**UNIVERSIDAD AUTONOMA GABRIEL**

**RENÉ MORENO**

**FACULTAD DE INGENIERIA EN CIENCIAS DE LA COMPUTACION Y**

**TELECOMUNICACIONES**



**SISTEMA PARA LA GESTION INVENTARIOS DE LA “CONSTRUCTORA GENEX”**

**Docente**:

* Ing. Balcazar Veizaga Evanz

**Materia**:

* INF-513

**Estudiantes:**

* Aruquipa Fernández Melissa 215151471 **SA**
* Carachuyo Limachi Ana Silvia 215152808 **SA**
* Torrez Condarco Carlos Eduardo 213125757 **SC** **Grupo**: 02

Santa Cruz de la Sierra – Estado plurinacional de Bolivia

2021

[Contenido CAPÍTULO I: PERFIL 5](#_Toc110923)

[1.1 INTRODUCCION 5](#_Toc110924)

[1.2. ANTECEDENTES Y JUSTIFICACION 5](#_Toc110925)

[1.2.1. ANTECEDENTES 5](#_Toc110926)

[1.2.2 JUSTIFICACION 8](#_Toc110927)

[1.3 EL PROBLEMA 8](#_Toc110928)

[1.3.1 SITUACION PROBLEMÁTICA 8](#_Toc110929)

[1.4 OBJETIVOS 8](#_Toc110930)

[1.4.1 OBJETIVO GENERAL 8](#_Toc110931)

[1.4.2 OBJETIVOS ESPECÍFICOS 9](#_Toc110932)

[1.5 METODOLOGÍA 9](#_Toc110933)

[1.6 ALCANCE 12](#_Toc110934)

[1.6.1 REQUISITOS FUNCIONALES 12](#_Toc110935)

[1.6.2 REQUISITOS NO FUNCIONALES 13](#_Toc110936)

[CAPITULO II: SISTEMA DE CORREO 13](#_Toc110937)

[2.1 MARCO TEÓRICO 13](#_Toc110938)

[2.1.1. CORREO ELECTRONICO 13](#_Toc110939)

[2.1.2 PROTOCOLO SMTP 15](#_Toc110940)

[2.1.3 MIME 17](#_Toc110941)

[2.1.4 POP3 18](#_Toc110942)

[2.1.5 SENDMAIL 19](#_Toc110943)

[CAPITULO III: DESARROLLO 21](#_Toc110944)

[3.1 FLUJO DE TRABAJO: CAPTURA DE REQUISITOS 21](#_Toc110945)

[3.1.1 IDENTIFICAR ACTORES Y CASOS DE USOS 21](#_Toc110946)

[3.1.2 DESCRIPCIÓN DE CASOS DE USOS 22](#_Toc110947)

[3.1.3. Modelo general de casos de uso 30](#_Toc110948)

[CAPITULO IV: ANÁLISIS 31](#_Toc110949)

[4.1 ANÁLISIS DE LA ARQUITECTURA 31](#_Toc110950)

[4.1.1 IDENTIFICACIÓN DE PAQUETES 31](#_Toc110951)

[4.1.2 RELACIÓN DE PAQUETES Y CASOS DE USO 31](#_Toc110952)

[4.2 ANÁLISIS DE CASOS DE USO 33](#_Toc110953)

[4.3 ANÁLISIS DE PAQUETES 37](#_Toc110954)

[CAPITULO V: DISEÑO 37](#_Toc110955)

[5.1 DISEÑO DE LA ARQUITECTURA 37](#_Toc110956)

[5.1.1 ARQUITECTURA LÓGICA 37](#_Toc110957)

[5.2 DISEÑO DE LA BASE DE DATOS 38](#_Toc110958)

[5.2.1 DISEÑO CONCEPTUAL 39](#_Toc110959)

[5.2.2 DISEÑO LÓGICO 40](#_Toc110960)

[5.2.3 DISEÑO FÍSICO 41](#_Toc110961)

[5.2.4. SCRIPT 44](#_Toc110962)

[5.2.4 DISEÑO DE CASOS DE USO 50](#_Toc110963)

[5.2.4.1 DIAGRAMAS DE SECUENCIA 50](#_Toc110964)

[CAPÍTULO VI: IMPLEMENTACIÓN 58](#_Toc110965)

[6.1 PLATAFORMA DE DESARROLLO DE SOFTWARE 58](#_Toc110966)

[6.1.1 LENGUAJE DE PROGRAMACIÓN 58](#_Toc110967)

[6.1.2 BASE DE DATOS 58](#_Toc110968)

[6.1.3 SISTEMAS OPERATIVOS 59](#_Toc110969)

[6.1.4 Otros 59](#_Toc110970)

[BIBLIOGRAFÍA 59](#_Toc110971)

**FIGURA 1:** INSTALACIONES DE LA CONSTRUCTORA GENEX ....................................... 6 **FIGURA 2:** UBICACIÓN SATÉLITE DE LA CONSTRUCTORA GENEX . **Error! Bookmark not defined.**

**FIGURA 3:** ESTRUCTURA ORGANIZACIONAL DE CONSTRUCTORA GENEX .............. 7

**FIGURA 4:** ESQUEMA DE FUNCIONAMIENTO DE CORREO ELECTRÓNICO .............. 14

**FIGURA 5:** ESQUEMA DE FUNCIONAMIENTO DEL PROTOCOLO SMTP ..................... 16

**FIGURA 6:** DIAGRAMA GENERAL DE CASOS DE USO .................................................... 30

**FIGURA 7:** IDENTIFICACIÓN DE PAQUETES ..................................................................... 31

**FIGURA 8:** RELACIÓN PAQUETE ADMINISTRACIÓN CON CU1, CU8........................... 31

**FIGURA 9:** RELACIÓN PAQUETE ADMINISTRACIÓN CON CU3, CU4. CU5 ................. 32

**FIGURA 10:** RELACIÓN PAQUETE ADMINISTRACIÓN CON CU7, CU6, CU2 ............... 32

**TABLA 1**: COMANDOS POP3 .................................................................................................. 19

**TABLA 2:** PRIORIZACIÓN DE CASOS DE USO .................................................................... 21

**TABLA 3:** DESCRIPCIÓN CU1: GESTIONAR USUARIO ..................................................... 22

**TABLA 4:** DESCRIPCIÓN CU2: GESTIONAR PROYECTO .................................................. 23

**TABLA 5:** DESCRIPCIÓN CU3: GESTIONAR MATERIAL .................................................. 24

**TABLA 6:** DESCRIPCIÓN CU4: GESTIONAR USUARIO ..................................................... 25

**TABLA 7:** DESCRIPCIÓN CU5: GESTIONAR MATERIAL .................................................. 26

**TABLA 8:** DESCRIPCIÓN CU6: REGISTRAR PRODUCCIÓN ............................................. 27

**TABLA 9:** DESCRIPCIÓN CU7: REGISTRAR DOTACIÓN .................................................. 28

**TABLA 10:** DESCRIPCIÓN CU8: REPORTES Y ESTADÍSTICAS ........................................ 29 CAPÍTULO I: PERFIL

## 1.1 INTRODUCCION

Vivimos en un mundo en el que las nuevas tecnologías están presentes en prácticamente todas las tareas cotidianas que realizamos a diario. De todas ellas, no podemos decir que las tecnologías informáticas sean las más importantes en nuestro día a día, pero por su parte, nadie puede negar que nos acercan y nos hacen más fáciles y cómodas muchas acciones. Las bases de datos son el método preferido para el almacenamiento estructurado de datos. Desde las grandes aplicaciones multiusuario, hasta los teléfonos móviles y las agendas electrónicas utilizan tecnología de bases de datos para asegurar la integridad de los datos y facilitar la labor tanto de usuarios como de los programadores que las desarrollaron. Sin embargo, es imprescindible conocer los aspectos clave de cada uno de estos temas para tener éxito en cualquier proyecto que implique trabajar con bases de datos. En este contexto, surge la necesidad de “La Constructora GENEX” de gestionar el inventario de sus almacenes. En este sistema de inventarios se busca que el diseño a implementar cumpla con la necesidad de administrar, controlar los materiales, herramientas, dotación y proyecto tanto como el egreso (entrega) e ingreso de material. Administrar las distintas herramientas que se encuentran presente en la misma siendo estas prestadas, devueltas o ingresadas a la empresa, controlar la dotación de equipos para los trabajadores y gestionar los proyecto que lleva a cabo la empresa “Genex” también tomando en cuenta la producción de prefabricado que se hace diariamente. Se busca tener un orden de toda la información en el inventario que se encuentra a disposición del encargado de la bodega dando facilidad de acceso a la información tanto como la ubicación, cantidad, disposición de materiales en los distintos depósitos, de manera que el personal pueda manejarlo eficientemente y sin complicaciones.

## 1.2. ANTECEDENTES Y JUSTIFICACION

#### 1.2.1. ANTECEDENTES

1.2.1.1 HISTORIA

Constructora Genex Ltda. Es una empresa boliviana constituida el 29 de octubre de 1979 en la ciudad de Santa Cruz de la Sierra.

De reconocida trayectoria en el rubro de la construcción y creadora de un centenar de obras que han contribuido a mejorar la calidad de vida de los bolivianos. A través de los años Genex ha ido posicionándose como uno de los actores más importantes del sector en el país, superando, con éxito, los permanentes desafíos que se ha trazado a lo largo de sus más de 30 años de existencia.

Gracias a su equipo humano de excelencia, inspirado en sólidos valores y movido por la más alta responsabilidad, nuestra empresa ha consolidado su éxito en los diferentes rubros de la construcción.

1.2.1.2 MISION

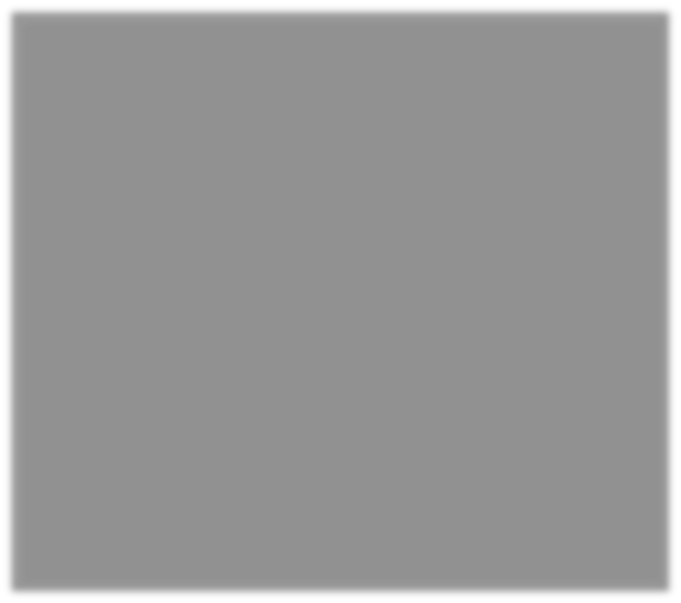
Somos una empresa líder en la construcción. Estamos capacitados y comprometidos para brindar a nuestros clientes calidad, servicio, confianza, seguridad y valorización, logrando sostenibilidad económica y social.

1.2.1.3 VISION

Continuaremos siendo líderes en la construcción y edificando las obras más importantes del país, contando con el apoyo humano y tecnológico que nos permita superar las expectativas de nuestros clientes, lograremos que nos recuerden siempre por el profesionalismo, seriedad y respaldo que les brindemos.

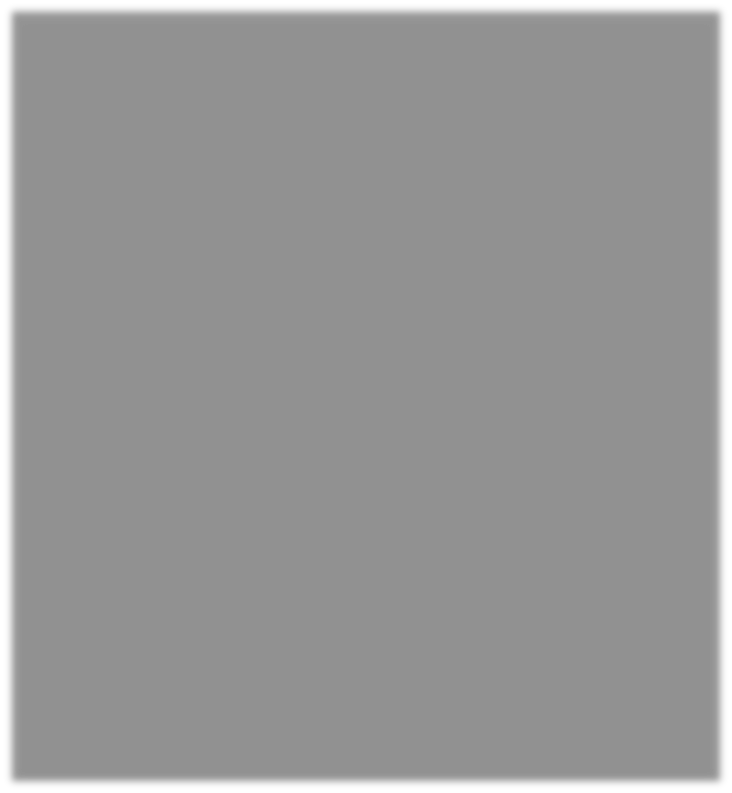
1.2.1.4 UBICACIÓN

La empresa se encuentra ubicada en: Santa Cruz de la Sierra, Séptimo anillo, Av. Santos Dumont, Av. Marcelo Quiroga Santa Cruz.



***Figura 1:*** *Instalaciones de la Constructora Genex*

***Fuente:*** *fotografía propia (2021)*



***Figura 2:*** *Ubicación satélite de la constructora Genex*

***Fuente:*** *Google Maps (2021).*

1.2.1.5

ESTRUCTURA ORGANIZACIONAL

Edgar Rojas

Gerente General

Ing. Eduardo Suarez

Encargado de toda el

área de bodegas

Fernando Camacho

Bodeguero

Ing. Franz Suarez

Encargado de obras

Lic. Jose Antonio

Mercado Banegas

Jefe de proyectos

***Figura 3:*** *Estructura organizacional de Constructora Genex*

***Fuente****: Creación propia (2021).*

#### 1.2.2 JUSTIFICACION

La información del inventario en la empresa se maneja en papel siendo posible que la información se pueda perder, debido a esto es necesario manejar un sistema de información que pueda satisfacer las necesidades que requiere éste. Es viable diseñar un sistema de información para almacenar los datos porque el manejo de los inventarios es voluminoso.

El mantener registrado cada préstamo o saber la información de la cantidad de materiales o herramientas que tiene la empresa y este proyecto resolvería todo eso ya que pensamos simplificar o mejorar todo lo relacionado al almacén o inventario que tiene una empresa.

## 1.3 EL PROBLEMA

#### 1.3.1 SITUACION PROBLEMÁTICA

La principal problemática es la de registrar datos manualmente, además de las posibles fallas o errores que este trabajo manual implica, administrar la información que se solicite se hace una tarea demasiada difícil y bastante tardía. A continuación, detallamos los registros en los cuales se generan dificultades:

* Registro de materiales (ingreso, disminución)
* Herramientas (ubicación, cantidad, estado) ✓ Préstamos
* Devolución.

1.3.2 FORMULACION DEL PROBLEMA

El encargado de bodega por ser la única persona encargada y al no contar con otro apoyo el solo debe tener información de las diferentes materiales dentro de la bodega como ser registro de los ingresos de los materiales, registro de préstamos de materiales para las diferentes situaciones como ser préstamos para proyecto, prestamos particulares también debe tener un control de la disminución de los materiales diariamente debe contener información actualizada de los materiales que se encuentran en la bodega.

## 1.4 OBJETIVOS

#### 1.4.1 OBJETIVO GENERAL

• Desarrollar un sistema de gestión de inventarios para la “Constructora Genex”.

#### 1.4.2 OBJETIVOS ESPECÍFICOS

* Recolectar información sobre la organización y el funcionamiento de la bodega de la empresa genex.
* Analizar la información recolectada y determinar los módulos y funciones de nuestro Sistema aplicando el proceso unificado con UML a través de diagramas de casos de usos y diagramas de clases.
* Diseñar una base de datos que soporte la información de los datos que a ser procesados en el SGBD postgres SQL.
* Implementar módulos que sean representativos del sistema como ser módulo de hora de partida y llegada, generar reportes del envió de las encomiendas y administración del pago de los socios, todo en el lenguaje de programación PHP y usando el editor Brackets.
* Realizar pruebas para detectar errores y comparar el nivel de los requerimientos que cumple con el usuario final.

## 1.5 METODOLOGÍA

Después de haber recaudado la información necesaria de la empresa, para su posterior análisis, nos permitió conocer el funcionamiento de estas para luego localizarnos en puntos críticos que es muy necesario resolver mediante el desarrollo de un sistema de información que tendrá un enfoque centrado en dos metodologías de desarrollo: EL Proceso Unificado De Desarrollo de Software (PUDS) y el Lenguaje Unificado De Modelado (UML).

* **Guiado por los casos de uso:** 
  + Los casos de uso son el instrumento para validar la arquitectura del software y extraer los casos de prueba.
  + Si tenemos definidos bien nuestros casos de uso el resultado será un producto correcto.
* **Centrado en la arquitectura:** 
  + Define una forma de organizar de las diferentes partes que tenga el software.
  + Lo que se busca es que el software sea flexible (a la hora de realizar los cambios).
  + Los modelos son proyecciones del análisis y el diseño constituye la arquitectura del producto a desarrollar.
* **Iterativo e incremental:** 
  + Es iterativo porque cada fase se repite.
  + Es Incremental porque cada ciclo genera una nueva versión que mejora las funcionalidades del anterior (hasta llegar al producto terminado o deseado).

Se implementará esta metodología dando más importancia en las fases que se indicaran a continuación:

* **Fase de inicio.**

**Captura de requisitos.**

* + - * Entrevista con el personal de salud para recaudar información en cuanto a las actividades que realiza para la carpeta familiar.
      * Identificar actores y casos de uso.
      * Priorizar casos de uso.
      * Estructuración de casos de uso.
      * Realizar los prototipos de interfaces de los casos de uso.

* **Fase de elaboración.**
  + - * Análisis.
      * Análisis de paquete.
      * Análisis de casos de uso.
      * Análisis de arquitectura.

**Resultado**: Especificación de requisitos del software.

* + - * Diseño de arquitectura.
      * Diseño de casos de uso.
      * Diseño de datos.

Resultado: Descripción del diseño del software.

* **Fase de construcción.**

**Implementación**

✓ Implementar los modelos de diseño en un lenguaje de programación.

**Resultado**: modelos de implementación y códigos fuentes.

1.5.1 METODOLOGÍA – P.U.D.S.

El proceso unificado de desarrollo describe actividades de trabajo como escribir casos de usos en disciplinas. Informalmente, una disciplina es un conjunto de actividades y artefactos relacionados en un área determinada como las actividades en el análisis de requisitos. En el proceso unificado un artefacto es un término general para cualquier producto realizado a través de los flujos de trabajo, podemos mencionar entre estos: código, grafico web, esquema de base de datos, documento de texto, diagramas, modelos, etc.

Las disciplinas del proceso Unificado serian:

* **Modelo de Negocio:** Cuando se desarrolla una única aplicación, esto incluye el modelado de los objetos del dominio. Cuando se está haciendo análisis del negocio a gran escala o reingeniería de procesos del negocio, eso incluye el modelado dinámico de los procesos del negocio de toda la empresa.
* **Requisitos**: Análisis de los requisitos para una aplicación, como escribir los casos de uso e identificación de requisitos no funcionales.
* **Diseño**: Todos los aspectos de diseño, incluyendo la arquitectura global, objetos, base de datos, red y cosas parecidas.
* **Entorno**: Se establecen las herramientas y la adaptación del proceso al proyecto en sí.
* **Modelos**: Los modelos son simplificaciones de la realidad, construimos modelos para comprender mejor los sistemas debido a la complejidad de los mismos, de otra forma no sería posible abordarlos en su totalidad como son en la realidad. A través del modelado se logran los siguientes objetivos:
  + Ayudar a visualizar como es el sistema
  + Permitir especificar la estructura del sistema
  + Especificar el comportamiento del sistema
  + Proporcionar plantillas que guían la construcción del sistema ✓ Un modelo tiene dos aspectos esenciales:
    - * Semántica: se trata de la información o significado del modelo.
      * Presentación visual: es la notación que muestra el modelo de una forma comprensible. Organiza la presentación del modelo.

## 1.6 ALCANCE

#### 1.6.1 REQUISITOS FUNCIONALES

* **RF1**: **Gestionar Usuario**

Este requisito permitirá que se pueda registrar, modificar y eliminar usuarios en el software

* **RF2**: **Gestionar proyecto**

Dentro de la empresa se realizan proyecto construcción de prefabricados, etc. Necesitamos registrar: Nombre de proyecto, encargado del proyecto, fecha de inicio.

* **RF3**: **Gestionar material**

De cada material necesitaremos la siguiente información: Código, Tipo de material, cantidad, Fecha.

* **RF4**: **Gestionar inventario**

El inventario representa la existencia de bienes almacenados dentro de la empresa destinada a realizar una operación ya sea para préstamos dentro de la empresa o para la repartición de los materiales para los proyecto, en los cuales se agregarán datos como: código, cantidad de herramientas, cantidad de materiales, cantidad de dotación, Numero de proyecto, etc.

* **RF5**: **Gestionar préstamo**

Cada proyecto se le asigna cierta cantidad de herramientas y la bodega se encarga del respectivo préstamo de herramientas para los diferentes proyecto dentro de la empresa y debe registrar la siguiente información: Nombre del proyecto, encargado, herramienta, cantidad de herramientas.

* **RF6: Registrar producción**

El encargado de la bodega tiene el control de producción de los proyecto de prefabricado dentro de la empresa, los cuales anota: Nombre del proyecto, cantidad de producción, tipo de prefabricado.

* **RF7**: **Gestionar dotación**

Este requisito permitirá que de cada material de dotación necesitaremos la siguiente información:

Código, Nombre de la dotación, cantidad.

* **RF8: Reportes y Estadísticas**

Este requisito permitirá que se puedan obtener gráficos y tablas estadísticas las cuales mostraran información referente a los eventos realizados respectos a los cursos, talleres y conferencias.

#### 1.6.2 REQUISITOS NO FUNCIONALES

1.6.2.1 SEGURIDAD

Funcionalidad encargada de mantener coherente la base de datos de registro del perfil profesional. En caso de eventualidades no deseadas, realizando copias de seguridad y restauración.

Los módulos que participan son:

* Validación de entrada de datos
* Validación de accesos
* Restauración de base de datos
* Backus de la base de datos

# CAPITULO II: SISTEMA DE CORREO

## 2.1 MARCO TEÓRICO

#### 2.1.1. CORREO ELECTRONICO

El Correo Electrónico es un sistema que permite el envío/recepción de mensajes a través de internet. El funcionamiento es similar al correo postal, con la comodidad que la entrega se efectúa en el acto. La comunicación es sumamente rápida; podemos enviar correos electrónicos a personas que están en la otra parte del mundo en cuestión de segundos.

Principalmente se usa este nombre para denominar al sistema que provee este servicio en Internet, mediante el protocolo SMTP, aunque por extensión también puede verse aplicado a sistemas análogos que usen otras tecnologías. Por medio de mensajes de correo electrónico se puede enviar, no solamente texto, sino todo tipo de documentos digitales. Su eficiencia, conveniencia y bajo coste (casi nulo) están logrando que el correo electrónico desplace al correo ordinario para muchos usos habituales.

2.1.1.1 FUNCIONAMIENTO DEL CORREO ELECTRONICO

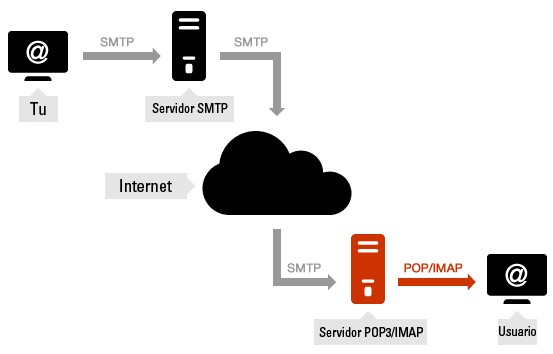
El correo electrónico gira alrededor del uso de las casillas de correo electrónico. Cuando se envía un correo electrónico, el mensaje se enruta de servidor a servidor hasta llegar al servidor de correo electrónico del receptor. Más precisamente, el mensaje se envía al servidor del correo electrónico (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 Protocol [Protocolo de Oficina de Correo]), 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 [Protocolo de Acceso a Mensajes de Internet]), 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.



***Figura 4:*** *Esquema de funcionamiento de correo electrónico*

***Fuente****: Curiosoando.com (3 julio, 2014).*

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.

#### 2.1.2 PROTOCOLO SMTP

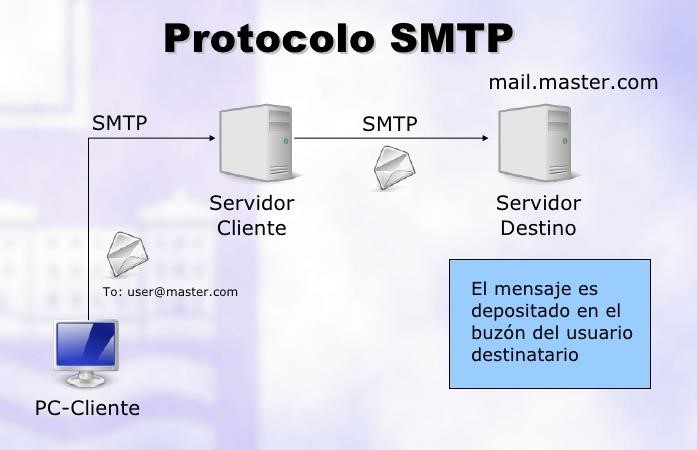
SMTP es Protocolo Simple de Transmisión de Correo ("Simple Mail Transfer Protocol", RFC 821). Este protocolo es el estándar de Internet para el intercambio de correo electrónico. SMTP necesita que el sistema de transmisión ponga a su disposición un canal de comunicación fiable y con entrega ordenada de paquetes, con lo cual, el uso del protocolo TCP (puerto 25) en la capa de transporte, es lo adecuado. Para que dos sistemas intercambien correo mediante el protocolo SMTP, no es necesario que exista una conexión interactiva, ya que este protocolo usa métodos de almacenamiento y reenvío de mensajes.

2.1.2.1 MODO DE COMUNICACIÓN SMTP

Cuando un servidor de SMTP, requiere transmitir un mensaje a otro servidor SMTP, el emisor (servidor que inicia la sesión SMTP) establece una conexión con el receptor (servidor que recibe petición de establecer sesión SMTP). Esta conexión es unidireccional, es decir, el emisor puede enviar correo al receptor, pero durante esa conexión, el receptor no puede enviar correo al emisor. Si el receptor tiene que enviar correo al emisor, tiene que esperar a que finalice la conexión establecida y establecer otra en sentido contrario, cambiando los papeles de emisor y receptor. Una vez establecida la conexión, el emisor envía comandos y mensajes.

Por lo tanto, el diseño de SMTP se basa en el siguiente modelo de comunicación:

* Como respuesta a una solicitud de un usuario de enviar un correo electrónico, el emisor SMTP establece una conexión con el receptor SMTP.
* El receptor SMTP debe ser el destinatario último del correo o un intermediario. Para ello el emisor genera los comandos SMTP en formato ASCII y los envía al receptor y el receptor genera las respuestas a los comandos enviados por el emisor.
* Una vez establecido el canal de transmisión, el emisor envía el comando MAIL para indicando que él es el emisor del correo. Si el receptor puede aceptar correo responde con el comando OK.
* El emisor envía el comando RCPT identificando el destinatario del correo. Si el receptor puede aceptar correo para ese destino responde con una respuesta OK; si no, responde rechazando el correo para ese destino.
* Una vez negociado el destino, el emisor comienza a enviar datos (cabeceras y cuerpo), terminando con una secuencia especial. Si el receptor ha procesado correctamente los
* datos, responde con el comando OK.



***Figura 5:*** *Esquema de funcionamiento del Protocolo SMTP*

***Fuente:*** *SlideShare.net (11 de febrero del 2010)*

#### 2.1.3 MIME

2.1.3.1 MIME HEADER

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.

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.

MIME está especificado en seis RequestforComments 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).

El protocolo SMPT soporta solo caracteres ASCII de 7 bit (véase también 8BITMIME). Esto limita los mensajes de correo electrónico, ya que incluyen solo caracteres suficientes para escribir en un número reducido de lenguajes, principalmente Inglés. Otros lenguajes basados en el Alfabeto latino es adicionalmente un componente fundamental en protocolos de comunicación como HTTP, el que requiere que los datos sean transmitidos como un e-mail aunque los datos pueden no ser un e-mail propiamente dicho. Los clientes de correo y los servidores de correo convierten automáticamente desde y a formato MIME cuando envían o reciben (SMTP/MIME) emails.

#### 2.1.4 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:

|  |  |
| --- | --- |
| **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** | Información acerca de los mensajes del servidor |
| **RETR** | Número del mensaje que se va a recoger |
| **DELE** | Número del mensaje que se va a eliminar |
| **LIST [msg]** | Número del mensaje que se va a mostrar |
| **NOOP** | Permite mantener la conexión abierta en caso de inactividad |
| **TOP <messageID><n>** | Comando que muestra n líneas del mensaje, cuyo número se da en el argumento. En el caso de una respuesta positiva del servidor, |
|  | éste enviará de vuelta los encabezados del mensaje, después una línea en blanco y finalmente las primerasn líneas del mensaje. |
| **UIDL [msg]** | Solicitud al servidor para que envíe una línea que contenga información sobre el mensaje que eventualmente se dará en el argumento. Esta línea contiene una cadena de caracteres denominada unique identifier listing (lista de identificadores únicos) que permite identificar de manera única el mensaje en el servidor, independientemente de la sesión. El argumento opcional un número relacionado |

***Tabla 1****: Comandos pop3*

***Fuente:*** *Creación propia (2021)*

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.

#### 2.1.5 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.

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 agente de transporte de correo 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 Mail UserAgent (MUA) como pueden ser elm, Eudora o pine; o provenientes de un Mail TransportAgent (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.

Todas estas características y muchas otras que posee el Sendmail deben ser configuradas y variarán de unos sistemas a otros. Para configurarlas hacemos uso del fichero de configuración de Sendmail. La revisión y modificación de este fichero es bastante complicada y necesita de una serie de conocimientos previos.

# CAPITULO III: DESARROLLO

## 3.1 FLUJO DE TRABAJO: CAPTURA DE REQUISITOS.

#### 3.1.1 IDENTIFICAR ACTORES Y CASOS DE USOS

* Identificar Actores
* **Administrador:** Persona que se encarga de manejar el sistema.
* **Trabajador**: Persona con un salario y con obligaciones dentro de la empresa.
* Identificar Casos de Uso
  + CU1 Gestionar usuario
  + CU2 Gestionar proyecto
  + CU3 Gestionar material
  + CU4 Gestionar inventario
  + CU5 Gestionar préstamo
  + CU6 Registrar producción
  + CU7 Administrar dotación
  + CU8 Reportes y estadísticas
* Priorizar Casos de Usos

|  |  |  |
| --- | --- | --- |
| **Casos de usos** | **Prioridad** | **Riesgo** |
| **CU1** Gestionar Usuario | Alto | Normal |
| **CU2** Gestionar proyecto | Alto | Normal |
| **CU3** Gestionar material | Medio | Normal |
| **CU4** Gestionar inventario | Medio | Normal |
| **CU5** Gestionar prestamo | Alto | Normal |
| **CU6** Registrar producción | Alto | Normal |
| **CU7** Administrar dotación | Alto | Crítico |
| **CU8** Reportes y Estadisticas | Alto | Normal |

***Tabla 2:*** *Priorización de Casos de uso* ***Fuente:*** *Creación propia (2021).*

#### 3.1.2 DESCRIPCIÓN DE CASOS DE USOS

CU1. Gestionar Usuario

**uc Gestionar Usuario**

**Gestionar Usuario**

**Usuario**

|  |  |
| --- | --- |
| **Caso de uso** | **Gestión de usuario** |
| **Propósito** | Esta funcionalidad permitirá que los usuarios que vayan a hacer uso del software puedan registrarse. |
| **Actores** | Administrador, trabajador |
| **Actor iniciador** | Administrado, trabajador |
| **Pre condición** | Ninguna |
| **Flujo principal** | REGISTRAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo se podrá registrar un usuario.   MODIFICAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectico el administrador podrá hacer modificaciones sobre la información del usuario.   ELIMINAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo el administrador podrá eliminar un usuario del sistema. |

***Tabla 3:*** *Descripción CU1: Gestionar Usuario*

***Fuente****: Creación Propia (2021)*

CU2. Gestionar Proyecto

**uc Gestionar Proyecto**

**Gestionar Proyecto**

**Administrador**

|  |  |
| --- | --- |
| **Nombre caso de uso** | **Gestionar Proyecto** |
| **Propósito** | Esta funcionalidad permitirá que el administrador pueda registrar los proyecto y asignar un encargado por proyecto |
| **Actores** | Administrador |
| **Actor iniciador** | Administrador |
| **Pre condición** | Ninguna |
| **Flujo principal** | REGISTRAR PROYECTO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo se podrá registrar un usuario y especificar si el nuevo proyecto.     ASIGNAR ENCARGADO PROYECTO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo para asignar un usuario encargado. |

***Tabla 4:*** *Descripción CU2: Gestionar Proyecto* ***Fuente:*** *Creación Propia (2021).*

CU3. Gestionar Material

**uc Gestionar Material**

**Gestionar Material**

**Administrador**

|  |  |
| --- | --- |
| **Nombre caso de uso** | **Gestionar Material** |
| **Propósito** | Esta funcionalidad permitirá que los usuarios que vayan a hacer uso del software puedan saber el sotck de los materiales. |
| **Actores** | Administrador, trabajador |
| **Actor iniciador** | Administrador, trabajador |
| **Pre condición** | Ninguna |
| **Flujo principal** | REGISTRAR MATERIAL   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo se podrá registrar un nuevo material.   VER STOCK   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo el administrador podrá ver el estado de stock de los materiales. |

***Tabla 5:*** *Descripción CU3: Gestionar Material* ***Fuente:*** *Creación Propia (2021).*

CU4. Gestionar Inventario

**uc Gestionar Inventario**

**Gestionar**

**Inventario**

**Administrador**

|  |  |
| --- | --- |
| **Nombre caso de uso** | **Gestionar Inventario** |
| **Propósito** | Esta funcionalidad permitirá que los usuarios que vayan a hacer uso del software podrán ver el stock de productos, registrar el ingreso y egreso de los materiales del almacén. |
| **Actores** | Administrador |
| **Actor iniciador** | Administrador |
| **Pre condición** | Ninguna |
| **Flujo principal** | REGISTRAR INGRESO Y EGRESO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo se podrá registrar el egreso o ingreso de un respectivo material.   VER STOCK   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo el administrador ver el estado actual de inventario. |

***Tabla 6:*** *Descripción CU4: Gestionar Usuario* ***Fuente:*** *Creación Propia (2021).*

CU5. Gestionar Prestamo

**uc Gestionar Prestamo**

**Gestionar Prestamo**

**Administrador**

|  |  |
| --- | --- |
| **Nombre caso de uso** | **Gestionar Préstamo** |
| **Propósito** | Esta funcionalidad permitirá que los usuarios que vayan a hacer uso del software puedan registrar, modificar y eliminar préstamos. |
| **Actores** | Administrador, trabajador |
| **Actor iniciador** | Administrador, trabajador |
| **Pre condición** | El material debe estar registrado |
| **Flujo principal** | REGISTRAR PRESTAMO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo se podrá registrar un nuevo préstamo.   MODIFICAR PRESTAMO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectico el administrador podrá hacer modificaciones sobre la información de algún préstamo.   ELIMINAR PRESTAMO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo el administrador podrá eliminar un préstamo realizado. |

***Tabla 7:*** *Descripción CU5: Gestionar Material*

***Fuente:*** *Creación Propia (2021)*

CU6. Registrar Producción

**uc Registrar Producción**

**Registrtar Producción**

**Administrador**

|  |  |
| --- | --- |
| **Nombre caso de uso** | **Registrar Producción** |
| **Propósito** | Esta funcionalidad permitirá que los usuarios que vayan a hacer uso del software puedan registrar. |
| **Actores** | Administrador, trabajador. |
| **Actor iniciador** | Administrador, trabajador |
| **Pre condición** | Debe estar definido en el caso de uso gestionar proyecto. |
| **Flujo principal** | REGISTRAR PRODUCCIÓN   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo se podrá registrar la cantidad de prefabricados de cada proyecto. Por ejemplo cordones, losetas. |

***Tabla 8:*** *Descripción CU6: Registrar Producción*

***Fuente:*** *Creación Propia (2021)*

CU7. Administrar Dotación

**uc Administrar Dotación**

**Administrar**

**Dotación**

**Administrador**

|  |  |
| --- | --- |
| **Nombre caso de uso** | **Administrar Dotación** |
| **Propósito** | Esta funcionalidad permitirá que los usuarios que vayan a hacer uso del software puedan registrar los tipos de dotación y un stock de la dotación. |
| **Actores** | Administrador |
| **Actor iniciador** | Administrador |
| **Pre condición** | Proyecto previamente definido en el caso de uso registrar proyecto. |
| **Flujo principal** | REGISTRAR TIPOS DE DOTACIÓN   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo, se podrá registrar un tipo de dotación para un proyecto existente.   VER STOCK DOTACIÓN   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectico el administrador podrá ver el stock de las dotaciones dadas a cada proyecto. |

***Tabla 9:*** *Descripción CU7: Registrar Dotación* ***Fuente:*** *Creación Propia (2021).*

CU8. Reportes y Estadísticas

**uc Reportes y Estadisticas**

**Reportes y**

**Estadisticas**

**Usuario**

|  |  |
| --- | --- |
| **Nombre caso de uso** | **Reportes y Estadísticas** |
| **Propósito** | Esta funcionalidad permitirá que el software pueda devolver información sobre su funcionamiento al usuario que la solicite utilizando tablas y gráficos estadísticos. |
| **Actores** | Administrador, trabajador |
| **Actor iniciador** | Administrador, trabajador |
| **Pre condición** | Ninguna |
| **Flujo principal** | VER ESTADISTICAS   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder ver gráficos estadísticos.   VER REPORTES   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder ver reportes. |

***Tabla 10:*** *Descripción CU8: Reportes y Estadísticas* ***Fuente****: Creación Propia (2021)*

#### 3.1.3. Modelo general de casos de uso

**uc Diagrama de Casos de uso**

**Gestionar Usuario**

**Gestionar Proyecto**

**Gestionar Material**

**Gestionar Inventario**

**Gestionar Prestamo**

**Registrar**

**Producción**

**Administrar**

**Dotación**

**Reportes y**

**Estadisticas**

**Usuario**

**Administrador**

***Figura 6:*** *Diagrama general de casos de uso*

***Fuente:*** *Creación Propia (2021)*

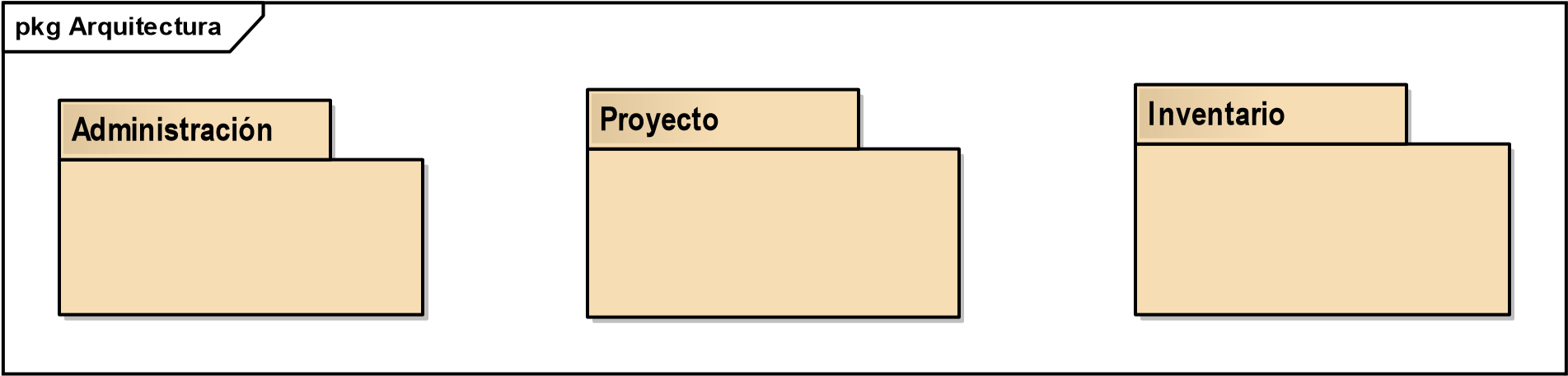
# CAPITULO IV: ANÁLISIS

## 4.1 ANÁLISIS DE LA ARQUITECTURA

#### 4.1.1 IDENTIFICACIÓN DE PAQUETES

Los paquetes que se definieron son:

* Administración
* Proyecto
* Inventario



***Figura 7:*** *Identificación de Paquetes*

***Fuente:*** *Creación Propia (2021)*

#### 4.1.2 RELACIÓN DE PAQUETES Y CASOS DE USO

**pkg Administracion**

**Administración**

**CU1. Gestionar**

**Usuario**

**CU8. Gestionar**

**Reportes y**

**Estadisticas**

«trace»

«trace»

***Figura 8:*** *Relación paquete Administración con CU1, CU8*  ***Fuente****: Creación Propia (2021).*

**pkg Inventario**

**Inventario**

**CU3. Gestionar**

**Material**

**CU4. Gestionar**

**Inventario**

**CU5. Gestionar**

**Prestamo**

«trace»

«trace»

«trace»

***Figura 9:*** *Relación paquete Administración con CU3, CU4. CU5*

***Fuente****: Creación Propia (2021)*

**pkg Proyectos**

**Proyecto**

**CU2. Gestionar**

**Proyecto**

**CU7. Administrar**

**Dotación**

**CU6. Registrar**

**Producción**

«trace»

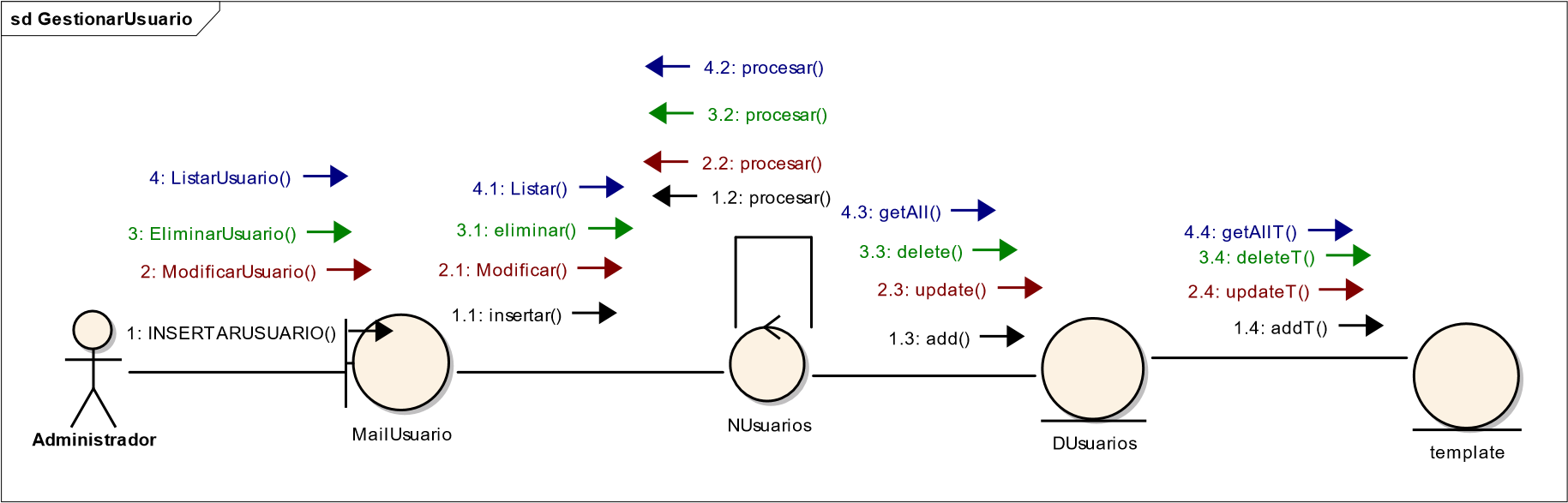
«trace»

«trace»

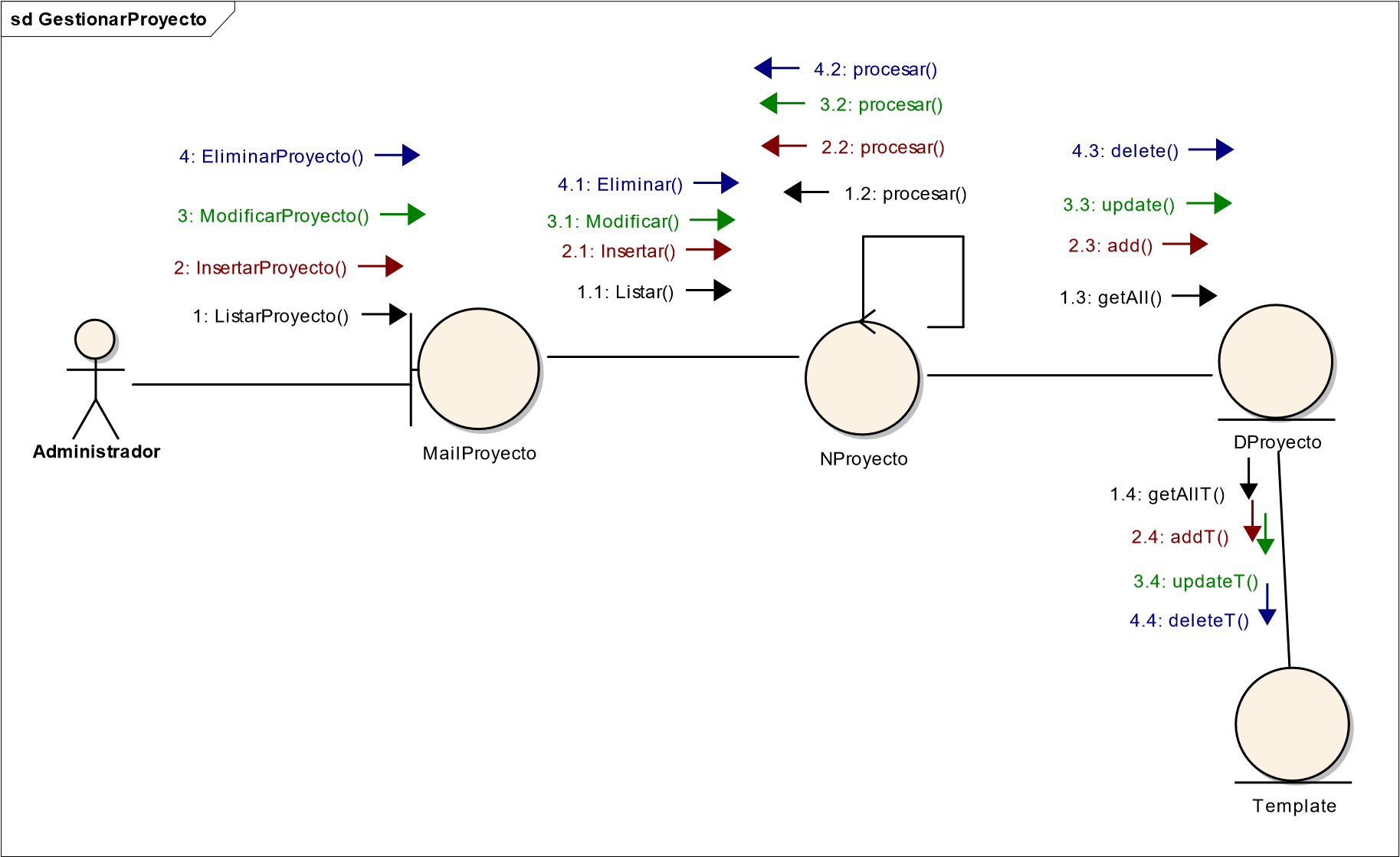
***Figura 10:*** *Relación paquete Administración con CU7, CU6, CU2* ***Fuente:*** *Creación Propia (2021).*

## 4.2 ANÁLISIS DE CASOS DE USO

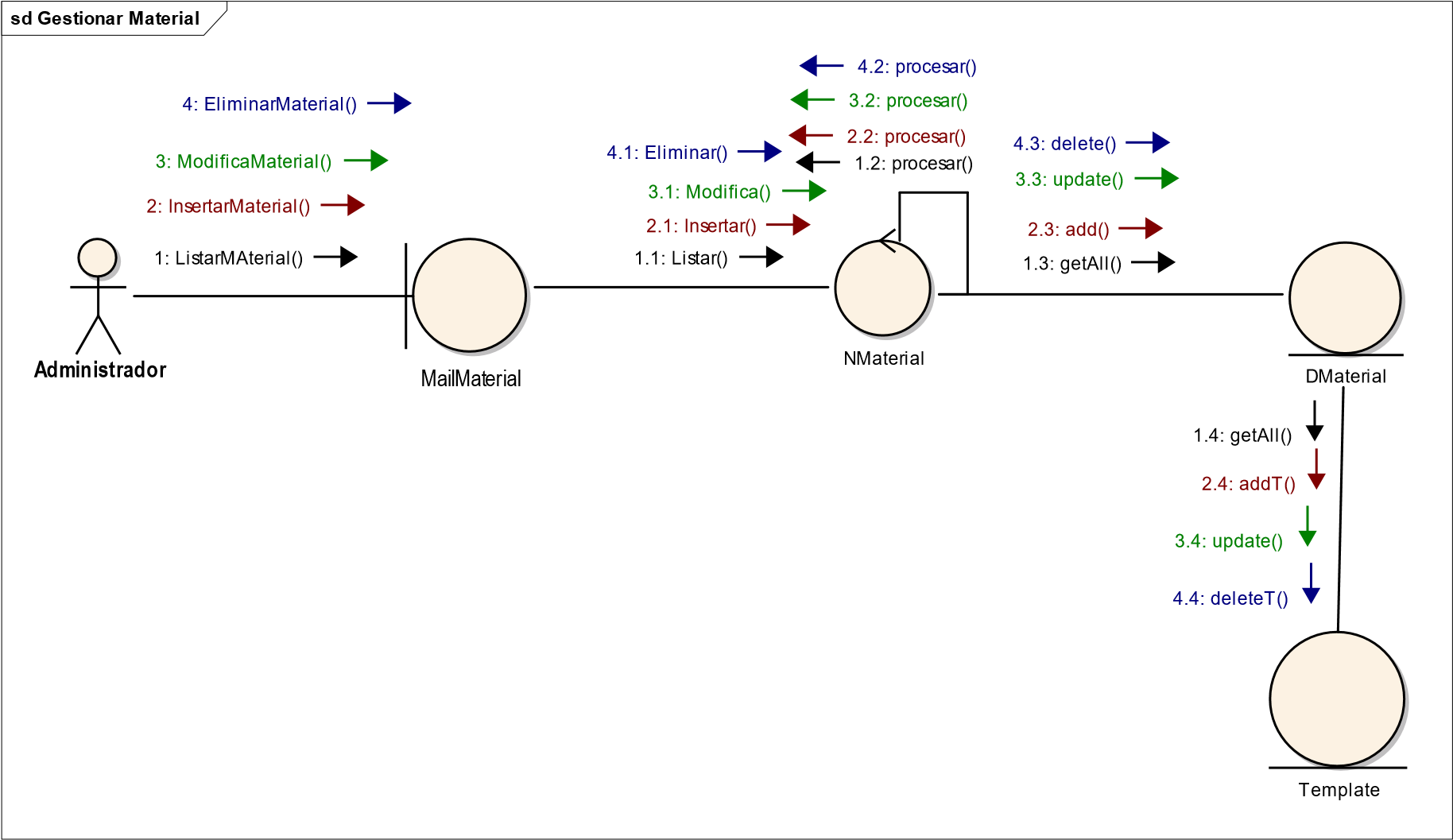
CU1. Gestionar Usuario



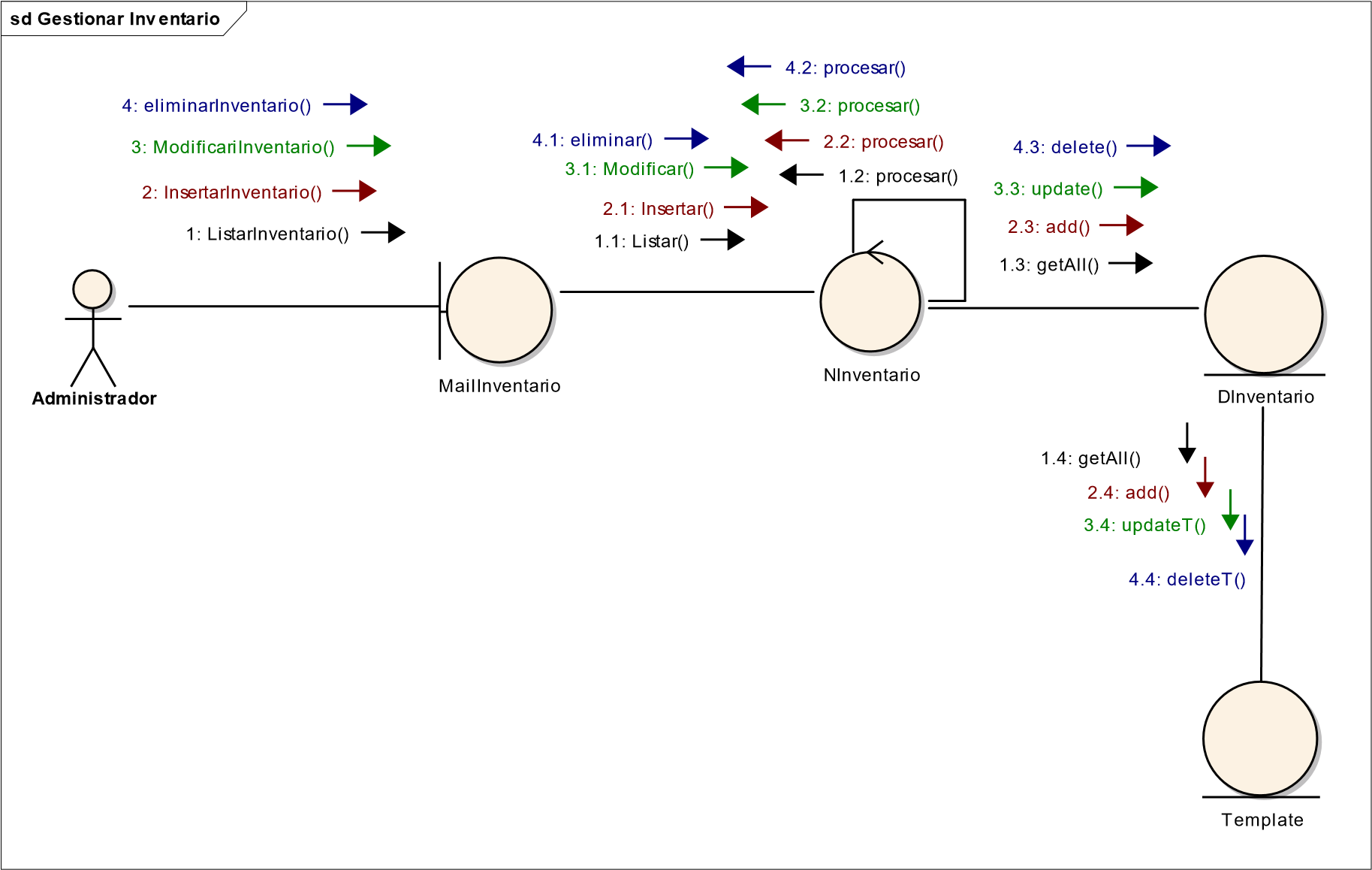
CU2. Gestionar proyecto



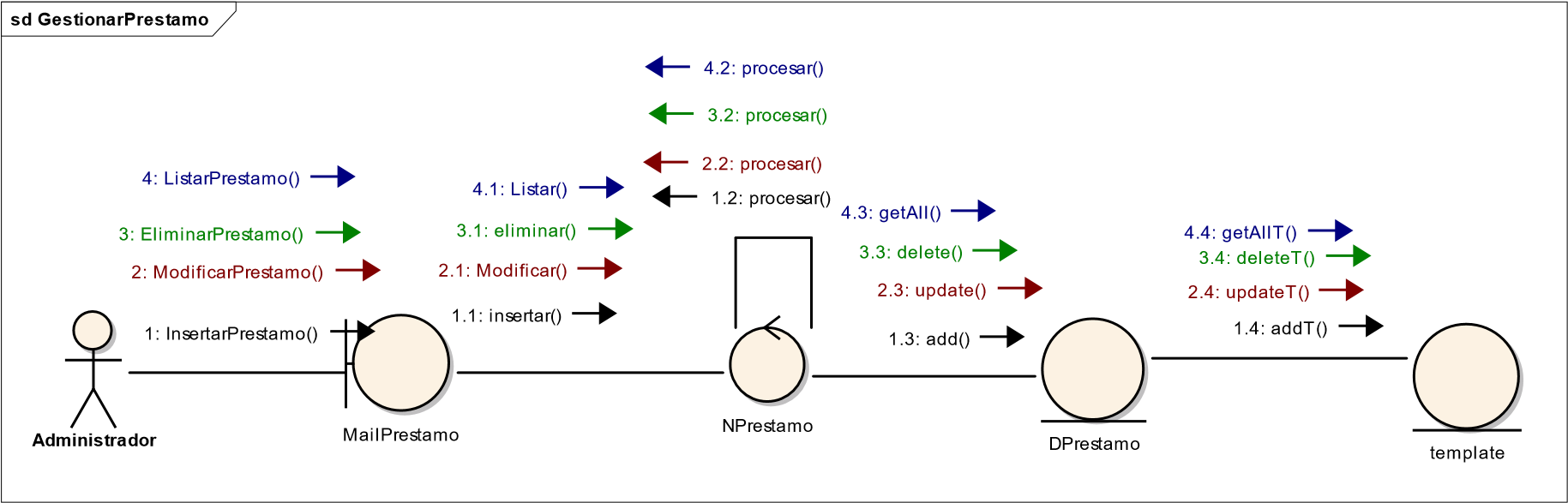
CU3. Gestionar material



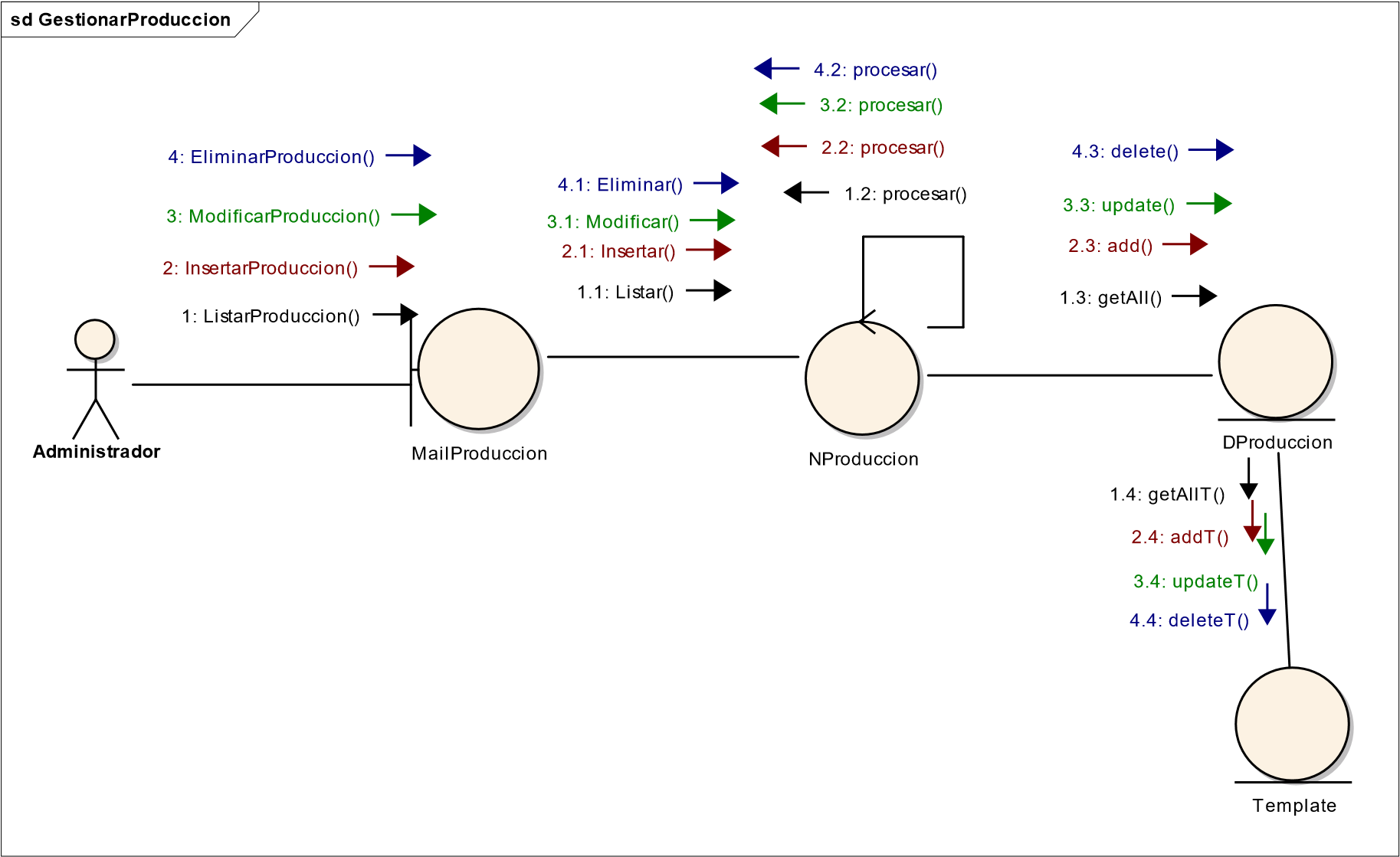
CU4. Gestionar inventario



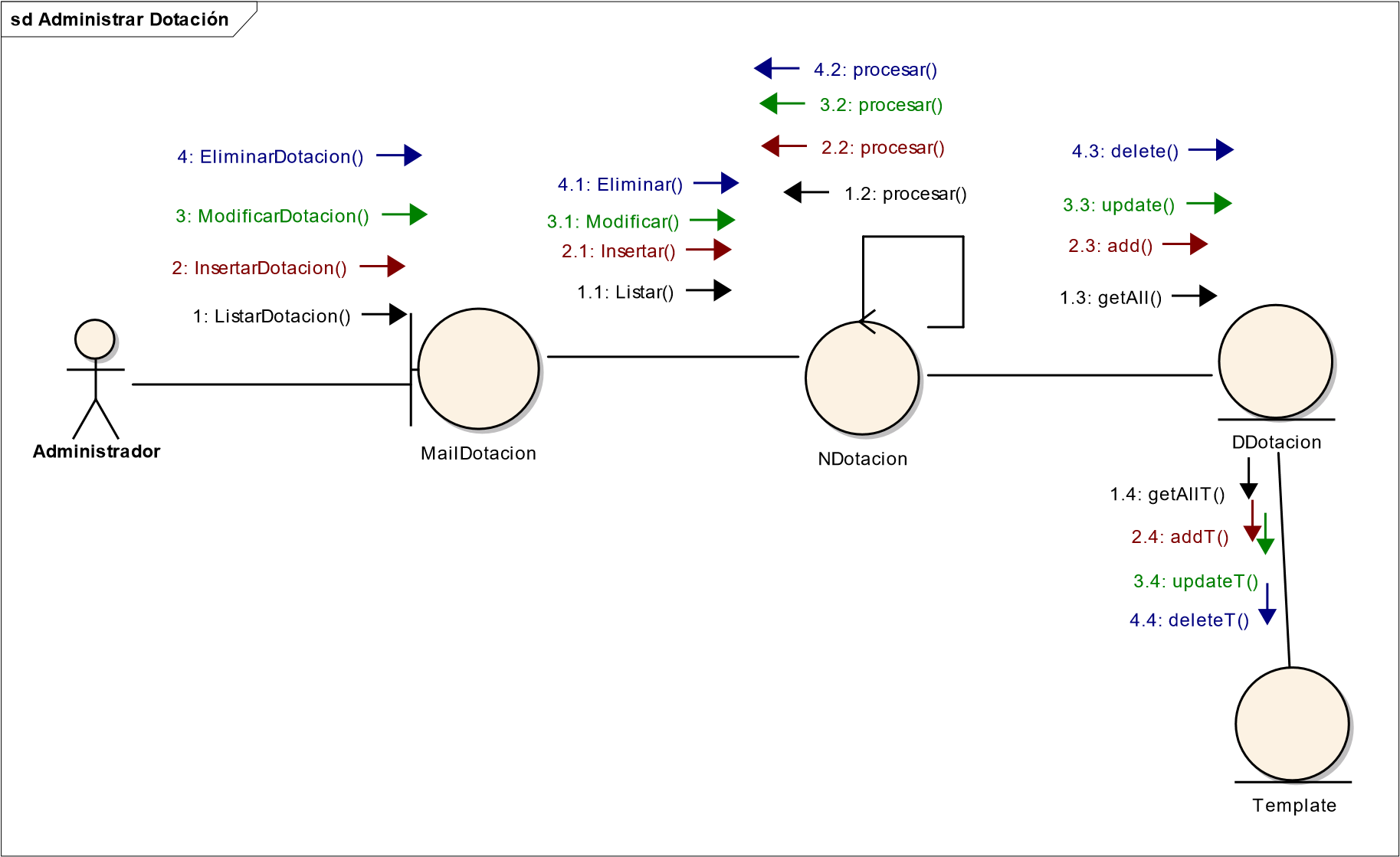
CU5. Gestionar préstamo



CU6. Registrar producción



CU7. Administrar dotación



CU8. Reportes y estadísticas

**sd Reporte Estadistica**

**Administrador**

DProducion

DMaterial

DHerramienta

NReporte

MailReporte

conexión

()

1:

Reporte

()

1.1:

ListarReporte

procesar

()

1.2:

1.3:

obtenerdatos

()

1.4:

()

obtenerDatos

1.5:

obtenerDatos

()

1.6:

buscar

()

1.7:

Buscar

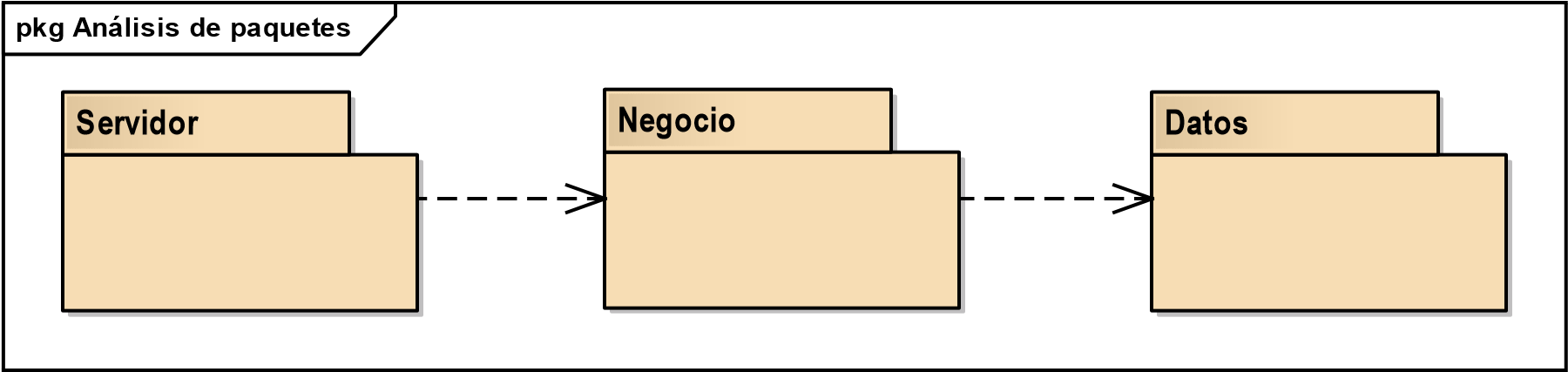
()

1.8:

buscar

()

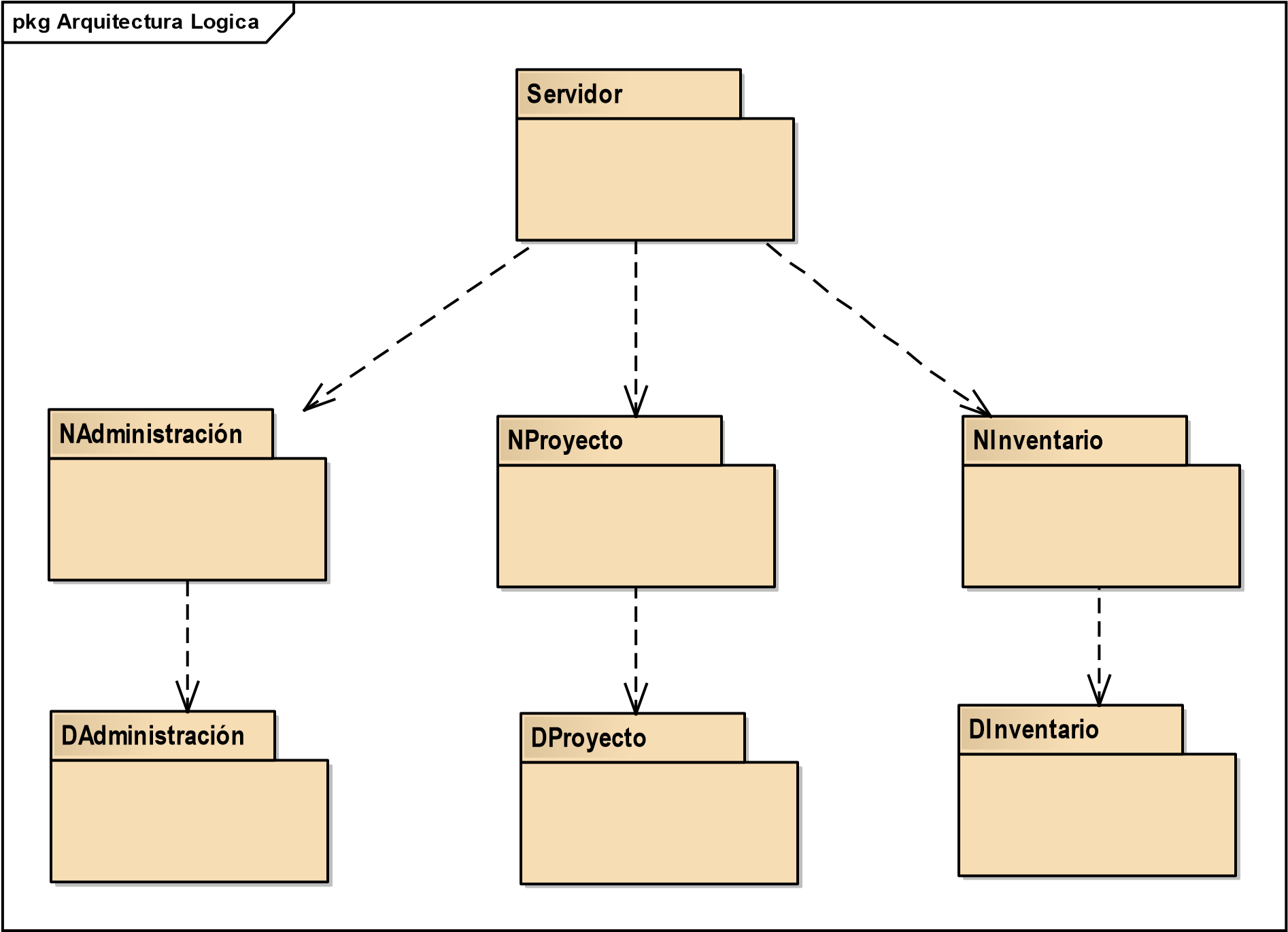
## 4.3 ANÁLISIS DE PAQUETES



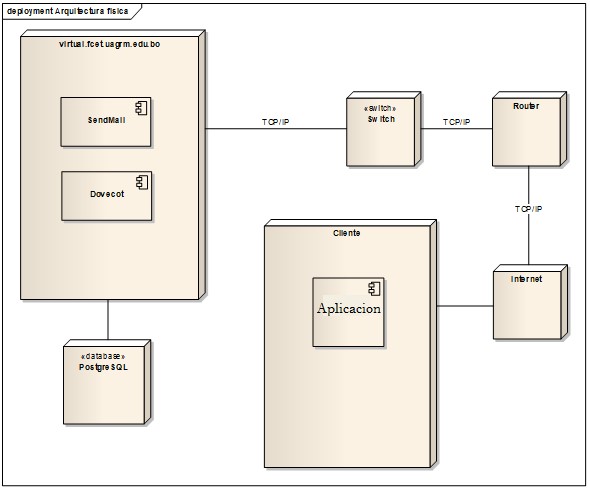
# CAPITULO V: DISEÑO

## 5.1 DISEÑO DE LA ARQUITECTURA

#### 5.1.1 ARQUITECTURA LÓGICA

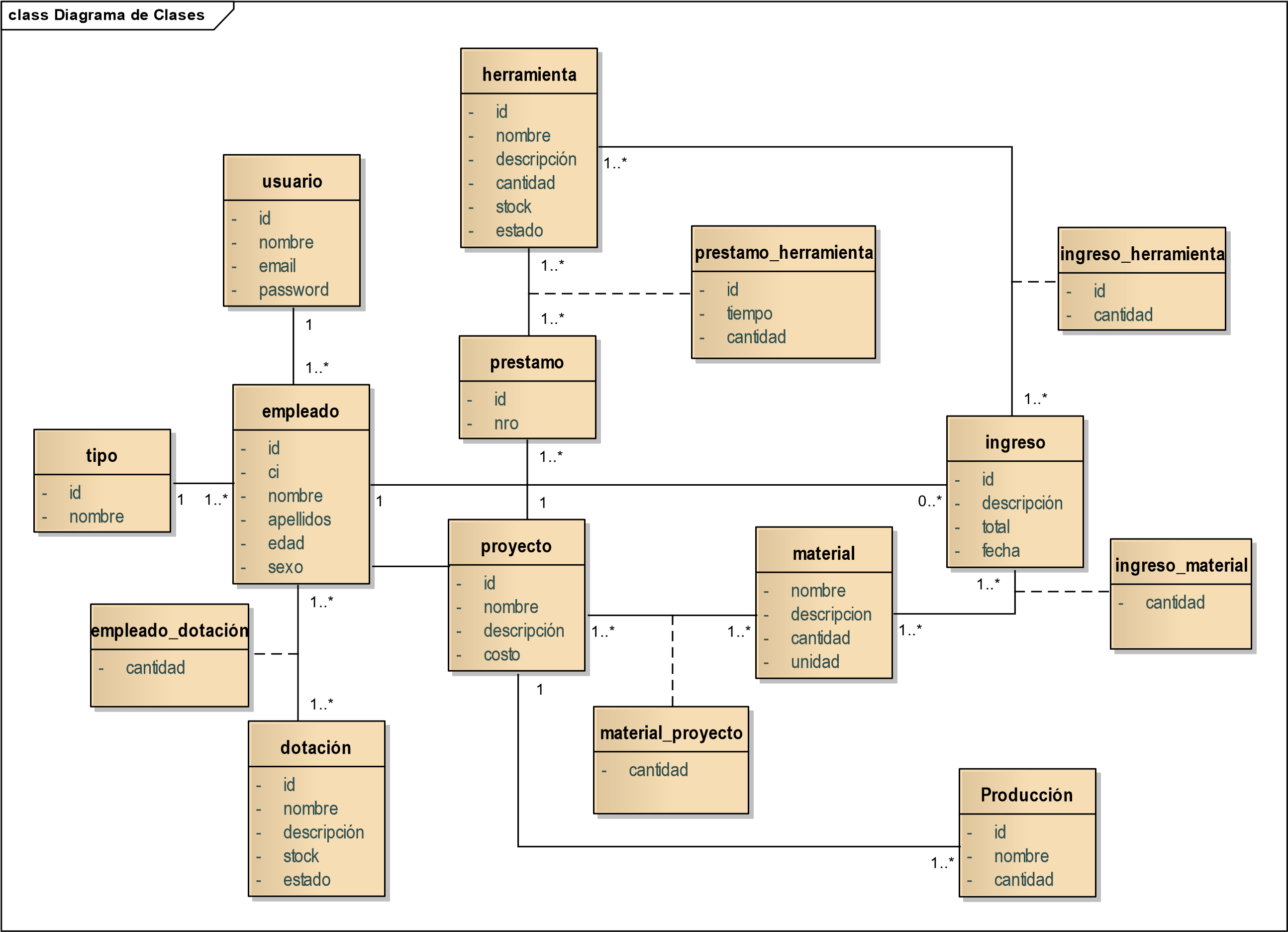


5.1.2 ARQUITECTURA FÍSICA



## 5.2 DISEÑO DE LA BASE DE DATOS

#### 5.2.1 DISEÑO CONCEPTUAL



39

#### 5.2.2 DISEÑO LÓGICO

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Herramienta** | | | | | | | | | | | | | | | | |
| Id | descripción | | | | Estado | | Cantidad | | | nombre | | | | | stock | |
| **Usuario** | | | | | | | | | | | | | | | | |
| Id | Email | | | | | nombre | | | | | | | password | | | |
| **Empleado** | | | | | | | | | | | | | | | | |
| Id | apellidos | | | | nombre | | edad | | | | sexo | | | | | ci |
| **Dotación** | | | | | | | | | | | | | | | | |
| Id | descripción | | | | estado | | | Nombre | | | | | | stock | | |
| **Material** | | | | | | | | | | | | | | | | |
| Nombre | | descripción | | | | unidad | | | | Cantidad | | | | | | |
| **Préstamo** | | | | | | | | | | | | | | | | |
| Id | | | | | | nro\_Prestamo | | | | | | | | | | |
| **Proyecto** | | | | | | | | | | | | | | | | |
| Id | | | nombre | | | costo | | | | | | descripción | | | | |
| **Tipo** | | | | | | | | | | | | | | | | |
| Id | | | | | | nombre | | | | | | | | | | |
| **Empleado\_Dotación** | | | | | | | | | | | | | | | | |
| Cantidad | | | | | | | | | | | | | | | | |
| **Material\_Proyecto** | | | | | | | | | | | | | | | | |
| Cantidad | | | | | | | | | | | | | | | | |
| **Ingreso\_Material** | | | | | | | | | | | | | | | | |
| Cantidad | | | | | | | | | | | | | | | | |
| **Ingreso\_Herramienta** | | | | | | | | | | | | | | | | |
| Id | | | | | | cantidad | | | | | | | | | | |
| **Prestamo\_Herramienta** | | | | | | | | | | | | | | | | |
| Id | | | | cantidad | | | | | tiempo | | | | | | | |
| **Material\_Proyecto** | | | | | | | | | | | | | | | | |
| Cantidad | | | | | | | | | | | | | | | | |

#### 5.2.3 DISEÑO FÍSICO

5.2.3.1. TABLAS DE VOLUMEN

**Herramienta**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **id** | Numero | No | Primaria |
| **Nombre** | Texto | 512 | No |
| **descripción** | Numero | 512 | No |
| **Estado** | Texto | 25 | No |
| **cantidad** | Texto | 512 | No |
| **stock** | Texto | 512 | No |

**Usuario**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **id** | Numero | No | Primaria |
| **nombre** | Texto | 512 | No |
| **email** | Texto | 512 | No |
| **Password** | Numero | 15 | No |

**Empleado**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **id** | Numero | No | Primaria |
| **Apellidos** | Texto | 512 | No |
| **Nombre** | Texto | 512 | No |
| **Edad** | Texto | 15 | No |
| **Sexo** | varchar | 1 | No |
| **Ci** | Texto | 15 | No |
| **Id\_Tipo** | Numero | No | Foránea |
| **Id\_Usuario** | Numero | No | Foránea |

**Dotación**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **id** | Numero | No | Primaria |
| **Descripción** | Texto | 512 | No |
| **Nombre** | Texto | 512 | No |
| **Estado** | Texto | 15 | No |
| **Stock** | Numero | 25 | No |

**Material**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **cantidad** | Numero | No | Primaria |
| **Nombre** | Texto | 512 | No |
| **Descripción** | Texto | 512 | No |
| **Unidad** | Numero | No | No |

**Préstamo**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **Id** | Fecha | No | Primaria |
| **Nro\_Prestamo** | Numero | 512 | No |

**Proyecto**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **Id** | Numero | No | Primaria |
| **Nombre** | Texto | 512 | No |
| **Costo** | Numero | No | No |
| **Descripción** | Texto | 512 | No |

**Tipo**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **Id** | Numero | No | Primaria |
| **Nombre** | Texto | 512 | No |

**Empleado\_Dotación**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **Id** | Numero | No | Primaria |
| **Dia** | Texto | 15 | No |

**Material\_Proyecto**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |  |
| **Id\_Proyecto** | Numero | No | Primaria Foránea | y |
| **Id\_Material** | Numero | No | Primaria Foránea | y |
| **cantidad** | Numero | 25 | No |  |

**Ingreso\_Material**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |  |
| **Id\_Material** | Numero | No | Primaria Foránea | y |
| **Id\_Ingreso** | Numero | No | Primaria Foránea | y |
| **Cantidad** | Numero | 15 | No |  |

**Ingreso\_Herramienta**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |  |
| **Id\_Ingreso** | Numero | No | Primaria Foránea | y |
| **Id\_Herramienta** | Numero | No | Primaria Foránea | y |
| **Cantidad** | Numero | 25 |  |  |

**Prestamo\_Herramienta**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |  |
| **Id\_Prestamo** | Numero | No | Primaria Foránea | y |
| **Id\_Herramienta** | Numero | No | Primaria Foránea | y |
| **Tiempo** | Numero | 512 | No |  |
| **cantidad** | Numero | 25 | No |  |

**Material\_Proyecto**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |  |
| **Id\_Material** | Numero | No | Primaria Foránea | y |
| **Id\_Proyecto** | Numero | No | Primaria Foránea | y |
| **cantidad** | Numero | 25 | No |  |

**Producción**

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Tipo de dato** | **Amplitud** | **Llave** |
| **Id** | Numero | No | Primaria |
| **Id\_Proyecto** | Numero | No | Foránea |
| **Nombre** | Texto | 512 | No |
| **Cantidad** | Numero | 25 | No |

#### 5.2.4. SCRIPT

/\* Create Tables \*/

CREATE TABLE dotacion

(

id integer NOT NULL DEFAULT NEXTVAL(('"dotacion\_id\_seq"'::text)::regclass), nombre text NULL, descripcion text NULL, stock integer NULL, estado smallint NULL, deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE empleado

(

id integer NOT NULL DEFAULT NEXTVAL(('"empleado\_id\_seq"'::text)::regclass), ci text NULL, nombre text NULL, apellido text NULL, edad integer NULL, sexo varchar(1) NULL, tipo\_id integer NOT NULL, deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE empleado\_dotacion

(

id integer NOT NULL DEFAULT

NEXTVAL(('"empleado\_dotacion\_id\_seq"'::text)::regclass), empleado\_id integer NULL, dotacion\_id integer NULL, cantidad integer NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE herramienta

(

id integer NOT NULL DEFAULT NEXTVAL(('"herramienta\_id\_seq"'::text)::regclass), nombre text NULL, descripcion text NULL, stock integer NULL, estado smallint NULL, deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE ingreso

(

id integer NOT NULL DEFAULT NEXTVAL(('"ingreso\_id\_seq"'::text)::regclass), descripcion text NULL, total decimal(8,2) NULL, fecha date NULL, empleado\_id integer NULL,

deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE ingreso\_dotacion

(

id integer NOT NULL DEFAULT

NEXTVAL(('"ingreso\_dotacion\_id\_seq"'::text)::regclass), ingreso\_id integer NULL, dotacion\_id integer NULL, cantidad integer NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE ingreso\_herramienta

(

id integer NULL DEFAULT

NEXTVAL(('"ingreso\_herramienta\_id\_seq"'::text)::regclass), ingreso\_id integer NULL, herramienta\_id integer NULL, cantidad integer NULL,

created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE ingreso\_material

(

id integer NOT NULL DEFAULT

NEXTVAL(('"ingreso\_material\_id\_seq"'::text)::regclass),

ingreso\_id integer NULL,

material\_id integer NULL, cantidad integer NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE material

(

id integer NOT NULL DEFAULT NEXTVAL(('"material\_id\_seq"'::text)::regclass), nombre text NULL, descripcion text NULL, cantidad integer NULL, unidad text NULL, deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE prestamo

(

id integer NOT NULL DEFAULT NEXTVAL(('"prestamo\_id\_seq"'::text)::regclass), nro text NOT NULL, fecha date NULL, fecha\_devolucion date NULL, proyecto\_id integer NULL, deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE prestamo\_herramienta

(

id integer NOT NULL DEFAULT

NEXTVAL(('"prestamo\_herramienta\_id\_seq"'::text)::regclass), prestamo\_id integer NULL, herramienta\_id integer NULL, tiempo integer NULL, cantidad integer NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE proyecto

(

id integer NOT NULL DEFAULT NEXTVAL(('"proyecto\_id\_seq"'::text)::regclass), nombre text NULL, descripcion text NULL, empleado\_id integer NULL, deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE proyecto\_material

(

id integer NOT NULL DEFAULT

NEXTVAL(('"proyecto\_material\_id\_seq"'::text)::regclass), proyecto\_id integer NULL, material\_id integer NULL,

cantidad integer NULL,

created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE tipo

(

id integer NOT NULL DEFAULT NEXTVAL(('"tipo\_id\_seq"'::text)::regclass), nombre text NULL,

deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

)

;

CREATE TABLE users

(

id integer NOT NULL DEFAULT NEXTVAL(('"users\_id\_seq"'::text)::regclass), name text NULL, email varchar(50) NULL, password text NULL, empleado\_id integer NULL,

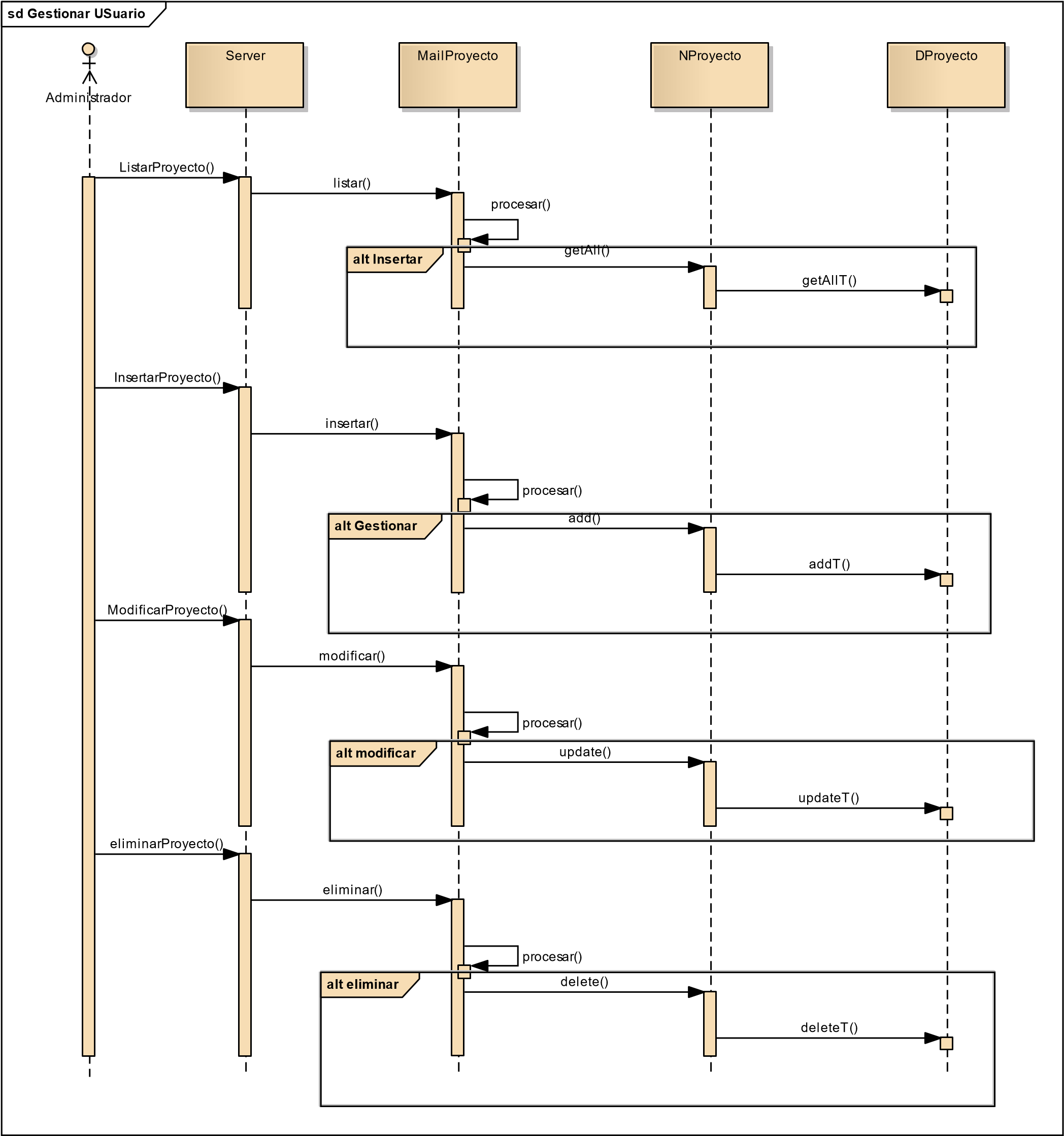
deleted\_at timestamp without time zone NULL, created\_at timestamp without time zone NULL, updated\_at timestamp without time zone NULL

);

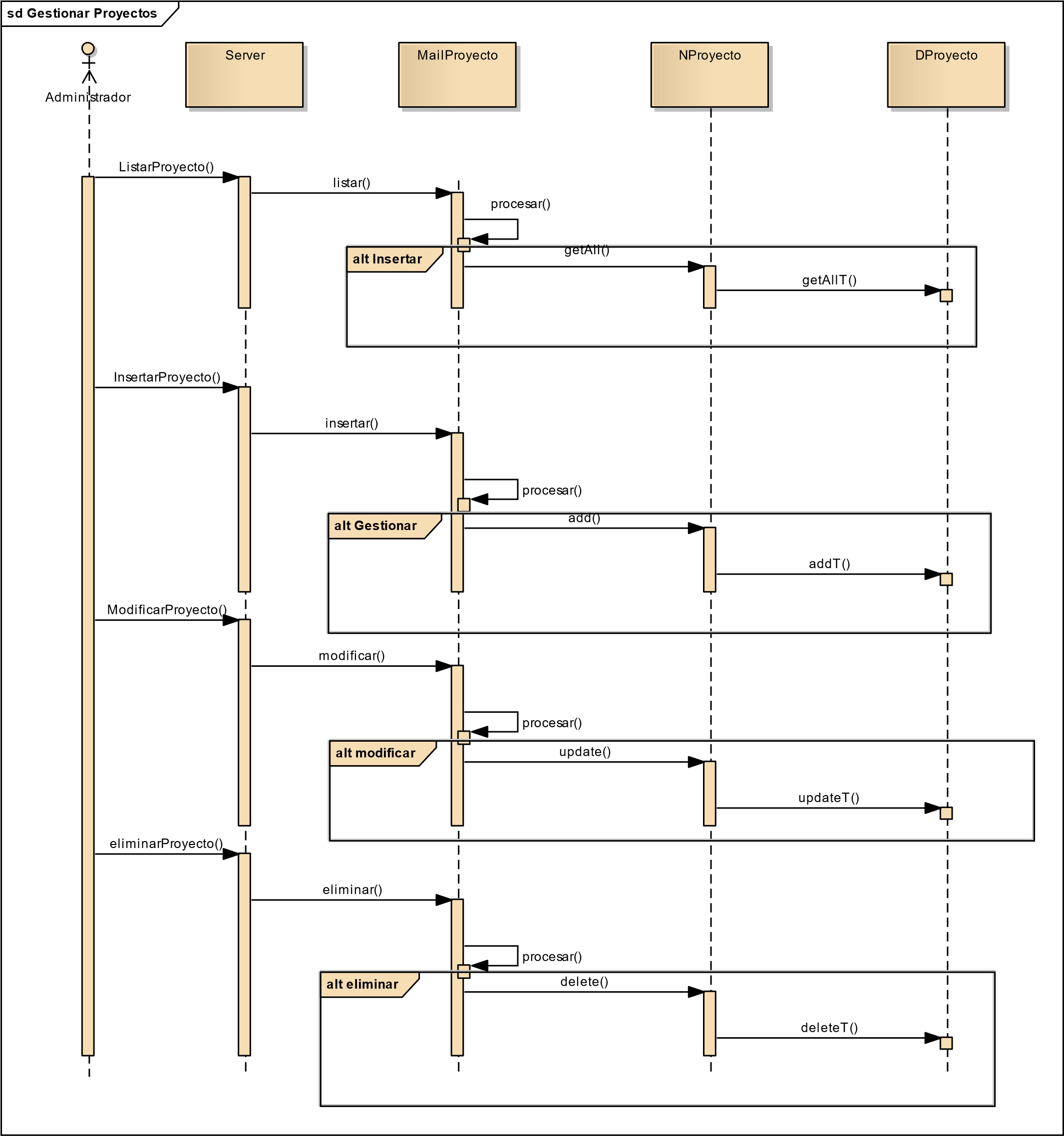
### 5.2.4 DISEÑO DE CASOS DE USO

##### 5.2.4.1 DIAGRAMAS DE SECUENCIA

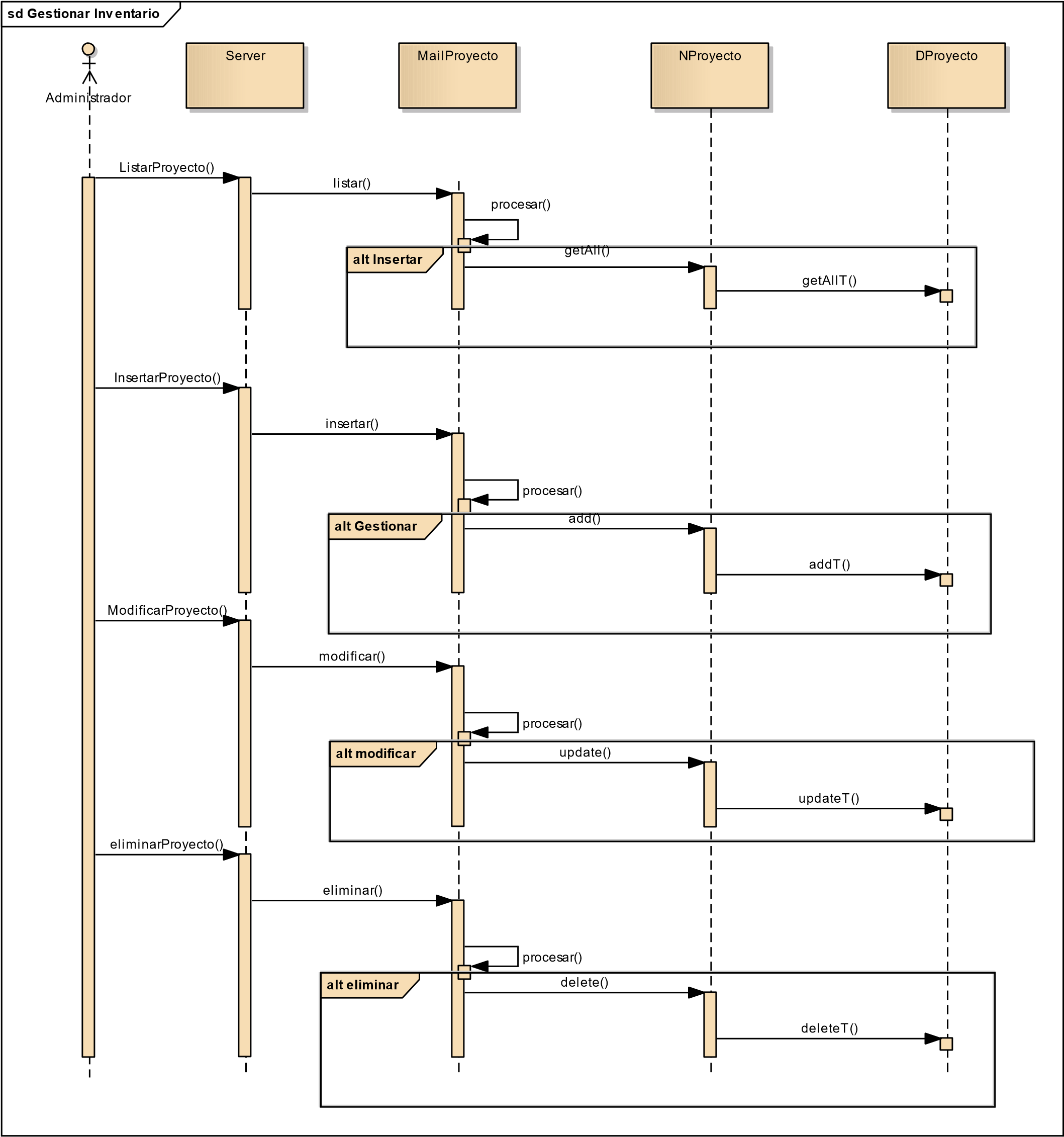
**CU1. GESTIONAR USUARIO**



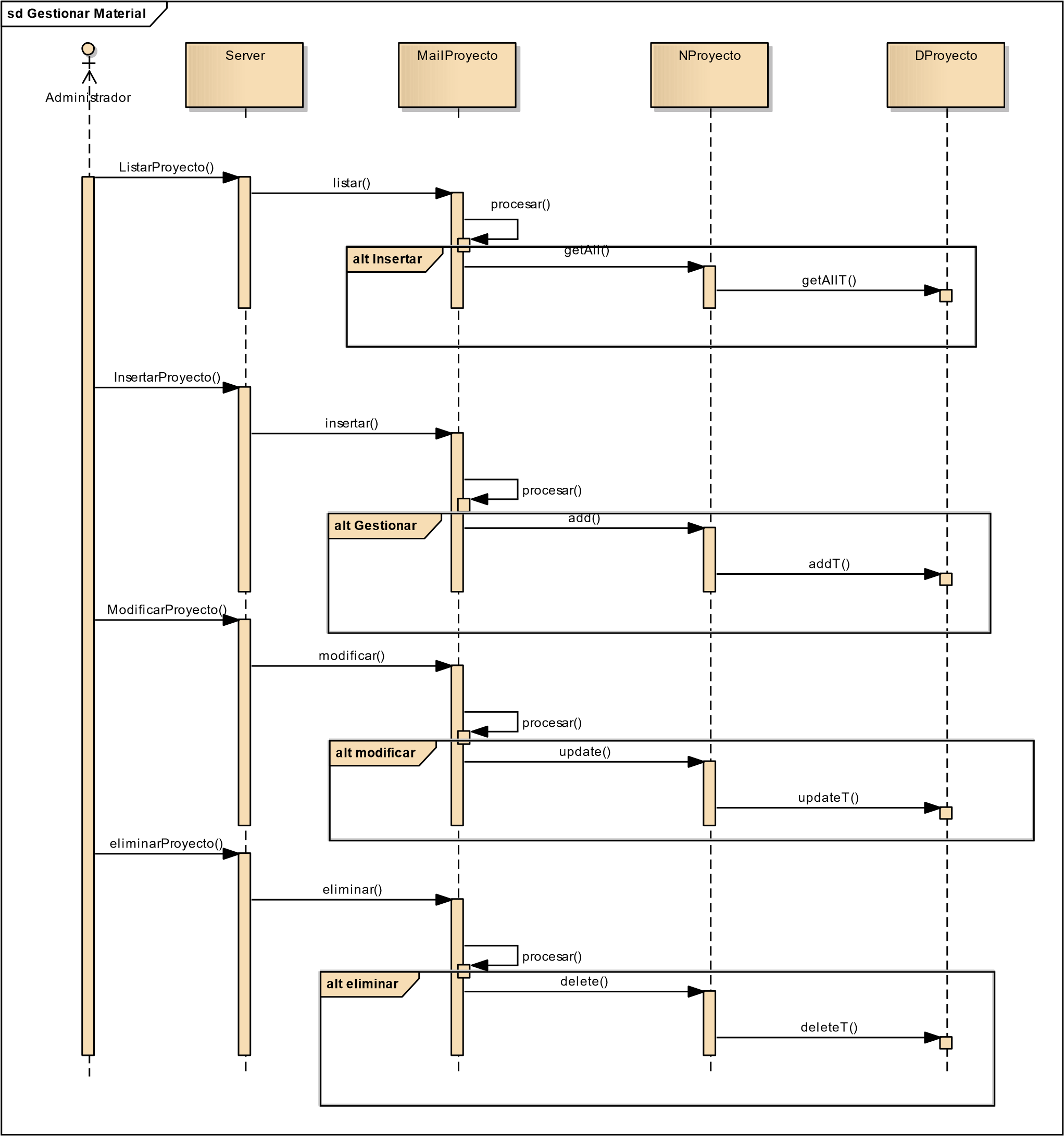
**CU2. GESTIONAR PROYECTO**



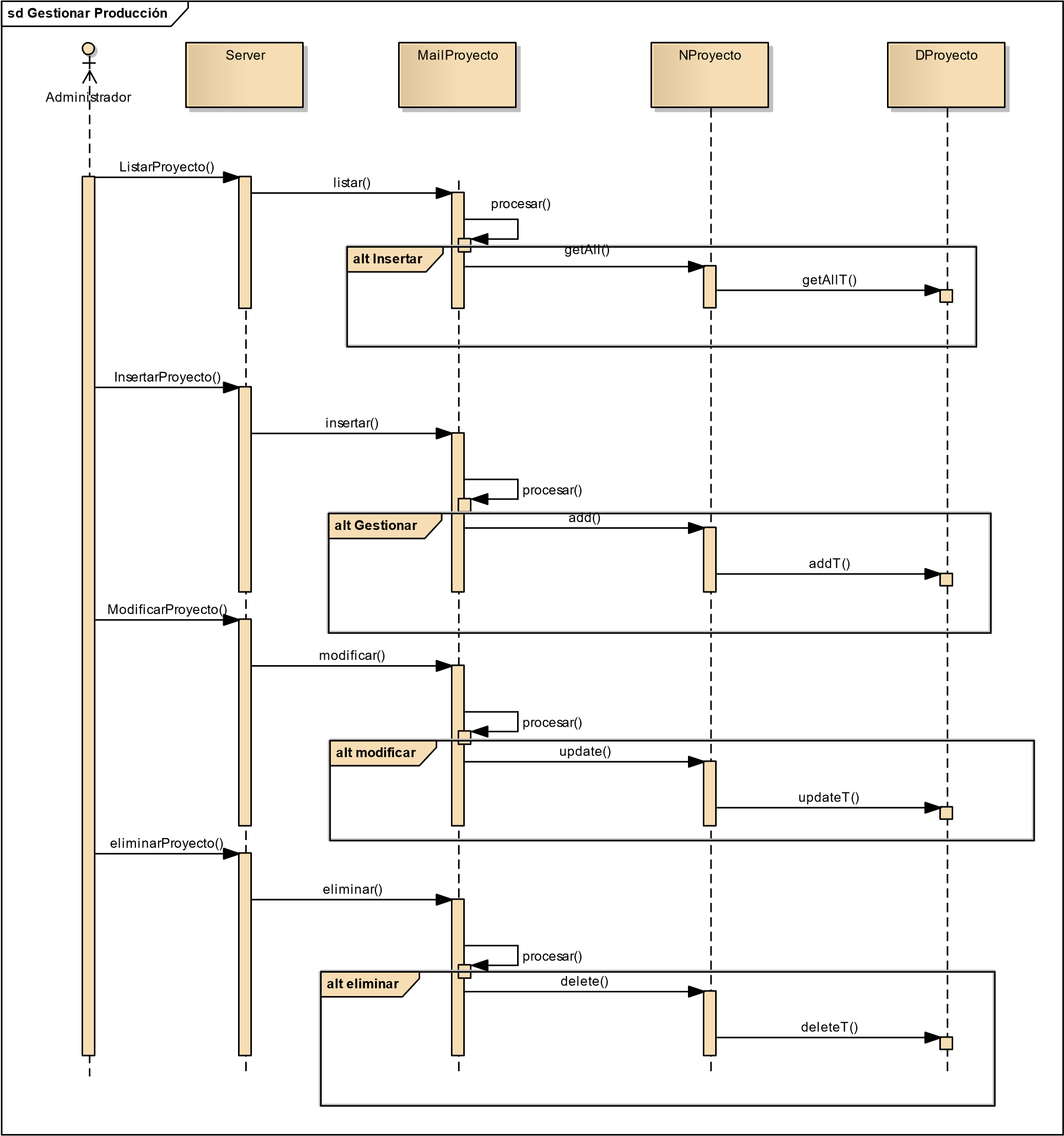
**CU3. GESTIONAR INVENTARIO**



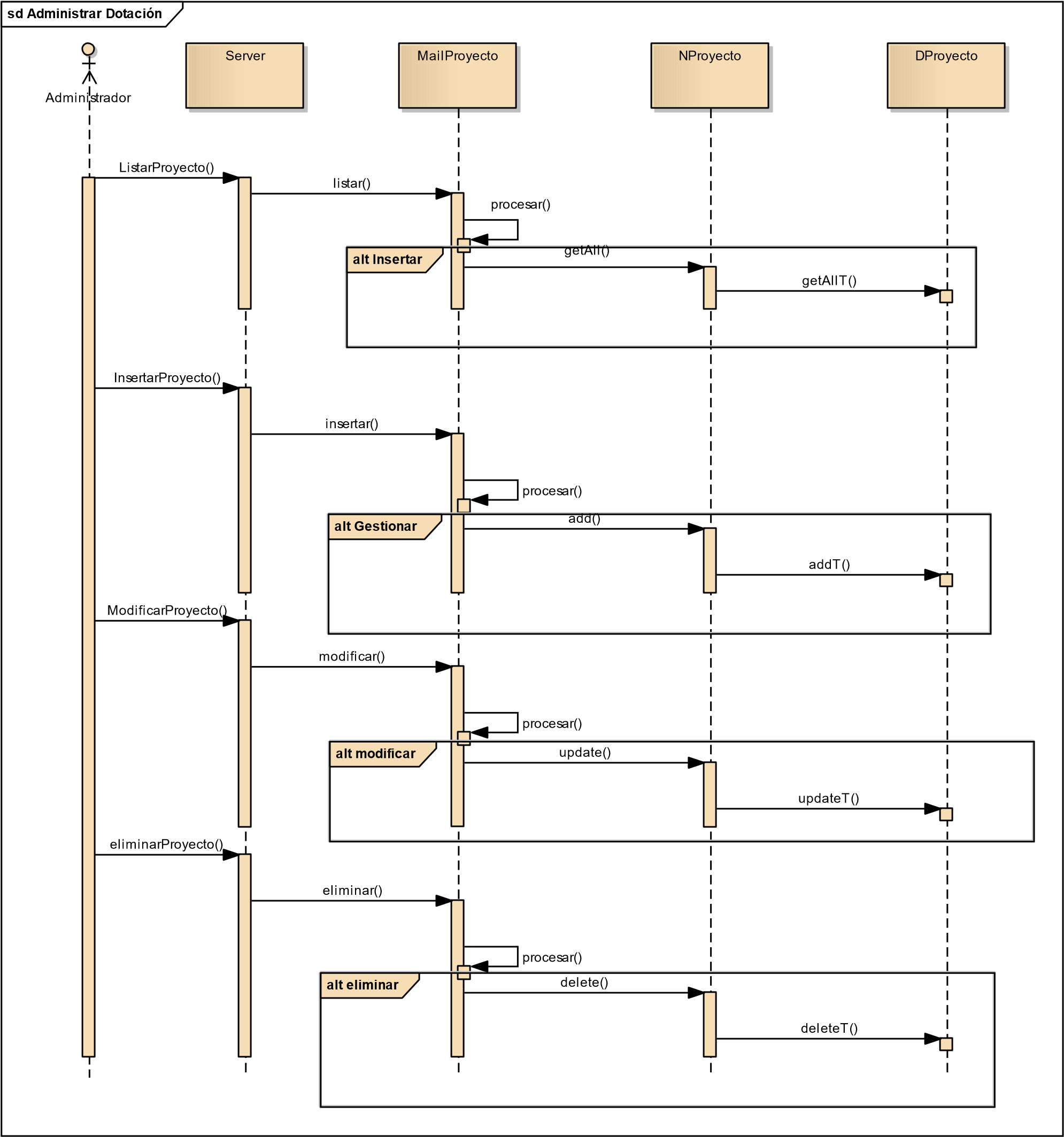
CU4. GESTIONAR MATERIAL



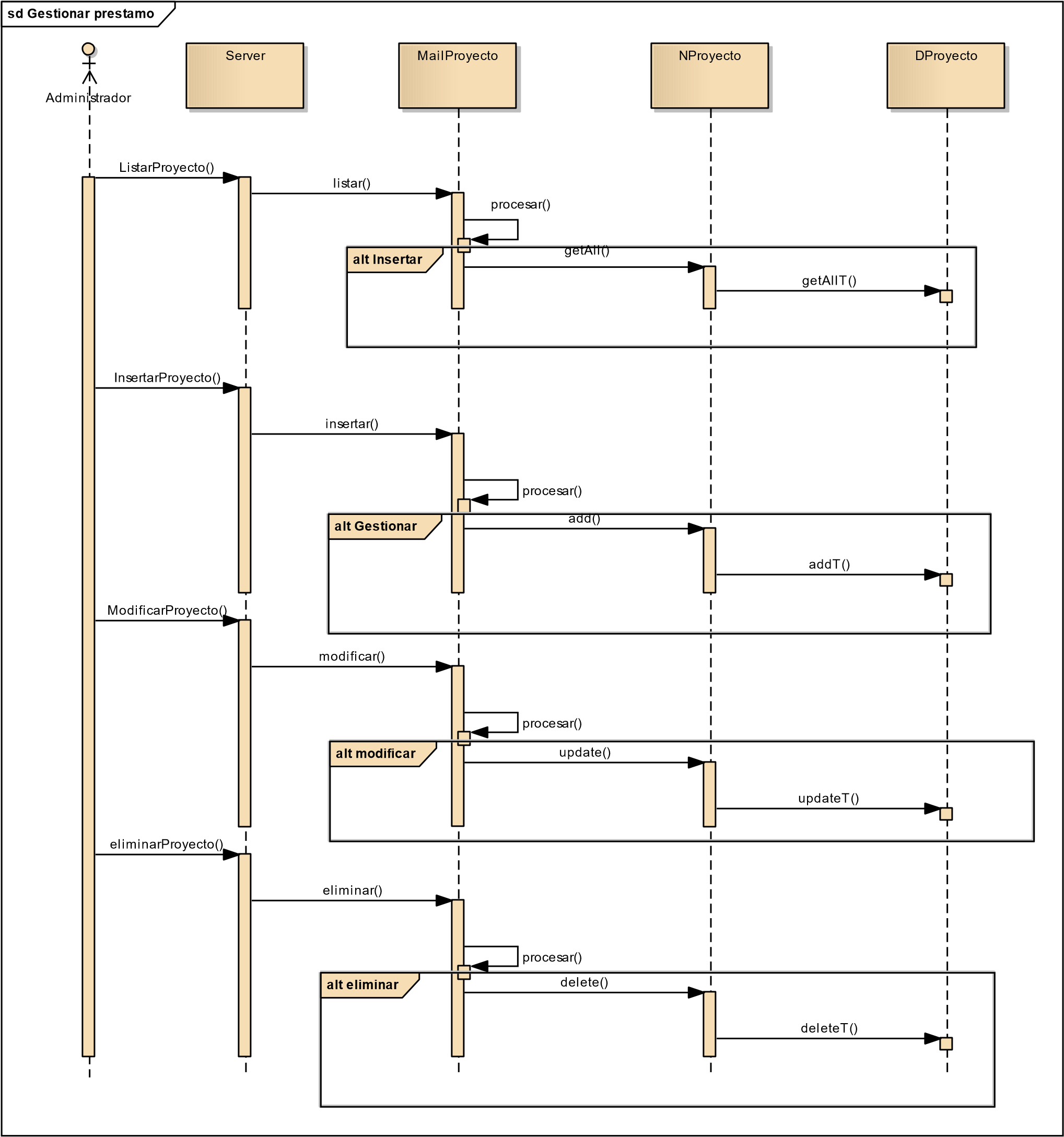
**CU5. REGISTRAR PRODUCCIÓN**



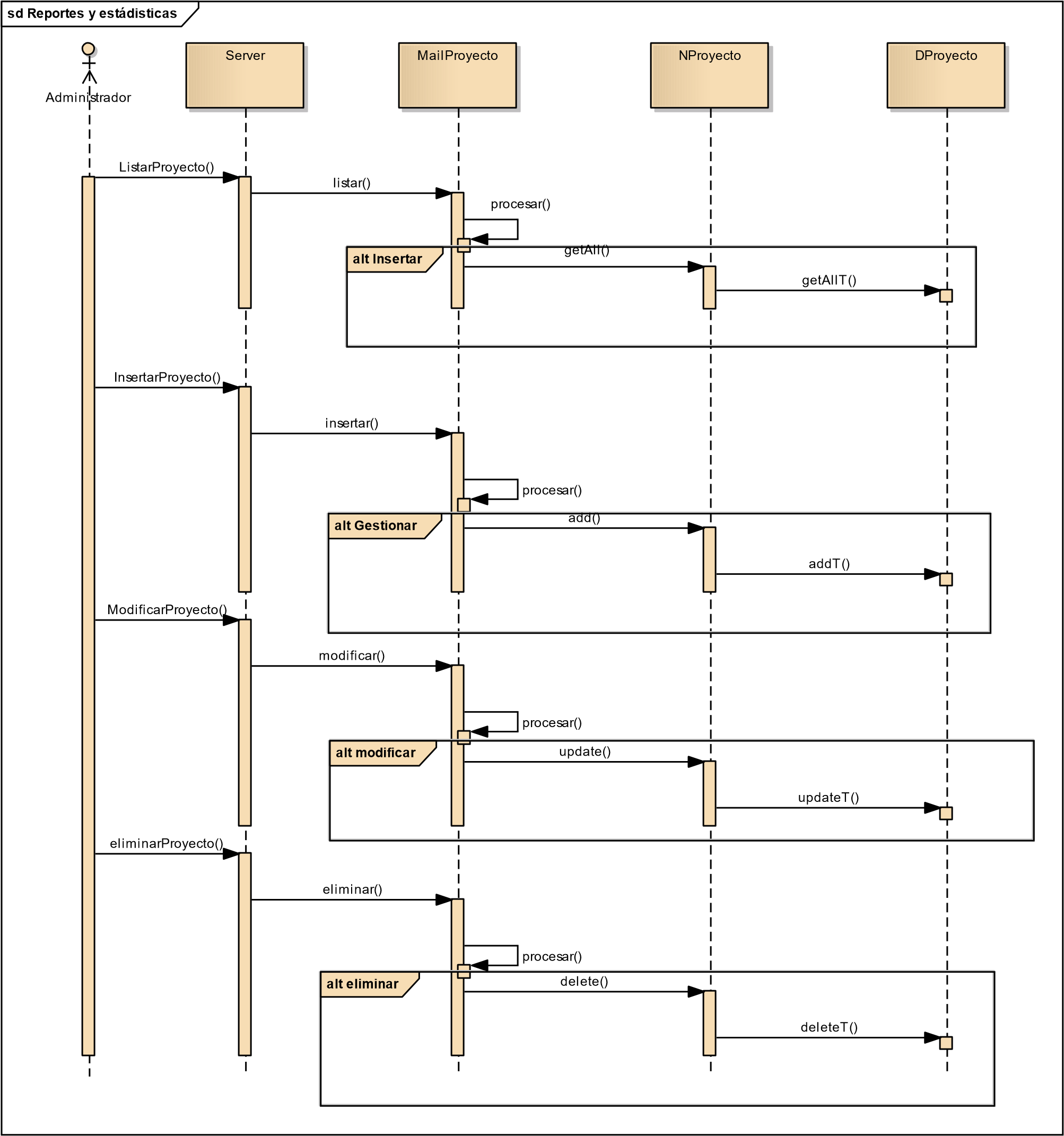
**CU6. ADMINISTRAR DOTACIÓN**



**CU7. GESTIONAR PRESTAMO**



**CU8. REPORTES Y ESTADISTICAS**



# CAPÍTULO VI: IMPLEMENTACIÓN

## 6.1 PLATAFORMA DE DESARROLLO DE SOFTWARE

#### 6.1.1 LENGUAJE DE PROGRAMACIÓN

**NetBeans Java IDE 8.0.2**

El IDE NetBeans es un entorno de desarrollo integrado (IDE) modular y basado en estándares, escrito con el lenguaje de programación JAVA. El proyecto de NetBeans consta de un IDE de código abierto y gran variedad de funciones; y una plataforma para aplicaciones de cliente enriquecidas que se puede utilizar como marco genérico para crear cualquier tipo de aplicación web. El lenguaje de programación utilizado para el desarrollo del proyecto es el lenguaje JAVA, el cual provee los componentes necesarios para la implementación de los requisitos del proyecto.

#### 6.1.2 BASE DE DATOS

**PostgreSQL**

PostgreSQL es un potente sistema de base de datos objeto-relacional de código abierto. Cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de fiabilidad e integridad de datos. Se ejecuta en los principales sistemas operativos que existen en la actualidad. Una base de datos de clase empresarial, PostgreSQL cuenta con características avanzadas tales como Multi-Version Control de concurrencia (MVCC), puntos en tiempo de recuperación, tablespaces, replicación asincrónica, transacciones anidadas (savepoints), respaldos online/hot, un sofisticado query planner/optimizer. Soporta el conjunto de caracteres internacional, codificaciones de caracteres multibyte, Unicode, mayúsculas y minúsculas. Es altamente escalable, tanto en la enorme cantidad de datos que puede manejar y en el número de usuarios concurrentes que puede administrar. Hay sistemas activos en PostgreSQL en entornos de producción que manejan más de 4 terabytes de datos. Algunos límites y caracteristicas generales que se incluyen en PostgreSQL son:

|  |  |
| --- | --- |
| **Tamaño máximo de la Base de datos** | **Ilimitado** |
| **Tamaño máximo de la tablas** | 32 TB |
| **Tamaño máximo de la fila** | 1.6 TB |
| **Tamaño máximo para cada campo** | 1 GB |
| **Máximo de filas por tabla** | Ilimitado |
| **Maximo de columnas por tabla** | 250-1600 dependiendo del tipo de columna |
| **Máximo de indices por tabla** | Ilimitado |

#### 6.1.3 SISTEMAS OPERATIVOS

**Sistema operativo Linux Windows 10**

No obstante, nuestro sistema de información soporta cualquier plataforma de desarrollo en el entorno móvil, web y de escritorio.

El lenguaje de programación PHP actualmente está disponible para cualquier sistema operativo de escritorio, tanto para pruebas locales como para desarrollo en el entorno web.

#### 6.1.4 Otros

**Architect Enterprise**

Es una herramienta que se utiliza en el desarrollo de software para todo tipo de industrias que permite el diseño UML y la cual abarca por completo el desarrollo de software, desde la administración de requerimientos como el diseño, pruebas y mantenimiento. Esta aplicación incluye los siguientes diagramas: Clases, Objetos, Paquetes, Componentes, Despliegue, Casos de uso, Comunicación, Interacción, Actividad, Secuencia, Tiempo, Navegación, Capas Máquina de estados.

# BIBLIOGRAFÍA

Jacobson - booch - rumbaugh/Andrés Otero (2000). Proceso unificado de desarrollo de software (4ta) (3). Madrid-España: ADDYSON WESLEY.

Nicolás Tedeschi (2008). Web Services un ejemplo práctico. Recuperado en: https://msdn.microsoft.com/es-es/library/bb972248.aspx [2021, 16 de mayo].

Evelio Martinez (2011). El correo electrónico y su historia. Recuperado en: http://www.eveliux.com/mx/El-correo-electronico-y-su-historia.html [2021, 16 de mayo].

Sajid Demian Lonngi Reyna (2009). Correo Electrónico. Recuperado en: https://sites.google.com/site/competenciaseinovaciones09/correo-electronico [2021,16 de mayo].

Universidad de Alicante (2016). Protocolos de correo electrónico: SMTP, POP3 e IMAP.

Recuperado en:

https://moodle201617.ua.es/moodle/pluginfile.php/62202/mod\_resource/content/9/comunicacion /page\_05.htm [2021, 17 de mayo].

Netbeans (2017). Bienvenido a NetBeans. Recuperado en:

https://netbeans.org/index\_es.html [2021, 17 de mayo].

Yendes Edwin (2015). PostgreSQL. Recuperado en: https://iutablog.wordpress.com/2017/05/11/postgresql/ [2021, 18 de mayo].