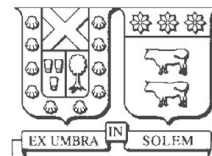




**Departamento de Informática**  
Universidad Técnica Federico Santa María



## Requisitos de Software

### Proyecto: Sistema de Tickets

Integrantes:

| Nombres y Apellidos | Email                             | ROL USM     |
|---------------------|-----------------------------------|-------------|
| Bryan Rosales       | bryan.rosales.13@sansano.usm.cl   | 201330040-5 |
| Daryl Hernandez     | daryl.hernandez.14@sansano.usm.cl | 201430001-k |
| Oscar Lizama        | oscar.lizama.12@gmail.com         | 201103043-5 |

## Contexto del Proyecto

### Objetivo del proyecto

Crear Software local que busca gestionar la administración de procesos a través de sistema de ticket de manera segura por interfaz web de la Armada de Chile.

### Resumen del Proyecto

El proyecto “ Sistema de Tickets” es un sistema que tiene como objetivo gestionar la administración de procesos a través de sistema de ticket por interfaz web.

El futuro sistema es independiente de otros, únicamente se debe considerar que correrá sobre una plataforma Web siendo su hardware un servidor dedicado, el cual tendrá que contar con los estándares del CLIENTE para su puesta en marcha (UPS, Sistemas de enfriamiento en sala de servidores).

Para poder desarrollar este producto. Como grupo grupo decidimos utilizar el lenguaje de programación ‘ Python’ para su creación, y para ser más específico, se usará el Framework ‘Django’ para la creación de un servicio estable y moderno que proteja al sistema de problemas de seguridad y nos permita tener una estética deseable para el CLIENTE. No obstante, si tuviésemos algún problema de gusto o compatibilidad con los gustos del parroquiano, podríamos cambiar a otro framework que nos permita tener un sistema Ad Hoc para él - Por ejemplo, Tornado-.

El enfoque que desarrollaremos será un modelo MVC que nos permita controlar de manera efectiva el desarrollo del Software. El cual complementaremos con un proceso iterativo incremental para poder ir realizando *test* en el hardware del cliente.

Si nuestro objetivo es ver si estamos bien encaminados en el desarrollo o cumplimos nuestras metas elegidas en un principio. Nuestro sistema debe poder realizar una correcta administración - CRUD- de un ticket del sistema, el cual debe poder ser testeado dentro de una máquina con las características entregadas por el cliente y así poder verificar nuestro éxito.

## Modelo de Dominio (Inicial)

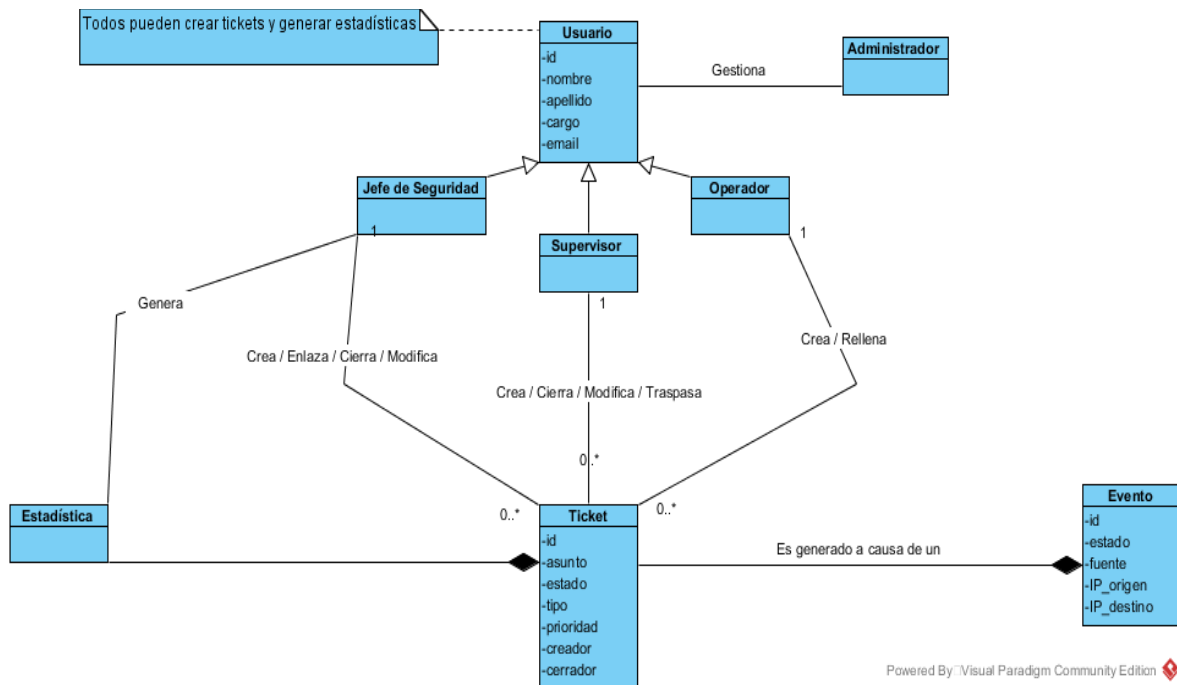


Figura 1: Modelo de dominio

Tabla 1: Entidades del Dominio

| Entidad           | Descripción (1 línea)  |
|-------------------|--|
| Usuario           | Abstracción de las características básicas de un Usuario.        |
| Jefe de Seguridad | Usuario encargado de la gestión de procesos del software.        |
| Supervisor        | Usuario encargado de la gestión de Tickets.                      |
| Operador          | Usuario encargado de informar eventos a través de Tickets.       |
| Administrador     | Usuario encargado de la creación o modificación de Usuarios.     |
| Ticket            | Elemento mediante el cual se registra o informa un Evento.       |
| Evento            | Suceso digno de registrar en un ticket.                          |
| Estadística       | Información obtenida tras el análisis de los Ticket registrados. |

## Actores y tareas clave

| Actor             | Descripción  |
|-------------------|--|
| Operador          | Usuario encargado de informar eventos a través de Tickets. |
| Supervisor        | Usuario encargado de la gestión de Tickets.                |
| Jefe de Seguridad | Usuario encargado de la gestión de procesos del software.  |

| Tarea clave                       | Descripción  |
|-----------------------------------|--|
| Gestión de Tickets                | El sistema proporcionará los medios para crear, modificar y eliminar tickets.                            |
| Exportar Estadísticas             | El sistema permitirá la generación de estadísticas a partir de la información recabada por los tickets.  |
| Gestión de Usuarios               | El sistema proporcionará los medios para crear, modificar y eliminar usuarios.                           |
| Muestra global estados de Tickets | El sistema dispondrá de una vista que permitirá visualizar todos los tickets, incluyendo los eliminados. |

## Requisitos clave funcionales y extra-funcionales

| Req. funcional                              | Descripción y medición   |
|---|--|
| CRUD de Usuarios.                           | Administración del software debe poder crear, modificar y eliminar usuarios de los distintos tipos.                                    |
| Registro de las sesiones realizadas.        | Toda sesión realizada por los usuarios de con los detalles de ella (fecha, hora, duración de la sesión).                               |
| Crear Ticket.                               | Los 3 tipos de usuario debe poder crear tickets que registren algún evento. En el ticket quedará registrado autor y fecha de creación. |
| Designar Encargado de Ticket.               | Usuario Jefe o Supervisor deben asignar los tickets a un Usuario Operador, no será necesariamente el autor del ticket.                 |
| Modificar Ticket.                           | Solo los Usuario Jefe y Supervisor pueden modificar los tickets. El Usuario Operador puede elevar la solicitud de modificación.        |
| Informar Modificación Ticket.               | Si algún Usuario Supervisor válida la modificación de algún ticket, se avisara Jefes, quienes podrán enmendar la modificación.         |
| Aplazar Ticket.                             | Usuarios Jefes y Supervisores podrán designar tickets no abiertos hasta alguna fecha límite. Operador puede elevar solicitud.          |
| Eliminar Ticket.                            | Usuarios Jefes y Supervisores pueden eliminar tickets. El usuario Operador no tiene permitida esta acción, si la solicitud.            |
| Cerrar Ticket.                              | Usuarios Jefes y Supervisores pueden Cerrar tickets. El usuario Operador no tiene permitida esta acción, si la solicitud.              |
| Supervisor cambia responsable de ticket.    | Usuarios jefes y Supervisores pueden cambiar al responsable de algún ticket (Operadores).  |
| Vincular Tickets.                           | Los Usuarios pueden identificar vínculos entre tickets y dejarlos registrados en la "ficha" del ticket.                                |
| Consultar Información y Extracción de ella. | El sistema ofrecerá a los Usuarios información general acerca de los eventos y procesos acaecidos/registrados por el software.         |
| Exportar informes estadísticos y gráficos.  | Los Usuarios pueden consultar la información general registrada por los tickets y extraerla en algún archivo.                          |
| Ingreso vía Web. (Intranet Institucional)   | El sistema y la interacción en él debe ser a través de una página Web únicamente accesible en la red interna de la Armada.             |

| Req. extra-funcional                      | Descripción y medición  |
|---|---|
| Respaldo de la Base de Datos.             | Para tener mayor seguridad el respaldo debe quedar en un espacio físico distinto del que utiliza el Sistema.              |
| Operar en Mozilla Firefox y Google Chrome | El Sistema debe operar en las últimas versiones de Chrome y Mozilla.  |
| Soportar conexiones simultáneas.          | Sistema debe soportar un mínimo de 20 usuarios conectados simultáneamente   |
| Desconectar inactivos.                    | Cerrar sesión de Usuarios después de 30 min de inactividad.   |
| Requisitos de comunicación.               | El protocolo de comunicación entre cliente/servidor será HTTPS por puerto 443. Solo se permitirá uso de IPv4.             |
| Aislación de internet.                    | Sistema no debe tener acceso a internet ni efectuar ninguna conexión hacia el exterior. Todo se realizará en la intranet. |

## Casos de Uso iniciales

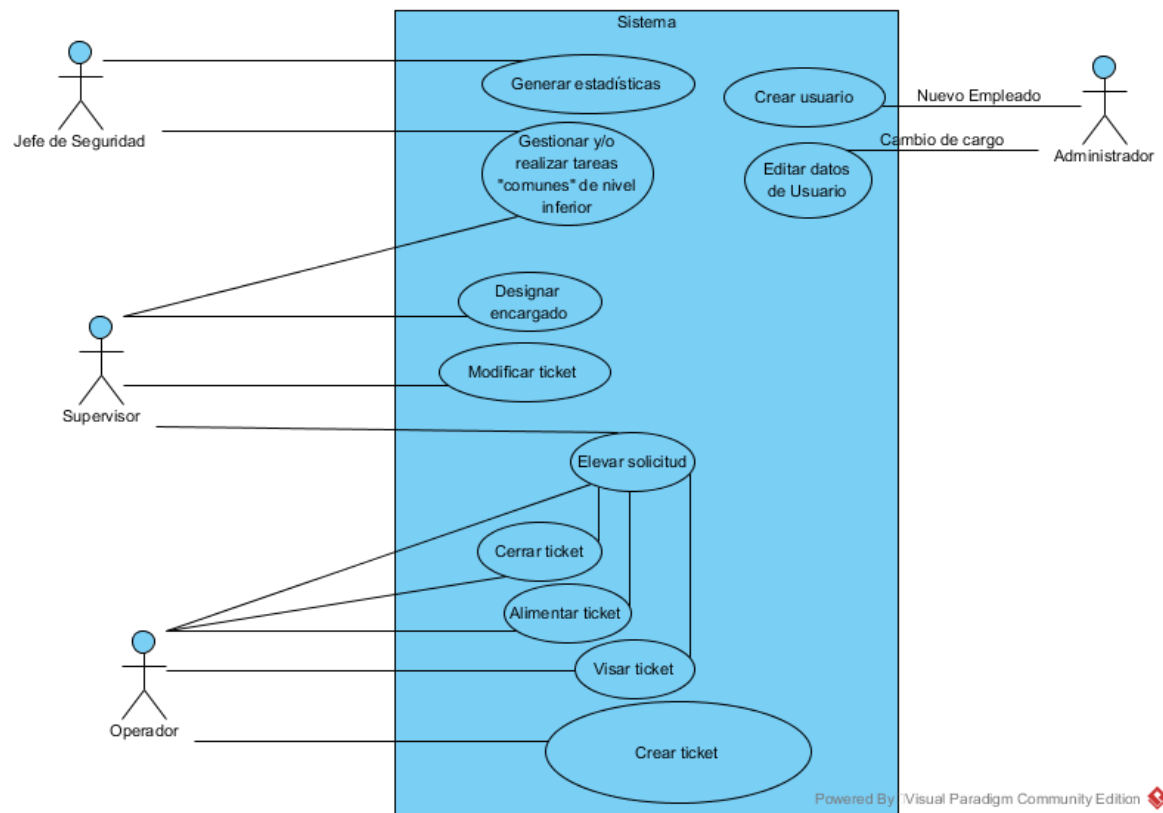


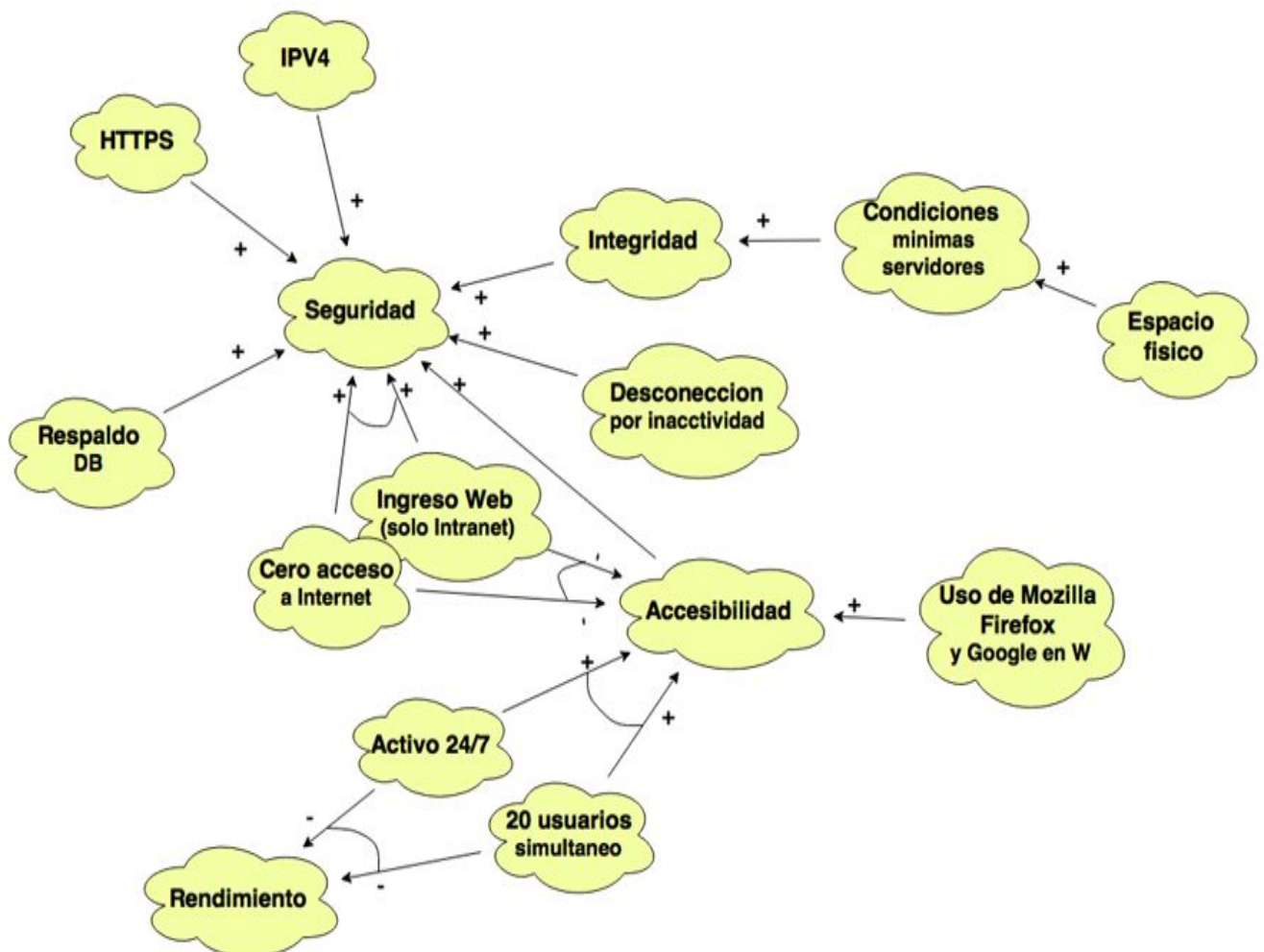
Figura 2: Diagrama de casos de uso

|                   |  |
|-------------------|--|
| Nombre            | Crear ticket   |
| Descripción       | El usuario detecta un evento y procede a la creación de un ticket asociado.  |
| Pre-condiciones   | Un evento ha ocurrido, la información a ingresar es válida / lógica  |
| Post-condiciones  | Un nuevo ticket es creado.   |
| Flujo principal   | Usuario ingresa a la vista de creación de tickets, rellena los campos asociados, y presiona el botón de guardar.                     |
| Flujo alternativo | No existe un campo necesario, usuario presiona botón crear campo, le da un nombre, lo rellena, y procede con la creación del ticket. |

|                   |   |
|-------------------|---|
| Nombre            | Cerrar ticket   |
| Descripción       | El incidente descrito por el ticket ya ha sido atendido, por lo que no hay necesidad de mantener el ticket abierto. |
| Pre-condiciones   | Usuario tiene privilegios necesarios.   |
| Post-condiciones  | El ticket es cerrado  |
| Flujo principal   | Usuario ingresa a la vista de cierre tickets, selecciona el ticket a cerrar, confirma.                              |
| Flujo alternativo | El ticket ya ha sido cerrado, se redirige a la vista de cerrado de tickets.   |

|                   |   |
|-------------------|---|
| Nombre            | Generar estadísticas  |
| Descripción       | Se desea obtener una visión general de los eventos ocurridos en un cierto periodo.  |
| Pre-condiciones   | Existen tickets en la base de datos.  |
| Post-condiciones  | Se generan / muestran las estadísticas.   |
| Flujo principal   | Usuario ingresa a la vista de estadísticas, selecciona el periodo de tiempo a evaluar, selecciona opciones de filtrado                    |
| Flujo alternativo | No existen tickets con las características del filtrado, se notifica al usuario de la situación y se redirige a la vista de estadísticas. |

## Identificación de softgoals





## Creación del árbol de utilidad de la solución

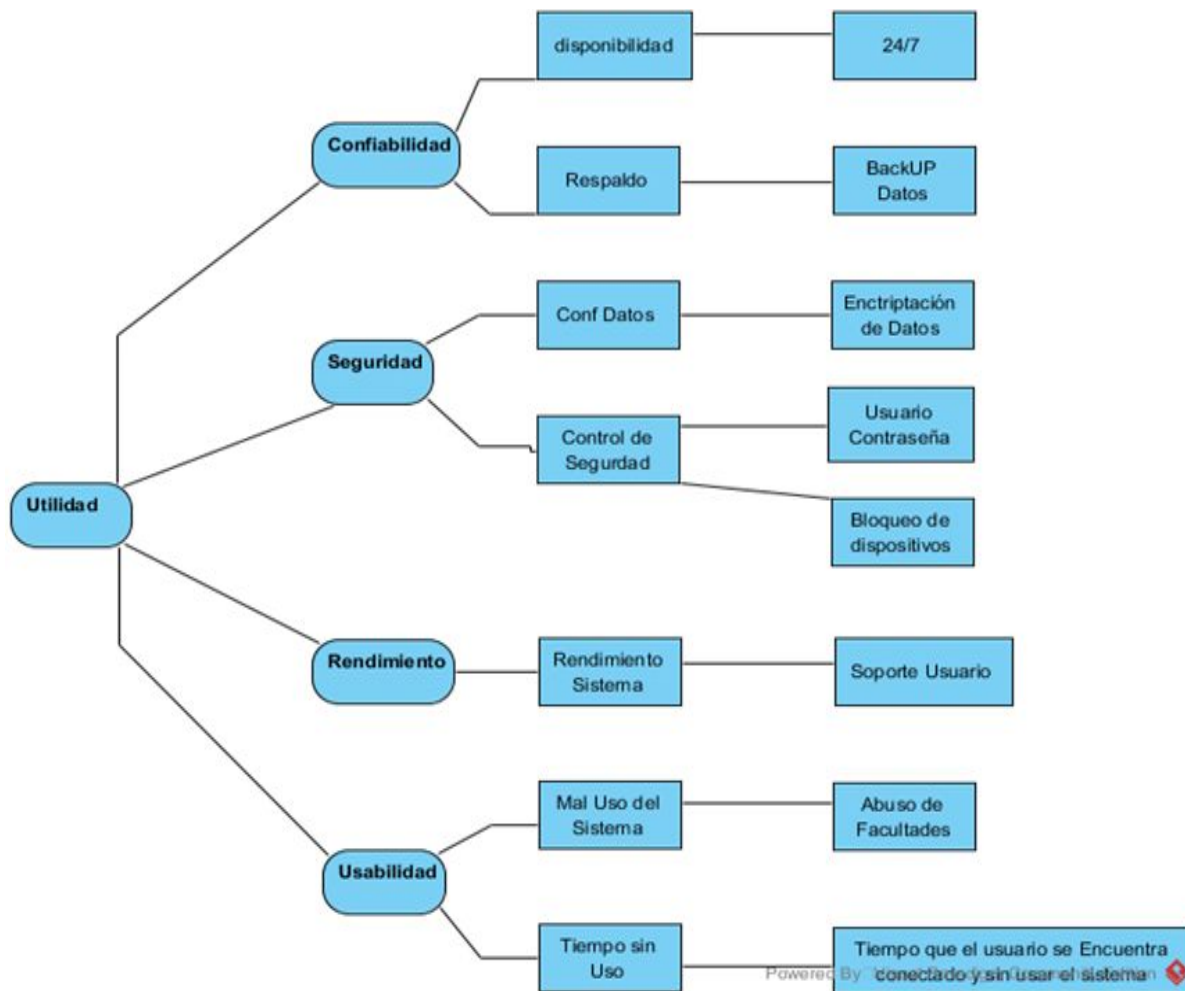


Figura 4: árbol de utilidad

## Identificación preliminar de riesgos para el proyecto

1. **Aparición de nueva amenaza computacional:** Como la seguridad es menester para el sistema necesitado por el cliente la aparición de un malware - por ejemplo Ransomware - pueden ser un problema para el sistema. Más aún, si se trata de algún 'virus' emergente.  
Estár a la vanguardia y realizar una investigación actualizada de las amenazas de sistemas, sería una alternativa para poder enfrentar el problema.
2. **Incompatibilidad Técnica del servidor con plataforma de Desarrollo. (recientemente aclarada por el cliente):** Siempre las condiciones de la máquina del cliente serán un problema o misterio para el desarrollador. Por lo tanto la compatibilidad o detalles de hardware pueden generar problemas técnicos para el desarrollo.  
El estar en constante contacto con el cliente y poder realizar una prueba - o beta- previo al sistema sería lo ideal para comprobar la factibilidad de la propuesta.
3. **Paralización Estudiantil:** Si Bien esta no es una amenaza del punto de vista del software y en un caso de desarrollo normal del mismo. Debido a l marco estudiantil y pedagógico que enmarca el proyecto, una paralización estudiantil, podría estancar o alentar el desarrollo de la plataforma WEB.  
Si bien este es un problema que se escapa de las manos de las manos de todas las partes. Una solución sería poder seguir en contacto y trabajar a pesar de las movilizaciones o acordar un plan de contingencia con el mismo.