



UCSC



INGENIERÍA CIVIL INFORMÁTICA

TALLER DE INGENIERÍA DE SOFTWARE I

Especificación de Requisitos

Integrantes:

Diego Araneda Hidalgo
Tomás Fell Tapia
Nicolás Parra Cruz
Ignacio Valdebenito Cuevas

Profesor:

Lorenzo Paredes

Ayudante:

Juan Baeza

1 de diciembre de 2023

1. Enunciado

En el marco de la nueva reglamentación sanitaria sobre el distanciamiento social y la prevención del COVID-19, La UCSC a través de su Unidad de Infraestructura DO, se ve en la necesidad de gestionar un proyecto que permita controlar el acceso vehicular al interior del Campus San Andrés.

Las variables que se deben considerar son:

1. Actualmente existe un alto movimiento vehicular al interior del campus san Andrés, tanto de vehículos de carrier, personal y alumnos de la UCSC.
2. Con el fin dar seguridad a todos los trabajadores, se debe restringir el acceso sólo a aquellos vehículos que estén registrados en una base de datos y asociados a personal o trabajadores UCSC.
3. Se debe contar con un sistema que detecte a aquellos vehículos registrado para un acceso más expedito, evitando así atochamientos en los accesos al campus en los rios de entrada y salida de personal.
4. Se requiere poder controlar el personal que ingresa, ya que en el retorno progresivo a la normalidad sólo tendrán esta capacidad los trabajadores que su jefatura directa determine.

La UCSC no posee ninguna tecnología que permita hacer lo antes señalado, sólo es controlado por un guardia en el acceso, por lo que deberá ser desarrollado desde cero.

2. Funcionalidades del Sistema

2.1. Plataforma Vehicular UCSC

El sistema tendrá un módulo que contendrá las siguientes funciones:

- Mantenedores.
- Gestión de usuario.
- Gestión vehicular.
- Gestión de ingreso.
- Recuperar contraseña.
- Activación de cuenta.
- Desactivación de cuenta.
- Evaluación de pagina.
- Reporte contingencia vehicular en horas.
- Gestión de perfil.
- Personalización de interfaz.
- Asignación cuarentena

2.2. Mantenedores

2.2.1. R1. Usuarios

Entradas: ID usuario, RUN, Nombre, Apellido, Correo institucional, Tipo de usuario, Teléfono, Salud, Contraseña.

Restricciones de los datos:

- **R1.1:** El RUN consta de un cuerpo de máximo 8 dígitos y sin dígito verificador, teniendo así el siguiente formato "12345678".
- **R1.2:** Para el nombre se utilizará una cadena de caracteres, con un máximo de 50, donde se permitirán los espacios.
- **R1.3:** Para el apellido se utilizará una cadena de caracteres, con un máximo de 50, donde se permitirán los espacios.
- **R1.4:** El correo electrónico debe ser una cadena de caracteres con un largo máximo de 30, con el carácter @ como división entre el nombre de usuario (lado izquierdo) y el dominio ucsc.cl (lado derecho).
- **R1.5:** Para el teléfono se utilizará el formato estándar telefónico chileno el cual es +56 9 XXXX-XXXX para el caso de los teléfonos celulares
- **R1.6:** El tipo de usuario solo permitirá dos entradas predefinidas, las cuales son Institucional y invitado.
- **R1.7:** Para la contraseña se utilizará una cadena de un máximo de 80 caracteres, en donde no se permitirá los espacios en blanco.

Obligatoriedad de los datos:

- **R1.8** Todos los campos mencionados son obligatorios para poder mostrar, añadir, editar o borrar una región dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R1.9** Al crear un usuario, el sistema validará cada campo según las restricciones mencionadas, además se verificará que el run ingresado no exista en la base de datos, si se cumplen todas las validaciones, entonces serán almacenados dentro de la base de datos, asociando un único id a cada usuario.
- **R1.10** Al mostrar los datos de un usuario, el sistema mostrará los datos a través de una tabla la cual contendrá los datos que fueron ingresados previamente por el administrador de la página.
- **R1.11** Al actualizar los datos de un usuario se requiere de la ID del registro el cual se quiere editar, además de sus distintos datos los cuales se cargarán en un modal para su modificación, el sistema validará cada campo según las restricciones mencionadas y si se cumplen todas las validaciones, entonces los campos serán actualizados dentro de la base de datos.
- **R1.12** Para eliminar los datos de un usuario se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar una confirmación de borrado, una vez confirmado el registro de dicho usuario se borra de la base de datos, actualizando la tabla de los usuarios.

Salida 1: Tabla Usuarios actualizada.

2.2.2. R2. Vehículos

Entradas: ID vehículo, Patente, Modelo, Color, Tipo vehículo, ID usuario

Restricciones de los datos:

- **R2.1:** La patente seguirá el formato establecido en Chile, el cual se conforma por 4 letras y 2 números, o bien, para motocicletas 3 letras y 2 números. Ejemplo: BB-CC·12 / AAA · 12
- **R2.2:** El modelo estará predefinido según el mantenedor R11
- **R2.3:** El color estará predefinido según el mantenedor R10
- **R2.4:** El tipo de vehículo estará predefinido según el mantenedor R4

Obligatoriedad de los datos:

- **R2.5** Todos los campos mencionados son obligatorios para poder mostrar, añadir, editar o borrar un vehículo dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R2.6** Al crear un vehículo, el sistema validará cada campo según las restricciones mencionadas, si se cumplen todas las validaciones, entonces serán almacenados dentro de la base de datos, asociando un único ID vehículo a cada vehículo y asociándolo con el ID usuario que lo ingreso.
- **R2.7** Al mostrar los datos de un vehículo, el sistema mostrará los datos a través de una tabla la cual contendrá los datos que fueron ingresados previamente por el administrador de la pagina.
- **R2.8** Al actualizar los datos de un vehículo se requiere de la ID del registro el cual se quiere editar, además de sus distintos datos los cuales se cargaran en un modal para su modificación, el sistema validará cada campo según las restricciones mencionadas y si se cumplen todas las validaciones, entonces los campos serán actualizados dentro de la base de datos.
- **R2.9** Para eliminar los datos de un vehículo se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar un confirmación de borrado, una vez confirmado el registro de dicho usuario se borra de la base de datos, actualizando la tabla de los vehículos.

Salida 1: Tabla Vehículos actualizada.

2.2.3. R3. Tipo usuario

Entradas: ID tipo usuario, Nombre.

Restricciones de los datos:

- **R3.1:** Para el nombre se utilizará una cadena de caracteres, con un máximo de 30, donde se permitirán los espacios.

Obligatoriedad de los datos:

- **R3.2** El campo mencionado es de carácter obligatorio para poder añadir, mostrar, editar o borrar un tipo de usuario dentro del sistema. No se puede omitir el dato mencionado anteriormente.

Lógica de la función:

- **R3.3** Al crear un tipo de usuario, el sistema validará cada campo según la restricción mencionada, si se cumple la validación mencionada, entonces será almacenado dentro de la base de datos, asociando un único id a cada tipo usuario.
- **R3.4** Al mostrar los datos de un tipo usuario, el sistema mostrará los datos a través de una tabla la cual contendrá los datos que fueron ingresados previamente por el administrador de la página .
- **R3.5** Al actualizar los datos de un tipo usuario se requiere de la ID del registro el cual se quiere editar, además de sus distintos datos los cuales se cargaran en un modal para su modificación, el sistema validará cada campo según la restricción mencionada y si se cumple la validación, entonces los campos serán actualizados dentro de la base de datos.
- **R3.6** Para eliminar los datos de un tipo usuario se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar un confirmación de borrado, una vez confirmado el registro de dicho tipo usuario se borra de la base de datos, actualizando la tabla de los tipo usuario.

Salida 1: Tabla Tipo Usuarios actualizada.

2.2.4. R4. Tipo vehículo

Entradas: ID tipo vehículo, Nombre.

Restricciones de los datos:

- **R4.1:** Para el nombre se utilizará una cadena de caracteres, con un máximo de 30, donde se permitirán los espacios.

Obligatoriedad de los datos:

- **R4.2** El campo mencionado es de carácter obligatorio para poder añadir, mostrar, editar o borrar un tipo de vehículo dentro del sistema. No se puede omitir el dato mencionado anteriormente.

Lógica de la función:

- **R4.3** Al crear un tipo de vehículo, el sistema validará cada campo según las restricciones mencionadas, entonces serán almacenados dentro de la base de datos, asociando un único id a cada tipo de vehículo.
- **R4.4** Al mostrar los datos de un tipo vehículo, el sistema mostrará los datos a través de una tabla la cual contendrá los datos que fueron ingresados previamente por el administrador de la página.
- **R4.5** Al actualizar los datos de un tipo de vehículo se requiere de la ID del registro el cual se quiere editar, además de sus distintos datos los cuales se cargaran en un modal para su modificación, el sistema validará cada campo según la restricción mencionada y si se cumple la validación, entonces los campos serán actualizados dentro de la base de datos.
- **R4.6** Para eliminar los datos de un tipo de vehículo se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar un confirmación de borrado, una vez confirmado el registro de dicho usuario se borra de la base de datos, actualizando la tabla de tipo de vehículo.

Salida 1: Tabla Tipo vehículo actualizada.

2.2.5. R5. Establecimiento

Entradas: código establecimiento, nombre establecimiento, dirección, código comuna.

Restricciones de los datos:

- **R5.1:** El código establecimiento debe estar definido por un identificador dentro de la base de datos, el cual será un número entero positivo.
- **R5.2:** Para el nombre establecimiento se utilizará una cadena de caracteres, con un máximo de 60, donde se permitirán los espacios.
- **R5.3:** Para la dirección se utilizará una cadena de caracteres, con un máximo de 60, donde se permitirán los espacios; ésta debe ser válida según la región/comuna.
- **R5.4:** El código comuna debe estar definido por un identificador dentro de la base de datos, el cual será un número entero positivo.

Obligatoriedad de los datos:

- **R5.6** Todos los campos mencionados son obligatorios para ingresar un establecimiento dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R5.7** Al crear un establecimiento el sistema validará cada campo según las restricciones mencionadas, además se verificará que el código establecimiento ingresado no exista en la base de datos, si se cumplen todas las validaciones, entonces serán almacenados dentro de la base de datos.

Salida 1: código establecimiento, nombre establecimiento, dirección, código comuna.

2.2.6. R6. Casetas

Entradas: ID caseta, nombre, dirección

Restricciones de los datos:

- **R6.1:** La ID caseta debe estar definida por un identificador dentro de la base de datos, el cual será un número entero positivo.
- **R6.2:** Para el nombre se utilizará una cadena de caracteres, con un máximo de 50, donde se permitirán los espacios.
- **R6.3:** Para la dirección se utilizará una cadena de caracteres, con un máximo de 70, donde se permitirán los espacios; ésta debe ser válida según la región/comuna.

Obligatoriedad de los datos:

- **R6.4** Todos los campos mencionados son obligatorios para poder mostrar, añadir, editar o borrar una región dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R6.5** Al ingresar una caseta, el sistema validará cada campo según las restricciones mencionadas. Si se cumplen todas las validaciones, entonces serán almacenados dentro de la base de datos.

Salida 1: ID caseta, nombre, dirección.

2.2.7. R7. Guardias

Entradas: ID guardia, Nombre, ID caseta.

Restricciones de los datos:

- **R7.1:** El ID guardia debe estar definida por un identificador dentro de la base de datos, el cual será un número entero positivo.
- **R7.2:** Para el nombre se utilizará una cadena de caracteres, con un máximo de 50, donde se permitirán los espacios.
- **R7.3:** La ID caseta debe estar definida según el R6.1.

Obligatoriedad de los datos:

- **R7.4** Todos los campos mencionados son obligatorios para poder mostrar, añadir, editar o borrar una región dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R7.5** Al ingresar un guardia, el sistema validará cada campo según las restricciones mencionadas, además se verificará que el ID caseta ingresado exista en la base de datos, si se cumplen todas las validaciones, entonces serán almacenados dentro de la base de datos.

Salida 1: ID guardia, Nombre, ID caseta.

2.2.8. R8. Comunas

Entradas: ID usuario, comuna, ID provincia.

Restricciones de los datos:

- **R8.1:** Para la comuna se utilizará una cadena de caracteres, con un máximo de 64, donde se permitirán los espacios.

Obligatoriedad de los datos:

- **R8.2** El campo mencionado es de carácter obligatorio para poder mostrar, editar o borrar una comuna dentro del sistema. No se puede omitir el dato mencionado anteriormente.

Lógica de la función:

- **R8.3** Al mostrar los datos de las comunas, el sistema mostrará todos los datos de las comunas.
- **R8.4** Al actualizar los datos de una comuna se requiere de la ID del registro el cual se quiere editar, además de sus distintos datos los cuales se cargaran en un modal para su modificación, el sistema validará cada campo según las restricciones mencionadas y si se cumplen todas las validaciones, entonces los campos serán actualizados dentro de la base de datos.
- **R8.5** Para eliminar los datos de una comuna se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar un confirmación de borrado, una vez confirmado el registro de la comuna se borrará de la base de datos, actualizando la tabla de las comunas.

Salida 1: Tabla Usuarios actualizada.

2.2.9. R9. Regiones

Entradas: ID región, región, Abreviatura, Capital.

Restricciones de los datos:

- **R9.1:** Para la región se utilizará una cadena de caracteres, con un máximo de 64, donde se permitirán los espacios.
- **R9.2:** Para la abreviatura se utilizará una cadena de caracteres, con un máximo de 4, donde no se permitirán los espacios.
- **R9.3:** Para la capital se utilizará una cadena de caracteres, con un máximo de 64, donde se permitirán los espacios.

Obligatoriedad de los datos:

- **R9.4** Todos los campos mencionados son obligatorios para poder mostrar, editar o borrar una región dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R9.5** Al mostrar los datos de las regiones, el sistema mostrará todos los datos de las regiones.
- **R9.6** Al actualizar los datos de una región se requiere de la ID del registro el cual se quiere editar, además de sus distintos datos los cuales se cargaran en un modal para su modificación, el sistema validará cada campo según las restricciones mencionadas y si se cumplen todas las validaciones, entonces los campos serán actualizados dentro de la base de datos.
- **R9.7** Para eliminar los datos de una región se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar un confirmación de borrado, una vez confirmado el registro de la región se borrara de la base de datos, actualizando la tabla de las regiones.

Salida 1: Tabla Regiones actualizada.

2.2.10. R10. Colores

Entradas: ID color, color.

Restricciones de los datos:

- **R10.1:** Para el color se utilizará una cadena de caracteres, con un máximo de 20, donde se permitirán los espacios.

Obligatoriedad de los datos:

- **R10.2** El campo mencionado es de carácter obligatorio para poder añadir, mostrar, editar o borrar un color dentro del sistema. No se puede omitir el dato mencionado anteriormente.

Lógica de la función:

- **R10.3** Al crear un color, este color ingresado debe cumplir con su validación, entonces será almacenado dentro de la base de datos, asociando un único id a cada color.
- **R10.4** Al mostrar los datos de un color, estos se verán a través de una tabla la cual contendrá los datos que fueron ingresados previamente por el usuario.
- **R10.5** Al actualizar los datos de un color se requiere de la ID del registro el cual se quiere editar, además de sus distintos datos los cuales se cargaran en un modal para su modificación, el sistema validará cada campo según la restricción mencionada y si se cumple la validación, entonces los campos serán actualizados dentro de la base de datos.
- **R10.6** Para eliminar los datos de un color se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar un confirmación de borrado, una vez confirmado el registro de dicho color se borra de la base de datos, actualizando la tabla de los colores.

Salida 1: Tabla Colores actualizada.

2.2.11. R11. Modelos

Entradas: ID modelo, modelo.

Restricciones de los datos:

- **R11.1:** Para el modelo se utilizará una cadena de caracteres, con un máximo de 40, donde se permitirán los espacios.

Obligatoriedad de los datos:

- **R11.2** El campo mencionado es de carácter obligatorio para poder añadir, mostrar, editar o borrar un modelo dentro del sistema. No se puede omitir el dato mencionado anteriormente.

Lógica de la función:

- **R11.3** Al crear un modelo, este modelo ingresado debe existir en el mercado actual, si se cumple la validación, entonces será almacenado dentro de la base de datos, asociando un único id a cada modelo.
- **R11.4** Al mostrar los datos de un modelo, estos se verán a través de una tabla la cual contendrá los datos que fueron ingresados previamente por el administrador de la página .
- **R11.5** Al actualizar los datos de un modelo se requiere de la ID del registro el cual se quiere editar, además de sus distintos datos los cuales se cargaran en un modal para su modificación, el sistema validará cada campo según la restricción mencionada y si se cumple la validación, entonces el campo será actualizado dentro de la base de datos.
- **R11.6** Para eliminar los datos de un modelo se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar un confirmación de borrado, una vez confirmado el registro de dicho modelo se borra de la base de datos, actualizando la tabla de los modelos.

Salida 1: Tabla Modelos actualizada.

2.3. Gestión de usuario

2.3.1. R12. Registro de usuario

Entradas: RUN, Nombre, Correo institucional, Tipo de usuario, Contraseña.

Restricciones de los datos:

- **R12.1:** EL RUN deberá seguir la restricción de R1.1.
- **R12.2:** EL nombre deberá seguir la restricción de R1.2.
- **R12.3:** La contraseña deberá seguir la restricción de R1.7.
- **R12.4:** EL correo deberá seguir la restricción de R1.4.
- **R12.5:** El tipo de usuario solo permitirá entradas predefinidas de R3.

Obligatoriedad de los datos:

- **R12.6** Todos los campos mencionados son obligatorios para registrar un usuario dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R12.7** Al ingresar los datos de un miembro institucional o invitado, el sistema validará cada campo según las restricciones mencionadas, además se verificará que el RUN ingresado no exista en la base de dato, si se cumplen todas las validaciones, entonces serán almacenados dentro de la base de datos, asociando un único id a cada usuario y direccionando a la pagina de login. En el caso en que el RUN ya exista de desplegara un mensaje indicando que el RUN ya esta registrado.
- **R12.8** Al registrarse exitosamente, se mandara un correo al usuario registrado para activar su cuenta.

Salida 1: Pantalla login.

Salida 2: Mensaje en R12.7.

Salida 3: Correo de activación.

2.3.2. R13. Login de usuario

Entradas: RUN, contraseña.

Restricciones de los datos:

- **R13.1:** EL RUN deberá seguir la restricción de R12.1.
- **R13.2:** La contraseña deberá seguir la restricción de R12.7.

Obligatoriedad de los datos:

- **R13.3** Todos los campos mencionados son obligatorios para el login del usuario dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R13.4** Al ingresar el RUN y la contraseña y presionar en iniciar sesión, se consultara que el usuario exista y tenga su cuenta activada, si esto se cumple se le direccionara a la pagina de inicio y se iniciara la sesión. En caso de que ocurra alguna error se mostrar un mensaje de inicio de sesión fallido.

Salida 1: Pantalla Inicio.

Salida 2: Mensajes en R13.4.

2.3.3. R14. Cerrar sesión

Entradas: correo, RUN

Restricciones de los datos:

- **R14.1:** EL correo deberá seguir la restricción de R12.4.
- **R14.2:** EL RUN deberá seguir la restricción de R12.1.

Obligatoriedad de los datos:

- **R14.3** Todos los campos mencionados son obligatorios para cerrar la sesión del usuario dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R14.4** Al estar la sesión activa, el RUN estará definido en una variable de sesión, cuando se presione cerrar sesión se analizara si el RUN esta definido, en caso de estarlo se destruye la sesión y se manda un correo de evaluación de la pagina.

Salida 1: Pantalla de login.

2.4. Activación de cuenta

2.4.1. R15. Activación de cuenta

Entradas: Correo, salud.

Restricciones de los datos:

- **R15.1:** Salud esta definido con tres valores predeterminados, los cuales son positivo, observación y negativo.
- **R15.2:** EL correo deberá seguir la restricción de R12.4.

Obligatoriedad de los datos:

- **R15.3** El correo y la salud es obligatorios para la activación de la cuenta del usuario dentro del sistema. No se puede omitir este campo.

Lógica de la función:

- **R15.4** Al momento de ingresar al link del correo, se direccionara a una encuesta sobre el COVID-19, en dicha encuesta se rescataran los valores para así asignar un estado de salud al usuario. Una vez terminada la encuesta se actualizara el valor de salud en la base de datos y se reemplazara el valor estado en la base de datos de 0 a 1, indicando así que la cuenta del usuario esta activada.

Salida 1: salud, estado **Salida 2:** Pantalla de login.

2.5. Recuperar contraseña

2.5.1. R16. Recuperar contraseña

Entradas: Correo, nueva contraseña, repetición nueva contraseña

Restricciones de los datos:

- **R16.1:** Correo tendrá la restricción R12.4.
- **R16.2:** Las contraseñas tendrán la restricción R12.7.

Obligatoriedad de los datos:

- **R16.3** El correo y las nuevas contraseñas son obligatorias para la recuperación de la contraseña de la cuenta del usuario dentro del sistema. No se puede omitir este campo.

Lógica de la función:

- **R16.4** Al momento de ingresar a recuperar contraseña se pedirá el correo de la cuenta a la cual se le quiere recuperar la contraseña, luego se enviara un correo con un link que contiene un token creado específicamente para ese usuario y el cual se actualiza en la base de datos, seguidamente se direccionara a la pagina de login mostrando un mensaje de “por favor revise su correo”.
- **R16.5** A continuación al momento de ingresar a dicho link se direccionara a una pagina de recuperar contraseña, en donde se pedirá la nueva contraseña y la repetición de esta, si ambas contraseñas no son iguales se mostrara el mensaje “Las contraseñas no son iguales se deberán cambiar. En caso de ser iguales, se actualizara la contraseña para ese usuario en la base de datos y se direccionara a la pagina de login mostrando un mensaje de “Inicia sesión con tu nueva contraseña”.

Salida 1: Pantalla de login con mensaje de R16.4. **Salida 2:** Mensaje de contraseñas de fallo de R16.5. **Salida 3:** Pantalla de login con mensaje de éxito de R16.5.

2.6. Gestión vehicular

2.6.1. R17. Registrar datos de vehículo

Entradas: Patente, modelo vehículo, color, tipo vehículo , fecha ingreso.

Restricciones de los datos:

- **R17.1:** La patente seguirá la restricción R2.1.
- **R17.2:** El modelo vehículo solo permitirá entradas predefinidas de R11.
- **R17.3:** El color solo permitirá entradas predefinidas de R10.
- **R17.4:** El tipo vehículo solo permitirá entradas predefinidas de R4.
- **R17.5:** La fecha ingreso, tendrá un formato “YYYY-MM-DD”.

Obligatoriedad de los datos:

- **R17.6** Todos los campos mencionados son obligatorios para registrar un vehículo dentro del sistema. No se puede omitir ninguno de estos datos mencionados anteriormente.

Lógica de la función:

- **R17.7** El usuario interactúa con un botón de añadir vehículo lo que permitirá ingresar datos. Posterior a la modificación, oprimir un botón de guardar para salvar los cambios realizados, almacenando la nueva información en la base de datos.

Salida 1: Tabla de mis vehículos actualizada.

2.6.2. R18. Eliminar vehículo.

Entradas: ID vehículo

Restricciones de los datos:

- **R18.1:** La patente se debe conformar mediante lo especificado en el R2.1.

Obligatoriedad de los datos:

- **R18.2** La patente es obligatoria para saber que registro de vehículo se quiere eliminar.

Lógica de la función:

- **R18.3** Se verifica que la patente existe en la base de datos. Si es correcto, entonces desplegará un mensaje de consulta indicando si desea eliminar los registros del vehículo correspondiente; confirmar eliminará la información al respecto dentro de la base de datos.

Salida 1: Patente

Salida 2: Mensaje de eliminación exitosa: de R18.3

2.6.3. R19. Modificar datos del vehículo.

Entradas: Patente, modelo, color, tipo vehículo.

Restricciones de los datos:

- **R19.1:** La patente seguirá la restricción R2.1.
- **R19.2:** El modelo vehículo solo permitirá entradas predefinidas de R11.
- **R19.3:** El color solo permitirá entradas predefinidas de R10.
- **R19.4:** El tipo vehículo solo permitirá entradas predefinidas de R4.

Obligatoriedad de los datos:

- **R19.5:** Ningún dato es de carácter obligatorio.

Lógica de la función:

- **R19.6:** El usuario interactúa con un botón de añadir vehículo lo que permitirá editar datos ya ingresados en etapas anteriores. Posterior a la modificación, oprimir un botón de guardar para salvar los cambios realizados, almacenando la nueva información en la base de datos.

Salida 1: Tabla de mis vehículos actualizada.

2.7. Gestión de ingreso/egreso

2.7.1. R20. Validar ingreso y egreso de vehículos.

Entradas: Código QR

Restricciones de los datos:

- **R20.1:** Los códigos QR deben almacenar tanto la id del usuario como la patente del vehículo.
- **R20.2:** La id del usuario debe ser existente dentro de la base de datos.
- **R20.3:** La patente debe seguir el formato establecido en R2.1 para permitir el ingreso al recinto.

Obligatoriedad de los datos:

- **R20.4:** El código QR y la información que almacena es de carácter obligatorio para la funcionalidad de la función.

Lógica de la función:

- **R20.5:** El usuario al momento de llegar al recinto se encontrará con un token lector de códigos QR donde se analizará la información contenida dentro del mismo.
- **R20.6:** Si la información coincide con las restricciones descritas en los R14.1, R14.2, R14.3, entonces se verificará la existencia de la id junto a la patente dentro de la base de datos.
- **R20.7:** Si el usuario, perteneciente a la id, posee dentro de sus vehículos registrados la patente almacenada en el QR y además no posee registros incompletos dentro de la tabla historial, entonces se validará el ingreso al recinto permitiendo su entrada. O bien, si no existe alguno de estos dentro de la base de datos, se negará su ingreso.
- **R20.8:** Se validará también, al momento de salir, si existen registros dentro de la base de datos que no posean hora de salida para que se añada conforme a su id su respectiva hora de egreso.
- **R20.9:** Al ser permitido el acceso de un usuario, la información extraída de la base de datos se enviará y registrará en el historial de ingreso vehicular.

Salida 1: Mensaje de acceso aceptado de R20.7.

Salida 1: Mensaje de acceso denegado de R20.7.

Salida 2: Mensaje de egreso aceptado de R20.8.

2.7.2. R21. Generar QR.

Entradas: Patente, ID usuario.

Restricciones de los datos:

- **R21.1:** La patente deberá seguir la restricción de R2.1.

Obligatoriedad de los datos:

- **R21.2:** La patente es obligatoria para poder controlar el acceso y por ende el conteo del aforo actual.

Lógica de la función:

- **R21.3:** Al momento de presionar el botón QR en mis vehículos, se desplegará en un modal el código QR con los datos de patente e ID usuario.

Salida 1: Modal de R.

2.7.3. R22. Registrar historial acceso vehicular.

Entradas: id historial, Hora entrada, Hora salida, Patente, Nombre, Correo electrónico, run.

Restricciones de los datos:

- **R22.1:** La hora_ingreso y hora_salida, tendrán formato “HH:mm:ss”. Utilizando también el huso “UTC-04:00 Santiago”.
- **R22.2:** El nombre tendrá formato establecido en R1.2.
- **R22.3:** El correo tendrá formato establecido en R1.4.
- **R22.4:** El run tendrá formato establecido en R1.1.

Obligatoriedad de los datos:

- **R22.6:** Todos los campos mencionados anteriormente son obligatorios para poder crear el registro dentro del sistema.

Lógica de la función:

- **R22.7:** Al momento de validar el ingreso de un usuario, se almacenará la información dentro de la base de datos, la cual consta de: hora_ingreso, hora_salida, Patente, Nombre, Correo institucional/personal y run. Posteriormente, al validar su salida, dentro de la misma tabla en la base de datos se almacenará también la hora_salida.

Salida 1: id historial, Hora entrada, Hora salida, Patente, Nombre, Correo electrónico, run.

2.7.4. R23. Verificar vehículo.

Entradas: QR

Restricciones de los datos:

- **R23.1:** Los códigos QR deben estar asociados a vehículos registrados en la base de datos del sistema.

Obligatoriedad de los datos:

- **R23.2** La presencia de un código QR válido es obligatoria para acceder al establecimiento.

Lógica de la función:

- **R23.3** Cuando un vehículo se acerca al acceso, el sistema debe activar un lector de códigos QR.
- **R23.4** Al momento de leer el código QR se verifica si es válido y si éste tiene asociado un vehículo que esté registrado en la base de datos. Si es válido, entonces el sistema debe permitir el acceso al campus señalando que puede ingresar. Por el contrario, si no es válido, o bien, el vehículo no está asociado al QR, entonces se impedirá el acceso señalando el motivo correspondiente a la falta.

Salida 1: RUN, Nombre, correo institucional, patente.

Salida 2: Mensaje de acceso aceptado de R22.4.

Salida 2: Mensaje de acceso denegado de R22.4.

2.8. Gestión de perfil

2.8.1. R24. Gestión invitados.

Entradas: RUN invitado, nombre invitado, correo invitado.

Restricciones de los datos:

- **R24.1:** EL RUN invitado deberá seguir la restricción de R1.1.
- **R24.2:** EL nombre invitado deberá seguir la restricción de R1.2.
- **R24.3:** El correo invitado debe ser una cadena de caracteres con un largo máximo de 30, con el carácter @ como división entre el nombre de usuario (lado izquierdo) y el dominio (lado derecho).

Obligatoriedad de los datos:

- **R24.4:** Todos los datos son de carácter obligatorio.

Lógica de la función:

- **R24.5:** Al Ingresar los datos requeridos el sistema validará que estos sean correctos. Si se cumple lo anterior se registrarán en la base de datos.
- **R24.6:** Para eliminar los datos de un invitado se requiere de la ID del registro el cual se quiere eliminar, para así luego de presionar el botón de eliminar desplegar un mensaje de confirmación de borrado, una vez confirmado, el registro de dicho invitado se borra de la base de datos, actualizando la tabla de los invitados del usuario.

Salida 1: Tabla de invitados actualizada.

2.8.2. R25. Modificar datos de perfil.

Entradas: nombre, correo institucional, correo secundario , teléfono, contraseña.

Restricciones de los datos:

- **R25.1:** EL nombre deberá seguir la restricción de R1.2.
- **R25.2:** EL correo deberá seguir la restricción de R1.4.
- **R25.3:** EL correo secundario deberá seguir la restricción de R24.3.
- **R25.4:** EL teléfono deberá seguir la restricción de R1.5.
- **R25.5:** La contraseña deberá seguir la restricción de R1.7.

Obligatoriedad de los datos:

- **R25.6** Ningún dato es de carácter obligatorio.

Lógica de la función:

- **R25.7** El usuario interactúa con las casilla que contienen los datos a modificar, luego de que el usuario interactué con el botón de guardar permitirá que se guarden datos que fueron modificados previamente por el usuario.

Salida 1: nombre, correo institucional, correo secundario , teléfono, contraseña.

2.8.3. R26. Modificar foto de perfil.

Entradas: RUN, foto perfil

Restricciones de los datos:

- **R26.1:** EL RUN deberá seguir la restricción de R1.1.
- **R26.2:** La foto perfil debe tener un formato .jpg, .jpeg o .png.

Obligatoriedad de los datos:

- **R26.3** Los datos RUN y foto perfil serán de carácter obligatorio, para saber de esta manera a qué usuario se le asigna una foto nueva de perfil.

Lógica de la función:

- **R26.4** Se verifica que el RUN exista dentro de la base de datos para luego poder asignarle la foto nueva a dicho perfil.
- **R26.5** Se ingresa la imagen elegida por el usuario mediante un botón subir imagen. Luego se guarda la imagen mediante un botón de guardar imagen en la base de datos, la imagen se guardará como una dirección en la base de datos.

Salida 1: Imagen actualizada.

2.9. Evaluación de pagina

2.9.1. R27. Ingresar Datos para la consulta.

Entradas: rapidez, interfaz, accesibilidad.

Restricciones de los datos:

- **R27.1:** La rapidez es un valor entero de 1 a 5, representado a través de cinco estrellas.
- **R27.2:** La interfaz es un valor entero de 1 a 5, representado a través de cinco estrellas.
- **R27.3:** La accesibilidad es un valor entero de 1 a 5, representado a través de cinco estrellas.

Obligatoriedad de los datos:

- **R27.4** Todos los datos indicados anteriormente son obligatorios para procesar la encuesta.

Lógica de la función:

- **R27.5** Los usuarios, al momento de responder la encuesta, deberán seleccionar la opción mediante selección de estrellas que más coincida con su opinión; se desplegará un formulario donde podrán evaluar la interfaz, la rapidez y la accesibilidad del sistema. Posterior a esto, se almacenarán las respuestas dentro de la base de datos.

Salida 1: satisfacción interfaz, satisfacción rapidez, satisfacción accesibilidad.

2.10. Asignación cuarentena

2.10.1. R28. Activación cuarentena.

Entradas: correo, cuarentena.

Restricciones de los datos:

- **R28.1:** EL correo deberá seguir la restricción de R12.4.
- **R28.2:** La cuarentena es un valor booleano (0 o 1).

Obligatoriedad de los datos:

- **R28.3** Todos los datos indicados anteriormente son obligatorios para procesar la asignación correcta de la cuarentena.

Lógica de la función:

- **R28.4** El administrador por orden del Ministerio de Salud presiona un botón llamado `.activar cuarentena`.^{el} cual cambia el valor de cuarentena una variable booleana de 0 a 1 y producto de este cambio se prohíbe el ingreso al establecimiento. Luego cuando el Ministerio de Salud de el comunicado de que es seguro reabrir se presionara nuevamente y la variable cuarentena retomara el valor 0.

Salida 1: correo, cuarentena.

2.11. Reporte contingencia vehicular en horas.

2.11.1. R29. Reporte contingencia vehicular en horas.

Entradas: id usuario, hora de ingreso.

Restricciones de los datos:

- **R29.1:** La hora ingreso debe estar definida por R22.1

Obligatoriedad de los datos:

- **R29.2** Todos los datos indicados anteriormente son obligatorios para procesar el reporte vehicular.

Lógica de la función:

- **R29.3** Al momento que van ingresando los usuarios al recinto, el gráfico se va actualizando; dependiendo la hora que ingrese, será representado en el gráfico de reporte.

Salida 1: Gráfico de R29.3

2.12. Desactivación de cuenta

2.12.1. R30. Desactivación de cuenta

Entradas: Correo, run, contraseña.

Restricciones de los datos:

- **R30.1:** EL correo deberá seguir la restricción de R12.4.
- **R30.2:** EL RUN deberá seguir la restricción de R12.1.
- **R30.3:** EL RUN deberá existir dentro de la base de datos.
- **R30.4:** La contraseña deberá seguir la restricción de R12.7.

Obligatoriedad de los datos:

- **R30.5** Todos los campos anteriores son de carácter obligatorio.

Lógica de la función:

- **R30.6** Al momento de ingresar al link del correo, se direccionara a una pagina en donde el usuario deberá de ingresar su RUN y contraseña para confirmar que quiere desactivar su cuenta.

Salida 1: Mensaje Cuenta Usuario Desactivada”. **Salida 2:** RUN, contraseña.

2.13. Personalización de interfaz

2.13.1. R31. Personalización de interfaz

Entradas: Tema actual, letra actual.

Lógica de la función:

- **R31.3** Al momento de presionar el botón para cambiar el tema este se pondrá en modo oscuro, al presionar de nuevo o recargar la pagina se colocara nuevamente el tema claro.
- **R31.4** En caso de presionar el botón de aumentar o disminuir la letra esta aumentara su tamaño o disminuirá dependiendo de la opción, en caso de refrescar la pagina volverá a su tamaño original.

Salida 1: Tema oscuro o tema claro de R31.3. **Salida 2:** Letra aumentada o disminuida de R31.4

2.14. Requisitos no funcionales

- El costo de mantenimiento estará sujeto a las modificaciones respectivas según la problemática.
- Las modificaciones tendrán un costo asociado dependiendo en qué etapa del proyecto se encuentre.

2.14.1. Atributos de calidad

- **Disponibilidad:** El sistema estará disponible 22 horas del día, teniendo en cuenta que a las 2 horas restantes serán de mantenimiento.
- **Portabilidad:** El software estará diseñado con programación modular, pruebas de compatibilidad cruzada.
- **Usabilidad:** Opciones para apoyar la accesibilidad en personas con discapacidades, como modificar el tamaño de las letras, el diseño de botones grandes y el contraste del fondo con el texto, además de sugerencias de posibles acciones a realizar.