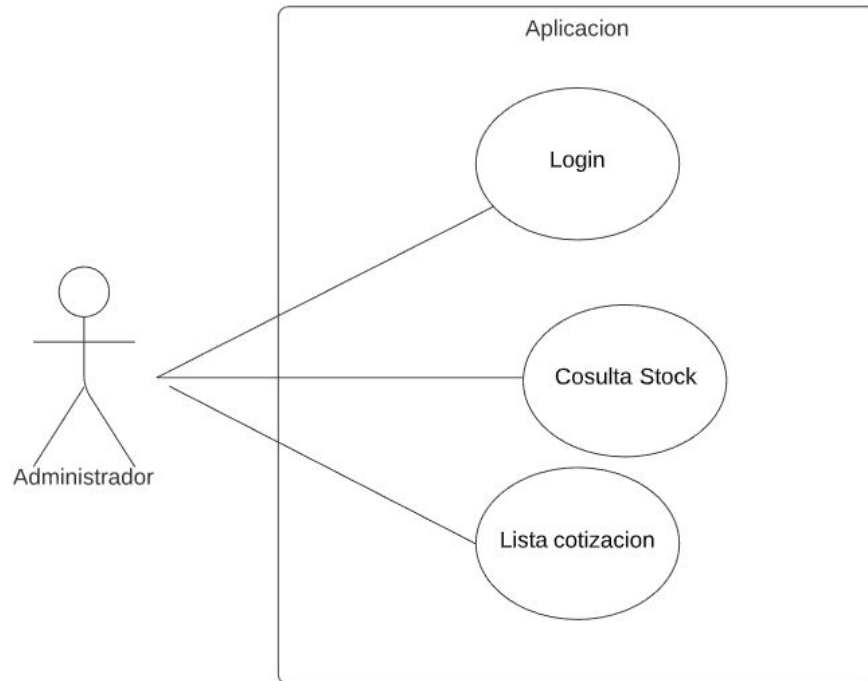


## Diagrama Casos de Uso

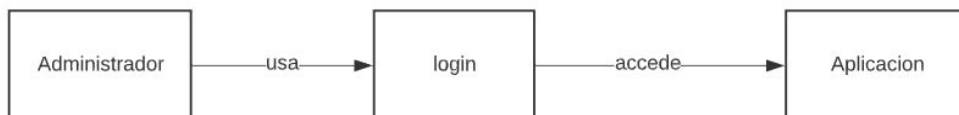
Black Rain | September 1, 2020

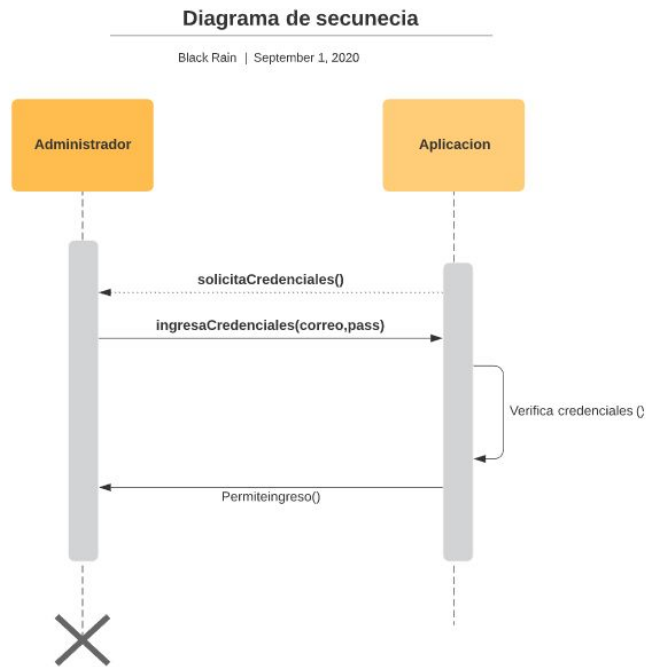


### 1. Modelo UML de caso de uso Login

## Modelo de clases Conceptuales

Black Rain | September 1, 2020





contratos:

Nombre: ingresoCredenciales(correo, pass)

Responsabilidades: Lanzar actividad para acceder a la aplicación

Tipo: Sistema

Referencias cruzadas: caso de uso login

Notas:

Excepciones:

Salida: se presenta actividad para ingresar datos

Precondiciones: Usuario administrador

Poscondiciones: verificar credenciales

Nombre: verificarcredenciales()

Responsabilidades: verificación de credenciales del administrador

Tipo: Sistema

Referencias cruzadas: caso de uso login

Notas:

Excepciones:

Salida: se presenta actividad para ingresar datos

Precondiciones: el administrador ingresa sus credenciales

Poscondiciones: se permite ingresar a la aplicación

Nombre: PermitelIngreso()

Responsabilidades: una vez verificadas las credenciales, lanza la actividad principal de la aplicación

Tipo: Sistema

Referencias cruzadas: caso de uso login

Notas:

Excepciones:

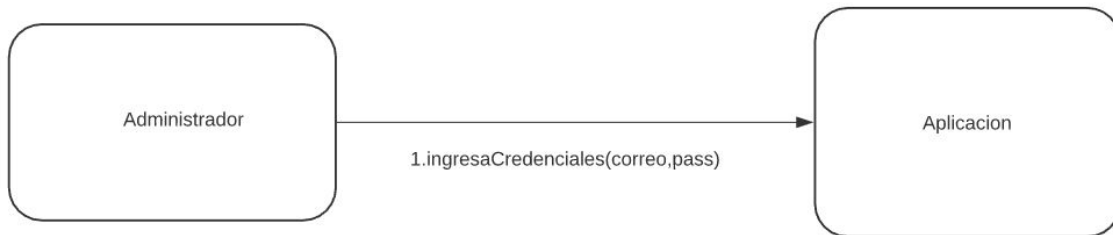
Salida: se presenta actividad principal

Precondiciones: las credenciales han sido verificadas correctamente

Poscondiciones:

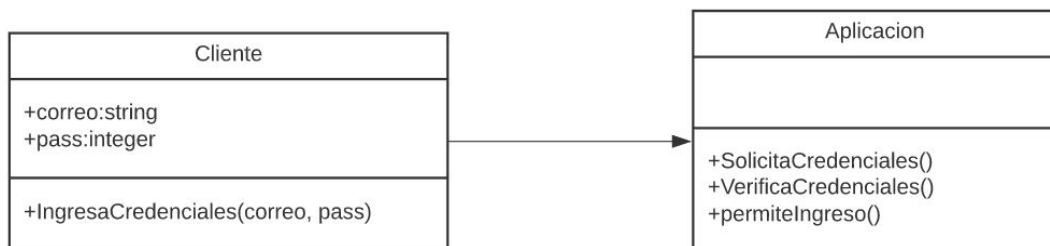
### Diagrama de Colaboracion

Black Rain | September 1, 2020



### Diagrama de Clases

Black Rain | September 1, 2020



## 2 Modelo UML Caso de uso Consulta de Stock

Modelo de clases conceptuales Consulta Stock

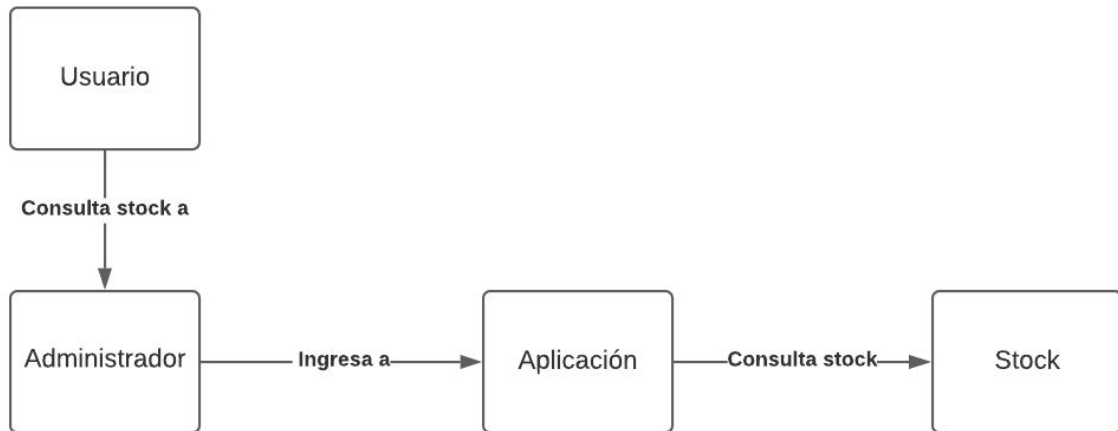


Diagrama de Clases Consulta Stock

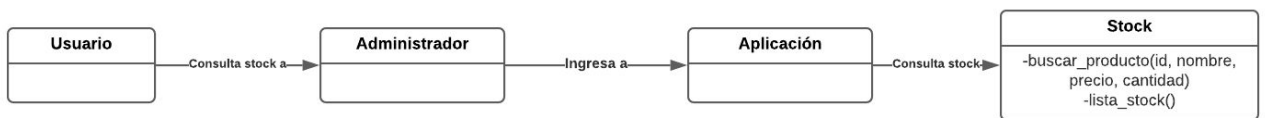
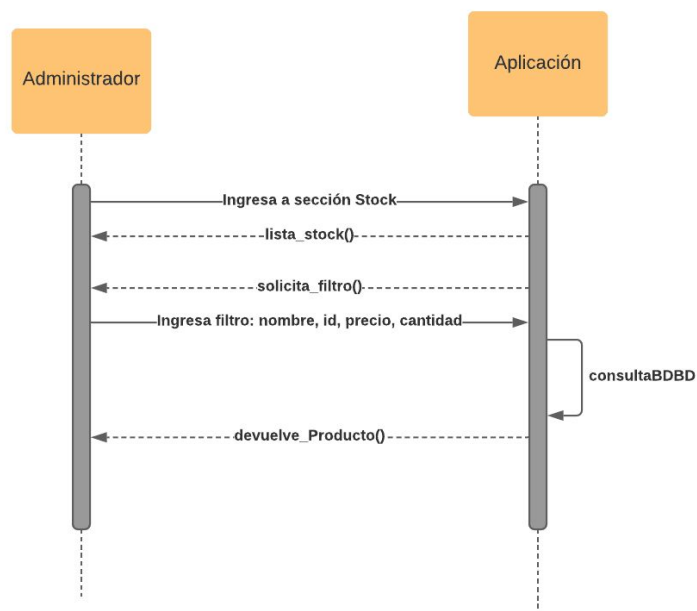


Diagrama de secuencia de Consulta Stock



## Contratos

**Nombre:** buscar\_Producto(id, nombre, precio, cantidad)

**Responsabilidades:** Es el encargado de realizar la búsqueda de algún producto según sea ingresado por el administrador.

**Tipo:** Sistema.

**Referencias cruzadas:**Caso de uso login.

**Notas:**

**Excepciones:** Si está mal ingresado el filtro, mostrar que existe un error. Si hay algún error de conexión con la BD mostrar un aviso.

**Salida:** Devuelve en pantalla el artículo/producto que se busca.

**Precondiciones:** Que sea ingresado con el formato correspondiente. Que esté en funcionalidad la BD.

**Poscondiciones:**

**Nombre:** lista\_Stock (id, nombre, precio, cantidad)

**Responsabilidades:** Es el encargado, luego de que se ingresa a la sección de stock, mostrar todo lo que se encuentra dentro de este.

**Tipo:** Sistema.

**Referencias cruzadas:**Caso de uso login.

**Notas:**

**Excepciones:** Si hay algún error de conexión con la BD mostrar un aviso.

**Salida:** Devuelve en pantalla la lista de artículos del stock.

**Precondiciones:** Que esté en funcionalidad la BD.

**Poscondiciones:**

**Nombre:** solicita\_Filtros (id, nombre, precio, cantidad)

**Responsabilidades:** Es el encargado, luego de que se ingresa a la sección de stock, mostrar todo lo que se encuentra dentro de este.

**Tipo:** Sistema.

**Referencias cruzadas:**Caso de uso login.

**Notas:**

**Excepciones:** Si hay algún error de conexión con la BD mostrar un aviso.

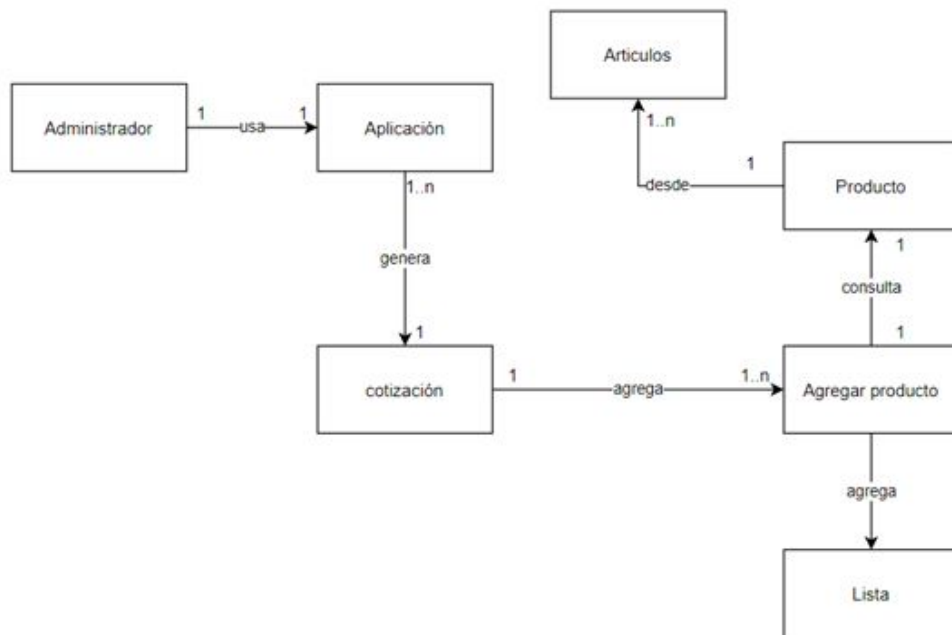
**Salida:** Devuelve en pantalla la lista de artículos del stock.

**Precondiciones:** Que esté en funcionalidad la BD.

**Poscondiciones:**

### 3 Modelo UML Caso de Uso Lista Cotización

#### Mapa de clases conceptuales



- Diagrama de secuencia del sistema



Contratos:

**Nombre:**Generar\_cotizacion(entero rut)  
**Responsabilidades:**Lanzar actividad para crear lista cotizacion  
**Tipo:**Sistema  
**Referencias cruzadas:**caso de uso generar cotización  
**Notas:**  
**Excepciones:**  
**Salida:** código rut  
**Precondiciones:**Usuario administrador  
**Poscondiciones:**Hacer uso del código rut

**Nombre:**Crear\_listacotizacion()  
**Responsabilidades:**Generar una nueva colección con el rut del usuario, en la base de datos.  
**Tipo:**Sistema  
**Referencias cruzadas:**caso de uso generar cotización  
**Notas:**  
**Excepciones:**Cotización ya creada, eliminar cotización anterior  
**Salida:**código documento  
**Precondiciones:**  
**Poscondiciones:** Productos a agregar pertenecientes a colección

**Nombre:**buscar\_producto()  
**Responsabilidades:**Consultar la existencia y disponibilidad del producto en la base de datos  
**Tipo:**Sistema  
**Referencias cruzadas:**caso de uso generar cotización  
**Notas:**  
**Excepciones:**  
**Salida:**booleano para confirmación de existencia de producto y entero con cantidad disponible  
**Precondiciones:**  
**Poscondiciones:**

**Nombre:**existe\_stock()  
**Responsabilidades:**Entregar la respuesta generada por la base de datos, y esperar por confirmación  
**Tipo:**Sistema  
**Referencias cruzadas:**caso de uso generar cotización  
**Notas:**  
**Excepciones:**  
**Salida:** nombre artículo y cantidad  
**Precondiciones:**Existe el stock deseado en la base de datos  
**Poscondiciones:**

**Nombre:**agregar\_producto()  
**Responsabilidades:**Generar colección con el producto y cantidad  
**Tipo:**Sistema

**Referencias cruzadas:** caso de uso generar cotización

**Notas:**

**Excepciones:**

**Salida:**

**Precondiciones:**

**Poscondiciones:**

**Nombre:** producto\_agregado()

**Responsabilidades:** Entregar confirmación de producto agregado y esperar nueva o término de cotización

**Tipo:** Sistema

**Referencias cruzadas:** caso de uso generar cotización

**Notas:**

**Excepciones:**

**Salida:**

**Precondiciones:**

**Poscondiciones:**

**Nombre:** Terminar\_cotizacion()

**Responsabilidades:** Terminar el colección, y entregar código colección a la componente encargada de generar el PDF

**Tipo:** Sistema

**Referencias cruzadas:** caso de uso generar cotización

**Notas:**

**Excepciones:**

**Salida:** Código de colección

**Precondiciones:**

**Poscondiciones:**

Diagrama de colaboración:

