Nuestro cliente es Manuel Sarmiento Soto, un investigador de un laboratorio de la facultad de farmacia de la US. Este laboratorio tiene un grave problema con la actual gestión del inventario, pedidos realizados, proyectos, etc. Actualmente todos estos datos se apuntan en una libreta, lo que les hace muy incómodas tareas tan simples como, por ejemplo, mirar cuánto queda de un producto. Nosotros proponemos implementar una aplicación que almacenaría en una base de datos los productos disponibles en el inventario, los pedidos realizados, las personas que trabajan en el laboratorio y los proyectos en los que se trabaja, cada una de estas componentes con todos los datos que se tienen sobre ellos.

Lo primero que pensamos almacenar en nuestra aplicación fue el inventario del laboratorio. Para esto, habría que tener guardados los productos que se tienen, cada uno con los ejemplares que hay y con la cantidad que hay en cada uno. Por ejemplo, un ejemplo de producto sería ácido sulfúrico, y de este puede haber 3 botes, cada uno con una cantidad diferente, con lo cual no solo habría que apuntar el número de botes, que nos daría el problema de que la cantidad exacta, y por otra parte, si se apunta solo la cantidad total, no se tendrían detalles sobre cada uno de los ejemplares, como la fecha de caducidad de cada bote, que puede ser diferente. Nuestro cliente clasifica los productos en dos niveles. Primero se clasifican según si son reactivos (productos químicos) o fungibles (productos como guantes, papel, filtros…) y dentro de cada una de estas categorías, se clasifican según la técnica para la que se utilicen. Para la clasificación por técnicas, en nuestra aplicación habría un apartado de técnicas, en las que se podrían añadir todas las que se quisieran, y a estas se le asociarían los productos deseados. Tras añadir una técnica en este apartado, automáticamente se añadiría un filtro en el apartado de productos por el cual podrías buscar solo los asociados a esa técnica.

Otro apartado es el de usuarios. En este apartado habría una lista de todos los usuarios registrados en la aplicación, cada uno con detalles como el tipo de usuario (profesor, alumno de grado, alumno de máster…), el nombre o el nombre de usuario. Según el tipo de usuario se podría acceder a unas características o a otras. Además, los usuarios estarían asociados a los proyectos en los que estuvieran trabajando, que se encontrarían en el apartado de proyectos. En este apartado se podrían consultar los detalles de cada proyecto, como la fecha de inicio, la de fin, el presupuesto o, como se ha dicho antes, las personas participantes en él. Además del presupuesto, almacenaríamos la cantidad de dinero que se haya gastado hasta ese momento, para así poder avisar con una alerta (el sistema de alertas se explicará más adelante) cuando se esté acercando al máximo que se puede gastar.

Uno de los apartados más importantes es el de pedidos. Nuestra aplicación no va a tener la opción de realizar un pedido directamente, ya que, al ser de la universidad de Sevilla, realizar un pedido requiere mucho papeleo con intermediarios que no sería posible sintetizar en nuestra aplicación. Aunque no se pueda realizar un pedido directamente, sí que se almacenaría, los pedidos que se han realizado, para que los usuarios puedan acceder fácilmente y poder saber si cierto producto se ha pedido ya. En esta lista aparecerían detalles como el precio del pedido completo, el precio al que se ha comprado cada producto (ya que puede variar el precio respecto al actual) y, muy importante, quién lo ha pedido, que puede ser una persona o un proyecto. En caso de haberse pedido a nombre de un proyecto, el precio de este se sumaría automáticamente a la cantidad que se lleva gastada de este, con lo cual no habría que preocuparse por añadir los gastos al proyecto manualmente, lo que haría muy cómodo llevar un registro de los gastos realizados a cargo de tal proyecto. Algo importante de este apartado, es que se los pedidos se enlazarían a los productos, pudiendo poner en la página del producto si ya se ha pedido o no. Así, si una persona va a pedir cierto producto, podría entrar en la página de este y, si ya está pedido, se evitaría que se hiciera un pedido repetido (lo que ha pasado anteriormente con el sistema actual).

El último apartado de nuestra aplicación es el de las alertas. En este apartado se almacenarían alertas sobre distintos apartados de la aplicación, las cuales se irían creando y eliminando automáticamente cuando se cumplieran ciertos requisitos. Hay alertas de tres tipos: presupuesto de proyecto, cantidad de producto y ejemplar caducado. La primera, serviría para que, al añadir un pedido a un proyecto o cambiar la cantidad de este gastada manualmente, en caso de haber superado el 90% del presupuesto total, aparecería una ventana que te avisaría de que ha saltado una alerta, y al ir a tal apartado, aparecería en la lista de todas las activas. En caso de que se ampliara el presupuesto o que manualmente se restara a la cantidad gastada, quedando por debajo del 90% del total, esta alerta se eliminaría automáticamente. Otro tipo de alerta es el de cantidad de producto. Es muy parecida a la anterior, solo que en lugar de ser con el presupuesto de un proyecto sería con la cantidad de un producto, siendo que en caso de bajar de cierta cantidad (establecida por el coordinador del laboratorio), saltaría la alerta, y si volviese a subir de tal cantidad, se eliminaría. El último tipo de alerta es el de caducidad de un ejemplar. Para esta, la aplicación comprobaría cada día si hay algún ejemplar caducado de algún producto, y en caso de haberlo, se añadiría la alerta, la cual se eliminaría al eliminar dicho ejemplar o editar su fecha de caducidad. Todas las alertas se clasificarían en nivel alto y bajo, que representan el nivel de prioridad que tiene. Para esta clasificación, todas las alertas en principio se crearían de nivel alto, y sería el usuario el que tendría que bajarlas de nivel. De esta forma, si por ejemplo, queda poco presupuesto de un proyecto, pero se sabe que ya no se va a gastar más, ya no es importante que quede poco dinero porque no se va a pasar del límite, o igualmente si queda poco de un producto pero ya se ha pedido más de este, dejaría de ser prioritaria, puesto que es cuestión de tiempo que se reponga. Con este sistema, haríamos que, manteniendo las alertas (para no olvidarnos de que se está en esa situación), el usuario sepa si realmente tiene que preocuparse por la situación de esta, y no actúe para solucionarlo, lo cual podría causar problemas en ciertos casos.