Ministerio de hacienda

Manual Técnico Reserva de Cita

Sistema de Atención al Contribuyente (SAC)

Tabla de contenido

[1. Breve descripción del Módulo Reserva de Cita. 2](#_Toc433715232)

[2. Diagrama Entidad – Relación 3](#_Toc433715233)

[3. Estructura de Componentes (Capa vista, controlador, servicio DAO y estructura de reportes) 4](#_Toc433715234)

[Vistas o páginas JSP 4](#_Toc433715235)

[Controladores 5](#_Toc433715236)

[Descripción de Controladores. 6](#_Toc433715237)

[Clases Form 9](#_Toc433715238)

[Clases Service 9](#_Toc433715239)

[Descripción de Clases Service 9](#_Toc433715240)

[Clases DAO 12](#_Toc433715241)

[Descripción de Clases DAO. 12](#_Toc433715242)

[4. Documentos Relacionados a la Base de Datos 15](#_Toc433715243)

[Tablas 15](#_Toc433715244)

[5. Seguridad 20](#_Toc433715245)

[6. Anexos 21](#_Toc433715246)

# Breve descripción del Módulo Reserva de Cita.

Es el módulo que hace posible que un contribuyente pueda reservar una cita en cualquiera de los centros de servicios que posee el Ministerio de Hacienda al mismo tiempo proporciona un catálogo de servicios y tramites que permiten una mejor atención.

Las reservas de citas se realizan por medio de la web y se accede por medio del Sistema de Servicio al Contribuyente (SSC) y se encuentra específicamente en el módulo de Solicitudes y Administración.

Un contribuyente previamente registrado para los servicios de web del Ministerio de Hacienda ingresa con su usuario para reservar su cita en las fechas disponibles y según le convenga.

El personal de servicios al contribuyente tendrá acceso a la opción de reserva de cita a terceros, en la cual es posible reservar una cita para un contribuyente que lo solicite en los centros de servicio.

# Diagrama Entidad – Relación

A continuación se listan las diferentes tablas utilizadas en el módulo Reserva de Citas. La siguiente tabla proporciona el esquema al que pertenece y el nombre de cada una de las tablas.

|  |  |  |
| --- | --- | --- |
| Tabla | Esquema | Anexo |
| GC\_RESERVA\_CITA | GESTION.COLAS | Anexo 1 |
| GC\_CONF\_TRAMITE |
| GC\_TRAMITE |
| TB\_UNIDAD\_RECEP | CATALOGOS | Anexo 2 |
| RC\_RUC | RUC | Anexo 3 |

Tabla 2.1 Diagrama ER.

# Estructura de Componentes (Capa vista, controlador, servicio DAO y estructura de reportes)

## Vistas o páginas JSP

En la siguiente tabla se listan las páginas JavaServer Pages (JSP)

|  |  |  |
| --- | --- | --- |
| Nombre de opción | Ubicación | Nombre |
| Reserva de Cita | WEB-INF\views\rs\ | print.jsp  comprobante.jsp |
| WEB-INF\views\rs\personal\ | index.jsp |
| Reserva de Cita a Terceros | WEB-INF\views\rs\terceros\ | terceros.jsp  index.jsp |

Tabla 3.1 Vistas o paginas JSP

## Controladores

En la siguiente tabla se proporcionan los controladores utilizados en las opciones del módulo de Reserva de Citas. Estos controladores se encuentran almacenados en el siguiente paquete:

**sv.gob.mh.dgii.ssc.rs.controller**

|  |  |
| --- | --- |
| Nombre de opción | Controlador |
| Reserva de Cita | ReservaCitaController.java |
| Reserva de Cita a Terceros | ReservaCitaTercerosController.java |
| -- | ReservaCitaUtility.java |

Tabla 3.2 Lista de Controladores

## Descripción de Controladores.

A continuación se proporciona una descripción de la funcionalidad principal de los controladores utilizados en el módulo de Reserva de Cita.

| **PANTALLA: RESERVA DE CITA** | |
| --- | --- |
| **Controlador:** | **ReservaCitaController.java** |
| **Descripción de Métodos de Negocio** | |
| **@ModelAttribute("reservaCitaForm")**  **public ReservaCitaForm createModel():** Método que se encarga de retorna hacia el form ReservaCitaForm.  **@RequestMapping("/comprobante/")**  **public String comprobante(HttpServletRequest request):** Método que proporciona el pantalla de comprobante de la reservación de cita, se carga por medio de la url: "rs/comprobante" y se usa la página comprobante.jsp  **@RequestMapping(value = "/", method = RequestMethod.GET)**  **public String index(Model model, HttpServletRequest request):** Método que proporciona la pantalla de reserva de citas con los componentes y parámetros iniciales. Utiliza el form ReservaCitaForm para mostrar el formulario para reservar cita. El formulario presenta por medio del servicio reservaCitaService los centros de servicio junto con los servicios disponibles para la configuración de la reserva.  **@RequestMapping(value="/tramites/")**  **public @ResponseBody ReservaCitaForm getTramites(Model model, HttpServletRequest request,HttpServletResponse response,@RequestParam(value="cUnidadRecep",required=true) String cUnidadRecep,@RequestParam(value="nServiciosId",required=true) Long nServiciosId):** Método que se utiliza para proporcionar los tramites disponibles para poder ser seleccionado en el proceso de reserva de cita. Se establecen las validaciones correspondientes para que el form no este vacion y que el usuario seleccione el tramite a realizar. Los tramites son cargado usando el servicio reservaCitaService y su método listTramites()  **@RequestMapping(value="/saveReserva/")**  **public @ResponseBody GcReservaCita saveReserva(Model model, HttpServletRequest request, HttpServletResponse response, @ModelAttribute(value = "reservaCitaForm") ReservaCitaForm reservaCitaForm):** Método que se encarga de guardar la reserva de cita, se invoca al método save() para este proceso. En este método se captura la configuración de la reserva de cita por medio del form reservaCitaForm y se setean los parámetros correspondientes. Posteriormente se proporcionan los parámetros para determinar el tiquete o comprobante de la reserva.  **@RequestMapping(value="/time/")**  **public @ResponseBody Map<String, Object> getActualTime():** Método que se usa para determinar la hora actual en la que se realiza la reserva de cita.  **@RequestMapping(value="/validateDay/")**  **public @ResponseBody Map<String, Object> validateDay(@RequestBody Map<String, Object> map):** Método que se encarga de validar la fecha seleccionada, es decir que se encuentre en los días de lunes a viernes y que no sea asueto. | |

Tabla 3.3 Controlador Reserva de Cita

| **PANTALLA: RESERVA DE CITA A TERCEROS** | |
| --- | --- |
| **Controlador:** | **ReservaCitaTercerosController.java** |
| **Descripción de Métodos de Negocio** | |
| **@ModelAttribute("reservaCitaForm")**  **public ReservaCitaForm createModel():** Método que se encarga de retorna hacia el form ReservaCitaForm.  **@RequestMapping("/comprobante/")**  **public String comprobante(HttpServletRequest request):** Método que proporciona el pantalla de comprobante de la reservación de cita, se carga por medio de la url: "rs/comprobante" y se usa la página comprobante.jsp  **@RequestMapping(value = "/", method = RequestMethod.GET)**  **public String indexTerceros(HttpServletRequest request):** Método que retorna la url: "rs/terceros/index" y carga la pagina index.jsp  **@RequestMapping(value = "/home", method = RequestMethod.GET)**  **public String terceros(Model model, HttpServletRequest request):** Método que proporciona la pantalla de reserva de citas con los componentes y parámetros iniciales. Utiliza el form ReservaCitaForm para mostrar el formulario para reservar cita. El formulario presenta por medio del servicio reservaCitaService los centros de servicio junto con los servicios disponibles para la configuración de la reserva. Se retorna la url: "rs/terceros/terceros" y se carga la pagina terceros.jsp  **@RequestMapping(value="/tramites/")**  **public @ResponseBody ReservaCitaForm getTramites(Model model, HttpServletRequest request,HttpServletResponse response,@RequestParam(value="cUnidadRecep",required=true) String cUnidadRecep,@RequestParam(value="nServiciosId",required=true) Long nServiciosId):** Método que se utiliza para proporcionar los tramites disponibles para poder ser seleccionado en el proceso de reserva de cita. Se establecen las validaciones correspondientes para que el form no se encuentre vacío y que el usuario seleccione el tramite a realizar. Los tramites son cargado usando el servicio reservaCitaService y su método listTramites().  **@RequestMapping(value="/saveReserva/")**  **public @ResponseBody GcReservaCita saveReserva(Model model, HttpServletRequest request, HttpServletResponse response, @ModelAttribute(value = "reservaCitaForm") ReservaCitaForm reservaCitaForm):** Método que se encarga de guardar la reserva de cita, se invoca al método save() para este proceso. En este método se captura la configuración de la reserva de cita por medio del form reservaCitaForm y se setean los parámetros correspondientes. Posteriormente se proporcionan los parámetros para determinar el tiquete o comprobante de la reserva.  **@RequestMapping(value="/validateNIT/")**  **public @ResponseBody Map<String, Object> validateNIT(Model model,@RequestParam(value="nit",required=true) String nit):** Método que valida si el NIT ingresado es válido, se usa el servicio rcRucService y su findByNit() para la búsqueda del NIT. Si e valido proporciona los datos del contribuyente en la pantalla de reserva de cita a terceros.  **@RequestMapping(value="/time/")**  **public @ResponseBody Map<String, Object> getActualTime():** Método que se usa para determinar la hora actual en la que se realiza la reserva de cita.  **@RequestMapping(value="/validateDay/")**  **public @ResponseBody Map<String, Object> validateDay(@RequestBody Map<String, Object> map):** Método que se encarga de validar la fecha seleccionada, es decir que se encuentre en los días de lunes a viernes y que no sea asueto. | |

Tabla 3.4 Controlador Reserva de Cita Terceros

|  |  |
| --- | --- |
| **PANTALLA: RESERVA DE CITA (Utilities)** | |
| **Controlador:** | **ReservaCitaUtility.java** |
| **Descripción de Métodos de Negocio** | |
| **public GcReservaCita save(ReservaCitaForm reservaCitaForm, TbUnidadRecep tbUnidadRecep):** Método que guarda la reserva de cita, crea las instancias GcReservaCita y GcTramite para proporcionarles los parámetros correspondientes a la reserva de cita configurada en el form ReservaCitaForm y el tramite seleccionado. En este proceso verifica el nit del tercero y se setean los datos y variables para que sean guardados.  **public void enviarCorreos(ReservaCitaForm form, GcReservaCita cita):** Método que se usa para el envio del correo de confirmación de la reserva de cita en línea. Usa el servicio reservaCitaService y el método saveCorreo()  **public String quitarGuionesNIT(String nit):** Método que se usa para la eliminación de guiones dentro del parámetro nit. | |

Tabla 3.5 Controlador Reserva de Cita (Utilities)

## Clases Form

Listado de las clases Form utilizadas en el Módulo de Reserva de Citas.

| Nombre de opción | Ubicación | Service |
| --- | --- | --- |
| Reserva de Citas | sv.gob.mh.dgii.ssc.rs.form | ReservaCitaForm.java |
| Reserva de Citas Terceros | ReservaCitaForm.java |

TABLA 3.6 Clases Form

## Clases Service

Listado de las clases Service utilizadas en el Módulo de Reserva de Citas.

| Nombre de opción | Ubicación | Service |
| --- | --- | --- |
| Reserva de Citas | sv.gob.mh.dgii.service.rc  sv.gob.mh.dgii.service.rs | RcRucService  ReservaCitaService |
| Reserva de Citas Terceros | sv.gob.mh.dgii.service.rc  sv.gob.mh.dgii.service.rs | RcRucService  ReservaCitaService |

TABLA 3.7 Listado de los Service y su ubicación.

## Descripción de Clases Service

A continuación se describen las clases service utilizadas en este módulo.

| Opción | Clase Service | Descripción del Services |
| --- | --- | --- |
| Reserva de Citas | ReservaCitaService.java  ReservaCitaServiceImpl.java | El Services implementa los métodos siguientes:  **public List<TbUnidadRecep> listUnidadRecep():** Metodo encargado de devolver listUnidadRecep invocándolo usando el DAO reservaCitaDao.  **public List<GcServicios> listServicios():** Metodo encargado de devolver listServicios invocándolo usando el DAO reservaCitaDao.  **public List<GcConfTramite> listTramites(Long nServiciosId):** Metodo encargado de devolver listTramites invocándolo usando el DAO reservaCitaDao.  **public TbUnidadRecep getUnidadRecep(String unidadRecep):** Metodo encargado de devolver getUnidadRecep usando el DAO reservaCitaDao para invocarlo.  **public Integer getValidDay(Date fecha):** Metodo encargado de devolver getValidDay utilizando el DAO reservaCitaDao para invocarlo.  **public String getCorrelativo(String unidadRecep):** Metodo encargado de devolver getCorrelativo utilizando el DAO reservaCitaDao para invocarlo.  **public void saveReservaCita(GcReservaCita reservaCita):** Metodo encargado de devolver saveReservaCita utilizando el DAO reservaCitaDao para invocarlo.  **public GcReservaCita getReservaCita():** Metodo encargado de devolver getReservaCita utilizando el DAO reservaCitaDao para invocarlo.  **public void saveCorreo(String from, String to, String body, String subject, String nombre):** Metodo encargado de devolver saveCorreo utilizando el DAO reservaCitaDao para invocarlo. |
| Reserva de Cita Terceros | rcRucService.java  rcRucServiceImpl.java | El service implementa los métodos siguientes:  **public RcRuc findByNit(String nit):** Metodo encargado de devolver findByNit usando el DAO rcRucDao para invocarlo. |
|  | El Services implementa los métodos siguientes:  **public List<TbUnidadRecep> listUnidadRecep():** Metodo encargado de devolver listUnidadRecep invocándolo usando el DAO reservaCitaDao.  **public List<GcServicios> listServicios():** Metodo encargado de devolver listServicios invocándolo usando el DAO reservaCitaDao.  **public List<GcConfTramite> listTramites(Long nServiciosId):** Metodo encargado de devolver listTramites invocándolo usando el DAO reservaCitaDao.  **public TbUnidadRecep getUnidadRecep(String unidadRecep):** Metodo encargado de devolver getUnidadRecep usando el DAO reservaCitaDao para invocarlo.  **public Integer getValidDay(Date fecha):** Metodo encargado de devolver getValidDay utilizando el DAO reservaCitaDao para invocarlo.  **public String getCorrelativo(String unidadRecep):** Metodo encargado de devolver getCorrelativo utilizando el DAO reservaCitaDao para invocarlo.  **public void saveReservaCita(GcReservaCita reservaCita):** Metodo encargado de devolver saveReservaCita utilizando el DAO reservaCitaDao para invocarlo.  **public GcReservaCita getReservaCita():** Metodo encargado de devolver getReservaCita utilizando el DAO reservaCitaDao para invocarlo.  **public void saveCorreo(String from, String to, String body, String subject, String nombre):** Metodo encargado de devolver saveCorreo utilizando el DAO reservaCitaDao para invocarlo. |

Tabla 3.8 Descripción de los Services.

## Clases DAO

En la siguiente tabla se listan las diferentes clases DAO que se usan en el Modulo de Reserva de Citas y los diferentes paquetes donde se pueden encontrar.

| Nombre de opción | Ubicación | Dao |
| --- | --- | --- |
| Reserva de Citas | sv.gob.mh.dgii.dao.rc  sv.gob.mh.dgii.dao.rs | rcRucDao.java  rcRucDaoImpl.java  reservaCitaDao.java  reservaCitaDaoImpl.java |
| Reserva de Citas Terceros | sv.gob.mh.dgii.dao.rc  sv.gob.mh.dgii.dao.rs | rcRucDao.java  rcRucDaoImpl.java  reservaCitaDao.java  reservaCitaDaoImpl.java |

Tabla 3.9 Lista de Dao y su ubicación.

## Descripción de Clases DAO.

En la siguiente tabla se detallan los métodos declarados en las clases DAO usados en las opciones de este módulo.

| Opción | Clase Dao | Descripción |
| --- | --- | --- |
| Reserva de Citas | ReservaCitaDao.java  ReservaCitaDaoImpl.java | El Dao implementa los métodos siguientes:  **public List<TbUnidadRecep> listUnidadRecep():** Encargado de ejecutar el query SQL "SELECT tb FROM TbUnidadRecep tb WHERE tb.mservicio IN('C','M') ORDER BY tb.cunidadRecep ASC ".  **public List<GcServicios> listServicios():** Metodo encargado de ejecutar el query "SELECT s FROM GcServicios s WHERE s.bActiva = 1 ORDER BY s.nServiciosId ASC ".  **public List<GcConfTramite> listTramites(Long nServiciosId):** Metodo encargado de ejecutar el query "SELECT t FROM GcConfTramite t WHERE t.nTramiteId.nServiciosId.nServiciosId = :nServiciosId ORDER BY t.nTramiteId.nTramiteId ASC ".  **public TbUnidadRecep getUnidadRecep(String unidadRecep):** Metodo encargado de ejecutar el query "SELECT tb FROM TbUnidadRecep tb WHERE tb.mservicio IN('C','M') AND tb.cunidadRecep = :unidadRecep ".  **public Integer getValidDay(Date fecha):** Metodo encargado de ejecutar el query "SELECT COUNT(\*) FROM TbAsueto t WHERE t.fasueto=:fecha "  **public String getCorrelativo(String unidadRecep):** Metodo encargado de ejecutar el query "SELECT TO\_CHAR(MAX(TO\_NUMBER(REGEXP\_REPLACE(NVL(rc.sCodVerifica,0),'[^[:digit:]]')))+1) FROM GcReservaCita rc WHERE rc.cUnidadRecep.cunidadRecep =:unidadRecep ".  **public void saveReservaCita(GcReservaCita reservaCita):** Metodo encargado de invocar al método save a través de getCurrentSession y getSessionFactory.  **public GcReservaCita getReservaCita():** Metodo encargado de ejecutar el query "SELECT rs FROM GcReservaCita rs WHERE rs.nReservaCitaId = (SELECT MAX(s.nReservaCitaId) FROM GcReservaCita s) ".  **public void saveCorreo(Object[] params):** Metodo encargado de ejecutar el query para invocar al procedimiento almacenado "{ call pkg\_siit\_functions.sendemail( ?, '', ? , replace( replace( ? , '~', chr(13) ), '^', chr(10) ), NULL , pkg\_siit\_functions.con\_smtp\_address , ?, ? ) }". |
| Reserva de Citas Terceros | ReservaCitaDao.java  ReservaCitaDaoImpl.java | El Dao implementa los métodos siguientes:  **public List<TbUnidadRecep> listUnidadRecep():** Encargado de ejecutar el query SQL "SELECT tb FROM TbUnidadRecep tb WHERE tb.mservicio IN('C','M') ORDER BY tb.cunidadRecep ASC ".  **public List<GcServicios> listServicios():** Metodo encargado de ejecutar el query "SELECT s FROM GcServicios s WHERE s.bActiva = 1 ORDER BY s.nServiciosId ASC ".  **public List<GcConfTramite> listTramites(Long nServiciosId):** Metodo encargado de ejecutar el query "SELECT t FROM GcConfTramite t WHERE t.nTramiteId.nServiciosId.nServiciosId = :nServiciosId ORDER BY t.nTramiteId.nTramiteId ASC ".  **public TbUnidadRecep getUnidadRecep(String unidadRecep):** Metodo encargado de ejecutar el query "SELECT tb FROM TbUnidadRecep tb WHERE tb.mservicio IN('C','M') AND tb.cunidadRecep = :unidadRecep ".  **public Integer getValidDay(Date fecha):** Metodo encargado de ejecutar el query "SELECT COUNT(\*) FROM TbAsueto t WHERE t.fasueto=:fecha "  **public String getCorrelativo(String unidadRecep):** Metodo encargado de ejecutar el query "SELECT TO\_CHAR(MAX(TO\_NUMBER(REGEXP\_REPLACE(NVL(rc.sCodVerifica,0),'[^[:digit:]]')))+1) FROM GcReservaCita rc WHERE rc.cUnidadRecep.cunidadRecep =:unidadRecep ".  **public void saveReservaCita(GcReservaCita reservaCita):** Metodo encargado de invocar al método save a través de getCurrentSession y getSessionFactory.  **public GcReservaCita getReservaCita():** Metodo encargado de ejecutar el query "SELECT rs FROM GcReservaCita rs WHERE rs.nReservaCitaId = (SELECT MAX(s.nReservaCitaId) FROM GcReservaCita s) ".  **public void saveCorreo(Object[] params):** Metodo encargado de ejecutar el query para invocar al procedimiento almacenado "{ call pkg\_siit\_functions.sendemail( ?, '', ? , replace( replace( ? , '~', chr(13) ), '^', chr(10) ), NULL , pkg\_siit\_functions.con\_smtp\_address , ?, ? ) }". |

Tabla 3.10 Clases Dao.

# Documentos Relacionados a la Base de Datos

## Tablas

En las siguientes páginas se describen cada uno de los campos de las tablas de la base de datos. **La descripción del campo está sujeta a los comentarios agregados en la base de datos.**

| GC\_CONF\_TRAMITE | | | |
| --- | --- | --- | --- |
| Secuencia: SEQ\_GC\_CONF\_TRAMITE | | | |
| Nombre del Atributo | Tipo de Dato | Null | Descripción |
| N\_CONF\_TRA\_ID | NUMBER | No | Llave primaria de la tabla |
| C\_UNIDAD\_RECEP | VARCHAR2(5 BYTE) | No | Código del centro de servicio |
| N\_TRAMITE\_ID | NUMBER | No | Referencia al trámite a que se refiere la configuración |
| N\_ATENCION\_PROM | NUMBER | No | Tiempo promedio de atención |
| N\_TIEMPO\_ESPERA | NUMBER | Yes | Tiempo, en minutos, de espera máximo que debe tardarse en la cola |
| N\_TIEMPO\_HOLGURA | NUMBER | Yes | Tiempo, en minutos, que debe esperarse antes de incluirlo en la cola |
| C\_USUARIO\_CREA | VARCHAR2(100 BYTE) | No | Código del usuario que crea el registro |
| C\_USUARIO\_MODI | VARCHAR2(100 BYTE) | No | Código del usuario que modifica el registro |
| FI\_VIGENCIA | DATE | No | Fecha en que el registro es creado |
| FF\_VIGENCIA | DATE | Yes | Fecha en que el registro deja de tener vigencia |
| F\_MODIFICA | DATE | Yes | Fecha en que el registro fue modificado |
| N\_COMPORTAMIENTO | NUMBER(6,0) | Yes | Comportamiento: 0-al final, 1-al inicio, 2- en medio, 3 por peso |
| N\_PESO | NUMBER(6,0) | Yes | peso de la reasignación |
| N\_PROM\_ESPERA | NUMBER | Yes | Promedio real (en segundos) de espera en cola |
| N\_PROM\_ATENCION | NUMBER | Yes | Promedio real (en segundos) de atención en escritorio |

| GC\_RESERVA\_CITA | | | |
| --- | --- | --- | --- |
| Secuencia: NO | | | |
| Nombre del Atributo | Tipo de Dato | Null | Descripción |
| N\_RESERVA\_CITA\_ID | NUMBER | No | Llave primaria de la tabla |
| C\_UNIDAD\_RECEP | VARCHAR2(5 BYTE) | No | Código del centro de servicio |
| N\_TRAMITE\_ID | NUMBER | No | Referencia al trámite que está solicitando la reserva de cita |
| FH\_RESERVACION | DATE | No | Fecha y hora en que se llevará a cabo la cita programada |
| S\_CORREO | VARCHAR2(256 BYTE) | Yes | Correo electrónico del contribuyente que hace la cita |
| NIT | VARCHAR2(14 BYTE) | Yes | Nit del contribuyente |
| N\_TELEFONO | VARCHAR2(25 BYTE) | Yes | Teléfono de contacto del contribuyente |
| S\_OBSERVACIONES | VARCHAR2(1024 BYTE) | Yes | Observaciones de la reserva de cita. |
| S\_COD\_VERIFICA | VARCHAR2(50 BYTE) | Yes | Código de verificación de la cita |
| N\_TIPO\_RESERVA | NUMBER | Yes | Forma en que fue hecha la reserva de cita |
| NIT\_TERCERO | VARCHAR2(14 BYTE) | Yes | Nit de la persona que hizo la cita en nombre de un tercero |
| B\_ESTADO | NUMBER(1,0) | No | Estado en que se encuentra la reserva de cita |

| GC\_TRAMITE | | | |
| --- | --- | --- | --- |
| Secuencia: NO | | | |
| Nombre del Atributo | Tipo de Dato | Null | Descripción |
| N\_TRAMITE\_ID | NUMBER | No | Llave primaria de la tabla |
| N\_SERVICIOS\_ID | NUMBER | No | Referencia al servicio a que pertenece el trámite |
| N\_PESO | NUMBER | No | Cuantificación de la prioridad que debe tener el tramite luego de aplicarle todos los criterios |
| B\_NIT\_REQUERIDO | NUMBER(1,0) | No | Bandera que indica si es obligatorio proporionar el nit |
| B\_ESCALAMIENTO | NUMBER(1,0) | No | Bandera que indica si el tramite será escalado al supervisor |
| D\_TRAMITE | VARCHAR2(256 BYTE) | Yes | Descripción del registro |
| C\_USUARIO\_CREA | VARCHAR2(100 BYTE) | No | Código del usuarioque crea el registro |
| C\_USUARIO\_MODI | VARCHAR2(100 BYTE) | No | Código del usuarioque modifica el registro |
| FI\_VIGENCIA | DATE | No | Fecha en que el registro es creado |
| FF\_VIGENCIA | DATE | Yes | Fecha en que el registro deja de tener vigencia |
| F\_MODIFICA | DATE | Yes | Fecha en que el registro fue modificado |
| B\_ACTIVA | NUMBER(1,0) | No | Bandera que indica si el registro esta activo o no |
| S\_NOMBRE | VARCHAR2(256 BYTE) | No | Nombre del tramite |
| N\_ORDEN | NUMBER | Yes | Orden en que se presentaran los tramites en pantalla |

| TB\_UNIDAD\_RECEP | | | |
| --- | --- | --- | --- |
| Secuencia: NO | | | |
| Nombre del Atributo | Tipo de Dato | Null | Descripción |
| C\_UNIDAD\_RECEP | VARCHAR2(5 BYTE) | No | Código del centro de servicio |
| C\_USUARIO | VARCHAR2(30 BYTE) | No | Usuario |
| D\_UNIDAD\_RECEP | VARCHAR2(40 BYTE) | No | Descripción |
| FH\_INGRESO | DATE | Yes | Fecha de ingreso |
| B\_STATUS | NUMBER(1,0) | No | Estado |
| M\_TIPO\_UNIDAD | VARCHAR2(1 BYTE) | No | Tipo de unidad |
| B\_DESPLEGABLE | NUMBER(1,0) | No | Se muestra 1 o 0 |
| C\_UNIDAD | VARCHAR2(1 BYTE) | Yes | Unidad asignada |
| C\_UNIDAD\_DGT | VARCHAR2(25 BYTE) | Yes | Unidad DGT |
| C\_DEP\_MUN | VARCHAR2(4 BYTE) | Yes | Código de departamento |
| S\_UBIC\_GEOGRAF | VARCHAR2(60 BYTE) | Yes | Descripción de la ubicación geográfica. |
| C\_UNIDAD\_RECEP\_SUP | VARCHAR2(5 BYTE) | Yes |  |
| M\_SERVICIO | VARCHAR2(1 BYTE) | Yes |  |

| RC\_RUC | | | |
| --- | --- | --- | --- |
| Secuencia: | | | |
| Nombre del Atributo | Tipo de Dato | Null | Descripción |
| NIT | VARCHAR2(14 BYTE) | No | número de identificación tributaria |
| C\_PAIS | VARCHAR2(4 BYTE) | No | código de país |
| C\_DEPARTAMENTO | VARCHAR2(2 BYTE) | No | código de departamento |
| C\_MUNICIPIO | VARCHAR2(2 BYTE) | No | código de municipio |
| C\_ADM\_TRIB | VARCHAR2(2 BYTE) | No | código de administración tributaria |
| C\_TERMINO | VARCHAR2(3 BYTE) | No | código de termino |
| C\_IMPORTANCIA | VARCHAR2(1 BYTE) | No | código de importancia |
| C\_CLASE | VARCHAR2(1 BYTE) | No | código de clase de contribuyente |
| C\_TIPO | VARCHAR2(4 BYTE) | No | código de tipo |
| S\_1APE\_RASOC | VARCHAR2(100 BYTE) | No | primer apellido o razón social de la empresa |
| S\_2APE\_ABREV | VARCHAR2(100 BYTE) | Yes | segundo apellido |
| S\_NOMBRES | VARCHAR2(40 BYTE) | Yes | nombres del contribuyente |
| F\_NAC\_CONST | DATE | No | fecha de nacimiento o constitución |
| B\_DOMICILIADO | NUMBER(1,0) | No | es domiciliado o no |
| F\_TERMINO | DATE | Yes | fecha de término del NIT |
| B\_INTERES\_FISCAL | NUMBER(1,0) | No | tiene interés fiscal |
| M\_SEXO | VARCHAR2(1 BYTE) | Yes | tipo de sexo |
| B\_IMPORTADOR | NUMBER(1,0) | No | es importador |
| V\_CAPITAL\_SOC | NUMBER(13,2) | Yes | valor de capital social |
| B\_ACTIVO | VARCHAR2(1 BYTE) | No | esta activo |
| B\_ALERTA | NUMBER(1,0) | Yes |  |

# Seguridad

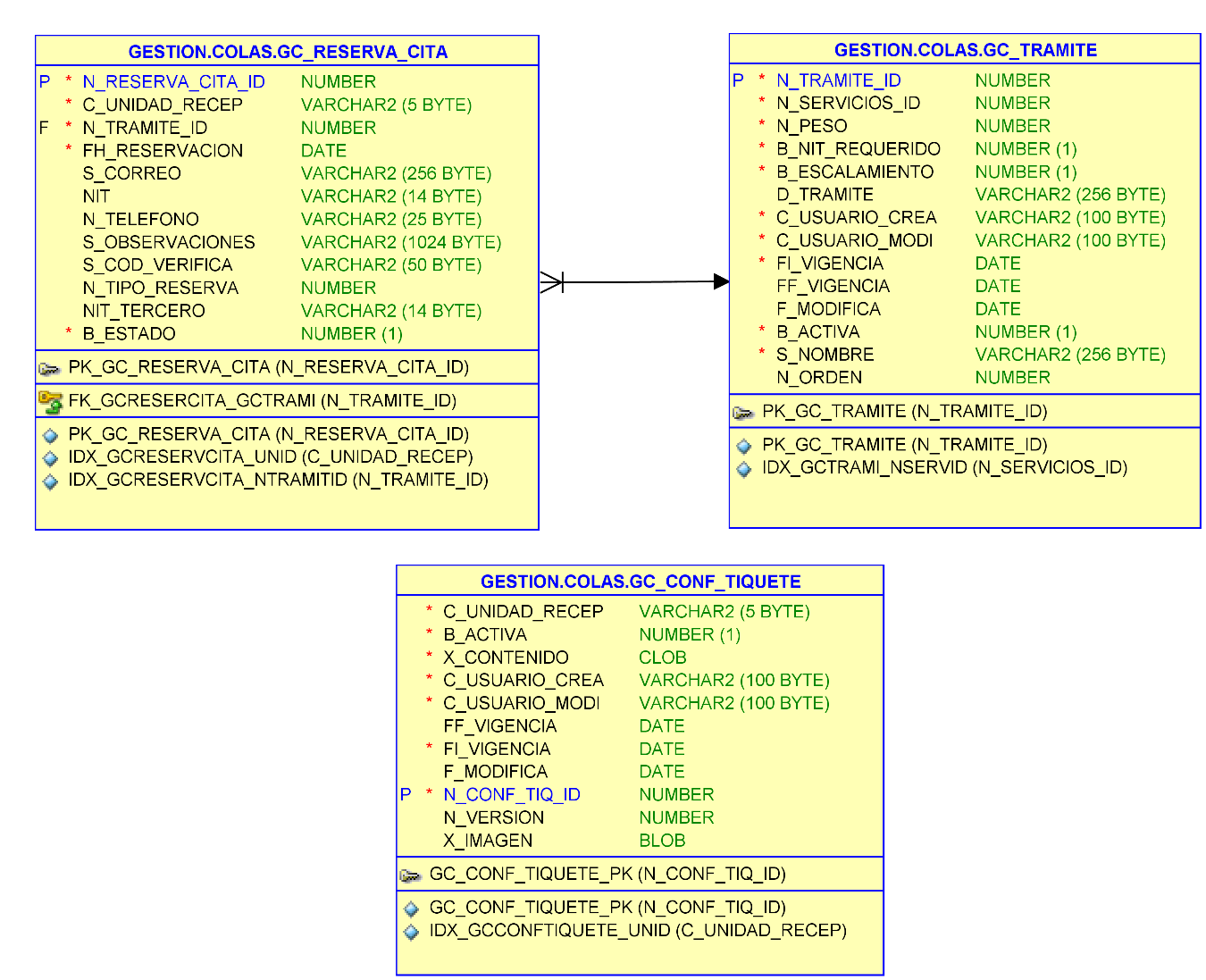
Estos son los roles de seguridad usados en el Módulo Reserva de Cita:

| Opción | URL | Roles | | |
| --- | --- | --- | --- | --- |
| Producción | | Desarrollo |
| Reserva de Citas | /servicios/cita/personal/ | ROLE\_ST\_R99 | ROLE\_ST\_R99  ROLE\_SW\_R9 | |
| Reserva de Citas Terceros | /servicios/cita/terceros/ | ROLE\_ST\_R99 | ROLE\_ST\_R99  ROLE\_SW\_R9 | |

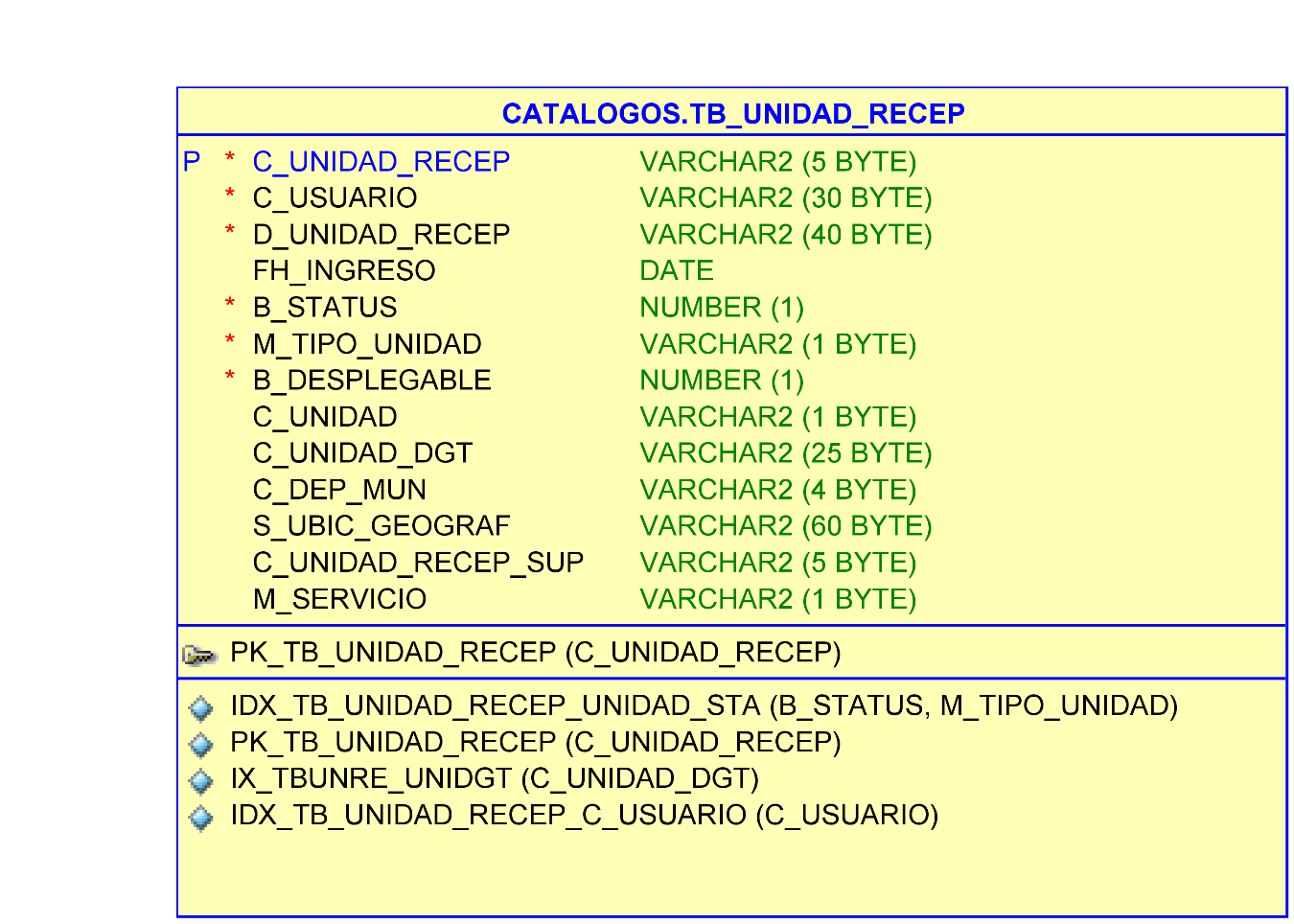
Tabla 5.1 Roles de Seguridad

# Anexos

Anexo 1:



Anexo 2:



Anexo 3:

