



**Departamento de Informática**  
Universidad Técnica Federico Santa María



## Requisitos de Software

### Proyecto: Seth Project

Integrantes:

Nombres y Apellidos	Email	ROL USM
Anghelo Carvajal	anghelo.carvajal.14@sansano.usm.cl	201473062-4
Gabriela Gallardo	gabriela.gallardo.13@sansano.usm.cl	20137395-7
Nicolas Lagos	nicolas.lagos.12@sansano.usm.cl	201204213-5

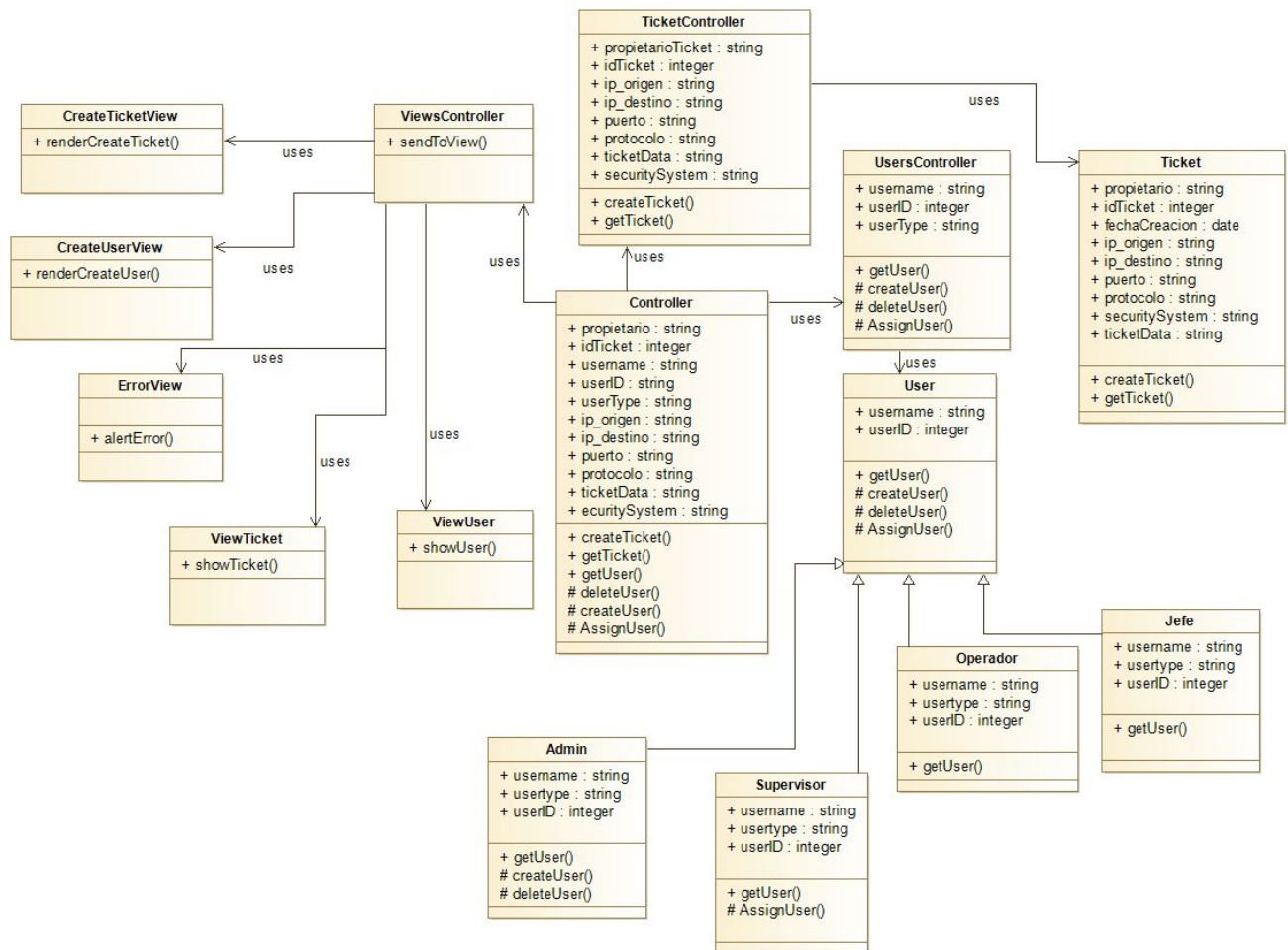
## Desarrollo del Prototipo

Este inciso se encuentra en el repositorio del equipo.

## Selección de Patrones de Diseño

Intención	Patrón de Diseño	Razonamiento
Se desea mostrar en el Diagrama de Clases la desconexión de un usuario después de la media hora.	Observer	La clase usuarios deberá ser capaz de avisarle a la clase controladora de logs del tiempo que lleva sin actividad un determinado usuario, con el fin de que esta pueda cerrar su sesión de forma oportuna en el caso que haya excedido el límite de tiempo especificado por el cliente y además de registrar el tiempo que permaneció en línea el usuario.
Dentro del diagrama de clases se aprecian el modelo separado del controlador y las vistas de cada entidad (por ejemplo usuarios)	Modelo vista controlador	Debido a que la estructura de mvc es un mecanismo fácil de manipular (Esto no implica que sea fácil), y dado a la experiencia del grupo usando este patrón en el paso, las maneras existentes en las que se puede usar y que crea una capa de seguridad adicional por tener la lógica del negocio separada del front end, llegamos a la conclusión que era una buena opción y decidimos usarla.

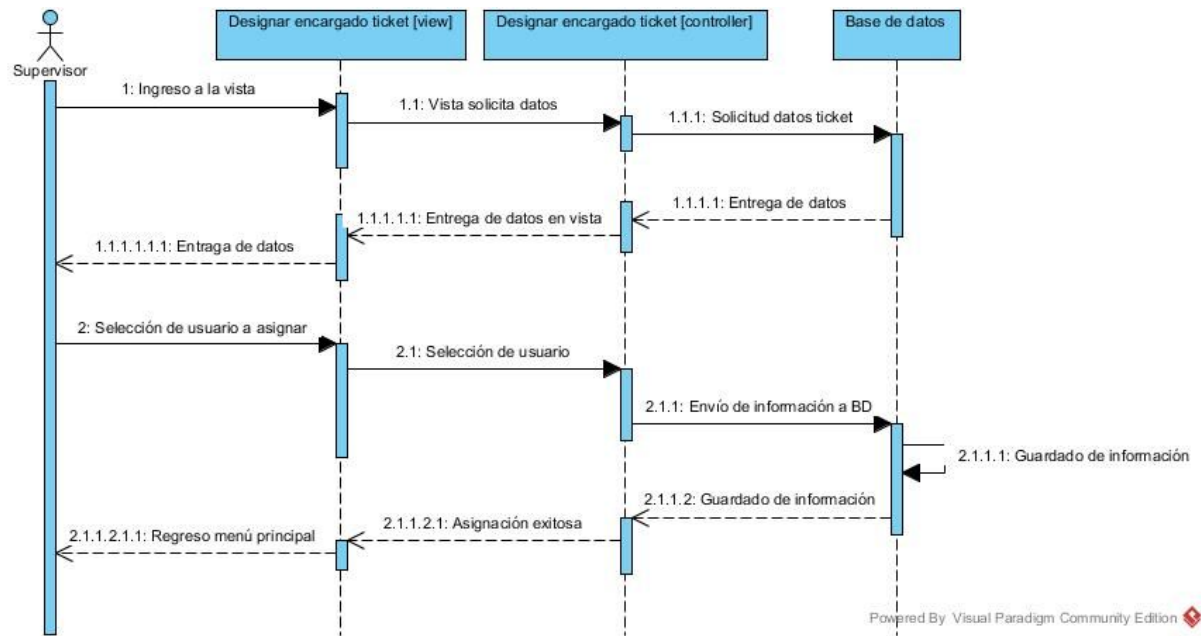
## Creación de Diagrama de Clases



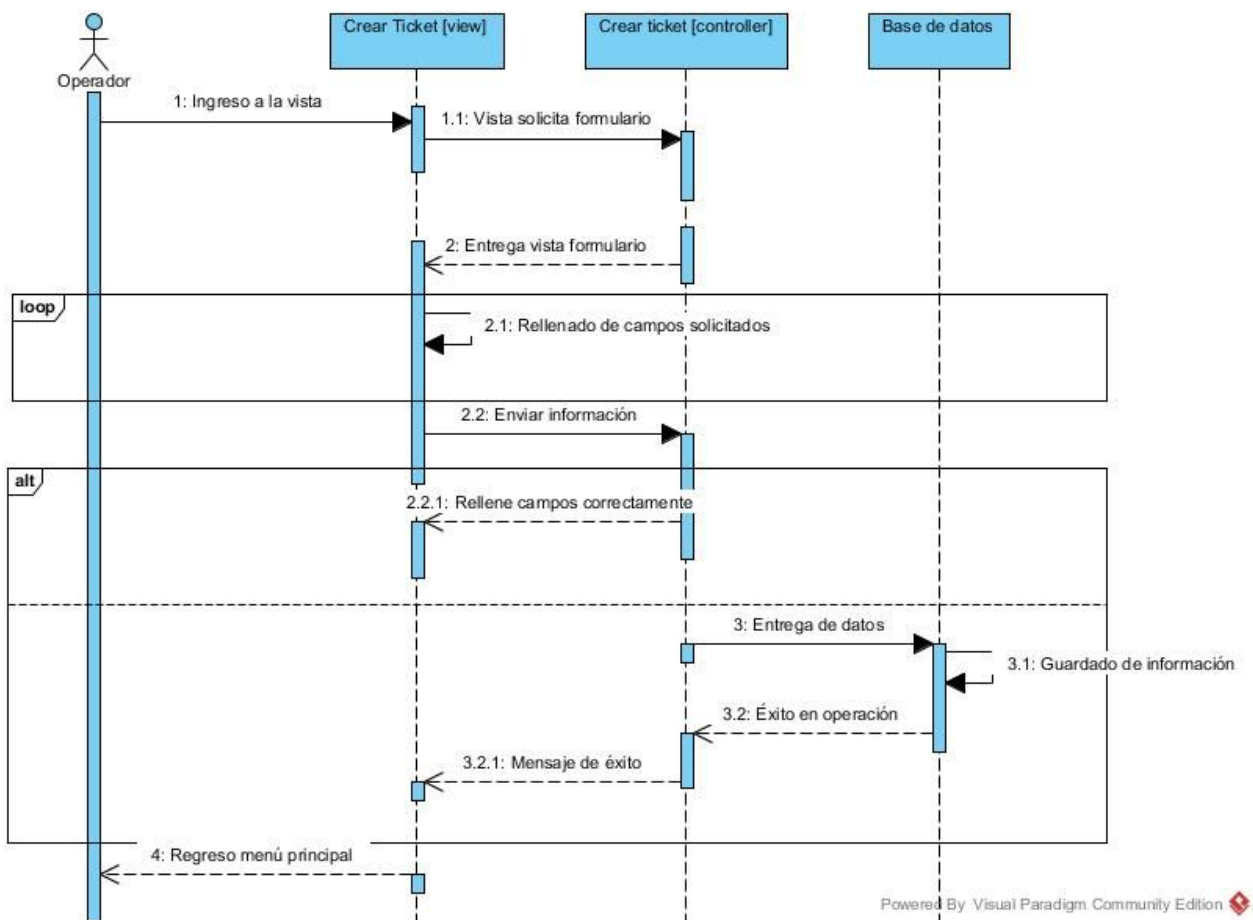
## Diagramas de Secuencia

### Asignación de ticket

Para el diagrama de secuencia de asignar usuario a ticket trabajamos con el supuesto de que se muestran todos los tickets sin asignar.



### Crear nuevo ticket



## Análisis de Trade-off

Como equipo solo se nos han señalado pequeños detalles dentro de los requisitos funcionales, más allá de agregar nuevas funcionalidades. Es por esto que dentro de este inciso señalaremos una modificación a un requisito más que una funcionalidad nueva.

Dentro de el requisito **Visualizar información** se nos ha señalado dentro de lo indicado (El sistema debe proveer una vista que muestre tickets “abiertos” y otra “aplazados”.) **verificar los logs del sistema**.

A continuación señalamos los pasos del enfoque QOCT:

1. [Question] *¿Qué cambios se deben realizar en el Sistema para agregar el componente de verificación de logs del sistema?*
2. [Options]
  - O1: Generar distintas vistas para poder visualizar los diferentes logs.
  - O2: Agregar una vista especial para poder visualizar los logs, se aplicarán filtros para visualizarlos de ser necesario.
3. [Criterios]
  - C1: Confiabilidad
  - C2: Mantenibilidad
  - C3: Portabilidad
  - C4: Escalabilidad
  - C5: Rendimiento

4. [Trade-offs]

Criterio\Opciones	O1	O2
C1	+	+
C2	-	+
C3	+	++
C4	-	+
C5	--	-

El análisis obtenido como equipo respecto a las opciones señaladas anteriormente es

el siguiente:

- *La opción 1 (O1) tiene buen impacto respecto a confiabilidad. Sin embargo, impacta negativamente aspectos de mantenibilidad, escalabilidad y rendimiento.*
- *La opción 2 (O2) Si bien se ve afectado el rendimiento al mostrar todos los logs en una sola vista, afecta de buena manera la confiabilidad, mantenibilidad y escalabilidad.*

Como equipo concluimos que la opción 2 afecta de mejor manera el sistema, ya que si bien se deben cargar muchos datos esto se realizará una vez y permitirá al usuario filtrar la información según la necesite. Esto afecta positivamente al sistema ya que cumple con la interfaz intuitiva y fácil de manejar que necesita el cliente. Además al reducir la cantidad de vistas se está evitando generar una gran cantidad de vistas innecesarias para acceder al mismo tipo de información