

Buenas tardes. Algunas dudas, sugerencias o preguntas :

* Si tienes familias de artículos, no creo que sea necesario Categorías de artículos. Puedes explicarme porque lo haces así ?

- En un principio, pensé incluir solamente 'Familia' de artículos como criterio de agrupación. La idea es, que en la emisión de informes de ventas, se pueda desglosar el porcentaje de ventas teniendo en cuenta este criterio (para que así el usuario administrador proceda a impulsar las ventas de artículos cuya 'Familia' no cumpla con la cuota de ventas ideal).
- Éste criterio de agrupación, aunque es válido, me pareció que puede arrojar un pobre resultado estadístico en los informes, por lo que decidí incluir una nueva capa de agrupación, que sería 'Categoría'.
- Por ejemplo, un negocio dedicado a la venta de ropa, puede tener la 'Familia' complementos, y dentro de complementos, las 'Categorías' cinturones, corbatas, bolsos, etc.

* Según tu manual, el proceso de venta inicia permitiendo el escaneo de códigos de barras de artículos....

¿ Y cuando seleccionamos el cliente a quien le vendemos ?

No especifiqué el proceso de selección porque debido a que es una parte con cierta complejidad, no quería encontrarme con ningún callejón sin salida en la implementación. No pretendo excusarme y asumo la falta de especificación, pero en un principio, mi aplicación móvil hacía consultas directamente a través de la red contra la base de datos MySQL, sin guardar en las tablas locales de SQLite (que era lo que en pensé que debía hacer la aplicación), y ahora lo hace consumiendo una API (JSON) de mi servidor Web (nube), y no sabía qué limitaciones podía encontrar. Igualmente, si me permites, voy a intentar explicar como es el proceso de selección de cliente:

- Una vez el empleado se ha autenticado en el sistema, se muestra un menú (que tendrá diferentes restricciones de acceso dependiendo el perfil de empleado) que permitirá elegir algunas opciones (alta fiscal de nuevo cliente, consultar ventas, cerrar ventas, dotaciones, etc).
- Entre estas opciones, está la de iniciar nueva venta, donde se debe elegir el perfil de cliente que se va a tratar (a través de un spinner).
- Por defecto, si no se especifica, la venta se inicia a un cliente anónimo.
- Si el cliente requiere factura, deberá indicarse el identificador del cliente fiscal (NIF ó CIF del tipo de sociedad) a través de una entrada de texto.
- Si el cliente es un empleado de la empresa, es VIP o tiene ciertos privilegios, éste recibirá un descuento personalizado sobre el importe de la venta (los diferentes tratos que se dan a los clientes dependerán del usuario administrador, reflejados en la tabla de 'Categoría de

clientes').

- La tabla de 'Categorías de clientes' puede recibir nuevos registros (o modificar los existentes) solo en el lado del servidor, mientras que la tabla de 'Clientes' puede modificarse o incluir nuevos registros tanto en el lado del servidor como en la aplicación (para agilizar el alta fiscal de un nuevo cliente en caliente directamente en el TPV).
- Una vez establecido del cliente, ya sí se inicia proceso de venta propiamente dicho.

* El tema de descuentos y promociones para las familias de clientes... yo lo dejaría como ampliación de proyecto.

Es ambicioso y complejo, para un primer proyecto.

- Como he explicado en el punto anterior, me parece una funcionalidad muy útil (además de ser algo común en este tipo de aplicaciones). No obstante, aunque hay bastante tiempo por delante, seguiré tu consejo y dejaré esta implementación como ampliación del proyecto, una vez lo haya finalizado si el tiempo me lo permite.

* Sobre los artículos:

cantidad vendida -> se calculará automáticamente con los registros de ventas.

- La intención es que durante el proceso de cierre de ventas de cada día, se produzca un recuento de la cantidad de cada artículo vendido en ese día. Este recuento se realiza sobre la tabla 'Detalles venta' de SQLite (agrupando por artículo y contabilizando con una consulta típica `SELECT cod_articulo, COUNT(*) FROM Articulos GROUP BY cod_articulo`). Una vez se ha contabilizado qué cantidad de cada artículo se ha vendido en el día, ésta cantidad se sumará al campo 'cantidad vendida' de la tabla 'Artículos' del servidor remoto.

referencia a periodo de descuento ->

¿eso significa un periodo de descuento (fecha inicio - fecha fin) por articulo ?

- Efectivamente, el periodo de descuento está definido por una fecha de inicio y una fecha fin (si la fecha actual está entre esas dos fechas, se aplica el importe de descuento, si no, se aplica el importe habitual). Los artículos que no estén en promoción tendrán ambos campos NULL, con lo que habrá que comprobar previamente este estado.

¿el descuento sera indicando un precio o un porcentaje de descuento sobre el precio de venta ?

- Error mío no haber indicado este campo de 'Precio en promoción' del artículo en la tabla 'Artículos' que es donde debería estar. En un principio pensé crear una tabla independiente con tres campos:
 - Códigos de artículos en promoción, fecha de inicio de promoción y fecha de fin de promoción, pero me di cuenta de que no me iba a aportar ninguna mejora, así que decidí incluir estos tres campos directamente en el artículo y dejarlos NULL en caso de que no estuviesen en promoción, pero por descuido dejé sin indicar el campo 'precio de venta en promoción'.

- El precio en promoción será un precio concreto (no porcentual), ya que es habitual que algunos artículos solamente se vendan durante una promoción y carezcan de un precio base habitual (o directamente que sea 0) sobre el que calcular el precio en descuento.