**Universidad de lima**

**Facultad De Ingeniería de Sistemas**

**“TAREA 3: MICROSERVICIOS”**

**INGENIERÌA DE SOFTWARE II**

**Profesor: Hernan Quintana**

**Sección: 803**

**Alumna:**

**Centeno Ochoa María Fernanda 20120307**

**Lima – Perú**

**Noviembre, 2017**

ÍNDICE:

[1 INTRODUCCIÓN 3](#_Toc498020907)

[2 REPORTE 1: “MICROSERVICES: THEORY AND APPLICATION” – LUCAS KRAUSE 4](#_Toc498020908)

[2.1 OBJETIVO 4](#_Toc498020909)

[2.2 INTRODUCCIÓN A ARQUITECTURA 4](#_Toc498020910)

[2.3 BENEFICIOS 5](#_Toc498020911)

[3 REPORTE 2: “WHAT I WISH I HAD KNOWN BEFORE SCALING UBER TO 1000 SERVICES” – MATT RANNEY 6](#_Toc498020912)

[3.1 ARQUITECTURA 6](#_Toc498020913)

[3.2 MANEJO DE FALLAS 6](#_Toc498020914)

[3.3 CULTURA ORGANIZACIONAL 6](#_Toc498020915)

[4 PREGUNTA 3 7](#_Toc498020916)

[4.1 CASO CINEMANZANA 7](#_Toc498020917)

[4.2 ARQUITECTURA 7](#_Toc498020918)

[4.3 EXPLICACIÓN: 8](#_Toc498020919)

# INTRODUCCIÓN

***El mundo de la ingeniería e informática está en constante cambio y revolucionando bajo todo tipo de influencias para las tecnologías que usan las organizaciones para adaptarse en sus contextos.***

***En la actualidad, una de esas tecnologías es la de los Microservices implementados en las diversas organizaciones de diferentes rubros. Por ello, la finalidad de esta tarea es profundizar el conocimiento sobre esta tecnología y comprender por qué ha llegado a ser una de las tendencias actuales, bajo los conocimientos enseñados en clase y dos charlas sobre dicho tema de Lucas Krause y Matt Ranney; además de un caso práctico del tema.***

# REPORTE 1: “MICROSERVICES: THEORY AND APPLICATION” – LUCAS KRAUSE

## OBJETIVO

En base en su experiencia en consultoría TI, el expositor busca exponer el por qué en la actualidad la inversion, tiempo, uso de los microservicios está creciendo dentro de las organizaciones. Adicionalmente desea puntualizar como una persona puede formar parte de esta revolución del uso de microservicios.

## INTRODUCCIÓN A ARQUITECTURA

Krause puntualiza el paradigma de una construcción de un edificio con la arquitectura de software enfocandose en la perspectiva de la comunicación de cada elemento, recurso y componenete de la construcción como en el código para que todo funcione eficazmente.

Del lado de la Arquitectura de TI, Lucas añadió el concepto de patrones de software para la organizacíon del código, con el objeto de un rendimiento major para la organización y así comprobar su funcionalidad. No obstante, el expositor sugiere que el analisis de cada paradigm sea más profundo para poder generalizer la aplicación del caso a todo contexto de las organizaciones dependiendo de sus metas y siguiendo sus estándares, manteniéndolo lo más alto posible.

Definió 2 tipos de Arquitecturas:

* **Monolíticas:** Sistema extenso que realiza diversas funciones pero no se le puede dar mantenimiento. Resulta doloroso para la organización por falta de concordancia.
* **Orientada a microservicios:** Funciones divididas para conceptualizar todo el contexto, es amigable para migraciones y mantenimiento. Lo más importante, brinda mayor facilidad de comprensión del trabajo.

## BENEFICIOS

Lo esencial de los microservicios es que te permite encapsular partes del trabajo, de esta forma, se esta preparado el código a prueba de fallo porque te Brinda identificar o automatizar el lugar del fallo. Cabe resaltar que cada función o parte del código en un arquitectura orienta a microservicios es independiente y por ende es visualmente factible y sencillo su operatividad.

Otro gran punto importante, es la accesibilidad a ser escalable a la organización porque al tener todo separado por paquetes distintos de trabajo solo es una pieza más que se añade al código; es decir, a la arquitectura. Así como la seguridad e integridad del código e información de la organización.

Su mantenibilidad cada vez es más barata, gracias a la bajada de precios de las últimas décadas, y es más factible tener todo el trabajo en la nube, por el aumento de su capacidad para procesar todo los códigos, y generear mayor rentabilidad.

Su punto encontra según el expositor, es que la organización se enfrentaría a una arquitectura y código cada vez más complejo de manejar y de requerir especialistas para su mantenibilidad. También de requerir mayor memoria o servidores para operar.

# REPORTE 2: “WHAT I WISH I HAD KNOWN BEFORE SCALING UBER TO 1000 SERVICES” – MATT RANNEY

## ARQUITECTURA

Según el expositor, la arquitectura de Uber esta basada en microservicios. Por ello, ha podido crecer rápidamente en el Mercado en comparación a sus tardíos competidores y por ello se pudo posicionar a nivel alto. Uber utiliza lenguajes stadar como java , python , entre otros por ello, sus especialista para su mantenimiento es accessible.

No obstante, Matt resaltó que al tener Uber distintos lenguages se vuelve trabajoso solucionar un problema, repitiendo la lógica de solución para cada lenguaje que esta organización maneja. Ranney recalcó que ahí es donde se ve afectada su tiempo de actividad y responsabilidades en vez de mejorar para satisfacer a sus clientes.

## MANEJO DE FALLAS

El expositor dio a notar que Uber require una dedicada y amplio tiempo para las pruebas ante fallos para dar el visto Bueno de las verificaciones en todo el funcionamiento de la organización. Esto da a entender su nivel de criticidad de la organización y el riesgo en su rendimiento para el cálculo de los precios por zona o de los tiempos por rutas; es decir, todo un conflict en la comunicación de las métricas core de UBER.

## CULTURA ORGANIZACIONAL

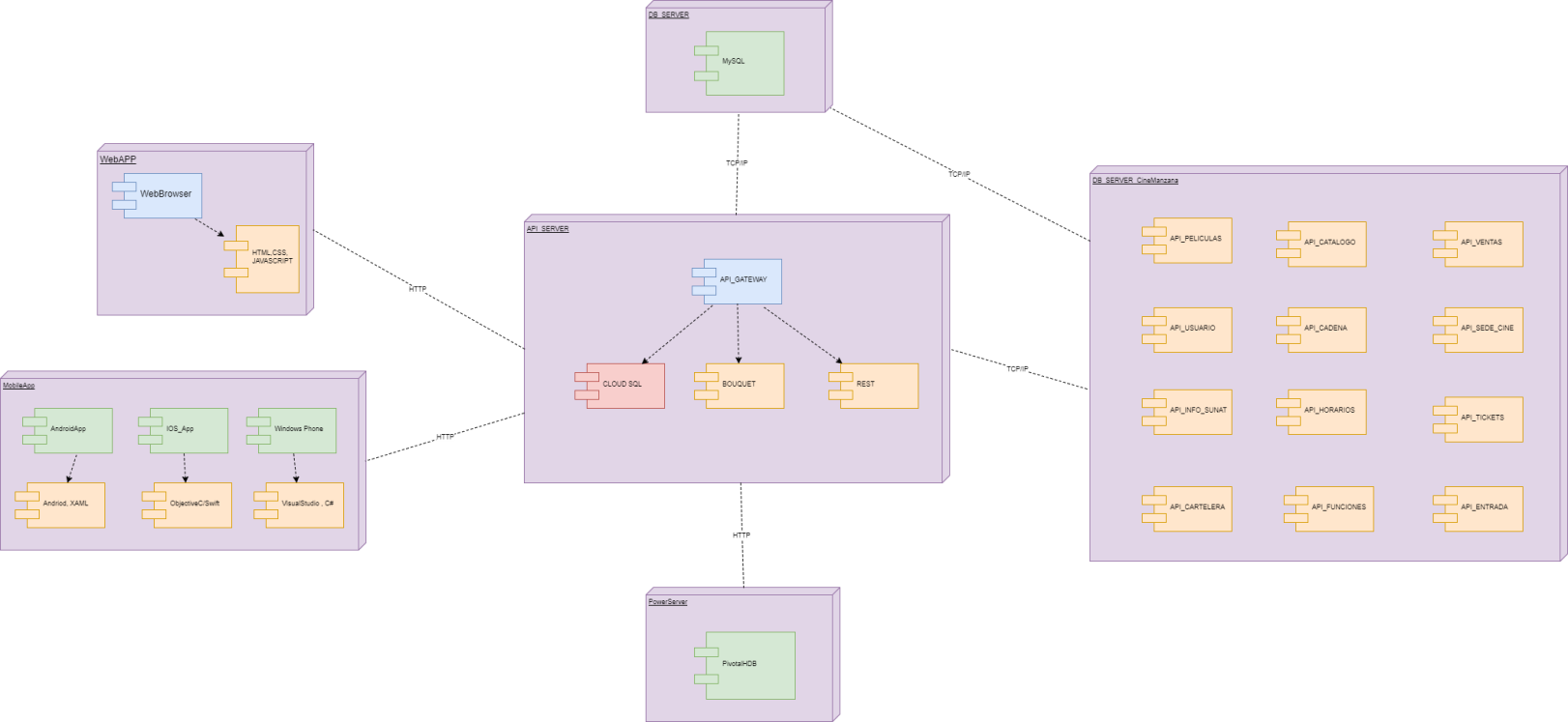
Matt Ranney, enfatizó un error inicial de UBER al no definer bien su lenguaje de trabajo para su produco Estrella; no obstante si afirma que la idea de negocio es excelente. Al generar cada Nuevo microservico con otrp lenguaje se require una alta contratación de personal para su mantenibilidad y esto conlleva a gastos elevados para UBER en vez de mejorar en la parte deTI.

# PREGUNTA 3

## CASO CINEMANZANA

* Adjunto en el repositorio bajo el nombre de ***“is2\_tarea3.pdf”.***

## ARQUITECTURA

* Adjunto en el repositorio bajo el nombre de ***“DiagramaArquitectura.png”***

## EXPLICACIÓN:

* WebApp:
  + WebBrowser: Plataforma Web (ingreso de Chrome, Mozulla, otros navegadores) de CineManzana.
  + Lenguaje: Java
  + Tecnologías: HTML, CSS,JAVASCRIPT
  + Conexion: Vía HTTP al servidor principal “API\_SERVER”
* MobileAPP:
  + AndroidApp: Plataforma móvil para smartphones Android
    - Lenguaje: Android,XML
  + IOS\_App: Plataforma móvil para smartphones IOS
    - Lenguaje:ObjectiveC/Swift
  + Windows Phone: Plataforma móvil para smartphones Windows
    - Lenguaje: VisualStudio, C#
* API\_SERVER:Motor principal para los microservicios y su funcionamiento
  + API\_GATEWAY:Gestor de entrada y salida de información de CineManzana
  + CloudSQL: Gestor del SQL en cloud para que la información sea ágil
  + Bouquet: Open source analytic (API\_BI) que soporta SQL.
  + REST: Build para el WebAPP.
* DB\_SERVER: Storage para la base de datos de la empresa (gestor)
  + Tecnología: MySQL
* PowerServer: Motor del API BI
  + Tecnología: PivotelHDB
* DB\_SERVER\_CINEMANZANA: Contiene todas las API’s requeridas para el negocio, tanto como consultas y respuestas para el funcionamiento y conexión de l ainformación.