipo?**

- a) Se esperaba la utilización de un Cliente Pesado o Client Side Render.
- b) Podría ser propuesta una aplicación híbrida.



# Arquitectura

**3. Considerando que el cliente debe recibir el detalle de su reclamo por mail, posterior a la carga del mismo.**

- a) ¿Cómo haría este proceso?**
- b) Considerando una Arquitectura de Microservicios ¿Qué servicios involucraría?**
- c) ¿Qué mecanismo de integración utilizará entre ambos servicios?**

- A. Proceso Asíncrono
- B. Microservicio de Reclamo y Microservicio de Notificaciones
- C. Pensando en un Productor - Consumidor, se espera una cola de mensajes/trabajo.

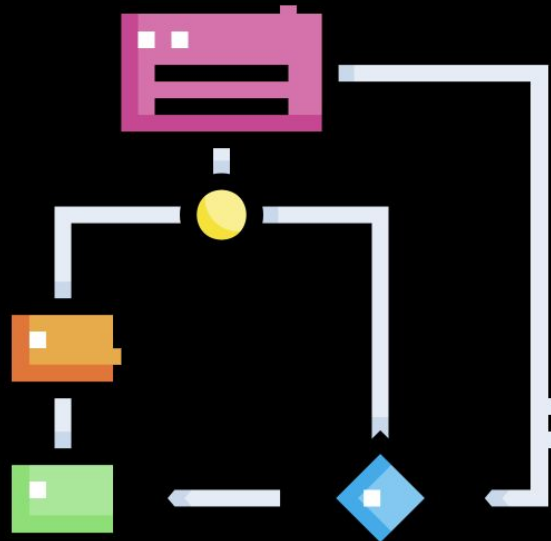
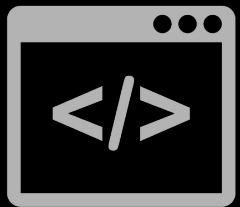
# Arquitectura

**4 Con el objetivo de trabajar en conjunto con los diferentes Ministerios de Seguridad de las jurisdicciones en la que están nuestros productos, los diferentes actores nos solicitaron acceso a las estadísticas. Cada actor requiere un conjunto de datos diferentes por lo que se requiere flexibilidad en el EndPoint. Se requiere minimizar las llamadas a los Endpoint disponibles.**

- a) ¿Qué tipo de integración propondría?**
- b) ¿Qué atributos de calidad consideraría? Defina cada uno.**
- c) ¿Qué cambios debería realizar sobre la Arquitectura del Sistema Actual?**

- A. Se espera una API GraphQL, principalmente apoyado en: "requiere un conjunto de datos diferentes por lo que se requiere flexibilidad en el EndPoint. Se requiere minimizar las llamadas a los Endpoint disponibles."
- B. Se puede proponer performance, se reduce la cantidad de llamadas al Servidor al devolver consultas agregadas, esto también impacta en la escalabilidad. Por otro lado, se puede hablar de flexibilidad y mantenibilidad. Si es importante que no se proponga Usabilidad, ya que no es una interfaz pensada para el Usuario Final.
- C. Se debe agregar como mínimo un componente de Estadísticas que exponga esta nueva Interfaz Saliente. Se debería pensar qué sucede con la lectura de los datos, se puede considerar una desnormalización y una base de datos separada o leer de una réplica de la base de datos.

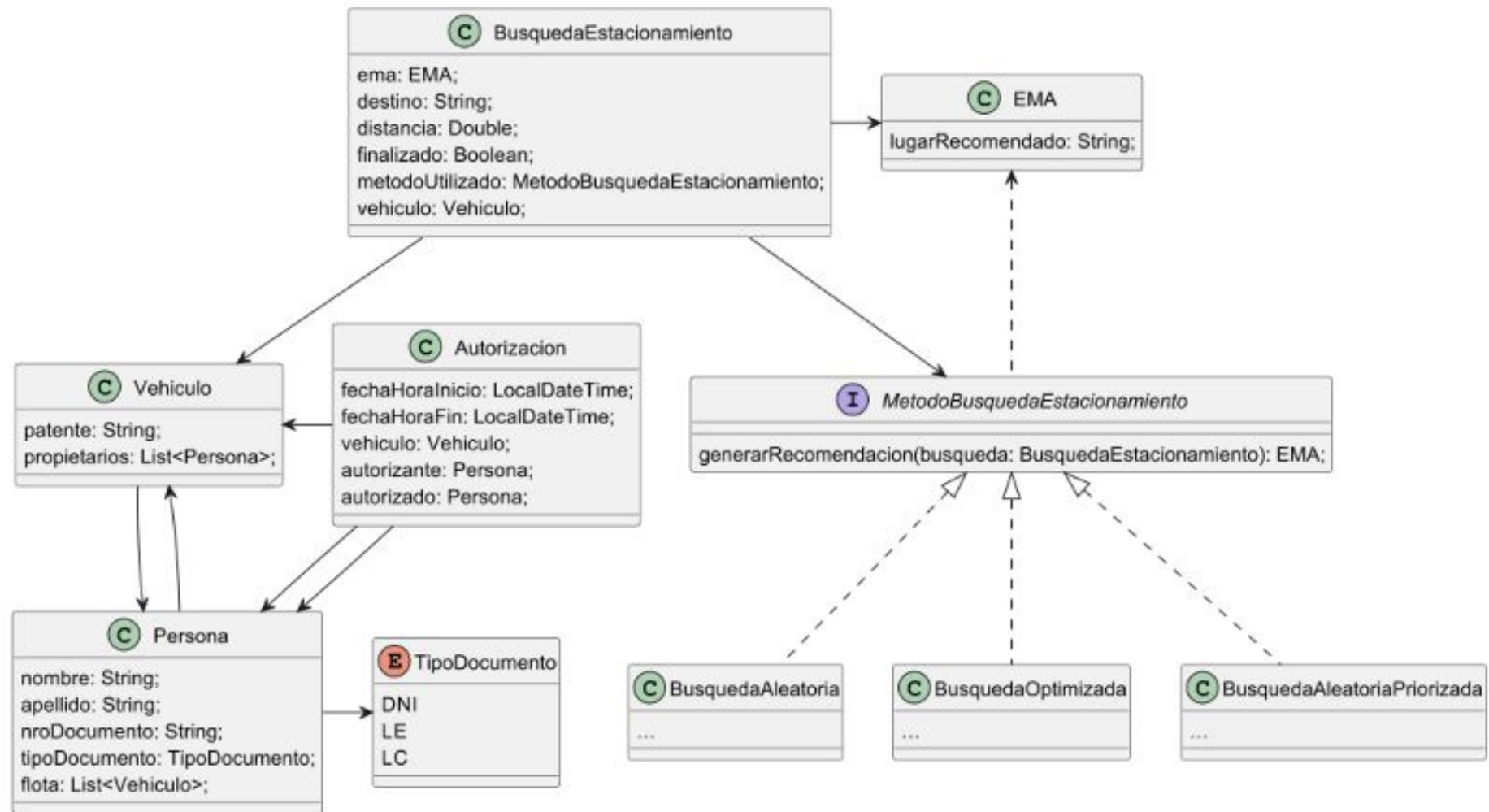
# Persistencia



# Persistencia

Nos han encargado realizar la persistencia de una parte de un Sistema de Búsqueda de Estacionamientos. De los vehículos se conoce su dominio (patente) y sus propietarios. De cada persona se conoce su nombre, apellido, número y tipo de documento. Además, cada persona puede gestionar su flota de vehículos familiares. Por otro lado, una persona puede otorgar autorizaciones para que cualquier otra persona pueda utilizar cualquiera de sus vehículos. La autorización puede ser por un periodo de tiempo en particular o sin límite.

Cada vez que una persona quiera encontrar un estacionamiento deberá dar de alta una Búsqueda de Estacionamiento, la cual contiene el destino, el vehículo que se estacionará, la distancia a la cual el Sistema debe comenzar a realizar la búsqueda y el método utilizado. Los métodos para encontrar un estacionamiento pueden ser: Búsqueda aleatoria, Búsqueda optimizada y Búsqueda aleatoria priorizada. La ejecución del proceso de búsqueda de estacionamiento arroja como resultado un EMA: el Estacionamiento Más Apropiado. Para esta primera iteración solo basta con saber, del EMA, el lugar.



# Persistencia

## Consideraciones

- **Vehículo** debe mapear contra su propia tabla
- **Persona** debe mapear con su propia tabla
  - La relación entre Persona y vehículo se da a través de 2 relaciones many to many, cada una con su propia tabla
    - vehiculo\_propietario
    - vehiculo\_flota
- **Autorización** debe mapear contra su propia tabla
  - Tiene dos relaciones contra la entidad Persona (dos fks)
- **Búsqueda de estacionamiento** debe mapear contra su propia tabla
  - El atributo “Método búsqueda Estacionamiento” es una interfaz stateless y se debe persistir utilizando un **converter**.
  - El EMA puede ser embebido.
- No se deben mapear los enumerados como tablas.

# Gracias

