

En la charla de GOTO 2016 Matt Ranney nos habla sobre las recomendaciones él tiene después de su experiencia escalando la aplicación de Uber. Ranney nos habla de cómo se necesita crear y dejar servicios a lo largo del crecimiento de la empresa, es por esta razón que se usan los microservicios (microservices) que evitan que se tengan monolitos en el sistema. Los micro servicios tienen varias ventajas entre estas su independencia de los otros servicios, lo que permite no solo que se puedan hacer en diferentes lenguajes sino también que, si se cae un servicio, no se va a caer todo el sistema. Como veremos a continuación El expositor Matt Ranney nos cuenta diferentes problemas que tuvo que afrontar a lo largo de su experiencia. El primero fue el Lenguaje, ya que fracturar la cultura, esto quiere decir que los equipos comienzan a agruparse por lenguaje que manejan y se separan del resto. Segundo es el RPC, dado que la mayor dificultad viene cuando se usa request (http/REST) dado que se tiene problemas con la interpretación, es por eso que Matt dice que los servidores no son browsers y que la mejor forma para comunicarse es a través de function calls. El tercero son los Repositorios, en la conferencia nos habla de que tener un solo repositorio como múltiples tienen sus ventajas, los repositorios únicos permiten hacer cambios grandes de todo el programa, pero los múltiples permiten manejar módulos más pequeños. Como comentario cuenta de que Uber paso en un mes de tener 7mil repos a 8mil al siguiente. El cuarto son problemas operacionales; con respecto a este tema habla sobre el trabajo en equipo y como estos pueden ser afectados por el ritmo de otros equipos en la empresa ya que para algunas realizar o completar algunas tareas se necesita que el otro equipo haya concluido su trabajo. Al manejar una gran cantidad de equipo se debe tener en cuenta la comunicación y coordinación. El quinto es la performance, debido a que Uber es una empresa gigante este maneja varias herramientas e idiomas, es por esto que sugiere el uso de dashboards; sin embargo, cada equipo maneja sus propias variables y métricas, para evitar esto se necesita el uso de un dashboard standard y que se haga automáticamente para liberar de esta tarea al equipo. El sexto es el Fanout, ya que la latencia hace que tengas que esperar para responder al usuario debido a que puedes tener un proceso lento en tu call chain que retrasa todo. Para evitar estos se usa tracing. El séptimo es el Tracing, como podemos ver en los ejemplos dados en la presentación, Matt nos muestra una imagen donde hace track de un request y como estos activan una serie de operación que vuelven lento el servicio; en el otro ejemplo pareciera como si el servicio no demorara y corriera correctamente, sin embargo, hace un numero gigante de request a la base de datos. Para concluir recomienda que se haga tracing a un porcentaje estadístico ya que hacerlo para todo sería muy laborioso. El octavo es el Loggin, debido a la gran cantidad de lenguajes y a la rotación del personal se necesita un loggin estructurado para poder contabilizarlos y poder mejorar el entendimiento; también evitar que algunas personas de logeen demasiada data y causen un problema al indexarlo. El noveno

es el Load Testing, dado que, habla de lo importante que es diseñar los sistemas para que estos puedan manejar el tráfico de prueba sin ser contabilizado como algo real. El décimo el Failure testing, ya que en mayor problema es que el personal no quiere someterse al failure testing ya que los errores que tienen su código los hacen sentirse atacados emocionalmente; sin embargo, esto es una parte importante que se tiene que ejecutar y uno tiene que acostumbrar al personal para que estén más predispuestos. El onceavo es de migraciones, uno debe evitar hacer migraciones, solo realizarlas por seguridad o compliance. El doceavo es Open Source; dado que existen problemas con el personal que puede llegar a desanimarse debido a que ellos trabajan en proyectos de manera privada y llegan otros productos externos superiores lo que hace parecer que sus esfuerzos y su trabajo fue inútil. Por ultimo tenemos a las Políticas, en esta parte de la presentación se habla sobre las políticas implementadas en los microservicios, dado que estos pueden violar la propiedad de otros.