

Sprint 1 parte 2

Santiago Zuluaga Guerrero

Grupo 13, Ciclo 4 G29

Lo primero que haría, sería crear todo lo visual, pagina de inicio, registro de clientes, registro de productos, página de ventas, página de check out y control de inventario. Ya teniendo lo visual en HTML con CSS, procedería a trabajar en el Backend, hacer que todas las paginas puedan interactuar entre sí. Que cuando llegue el cliente, se pueda registrar y si ya está registrado que acceda a su información para luego proceder a la pagina de ventas. Donde se podrán ver todos los productos disponibles y sus precios, posterior a eso pueda ir a la pagina de checkout y pagar. Al momento de pagar el backend se encargue de actualizar el inventario de los productos que se encuentran en la base de datos dentro del mismo ordenador. Esto haría que el aplicativo sea más económico, pero si la cafetería se expandiera sería un gran problema a largo plazo debido a que desarrollar nuevo código o nuevas utilidades requiere de una actualización de todo el código fuente ya que este se encuentra todo integrado y si algo llegara a fallar sería catastrófico.

Por otra parte, si el café decidiera hacer su aplicativo por microservicios, podría integrar todas las funcionalidades por APIs, que pueden estar escritas en diferentes lenguajes y su escalabilidad sería más fácil. En ese orden de ideas, la base de datos podría estar en la nube, permitiendo que si el café se expande no tendría que preocuparse de tener un servidor propio lo cual ahorraría un poco de dinero, sin embargo, el desarrollo e integración sería más costoso. Incluso, la pasarela de pagos la podría hacer mediante otra integración, la parte visual podría hacerse con plantillas o servicios ya creados. De esa forma, si se quisiera actualizar algo, no es necesario actualizar todo el aplicativo sino cada integración por aparte evitando posibles fallos que afecten todo el sistema.

En ese orden de ideas, las ventajas de tener un sistema monolítico es que es económico y rápido. Sin embargo, el café estaría limitado al crecimiento (expansión de puntos) y desarrollo de nuevas utilidades. Si decidiera tener un sistema por microservicios, sería muy fácil agregar nuevas utilidades y servicios adicionales sin tener que afectar el código base. También, hace que el sistema requiera de menos recursos para funcionar, finalmente, una tecnología por microservicios permite promover y evolucionar el crecimiento del software de una forma más eficiente, fácil y rápida que en uno monolítico.