

Proyecto TIS 1:

# Especificación de Requisitos

Asignatura:	Taller de Ingeniería de Software 1.
Código:	IN1080C.
Profesor:	Lorenzo Paredes.
Integrantes:	Felipe Arrans. Christopher Moyano. Damián Pantoja. José Rivas.
Fecha:	01/12/2023.

# Funcionalidad principal

- 1.- Enviar ticket
- 2.- Responder ticket
- 3a.- Trazabilidad de ticket
- 3b.- Seguimiento de ticket
- 4.- Cambiar estado de ticket
- 5.- Cambiar visibilidad de ticket
- 6.- Cerrar ticket

# Listado de funciones

- 7.- Filtrar lenguaje de ticket.
- 8.- Visualizar gráficos estadísticos de los tickets según tipo de ingreso.
- 9.- Login de usuario.
- 10.- Suscripción a actualizaciones de ticket.
- 11.- Visualizar tickets públicos anteriores.
- 12.- Calificar funcionamiento del sistema.
- 13.- Visualizar en el mapa la cantidad de felicitaciones, reclamos, sugerencias.
- 14.- Analizar estadísticas de calificación del sistema.
- 15.- Exportar datos de tickets a un archivo.
- 16.- Cambiar a modo oscuro.
- 17.- Derivación de ticket a otra municipalidad o departamento.
- 18.- Muestra de calendarios con próximos proyectos.
- 19.- Agendar visita presencial a la municipalidad.
- 20.- Asignar roles con permisos.

# Mantenedores

## **R21. Mantenedor de municipalidades.**

- R21.1 Ingresar municipalidad.
- R21.2 Visualizar municipalidades.
- R21.3 Actualizar municipalidad.
- R21.4 Eliminar municipalidad.

## **R22. Mantenedor de departamentos.**

- R22.1 Ingresar departamento.
- R22.2 Visualizar departamentos.
- R22.3 Actualizar departamento.
- R22.4 Eliminar departamento.

## **R23. Mantenedor de regiones.**

- R23.1 Ingresar región.
- R23.2 Visualizar regiones.
- R23.3 Actualizar región.
- R23.4 Eliminar región.

**R24. Mantenedor de comunas.**

- R24.1 Ingresar comuna.
- R24.2 Visualizar comunas.
- R24.3 Actualizar comuna.
- R24.4 Eliminar comuna.

**R25. Mantenedor de roles.**

- R25.1 Ingresar rol.
- R25.2 Visualizar roles.
- R25.3 Actualizar rol.
- R25.4 Eliminar rol.

**R26. Mantenedor de proyectos.**

- R26.1 Ingresar proyecto.
- R26.2 Visualizar proyectos.
- R26.3 Actualizar proyecto.
- R26.4 Eliminar proyecto.

**R27. Mantenedor de estados de ticket.**

- R27.1 Ingresar estado de ticket.
- R27.2 Visualizar estados de ticket.
- R27.3 Actualizar estado de ticket.
- R27.4 Eliminar estado de ticket

**R28. Mantenedor de palabras ofensivas.**

- R28.1 Ingresar palabra ofensiva.
- R28.2 Visualizar palabras ofensivas.
- R28.3 Actualizar palabra ofensiva.
- R28.4 Eliminar palabra ofensiva.

# Funcionalidad principal

**R1.** Enviar ticket: permite a un usuario enviar ticket al sistema.

**Entradas:** rut\_usuario, tipo\_solicitud, asunto\_ticket, detalles\_solicitud, cod\_departamento, direccion.

Requisitos de la función:

**R1.1** Para el rut\_usuario se utilizará el RUT de la persona que esté enviando la solicitud, con números, sin puntos, sin dígito verificador ni guión.

**R1.2** Para el tipo\_solicitud solo puede ser un valor del dominio (felicitacion, reclamo, sugerencia).

**R1.3** asunto\_ticket solo puede aceptar letras, letras con tilde y números.

**R1.4** detalles\_solicitud solo puede aceptar letras, letras con tilde y números.

**R1.5** cod\_departamento solo acepta números de hasta 20 dígitos.

**R1.6** cod\_departamento solo acepta números de hasta 20 dígitos.

**R1.7** Los requisitos anteriores son obligatorios para ingresar un ticket.

**R1.8** Se podrá ingresar una ubicación junto con el ticket por medio de un mapa interactivo, o un campo de autocompletado de direcciones. Este campo es opcional

**R1.9** Al ingresar un ticket este se registrará en el sistema con "codigo\_ticket" que será un número único de hasta 20 dígitos

**R1.10** Se hará un registro que contendrá la hora y fecha de envío junto con la siguiente información que también se asignará al ticket

**R1.10.1** El estado solicitud se asignará por defecto a "Pendiente de Revisión".

**R1.10.2** La visibilidad será por defecto "Privado".

**R1.11** El envío correcto del ticket desplegará un mensaje de "Envío exitoso".

**Salida 1:** cod\_ticket, asunto\_ticket, detalles\_solicitud, cod\_departamento, estado\_solicitud, visibilidad\_solicitud, fecha\_hora\_registro.

**Salida 2:** Despliegue de mensaje (indicado en **R1.11**).

**Salida 3:** Dirección en caso de haberla ingresado.

**R.2** Responder ticket: Los encargados de departamento podrán responder a tickets enviados a su departamento.

**Entradas:** codigo\_ticket, estado\_solicitud, detalles\_respuesta, rut\_encargado, fecha\_hora\_registro.

**R2.1:** El codigo\_ticket deberá consistir en un número o números de hasta 20 dígitos.

**R2.2:** El codigo\_ticket deberá estar registrado en el sistema y no pueden haber dos tickets en el sistema con el mismo código.

**R2.3:** detalles\_respuesta deberá contener letras, letras con tildes, números o comillas simples y deberá permitir múltiples líneas.

**R2.4:** El estado solicitud deberá ser un código disponible en la base de datos que no pertenezca a los estados "Remitido" o "Cerrado".

**R2.5:** Para el rut encargado se utilizará el RUT de la persona que esté respondiendo

la solicitud con números sin puntos, sin dígito verificador ni guión.

- **R2.5.1:** Si el encargado es extranjero se utilizará el DNI usando el mismo formato.
- **R2.5.2:** Sólo se aceptará como máximo de 12 caracteres alfanuméricos.

**R2.6:** fecha\_hora\_registro será la fecha y hora que el ticket ha sido respondido en formato **timestamp**.

**R2.7:** Los requisitos anteriores son obligatorios para ingresar una respuesta.

**R2.8:** En el caso que detalles\_respuesta no tenga valor (estén vacíos), no se permitirá hacer envío de la respuesta.

**R2.9:** Una vez que la respuesta ha sido enviada, el sistema debe generar un registro de la respuesta, incluyendo la fecha y hora de respuesta que se registrará usando fecha\_hora\_registro.

**R2.10:** Si el envío de la respuesta es exitoso, se le debe mostrar al encargado un mensaje de confirmación y se enviará un correo al usuario creador del ticket y a aquellos suscritos al ticket.

**Salida 1:** codigo\_ticket, estado\_solicitud, detalles\_respuesta, rut\_encargado, fecha\_hora\_registro.

**Salida 2:** Confirmación de que la respuesta ha sido enviada con éxito (**R2.10**).

**R.3a** Trazabilidad de ticket: permite ver en qué estado se encuentra un ticket ingresado al sistema junto con todos sus cambios registrados en una línea de tiempo.

Entradas: codigo\_ticket.

Requisitos de la función:

**R3a.1:** El codigo\_ticket deberá consistir en un número o números de hasta 20 dígitos.

**R3a.2:** El codigo\_ticket deberá estar registrado en el sistema y no pueden haber dos tickets en el sistema con el mismo código.

**R3a.3** El requisito anterior es obligatorio para llevar a cabo el seguimiento de un ticket.

**R3a.4** Si el código de ticket ingresado corresponde a uno registrado en el sistema, éste recuperará el estado de la solicitud, incluyendo sus cambios en la visibilidad y, en caso de haberse ingresado respuestas, cada modificación será detallada con su respectiva fecha y hora. Todo este historial se presentará de manera organizada a través de una interfaz dedicada con el fin de proporcionar una visión cronológica y clara de los eventos asociados al código de ticket en cuestión.

**R3a.5** Si el codigo\_ticket ingresado NO pertenece a uno registrado en el sistema se desplegará el mensaje de código invalido.

**Salida 1:** interfaz de seguimiento (línea de tiempo).

**Salida 2:** Despliegue de mensaje (indicado en **R3a.5**).

**R3b.** Seguimiento de ticket: permite ver en qué estado se encuentra un ticket ingresado al sistema junto con su historial.

Entradas: codigo\_ticket.

Requisitos de la función:

**R3b.1** Para el codigo\_ticket se utilizarán números enteros en un bigint de 20 dígitos.

**R3b.2** El requisito anterior es obligatorio para llevar a cabo el seguimiento de un ticket.

**R3b.3** Si el `codigo_ticket` ingresado pertenece a uno registrado en el sistema, el sistema recuperará el `estado_solicitud` del ticket correspondiente a dicho `codigo_ticket` y los mostrará en una interfaz específica para dicho propósito.

**R3b.4** Si el `codigo_ticket` ingresado NO pertenece a uno registrado en el sistema se desplegará el mensaje “El `codigo_ticket` introducido no corresponde con ningún ticket registrado en el sistema”.

**Salida 1:** `codigo_ticket`, `estado_solicitud`, interfaz de seguimiento.

**Salida 2:** Despliegue de mensaje (indicado en **R3b.4**).

**R.4** Cambiar estado de ticket: permite cambiar el estado en el que se encuentra el ticket para una mayor comunicación.

**Entradas:** `codigo_ticket`, `estado_solicitud`, `fecha_hora_registro`.

Requisitos de la función:

**R4.1:** El `codigo_ticket` deberá consistir en un número o números de hasta 20 dígitos.

**R4.2:** El `codigo_ticket` deberá estar registrado en el sistema y no pueden haber dos tickets en el sistema con el mismo código.

**R4.3:** El `estado_solicitud` deberá ser un código disponible en la base de datos que no sea “Remitido” o “Cerrado”.

**R4.4:** `fecha_hora_registro` será la fecha y hora que el ticket ha sido modificado en formato **timestamp**.

**R4.5** Los requisitos anteriores son obligatorios para cambiar el estado de un ticket.

**R4.6:** El `estado_ticket` deberá ser distinto del estado actual del ticket, de lo contrario no se permitirá su cambio.

**R4.7:** Si el `codigo_ticket` ingresado NO pertenece a uno registrado en el sistema se desplegará el mensaje de código inválido.

**R4.8:** Una vez cambiado el estado del ticket se desplegará un mensaje de operación exitosa, se realizará un registro usando `fecha_hora_registro` y se enviará un correo al usuario creador del ticket y a aquellos suscritos al ticket.

**Salida 1:** `codigo_ticket`, `estado_solicitud`, `fecha_hora_registro`.

**Salida 2:** Despliegue de mensaje (indicado en **R4.8**).

**R.5** Cambiar visibilidad de ticket: permite cambiar la visibilidad en la que se encuentra el ticket para una mayor comunicación.

**Entradas:** `codigo_ticket`, `visibilidad_solicitud`, `fecha_hora_registro`.

Requisitos de la función:

**R5.1:** El `codigo_ticket` deberá consistir en un número o números de hasta 20 dígitos.

**R5.2:** El `codigo_ticket` deberá estar registrado en el sistema y no pueden haber dos tickets en el sistema con el mismo código.

**R5.3:** `visibilidad_solicitud` será un booleano cuyos valores sean “true” para “Público” y “false” para “Privado”.

**R5.4:** `fecha_hora_registro` será la fecha y hora que el ticket ha sido modificado en formato **timestamp**.

**R5.5** Los requisitos anteriores son obligatorios para cambiar el estado de un ticket.

**R5.6:** La visibilidad\_solicitud deberá ser distinta de la visibilidad actual del ticket, de lo contrario no se permitirá su cambio.

**R5.7:** Si el codigo\_ticket ingresado NO pertenece a uno registrado en el sistema se desplegará el mensaje de código invalido.

**R5.8:** Una vez cambiada visibilidad\_solicitud se desplegará un mensaje de operación exitosa, se realizará un registro usando fecha\_hora\_registro y se enviará un correo al usuario creador del ticket y a aquellos suscritos al ticket.

**Salida 1:** codigo\_ticket, visibilidad\_solicitud, fecha\_hora\_registro.

**Salida 2:** Despliegue de mensaje (indicado en **R5.8**).

**R.6** Cerrar ticket: permite cerrar un ticket para marcar .

**Entradas:** codigo\_ticket, rut\_encargado, fecha\_hora\_registro.

Requisitos de la función:

**R6.1:** El codigo\_ticket deberá consistir en un número o números de hasta 20 dígitos.

**R6.2:** El codigo\_ticket deberá estar registrado en el sistema y no pueden haber dos tickets en el sistema con el mismo código.

**R6.3:** Para el rut encargado se utilizará el RUT de la persona que esté respondiendo la solicitud con números sin puntos, sin dígito verificador ni guión.

- **R6.3.1:** Si el encargado es extranjero se utilizará el DNI usando el mismo formato.
- **R6.3.2:** Sólo se aceptará como máximo 12 caracteres alfanuméricos.

**R6.4:** fecha\_hora\_registro será la fecha y hora que el ticket ha sido modificado en formato **timestamp**.

**R6.5:** Los requisitos anteriores son obligatorios para cerrar un ticket.

**R6.6:** Mediante una interfaz se podrá dar la opción de cerrar el ticket asignado a codigo\_ticket.

**R6.7:** Se desplegará un mensaje de confirmación si realmente desea cerrar el ticket.

- **R6.7.1:** Al aceptar este mensaje el ticket asignado a codigo\_ticket modificará su estado\_solicitud a "Cerrado".
- **R6.7.2:** Los tickets con estado "Cerrado" ya no podrán recibir modificaciones ni respuestas.
- **R6.7.3:** Se realizará un registro de esto usando fecha\_hora\_registro.

**R6.8:** Si el codigo\_ticket ingresado NO pertenece a uno registrado en el sistema se desplegará el mensaje de código invalido.

**R6.9:** Una vez cerrado se enviará un correo al usuario creador del ticket y a aquellos suscritos al ticket.

**Salida 1:** codigo\_ticket, rut\_encargado, estado\_solicitud, fecha\_hora\_registro.

**Salida 2:** Despliegue de mensaje (indicado en **R6.8**).

# Funciones

**R7.** Filtrar lenguaje de ticket: permite ver si un ticket contiene palabras ofensivas para no guardarse en la base de datos.

Entradas: `codigo_ticket`, `descripcion_ticket`

Requisitos de la función:

**R7.1** Para el `codigo_ticket` se utilizarán números enteros en un rango del 1 al 10000000.

**R7.2** La `descripcion_ticket` será recuperada desde el ticket correspondiente al `codigo_ticket` según -aquí va la referencia de la descripción en ingreso de ticket-

**R7.3** `codigo_ticket` y `descripcion_ticket` son obligatorios para llevar a cabo la filtración de lenguaje ofensivo en un ticket.

**R7.6** El sistema tendrá en la base de datos un apartado con palabras ofensivas que indicará cuáles serán las palabras a considerar a la hora someter un ticket al proceso de filtrado.

**R7.6** Cuando un ticket ingrese al sistema, se realizará automáticamente el proceso de filtrado comparando cada una de las palabras de `descripcion_ticket` con las del archivo texto del sistema.

**R7.7** Terminado el proceso de filtrado, si se hallan coincidencias, el ticket será marcado como ofensivo y no se enviará a los tickets por revisar, además se desplegará un mensaje diciendo: "El texto contiene malas palabras. No se puede enviar el ticket." .

**R7.8** Terminado el proceso de filtrado, si NO se hayan coincidencias, el ticket NO será marcado como ofensivo y se enviará a los tickets por revisar, además se desplegará un mensaje diciendo: "Ticket enviado correctamente." .

**Salida:** se desplegará los mensajes descritos en **R7.7** y **R7.8**.

**R8.** Visualizar gráficos estadísticos de los tickets según tipo de ingreso: permite a los administradores ver un gráfico que muestra el comportamiento de los tickets ingresados.

Entradas: `lista_tickets`, `cont_reclamos`, `cont_sugerencias`, `cont_felicitaciones`, `codigo_municipio`.

Requisitos de la función:

**R8.1** La `lista_tickets` estará compuesta por todos los tickets pertenecientes al municipio pertinente registrados en el sistema.

**R8.1.1** Cada ticket tendrá -aquí va lo del ingreso o sea cada dato perteneciente al ticket-

**R8.2** Para `cont_reclamos` se utilizarán números enteros en un rango del 0 al 10000000 el cual se obtendrá sumando los valores en la base de datos.

**R8.3** Para `cont_sugerencias` se utilizarán números enteros en un rango del 0 al 10000000 el cual se obtendrá sumando los valores en la base de datos.

**R8.4** Para `cont_felicitaciones` se utilizarán números enteros en un rango del 0 al 10000000 el cual se obtendrá sumando los valores en la base de datos.

**R8.5** El `codigo_municipio` será recuperado desde el municipio correspondiente según -aquí va la referencia de la descripción en ingreso municipio-

**R8.6** Los requisitos anteriores son obligatorios para llevar a cabo la



visualización del gráfico comparativo de los tickets.

**R8.7** Con el `codigo_municipio` se rescatará la `lista_tickets` perteneciente al municipio correspondiente.

**R8.8** Si el sistema no encuentra tickets en los registros, se desplegará el siguiente mensaje “No hay tickets disponibles para realizar el análisis gráfico”

**R8.9** Mediante la `lista_tickets`, el sistema analizará cada ticket para ver a que `tipo_ingreso` corresponde.

**R8.10** Si el `tipo_ingreso` es ‘reclamo’ el sistema incrementará en 1 el valor de `cont_reclamos`.

**R8.11** Si el `tipo_ingreso` es ‘sugerencia’ el sistema incrementará en 1 el valor de `cont_sugerencias`.

**R8.12** Si el `tipo_ingreso` es ‘felicitacion’ el sistema incrementará en 1 el valor de `cont_felicitaciones`.

**R8.13** Teniendo el análisis de los tickets listo, mediante los valores de `cont_reclamos`, `cont_sugerencias`, `cont_felicitaciones` el sistema calculará los porcentajes de cada uno en función de la suma de los tres.

**R8.14** Con los porcentajes de `cont_reclamos`, `cont_sugerencias`, `cont_felicitaciones` el sistema generará el gráfico (de tipo circular) final de Reclamos vs Sugerencias vs Felicitaciones del municipio correspondiente.

**Salida 1:** Gráfico circular de Reclamos vs Sugerencias vs Felicitaciones.

**R9.** Login de usuario: permite a los usuarios registrados en el sistema iniciar sesión.

Entradas: `rut_usuario`, contraseña.

Requisitos de la función:

**R9.1** Para el `rut_usuario` se utilizará el formato sin puntos ni dígito verificador, siendo este una cadena con la siguiente estructura: xxxxxxxx.

**R9.1.1** Cada ‘x’ será un número entero en un rango del 0 al 9.

**R9.2** Para la contraseña se utilizarán los caracteres del alfabeto en minúscula desde la letra ‘a’ hasta la ‘z’, caracteres del alfabeto en mayúscula desde la letra ‘A’ hasta la ‘Z’, números enteros en un rango del 0 al 9 y caracteres especiales tales como ‘.’, ‘-’, ‘\_’. Esta tendrá un largo mínimo de 8 caracteres y un máximo de 16.

**R9.3** Los requisitos anteriores son obligatorios para iniciar sesión en el sistema.

**R9.4** Si el `rut_usuario` y la contraseña ingresados se encuentran fuera de la sintaxis indicada en **R4.1** y **R4.2** se desplegará el siguiente mensaje: “Ingrese datos que sean válidos”.

**R9.5** Se verificarán los datos ingresados para ver si coinciden con los registrados en el sistema.

**R9.6** Si los datos ingresados NO coinciden con los registrados en el sistema, el proceso de login será interrumpido y se desplegará el mensaje “El rut ingresado no está vinculado a ninguna cuenta existente o la contraseña introducida es incorrecta”. Posteriormente se dará la oportunidad de volver a ingresar los datos.

**R9.7** Si los datos ingresados coinciden con los registrados en el sistema, se continuará con el proceso de login pasando a verificar el rol del usuario que inicio sesión.

**R9.8** Verificado el rol del usuario, el proceso de login será completado desplegando la pantalla correspondiente según el rol de este.

**Salida 1:** Pantalla correspondiente al rol.

**Salida 2:** Despliegue de mensajes (indicados en **R9.4** y **R9.6**).

**R10.** Suscripción a actualizaciones de ticket: permite suscribirse a un ticket para recibir notificaciones sobre cuando sea actualizado por correo electrónico.

**Entradas:** rut\_usuario, cod\_ticket, correo\_electronico\_usuario.

**R10.1:** El cod\_ticket deberá consistir en un número o números de hasta 20 dígitos.

**R10.2:** El cod\_ticket deberá estar registrado en el sistema y no pueden haber dos tickets en el sistema con el mismo código.

**R10.3:** El correo\_electronico\_usuario será un string con formato de correo electronico incluyendo arroba (@).

**R10.4:** Para el rut encargado se utilizará el RUT de la persona que esté respondiendo la solicitud con números sin puntos, sin dígito verificador ni guión.

- **R10.4.1:** Si el encargado es extranjero se utilizará el DNI usando el mismo formato.
- **R10.4.2:** Sólo se aceptará como máximo de 12 caracteres alfanuméricos.

**R10.5:** Los requisitos anteriores son obligatorios para suscribirse a un ticket.

**R10.6:** El sistema mostrará una opción para recibir o no notificaciones de ese ticket cuando una persona esté visualizando un ticket.

**R10.7:** El botón dirá “Recibir notificaciones” si la persona no está suscrita, o “Dejar de Recibir Notificaciones” si es que lo esta.

**R10.8:** Se rescatará el correo electrónico de rut\_usuario mediante correo\_electronico\_usuario.

**R10.9:** Cada vez que haya una actualización o cambio en el ticket al que la persona se ha suscrito, el sistema enviará automáticamente una notificación por correo electrónico informando de la actualización.

**Salida 1:** Despliegue de botón en **R10.7**

**R.11** Visualizar tickets públicos anteriores: El sistema permite a las personas acceder y visualizar un registro de los tickets previamente ingresados. Se presenta en forma de listado y al seleccionar un ticket, se pueden ver los detalles completos del mismo.

**Entradas:** código\_ticket, rut\_ciudadano, código\_depto, tipo\_solicitud, contenido\_solicitud, estado\_solicitud, fecha\_hora\_envio, foto\_adjunta.

**R11.1:** Las personas tendrán una interfaz donde se mostrarán los tickets registrados, organizados generalmente por fecha de ingreso, mostrando los más recientes

primero.

**R11.2:** Se rescatará mediante una consulta SQL en la tabla de tickets todos los datos de los tickets.

**R11.3:** El sistema permitirá a las personas elegir cuántos tickets desean visualizar por página, con opciones para mostrar 20, 50 o 100 tickets en una sola vista.

**R11.4:** El sistema permitirá a la persona regresar al listado general o avanzar a otros tickets sin necesidad de regresar al inicio.

**Salidas:** código\_ticket, rut\_ciudadano, código\_depto, tipo\_solicitud, contenido\_solicitud, estado\_solicitud, fecha\_hora\_envio, foto\_adjunta.

**R.12** Calificar funcionamiento del sistema: Tras el ingreso de un ticket, el sistema enviará un correo electrónico a la persona para solicitar una calificación sobre el funcionamiento del sistema antes de permitirle realizar otro ingreso. La calificación se recoge a través de un link que redirige a la página con el formulario de calificación.

**Entradas:** calificacion, comentario, rut\_usuario, correo\_electronico.

**R12.1:** calificacion es un entero del 1 al 7 que será la calificación asignada por el usuario.

**R12.2:** comentario es un texto de 500 caracteres que será donde el usuario puede comentar la experiencia.

**R12.3:** Tras el ingreso de un ticket, el sistema rescata de la sesión el correo electrónico del usuario, y luego genera y envía un correo electrónico a la dirección.

**R12.4:** El correo electrónico incluirá un mensaje solicitando la calificación del sistema y proporcionará un link que redirige a la página de calificación.

**R12.5:** El módulo de calificación presentará una interfaz donde las personas pueden:

- Seleccionar una calificación numérica del 1 al 7, siendo 1 la calificación más baja y 7 la más alta.
- Ingresar comentarios en un campo de texto para detallar su experiencia con el sistema. Este campo permitirá un máximo de 500 caracteres.
- Presionar un botón de enviar para registrar su feedback.

**R12.6:** Una vez que la persona ha seleccionado una calificación y ha ingresado sus comentarios (si opta por hacerlo), puede presionar el botón de enviar para registrar su feedback en la base de datos del sistema.

**R12.7:** En caso de que calificacion no haya sido seleccionado, se desplegará el mensaje "Completar calificación"

**R12.8:** Tras el envío exitoso, el sistema mostrará un mensaje de agradecimiento por la retroalimentación proporcionada.

**Salida 1:** calificación, comentario

**Salida 2:** Despliegue de mensaje en **R12.7**.

**Salida 3:** Despliegue de mensaje en **R12.8**.

**R.13** Visualizar en el mapa la cantidad de felicitaciones, reclamos, sugerencias: El sistema proporciona una sección que presenta un mapa interactivo, donde cada municipalidad está representada por un punto distintivo. Al seleccionar un punto correspondiente a una municipalidad, el usuario puede ver estadísticas relevantes sobre esa municipalidad.

**Entradas:** Mapa representativo de la zona mediante API de Google Maps, cod\_municipalidad.

**R13.1:** El sistema mostrará un mapa interactivo obtenido mediante la API de Maps Embed en el que cada municipalidad, rescatada de la base de datos mediante su cod\_municipalidad, aparecerá representada por un punto o marcador. Este marcador puede variar en color o tamaño según la cantidad o tipo de ingresos para facilitar la visualización.

**R13.2:** Al pasar el cursor sobre un punto o marcador, se proporcionará una vista previa breve de la municipalidad, como el nombre de la municipalidad y el total de ingresos.

**R13.3:** Al seleccionar (hacer clic en) un punto o marcador, se desplegará un panel o ventana emergente con estadísticas detalladas de esa municipalidad. Las estadísticas incluirán:

- Número total de ingresos.
- Porcentaje de cada tipo de ingreso (felicidades, reclamos, sugerencias).
- Tiempo medio de respuesta a los tickets.

**RF13.4:** El sistema permitirá a los usuarios cerrar el panel o ventana emergente para regresar al mapa general y seleccionar otra municipalidad si lo desean.

**R13.5:** Además de las estadísticas, el panel o ventana emergente podría ofrecer enlaces a acciones relevantes, como dirigirse a la página oficial de la municipalidad.

**Salidas:** Vista del mapa interactivo con marcadores de municipalidades, Estadísticas detalladas de la municipalidad seleccionada.

**R.14** Analizar estadísticas de calificación del sistema: Los administradores del sistema podrán visualizar las calificaciones promedio de toda la municipalidad o de un departamento en específico, junto con gráficos de las calificaciones por tiempo.

**Entradas:** id municipalidad

- **R14.1:** El id municipalidad deberá consistir en un número o números.

- **R14.2:** Los dos requisitos anteriores son obligatorios
- **R14.3:** Si id municipalidad no se encuentra registrado en el sistema, se mostrará el siguiente mensaje: "Ingrese un id municipalidad válido"
- **R14.4:** Si id municipalidad se encuentra en el sistema, se devolverán la calificación promedio de cada departamento en dicha municipalidad, además de un gráfico con el total de calificaciones durante los últimos 7 días.

**Salida 1:** Calificación promedio de cada departamento y gráfico con calificación de los últimos 7 días.

**R.15** Exportar datos de tickets a un archivo: Los encargados y administradores mediante interacción de la aplicación web podrán exportar los datos de los tickets en la base de datos dentro de un rango de fecha dado a un archivo CSV.

- **Entradas:** fecha inicio, fecha termino
- **R15.1:** Tanto fecha inicio como fecha término tendrán el mismo formato que será formato fecha de la forma dd/mm/aaaa
- **R15.2:** El requisito anterior es obligatorio
- **R15.3:** Si el sistema no encuentra tickets entre la fecha inicio y fecha término devolverá un mensaje indicando "No hay datos para exportar"
- **R15.4:** En caso de haber tickets registrados entre la fecha, el sistema generará un archivo en formato CSV de los tickets que se encuentren en esa fecha
  - R15.4.1: Los datos de los tickets a incluir en el archivo son código ticket, estado solicitud, asunto, respuesta, fecha y hora de respuesta.
- **R15.5:** El archivo deberá poder ser descargado y almacenado en algún archivo local.
- **Salida 1:** Archivo CSV con los datos descritos en R15.4 y R15.4.1.
- **Salida 2:** Mensaje indicando que no hay datos para exportar si no se encuentran tickets en el rango de fechas seleccionado.

**R.16** Cambiar a modo oscuro: Los usuarios podrán optar por visualizar la aplicación web en modo oscuro.

- **R16.1:** La aplicación debe ofrecer un "Modo oscuro" que permite a los usuarios activar una paleta de colores de tonalidades oscuras para mejorar la visibilidad de la información en la página en la noche.
- **R16.2:** Los usuarios deben poder acceder y desactivar el modo oscuro desde la interfaz de la aplicación al presionar un botón.
- **R16.3:** El modo oscuro debe aplicarse de manera consistente en toda la aplicación, incluyendo todas las páginas y ventanas emergentes.
- **R16.4:** La activación del modo oscuro debe ser reversible, permitiendo a los usuarios volver al diseño visual estándar de la aplicación en cualquier momento.
- **Salida:** Cambio en la paleta de colores y estilo visual de la aplicación cuando se activa el modo de alto contraste.

**R.17** Derivación de ticket a otra municipalidad o departamento

Entradas: id\_municipalidad o codigo\_depto

- **R.17.1** Al ingresar a algún ticket este debe de tener un botón que diga derivar
- **R.17.2** Al pulsar este botón se desplegará un cuadro de opciones, el cual tendrá dos opciones municipalidad y departamento
- **R.17.3** Al elegir la opción se desplegará un cuadro de texto, en el cual se debe de escribir el id o código correspondiente en el caso de ser municipalidad id municipalidad y en caso de ser departamento codigo\_depto
- **R.17.4** Si el id o código ingresado indicado en **R.17.3** no corresponde con los datos del sistema se desplegará el mensaje: “id o código no existe favor colocar un id o código que exista”
- **R.17.5** Si el id o código ingresado indicado en **R.17.3** corresponde con los datos del sistema se desplegará el mensaje: “la derivación se ha hecho con éxito”
- **salida:** mensaje indicado en **R.17.4** y **R.17.5**

**R.18** Muestra de calendarios con próximos proyectos

**entradas:** evento, descripcion, fecha\_inicio, fecha\_termino

- **R.18.1** Se realizará un calendario como comúnmente lo conocemos con años separados en meses, los meses separados en semana y las semanas separadas en días.
- **R.18.2** Este calendario contendrá próximos proyectos elegidos por los trabajadores del municipio
- **R.18.3** A medida que vayan apareciendo nuevos proyectos este calendario se va a ir actualizando
- **R.18.4** se obtendrán los valores de evento, descripción, fecha\_inicio y fecha\_termino desde la base de datos.
- **R.18.5:** se desplegará un modal con los valores obtenidos en **R.18.4.** al presionar el evento en el calendario.
- **R.18.6:** los eventos se mostrarán a lo largo del calendario, iniciando en su fecha\_inicio y terminando en su fecha\_termino.
- **Salida1:** calendario gráfico en pantalla con los eventos de los proyectos
- **Salida2:** modal descrito en **R.18.5**

**R.19** Agendar visita presencial a la municipalidad

**Entradas:** fecha, hora, rut, nombre

- **R.19.1** Se mostrará en pantalla un calendario de manera gráfica, separado en días, meses y años correspondientes.
- **R.19.2** El usuario al hacer click en un día, se le desplegará un modal en el cual aparecerá su nombre, su rut, los cuales serán obtenidos de la base de datos, el día que presionó en fecha y las horas disponibles en modo de selección de opción dependiendo de la municipalidad.
- **R.19.3:** al presionar un día anterior a la fecha actual se desplegará una alerta diciendo: “Aviso, No puede agendar una hora en una fecha anterior a la actual”.
- **R.19.4:** a medida de que algún usuario elija una hora para atenderse está será sacada de las opciones de la lista de horas para atenderse en la municipalidad.
- **R.19.5:** al guardar correctamente los datos se desplegará una alerta diciendo: “¡éxito! registro exitoso ”
- **Salida:** Mensajes indicados en **R.19.3** y **R.19.5**

**R.20** Asignar roles con permisos: El sistema tendrá un sistema de roles y permisos para asignar distintos apartados de la aplicación web para permitir o prohibir su acceso a los usuarios.

**Entradas:** rut\_usuario, cod\_rol, cod\_permiso

- **R.20.1:** Para el rut encargado se utilizará el RUT de la persona que esté respondiendo la solicitud con números sin puntos, sin dígito verificador ni guión.
  - **R.20.1.1:** Si el encargado es extranjero se utilizará el DNI usando el mismo formato.
  - **R.20.1.2:** Sólo se aceptará como máximo de 12 caracteres alfanuméricos.
- **R.20.2:** cod\_rol deberá ser un numero bigint de 20 dígito único
- **R.20.3:** cod\_permiso deberá ser un numero bigint de 20 dígito único
- **R.20.4:** El usuario podrá visualizar al rut\_usuario ingresado para ver su rol asignado.
- **R.20.5:** El usuario podrá modificar el rol asignado del usuario seleccionado a alguno disponible en la base de datos.
- **R.20.6:** Cada rol contará con uno o varios permisos.
- **R.20.7:** El usuario podrá crear roles con uno o varios permisos, cada uno dado por cod\_permiso.
- **R.20.8:** Si el rut\_usuario existe y el nuevo cod\_rol asignado existe, entonces se modificará el rol del usuario ingresado exitosamente.
- **Salida:** rut\_usuario, cod\_rol.



# Mantenedores:

## R21. Mantenedor de municipalidades

**R21.1** Ingresar Municipalidad: permite al administrador ingresar nuevas municipalidades al sistema.

**Entradas:** cod\_municipalidad, cod\_comuna, cod\_region, nombre\_municipalidad,, correo\_electronico\_municipalidad,, alcalde\_actual, sitio\_web, fecha\_fundación, escudo.

**R21.1.1** cod\_municipalidad será un número entero en un rango del 1 al 100000.

**R21.1.2** cod\_comuna será un número entero en un rango del 1 al 1000.

**R21.1.3** cod\_region será un número entero en un rango del 1 al 1000.

**R21.1.4** nombre\_municipalidad será un texto con una longitud máxima de 200 caracteres.

**R21.1.5** numero\_municipalidad será el signo + junto a un número de 11 dígitos siguiendo el formato:

- +56 2 XX-XXX-XXX para un teléfono fijo de la Región Metropolitana;
- +56 YY X-XXX-XXX para un teléfono fijo de otra región del país.
- +56 9 XXXX-XXXX para un teléfono celular.

**R21.1.6** correo\_electronico\_municipalidad será un texto con un formato válido de email y una longitud máxima de 150 caracteres.

**R21.1.7** direccion\_municipalidad es un atributo compuesto que consta con 2 partes:

**R21.1.7.1** Calle será un string de máximo 50 caracteres de largo pertenecientes al alfabeto en minúscula.

**R21.1.7.2** número será un número entero con una longitud máxima de 4 caracteres.

**R21.1.8** alcalde\_actual será un texto con una longitud máxima de 50 caracteres.

**R21.1.9** sitio\_web será un texto con formato de URL y una longitud máxima de 150 caracteres.

**R21.1.10** fecha\_fundación será una fecha con formato DD/MM/AAAA.

**R21.1.7.1** DD será un número entero en un rango de 01 al 31.

**R21.1.7.2** MM será un número entero en un rango de 01 al 12.

**R21.1.7.3** AAAA tendrá formato xxxx.

**R21.1.7.3.1** Cada 'x' será un número entero en un rango del 0 al 9.

**R21.1.11** escudo\_municipalidad será un archivo de imagen en formato .jpg .jpeg .png con un tamaño máximo de 4MB.

**R21.1.12** Todos los campos son obligatorios para registrar una municipalidad, excepto Alcalde actual, Sitio web oficial, Escudo. En caso de que alguno no esté completado se desplegará el mensaje "Completar datos"

**R21.1.13** Cumpliendo con los requisitos obligatorios, la municipalidad será ingresada al sistema, desplegando el mensaje "Municipalidad ingresada con éxito".

**R21.1.14** El sistema maneja internamente el cod\_municipalidad autoincrementando en 1 cada vez que se ingresa una nueva municipalidad.

**Salida 1:** cod\_municipalidad, cod\_comuna, cod\_region, nombre\_municipalidad, numero\_municipalidad, correo\_electronico\_municipalidad, direccion\_municipalidad, alcalde\_actual, sitio\_web, fecha\_fundacion, escudo

**Salida 2:** Despliegue de mensaje en **R21.1.12**.

**Salida 3:** Despliegue de mensaje en **R21.1.13**.



**R21.2** Actualizar Municipalidad: Se rescatan los registros de la base de datos para ser modificados.

**Entradas:** cod\_municipalidad, cod\_comuna, cod\_region, nombre\_municipalidad, num\_municipalidad, correo\_electronico\_municipalidad, direccion, alcalde\_actual, sitio\_web, fecha\_fundación, escudo, cod\_administrador, **historial\_cambios\_m**, **cod\_cambio\_m**.

**R21.2.1** Se requiere cod\_municipalidad para identificar y actualizar la municipalidad correspondiente.

**R21.2.2** los campos cod\_municipalidad, cod\_comuna, cod\_region, nombre\_municipalidad, telefono\_municipalidad, correo\_electronico\_municipalidad, direccion, alcalde\_actual, sitio\_web, fecha\_fundación, escudo son rescatados de la base de datos; visibles y editables para el administrador del sistema.

**R21.2.3** historial\_cambios\_m es una tabla que guarda, mediante un trigger, las modificaciones en los campos de la tabla municipalidad.

Registra:

```
cod_cambio_m SERIAL PRIMARY KEY,  
operacion CHAR(1) NOT NULL, (¿Necesario?)  
fecha_cambio_m TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
cod_administrador_old INT,  
cod_municipalidad_old INT,  
cod_comuna_old INT,  
cod_region_old INT,  
nombre_municipalidad_old TEXT,  
num_municipalidad_old INT,  
correo_electronico_municipalidad_old TEXT,  
direccion_old TEXT,  
alcalde_actual_old TEXT,  
sitio_web_old TEXT,  
fecha_fundación_old DATE,  
escudo_old BYTEA,  
cod_administrador_new INT,  
cod_municipalidad_new INT,  
cod_comuna_new INT,  
cod_region_new INT,  
nombre_municipalidad_new TEXT,  
num_municipalidad_new INT,  
correo_electronico_municipalidad_new TEXT,  
direccion_new TEXT,  
alcalde_actual_new TEXT,  
sitio_web_new TEXT,  
fecha_fundación_new DATE,  
escudo_new TEXT
```

**R21.2.3.1:** Los campos “\_new” deberán ser completados con el mismo tipo de dato original. En caso de que se intente insertar un dato inválido se desplegará el mensaje “Dato inválido”.

**R21.2.3.2:** En caso de que alguno de los campos de historial\_cambios\_m no sea actualizado se mantendrán los campos “\_new” igual al “\_old”, ya que no todos los campos son obligatorios para actualizar; solo los que el administrador del sistema desee modificar.

**R21.2.3.3:** Al seleccionar “Actualizar municipalidad” se guardarán los cambios en la

base de datos.

**R21.2.3.4:** Se registrarán las modificaciones exitosas en `historial_cambios_m` detallando los campos actualizados por cuál administrador, identificado por `cod_administrador`.

**R21.2.4:** El registro de actualización a la municipalidad será ingresado al sistema, desplegando el mensaje “Municipalidad actualizada con éxito”.

**R21.2.5:** El sistema maneja internamente el **cod\_cambio\_m** autoincrementando en 1 cada vez que se modifica una municipalidad.

**Salida 1:** `cod_cambio_m`, `operacion`, `fecha_cambio_m`, `cod_administrador_old`, `cod_municipalidad_old`, `cod_comuna_old`, `cod_region_old`, `nombre_municipalidad_old`, `num_municipalidad_old`, `correo_electronico_municipalidad_old`, `direccion_old`, `alcalde_actual_old`, `sitio_web_old`, `fecha_fundación_old`, `escudo_old`, `cod_administrador_new`, `cod_municipalidad_new`, `cod_comuna_new`, `cod_region_new`, `nombre_municipalidad_new`, `num_municipalidad_new`, `correo_electronico_municipalidad_new`, `direccion_new`, `alcalde_actual_new`, `sitio_web_new`, `fecha_fundación_new`, `escudo_new`

**Salida 2:** Despliegue de mensaje en **R21.2.3.1**.

**Salida 3:** Despliegue de mensaje en **R21.2.4**.

**R21.4 Eliminar municipalidad:** Función que permite eliminar una municipalidad existente en el sistema.

**Entrada:** `cod_municipalidad`.

**Requisitos:**

**R21.4.1** Para `cod_municipalidad` se utilizarán números enteros en un rango del 1 al 1000.

**R21.4.2** El requisito anterior es obligatorio para llevar a cabo la eliminación de una municipalidad.

**R21.4.3** Mediante la función de visualización de municipalidades (**R17.2**) se mostrarán todas las municipalidades registradas en el sistema junto con una opción para eliminarlas individualmente.

**R21.4.4** Al intentar eliminar una municipalidad del sistema se desplegará el mensaje “¿Estás seguro de que quieres eliminar esta municipalidad?” y se esperará la confirmación o cancelación del usuario.

**R21.4.5** Si el usuario confirma la eliminación, se eliminará la municipalidad del registro del sistema y se mostrará el mensaje “Municipalidad eliminada”.

**R21.4.6** Si el usuario cancela la eliminación, el mensaje desplegado de

**R21.4.4** desaparecerá y se retornará a la visualización de municipalidades.

**Salida 1:** `cod_municipalidad`.

**Salida 2:** Despliegue de mensajes (indicados en **R21.4.4** y **R21.4.5**).

## **R22. Mantenedor de departamentos**

**R22.1** Ingresar Departamento: permite al administrador ingresar nuevos departamentos al sistema.

**Entradas:**        `cod_departamento`,        `nombre_departamento`,        `jefe_departamento`,  
                  `descripcion_departamento`,        `telefono_contacto_departamento`,  
                  `correo_electronico_departamento`, `cod_municipalidad`.

**R22.1.1** Para el `cod_departamento` se utilizarán números enteros en un rango del 1 al 200.

**R22.1.2** `nombre_departamento` será un texto con una longitud máxima de 150 caracteres.

**R22.1.3** `jefe_departamento` será un texto con una longitud máxima de 50 caracteres.

**R22.1.4** `descripcion_departamento` será un texto con una longitud máxima de 300 caracteres.

**R22.1.5** `telefono_contacto` seguirá el formato previamente descrito:

+56 2 XX-XXX-XXX para un teléfono fijo de la Región Metropolitana;

+56 YY X-XXX-XXX para un teléfono fijo de otra región del país.

+56 9 XXXX-XXXX para un teléfono celular.

**R22.1.6** `correo_electronico_departamento` será un texto con un formato válido de email y una longitud máxima de 150 caracteres.

**R22.1.7** `cod_municipalidad` será un número entero que hace referencia al código de la municipalidad a la que pertenece el departamento, utilizando números enteros en un rango del 1 al 100000.

**R22.1.8** Todos los campos son obligatorios para registrar un departamento, excepto `descripcion_departamento`, `jefe_departamento` y `telefono_contacto_departamento`. En caso de que alguno no esté completado se desplegará el mensaje "Completar datos".

**R22.1.9** Cumpliendo con los requisitos obligatorios, el departamento será ingresado al sistema, desplegando el mensaje "Departamento ingresado con éxito".

**R22.1.10** El sistema maneja internamente el `cod_departamento` autoincrementando en 1 cada vez que se ingresa un nuevo departamento.

**Salida 1:**        `cod_departamento`,        `nombre_departamento`,        `jefe_departamento`,  
                  `descripcion_departamento`, `telefono_contacto`, `correo_electronico`, `cod_municipalidad`.

**Salida 2:** Despliegue de mensaje en **R22.2.3.1**.

**Salida 3:** Despliegue de mensaje en **R22.2.4**.

## **R22.2. Actualizar Departamento:** Actualización de Información de Departamento.

**Entradas:** cod\_departamento, nombre\_departamento, jefe\_departamento, descripcion\_departamento, telefono\_contacto\_departamento, correo\_electronico\_departamento, cod\_municipalidad.

**R22.2.1** Se requiere cod\_departamento para identificar y actualizar el departamento correspondiente.

**R22.2.2** Los campos cod\_departamento, nombre\_departamento, jefe\_departamento, telefono\_contacto\_departamento, correo\_electronico\_departamento, cod\_municipalidad son rescatados de la base de datos; visibles y editables para el administrador del sistema.

**R22.2.3** historial\_cambios\_d es una tabla que guarda, mediante un trigger, las modificaciones en los campos de la tabla departamento.

**Registra:**

cod\_cambio\_d SERIAL PRIMARY KEY,  
fecha\_cambio\_d TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,  
cod\_departamento\_old INT,  
nombre\_departamento\_old TEXT,  
jefe\_departamento\_old TEXT,  
descripcion\_departamento\_old TEXT,  
telefono\_contacto\_departamento\_old TEXT,  
correo\_electronico\_departamento\_old TEXT,  
cod\_municipalidad\_old INT,  
cod\_departamento\_new INT,  
nombre\_departamento\_new TEXT,  
jefe\_departamento\_new TEXT,  
descripcion\_departamento\_new TEXT,  
telefono\_contacto\_departamento\_new TEXT,  
correo\_electronico\_departamento\_new TEXT,  
cod\_municipalidad\_new INT

**R22.2.3.1:** Los campos “\_new” deberán ser completados con el mismo tipo de dato original. En caso de que se intente insertar un dato inválido se desplegará el mensaje “Dato inválido” y no se permitirá la actualización.

**R22.2.3.2:** En caso de que alguno de los campos de historial\_cambios\_d no sea actualizado se mantendrán los campos “\_new” igual al “\_old”, ya que no todos los campos son obligatorios para actualizar; solo los que el administrador del sistema desee modificar.

**R22.2.3.3:** Al seleccionar “Actualizar departamento” se guardarán los cambios en la base de datos.

**R22.2.4:** Se registrarán las modificaciones exitosas en historial\_cambios\_d detallando los campos actualizados por cuál administrador, identificado por cod\_administrador.

**R22.2.5:** El registro de actualización al departamento será ingresado al sistema, desplegando el mensaje “Departamento actualizado con éxito”.

**R22.2.6:** El sistema maneja internamente el cod\_cambio\_d autoincrementando en 1 cada vez que se modifica un departamento.

**Salida 1:** cod\_cambio\_d, operacion, fecha\_cambio\_d, cod\_departamento\_old, nombre\_departamento\_old, jefe\_departamento\_old, descripcion\_departamento\_old,

telefono\_contacto\_departamento\_old, correo\_electronico\_departamento\_old,  
cod\_municipalidad\_old, cod\_departamento\_new, nombre\_departamento\_new,  
jefe\_departamento\_new, descripcion\_departamento\_new,  
telefono\_contacto\_departamento\_new, correo\_electronico\_departamento\_new,  
cod\_municipalidad\_new

**Salida 2: Despliegue de mensaje en R22.2.3.1.**

**Salida 3: Despliegue de mensaje en R22.2.5.**

**R22.4** Eliminar departamento: Función que permite eliminar una departamento existente en el sistema.

**Entrada:** cod\_departamento.

**Requisitos:**

**R22.4.1** Para cod\_departamento se utilizarán números enteros en un rango del 1 al 50.

**R22.4.2** El requisito anterior es obligatorio para llevar a cabo la eliminación de un departamento.

**R22.4.3** Mediante la función de visualización de departamentos (**R18.2**) se mostrarán todos los departamentos registrados en el sistema junto con una opción para eliminarlos individualmente.

**R22.4.4** Al intentar eliminar un departamento del sistema se desplegará el mensaje “¿Estás seguro de que quieres eliminar este departamento?” y se esperará la confirmación o cancelación del usuario.

**R22.4.5** Si el usuario confirma la eliminación, se eliminará el departamento del registro del sistema y se mostrará el mensaje “Departamento eliminado”.

**R22.4.6** Si el usuario cancela la eliminación, el mensaje desplegado de

**R22.4.4** desaparecerá y se retornará a la visualización de departamentos.

**Salida 1:** cod\_departamento.

**Salida 2:** Despliegue de mensajes (indicados en **R22.4.4** y **R22.4.5**).

## **R23 Mantenedor de regiones**

### **R23.1 Ingresar región:**

- **Entradas:** id\_region, nombre\_region
- **R23.1.1:** Para el id\_region se utilizarán números enteros positivos de 2 dígitos para identificar cada región.
- **R23.1.2:** Para nombre\_region solo se aceptará caracteres del abecedario, con un tamaño máximo para este campo de 30.
- **R23.1.3:** Los requisitos anteriores son obligatorios para ingresar una región.
- **R23.1.4:** Si el id\_region ingresado ya se encuentra en el sistema, no se realizará el ingreso de región y se desplegará el mensaje "ID de Región ingresado ya existe".
- **R23.1.5:** Si el id\_region ingresado no se encuentra en el sistema, entonces se hará ingreso de la región y se desplegará el mensaje "Región ingresada con éxito".
- **Salida 1:** código\_region, nombre\_region
- **Salida 2:** Despliegue de mensajes de región ingresada y no ingresada.

### **R23.2 Actualizar región:**

- **Entradas:** id\_region, nombre\_region
- **R23.2.1:** Para el id\_region se utilizarán números enteros positivos de 2 dígitos para identificar cada región.
- **R23.2.2:** Para nombre\_region solo se aceptará caracteres del abecedario, con un tamaño máximo para este campo de 30.
- **R23.2.3:** Los requisitos anteriores son obligatorios para actualizar una región.
- **R23.2.4:** Si el id\_region ingresado no se encuentra en el sistema, se desplegará el mensaje "ID de Región ingresado no existe".
- **R23.2.5:** Si el id\_region ingresado se encuentra en el sistema, entonces se reemplaza el nombre\_region del sistema por el ingresado y se desplegará el mensaje "Región actualizada con éxito".
- **Salida 1:** código\_region, nombre\_region.
- **Salida 2:** Despliegue de mensajes de región actualizada y región no existe.

### **R23.4 Eliminar región:** Función que permite eliminar una región existente en el sistema.

**Entrada:** cod\_region.

#### **Requisitos:**

**R23.4.1** Para cod\_region se utilizarán números enteros en un rango del 1 al 50.

**R23.4.2** El requisito anterior es obligatorio para llevar a cabo la eliminación de una región.

**R23.4.3** Mediante la función de visualización de regiones (**R19.2**) se mostrarán todas las regiones registradas en el sistema junto con una opción para eliminarlas individualmente.

**R23.4.4** Al intentar eliminar una región del sistema se desplegará el mensaje "¿Estás seguro de que quieres eliminar esta región?" y se esperará la confirmación o cancelación del usuario.

**R23.4.5** Si el usuario confirma la eliminación, se eliminará la región del registro del sistema y se mostrará el mensaje "Región eliminada".

**R23.4.6** Si el usuario cancela la eliminación, el mensaje desplegado de **R23.4.4** desaparecerá y se retornará a la visualización de regiones.

**Salida 1:** cod\_region.

**Salida 2:** Despliegue de mensajes (indicados en **R23.4.4** y **R23.4.5**).

## **R24. Mantenedor de comunas**

**R24.1** Ingresar comuna: Función que permite al administrador ingresar nuevas comunas al sistema.

**Entrada:** cod\_comuna, cod\_region, nombre\_comuna.

### **Requisitos:**

**R24.1.1** Para cod\_comuna se utilizarán números enteros en un rango del 1 al 1000.

**R24.1.2** Para cod\_region se utilizarán números enteros en un rango del 1 al 50.

**R24.1.3** Para nombre\_comuna se utilizará un string de máximo 50 caracteres de largo pertenecientes al alfabeto.

**R24.1.4** Todos los requisitos anteriores son obligatorios para el ingreso de una comuna.

**R24.1.5** Mediante la función de visualización de comunas (**R24.2**) se mostrarán todas las comunas registradas en el sistema junto con la opción para ingresar una nueva.

**R24.1.6** Al elegir la opción para ingresar una nueva comuna se desplegará un formulario para ingresar los datos de esta.

**R24.1.7** Si todos los datos son válidos, se almacenará la información de la comuna en el sistema y se desplegará el mensaje "Comuna ingresada"

**R24.1.8** cod\_comuna será manejado internamente por el sistema autoincrementando en 1 su valor cada vez que se ingrese una nueva comuna.

**Salida 1:** cod\_comuna, cod\_region, nombre\_comuna.

**Salida 2:** Despliegue de mensaje (indicado en **R24.1.8**).

**R24.2** Visualizar comunas: Función que permite ver todas las comunas existentes en el sistema.

**Entrada:** listado\_comunas, cod\_comuna, cod\_region, nombre\_comuna.

**Requisitos:**

**R24.2.1** listado\_comunas será un listado que contendrá todas las comunas registradas en el sistema.

**R24.2.1.1** Cada comuna estará conformada por su cod\_comuna, cod\_region, nombre\_comuna.

**R24.2.2** Para cod\_comuna se utilizarán números enteros en un rango del 1 al 1000.

**R24.2.3** Para cod\_region se utilizarán números enteros en un rango del 1 al 50.

**R24.2.4** Para nombre\_comuna se utilizará un string de máximo 50 caracteres de largo pertenecientes al alfabeto.

**R24.2.5** Todos los requisitos anteriores son obligatorios para la visualización de las comunas.

**R24.2.6** Mediante cod\_comuna, se rescatará toda la información de cada comuna registrada en el sistema y se creará listado\_comunas.

**R24.2.7** El administrador visualizará las comunas mediante listado\_comunas y tendrá opciones para agregar una nueva comuna, modificar una comuna existente en el sistema o eliminar una comuna existente en el sistema.

**R24.2.8** Si el sistema no tiene ninguna comuna registrada, el administrador visualizará listado\_comunas, pero este aparecerá vacío.

**Salida 1:** listado\_comunas, cod\_comuna, cod\_region, nombre\_comuna.



**R24.3** Actualizar comuna: Función que permite al administrador actualizar una comuna existente en el sistema.

**Entrada:** cod\_comuna, cod\_region, nombre\_comuna.

**Requisitos:**

**R24.3.1** Para cod\_comuna se utilizarán números enteros en un rango del 1 al 1000.

**R24.3.2** Para cod\_region se utilizarán números enteros en un rango del 1 al 50.

**R24.3.3** Para nombre\_comuna se utilizará un string de máximo 50 caracteres de largo pertenecientes al alfabeto.

**R24.3.4** Todos los requisitos anteriores son obligatorios para actualizar los datos de una comuna.

**R24.3.5** Mediante la función de visualización de comunas (**R20.2**) se mostrarán todas las comunas registradas en el sistema junto con la opción para actualizarlas individualmente.

**R24.3.6** Al elegir la opción para actualizar una comuna se desplegará un formulario para ingresar los nuevos datos de esta. El formulario mostrará por defecto los datos actuales de la comuna.

**R24.3.7** El cod\_comuna no podrá ser actualizado mediante esta función.

**R24.3.8** Si todos los datos son válidos, se almacenará la información actualizada de la comuna en el sistema y se desplegará el mensaje "Comuna actualizada".

**Salida 1:** cod\_comuna, cod\_region, nombre\_comuna.

**Salida 2:** Despliegue de mensaje (indicado en **R24.3.8**).

**R24.4** Eliminar comuna: Función que permite eliminar una comuna existente en el sistema.

**Entrada:** cod\_comuna.

**Requisitos:**

**R24.4.1** Para cod\_comuna se utilizarán números enteros en un rango del 1 al 1000.

**R24.4.2** El requisito anterior es obligatorio para llevar a cabo la eliminación de una comuna.

**R24.4.3** Mediante la función de visualización de comunas (**R20.2**) se mostrarán todas las comunas registradas en el sistema junto con una opción para eliminarlas individualmente.

**R24.4.4** Al intentar eliminar una comuna del sistema se desplegará el mensaje "¿Estás seguro de que quieres eliminar esta comuna?" y se esperará la confirmación o cancelación del usuario.

**R24.4.5** Si el usuario confirma la eliminación, se eliminará la comuna del registro del sistema y se mostrará el mensaje "Comuna eliminada".

**R24.4.6** Si el usuario cancela la eliminación, el mensaje desplegado de **R24.4.4** desaparecerá y se retornará a la visualización de comunas.

**Salida 1:** cod\_comuna.

**Salida 2:** Despliegue de mensajes (indicados en **R24.4.4** y **R24.4.5**).

## **R25. Mantenedor de roles**

**R25.1** Ingresar rol: Función que permite al administrador ingresar nuevos roles al sistema.

**Entrada:** cod\_rol, nombre\_rol.

**Requisitos:**

**R25.1.1** Para cod\_rol se utilizarán números enteros en un rango del 1 al 20.

**R25.1.2** Para nombre\_rol se utilizará un string de máximo 50 caracteres de largo pertenecientes al alfabeto.

**R25.1.3** Todos los requisitos anteriores son obligatorios para el ingreso de un rol.

**R25.1.4** Mediante la función de visualización de roles (**R25.2**) se mostrarán todos los roles registrados en el sistema junto con la opción para ingresar uno nuevo.

**R25.1.5** Al elegir la opción para ingresar un nuevo rol se desplegará un formulario para ingresar los datos de este.

**R25.1.6** Si todos los datos son válidos, se almacenará la información del rol en el sistema y se desplegará el mensaje "Rol ingresado"

**R25.1.7** cod\_rol será manejado internamente por el sistema autoincrementando en 1 su valor cada vez que se ingrese un nuevo rol.

**Salida 1:** cod\_rol, nombre\_rol.

**Salida 2:** Despliegue de mensaje (indicado en **R25.1.7**).

**R25.2** Visualizar roles: Función que permite ver todos los roles existentes en el sistema.

**Entrada:** listado\_rols, cod\_rol, nombre\_rol.

**Requisitos:**

**R25.2.1** listado\_rols será un listado que contendrá todos los roles registrados en el sistema.

**R25.2.1.1** Cada rol estará conformado por su cod\_rol, nombre\_rol.

**R25.2.2** Para cod\_rol se utilizarán números enteros en un rango del 1 al 20.

**R25.2.3** Para nombre\_rol se utilizará un string de máximo 50 caracteres de largo pertenecientes al alfabeto.

**R25.2.5** Todos los requisitos anteriores son obligatorios para la visualización de los roles.

**R25.2.6** Mediante cod\_rol, se rescatará toda la información de cada rol registrado en el sistema y se creará listado\_rols.

**R25.2.7** El administrador visualizará los roles mediante listado\_rols y tendrá opciones para agregar un nuevo rol, modificar un rol existente en el sistema o eliminar un rol existente en el sistema.

**R25.2.8** Si el sistema no tiene ningún rol registrado, el administrador visualizará listado\_rols, pero este aparecerá vacío.

**Salida 1:** listado\_rols, cod\_rol, nombre\_rol.

**R25.3** Actualizar rol: Función que permite al administrador actualizar un rol existente en el sistema.

**Entrada:** cod\_rol, nombre\_rol.

**Requisitos:**

**R25.3.1** Para cod\_rol se utilizarán números enteros en un rango del 1 al 20.

**R25.3.2** Para nombre\_rol se utilizará un string de máximo 50 caracteres de largo pertenecientes al alfabeto.

**R25.3.3** Todos los requisitos anteriores son obligatorios para actualizar los datos de un rol.

**R25.3.4** Mediante la función de visualización de roles (**R25.2**) se mostrarán todos los roles registrados en el sistema junto con la opción para actualizarlos individualmente.

**R25.3.5** Al elegir la opción para actualizar un rol se desplegará un formulario para ingresar los nuevos datos de este. El formulario mostrará por defecto los datos actuales del rol.

**R25.3.6** El cod\_rol no podrá ser actualizado mediante esta función.

**R25.3.7** Si todos los datos son válidos, se almacenará la información actualizada del rol en el sistema y se desplegará el mensaje "Rol actualizado".

**Salida 1:** cod\_rol, nombre\_rol.

**Salida 2:** Despliegue de mensaje (indicado en **R25.3.7**).

**R25.4** Eliminar rol: Función que permite eliminar un rol existente en el sistema.

**Entrada:** cod\_rol.

**Requisitos:**

**R25.4.1** Para cod\_rol se utilizarán números enteros en un rango del 1 al 20.

**R25.4.2** El requisito anterior es obligatorio para llevar a cabo la eliminación de un rol.

**R25.4.3** Mediante la función de visualización de roles (**R25.2**) se mostrarán todos los roles registrados en el sistema junto con una opción para eliminarlos individualmente.

**R25.4.4** Al intentar eliminar un rol del sistema se desplegará el mensaje "¿Estás seguro de que quieres eliminar este rol?" y se esperará la confirmación o cancelación del usuario.

**R25.4.5** Si el usuario confirma la eliminación, se eliminará el rol del registro del sistema y se mostrará el mensaje "Rol eliminado".

**R25.4.6** Si el usuario cancela la eliminación, el mensaje desplegado de

**R25.4.4** desaparecerá y se retornará a la visualización de roles.

**Salida 1:** cod\_rol.

**Salida 2:** Despliegue de mensajes (indicados en **R25.4.4** y **R25.4.5**).

## **R26. Mantenedor de proyectos**

**R26.1** Ingresar proyecto: Función que permite al administrador ingresar nuevos proyectos al sistema.

**Entrada:**           cod\_proyecto,           cod\_departamento,           nombre\_proyecto,  
descripcion\_proyecto, fecha\_inicio\_proyecto, fecha\_termino\_estimada\_proyecto.

**Requisitos:**

**R26.1.1** Para cod\_proyecto se utilizarán números enteros en un rango del 1 al 1000.

**R26.1.2** Para cod\_departamento se utilizarán números enteros en un rango del 1 al 50.

**R26.1.3** Para nombre\_proyecto se utilizará un string de caracteres alfanuméricos y el signo ' (comilla simple) de máximo 50 caracteres.

**R26.1.4** Para descripcion\_proyecto se utilizará un string de máximo 500 caracteres de largo pertenecientes al código ASCII.

**R26.1.5** Para fecha\_inicio\_proyecto se utilizará un valor tipo date con el formato DD/MM/AAAA.

**R26.1.5.1** DD será el día.

**R26.1.5.2** MM será el mes.

**R26.1.5.3** AAAA será el año.

**R26.1.6** Para fecha\_termino\_estimada\_proyecto se utilizará el mismo formato de fecha\_inicio\_proyecto (**R22.1.5**).

**R26.1.7** Todos los requisitos anteriores son obligatorios para el ingreso de un proyecto.

**R26.1.8** Mediante la función de visualización de proyectos (**R22.2**) se mostrarán todos los proyectos registrados en el sistema junto con la opción para ingresar uno nuevo.

**R26.1.9** Al elegir la opción para ingresar un nuevo proyecto se desplegará un formulario para ingresar los datos de este.

**R26.1.10** Si todos los datos son válidos, se almacenará la información del proyecto en el sistema y se desplegará el mensaje "Proyecto ingresado"

**R26.1.11** cod\_proyecto será manejado internamente por el sistema autoincrementando en 1 su valor cada vez que se ingrese un nuevo proyecto.

**Salida 1:**           cod\_proyecto,           cod\_departamento,           nombre\_proyecto,  
descripcion\_proyecto, fecha\_inicio\_proyecto, fecha\_termino\_estimada\_proyecto.

**Salida 2:** Despliegue de mensaje (indicado en **R26.1.10**).

**R26.2** Visualizar proyectos: Función que permite ver todos los proyectos existentes en el sistema.

**Entrada:** listado\_proyectos, cod\_proyecto, cod\_departamento, nombre\_proyecto, descripcion\_proyecto, fecha\_inicio\_proyecto, fecha\_termino\_estimada\_proyecto.

**Requisitos:**

**R26.2.1** listado\_proyectos será un listado que contendrá todos los proyectos registrados en el sistema.

**R26.2.1.1** Cada proyecto estará conformado por su cod\_proyecto, cod\_departamento, nombre\_proyecto, descripcion\_proyecto, fecha\_inicio\_proyecto, fecha\_termino\_estimada\_proyecto.

**R26.2.2** Para cod\_proyecto se utilizarán números enteros en un rango del 1 al 1000.

**R26.2.3** Para cod\_departamento se utilizarán números enteros en un rango del 1 al 50.

**R26.2.4** Para nombre\_proyecto se utilizará un string de caracteres alfanuméricos y el signo ' (comilla simple) de máximo 50 caracteres.

**R26.2.5** Para descripcion\_proyecto se utilizará un string de máximo 500 caracteres de largo pertenecientes al código ASCII.

**R26.2.6** Para fecha\_inicio\_proyecto se utilizará un valor tipo date con el formato DD/MM/AAAA.

**R26.2.6.1** DD será el día.

**R26.2.6.2** MM será el mes.

**R26.2.6.3** AAAA será el año.

**R26.2.7** Para fecha\_termino\_estimada\_proyecto se utilizará el mismo formato de fecha\_inicio\_proyecto (**R26.2.6**).

**R26.2.8** Todos los requisitos anteriores son obligatorios para la visualización de los proyectos.

**R26.2.9** Mediante cod\_proyecto, se rescatará toda la información de cada proyecto registrado en el sistema y se creará listado\_proyectos.

**R26.2.10** El administrador visualizará los proyectos mediante listado\_proyectos y tendrá opciones para agregar un nuevo proyecto, modificar un proyecto existente en el sistema o eliminar un proyecto existente en el sistema.

**R26.2.11** Si el sistema no tiene ningún proyecto registrado, el administrador visualizará listado\_proyectos, pero este aparecerá vacío.

**Salida 1:** listado\_proyectos, cod\_proyecto, cod\_departamento, nombre\_proyecto, descripcion\_proyecto, fecha\_inicio\_proyecto, fecha\_termino\_estimada\_proyecto.

**R26.3** Actualizar proyecto: Función que permite al administrador actualizar un proyecto existente en el sistema.

**Entrada:** cod\_proyecto, cod\_departamento, nombre\_proyecto, descripcion\_proyecto, fecha\_inicio\_proyecto, fecha\_termino\_estimada\_proyecto.

**Requisitos:**

**R26.3.1** Para cod\_proyecto se utilizarán números enteros en un rango del 1 al 1000.

**R26.3.2** Para cod\_departamento se utilizarán números enteros en un rango del 1 al 50.

**R26.3.3** Para nombre\_proyecto se utilizará un string de caracteres alfanuméricos y el signo ' (comilla simple) de máximo 50 caracteres.

**R26.3.4** Para descripcion\_proyecto se utilizará un string de máximo 500 caracteres de largo pertenecientes al código ASCII.

**R26.3.5** Para fecha\_inicio\_proyecto se utilizará un valor tipo date con el formato DD/MM/AAAA.

**R26.3.5.1** DD será el día.

**R26.3.5.2** MM será el mes.

**R26.3.5.3** AAAA será el año.

**R26.3.6** Para fecha\_termino\_estimada\_proyecto se utilizará el mismo formato de fecha\_inicio\_proyecto (**R26.3.5**).

**R26.3.7** Todos los requisitos anteriores son obligatorios para actualizar los datos de un proyecto.

**R26.3.8** Mediante la función de visualización de proyectos (**R22.2**) se mostrarán todos los proyectos registrados en el sistema junto con la opción para actualizarlos individualmente.

**R26.3.9** Al elegir la opción para actualizar un proyecto se desplegará un formulario para ingresar los nuevos datos de este. El formulario mostrará por defecto los datos actuales del proyecto.

**R26.3.10** El cod\_proyecto y cod\_departamento no podrán ser actualizados mediante esta función.

**R26.3.1** Si todos los datos son válidos, se almacenará la información actualizada del proyecto en el sistema y se desplegará el mensaje "Proyecto actualizado".

**Salida 1:** cod\_proyecto, cod\_departamento, nombre\_proyecto, descripcion\_proyecto, fecha\_inicio\_proyecto, fecha\_termino\_estimada\_proyecto.

**Salida 2:** Despliegue de mensaje (indicado en **R26.3.7**).

**R26.4** Eliminar proyecto: Función que permite eliminar un proyecto existente en el sistema.

**Entrada:** cod\_proyecto.

**Requisitos:**

**R26.4.1** Para cod\_proyecto se utilizarán números enteros en un rango del 1 al 1000.

**R26.4.2** El requisito anterior es obligatorio para llevar a cabo la eliminación de un proyecto.

**R26.4.3** Mediante la función de visualización de proyectos (**R26.2**) se mostrarán todos los proyectos registrados en el sistema junto con una opción para eliminarlos individualmente.

**R26.4.4** Al intentar eliminar un proyecto del sistema se desplegará el mensaje “¿Estás seguro de que quieres eliminar este proyecto?” y se esperará la confirmación o cancelación del usuario.

**R26.4.5** Si el usuario confirma la eliminación, se eliminará el proyecto del registro del sistema y se mostrará el mensaje “Proyecto eliminado”.

**R26.4.6** Si el usuario cancela la eliminación, el mensaje desplegado de **R26.4.4** desaparecerá y se retornará a la visualización de proyectos.

**Salida 1:** cod\_proyecto.

**Salida 2:** Despliegue de mensajes (indicados en **R26.4.4** y **R26.4.5**).

## **R27. Mantenedor de estados de ticket**

**R27.1** Ingresar estado de ticket: Función que permite al administrador ingresar nuevos estados de ticket al sistema.

**Entrada:** cod\_estado, nombre\_estado, descripcion\_estado.

**Requisitos:**

**R27.1.1** Para cod\_estado se utilizarán números enteros en un rango del 1 al 20.

**R27.1.2** Para nombre\_estado se utilizará un string de máximo 50 caracteres alfanuméricos

**R27.1.3** Para descripcion\_estado se utilizará un string de máximo 250 caracteres de largo pertenecientes al código ASCII.

**R27.1.4** Todos los requisitos anteriores son obligatorios para el ingreso de un estado de ticket.

**R27.1.5** Mediante la función de visualización de estados de ticket (**R22.2**) se mostrarán todos los estados de ticket registrados en el sistema junto con la opción para ingresar uno nuevo.

**R27.1.6** Al elegir la opción para ingresar un nuevo estado de ticket se desplegará un formulario para ingresar los datos de este.

**R27.1.7** Si todos los datos son válidos, se almacenará la información del estado de ticket en el sistema y se desplegará el mensaje "Estado de ticket ingresado"

**R27.1.8** cod\_estado será manejado internamente por el sistema autoincrementando en 1 su valor cada vez que se ingrese un nuevo estado de ticket.

**Salida 1:** cod\_estado, nombre\_estado, descripcion\_estado.

**Salida 2:** Despliegue de mensaje (indicado en **R27.1.7**).



**R27.2** Visualizar estados de ticket: Función que permite ver todos los estados de ticket existentes en el sistema.

**Entrada:** listado\_estados, cod\_estado, nombre\_estado, descripcion\_estado.

**Requisitos:**

**R27.2.1** listado\_estados será un listado que contendrá todos los estados de ticket registrados en el sistema.

**R27.2.1.1** Cada estados de ticket estará conformado por su cod\_estado, nombre\_estado, descripcion\_estado.

**R27.2.2** Para cod\_estado se utilizarán números enteros en un rango del 1 al 20.

**R27.2.3** Para nombre\_estado se utilizará un string de máximo 50 alfanuméricos

**R27.2.4** Para descripcion\_estado se utilizará un string de máximo 250 caracteres de largo pertenecientes al código ASCII.

**R27.2.5** Todos los requisitos anteriores son obligatorios para la visualización de los estados de ticket.

**R27.2.6** Mediante cod\_estado, se rescatará toda la información de cada estado de ticket registrado en el sistema y se creará listado\_estados.

**R27.2.7** El administrador visualizará los estados de ticket mediante listado\_estados y tendrá opciones para agregar un nuevo estado de ticket, modificar un estado de ticket existente en el sistema o eliminar un estado de ticket existente en el sistema.

**R27.2.8** Si el sistema no tiene ningún estado de ticket registrado, el administrador visualizará listado\_estados, pero este aparecerá vacío.

**Salida 1:** listado\_estados, cod\_estado, nombre\_estado, descripcion\_estado.

**R27.3** Actualizar estado de ticket: Función que permite al administrador actualizar un estado de ticket existente en el sistema.

**Entrada:** cod\_estado, nombre\_estado, descripcion\_estado.

**Requisitos:**

**R27.3.1** Para cod\_estado se utilizarán números enteros en un rango del 1 al 20.

**R27.3.2** Para nombre\_estado se utilizará un string de máximo 50 caracteres alfanuméricos

**R27.3.3** Para descripcion\_estado se utilizará un string de máximo 250 caracteres de largo pertenecientes al código ASCII.

**R27.3.4** Todos los requisitos anteriores son obligatorios para actualizar los datos de un estado de ticket.

**R27.3.5** Mediante la función de visualización de estados de ticket (**R22.2**) se mostrarán todos los estados de ticket registrados en el sistema junto con la opción para actualizarlos individualmente.

**R27.3.6** Al elegir la opción para actualizar un estado de ticket se desplegará un formulario para ingresar los nuevos datos de este. El formulario mostrará por defecto los datos actuales del estado de ticket.

**R27.3.7** El cod\_estado no podrá ser actualizado mediante esta función.

**R27.3.8** Los estados "Pendiente de revisión", "En Proceso", "Cerrado" y "Remitido" no pueden ser actualizados.

**R27.3.9** Si todos los datos son válidos, se almacenará la información actualizada del estado de ticket en el sistema y se desplegará el mensaje "Estado de ticket actualizado".

**Salida 1:** cod\_estado, nombre\_estado, descripcion\_estado.

**Salida 2:** Despliegue de mensaje (indicado en **R27.3.9**).

**R27.4** Eliminar estado de ticket: Función que permite eliminar un estado de ticket existente en el sistema.

**Entrada:** cod\_estado.

**Requisitos:**

**R27.4.1** Para cod\_estado se utilizarán números enteros en un rango del 1 al 20.

**R27.4.2** El requisito anterior es obligatorio para llevar a cabo la eliminación de un estado de ticket.

**R27.4.3** Mediante la función de visualización de estados de ticket (**R27.2**) se mostrarán todos los estados de ticket registrados en el sistema junto con una opción para eliminarlos individualmente.

**R27.4.4** Al intentar eliminar un estado de ticket del sistema se desplegará el mensaje “¿Estás seguro de que quieres eliminar este estado de ticket?” y se esperará la confirmación o cancelación del usuario.

**R27.4.5** Si el usuario confirma la eliminación, se eliminará el estado de ticket del registro del sistema y se mostrará el mensaje “Estado de ticket eliminado”.

**R27.4.6** Si el usuario cancela la eliminación, el mensaje desplegado de **R27.4.4** desaparecerá y se retornará a la visualización de estados de ticket.

**R27.4.7** Los estados “Pendiente de revisión”, “En Proceso”, “Cerrado” y “Remitido” no pueden ser eliminados.

**Salida 1:** cod\_estado.

**Salida 2:** Despliegue de mensajes (indicados en **R27.4.4** y **R27.4.5**).

## **R28. Mantenedor de palabras ofensivas**

**R28.1** Ingresar palabra ofensiva: Función que permite al administrador ingresar nuevas palabras ofensivas al sistema.

**Entrada:** cod\_palabra, palabra.

**Requisitos:**

**R28.1.1** Para cod\_palabra se utilizarán números enteros en un rango del 1 al 500.

**R28.1.2** Para palabra se utilizará un string de máximo 255 caracteres del alfabeto.

**R28.1.3** Todos los requisitos anteriores son obligatorios para el ingreso de una palabra ofensiva.

**R28.1.4** Mediante la función de visualización de palabras ofensivas (**R22.2**) se mostrarán todas las palabras ofensivas registradas en el sistema junto con la opción para ingresar una nueva.

**R28.1.5** Al elegir la opción para ingresar un nuevo palabra ofensiva se desplegará un formulario para ingresar los datos de esta.

**R28.1.6** Si todos los datos son válidos, se almacenará la información del palabra ofensiva en el sistema y se desplegará el mensaje "Palabra ofensiva ingresada"

**R28.1.7** cod\_palabra será manejado internamente por el sistema autoincrementando en 1 su valor cada vez que se ingrese un nuevo palabra ofensiva.

**Salida 1:** cod\_palabra, palabra.

**Salida 2:** Despliegue de mensaje (indicado en **R28.1.6**).

**R28.2** Visualizar palabras ofensivas: Función que permite ver todas las palabras ofensivas existentes en el sistema.

**Entrada:** listado\_palabras, cod\_palabra, palabra.

**Requisitos:**

**R28.2.1** listado\_palabras será un listado que contendrá todas las palabras ofensivas registradas en el sistema.

**R28.2.1.1** Cada palabra ofensiva estará conformada por su cod\_palabra, palabra.

**R28.2.2** Para cod\_palabra se utilizarán números enteros en un rango del 1 al 500.

**R28.2.3** Para palabra se utilizará un string de máximo 255 caracteres del alfabeto.

**R28.2.4** Todos los requisitos anteriores son obligatorios para la visualización de las palabras ofensivas.

**R28.2.5** Mediante cod\_palabra, se rescatará toda la información de cada palabra ofensiva registrada en el sistema y se creará listado\_palabras.

**R28.2.6** El administrador visualizará las palabras ofensivas mediante listado\_palabras y tendrá opciones para agregar una nueva palabra ofensiva, modificar un palabra ofensiva existente en el sistema o eliminar una palabra ofensiva existente en el sistema.

**R28.2.7** Si el sistema no tiene ninguna palabra ofensiva registrada, el administrador visualizará listado\_palabras, pero este aparecerá vacío.

**Salida 1:** listado\_estados, cod\_estado, nombre\_estado, descripcion\_estado.

**R28.3** Actualizar palabra ofensiva: Función que permite al administrador actualizar una palabra ofensiva existente en el sistema.

**Entrada:** cod\_palabra, palabra.

**Requisitos:**

**R28.3.1** Para cod\_palabra se utilizarán números enteros en un rango del 1 al 500.

**R28.3.2** Para palabra se utilizará un string de máximo 255 caracteres del alfabeto.

**R28.3.3** Todos los requisitos anteriores son obligatorios para actualizar los datos de una palabra ofensiva.

**R28.3.4** Mediante la función de visualización de palabras ofensivas (**R22.2**) se mostrarán todas las palabras ofensivas registradas en el sistema junto con la opción para actualizarlas individualmente.

**R28.3.5** Al elegir la opción para actualizar una palabra ofensiva se desplegará un formulario para ingresar los nuevos datos de esta. El formulario mostrará por defecto los datos actuales de la palabra ofensiva.

**R28.3.6** El cod\_palabra no podrá ser actualizado mediante esta función.

**R28.3.7** Si todos los datos son válidos, se almacenará la información actualizada de la palabra ofensiva en el sistema y se desplegará el mensaje "Palabra ofensiva actualizada".

**Salida 1:** cod\_palabra, palabra.

**Salida 2:** Despliegue de mensaje (indicado en **R28.3.7**).

**R28.4** Eliminar palabra ofensiva: Función que permite eliminar una palabra ofensiva existente en el sistema.

**Entrada:** cod\_palabra.

**Requisitos:**

**R28.4.1** Para cod\_palabra se utilizarán números enteros en un rango del 1 al 500.

**R28.4.2** El requisito anterior es obligatorio para llevar a cabo la eliminación de una palabra ofensiva.

**R28.4.3** Mediante la función de visualización de palabras ofensivas (**R28.2**) se mostrarán todas las palabras ofensivas registradas en el sistema junto con una opción para eliminarlas individualmente.

**R28.4.4** Al intentar eliminar una palabra ofensiva del sistema se desplegará el mensaje "¿Estás seguro de que quieres eliminar esta palabra ofensiva?" y se esperará la confirmación o cancelación del usuario.

**R28.4.5** Si el usuario confirma la eliminación, se eliminará la palabra del registro del sistema y se mostrará el mensaje "Palabra ofensiva eliminada".

**R28.4.6** Si el usuario cancela la eliminación, el mensaje desplegado de

**R28.4.4** desaparecerá y se retornará a la visualización de palabras ofensivas.

**Salida 1:** cod\_palabra.

**Salida 2:** Despliegue de mensajes (indicados en **R28.4.4** y **R28.4.5**).