

Llamado a MicroServicios con IFTTT

Barrionuevo Cesar, Bracero Alexis, León Jhordy, Sanchez Luis

Universidad de las Fuerzas Armadas - ESPE

Departamento de Eléctrica y Electrónica

Email: cbarrionuevo@espe.edu.ec, aabracer06@espe.edu.ec, jkleon@espe.edu.ec, lsanchez@espe.edu.ec

Abstract:

The following document details the realization of a system that makes a call to a microservice through IFTTT, which will automate the service of obtaining the name of a video on YouTube at the moment of giving a Like, besides creating a general registry, all in real time.

Keywords: IFTTT, MicroService.

Resumen: En el siguiente documento se detalla la realización de un sistema que realice un llamado a un microservicio mediante IFTTT, el cual automatizará el servicio de obtención del nombre de un video en YouTube al momento de dar un Like, además de crear un registro general, todo en tiempo real.

Palabras Claves: IFTTT, Microservicio.

I. INTRODUCCIÓN

En el mundo moderno cualquier actividad que no pueda ser realizada en forma eficiente mediante herramientas remotas representa una enorme desventaja, tanto competitivamente como de calidad de vida. Dentro de esta perspectiva, la automatización de los espacios y su dinamización con la incorporación de herramientas y tecnologías de la información, es más que capricho frívolo, una necesidad inminente. Tareas tan simples como controlar de forma verbal las luces de un departamento se convierten en la vertiginosa dinámica global en un problema, para aquellas instituciones y personas que no están preparadas.

Gracias al avance de la tecnología, las ciudades permiten mejorar su sostenimiento y las personas incrementar su calidad de vida. Debido a esa motivación se plantea una alternativa, el llamado de micro servicios mediante IFTTT.

II. ESTADO DEL ARTE

(Yubal,2019) menciona " Puedes pensar en ello como algo muy parecido a los atajos de iOS, las rutinas de Google Assistant o las rutinas de Alexa. Se trata de crear automatismos, de manera que cuando realizas una acción con alguna de tus aplicaciones, automáticamente se realizan las otras tareas que hayas asignado. Es una herramienta que puede cambiar por completo cómo utilizas algunos servicios y redes sociales.

Las siglas de IFTTT significan "If This, Then That", lo que traducido al español significaría "Si esto, entonces aquello". Con este nombre, se refiere a que es una página con la que puedes crear y programar acciones entre diferentes aplicaciones, de manera que si haces o pasa determinada cosa con una aplicación, entonces automáticamente realiza una acción con ella."

Como se ha mencionado en la introducción del trabajo de Yubal, el beneficio de IFTTT es crear acciones automáticas, y se ha implantado en el presente proyecto, pues cuando se realiza determinada acción en el sitio web de Youtube se llama a un microservicio el cual realizará la tarea para la que le ha sido programado.

Según (Suárez,2019) " IFTTT permite de una forma muy fácil e intuitiva pero a la vez extremadamente potente programar, reutilizar, redirigir, concentrar y archivar. IFTTT cuenta con casi 300 servicios activos que nos permiten combinar o sincronizar procesos automatizados entre sí. El funcionamiento del servicio es muy intuitivo, ya que la aplicación nos permite elegir un servicio en cuestión (Facebook, Twitter, WordPress, Pinterest, UBER, etc.) y ver cuáles son los servicios con los que podemos conectarlo. También podemos dejar que sea el propio IFTTT el que nos recomiende los applets más destacados o nos ofrezca servicios personalizados según nuestro perfil y nuestras preferencias." En este documento el autor muestra de una manera general la forma para usar IFTTT, en otras palabras primero se debe elegir un servicio que será conocido por la mayoría de las personas, estos pueden ser redes sociales, sitios web, entre otros, al compararlo con el este proyecto, se eligió un sitio web que se le denomina "YouTube". Posteriormente se elige el servicio al que se quiere conectarlo, en si IFTTT brinda algunas opciones pero para el presente proyecto se eligió un microservicio el cual recopilará el nombre del video cuando el usuario dé un Like (Me Gusta).

(Daniele,2015) menciona en su publicación " IFTTT permite que los usuarios de WordPress reduzcan el número de plugins instalados, aumenta el número de conexiones con servicios web de terceros, mejora la automatización y la eficiencia en las actividades de creación de blogs. Además IFTTT consta de applets, cada applet tiene un trigger y al menos una acción. Un trigger es la señal para que el applet realice una o más acciones. Tomemos, como ejemplo, el International Space Station notification applet: este envía una notificación a

cualquier dispositivo IOS o Android con la aplicación IFTTT instalada cada vez que el IIS pasa a través de la ubicación del usuario."

En el presente artículo el autor muestra varias de las ventajas de utilizar IFTTT, en donde la principal es la automatización y eficiencia, comparado con el presente trabajo al enlazar IFTTT con el micro servicio se crea una eficiencia para el usuario destinado, también como reducción de tiempo y así muchos beneficios más que dependerán del uso que se le de.

III. MARCO TEÓRICO

Microservicios

Los microservicios son un "tipo de arquitectura de software" y no un API o tecnología, la cual podemos instalar y utilizar, sin embargo, sí que existen frameworks que implementan esta arquitectura para facilitarnos la vida, como es el caso de Spring Boot en Java o Express para NodeJS. La idea en la arquitectura de microservicio se centra en separar la funcionalidad en pequeñas aplicaciones independientes que puedan operar con completa autonomía. (Blancarte,2015).

IFTTT

IFTTT es una plataforma que pone en contacto diferentes aplicaciones web o servicios de Internet. Lo hace de una manera muy peculiar, a través de la lógica. El significado de IFTTT es «If This Then That» lo que viene a decir «Que si pasa esto, haz lo otro».

IFTTT nos permite configurar un disparador o un evento que desencadenará una reacción. La finalidad de IFTTT es que cuando suceda algo, una acción, lance otro proceso, reacción. (del Valle Hernández, 2017).

IoT

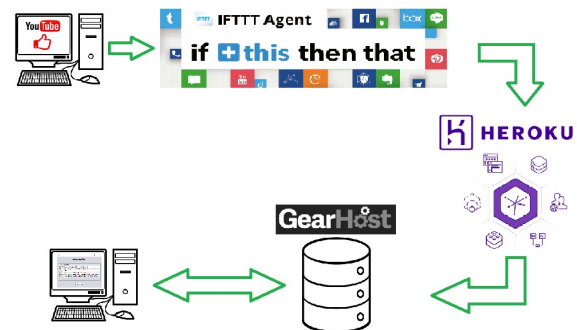
Internet de las Cosas es el concepto de objetos de todos los días, mediante el uso de sensores integrados para recopilar datos y seguir una acción con esos datos a través de una red. Dicho de manera simple, Internet de las Cosas es el futuro de la tecnología que puede hacer nuestras vidas más eficientes.

Campos Empleados del IoT

(SAS,2019) La IoT es más que una simple comodidad para los consumidores. Ofrece nuevas fuentes de datos y modelos de operación de negocios que pueden elevar la productividad en diversas industrias.

- Atención a la Salud
- Manufactura
- Telecomunicaciones
- Transporte
- Servicios Públicos

IV. DIAGRAMA ESQUEMÁTICO



V. EXPLICACIÓN DEL CÓDIGO FUENTE

- Primero se crea el API REST que va a ser subido a internet mediante Heroku el cual permitirá interactuar con IFTTT para realizar el llamado del micro servicio, el mismo consta de 3 clases:

Controlador

Aquí se crea un RequestMapping el mismo que obtiene el título del vídeo al cual el usuario dio "like" para después mandar este a una base de datos temporal correspondiente a IFTTT mediante un método POST en formato JSON.

```

1 package ec.edu.espe.rest;
2
3 import com.google.gson.JsonObject;
4 import com.google.gson.JsonParser;
5 import java.sql.Connection;
6 import java.sql.PreparedStatement;
7 import org.springframework.web.bind.annotation.RequestBody;
8 import org.springframework.web.bind.annotation.RequestMapping;
9 import org.springframework.web.bind.annotation.RequestMethod;
10 import org.springframework.web.bind.annotation.RestController;
11
12 @RestController
13 public class Controlador
14 {
15
16     @RequestMapping(value = "/insertarVideo", method = RequestMethod.POST)
17     public void insertarPelicula(@RequestBody String body)
18     {
19         JsonObject JSON = new JsonParser().parse(body).getAsJsonObject();
20         String titulo = JSON.get("titulo").getAsString();
21
22         String query = "INSERT INTO videos (titulo) VALUES (?);";
23         try
24         {
25             PreparedStatement preparedStatement = ConexionBD.getConexion().prepareStatement(query);
26             preparedStatement.setString(1, titulo);
27             preparedStatement.executeUpdate();
28         }
29         catch (Exception e)
30         {
31             e.printStackTrace();
32         }
33     }
34 }

```

ConexionBD

A continuación se tiene la clase ConexionBD en la cual se realiza la conexión con una base tipo MySQL en Gear Host la misma que almacenará los títulos de los vídeos que le gustan al usuario.

```

1 package ec.edu.espe.rest;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5
6 public class ConexionBD
7 {
8     private static final String URL = "jdbc:mysql://den1.mysql4.gear.host/ifttt";
9     private static final String USUARIO = "ifttt";
10    private static final String CONTRASEÑA = "Ik7IU_hKU3L~";
11    private static ConexionBD instancia;
12    private static Connection conexion;
13
14    private ConexionBD()
15    {
16        conexion = iniciarConexion();
17    }
18
19    private Connection iniciarConexion()
20    {
21        Connection conexionAux = null;
22        try
23        {
24            Class.forName("com.mysql.cj.jdbc.Driver").newInstance();
25            conexionAux = DriverManager.getConnection(URL, USUARIO, CONTRASEÑA);
26        }
27        catch (Exception e)
28        {
29            e.printStackTrace();
30        }
31        return conexionAux;
32    }
33
34    public static Connection getConexion()
35    {
36        if (instancia == null || conexion==null)
37        {
38            instancia = new ConexionBD();
39        }
40        return conexion;
41    }
42 }

```

RestApplication

Finalmente se tiene la clase RestApplication es la clase principal del proyecto la cual se encarga de ejecutar el API REST.

```

1 package ec.edu.espe.rest;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5 import org.springframework.context.annotation.ComponentScan;
6
7 @SpringBootApplication
8 public class RestApplication
9 {
10
11     public static void main(String[] args)
12     {
13         SpringApplication.run(RestApplication.class, args);
14     }
15
16 }

```

- Por ultimo se crea un cliente desktop el cual visualizara en una interfaz desarrollada en java una tabla con el listado de los videos que el usuario dio "like", este consta de 2 clases:

ConexionBD

En esta clase se realiza la conexión entre el cliente con la base de datos, con sus respectivas credenciales.

```

1 package ec.edu.espe.rest;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5
6 public class ConexionBD
7 {
8     private static final String URL = "jdbc:mysql://den1.mysql4.gear.host/ifttt";
9     private static final String USUARIO = "ifttt";
10    private static final String CONTRASEÑA = "Ik7IU_hKU3L~";
11    private static ConexionBD instancia;
12    private static Connection conexion;
13
14    private ConexionBD()
15    {
16        conexion = iniciarConexion();
17    }
18
19    private Connection iniciarConexion()
20    {
21        Connection conexionAux = null;
22        try
23        {
24            Class.forName("com.mysql.cj.jdbc.Driver").newInstance();
25            conexionAux = DriverManager.getConnection(URL, USUARIO, CONTRASEÑA);
26        }
27        catch (Exception e)
28        {
29            e.printStackTrace();
30        }
31        return conexionAux;
32    }
33
34    public static Connection getConexion()
35    {
36        if (instancia == null || conexion==null)
37        {
38            instancia = new ConexionBD();
39        }
40        return conexion;
41    }
42 }

```

Ventana

Finalmente tenemos la clase ventana en la cual se crea una interfaz y mediante el botón consultar se envia el String de la petición para la abstracción de los datos de la base de datos, los mismos que serán representados en una tabla creada mediante un ArrayList.

```

96 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
97 {
98     ArrayList<String> videos=new ArrayList<>();
99     String query = "SELECT videos.titulo FROM videos;";
100     try
101     {
102         PreparedStatement preparedStatement = ConexionBD.getConexion().prepareStatement(query);
103         ResultSet resultSet = preparedStatement.executeQuery();
104
105         while (resultSet.next())
106         {
107             videos.add(resultSet.getString("titulo"));
108         }
109         vaciarTabla();
110         llenarTabla(videos);
111     }
112     catch (Exception e)
113     {
114         e.printStackTrace();
115     }
116 }

```

VI. CONCLUSIONES

- Se puede concluir que la importancia del Internet de las Cosas, hoy en día, radica en la necesidad de tener control sobre casi cualquier actividad a la que pueda aplicarse conexión a internet, donde el usuario pueda gestionarlo, observarlo o que éste se integre e interactúe con otra 'cosa' para obtener datos relevantes o para poder automatizar algún

proceso de la vida cotidiana por medio de micro servicios que puedan facilitar tomar decisiones por parte del usuario. La ventaja de poder controlar algún dispositivo remotamente por medio de internet, ha hecho que el IoT crezca, no solo en diversos proyectos de investigación, sino también en producción de hardware y en la automatización de procesos.

- Luego de realizar la presente investigación se determinó que utilizar IFTTT es de gran ayuda para lograr automatizar acciones de la vida cotidiana haciéndola mas simple y sencilla para el usuario, esto debido a que es una app para desarrolladores la cual ofrece un sin numero de applets dirigidas a la automatización de varios servicio.
- Se puede concluir que el utilizar IFTTT al momento de realizar llamado a microservicios es una gran elección ya que tiene un ambiente amigable por el usuario y no se necesita ser un experto en programación para lograr automatizar algún proceso requerido.

VII. RECOMENDACIONES

- Hoy en día se recomienda usar aplicaciones que permitan automatizar tareas o controlar objetos a través del internet, debido a que sin tanta complejidad pueden ayudar al usuario a mejorar su estilo de vida de una manera cómoda segura y efectiva.
- Al momento de realizar el llamado a microservicios es recomendable utilizar IFTTT debido a que en la actualidad es la aplicación dominante de la automatización de servicios en Internet, cuenta con varias opciones para acomodarse al gusto en caso que no le guste alguno al usuario, puede utilizar otro y de esta manera sacarle el mayor provecho.
- En la presente investigación se realizó con éxito el llamado de un microservicio mediante el uso de IFTTT sin embargo existen varias aplicaciones para el llamado de microservicios y las cuales se recomienda analizar debido que cada una tiene su propio propósito de automatización, o en caso general tienes diferentes implementaciones dependiendo de lo que el usuario necesite realizar.

VIII. REFERENCIAS

- Suárez, C. J. S. (2019). Ifttt: Herramientas para mejorar tu productividad.IEEE,3(7)
- Yubal. (2019). Qué es ifttt y cómo lo puedes utilizar para crear automatismos en tusaplicaciones.Universitat Oberta de Catalunya.
- Blancarte, O. (2015). Software architect.IEEE.
- Daniele, C. (2015). Cómo usar recetas de wordpress ifttt para automatizar el flujo detrabajo.IEEE.
- del Valle Hernández, L. (2017). Cómo configurar ifttt con arduino y nodemcu.IEEE.
- SAS. (2019). Internet de las cosas (iot).IEEE.