UNIVERSIDAD AUTÓNOMA GABRIEL RENE MORENO

FACULTAD DE INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN Y TELECOMUNICACIONES



SISTEMA DE GESTIÓN DE AUXILIARES PARA LA FACULTAD DE INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN Y TELECOMUNICACIONES

Alumnos

* Duarte Barrientos Rodrigo 218017243
* Mercado Oudalova Erick Denis 216032385
* Cruz Garnica Alejandro 218015372

Materia: Tecnología Web INF513-SA

Docente: Ing. Balcázar Veizaga Evans

Fecha: 22/12/2022

Santa Cruz – Bolivia

Contenido

[1. Perfil 4](#_Toc122585792)

[1.1. INTRODUCCIÓN 4](#_Toc122585793)

[1.2. ANTECEDENTES 4](#_Toc122585794)

[1.2.1. ANTECEDENTES DE AUXILIARES UAGRM 4](#_Toc122585795)

[1.3. SÍNTESIS HISTÓRICA 5](#_Toc122585796)

[1.4. JUSTIFICACIÓN 5](#_Toc122585797)

[1.5. MISIÓN 5](#_Toc122585798)

[1.6. VISIÓN 6](#_Toc122585799)

[1.7. OBJETIVOS 6](#_Toc122585800)

[1.7.1. OBJETIVO GENERAL 6](#_Toc122585801)

[1.7.2. OBJETIVOS ESPECÍFICOS 6](#_Toc122585802)

[1.8. DESCRIPCIÓN DEL PROBLEMA 6](#_Toc122585803)

[1.9. ALCANCE 6](#_Toc122585804)

[1.1. ELEMENTOS DEL SISTEMA 7](#_Toc122585805)

[1.1.1. HARDWARE 7](#_Toc122585806)

[1.1.2. SOFTWARE 7](#_Toc122585807)

[1.1.3. DATOS 7](#_Toc122585808)

[1.1.4. PERSONAS 7](#_Toc122585809)

[1.1.5. PROCESOS 7](#_Toc122585810)

[1.2. TECNOLOGÍAS PARA EL DESARROLLO DE SOFTWARE 8](#_Toc122585811)

[1.2.1. SISTEMA OPERATIVO 8](#_Toc122585812)

[1.2.2. SISTEMA DE GESTIÓN DE BASE DE DATOS 8](#_Toc122585813)

[1.2.3. HERRAMIENTAS CREATELY 9](#_Toc122585814)

[1.3. ENTORNO DE DESARROLLO 9](#_Toc122585815)

[2. MARCO TEÓRICO 9](#_Toc122585816)

[2.1. PROTOCOLO TCP/IP 10](#_Toc122585817)

[2.2. CORREO ELECTRÓNICO 11](#_Toc122585818)

[2.2.1. FUNDAMENTO DEL CORREO ELECTRONICO 11](#_Toc122585819)

[2.3. PROTOCOLO SMTP 12](#_Toc122585820)

[2.3.1. MODO DE COMUNICACIÓN SMTP 13](#_Toc122585821)

[2.4. MIME 15](#_Toc122585822)

[2.4.1. MIME HEADERS 16](#_Toc122585823)

[2.5. PROTOCOLO POP3 16](#_Toc122585824)

[2.6. SENDMAIL 17](#_Toc122585825)

[3. REQUISITOS 20](#_Toc122585826)

[3.1. REQUISITOS DEL SISTEMA 20](#_Toc122585827)

[3.1.1. REQUERIMIENTOS FUNCIONALES 20](#_Toc122585828)

[3.2. IDENTIFICACIÓN DE ACTORES Y CASOS DE USO 20](#_Toc122585829)

[3.2.1. IDENTIFICACIÓN DE ACTORES Y ROLES 20](#_Toc122585830)

[3.2.2. LISTA DE CASOS DE USO 20](#_Toc122585831)

[3.3. PRIORIZACIÓN DE CASOS DE USO 20](#_Toc122585832)

[3.4. DETALLE DE LOS CASOS DE USO 21](#_Toc122585833)

[4. FLUJO DE TRABAJO: ANÁLISIS 31](#_Toc122585834)

[4.1. ANÁLISIS DE ARQUITECTURA 31](#_Toc122585835)

[4.1.1. IDENTIFICAR PAQUETE 31](#_Toc122585836)

[4.2. ANÁLISIS DE CASOS DE USO 31](#_Toc122585837)

[4.2.1. DIAGRAMA DE COMUNICACIÓN 31](#_Toc122585838)

[ CU2: Gestionar Periodo 31](#_Toc122585839)

[ CU4: Gestionar Materias 31](#_Toc122585840)

[ CU5: Gestionar Postulantes 32](#_Toc122585841)

[ CU6: Gestionar Evaluaciones 32](#_Toc122585842)

[ CU8: Gestionar Reportes y Informes 32](#_Toc122585843)

[5. Flujo de trabajo: Diseño 36](#_Toc122585844)

[5.1 Diseño de la Arquitectura 36](#_Toc122585845)

[5.2 Diseño de Casos de Uso 37](#_Toc122585846)

[6. Diseño de Datos 45](#_Toc122585847)

[6.1 Diseño Conceptual 45](#_Toc122585848)

[6.1.1 Diagrama de Clases 45](#_Toc122585849)

[6.2. DISEÑO LÓGICO 46](#_Toc122585850)

[6.2.1. MAPEO 46](#_Toc122585851)

[6.3. DISEÑO FÍSICO 47](#_Toc122585852)

[6.3.1. TABLA DE VOLUMEN 47](#_Toc122585853)

[6.3.2. Script 56](#_Toc122585854)

[CONCLUSIÓN 70](#_Toc122585855)

[RECOMENDACIÓN 70](#_Toc122585856)

# Perfil

## INTRODUCCIÓN

Actualmente nuestras generaciones se viven en constante comunicaciones entre nosotros. Nos comunicamos día y noche, todo esto inicio con el uso del correo electrónico el cual está muy presente en el día a día. Solo hace falta ver la facilidad con la que uno puede dialogar con otra persona con tan solo un mensaje, pero ahora nos queda saber, ¿Cómo lo hace? ¿Cómo funciona?, si bien su uso cotidiano nos ha llevado a una normalización automática ha hecho que nuestra curiosidad haya ido disminuyendo. En este proyecto el cual está encaminado a uno más grande, nos permitirá conocer el proceso por el cual un correo es enviado, leído, procesado y devuelto con alguna data de confirmación

## ANTECEDENTES

Los orígenes del correo electrónico son anteriores al Internet, y datan del año 1961 cuando el Massachusetts Institute of Technology, se hizo a un computador modelo IBM 7090 de tiempo compartido, el cual se prestaba para que varios usuarios iniciaran sesión desde terminales remotas y el almacenamiento de información en el disco del equipo. Inicialmente ese sistema comenzó a ser utilizado de manera informal para el intercambio de mensajes, sin embargo, en el año 1965 se creó el servicio MAIL, el cual era mucho más eficiente y permitía el envío de mensajes entre usuarios de ese equipo de computación.

El primer mensaje de correo electrónico enviado a través de la red se hizo a través de la red ARPANET en el año 1971, y tenía solo contenido de texto; la particularidad de este envió fue que se hizo con dos computadoras, una junto a la otra. Gracias ese tipo de experimentos, empezaron a surgir ideas sobre el correo electrónico, y fue Ray Tomlinson el encargado de experimentar con el protocolo CYPNET para enviar mensajes por medio de red, que en ese tiempo solo permitían comunicación entre los usuarios de un mismo computador.

En esa misma época, Ray Tomlinson incorporo el uso de la arroba (@), como elemento de división entre el nombre del usuario y la computadora en que estaba la cuenta del usuario que oficia como destinatario. Con los sistemas anteriores no era necesario especificar la máquina de destino porque toda la información que se transmitía era de manera local; pero con ese nuevo sistema era necesario distinguir el correo local del correo de red.

Fue entonces hasta el año 1977 cuando el correo electrónico se volvió un servicio de red estandarizado, gracias a la culminación de diversos estándares parciales por parte de la especificación RFC 733.

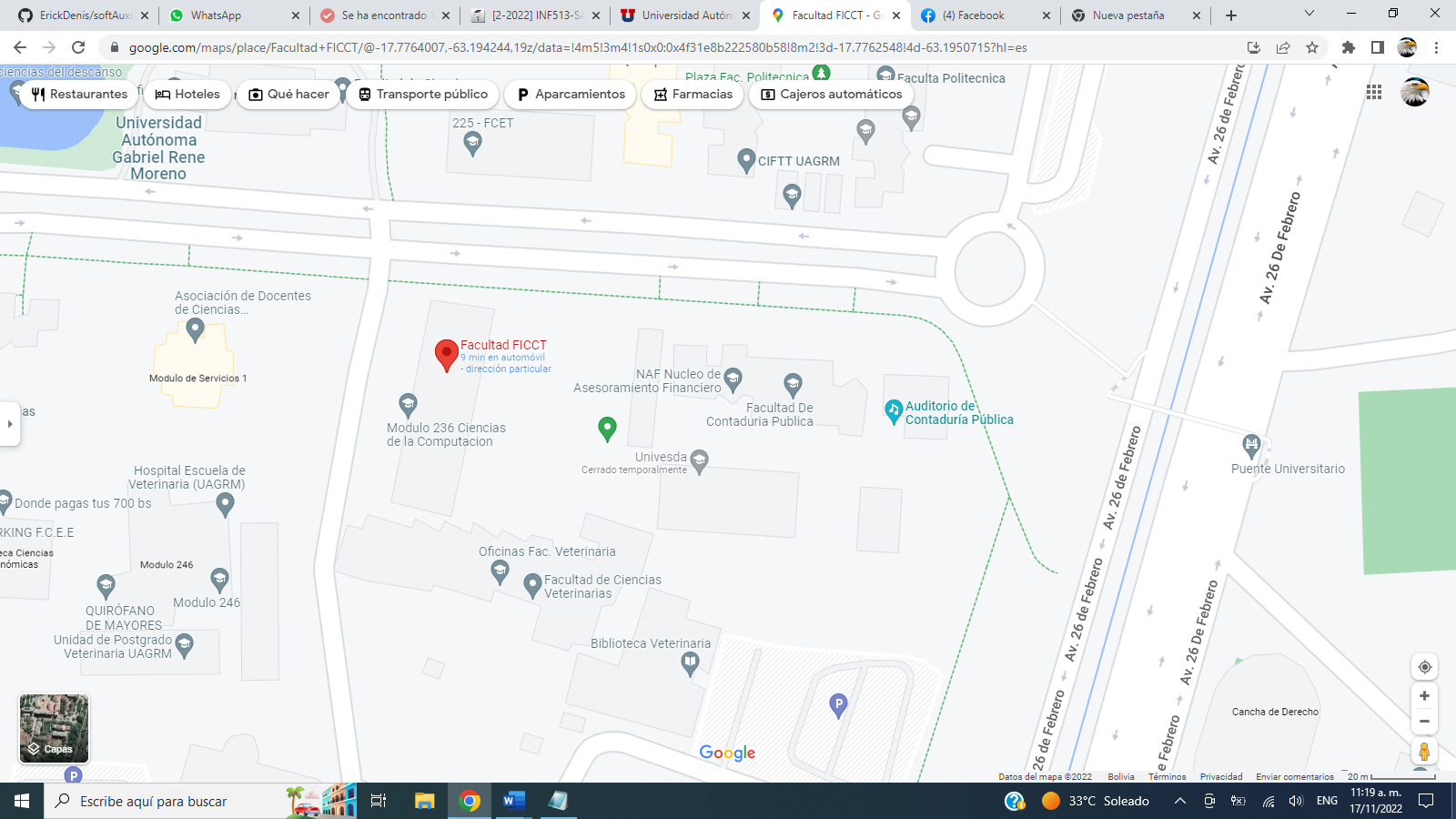
## ANTECEDENTES DE AUXILIARES UAGRM

Un auxiliar de docencia es el estudiante, en calidad de becario, la función del auxiliar de docencia es apoyar la labor académica del docente, participando en el proceso de enseñanza – aprendizaje, mediante el desarrollo de contenidos programáticos prácticos, para mejorar los conocimientos de los estudiantes en determinadas asignaturas, ejecutados en el aula o sala de cómputo o salas de apoyo a la docencia o laboratorio. Las materias pueden ser troncales (asignatura básica de la ciencia o troncal) y cuantitativas (asignatura básica o básica instrumental) de los planes de estudios, posibilitando la superación académica y el logro de los objetivos de la U.A.G.R.M. Para ser un auxiliar se le realiza previamente un examen de competencia para ver cual está más apto para realizar dicho cargo.

## SÍNTESIS HISTÓRICA

Inicialmente existía la carrera de Ingeniería Informática, fundada el 16 de julio de 1987; del cual en el año 2003 se crean dos menciones de estudios: Mención Ciencias de la Computación y Mención Análisis de Sistemas.

Dichas menciones son replanteadas para ser planes de estudios, de donde en el año 2005 se eliminan y se establecen tres planes de estudios de donde cada uno tendrá un título por separado: Ingeniería Informática, Ingeniería en Sistemas e Ingeniería en Redes y Telecomunicaciones; es con estas tres carreras con las que nace la facultad.



## JUSTIFICACIÓN

El e-mail se ha vuelto un servicio indispensable para la compartición de documentos e información privada o personal, todas las instituciones lo usan en su totalidad, solicitan a sus usuarios como un identificador único por persona y como servicio de uso personal para la comunicación entre instituciones -usuario.

Con este proyecto expandiremos la difusión de información mediante correo electrónico, para la mejor accesibilidad de los usuarios del sistema, sin restricciones de hora ni espacio, lo que conlleva el beneficio de la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones de la UAGRM.

## MISIÓN

La Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones es una unidad de gestión, organización y desarrollo de actividades académicas, investigación y extensión para la formación integral de profesionales en ciencias de la computación y telecomunicación para la creación, difusión y cultivo de las tecnologías de la información y comunicación en pre y postgrado para contribuir al logro del desarrollo humano sostenible.

## VISIÓN

Para el año 2018, la Facultad de Ingeniería en Ciencias de la Computación, Redes y Telecomunicaciones, es reconocida por la calidad de gestión académica - administrativa, de sus programas formativos de pre y postgrado, por la generación de conocimiento científico y tecnológico pertinente y adecuado a los requerimientos del desarrollo científico y tecnológico a nivel nacional e internacional.

## OBJETIVOS

### OBJETIVO GENERAL

Desarrollar un sistema para la gestión de Auxiliares para la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones, el cual funcione vía correo electrónico.

### OBJETIVOS ESPECÍFICOS

* Recabar información del funcionamiento y de los procesos de negocio de la construcción mediante observación de los procesos de negocio, entrevista con el vice decano de la FICCT y auxiliares.
* Analizar la información obtenida en la investigación para cubrir los requerimientos del negocio.
* Identificar los requisitos funcionales del sistema a través de los requerimientos del usuario.
* Diseñar e implementar una base de datos utilizando un SGBD “PostgreSql”, capaz de soportar todos los requerimientos del sistema para la gestión de Auxiliares para la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones.
* Realizar pruebas pertinentes y necesarias para justificar el buen funcionamiento del sistema o encontrar posibles fallas para luego proceder a corregirlas.

## DESCRIPCIÓN DEL PROBLEMA

La facultad de Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones, enfrenta problemas al llevar a cabo la correcta gestión de auxiliares, tales como:

* Falta de gestión de convocatoria.
* Falta de reportes sobre las auxiliaturas y otra información necesaria.
* Falta de información acerca de las auxiliaturas.
* No se sabe que auxiliares pertenecen a las materias correspondientes.

## ALCANCE

El sistema para la gestión de los auxiliares, vía correo tendrá las siguientes funciones:

* Gestionar Usuarios
* Gestionar Periodo
* Gestionar Convocatoria
* Gestionar Materias Ofertadas
* Gestionar Postulantes
* Gestionar Evaluaciones
* Gestionar Carga Horaria Auxiliares
* Gestionar Reportes y Estadísticas

## ELEMENTOS DEL SISTEMA

### HARDWARE

Los recursos físicos necesarios para el funcionamiento del sistema serán:

* El servidor.

### SOFTWARE

Entre los recursos lógicos, manejaremos los siguientes:

* Sistema Operativo Fedora Server 34(64-bits).
* SGBD PostgreSQL.

### DATOS

Entre los datos que manejará el sistema tenemos:

* Productos.
* Mensualidades.
* Asistencias.
* Rutinas y ejercicios.
* Cuentas de nuestros clientes.

### PERSONAS

Quien o quienes tendrán acceso al sistema:

* Administrador.
* Entrenadores.
* Clientes.

### PROCESOS

Las políticas de negocio a seguir son:

* Gestionar usuarios y datos personales.
* Gestionar Productos.
* Gestionar Servicios.
* Gestionar Ventas.
* Gestionar Rutinas y Ejercicios.
* Realizar Reportes.

## TECNOLOGÍAS PARA EL DESARROLLO DE SOFTWARE

### SISTEMA OPERATIVO

El Sistema operativo utilizado como plataforma para nuestro proyecto es Fedora Server 34, ya que es un sistema que cumple con todos los requerimientos necesarios, y además es distribución libre y al ser Linux es seguro.

### SISTEMA DE GESTIÓN DE BASE DE DATOS

Las características que pueda tener un Gestor de Base de Datos son muy importantes, ya que tienen que cumplir con las exigencias de nuestro proyecto, entre las más importantes se pueden mencionar los siguientes:

* Control de redundancia con el fin de evitar que los usuarios tengan su información repetida y que provoque problemas en la administración.
* Representación de vínculos complejos.
* Cumplir con las restricciones de integridad.
* Debe funcionar con el Sistema Operativo Linux Fedora 34 y ser versión libre.
* Respaldo y recuperación de los datos cuando ocurren fallos en el sistema.
  + Transacciones.
  + Disparadores.
  + Backup y recuperación.
  + Procedimientos almacenados.
  + Integridad referencial.

Tomando en cuenta las características analizadas elegimos como gestor de Base de Datos PostgreSQL, el cual cumple con nuestras exigencias mencionadas con anterioridad.

### HERRAMIENTAS CREATELY

Creately es un espacio de trabajo visual para la colaboración en equipo. Que permite el uso de diagramas, dibujos, imágenes y texto en un lienzo infinito para colaborar en línea en tiempo real. Compatible con más de 50 tipos de diagramas de flujo, mapas mentales, organigramas, UML y diseños de bases de datos, diagramas de red, diagramas de Gantt, diagramas de procesos comerciales y más. Permite trabajar de forma colaborativa y remota o en la misma ubicación. En la actualidad más de 4 millones de usuarios en todo el mundo confían en Creately

## ENTORNO DE DESARROLLO

Se optó por Apache NetBeans ya que es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. Apache NetBeans es un producto libre y gratuito sin restricciones de uso.



Capas del Protocolo TCP/IP, **Fuente:** Web (2018)

La plataforma Apache NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de Apache NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma Apache NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

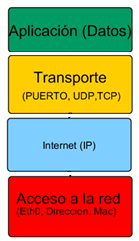
# MARCO TEÓRICO

El marco teórico que fundamenta este proyecto proporcionará al lector una idea más clara acerca de este tema. Se encontrarán los conceptos muy básicos, los complementarios y los específicos. Primero partiremos con la definición de protocolo TCP/IP, correo electrónico, POP3, SMTP, Mensajes en formato MIME.

## PROTOCOLO TCP/IP

Los protocolos son conjuntos de normas para formatos de mensaje y procedimientos que permiten a las máquinas y los programas de aplicación intercambiar información. Cada máquina implicada en la comunicación debe seguir estas normas para que el sistema principal de recepción pueda interpretar el mensaje. El conjunto de protocolos TCP/IP puede interpretarse en términos de capas (o niveles).

Esta figura muestra las capas del protocolo TCP/IP. Empezando por la parte superior son: capa de aplicación, capa de transporte, capa de red, capa de interfaz de red y hardware.



Capas del Protocolo TCP/IP, **Fuente:** Web (2018)

TCP/IP define cuidadosamente cómo se mueve la información desde el remitente hasta el destinatario. En primer lugar, los programas de aplicación envían mensajes o corrientes de datos a uno de los protocolos de la capa de transporte de Internet, UDP (User Datagram Protocol) o TCP (Transmission Control Protocolo). Estos protocolos reciben los datos de la aplicación, los dividen en partes más pequeñas llamadas paquetes, añaden una dirección de destino y, a continuación, pasan los paquetes a la siguiente capa de protocolo, la capa de red de Internet.

La capa de red de Internet pone el paquete en un datagrama de IP (Internet Protocol), pone la cabecera y la cola de datagrama, decide dónde enviar el datagrama (directamente a un destino o a una pasarela) y pasa el datagrama a la capa de interfaz de red.

## CORREO ELECTRÓNICO

El correo electrónico [servicio de red](https://es.wikipedia.org/wiki/Servicio_de_red) que permite a los usuarios enviar y recibir mensajes (también denominados [mensajes](https://es.wikipedia.org/wiki/Mensaje) electrónicos o cartas digitales) mediante [redes de comunicación electrónica](https://es.wikipedia.org/wiki/Red_de_telecomunicaci%C3%B3n). ​ El término «correo electrónico» proviene de la analogía con el [correo](https://es.wikipedia.org/wiki/Correo) postal: ambos sirven para enviar y recibir mensajes, y se utilizan «buzones» intermedios ([servidores de correo](https://es.wikipedia.org/wiki/Servidor_de_correo)). Por medio del correo electrónico se puede enviar no solamente [texto](https://es.wikipedia.org/wiki/Texto_informativo), sino todo tipo de [archivos digitales](https://es.wikipedia.org/wiki/Archivo_inform%C3%A1tico), si bien suelen existir limitaciones al tamaño de los [archivos adjuntos](https://es.wikipedia.org/wiki/Archivo_adjunto).

Los sistemas de correo electrónico se basan en un modelo de [almacenamiento y reenvío](https://es.wikipedia.org/wiki/Almacenamiento_y_reenv%C3%ADo), de modo que no es necesario que ambos extremos se encuentren conectados simultáneamente. Para ello se emplea un servidor de correo que hace las funciones de intermediario, guardando temporalmente los mensajes antes de enviarse a sus destinatarios. En [Internet](https://es.wikipedia.org/wiki/Internet), existen multitud de estos servidores, que incluyen a [empresas](https://es.wikipedia.org/wiki/Correo_corporativo), [proveedores de servicios de internet](https://es.wikipedia.org/wiki/Proveedor_de_servicios_de_Internet) y proveedores de correo tanto libres como de pago.

### FUNDAMENTO DEL CORREO ELECTRONICO

El correo electrónico gira alrededor del uso de las casillas de correo electrónico. Cuando se envía un correo electrónico, el mensaje se enruta de servidor a servidor hasta llegar al servidor de correo electrónico del receptor. Más precisamente, el mensaje se envía al servidor del correo electrónico (llamado MTA, del inglés Mail TransportAgent [Agente de Transporte de Correo]) que tiene la tarea de transportarlos hacia el MTA del destinatario. En Internet, los MTA se comunican entre sí usando el protocolo SMTP, y por lo tanto se los llama servidores SMTP (o a veces servidores de correo saliente).

Luego el MTA del destinatario entrega el correo electrónico al servidor del correo entrante (llamado MDA, del inglés Mail DeliveryAgent [Agente de Entrega de Correo]), el cual almacena el correo electrónico mientras espera que el usuario lo acepte. Existen dos protocolos principales utilizados para recuperar un correo electrónico de un MDA:

* POP3 (Post Office Protocolo), el más antiguo de los dos, que se usa para recuperar el correo electrónico y, en algunos casos, dejar una copia en el servidor.
* IMAP (Internet Message Access Protocol), el cual se usa para coordinar el estado de los correos electrónicos (leído, eliminado, movido) a través de múltiples clientes de correo electrónico. Con IMAP, se guarda una copia de cada mensaje en el servidor, de manera que esta tarea de sincronización se pueda completar.

Por esta razón, los servidores de correo entrante se llaman servidores POP o servidores IMAP, según el protocolo usado.



Funcionamiento del Correo Electrónico en Internet, **Fuente:** Web (2018)

Usando una analogía del mundo real, los MTA actúan como la oficina de correo (el área de clasificación y de transmisión, que se encarga del transporte del mensaje), mientras que los MDA actúan como casillas de correo, que almacenan mensajes (tanto como les permita su volumen), hasta que los destinatarios controlan su casilla. Esto significa que no es necesario que los destinatarios estén conectados para poder enviarles un correo electrónico.

Para evitar que cualquiera lea los correos electrónicos de otros usuarios, el MDA está protegido por un nombre de usuario llamado registro y una contraseña. La recuperación del correo se logra a través de un programa de software llamado MUA (Mail UserAgent [Agente Usuario de Correo]). Cuando el MUA es un programa instalado en el sistema del usuario, se llama cliente de correo electrónico (tales como Mozilla Thunderbird, Microsoft Outlook, Eudora Mail, Incredimail o Lotus Notes).Cuando se usa una interfaz de web para interactuar con el servidor de correo entrante, se llama correo electrónico.

## PROTOCOLO SMTP

El protocolo para transferencia simple de correo (en inglés Simple Mail Transfer Protocol o SMTP) es un protocolo de red utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos (PDA, teléfonos móviles, impresoras, etc). Fue definido inicialmente en agosto de 1982 por el RFC 821 (para la transferencia) y el RFC 822 (para el mensaje). Son estándares oficiales de Internet que fueron reemplazados respectivamente por el RFC 2821 y el RFC 2822, que a su vez lo fueron por el RFC 5321 y el RFC 5322.

El funcionamiento de este protocolo se da en línea, de manera que opera en los servicios de correo electrónico. Sin embargo, este protocolo posee algunas limitaciones en cuanto a la recepción de mensajes en el servidor de destino (cola de mensajes recibidos). Como alternativa a esta limitación se asocia normalmente a este protocolo con otros, como el POP o IMAP, otorgando a SMTP la tarea específica de enviar correo, y recibirlos empleando los otros protocolos antes mencionados (POP O IMAP).

* + 1. MODO DE COMUNICACIÓN SMTP

El proceso de entrega del email es en realidad muy similar al correo clásico: un sistema organizado se encarga de transportar su mensaje a lo largo de una serie de pasos y deposita el mismo en su destinatario. En este proceso, el servidor SMTP es simplemente una computadora que ofrece un servicio de SMTP, la cual actúa más o menos como un cartero electrónico. Una vez que el mensaje ha sido entregado al servidor, este se encarga de concretar la entrega a sus destinatarios.

Básicamente, el viaje desde de un mensaje desde su computadora hasta su lector se realiza de la siguiente forma:

1. Usted envía un email con su cliente de email de escritorio o vía web desde su propia dirección de correo (por ejemplo marcos@gmail.com) a un contacto determinado (por ejemplo julia@yahoo.com). En la jerga, el cliente de email de escritorio o servicio webmail es denominado MUA – Message User Agent (Agente de Mensaje de Usuario).
2. El mensaje es enviado generalmente utilizando el puerto 25 a un servidor SMTP (por ejemplo smtp.servidor.com) el cuál es configurado en su cliente de correo y actúa como MTA – Message Transfer Agente (Agente de Transferencia de Mensaje).
3. El cliente y el servidor comienzan una breve “conversación” en donde el servidor verifica toda la información respectiva a la transmisión del mensaje (remitente, destinatarios, dominios, etc.). Tenga en cuenta que el lenguaje SMTP define sólo la transmisión del mensaje, y no observa el contenido del cuerpo del mensaje.
4. Luego, si el dominio al cual pertenece el destinatario pose su cuenta directamente conectada al servidor, el email es entregado inmediatamente. En caso contrario, el servidor SMTP entrega el mensaje a otro servidor más cercano al destinatario (en la jerga, estos pasajes se denominan entregas). En nuestro mensaje, el servidor web se conecta con el servidor de Dominio, el cual (si todo es correcto) recibe el email y lo almacena.
5. ¿Qué sucede si el servidor del destinatario se encuentra caído u ocupado? El servidor SMTP simplemente entrega el mensaje a un servidor de backup: si ninguno se encuentra disponible, el email es encolado y se reintenta la entrega en forma periódica. Después de un periodo determinado, si la entrega no es exitosa, el mensaje es retornado como no entregable.
6. Si no hay problemas, el segmento final es controlado por POP, otro protocolo que recoge el email desde el servidor receptor y lo coloca en la bandeja de entrada del destinatario.



Esquema del Funcionamiento de SMTP, **Fuente:** Web (2018)

Cabe destacar, de todas formas, que los servidores SMTP utilizados al enviar sus “emails normales” – por ejemplo, aquellos asociados a proveedores tales como Hotmail, Gmail, Yahoo – son compartidos entre muchos usuarios con direcciones IP no dedicadas, es decir compartidas.

Esto significa que usted podría terminar utilizando una dirección IP que también ha sido utilizada por un spammer: un hecho que puede afectar en forma negativa la capacidad de entrega de sus mensajes. También, estos proveedores utilizan límites estrictos en la cantidad de emails que usted puede enviar (por ejemplo, Yahoo limita a 100 emails por hora).

## MIME

Multipurpose Internet Mail Extensions o MIME (en español "extensiones multipropósito de correo de internet") son una serie de convenciones o especificaciones dirigidas al intercambio a través de Internet de todo tipo de archivos (texto, audio, vídeo, etc.) de forma transparente para el usuario. Una parte importante del MIME está dedicada a mejorar las posibilidades de transferencia de texto en distintos idiomas y alfabetos.

En sentido general las extensiones de MIME van encaminadas a soportar:

* Texto en conjuntos de caracteres distintos de US-ASCII
* Adjuntos que no son de tipo texto
* Cuerpos de mensajes con múltiples partes (multi-part)
* Información de encabezados con conjuntos de caracteres distintos de ASCII.

Prácticamente todos los mensajes de correo electrónico escritos por personas en Internet y una proporción considerable de estos mensajes generados automáticamente son transmitidos en formato MIME a través de SMTP. Los mensajes de correo electrónico en Internet están tan cercanamente asociados con el SMTP y MIME que usualmente se les llama mensaje SMTP/MIME.

En 1991 la IETF (Grupo de Trabajo en Ingeniería de Internet, Internet Engineering Task Force en inglés) comenzó a desarrollar esta norma y desde 1994 todas las extensiones MIME están especificadas de forma detallada en diversos documentos oficiales disponibles en Internet.

MIME está especificado en seis Request for Comments o RFC (en español "solicitud de comentarios): RFC 2045, RFC 2046, RFC 2047, RFC 4288, RFC 4289 y RFC 2077.

Los tipos de contenido definidos por el estándar MIME tienen gran importancia también fuera del contexto de los mensajes electrónicos. Ejemplo de esto son algunos protocolos de red tales como HTTP de la Web. HTTP requiere que los datos sean transmitidos en un contexto de mensajes tipo e-mail aunque los datos pueden no ser un e-mail propiamente dicho.

En la actualidad ningún programa de correo electrónico o navegador de Internet puede considerarse completo si no acepta MIME en sus diferentes facetas (texto y formatos de archivo).

* + 1. MIME HEADERS

A continuación, el formato de un mensaje de correo electrónico por SMTP en formato MIME:

HELO mail.tecnoweb.org.bo

MAIL FROM: grupo07sc@tecnoweb.org.bo

RCPT To: tecnoweb@hotmail.com

DATA

From: mitiendaonline@uagrm.edu.bo

To: tecnoweb@hotmail.com

Subject: Test de Envio

Mime-Version: 1.0;

Content-Type: text/html; charset="ISO-8859-1";

Content-Transfer-Encoding: 7bit;

<html>

<body>

<h2>Este es un link a la UAGRM!</h2>

click aqui!<a href="http://www.uagrm.edu.bo">UAGRM</a>

</body>

</html>

.

QUIT

## PROTOCOLO POP3

El protocolo POP (Protocolo de oficina de correos), como su nombre lo indica, permite recoger el correo electrónico en un servidor remoto (servidor POP). Es necesario para las personas que no están permanentemente conectadas a Internet, ya que así pueden consultar sus correos electrónicos recibidos sin que ellos estén conectados.

Existen dos versiones principales de este protocolo, POP2 y POP3, a los que se le asignan los puertos 109 y 110 respectivamente, y que funcionan utilizando comandos de texto radicalmente diferentes.

Al igual que con el protocolo SMTP, el protocolo POP (POP2 y POP3) funciona con comandos de texto enviados al servidor POP. Cada uno de estos comandos enviados por el cliente (validados por la cadena CR/LF) está compuesto por una palabra clave, posiblemente acompañada por uno o varios argumentos, y está seguido por una respuesta del servidor POP compuesta por un número y un mensaje descriptivo.

A continuación, se brinda un resumen de los principales comandos POP3:

|  |  |
| --- | --- |
| Comandos POP3 | |
| Comando | Descripción |
| USER | Este comando permite la autenticación. Debe estar seguido del nombre de usuario, es decir, una cadena de caracteres que identifique al usuario en el servidor. El comando USER debe preceder al comando PASS. |
| PASS | El comando PASS permite especificar la contraseña del usuario cuyo nombre ha sido especificado por un comando USER previo. |
| STAT | Comando que muestra la cantidad de mensajes que hay. |
| RETR | Comando que recoge un mensaje en concreto. |
| DELE | Comando que elimina un mensaje. |
| LIST | Comando que muestra algunos detalles de un mensaje. |
| NOOP | Comando que permite mantener la conexión abierta en caso de inactividad. |
| TOP | Comando que muestra un determinado número de líneas, dado un mensaje. |
| UIDL | Esta línea contiene una cadena de caracteres denominada UIL que permite identificar de manera única el mensaje en el servidor, independientemente de la sesión. El argumento opcional es un número relacionado con un mensaje existente en el servidor POP, es decir, un mensaje que no se ha borrado |
| QUIT | El comando QUIT solicita la salida del servidor POP3. Lleva a la eliminación de todos los mensajes marcados como eliminados y envía el estado de esta acción. |

Comandos de descripción del protocolo POP3, **Fuente:** Web (2017)

Por lo tanto, el protocolo POP3 administra la autenticación utilizando el nombre de usuario y la contraseña. Sin embargo, esto no es seguro, ya que las contraseñas, al igual que los correos electrónicos, circulan por la red como texto sin codificar (de manera no cifrada).

En realidad, según RFC 1939, es posible cifrar la contraseña utilizando un algoritmo MD5 y beneficiarse de una autenticación segura. Sin embargo, debido a que este comando es opcional, pocos servidores lo implementan. Además, el protocolo POP3 bloquea las bandejas de entrada durante el acceso, lo que significa que es imposible que dos usuarios accedan de manera simultánea a la misma bandeja de entrada.

## SENDMAIL

Es un MTA (agente de transporte de correo), que es el programa que se encarga de mover el correo de una máquina a otra. El propósito principal de Sendmail, como cualquier otro MTA; es el de transferir correo de forma segura entre hosts, usualmente usando el protocolo SMTP. Sin embargo, Sendmail es altamente configurable, permitiendo el control sobre casi cada aspecto del manejo de correos, incluyendo el protocolo utilizado. Muchos administradores de sistemas seleccionan Sendmail como su MTA debido a su poder y escalabilidad.



Logo de Sendmail, **Fuente:** Web (2021)

Sendmail lleva incorporado aliasing y fordwarding, rutado automático hacia puertas de enlace, y una configuración flexible. Es una solución potente para cualquier entorno.

Sendmail es el MTA más común de Internet (en los sistemas UNIX), aunque actúa principalmente como MTA, también puede ser utilizado como MUA (aunque no posee interfaz de usuario).

Las misiones básicas de Sendmail son las siguientes:

* Recogida de mails provenientes de un MUA como pueden ser Eudora o pine; o provenientes de un MTA como puede ser el propio Sendmail.
* Elección de la estrategia de reparto de los mails, basándose en la información de la dirección del destinatario contenida en la cabecera:
* Si el mail es local en nuestro sistema, enviará el mail al programa de reparto local de mails.
* Si el mail no es local, Sendmail utilizará el DNS de nuestro sistema para determinar el host al que debe ser enviado el mail. Para transferir el mensaje.
* Iniciará una sesión SMTP con el MTA de dicho host.
* Si no es posible mandar el mail a su destino (porque la maquina receptora esta desconectada, o va muy lenta), Sendmail almacenará los mails en una cola de correo, y volverá a intentar el envío del mail un tiempo después. Si el mail no puede ser enviado tras un tiempo razonable, el mail será devuelto a su autor con un mensaje de error. Sendmail debe garantizar que cada mensaje llegue correctamente a su destino, o si hay error este debe ser notificado (ningún mail debe perderse completamente).

Reformatear el mail antes de pasarlo a la siguiente máquina, según unas reglas de reescritura. Según el tipo de conexión que poseamos con una determinada máquina, o según el agente de transporte al que vaya dirigido el mail, necesitaremos cambiar los formatos de las direcciones del remitente y del destinatario, algunas líneas de la cabecera del mail, o incluso puede que necesitemos añadir alguna línea a la cabecera. Sendmail debe realizar todas estas tareas para conseguir la máxima compatibilidad entre usuarios distintos.

Otra función muy importante de Sendmail es permitir el uso de "alias" entre los usuarios del sistema; lo que nos permitirá (entre otras funciones) crear y mantener listas de correo entre grupos.

Ejecución como agente de usuario (MUA). Aunque no posee interfaz de usuario, Sendmail también permite el envío directo de mails a través de su ejecutable.

# REQUISITOS

## REQUISITOS DEL SISTEMA

El sistema para la gestión de los auxiliares dentro de la facultad de Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones esta contiene la siguiente lista de requerimientos funcionales.

### REQUERIMIENTOS FUNCIONALES

* El sistema podrá gestionar los datos de los usuarios del sistema.
* El sistema podrá gestionar los datos de los periodos que existen dentro del año.
* El sistema podrá gestionar las convocatorias realizadas dentro de los periodos.
* El sistema podrá gestionar las materias ofertadas.
* El sistema podrá gestionar los postulantes.
* El sistema podrá gestionar las evaluaciones.
* El sistema podrá gestionar la carga horaria de los auxiliares.
* El sistema podrá gestionar reportes y estadísticas.

## IDENTIFICACIÓN DE ACTORES Y CASOS DE USO

### IDENTIFICACIÓN DE ACTORES Y ROLES

Los actores que participaran en el sistema, que se definieron son los siguientes:

* **Administrador**: tiene acceso total al sistema.

### LISTA DE CASOS DE USO

* Gestionar Usuarios
* Gestionar Periodo
* Gestionar Convocatoria
* Gestionar Materias Ofertadas
* Gestionar Postulantes
* Gestionar Evaluaciones
* Gestionar Carga Horaria Auxiliares
* Gestionar Reportes y Estadísticas

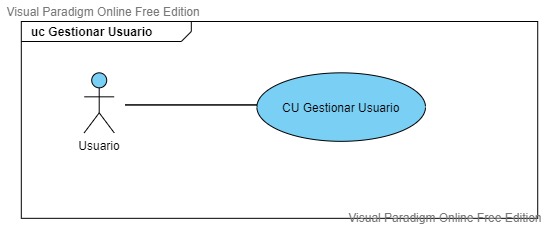
## PRIORIZACIÓN DE CASOS DE USO

A continuación, se detalla en una tabla la priorización de casos de uso con tres niveles de clasificación: Alta, Media y Baja.

|  |  |
| --- | --- |
| CASO DE USO | PRIORIDAD |
| CU1: Gestionar Usuarios. | Alto |
| CU2: Gestionar Periodo. | Medio |
| CU3: Gestionar Convocatoria. | Medio |
| CU4: Gestionar Materias Ofertadas. | Alto |
| CU5: Gestionar Postulantes. | Bajo |
| CU6: Gestionar Evaluaciones. | Media |
| CU7: Gestionar Carga Horaria Auxiliares. | Alto |
| CU8: Gestionar Reportes y Estadísticas. | Alto |

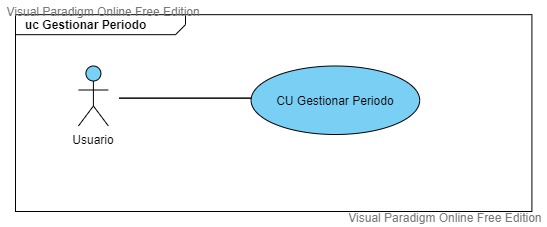
## DETALLE DE LOS CASOS DE USO

CU1 Gestionar Usuarios



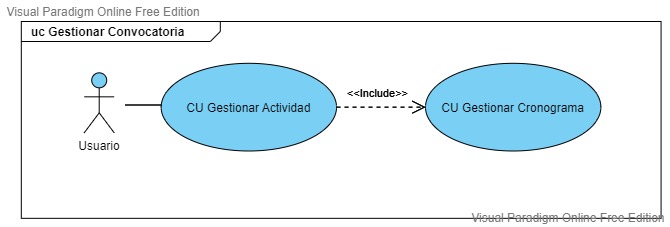
|  |  |
| --- | --- |
| Nombre: Gestionar Usuario | Código: CU 01 |
| Actores: Usuario | |
| Objetivo: Permitir gestionar los usuarios. | |
| Descripción: Se podrá:   * Listar: Usuarios, Usuarios por Rol ID, Usuario por Nombre de Rol, Usuario por Nombre de Usuario y Usuario por Apellido de Usuario. * Registrar: Decano, Vice, Auxiliar. * Modificar: Decano, Vice, Auxiliar. * Eliminar: Decano, Vice, Auxiliar. | |
| Flujo Normal:   * Se debe ingresar el comando correspondiente a la acción deseada, con sus parámetros respectivos. * El sistema debe evaluar si la cantidad de parámetros es correcta. * El sistema debe verificar si la sintaxis es correcta. | |
| Excepción:   * Comando desconocido. * Error en cantidad de parámetros. * Parámetros incorrectos. | |

CU2: Gestionar Periodo



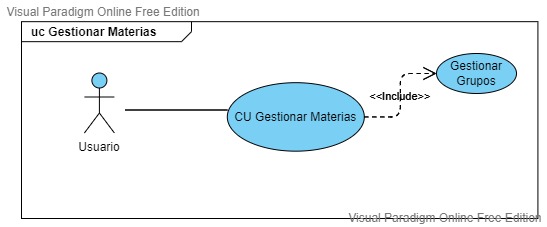
|  |  |
| --- | --- |
| Nombre: Gestionar Periodo | Código: CU 02 |
| Actores: Usuario | |
| Objetivo: Permitir gestionar los periodos que maneja la institución. | |
| Descripción: Se podrá:   * Listar: Periodos por fecha de inicio y fecha fin. * Registrar: Nombre, Semestre, Año. * Modificar: Nombre, Semestre, Año. * Eliminar: Nombre, Semestre, Año. | |
| Pre-Condición: ninguna. | |
| Flujo Normal:   * Se debe ingresar el comando correspondiente a la acción deseada, con sus parámetros respectivos. * El sistema debe evaluar si la cantidad de parámetros es correcta. * El sistema debe verificar si la sintaxis es correcta. | |
| Excepción:   * Comando desconocido. * Error en cantidad de parámetros. * Parámetros incorrectos. | |
| Post-Condición:   * El administrador podrá registrar, modificar o eliminar los periodos. | |

CU3 Gestionar Convocatoria



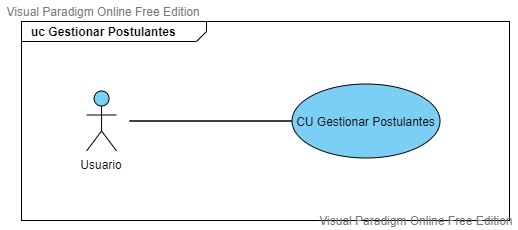
|  |  |
| --- | --- |
| Nombre: Gestionar Convocatoria | Código: CU 03 |
| Actores: Usuario | |
| Objetivo: Permitir gestionar las convocatorias para cada periodo que se tenga. | |
| Descripción: Se podrá:   * Listar: Convocatoria, Convocatoria por Periodo ID, Convocatoria por Nombre Periodo, Convocatoria por Nombre de Creador, Convocatoria por Apellido de Creador y Convocatoria por ID de Creador. * Registrar: Actividad y Cronograma. * Eliminar: Actividad y Cronograma. | |
| Flujo Normal:   * Se debe ingresar el comando correspondiente a la acción deseada, con sus parámetros respectivos. * El sistema debe evaluar si la cantidad de parámetros es correcta. * El sistema debe verificar si la sintaxis es correcta. | |
| Excepción:   * Comando desconocido. * Error en cantidad de parámetros. * Parámetros incorrectos. | |

CU4 Gestionar Materias



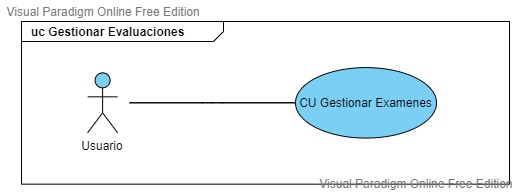
|  |  |
| --- | --- |
| Nombre: Gestionar Materias | Código: CU |
| Actores: Usuario | |
| Objetivo: Permitir gestionar las materias y sus grupos que estarán habilitadas para auxiliaturas | |
| Descripción: Se podrá:   * Listar con o sin filtros * Registrar * Eliminar * Ver | |
| Pre-Condición: Conocer el comando y el formato que se debe enviar | |
| Flujo Normal:   * Listar   + Para listar el usuario debe enviar el comando LISMAT y se le enviara una respuesta con el listado de las materias   + Para listar filtrado el usuario debe enviar el comando con el siguiente formato LISMAT:nombre=valor:sigla=valor y se le devolverá la lista de las materias que contengan esos filtros, estos filtros no son obligatorios pero si los nombres de las claves   + Para listar los grupos el usuario debe enviar el comando con el siguiente formato LISMATGRUP o con filtros LISMATGRUP:materia=valor:sigla=valor:grupo=valor:carrera=valor * Registrar   + Para registrar una MATERIA el usuario deberá enviar el comando con el siguiente formato REGMAT:nombre=valor:sigla=valor   + el orden de los parámetros no es importante, pero si el nombre de las claves (nombre, sigla)   + Para registrar un GRUPO el usuario deberá enviar el comando con el siguiente formato REGMATGRUP:materia=valor:grupo=valor:horario=valor:docente=valor:carrera=valor   + el orden de los parámetros no es importante, pero si el nombre de las claves (materia, grupo, horario, docente, carrera) * Eliminar   + Para eliminar una materia el usuario deberá enviar el comando con el siguiente formato DELMAT:id=valor   + la clave únicamente puede ser el id y el valor numérico entero * Ver   + Para ver una materia el usuario debe enviar el comando con el siguiente formato VERMAT:id=valor   + La clave únicamente puede ser id y el valor numérico entero | |
| Excepción:   * Comando desconocido. * Error en cantidad o formato de parámetros * Parámetros incorrectos. * Error interno o de conexión o integridad de la base de datos | |
| Post-Condición:   * El usuario se ha registrado, modificado, eliminado o visto. | |

CU5 Gestionar Postulantes



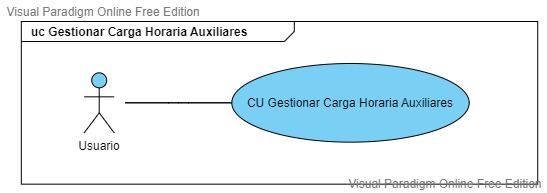
|  |  |
| --- | --- |
| Nombre: Gestionar Postulantes | Código: CU |
| Actores: Usuario | |
| Objetivo: Permitir gestionar los postulantes para auxiliares | |
| Descripción: Se podrá:   * Listar con o sin filtros * Registrar * Ver por regisro | |
| Pre-Condición: Conocer el comando y el formato que se debe enviar | |
| Flujo Normal:   * Listar   + Para listar el usuario debe enviar el comando LISPOS y se le enviara una respuesta con el listado de los postulantes   + Para listar filtrado el usuario debe enviar el comando con el siguiente formato LIPOS:nombre=valor:periodo=valor:carrera=valor y se le devolverá la lista de los postulantes que contengan esos filtros, estos filtros no son obligatorios pero si los nombres de las claves * Registrar   + Para registrar una MATERIA el usuario deberá enviar el comando con el siguiente formato REGPOS: nombres=valor:apellidos=valor:registro=valor:ci=valor:carrera=valor:periodo=valor:materias=valor1,valor2,..   + el orden de los parámetros no es importante, pero si el nombre de las claves (nombre, sigla) * Ver   + Para ver un postulante el usuario debe enviar el comando con el siguiente formato VERPOS:registro=valor   + El valor de registro debe ser un registro valido | |
| Excepción:   * Comando desconocido. * Error en cantidad o formato de parámetros * Parámetros incorrectos. * Error interno o de conexión o integridad de la base de datos | |
| Post-Condición:   * El usuario se ha registrado, modificado, eliminado o visto. | |

CU6 Gestionar Evaluaciones



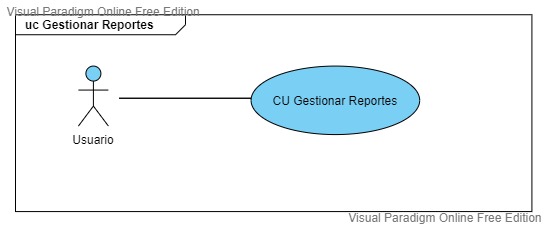
|  |  |
| --- | --- |
| Nombre: Gestionar Evaluaciones | Código: CU |
| Actores: Usuario | |
| Objetivo: Permitir gestionar las evaluaciones de una materia | |
| Descripción: Se podrá:   * Listar con o sin filtros * Registrar * Listar evaluaciones de un postulante por registro | |
| Pre-Condición: Conocer el comando y el formato que se debe enviar | |
| Flujo Normal:   * Listar   + Para listar el usuario debe enviar el comando LISEVAL y se le enviara una respuesta con el listado de todas las evaluaciones   + Para listar filtrado el usuario debe enviar el comando con el siguiente formato LISEVAL:sigla=valor:periodo=valor:docente=valor y se le devolverá la lista de las evaluaciones que contengan esos filtros, estos filtros no son obligatorios pero si los nombres de las claves * Registrar   + Para registrar una evaluación el usuario deberá enviar el comando con el siguiente formato REGEVAL:materia=INF115:periodo=3:docente=4:fecha=2022.08.19-19.45   + el orden de los parámetros no es importante, pero si el nombre de las claves * Listar evaluaciones de un postulante   + Para obtener las evaluaciones de un postulante se debe ejecutar el siguiente comando:   + LISEVALPOS:registro=valor   + El valor debe ser un registro válido y existente de un postulante | |
| Excepción:   * Comando desconocido. * Error en cantidad o formato de parámetros * Parámetros incorrectos. * Error interno o de conexión o integridad de la base de datos | |
| Post-Condición:   * El usuario se ha registrado, modificado, eliminado o visto. | |

CU7 Gestionar Carga Horaria Auxiliares



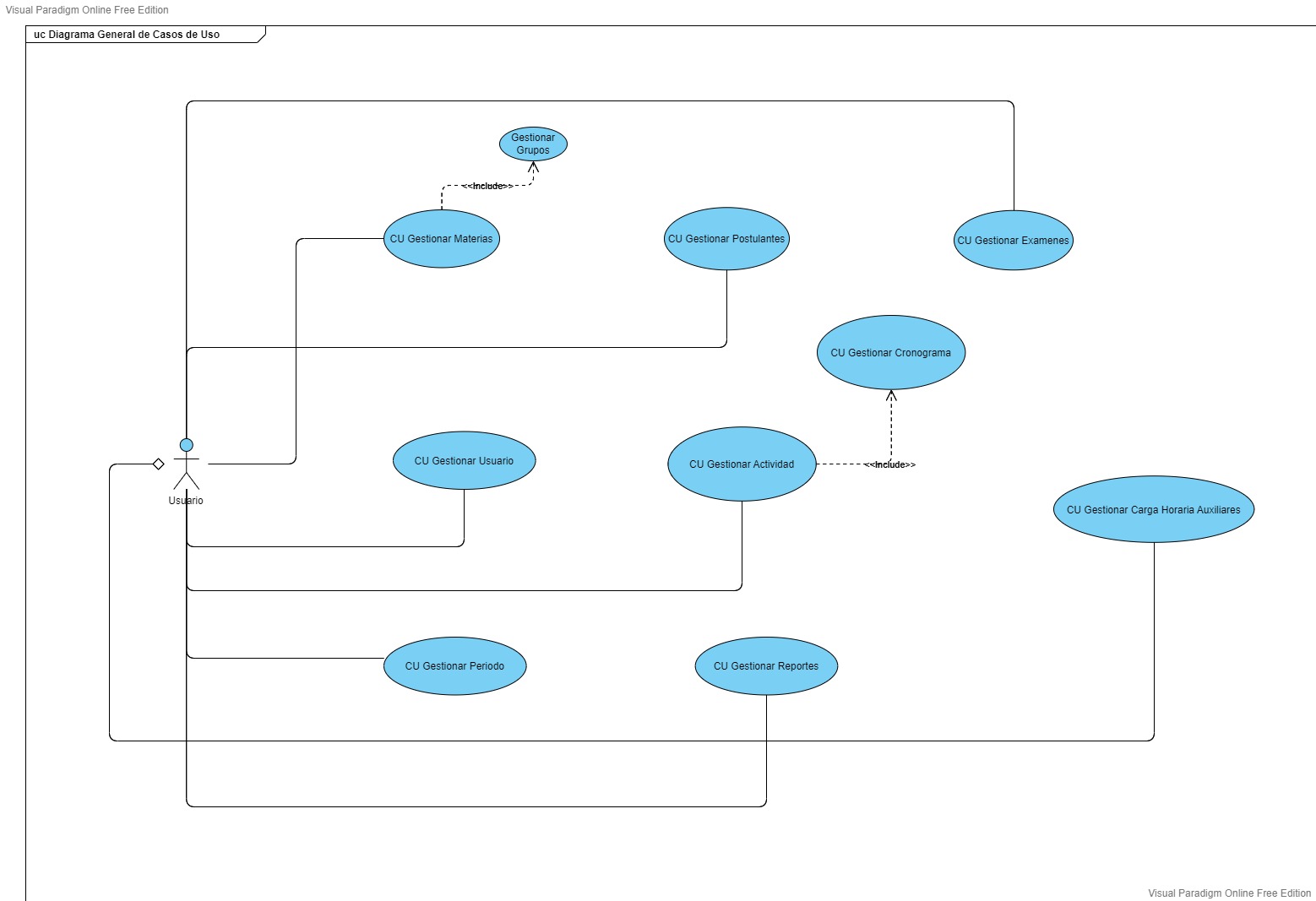
|  |  |
| --- | --- |
| Nombre: Gestionar Carga Horaria Auxiliares | Código: CU 07 |
| Actores: Usuario | |
| Objetivo: Permitir gestionar las cargas horarias de auxiliares. | |
| Descripción: Se podrá:   * Listar: Carga Horaria Auxiliares, Carga Horaria Auxiliares por Nombre de Auxiliar, Carga Horaria Auxiliares por Apellido de Auxiliar, Carga Horaria Auxiliares por Nombre de Docente, Carga Horaria Auxiliares por Apellido de Docente, Carga Horaria Auxiliares por Sigla de Materia, Carga Horaria Auxiliares por Sigla de Grupo, Carga Horaria Auxiliares por Horario Asignado. * Registrar: Carga Horaria Auxiliares. * Eliminar: Carga Horaria Auxiliares. | |
| Flujo Normal:   * Se debe ingresar el comando correspondiente a la acción deseada, con sus parámetros respectivos. * El sistema debe evaluar si la cantidad de parámetros es correcta. * El sistema debe verificar si la sintaxis es correcta. | |
| Excepción:   * Comando desconocido. * Error en cantidad de parámetros. * Parámetros incorrectos. | |

CU8: Gestionar Reportes e Informes



|  |  |
| --- | --- |
| Nombre: Gestionar Reportes e Informes | Código: CU 08 |
| Actores: Usuario | |
| Objetivo: Permitir visualizar los reportes referentes a los módulos del sistema como ser el de postulantes por materia | |
| Descripción: Se podrá:   * Listar la cantidad de postulantes por materia al semestre. * Listar la cantidad y el porcentaje de aprobados por materia. * Listar la cantidad de cargas horarias que se manejan por materias. | |
| Pre-Condición: ninguna. | |
| Flujo Normal:   * Se debe ingresar el comando correspondiente a la acción deseada, con sus parámetros respectivos. * El sistema debe evaluar si la cantidad de parámetros es correcta. * El sistema debe verificar si la sintaxis es correcta. | |
| Excepción:   * Comando desconocido. * Error en cantidad de parámetros. * Parámetros incorrectos. | |
| Post-Condición:   * El administrador podrá visualizar el listado de reporte de acuerdo al comando ingresado. | |

Diagrama General de Casos de Uso



# FLUJO DE TRABAJO: ANÁLISIS

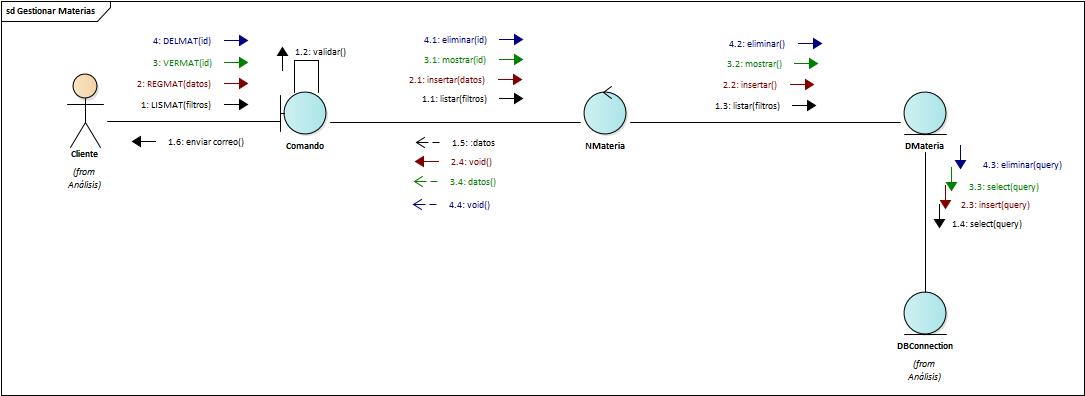
## ANÁLISIS DE ARQUITECTURA

### 4.1.1. IDENTIFICAR PAQUETE

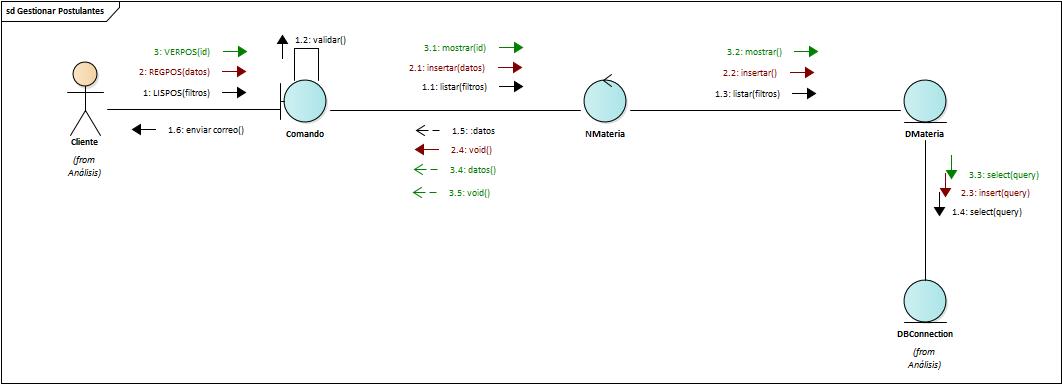
* P1 Comunicaciones
* P2 Negocio
* P3 Datos
* P4 Templates
* P5 Gestion de Auxiliares

## ANÁLISIS DE CASOS DE USO

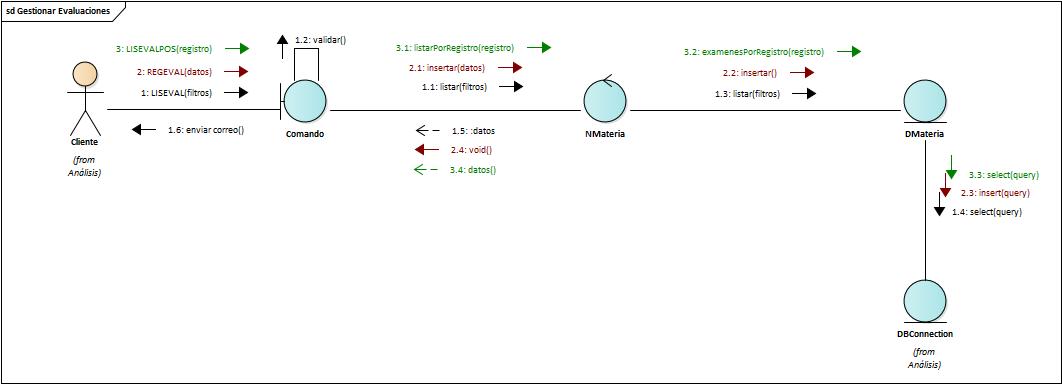
* + 1. DIAGRAMA DE COMUNICACIÓN
* CU2: Gestionar Periodo
* CU4: Gestionar Materias



* CU5: Gestionar Postulantes



* CU6: Gestionar Evaluaciones



* CU8: Gestionar Reportes y Informes
* CU1 Gestionar Usuarios

|  |  |
| --- | --- |
| Comando | Listar Usuario |
| Sintaxis | LISTARUSUARIO |
| Comando | **Listar Usuario Filtrado por Rol ID** |
| Sintaxis | LISTARUSERFILTRADOPORROLID:rol\_id |
| Comando | **Listar Usuario Filtrado por Nombre de Rol** |
| Sintaxis | LISTARUSERFILTRADOPORNOMBREDEROL: rol\_nombre |
| Comando | **Listar Usuario Filtrado por Nombres de Usuario** |
| Sintaxis | LISTARUSERFILTRADOPORNOMBREDEUSUARIO:nombres\_usuario |
| Comando | **Listar Usuario Filtrado por Apellidos de Usuario** |
| Sintaxis | LISTARUSERFILTRADOPORAPELLIDODEUSUARIO:apellidos\_usuario |
| Comando | **Ver Usuario** |
| Sintaxis | VERUSUARIO:usuario\_id |
| Comando | **Dar de Baja al Usuario** |
| Sintaxis | DARDEBAJAUSUARIO:usuario\_id |
| Comando | **Registrar Usuario** |
| Sintaxis | REGISTRARUSUARIO:nombres\_usuario:apellidos\_usuario:genero:fecha\_de\_cumpleaños:url\_foto:carnet\_de\_identidad:direccion\_de\_vivienda:rol\_id:email:password |

* CU2 Gestionar Periodo

|  |  |
| --- | --- |
| Comando | Registrar Periodo |
| Sintaxis | REGPER: nombre\_periodo:semestre\_periodo:año\_periodo |
| Comando | **Editar Periodo** |
| Sintaxis | EDITPER: nombre\_periodo:semestre\_periodo:año\_periodo |
| Comando | **Eliminar Periodo** |
| Sintaxis | DELPER:id\_periodo |
| Comando | **Ver Periodo** |
| Sintaxis | VERPER: id\_periodo |
| Comando | **Listar Periodo** |
| Sintaxis | LISPER: fecha\_inicio\_periodo:fecha\_fin\_periodo |

* CU3 Gestionar Convocatoria

|  |  |
| --- | --- |
| Comando | Listar Convocatoria |
| Sintaxis | LISTARCONVOCATORIA |
| Comando | **Listar Convocatoria Filtrado por Periodo ID** |
| Sintaxis | LISTARCONVOCATORIAFILTRADOPORPERIODOID:periodo\_id |
| Comando | **Listar Convocatoria Filtrado por Nombres de Periodo** |
| Sintaxis | LISTARCONVOCATORIAFILTRADOPORPERIODONOMBRE:nombre\_periodo |
| Comando | **Listar Convocatoria Filtrado por Nombres de Creador** |
| Sintaxis | LISTARCONVOCATORIAFILTRADOPORNOMBRECREADOR:nombres\_creador |
| Comando | **Listar Convocatoria Filtrado por Apellidos de Creador** |
| Sintaxis | LISTARCONVOCATORIAFILTRADOPORAPELLIDOCREADOR:apellidos\_creador |
| Comando | **Listar Convocatoria Filtrado por Creador ID** |
| Sintaxis | LISTARCONVOCATORIAFILTRADOPORIDDELCREADOR:creador\_id |
| Comando | **Ver Convocatoria** |
| Sintaxis | VERCONVOCATORIA:convocatoria\_id |
| Comando | **Dar de Baja a la Convocatoria** |
| Sintaxis | DARDEBAJACONVOCATORIA:convocatoria\_id |
| Comando | **Registrar Convocatoria** |
| Sintaxis | REGISTRARCONVOCATORIA:nombre\_convocatoria:descripción\_convocatoria:autor\_de\_la\_convocatoria:usuario\_id:periodo\_id:  fecha\_inicio1|fecha\_fin1|hora\_inicio1|hora\_fin1:  fecha\_inicio2|fecha\_fin2|hora\_inicio2|hora\_fin2:  ….. |

* CU7 Gestionar Carga Horaria Auxiliar

|  |  |
| --- | --- |
| Comando | Listar Carga Horaria Auxiliar |
| Sintaxis | LISTARCARGAHORARIAAUXILIARES |
| Comando | **Listar Carga Horaria Auxiliar por Nombres de Auxiliar** |
| Sintaxis | LISTARCARGAHORARIAAUXILIARESFILTRADOPORNOMBREDEAUXILIAR:nombres\_auxiliar |
| Comando | **Listar Carga Horaria Auxiliar por Apellidos de Auxiliar** |
| Sintaxis | LISTARCARGAHORARIAAUXILIARESFILTRADOPORAPELLIDODEAUXILIAR:apellidos\_ auxiliar |
| Comando | **Listar Carga Horaria Auxiliar por Nombres de Docente** |
| Sintaxis | LISTARCARGAHORARIAAUXILIARESFILTRADOPORNOMBREDEDOCENTE:nombres\_docente |
| Comando | **Listar Carga Horaria Auxiliar por Apellidos de Docente** |
| Sintaxis | LISTARCARGAHORARIAAUXILIARESFILTRADOPORAPELLIDODEDOCENTE:apellidos\_ docente |
| Comando | **Listar Carga Horaria Auxiliar Filtrado por Sigla de Materia** |
| Sintaxis | LISTARCARGAHORARIAAUXILIARESFILTRADOPORSIGLAMATERIA:sigla\_materia |
| Comando | **Listar Carga Horaria Auxiliar Filtrado por Sigla de Grupo** |
| Sintaxis | LISTARCARGAHORARIAAUXILIARESFILTRADOPORSIGLAGRUPO:sigla\_grupo |
| Comando | **Listar Carga Horaria Auxiliar Filtrado por Horas Asignadas** |
| Sintaxis | LISTARCARGAHORARIAAUXILIARESFILTRADOPORHORASASIGNADAS:horas\_asignadas |
| Comando | **Ver Carga Horaria Auxiliar** |
| Sintaxis | VERCARGAHORARIAAUXILIAR:carga\_horaria\_auxiliar\_id |
| Comando | **Registrar Carga Horaria Auxiliar** |
| Sintaxis | REGISTRARCARGAHORARIAAUXILIAR:grupo\_id:carga\_horaria\_id |

* CU8: Gestionar Reportes e Informes

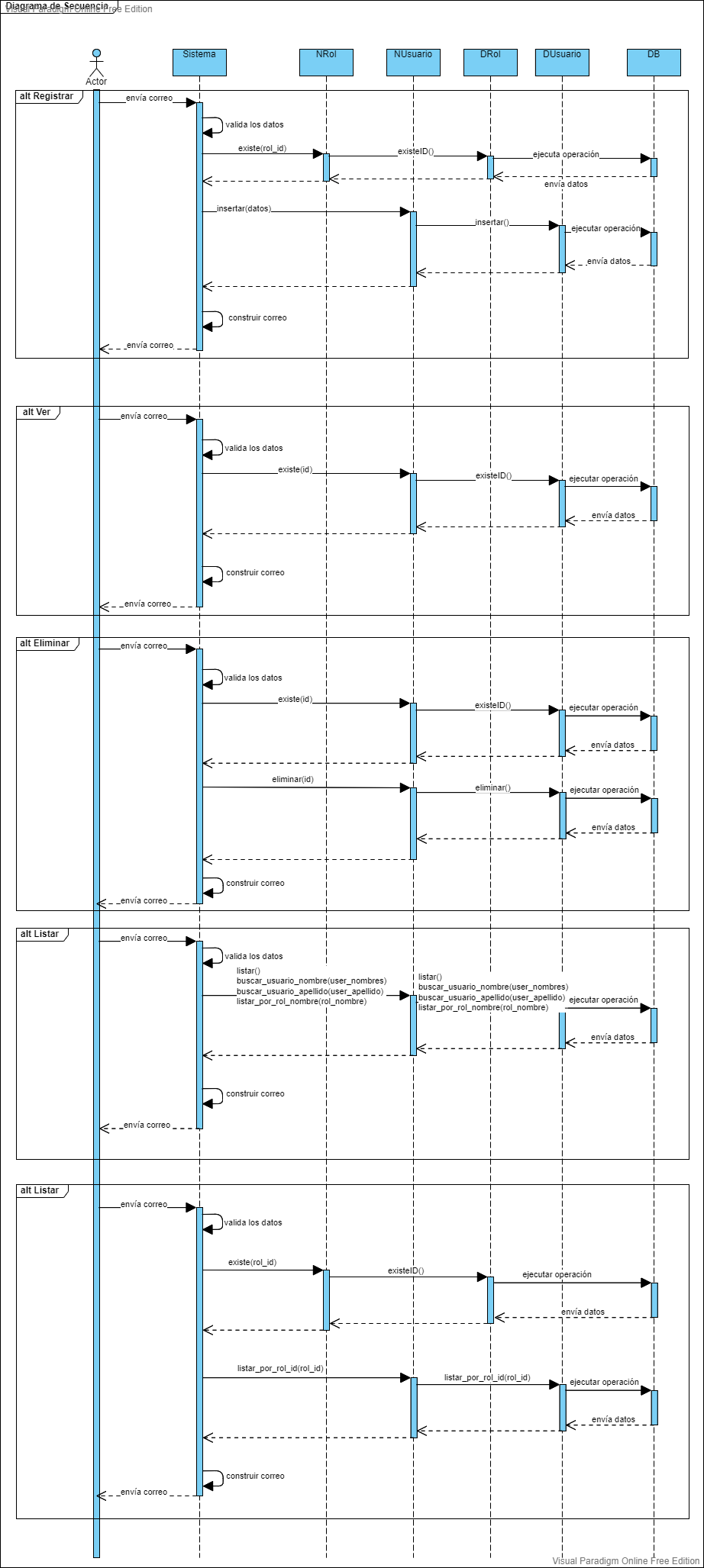
|  |  |
| --- | --- |
| Comando | Reporte Postulante por Periodo |
| Sintaxis | REPORPOSPER: fecha\_inicio : fecha\_fin |
| Comando | **Reporte Auxiliar por Carga Horaria** |
| Sintaxis | REPORAUXCAR: \* (Todos) |
| Comando | **Reporte Grupos por Materia** |
| Sintaxis | REPORMATGRUP: \* (Todos) |
| Comando | **Reporte Aprobados por Materia** |
| Sintaxis | REPORMATAPRO: \* (Todos) |

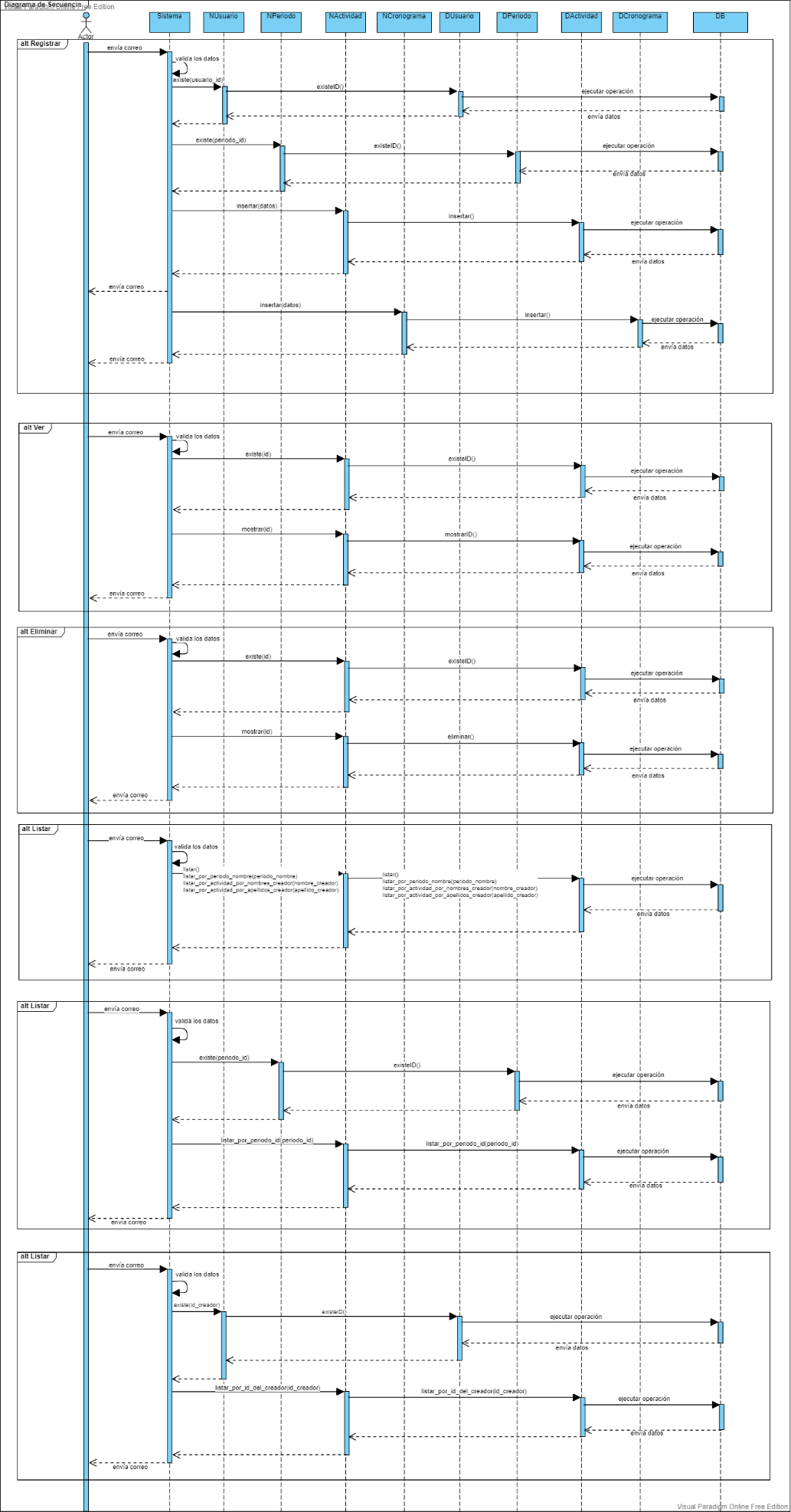
# Flujo de trabajo: Diseño

## 5.1 Diseño de la Arquitectura

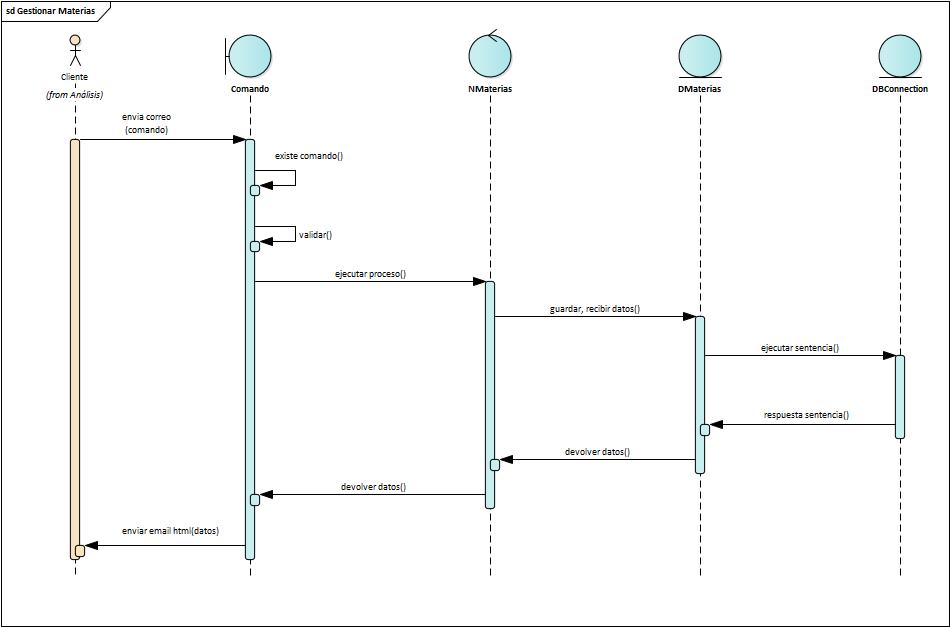


## 5.2 Diseño de Casos de Uso

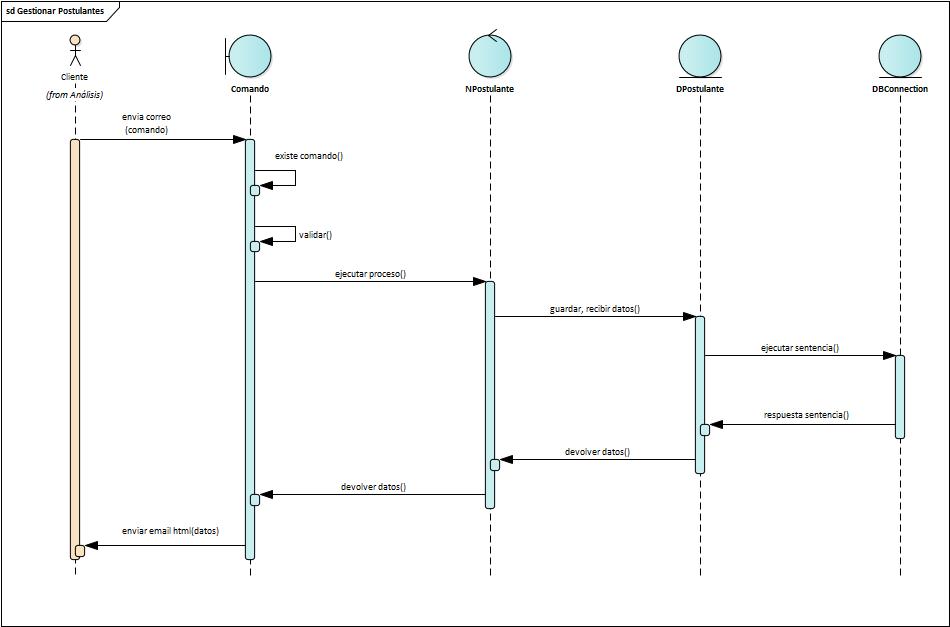
* Gestionar Usuarios
* Gestionar Periodo
* Gestionar Convocatoria



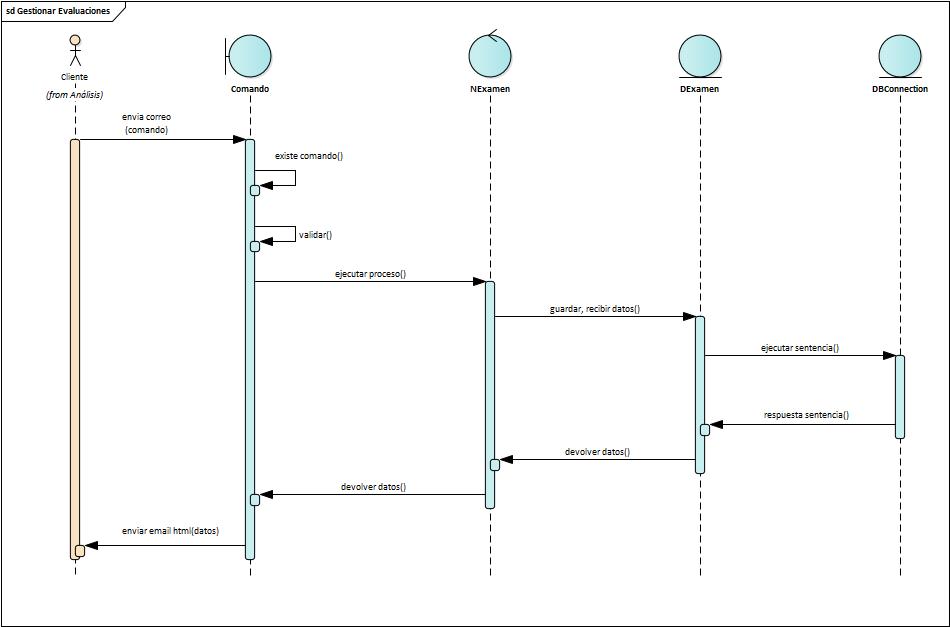
* Gestionar Materias Ofertadas



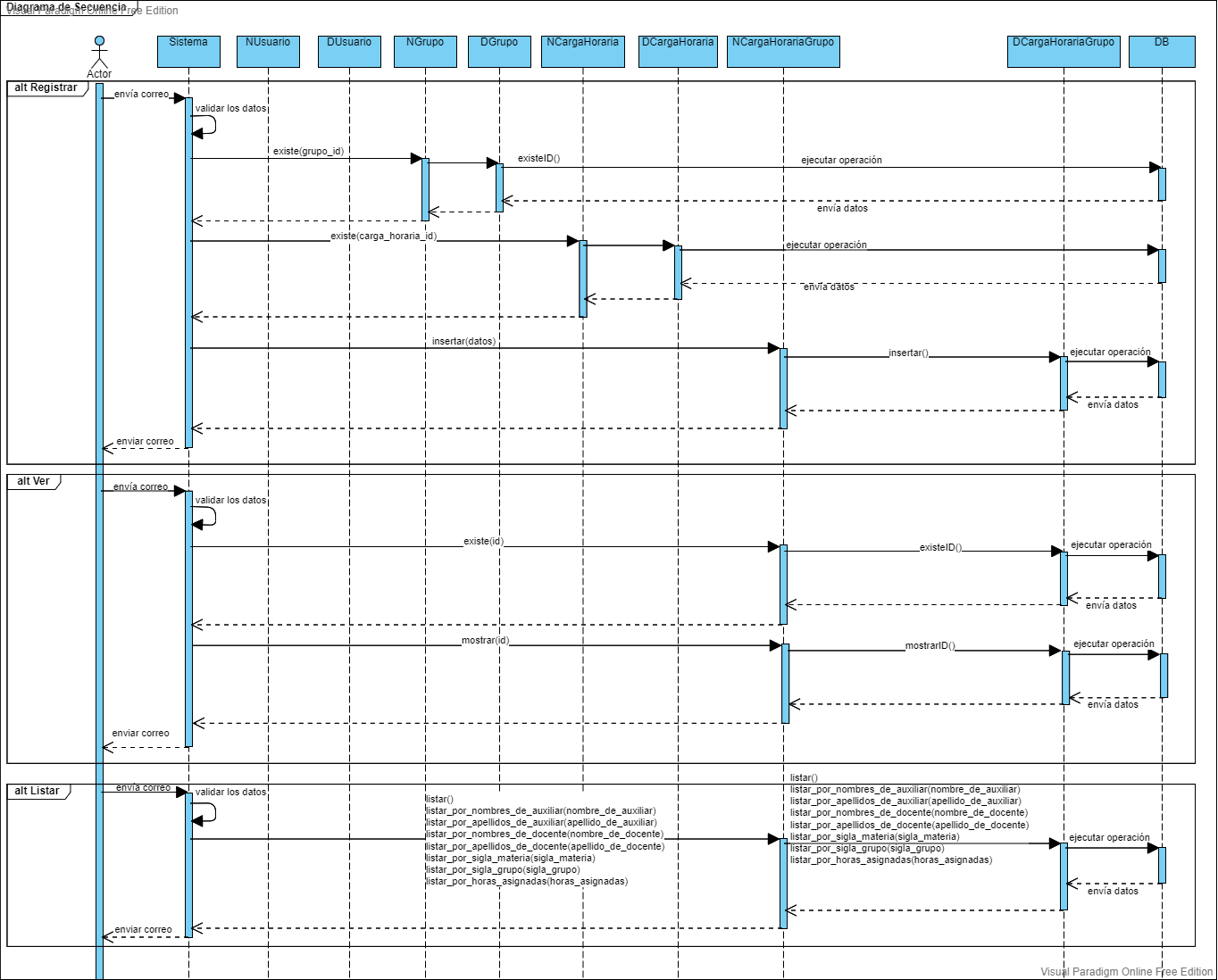
* Gestionar Postulantes



* Gestionar Evaluaciones



* Gestionar Carga Horaria Auxiliares



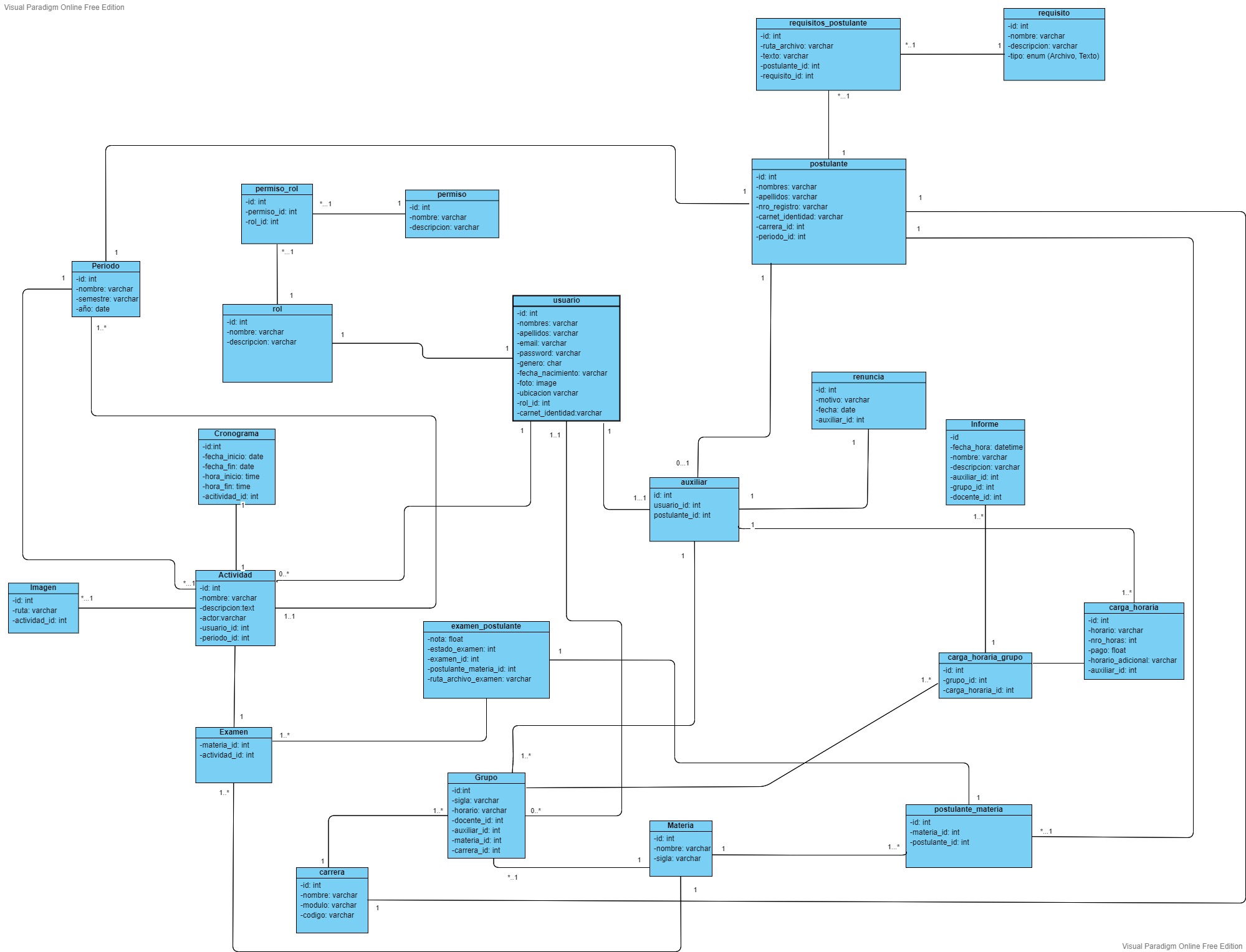
* Gestionar Reportes y Estadísticas



# Diseño de Datos

## Diseño Conceptual

### Diagrama de Clases



## DISEÑO LÓGICO

### MAPEO

## 

|  |  |  |
| --- | --- | --- |
| LLAVE PRIMARIA | LLAVE FORANEA | LLAVE PRIMARIA Y FORANEA |

* Usuario

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | nombres | apellidos | genero | fecha\_de\_nacimiento | foto | direccion\_de\_vivienda |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| carnet\_de\_identidad | email | password | estado | rol\_id |

* Rol

|  |  |  |  |
| --- | --- | --- | --- |
| id | nombre | descripcion | estado |

* Permiso

|  |  |  |  |
| --- | --- | --- | --- |
| id | nombre | descripcion | estado |

* Periodo\_Rol

|  |  |  |  |
| --- | --- | --- | --- |
| id | rol\_id | permiso\_id | estado |

* Periodo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | nombre | semestre | año | estado |

* Postulante

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id | nombres | apellidos | nro\_registro | carnet\_identidad | estado | carrera\_id | periodo\_id |

* Requisito\_Postulante

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | ruta\_archivo | texto | postulante\_id | requisito\_id |

* Requisito

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | nombre | descripcion | tipo | estado |

* Actividad

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id | nombre | descripcion | actor | Usuario\_id | Periodo\_id |

* Cronograma

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id | Fecha\_inicio | Fecha\_fin | Hora\_inicio | Hora\_fin | Actividad\_id |

* Imagen

|  |  |  |
| --- | --- | --- |
| id | ruta | Actividad\_id |

* Auxiliar

|  |  |  |
| --- | --- | --- |
| id | Usuario\_id | Actividad\_id |

* Renuncia

|  |  |  |  |
| --- | --- | --- | --- |
| id | motivo | fecha | usuario\_id |

* Informe

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | Fecha\_hora | nombre | descripcion | Auxiliar\_id | Grupo\_id | Docente\_id |

* Examen

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id | materia\_id | actividad\_id | estado | created\_at | updated\_at |

* Carrera

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | nombre | modulo | codigo | estado | created\_at | updated\_at |

* Grupo

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | sigla | horario | docente\_id | auxiliar\_id | materia\_id | carrera\_id | estado | created\_at | updated\_at |

* Materia

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id | nombre | sigla | estado | created\_at | updated\_at |

* Postulante\_materia

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id | materia\_id | postulante\_id | estado | created\_at | updated\_at |

* Carga\_horario\_grupo

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id | grupo\_id | carga\_horaria\_id | estado | created\_at | updated\_at |

* Carga\_horaria

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | horario | nro\_horas | pago | horario\_adicional | auxiliar\_id | estado | created\_at | updated\_at |

* Examen\_postulante

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | nota | estado\_examen | ruta\_archivo\_examen | examen\_id | postulante\_materia\_id | estado | created\_at | updated\_at |

## DISEÑO FÍSICO

### TABLA DE VOLUMEN

* Tabla Usuario

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Nombres | Cadena | Nombres de usuario | No | No |
| Apellidos | Cadena | Apellidos de postulante | No | No |
| Genero | Char | Genero del usuario | No | No |
| Fecha\_de\_nacimiento | Date | Fecha de nacimiento del usuario | No | No |
| Foto | Cadena | Url de la foto del usuario | No | No |
| Direccion\_de\_vivienda | Cadena | Dirección de vivienda del usuario | No | No |
| Carnet\_de\_identidad | Cadena | Carnet de identidad del usuario | No | No |
| Rol\_id | Numero entero | Identificador del rol | No | Foránea |
| Email | Cadena | Correo electrónico del usuario | No | No |
| Password | Cadena | Contraseña del usuario | No | No |
| Estado | Boolean | Estado del usuario si está activo o inactivo | Si | No |

* Tabla Rol

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Nombre | Cadena | Nombres de rol | No | No |
| Descripción | Cadena | Descripción de rol | No | No |
| Estado | Boolean | Estado del rol si está activo o inactivo | Si | No |

* Tabla Permiso

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Nombre | Cadena | Nombres de permiso | No | No |
| Descripción | Cadena | Descripción de permiso | No | No |
| Estado | Boolean | Estado del permiso si está activo o inactivo | Si | No |

* Tabla Permiso\_Rol

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Rol\_id | Numero entero | Identificador del rol | No | Foránea |
| Permiso\_id | Numero entero | Identificador del permiso | No | Foránea |
| Estado | Boolean | Estado del Periodo\_Rol si está activo o inactivo | Si | No |

* Tabla Periodo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Nombre | Cadena | Nombre del periodo | No | No |
| Semestre | Numero entero | Semestre del permiso | No | No |
| Año | Cadena | Año del periodo |  |  |
| Estado | Boolean | Estado del Periodo\_Rol si está activo o inactivo | Si | No |

* Tabla Postulante

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Nombres | Cadena | Nombres de postulante | No | No |
| Apellidos | Cadena | Apellidos de postulante | No | No |
| Nro\_registro | Cadena | Nro de registro del postulante | No | No |
| Carnet\_identidad | Cadena | Identificador CI del postulante | No | No |
| Carrera\_id | Numero entero | Identificador de la carrera | No | Foránea |
| Periodo\_id | Numero entero | Identificador del periodo | No | Foránea |
| Estado | Boolean | Estado del postulante si está activo o inactivo | Si | No |

* Tabla Requisito\_Postulante

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Ruta\_archivo | Cadena | Ruta del archivo de los requisitos del postulante | Si | No |
| Texto | Texto | Texto del requisito postulante | Si | No |
| Postulante\_id | Numero entero | Identificador del postulante | No | Foránea |
| Requisito\_id | Numero entero | Identificador del requisito | No | Foránea |
| Estado | Boolean | Estado del requisito si está activo o inactivo | Si | No |

* Tabla Requisito

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Nombre | Cadena | Nombres de requisito | No | No |
| Descripción | Cadena | Descripción de requisito | No | No |
| Tipo | Cadena | Tipo de requisito | No | No |
| Estado | Boolean | Estado del requisito si está activo o inactivo | Si | No |

* Tabla Actividad

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Nombre | Cadena | Nombre de la actividad | No | No |
| Descripcion | Texto | Descripcion de la actividad | No | No |
| Actor | Cadena | Personal a cargo de la actividad | No | No |
| Usuario\_id | Numero entero | Identificación del usuario | No | Foranea |
| Periodo\_id | Numero entero | Identificación del periodo | No | Foranea |

* Tabla Cronograma

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Fecha\_inicio | Fecha | Fecha de inicio del cronograma | No | No |
| Fecha\_fin | Fecha | Fecha de finalizacion del cronograma | No | No |
| Hora\_inicio | Tiempo | Hora de inicio del cronograma | No | No |
| Hora\_fin | Tiempo | Hora de finalizacion del cronograma | No | No |
| Actividad\_id | Numero entero | Identificación de la actividad | No | Foranea |

* Tabla Imagen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Ruta | Cadena | Ruta web de la imagen | No | No |
| actividad\_id | Numero entero | Identificación de la actividad | No | Foranea |

* Tabla Auxiliar

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Usuario\_id | Numero entero | Identificación del usuario | No | Foranea |
| Postulante\_id | Numero entero | Identificación del postulante | No | Foranea |

* Tabla Informe

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Fecha\_hora | Fecha y tiempo | Fecha y hora del informe | No | No |
| Nombre | Cadena | Nombre del informe | No | No |
| Descripcion | Cadena | Descripcion del informe | No | No |
| Auxiliar\_id | Numero entero | Identificacion del auxiliar | No | Foránea |
| Grupo\_id | Numero entero | Identificación del grupo a cargo del auxiliar | No | Foránea |
| Docente\_id | Numero entero | Identificacion del docente | NO | Foránea |

* Tabla Renuncia

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Nulo | Llave |
| Id | Numero entero | Identificador | No | Primaria |
| Motivo | Cadena | Motivo de la renuncia | No | No |
| Fecha | Fecha | Fecha de la renuncia | No | No |
| Auxiliar\_id | Numero entero | Identificación del auxiliar | No | Foránea |

* Examen

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Tamaño | Nulo | Llave |
| id | Entero | Identificador único de registro | 64 bytes | No | Primaria |
| materia\_id | Entero | Identificador de la Materia | 64 bytes | No | Foránea |
| actividad\_id | Entero | Identificador de la Actividad | 64 bytes | No | Foránea |
| estado | Entero | Identificador de estado del registro | 16 bytes | No | No |
| created\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se crea el registro | - | Si | No |
| updated\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se actualiza el registro | - | Si | No |

* Carrera

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Tamaño | Nulo | Llave |
| id | Entero | Identificador único de registro | 64 bytes | No | Primaria |
| nombre | Cadena | Nombre de la carrera | 100 bytes | No | No |
| modulo | Cadena | Numero/Nombre del módulo de la carrera | 100 bytes | No | No |
| codigo | Cadena | Codigo único de la carrera | 20 bytes | No | No |
| estado | Entero | Identificador de estado del registro | 16 bytes | No | No |
| created\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se crea el registro | - | Si | No |
| updated\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se actualiza el registro | - | Si | No |

* Grupo

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Tamaño | Nulo | Llave |
| id | Entero | Identificador único de registro | 64 bytes | No | Primaria |
| sigla | Cadena | Sigla del Grupo | 20 bytes | No | No |
| modulo | Cadena | Numero/Nombre del módulo de la carrera | 100 bytes | No | No |
| horario | Cadena | Horario descriptivo del grupo | 150 bytes | No | No |
| docente\_id | Entero | Identificador del usuario del docente del grupo | 64 bytes | No | Foránea |
| auxiliar\_id | Entero | Identificador del usuario del auxiliar del grupo | 64 bytes | Si | Foránea |
| carrera\_id | Entero | Identificador de la carrerea del grupo | 64 bytes | No | Foránea |
| materia\_id | Entero | Identificador de la materia del grupo | 64 bytes | No | Foránea |
| estado | Entero | Identificador de estado del registro | 16 bytes | No | No |
| created\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se crea el registro | - | Si | No |
| updated\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se actualiza el registro | - | Si | No |

* Materia

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Tamaño | Nulo | Llave |
| id | Entero | Identificador único de registro | 64 bytes | No | Primaria |
| nombre | Cadena | Nombre de la carrera | 100 bytes | No | No |
| sigla | Cadena | Sigla única de la materia | 20 bytes | No | No |
| estado | Entero | Identificador de estado del registro | 16 bytes | No | No |
| created\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se crea el registro | - | Si | No |
| updated\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se actualiza el registro | - | Si | No |

* Postulante Materia

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Tamaño | Nulo | Llave |
| id | Entero | Identificador único de registro | 64 bytes | No | Primaria |
| materia\_id | Entero | Identificador único de la materia | 64 bytes | No | Foránea |
| postulante\_id | Entero | Identificador único del postulante | 64 bytes | No | Foránea |
| estado | Entero | Identificador de estado del registro | 16 bytes | No | No |
| created\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se crea el registro | - | Si | No |
| updated\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se actualiza el registro | - | Si | No |

* Carga Horaria Grupo

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Tamaño | Nulo | Llave |
| id | Entero | Identificador único de registro | 64 bytes | No | Primaria |
| grupo\_id | Entero | Identificador único del grupo | 64 bytes | No | Foránea |
| carga\_horaria\_id | Entero | Identificador único de la carga horaria | 64 bytes | No | Foránea |
| estado | Entero | Identificador de estado del registro | 16 bytes | No | No |
| created\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se crea el registro | - | Si | No |
| updated\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se actualiza el registro | - | Si | No |

* Carga Horaria

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Tamaño | Nulo | Llave |
| id | Entero | Identificador único de registro | 64 bytes | No | Primaria |
| horario | Cadena | Horario descriptivo de la carga horaria | 150 bytes | No | No |
| nro\_horas | Entero | Cantidad de horas asignadas a la carga | 16 bytes | No | No |
| pago | Numerico | El pago asignado a esa carga | 16 bytes | No | No |
| horario\_adicional | Cadena | Horario adicional que se cambié o añada después del registro | 150 bytes | No | No |
| auxiliar\_id | Entero | Identificador del usuario del auxiliar del grupo | 64 bytes | No | Foránea |
| estado | Entero | Identificador de estado del registro | 16 bytes | No | No |
| created\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se crea el registro | - | Si | No |
| updated\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se actualiza el registro | - | Si | No |

* Examen Postulante

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Descripción | Tamaño | Nulo | Llave |
| id | Entero | Identificador único de registro | 64 bytes | No | Primaria |
| nota | Flotante | La nota entre 0 y la máxima nota, asignada por el docente | 53 bytes | No | No |
| estado\_examen | Entero | Número que indica el estado del examen (2 aprobado, 1 desaprobado, 0 sin nota) | 32bytes | No | No |
| examen\_id | Entero | Identificador del examen | 64 bytes | No | Foránea |
| postulante\_materia\_id | Entero | Identificador único de la materia a la que el postulante postuló | 64 bytes | No | Foránea |
| ruta\_archivo\_examen | Texto | La ruta relativa del archivo o foto del examen | 1000 bytes | No | No |
| estado | Entero | Identificador de estado del registro | 16 bytes | No | No |
| created\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se crea el registro | - | Si | No |
| updated\_at | Timestamp | Valor numérico que almacena la fecha y hora en la que se actualiza el registro | - | Si | No |

### Script

CREATE TABLE "actividad" (

"id" int8 NOT NULL DEFAULT nextval('actividad\_id\_seq'::regclass),

"nombre" varchar(100) COLLATE "pg\_catalog"."default" NOT NULL,

"descripcion" text COLLATE "pg\_catalog"."default" NOT NULL,

"actor" varchar(100) COLLATE "pg\_catalog"."default",

"usuario\_id" int8 NOT NULL,

"periodo\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for auxiliar

-- ----------------------------

CREATE TABLE "auxiliar" (

"id" int8 NOT NULL DEFAULT nextval('auxiliar\_id\_seq'::regclass),

"postulante\_id" int8 NOT NULL,

"usuario\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for carga\_horaria

-- ----------------------------

CREATE TABLE "carga\_horaria" (

"id" int8 NOT NULL DEFAULT nextval('carga\_horaria\_id\_seq'::regclass),

"horario" varchar(150) COLLATE "pg\_catalog"."default" NOT NULL,

"nro\_horas" int2 NOT NULL,

"pago" numeric(16,2) NOT NULL,

"horario\_adicional" varchar(150) COLLATE "pg\_catalog"."default" NOT NULL,

"auxiliar\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for carga\_horaria\_grupo

-- ----------------------------

CREATE TABLE "carga\_horaria\_grupo" (

"id" int8 NOT NULL DEFAULT nextval('carga\_horaria\_grupo\_id\_seq'::regclass),

"grupo\_id" int8 NOT NULL,

"carga\_horaria\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for carrera

-- ----------------------------

CREATE TABLE "carrera" (

"id" int8 NOT NULL DEFAULT nextval('carrera\_id\_seq'::regclass),

"nombre" varchar(100) COLLATE "pg\_catalog"."default" NOT NULL,

"modulo" varchar(100) COLLATE "pg\_catalog"."default" NOT NULL,

"codigo" varchar(20) COLLATE "pg\_catalog"."default" NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for cronograma

-- ----------------------------

CREATE TABLE "cronograma" (

"id" int8 NOT NULL DEFAULT nextval('cronograma\_id\_seq'::regclass),

"fecha\_inicio" date NOT NULL,

"fecha\_fin" date NOT NULL,

"hora\_inicio" time(0),

"hora\_fin" time(0),

"actividad\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for examen

-- ----------------------------

CREATE TABLE "examen" (

"id" int8 NOT NULL DEFAULT nextval('examen\_id\_seq'::regclass),

"materia\_id" int8 NOT NULL,

"actividad\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for examen\_postulante

-- ----------------------------

CREATE TABLE "examen\_postulante" (

"id" int8 NOT NULL DEFAULT nextval('examen\_postulante\_id\_seq'::regclass),

"nota" float8 NOT NULL,

"estado\_examen" int4 NOT NULL,

"examen\_id" int8 NOT NULL,

"postulante\_materia\_id" int8 NOT NULL,

"ruta\_archivo\_examen" text COLLATE "pg\_catalog"."default" NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for failed\_jobs

-- ----------------------------

CREATE TABLE "failed\_jobs" (

"id" int8 NOT NULL DEFAULT nextval('failed\_jobs\_id\_seq'::regclass),

"uuid" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"connection" text COLLATE "pg\_catalog"."default" NOT NULL,

"queue" text COLLATE "pg\_catalog"."default" NOT NULL,

"payload" text COLLATE "pg\_catalog"."default" NOT NULL,

"exception" text COLLATE "pg\_catalog"."default" NOT NULL,

"failed\_at" timestamp(0) NOT NULL DEFAULT CURRENT\_TIMESTAMP

)

;

-- ----------------------------

-- Table structure for grupo

-- ----------------------------

CREATE TABLE "grupo" (

"id" int8 NOT NULL DEFAULT nextval('grupo\_id\_seq'::regclass),

"sigla" varchar(20) COLLATE "pg\_catalog"."default" NOT NULL,

"horario" varchar(150) COLLATE "pg\_catalog"."default" NOT NULL,

"docente\_id" int8 NOT NULL,

"auxiliar\_id" int8,

"carrera\_id" int8 NOT NULL,

"materia\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for imagen

-- ----------------------------

CREATE TABLE "imagen" (

"id" int8 NOT NULL DEFAULT nextval('imagen\_id\_seq'::regclass),

"ruta" text COLLATE "pg\_catalog"."default" NOT NULL,

"actividad\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for informe

-- ----------------------------

CREATE TABLE "informe" (

"id" int8 NOT NULL DEFAULT nextval('informe\_id\_seq'::regclass),

"fecha" timestamp(0) NOT NULL,

"nombre" varchar(100) COLLATE "pg\_catalog"."default" NOT NULL,

"descripcion" text COLLATE "pg\_catalog"."default" NOT NULL,

"auxiliar\_id" int8 NOT NULL,

"grupo\_id" int8 NOT NULL,

"docente\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for materia

-- ----------------------------

CREATE TABLE "materia" (

"id" int8 NOT NULL DEFAULT nextval('materia\_id\_seq'::regclass),

"nombre" varchar(100) COLLATE "pg\_catalog"."default" NOT NULL,

"sigla" varchar(20) COLLATE "pg\_catalog"."default" NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for migrations

-- ----------------------------

CREATE TABLE "migrations" (

"id" int4 NOT NULL DEFAULT nextval('migrations\_id\_seq'::regclass),

"migration" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"batch" int4 NOT NULL

)

;

-- ----------------------------

-- Table structure for password\_resets

-- ----------------------------

CREATE TABLE "password\_resets" (

"email" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"token" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"created\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for periodo

-- ----------------------------

CREATE TABLE "periodo" (

"id" int8 NOT NULL DEFAULT nextval('periodo\_id\_seq'::regclass),

"nombre" varchar(100) COLLATE "pg\_catalog"."default" NOT NULL,

"semestre" int2 NOT NULL,

"año" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for permiso

-- ----------------------------

CREATE TABLE "permiso" (

"id" int8 NOT NULL DEFAULT nextval('permiso\_id\_seq'::regclass),

"nombre" varchar(100) COLLATE "pg\_catalog"."default" NOT NULL,

"descripcion" text COLLATE "pg\_catalog"."default" NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for permiso\_rol

-- ----------------------------

CREATE TABLE "permiso\_rol" (

"id" int8 NOT NULL DEFAULT nextval('permiso\_rol\_id\_seq'::regclass),

"rol\_id" int8 NOT NULL,

"permiso\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for personal\_access\_tokens

-- ----------------------------

CREATE TABLE "personal\_access\_tokens" (

"id" int8 NOT NULL DEFAULT nextval('personal\_access\_tokens\_id\_seq'::regclass),

"tokenable\_type" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"tokenable\_id" int8 NOT NULL,

"name" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"token" varchar(64) COLLATE "pg\_catalog"."default" NOT NULL,

"abilities" text COLLATE "pg\_catalog"."default",

"last\_used\_at" timestamp(0),

"expires\_at" timestamp(0),

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for postulante

-- ----------------------------

CREATE TABLE "postulante" (

"id" int8 NOT NULL DEFAULT nextval('postulante\_id\_seq'::regclass),

"nombres" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"apellidos" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"nro\_registro" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"carnet\_identidad" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"carrera\_id" int8 NOT NULL,

"periodo\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for postulante\_materia

-- ----------------------------

CREATE TABLE "postulante\_materia" (

"id" int8 NOT NULL DEFAULT nextval('postulante\_materia\_id\_seq'::regclass),

"materia\_id" int8 NOT NULL,

"postulante\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for renuncia

-- ----------------------------

CREATE TABLE "renuncia" (

"id" int8 NOT NULL DEFAULT nextval('renuncia\_id\_seq'::regclass),

"motivo" varchar(256) COLLATE "pg\_catalog"."default" NOT NULL,

"fecha" date NOT NULL,

"auxiliar\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for requisito

-- ----------------------------

CREATE TABLE "requisito" (

"id" int8 NOT NULL DEFAULT nextval('requisito\_id\_seq'::regclass),

"nombre" varchar(50) COLLATE "pg\_catalog"."default" NOT NULL,

"descripcion" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"tipo" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for requisitos\_postulante

-- ----------------------------

CREATE TABLE "requisitos\_postulante" (

"id" int8 NOT NULL DEFAULT nextval('requisitos\_postulante\_id\_seq'::regclass),

"ruta\_archivo" varchar(255) COLLATE "pg\_catalog"."default",

"texto" varchar(255) COLLATE "pg\_catalog"."default",

"postulante\_id" int8 NOT NULL,

"requisito\_id" int8 NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for rol

-- ----------------------------

CREATE TABLE "rol" (

"id" int8 NOT NULL DEFAULT nextval('rol\_id\_seq'::regclass),

"nombre" varchar(100) COLLATE "pg\_catalog"."default" NOT NULL,

"descripcion" text COLLATE "pg\_catalog"."default" NOT NULL,

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Table structure for users

-- ----------------------------

CREATE TABLE "users" (

"id" int8 NOT NULL DEFAULT nextval('users\_id\_seq1'::regclass),

"nombres" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"apellidos" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"genero" char(1) COLLATE "pg\_catalog"."default" NOT NULL,

"fecha\_de\_nacimiento" date NOT NULL,

"foto" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"direccion\_de\_vivienda" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"carnet\_de\_identidad" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"rol\_id" int8 NOT NULL,

"email" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"email\_verified\_at" timestamp(0),

"password" varchar(255) COLLATE "pg\_catalog"."default" NOT NULL,

"remember\_token" varchar(100) COLLATE "pg\_catalog"."default",

"estado" int2 NOT NULL DEFAULT '1'::smallint,

"created\_at" timestamp(0),

"updated\_at" timestamp(0)

)

;

-- ----------------------------

-- Primary Key structure for table actividad

-- ----------------------------

ALTER TABLE "actividad" ADD CONSTRAINT "actividad\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table auxiliar

-- ----------------------------

ALTER TABLE "auxiliar" ADD CONSTRAINT "auxiliar\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table carga\_horaria

-- ----------------------------

ALTER TABLE "carga\_horaria" ADD CONSTRAINT "carga\_horaria\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table carga\_horaria\_grupo

-- ----------------------------

ALTER TABLE "carga\_horaria\_grupo" ADD CONSTRAINT "carga\_horaria\_grupo\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Uniques structure for table carrera

-- ----------------------------

ALTER TABLE "carrera" ADD CONSTRAINT "carrera\_codigo\_unique" UNIQUE ("codigo");

-- ----------------------------

-- Primary Key structure for table carrera

-- ----------------------------

ALTER TABLE "carrera" ADD CONSTRAINT "carrera\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table cronograma

-- ----------------------------

ALTER TABLE "cronograma" ADD CONSTRAINT "cronograma\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table examen

-- ----------------------------

ALTER TABLE "examen" ADD CONSTRAINT "examen\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table examen\_postulante

-- ----------------------------

ALTER TABLE "examen\_postulante" ADD CONSTRAINT "examen\_postulante\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Uniques structure for table failed\_jobs

-- ----------------------------

ALTER TABLE "failed\_jobs" ADD CONSTRAINT "failed\_jobs\_uuid\_unique" UNIQUE ("uuid");

-- ----------------------------

-- Primary Key structure for table failed\_jobs

-- ----------------------------

ALTER TABLE "failed\_jobs" ADD CONSTRAINT "failed\_jobs\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Uniques structure for table grupo

-- ----------------------------

ALTER TABLE "grupo" ADD CONSTRAINT "grupo\_sigla\_materia\_id\_unique" UNIQUE ("sigla", "materia\_id");

-- ----------------------------

-- Primary Key structure for table grupo

-- ----------------------------

ALTER TABLE "grupo" ADD CONSTRAINT "grupo\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table imagen

-- ----------------------------

ALTER TABLE "imagen" ADD CONSTRAINT "imagen\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table informe

-- ----------------------------

ALTER TABLE "informe" ADD CONSTRAINT "informe\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Uniques structure for table materia

-- ----------------------------

ALTER TABLE "materia" ADD CONSTRAINT "materia\_sigla\_unique" UNIQUE ("sigla");

-- ----------------------------

-- Primary Key structure for table materia

-- ----------------------------

ALTER TABLE "materia" ADD CONSTRAINT "materia\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table migrations

-- ----------------------------

ALTER TABLE "migrations" ADD CONSTRAINT "migrations\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Indexes structure for table password\_resets

-- ----------------------------

CREATE INDEX "password\_resets\_email\_index" ON "password\_resets" USING btree (

"email" COLLATE "pg\_catalog"."default" "pg\_catalog"."text\_ops" ASC NULLS LAST

);

-- ----------------------------

-- Uniques structure for table periodo

-- ----------------------------

ALTER TABLE "periodo" ADD CONSTRAINT "periodo\_semestre\_año\_unique" UNIQUE ("semestre", "año");

-- ----------------------------

-- Primary Key structure for table periodo

-- ----------------------------

ALTER TABLE "periodo" ADD CONSTRAINT "periodo\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table permiso

-- ----------------------------

ALTER TABLE "permiso" ADD CONSTRAINT "permiso\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table permiso\_rol

-- ----------------------------

ALTER TABLE "permiso\_rol" ADD CONSTRAINT "permiso\_rol\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Indexes structure for table personal\_access\_tokens

-- ----------------------------

CREATE INDEX "personal\_access\_tokens\_tokenable\_type\_tokenable\_id\_index" ON "personal\_access\_tokens" USING btree (

"tokenable\_type" COLLATE "pg\_catalog"."default" "pg\_catalog"."text\_ops" ASC NULLS LAST,

"tokenable\_id" "pg\_catalog"."int8\_ops" ASC NULLS LAST

);

-- ----------------------------

-- Uniques structure for table personal\_access\_tokens

-- ----------------------------

ALTER TABLE "personal\_access\_tokens" ADD CONSTRAINT "personal\_access\_tokens\_token\_unique" UNIQUE ("token");

-- ----------------------------

-- Primary Key structure for table personal\_access\_tokens

-- ----------------------------

ALTER TABLE "personal\_access\_tokens" ADD CONSTRAINT "personal\_access\_tokens\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Uniques structure for table postulante

-- ----------------------------

ALTER TABLE "postulante" ADD CONSTRAINT "postulante\_nro\_registro\_periodo\_id\_unique" UNIQUE ("nro\_registro", "periodo\_id");

-- ----------------------------

-- Primary Key structure for table postulante

-- ----------------------------

ALTER TABLE "postulante" ADD CONSTRAINT "postulante\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Uniques structure for table postulante\_materia

-- ----------------------------

ALTER TABLE "postulante\_materia" ADD CONSTRAINT "postulante\_materia\_materia\_id\_postulante\_id\_unique" UNIQUE ("materia\_id", "postulante\_id");

-- ----------------------------

-- Primary Key structure for table postulante\_materia

-- ----------------------------

ALTER TABLE "postulante\_materia" ADD CONSTRAINT "postulante\_materia\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table renuncia

-- ----------------------------

ALTER TABLE "renuncia" ADD CONSTRAINT "renuncia\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Checks structure for table requisito

-- ----------------------------

ALTER TABLE "requisito" ADD CONSTRAINT "requisito\_tipo\_check" CHECK (tipo::text = ANY (ARRAY['Archivo'::character varying, 'Texto'::character varying]::text[]));

-- ----------------------------

-- Primary Key structure for table requisito

-- ----------------------------

ALTER TABLE "requisito" ADD CONSTRAINT "requisito\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Uniques structure for table requisitos\_postulante

-- ----------------------------

ALTER TABLE "requisitos\_postulante" ADD CONSTRAINT "requisitos\_postulante\_postulante\_id\_requisito\_id\_unique" UNIQUE ("postulante\_id", "requisito\_id");

-- ----------------------------

-- Primary Key structure for table requisitos\_postulante

-- ----------------------------

ALTER TABLE "requisitos\_postulante" ADD CONSTRAINT "requisitos\_postulante\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Primary Key structure for table rol

-- ----------------------------

ALTER TABLE "rol" ADD CONSTRAINT "rol\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Uniques structure for table users

-- ----------------------------

ALTER TABLE "users" ADD CONSTRAINT "users\_email\_unique" UNIQUE ("email");

-- ----------------------------

-- Primary Key structure for table users

-- ----------------------------

ALTER TABLE "users" ADD CONSTRAINT "users\_pkey" PRIMARY KEY ("id");

-- ----------------------------

-- Foreign Keys structure for table actividad

-- ----------------------------

ALTER TABLE "actividad" ADD CONSTRAINT "actividad\_periodo\_id\_foreign" FOREIGN KEY ("periodo\_id") REFERENCES "periodo" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "actividad" ADD CONSTRAINT "actividad\_usuario\_id\_foreign" FOREIGN KEY ("usuario\_id") REFERENCES "users" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table auxiliar

-- ----------------------------

ALTER TABLE "auxiliar" ADD CONSTRAINT "auxiliar\_postulante\_id\_foreign" FOREIGN KEY ("postulante\_id") REFERENCES "postulante" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "auxiliar" ADD CONSTRAINT "auxiliar\_usuario\_id\_foreign" FOREIGN KEY ("usuario\_id") REFERENCES "users" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table carga\_horaria

-- ----------------------------

ALTER TABLE "carga\_horaria" ADD CONSTRAINT "carga\_horaria\_auxiliar\_id\_foreign" FOREIGN KEY ("auxiliar\_id") REFERENCES "auxiliar" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table carga\_horaria\_grupo

-- ----------------------------

ALTER TABLE "carga\_horaria\_grupo" ADD CONSTRAINT "carga\_horaria\_grupo\_carga\_horaria\_id\_foreign" FOREIGN KEY ("carga\_horaria\_id") REFERENCES "carga\_horaria" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "carga\_horaria\_grupo" ADD CONSTRAINT "carga\_horaria\_grupo\_grupo\_id\_foreign" FOREIGN KEY ("grupo\_id") REFERENCES "grupo" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table cronograma

-- ----------------------------

ALTER TABLE "cronograma" ADD CONSTRAINT "cronograma\_actividad\_id\_foreign" FOREIGN KEY ("actividad\_id") REFERENCES "actividad" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table examen

-- ----------------------------

ALTER TABLE "examen" ADD CONSTRAINT "examen\_actividad\_id\_foreign" FOREIGN KEY ("actividad\_id") REFERENCES "actividad" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "examen" ADD CONSTRAINT "examen\_materia\_id\_foreign" FOREIGN KEY ("materia\_id") REFERENCES "materia" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table examen\_postulante

-- ----------------------------

ALTER TABLE "examen\_postulante" ADD CONSTRAINT "examen\_postulante\_examen\_id\_foreign" FOREIGN KEY ("examen\_id") REFERENCES "examen" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "examen\_postulante" ADD CONSTRAINT "examen\_postulante\_postulante\_materia\_id\_foreign" FOREIGN KEY ("postulante\_materia\_id") REFERENCES "postulante\_materia" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table grupo

-- ----------------------------

ALTER TABLE "grupo" ADD CONSTRAINT "grupo\_auxiliar\_id\_foreign" FOREIGN KEY ("auxiliar\_id") REFERENCES "auxiliar" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "grupo" ADD CONSTRAINT "grupo\_carrera\_id\_foreign" FOREIGN KEY ("carrera\_id") REFERENCES "carrera" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "grupo" ADD CONSTRAINT "grupo\_docente\_id\_foreign" FOREIGN KEY ("docente\_id") REFERENCES "users" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "grupo" ADD CONSTRAINT "grupo\_materia\_id\_foreign" FOREIGN KEY ("materia\_id") REFERENCES "materia" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table imagen

-- ----------------------------

ALTER TABLE "imagen" ADD CONSTRAINT "imagen\_actividad\_id\_foreign" FOREIGN KEY ("actividad\_id") REFERENCES "actividad" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table informe

-- ----------------------------

ALTER TABLE "informe" ADD CONSTRAINT "informe\_auxiliar\_id\_foreign" FOREIGN KEY ("auxiliar\_id") REFERENCES "auxiliar" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "informe" ADD CONSTRAINT "informe\_docente\_id\_foreign" FOREIGN KEY ("docente\_id") REFERENCES "users" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "informe" ADD CONSTRAINT "informe\_grupo\_id\_foreign" FOREIGN KEY ("grupo\_id") REFERENCES "grupo" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table permiso\_rol

-- ----------------------------

ALTER TABLE "permiso\_rol" ADD CONSTRAINT "permiso\_rol\_permiso\_id\_foreign" FOREIGN KEY ("permiso\_id") REFERENCES "permiso" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "permiso\_rol" ADD CONSTRAINT "permiso\_rol\_rol\_id\_foreign" FOREIGN KEY ("rol\_id") REFERENCES "rol" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table postulante

-- ----------------------------

ALTER TABLE "postulante" ADD CONSTRAINT "postulante\_carrera\_id\_foreign" FOREIGN KEY ("carrera\_id") REFERENCES "carrera" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "postulante" ADD CONSTRAINT "postulante\_periodo\_id\_foreign" FOREIGN KEY ("periodo\_id") REFERENCES "periodo" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table postulante\_materia

-- ----------------------------

ALTER TABLE "postulante\_materia" ADD CONSTRAINT "postulante\_materia\_materia\_id\_foreign" FOREIGN KEY ("materia\_id") REFERENCES "materia" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "postulante\_materia" ADD CONSTRAINT "postulante\_materia\_postulante\_id\_foreign" FOREIGN KEY ("postulante\_id") REFERENCES "postulante" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table renuncia

-- ----------------------------

ALTER TABLE "renuncia" ADD CONSTRAINT "renuncia\_auxiliar\_id\_foreign" FOREIGN KEY ("auxiliar\_id") REFERENCES "auxiliar" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table requisitos\_postulante

-- ----------------------------

ALTER TABLE "requisitos\_postulante" ADD CONSTRAINT "requisitos\_postulante\_postulante\_id\_foreign" FOREIGN KEY ("postulante\_id") REFERENCES "postulante" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE "requisitos\_postulante" ADD CONSTRAINT "requisitos\_postulante\_requisito\_id\_foreign" FOREIGN KEY ("requisito\_id") REFERENCES "requisito" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

-- ----------------------------

-- Foreign Keys structure for table users

-- ----------------------------

ALTER TABLE "users" ADD CONSTRAINT "users\_rol\_id\_foreign" FOREIGN KEY ("rol\_id") REFERENCES "rol" ("id") ON DELETE NO ACTION ON UPDATE NO ACTION;

# CONCLUSIÓN

El sistema diseñado cumple con los requerimientos propuestos por el cliente en pos del beneficio de los auxiliares, en este documento se encuentra la información necesaria para comprender el funcionamiento del sistema y hacer uso del mismo. Gracias al uso de las bondades del correo electrónico, pudimos desarrollar este sistema que atiende las necesidades que se tenían los auxiliares.

# RECOMENDACIÓN

Se recomienda que, para tener comprensión del funcionamiento de este proyecto, se lea este documento en cuestión, mismo en el cual se especifican las funcionalidades y como ejecutar estas mediante comandos que se envían por correo electrónico