

# Informe de Avance

Fecha: 24/06/2020

URL del repositorio\*:

<https://github.com/jemi14/Grupo-19-OO2-2020>

Nombre del Proyecto\*:

Grupo-19-OO2-2020

**El BRANCH:** master.

Nombre de la BDD\*:

grupo-19-bdd-oo2-2020

Nombre de usuario de GIT:

Griselda Benitez: [jemi14](#)

Marcelo Wolf: [MarceloWolf](#)

Agustin Trotta: [WindarkRey](#)

Nicolas Romero: [NRomeroO \(Abandonó\)](#)

Supuestos:

Seguimos agrandando el DC pero sigue siendo un DC con pocas entidades, vamos escalando poco a poco. Los sueldos se calculan en las consultas web, no se persisten, se calculan por mes para el vendedor elegido. Asumimos que cuando un comprador llega a uno de los locales, solo interactúa con un único vendedor (Se asigna aleatoriamente un vendedor al hacer el pedido, entre los vendedores de ese local, obviamente). Si hay stock suficiente en una sucursal, a la fecha del pedido le sumamos una semana para determinar la fecha de llegada del pedido.

Si el stock no alcanza y necesitamos de mas de una sucursal el pedido llega 15 días mas tarde.

Si se piden mas productos de los que hay en todas las sucursales, se toma el pedido igual, pero no se define la fecha de entrega, se le avisa al comprador que cuando tengamos stock se termina el pedido (no implementado).

Para dar alta a un Local, primero hay que dar de alta a un Gerente, ya que hay una relación Uno a Uno, por lo cual cada Local tiene un solo gerente y no pueden repetirse.

Para generar un pedido el cliente ya tiene que estar cargado, así aparece en la lista de clientes para elegir.

Inconvenientes/Obstáculos:

Muchos de los inconvenientes planteados en las entregas anteriores han sido superados.

Nos quedo en el tintero dar privilegios según Usuario en particular, solo tenemos privilegios generales como admin, vendedor, gerente, cliente, pero nos hubiera gustado que el vendedor solo pueda hacer cosas en "su local", lo mismo para el gerente.

Cada pedido es de un único pedido, también nos hubiera agradado hacer un carrito de pedidos, avanzamos con la estructura de un solo producto y cuando nos dimos cuenta ya nos era imposible escalar a varios por tiempo y diseño.

\* Campos Obligatorios.

Mas en lo personal y grupal, perdimos a uno de los integrantes puesto que estaba muy atrasado (Nicolás Romero), Trotta Agustín nunca pudo instalar las tecnologías necesarias por impedimentos de hardware, por este motivo se encargo de la carga de la base de datos. Como ya se hablo en las clases Wolf Marcelo también tuvo inconvenientes técnicos por lo cual fue el encargado de la creación de los Casos de Prueba y Diagramas de Clase. Además, entre ellos han construido el Login. Las tareas vinculadas a spring fueron realizadas por Griselda Benitez, estando todos en constante comunicación.

## Objetivos Alcanzados\*:

**ABM: Producto – Cliente – Gerente – Vendedor**

**Alta: Stock – Local (Bajas y modificaciones no las hicimos para avanzar)**

**Alta: Pedido (Se crea un pedido, de un producto, con un Cliente, una sucursal, verificando el stock).**

**Vista de pedidos.**

Los pedidos superan los siguientes 3 casos:

**Caso 1:** Si se pide un producto a un negocio que tiene stock, se genera el pedido sin problema, y se lo entrega dentro de los 7 días

**Caso 2:** Se piden mas productos de los que tenemos (Sumando todos los locales), se genera el pedido, pero se le pone inactivo, cuando se tenga más stock recién ahí se activara el pedido.

**Caso 3:** Tenemos stock, pero necesitamos mas de un local, si el local elegido no tiene el stock buscamos al mas cercano, si con la suma de estos dos alcanza el pedido se genera sin problema, solo que el envío demora 15 días en vez de 7 (Regla de negocio). Si con el 2do local tampoco nos alcanza y necesitamos un 3ro aun no lo solucionamos (trabajando en eso).

**Login funcionalidades con permisos.**

**Pedidos separados por aceptados y pendientes.**

**Pedidos entre fechas.**

**Pedidos entre fechas y vendedores.**

**Calculo de sueldo por mes por vendedor.**

**Generación de facturas.** (Se generan en el directorio raíz del proyecto para evitar compatibilidad entre Windows y Linux)

**Login disponibles:**

User: admin Pass: admin

User: gerente Pass: gerente

User: vendedor Pass: vendedor

User: cliente Pass: cliente

**¿Cómo usarlo?:**

Ejecutar primero los comandos sql, para crear las tablas y generar los insert para las pruebas.

Luego iniciar Spring, siempre recordar configurar el horario en -03UTC y el user y pass de MySQL, errores típicos que impiden ejecutar.

Recomendamos probar todo con admin primero, para luego apreciar las limitaciones de otros privilegios.

**Para simplificar el acceso**

**Link Programa:** <https://github.com/jemi14/Grupo-19-OO2-2020/tree/master/GRUPO-19-OO2>

**Link SQL:** <https://github.com/jemi14/Grupo-19-OO2-2020/blob/master/Final-24-6.sql>

**Link documentos:** <https://github.com/jemi14/Grupo-19-OO2-2020/tree/Documentacion>

\* Campos Obligatorios.