**Sistemas Distribuidos**

**Proyecto**

Arquitectura de micro-servicios con caché para reducir latencia

**Profesora:**

* PhD. Abad Cristina

**Integrantes:**

* Basurto Cedeño Sergio Andrés
* Guerrero García Juan Joseph
* Pablo Elías Rugel Ayala

**Paralelo:**

* 1

**Descripción y justificación de decisiones de diseño del proyecto**

**Introducción**

El presente proyecto se basa en emplear los conocimientos adquiridos durante el transcurso de la materia en la cuestión de implementar un sistema web, con una arquitectura de micro-servicios y una caché para reducir latencia de acceso a la base de datos.

El sistema permitirá ver las noticias más populares de un periódico (top 10, en popularidad), las cuales estarán almacenadas en una base de datos.

**Objetivos Generales**

* Aplicar los conocimientos de la materia sistemas distribuidos para elaborar un sistema distribuidos.
* Gestionar e implementar un sistema web con un arquitectura de micro-servicios y una caché.

**Objetivos Específicos**

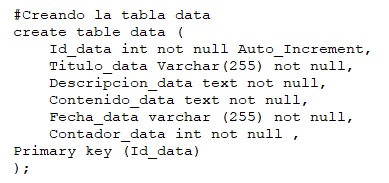
* Elaborar un sitio Web el cual muestre las noticias del día y las 10 noticias más vistos del día usando conocimientos de caché, bases de datos y servicios Web.
* Diseñar e implementar un sistema de gestión de bases de datos en MySQL que contenga los registros de las noticias del día.
* Emplear un sistema distribuido de propósito general para caché basado en memoria.

**Estructuras de datos usados**

Se estructuró una base de datos en MYSQL para receptar datos de un dataset de noticias,

El siguiente código es sumamente importante al momento de la organización de la base de datos con la cual trabajaremos.

Con este bloque creamos la tabla data en la cual consta de: Id\_data, Titulo\_data, Descripcion\_data, Contenido\_data, Fecha\_data, Contador\_data y la clave primaria Id\_data**.**

****

**Describiremos la base de datos:**

**Id\_data:** intel cual es nuestro identificador de cada noticia

**Titulo\_data:** varchar (255) el título de la noticia

**Descripcion\_data:**  text resumen del texto

**Contenido\_data:** text toda la noticia.

**Fecha\_data:** Varchar (255) fecha de la noticia.

**Contador\_data:** int donde almacenaremos valores emulando clicks.

**Librerías y Middlewares usados**

**Librerías para el uso Memcached**

**import sys:** El módulo provee acceso a funciones y objetos mantenidos por el intérprete.

**import MySQLdb:** Permite conectarnos a una base de datos con MySQL, una vez establecida la conexión, se crea un “cursor”, el cual es una estructura de control que se usa para recorrer y procesar los records de un result set.

**import memcache:** Ejecuta y particiona los datos en varios fragmentos, almacenamos en la caché objetos como, por ejemplo, nuestra base de datos.

**Apache thrift**: Framework que utilizamos por la diversidad de recursos que existían en internet para solucionar problemas del proyecto.

**Manejo de errores**

**Página web:** Se cargó una página con php, al generar el top 10 se carga otra página, disminuyendo el uso de recursos comparado a otras páginas donde toda la información la suelen cargar en una misma página.

**Base de datos**: La base de datos comparada con otras puede tener recursos desperdiciados en memoria dado los valores asignados a los varchar como por ejemplo el campo título.

**Caché:** La caché no se comparó pero tuvo un rendimiento aceptable al momento de cargar las noticias y almacenarlas.

**Lenguaje de programación usado.**

Se escogió Python y Java por la diversidad de documentación que ambos lenguajes poseen en internet, y que ambos trabajen bajo Apache thrift, dado que este tiene la versatilidad de trabajar con distintos lenguajes.

**Mecanismos de sincronización usados**

**Memcached**: Sistema distribuido de propósito general para caché basado en memoria, de suma importancia para cumplir los requerimientos de cache del proyecto.

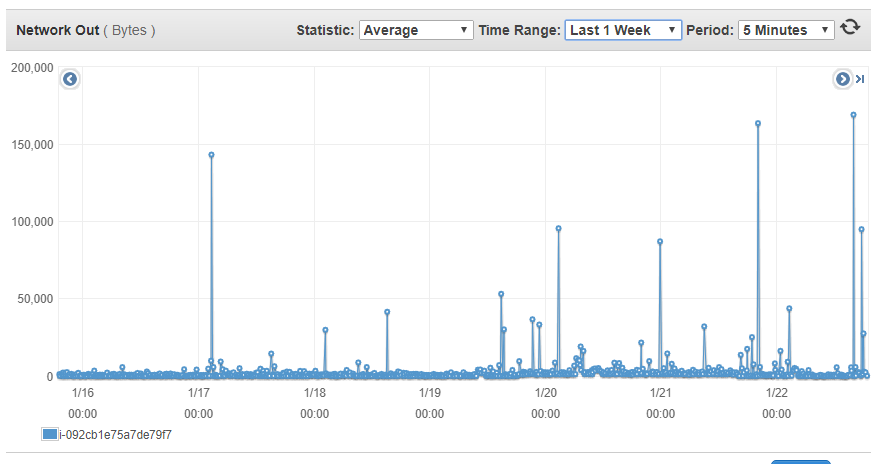
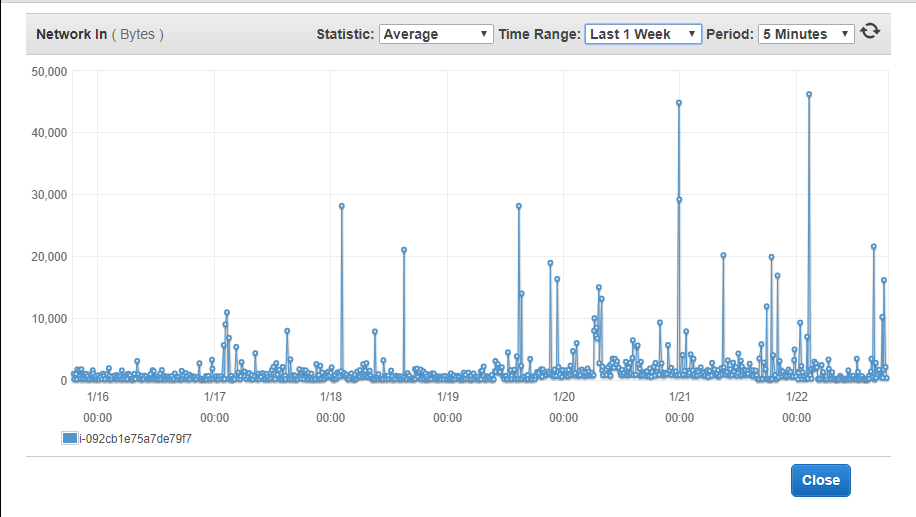
**Mysql**: Base de datos escogida para trabajar.

**Xampp**: Servidor web de plataforma, software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl, lo instalamos por la necesidad de cargar la página web.

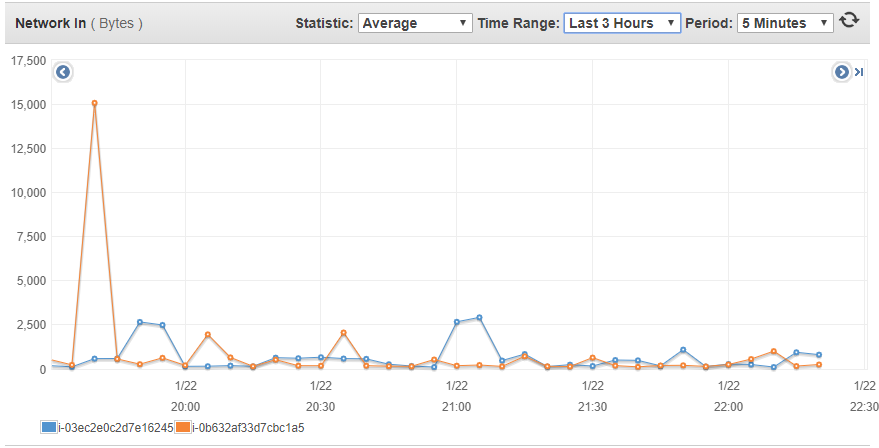
**Php**: Lenguaje necesario para los request desde HTML

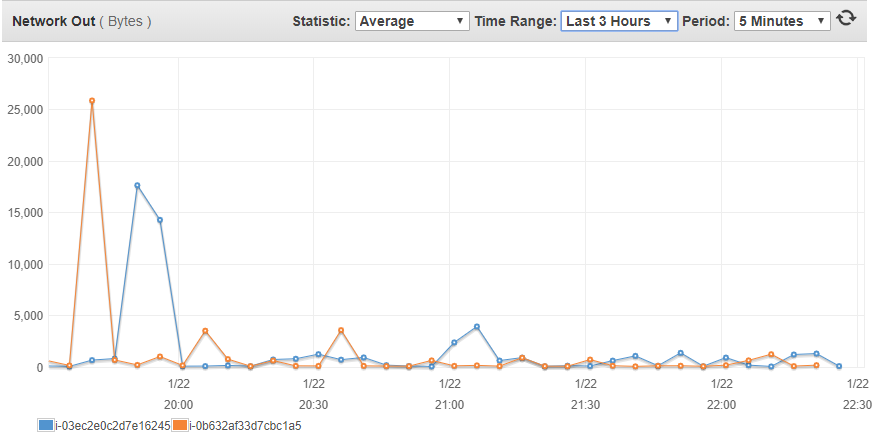
**Pruebas de rendimiento desde AWS**

**AWS DATABASE**



**AWS CACHE**





**Nube usada**

Se empleó AWS (Amazon Web Services) debido a que nos otorga ciertas ventajas como el levantamiento de servidores en la nube y demás, además por ser sugerencia de la documentación del proyecto, se crearon 3 máquinas virtuales las cuales:

1.- Servidor Web: Donde alojaremos nuestro sitio Web.

2.- Base de datos: Donde tenemos almacenados los datos de las noticias.

3.- Caché: Aquí estarán almacenados nuestros datos temporales de las noticias de cada día, todo estará levantado en Memcached.