



# MÉTODO DE LA INGENIERÍA

Nombre del sistema  
ASSIST-ME [PERFIL]

Equipo

Joe Hernández  
Christian Tamayo  
Cristian A. Cobo  
Kliver D. Girón

## **PROYECTO INTEGRADOR I**

Departamento TIC – Facultad de ingeniería  
Universidad ICESI



## **1. DESCRIPCIÓN DEL CONTEXTO**

Globant S.A. es una compañía de TI dedicada al desarrollo y mantención de Software. Actualmente, cuenta con un amplio portafolio de servicios, entre los que se encuentran: consultorías, automatización de procesos, Big Data, desarrollo de aplicaciones, entre muchos más.

Debido a la gran cantidad de servicios que prestan, la compañía se divide en diferentes unidades organizacionales llamadas “Studios”, los cuales cuentan con un correo por studio (ej: bigdata@globant.com , gaming@globant.com , cybersecurity@globant.com). Estos correos permiten que cualquier duda se comunique directamente con el Studio, que se supone tiene mayor conocimiento sobre cierto tema. Sin embargo, se ha observado que por este medio se hacen ciertas preguntas que podrían ser de interés general para muchos Globers y el conjunto de todas esas preguntas podrían ser un recurso más para el personal. Reenviar estos correos, archivar los que se consideren importantes, y buscar las personas que también puedan saber sobre el tema resulta bastante dispendioso para cada Studio. Por todo lo anterior, la compañía Globant tiene la iniciativa de desarrollar una aplicación web de preguntas y respuestas para el desarrollo interno de la compañía, que cuente con dos roles, uno de usuario y otro de administrador. Esta aplicación le permitirá a los Globers antiguos y a nuevos integrantes de la organización, tener acceso a una herramienta donde pueda retroalimentar y ser retroalimentados con las preguntas de los diferentes miembros de la compañía.

### **1.1. PROBLEMA**

Las preguntas realizadas a los estudios de Globant no son de fácil acceso para las personas interesadas y responder a todas ellas resulta ser una tarea dispendiosa para cada studio.

### **1.2. ESPECIFICACIÓN DE REQUERIMIENTOS**

1.2.1. Crear cuenta de usuario. Donde debe especificarse nombre de usuario, correo y contraseña.

1.2.2. Modificar el perfil de usuario. El perfil del usuario contiene:

cuántas preguntas ha realizado, cuántas preguntas ha contestado, cantidad de votos positivos que ha recibido, cantidad de votos de pregunta interesante que ha recibido, fotografía de perfil, una descripción, intereses o conocimientos, país y ciudad de residencia.



1.2.3. Visualizar perfil de Globber. El perfil de si mismo como el perfil de otro glober contiene la misma información del perfil del usuario. (Ver RF 1.2.2)

1.2.4. Visualizar notificaciones. Por medio de una ventana donde se muestra una notificación a un evento que está relacionado con el usuario, tal como respuestas a preguntas realizadas, cuando se marquen preguntas con insignias o respuestas del usuario hacia otras preguntas.

1.2.5. Iniciar y cerrar sesión. Por medio de su usuario y contraseña.

## **2. RECOPIACIÓN DE INFORMACIÓN**

Con el fin de abordar la situación problema de la mejor manera, se presenta un marco teórico que se considera relevante para el contexto de este. Así mismo, el estado del arte presentado el cual permite ver soluciones realizadas en contextos similares al problema.

### **2.1 Librerías de Google para iniciar sesión:**

En primer lugar, las librerías o Api de Google es un conjunto de API desarrollada por Google, las cuales permiten la comunicación e integración de los Servicios de Google con otros servicios. Ejemplos incluyen las API de Búsqueda, Gmail, Traductor o Maps. Las aplicaciones de terceros pueden usar esas API para extender la funcionalidad de sus servicios. Ahora bien, a veces, queremos que los usuarios inicien sesión utilizando sus credenciales existentes de aplicaciones de terceros, como Facebook, Twitter, Google, etc. Por lo tanto, Google ofrece una API de administración (autorización) que se descompone de la siguiente manera:

#### **2.1.1. Autorizar solicitudes**

La aplicación que deseamos desarrollar debería utilizar OAuth 2.0 para autorizar las solicitudes. No se admitiría ningún otro protocolo de autorización. Sin embargo, se usa el inicio de sesión de Google, algunos aspectos de la autorización se gestionan automáticamente.

#### **2.1.2. Autorizar solicitudes con OAuth 2.0**

Este es el punto clave, ya que, todas las solicitudes a la API de Analytics deben estar autorizadas por un usuario autenticado. Sin embargo, se debe tener en cuenta que los detalles del proceso de autorización para OAuth 2.0, o "flujo", varían ligeramente dependiendo del tipo de aplicación que se esté creando. El siguiente proceso general se aplica a todos los tipos de aplicación:

- 2.1.2.1. Cuando se crea la aplicación, se debe registrar con la consola de APIs de Google. A continuación, Google proporciona la información que necesitará posteriormente, como un ID de cliente y un secreto de cliente.

- 2.1.2.2. Activar la API de Analytics en la consola de APIs de Google. Si la API no aparece en la lista de la consola de APIs, se salta este paso.
- 2.1.2.3. Cuando la aplicación necesite acceder a los datos del usuario, esta solicitará a Google un determinado alcance de acceso.
- 2.1.2.4. Google muestra una pantalla de consentimiento al usuario pidiéndole su autorización para que la aplicación solicite algunos de sus datos.
- 2.1.2.5. Si el usuario aprueba esta petición, Google proporcionará a la aplicación un token de acceso de corta duración.
- 2.1.2.6 La aplicación solicita los datos del usuario y adjunta el token de acceso a la solicitud.
- 2.1.2.7. Si Google determina que tu solicitud y el token son válidos, muestra los datos solicitados.

En algunos flujos se incluyen pasos adicionales, como el uso de tokens de actualización para adquirir nuevos tokens de acceso. También se debe tener en cuenta que para solicitar acceso utilizando OAuth 2.0, la aplicación necesita la información del alcance, así como la información que proporciona Google durante el registro de la aplicación (como el ID y el secreto de cliente).

### **2.1.3. Flujos de OAuth 2.0 habituales**

A continuación, se enumeran los casos prácticos habituales de determinados flujos de OAuth 2.0:

- 2.1.3.1 Servidor web Este flujo resulta adecuado para el acceso automático, sin conexión o programado de los datos de Google Analytics de un usuario.  
Ejemplo: Actualizar automáticamente los paneles de usuario con los datos de Google Analytics más recientes.
- 2.1.3.2 Cliente Resulta ideal en el caso de que los usuarios interactúen directamente con la aplicación para acceder a sus datos de Google Analytics desde un navegador. Con este flujo no se necesitan funciones en el servidor, pero no se pueden elaborar informes automatizados, sin conexión o programados.  
Ejemplo: Una herramienta de creación de informes basada en navegador, como el Explorador de solicitudes de Google Analytics.
- 2.1.3.3 Aplicaciones instaladas: Este flujo es para aplicaciones que se distribuyen en paquetes y que instala el usuario. Es necesario que la aplicación o el usuario tenga acceso a un navegador para completar el flujo de autenticación.  
Ejemplos: - Un widget de ordenador en un PC o Mac. - Un complemento de un sistema de gestión de contenido. La ventaja de este flujo, en relación con el de servidor web



o de cliente, es que se puede usar un solo proyecto de la consola de APIs para la aplicación. De este modo, se pueden elaborar informes consolidados y se simplifica la instalación para los usuarios.

**2.1.3.4 Cuentas de servicio** Las cuentas de servicio son útiles para el acceso automatizado, sin conexión o programado a los datos de Google Analytics de tu propia cuenta. Por ejemplo, para crear un panel activo de tus propios datos de Google Analytics y compartirlo con otros usuarios.

**2.1.4. Solucionar problemas** No se otorgará la autorización en los casos siguientes:

**2.1.4.1** Se muestra el código de estado 401 si tu `access_token` ha caducado o si utilizas un alcance erróneo para la API.

**2.1.4.2** Se muestra el código de estado 403 si el usuario autorizado no tiene acceso a la vista (perfil). Asegúrate de estar autorizado con el usuario correcto y que este tenga la vista (perfil) que has seleccionado.

**2.1.5. Espacio de OAuth 2.0** Esta herramienta permite llevar a cabo todo el proceso de autorización mediante una interfaz web. En la herramienta también se muestran todos los encabezados de solicitud HTTP necesarios para realizar solicitud una autorizada. Si no se consigue que la autorización funcione en la aplicación, se debe probar que lo haga mediante el espacio de OAuth 2.0. Después, puedes comparar los encabezados HTTP y enviar una solicitud al espacio sobre lo que tu aplicación envía a Google Analytics. Esta comprobación es una forma sencilla de garantizar que se aplica el formato correcto a tus solicitudes.

**Concesión no válida:** Cuando se intenta utilizar un token de actualización, se obtiene un error `invalid_grant` en los casos siguientes: - El reloj de tu servidor no está sincronizado con el protocolo NTP.

- Se ha superado el límite de tokens de actualización. Las aplicaciones pueden solicitar varios tokens de actualización para acceder a una sola cuenta de Google Analytics. Por ejemplo, si un usuario quiere instalar una aplicación en varios ordenadores y acceder a la misma cuenta de Google Analytics, necesitará un token distinto para cada ordenador. Cuando el número de tokens de actualización supera el límite, los tokens anteriores dejan de ser válidos. Si la aplicación intenta utilizar un token de actualización invalidado, se devuelve la respuesta de error `invalid_grant`. El límite para cada par único de cliente OAuth 2.0 y cuenta de Google Analytics es de 25 tokens de actualización. Si la aplicación sigue solicitando tokens de actualización para el mismo par de cliente y cuenta, cuando se solicita el token número 26, el primer token de actualización que se emitió anteriormente deja de ser válido. El token de actualización número 27 que se solicite invalidará el segundo token emitido anteriormente, y así sucesivamente.

## **2.2 Base de datos**

Una base de datos es un “almacén” que nos permite guardar grandes cantidades de información de forma organizada para que luego podamos encontrar y utilizar fácilmente.

## **2.3 Microsoft SQL Server**

Es un sistema de gestión de base de datos relacional desarrollado por Microsoft. Como un servidor de base de datos, es un producto de software con la función principal de almacenar y recuperar datos según lo solicitado por otras aplicaciones de software que pueden funcionar ya sea en el mismo ordenador o en otro ordenador a través de una red.

## **2.4 Diseño responsive**

Diseño cuyo objetivo es adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visitarlas. Cada dispositivo tiene sus características concretas y esta tecnología pretende que con un único diseño web, todo se vea correctamente en cualquiera de estos.

## **2.5 Analítica de datos**

Es una ciencia que tiene como propósito tomar un conjunto de datos y sacar conclusiones sobre ellos. Normalmente se divide en tres ramas: el análisis exploratorio de datos, para descubrir nuevas herramientas de los datos; el análisis confirmatorio, para confirmar

teorías sobre un grupo de datos; y el análisis cualitativo donde se examinan datos de tipo no numérico para inferir conclusiones

## 2.6 Función autocompletar de Google:

A la hora de realizar una búsqueda en Google automáticamente -y en tiempo real- la plataforma sugiere opciones de palabras o frases de las cuales se puede tratar la búsqueda en cuestión. Esta funcionalidad de autocompletado se realiza con el fin de reducir el tiempo de teclado y se basa en las siguientes fuentes de predicción:

- Términos que se ingresan. Esto ya ocurre cuando se ha ingresado una palabra.
- Búsquedas relevantes realizadas anteriormente.
- Búsquedas de otras personas que se volvieron populares en el radar (Tendencias).

Las predicciones se realizan automáticamente por medio de un algoritmo que depende de los términos descritos anteriormente.

## 2.7 ¿Cuales son los algoritmos más usados para hallar textos similares?

Para textos cortos los más recomendados son Td-idf o BM-25 para pesar, y luego usar similaridad por coseno. Para textos largos se recomienda usar “Locality-Sensitive Hashing”. Tf-idf (Term frequency - inverse document frequency) Es una medida numérica que expresa la relevancia de una palabra en un documento. Este valor crece proporcionalmente a la cantidad de veces que se repite la palabra en el documento pero es compensada por la cantidad de veces que aparece la palabra en toda la colección de documentos, esto con el fin de manejar el hecho de que algunas palabras son más comunes que otras. Okapi BM25 Es una función que sirve para clasificar documentos semejantes de acuerdo a su relevancia en una consulta dada. Está basado en un modelo de trabajo probabilístico. BM25 está basado en el concepto de bolsa de palabras, con esto se representan los documentos que se desean ordenar en función de su relevancia con una consulta dada. Por lo que tenemos que, dada una consulta Q, que contiene las palabras claves  $q_1, q_2, \dots, q_n$  el valor de relevancia para un documento D será,

$$\text{score}(D, Q) = \sum_{i=1}^n \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot \left(1 - b + b \cdot \frac{|D|}{\text{avgdl}}\right)},$$

donde  $f(q_i, D)$  es la frecuencia con la que aparecen las palabras clave en el documento D.  $|D|$  es la longitud del documento en palabras y avgdl es la longitud promedio de la colección de documentos.  $b$  y  $k_1$  son parámetros que suelen depender de las características de la colección, aunque normalmente se le asignan valores  $k_1 = 1.25$  y  $b = 0.75$ .  $\text{IDF}(q_i)$  es el peso de las palabras clave que aparecen en la consulta y se calcula de la siguiente manera,



$$IDF(q_i) = \log \frac{N - n(q_i) + 0.5}{n(q_i) + 0.5}$$

Donde N es el número de documentos en la colección y  $n(q_i)$  es el número de documentos que contienen la palabra clave  $q_i$ .

Extraer palabras clave de un texto corto El producto de extraer palabras claves de un texto corto se puede lograr mediante el uso de machine learning. De acuerdo al autor del texto “Extracting Keywords From Short Text”, expuso tres opciones distintas como soluciones:

- a. Estilo Scrabble: se toma la oración y se calcula cada valor de cada palabra como en el juego de Scrabble y se retornan las mayor puntuadas como las más importantes. Luego se toman las palabras de la siguiente oración y se compara con las anteriores para buscar aquellas palabras relacionadas o repetidas. En este caso, las palabras con mayores puntajes tienden a ser las más importantes y las que más se repiten excepto por ciertos casos; para evitar esto se implementa una lista que contenga este tipo de palabras consideradas como excepciones y no contarlas en el ranking. Al finalizar, el ranking debe de tener una lista de puntajes en el cual las palabras más importantes estarían de primeras.
- b. Método RAKE (Rapid Automated Keyword Extraction): Este método consiste en eliminar las palabras más comunes en el idioma como conectores entre otros para al final dejar las palabras que se podrían considerar no comunes o las importantes del texto. En este caso no se garantiza perfectamente que sean las palabras más importantes del texto corto. Por último, RAKE se encuentra implementado en varios lenguajes de programación lo cual lo hace bastante simple y sencillo de usar.
- c. Inventar un nuevo Lenguaje: En este caso sería crear un lenguaje mediante el uso de palabras importantes como secuencia y hacer que la máquina aprenda a reconocerlas mediante una base de datos, tomándolas y creando un nuevo lenguaje basándose en ellas. Primero es necesario generar un set de datos, en el cual se pueda relacionar las palabras claves a algún tema y así empezar a categorizarlas. Luego, se cargan los datos del programa en la red neuronal con una estructura base. Para continuar, se ejecuta la red neuronal y se deja entrenando la secuencia de secuencias hasta que termine, al terminar la red neuronal ya debería de tener la capacidad de identificar algunas o varias palabras claves dentro del texto que se ingrese. Para mejores resultados es preferible que se entrene la red neuronal con una cantidad de datos mayor.
- d. De acuerdo a la página Elastic, en el artículo titulado “More Like this Query” o Consulta Mas Como Esto, consiste en encontrar documentos parecidos en base a un conjunto de documentos mediante la búsqueda de términos similares dentro de los documentos. Para lograr este proceso, se seleccionan una cantidad representativa de términos en el documento ingresado, y luego se genera una petición de búsqueda en la base de datos para encontrar textos que contengan términos similares. Para

que esta consulta funcione, primero se debe de clasificar el texto ingresado dentro de un tipo de consulta, en este caso se utiliza como método la fórmula de puntaje de Lucene (buscar similitudes dentro del texto, asignarles un puntaje y comparar donde pertenece). Luego, se toma el texto del documento ingresado y se analiza para poder extraer n cantidad de términos clave y utilizarlos en la consulta para la búsqueda de textos similares. Al finalizar el proceso, se retorna un conjunto de documentos similares al que se ingresó. API's para extraer palabras clave

## **2.8 Keywords Extraction API Documentation de Rapid Api**

Esta API permite tomar palabras clave de uno o varios URL o textos. Como parámetros de entrada pide un string de JSON especificando la entrada como tal de textos (el texto plano o el url), seguido del tipo de entrada y el número de palabras clave a encontrar. Cabe resaltar que para esta API la versión gratuita admite solo hasta 100 requests, y es necesario meter tarjeta.

## **3. ESTADO DEL ARTE**

Se han identificado dos plataformas web que tienen funcionalidades parecidas a las solicitadas, como lo es StackOverflow y Quora, cuya estructura será explicada y se tomará como base para realizar la interfaz gráfica.

### **3.1 StackOverflow :**

Stack Overflow es un sitio de preguntas y respuestas para programadores y entusiastas del desarrollo y uso de software. Hace parte de la red de sitios de preguntas y respuestas de Stack Exchange. (Tomado de <https://es.stackoverflow.com/tour> )

Imagen

En la parte izquierda cuenta con un panel que permite navegar por las diferentes vistas (inicio, preguntas, etiquetas, usuarios) y de esta manera moverse por la página. En el parte central cuenta con la zona de preguntas y respuestas, cada publicación de pregunta consta de 3 contadores: votos, respuestas y vistas, también posee un título y una zona de etiquetas que permite hacer referencia al tema de la pregunta. En la parte derecha posee un ranking que redirecciona a las preguntas más populares en la red. Si se accede a una pregunta aparecerá la siguiente interfaz:

Imagen

Cada pregunta consta con una descripción, una zona para agregar código, imágenes, o demás elementos que ayuden a comprender la pregunta, una zona de etiquetas y una zona para agregar las diferentes respuestas a la pregunta. En la parte derecha se muestra un listado de preguntas relacionadas a la misma.

### **3.2 Quora**

es una red social creada para la distribución masiva de conocimientos, en donde cualquier persona puede realizar una pregunta y cualquiera puede dar una respuesta. (Tomado de <https://es.quora.com/Qu%C3%A9-es-Quora>). La página de Quora luce de esta manera:

imagen

La página posee en su parte izquierda un campo con los canales de interés de cada usuario, esto permite navegar y consultar las preguntas de interés. En su parte media muestra las preguntas más relevantes del canal que seleccionaste, permitiendo ver las respuestas y agregar comentarios a los mismos. En la parte superior posee botones para las acciones principales: ir a inicio, responder, añadir una pregunta y buscar.

## **4. BUSQUEDA DE SOLUCIONES CREATIVAS**

En el módulo de perfil de usuario, identificamos todas las necesidades que se deben tener en cuenta acerca de la administración de la sesión de un usuario así como la información que esté puede ver de sí mismo y de otros acorde a las funcionalidades de la aplicación web. Para poder realizar todo lo anteriormente mencionado, se propuso varias ideas de cómo se debe implementar los servicios. Fueron propuestos por lluvia de ideas y se agrupo de manera más compacta de acuerdo a los requerimientos funcionales:

### **4.1. Iniciar sesión en la aplicación web. Mediante la cual un usuario, anteriormente registrado, podrá interactuar en la aplicación web.**

- a. Inicio de sesión mediante Google. Mediante un API ofrecido por los servicios de Google, se enlaza esta a la aplicación para que pueda iniciar sesión un usuario anteriormente registrado y vinculado su cuenta de Google.

- b. Inicio de sesión mediante una base de datos de la empresa Globant. Mediante una base de datos de registro de los usuarios de la compañía, los usuarios podrán iniciar sesión introduciendo sus datos anteriormente registrados en Globant
- c. Inicio de sesión por medio de un perfil de facebook. Esta puede ser una segunda opción o alternativa de respaldo para que un usuario de Globant entre a la plataforma. Por lo tanto, la aplicación puede tener varios métodos de inicio de sesión.
- d. Inicio de sesión mediante la huella dactilar registrada del usuario en la base de datos de la compañía o aplicación web.
- e. Inicio de sesión mediante la identificación del rostro del usuario registrado en la base de datos de la compañía o aplicación web.

#### **4.2. Visualizar perfil de usuario. Mediante la cual un usuario podrá visualizar toda la información necesaria de sí mismo u otro usuario.**

- a. Implementar página de perfil del usuario como las de facebook donde toda la información básica de un usuario se muestre en una página.
- b. Implementar página de perfil como la de linkedin donde se muestran los datos necesarios del usuario. Aparte, en otras páginas, se muestran las preguntas realizadas y las respuestas dadas.
- c. Implementar página de perfil como la de StackOverflow donde en la parte superior de la página están ubicadas dos tabs, la primera tab está relacionada con la información del perfil del usuario mientras que la segunda tab muestra las preguntas realizadas como las respuestas realizadas.

#### **4.3 Notificar acciones de la aplicación web. Estas informan a los usuarios que han planteado preguntas en la plataforma y otros han respondido a ellas.**

- a. Notificar a los usuarios cuando sus preguntas son respondidas mediante su correo o email empresarial.
- b. Notificaciones flotantes cuando un glober responde las preguntas de otro. Estas notificaciones aparecerán en el escritorio durante unos 10 segundos.
- c. Notificaciones en la misma aplicación, es decir, tener un apartado de interactividad donde las notificaciones puedan verse de forma cronológica. Y cuando una o varias de ellas son revisadas eliminarlas del apartado.
- d. Notificaciones mediante el correo empresarial y dentro de la misma plataforma. Al notificarse por ambas partes podemos tener un historial de interactividad, es decir,



cuando un usuario revisa las notificaciones desde la aplicación éstas ya no vuelven a aparecer en ella, sin embargo se pueden mirar cuantas veces quiera desde el correo empresarial.

#### **4.4 Transición de ideas a diseños preliminares**

### **5. EVALUACIÓN Y SELECCIÓN DE SOLUCIONES**

**5.1.** Criterios de evaluación para soluciones al registro e inicio de sesión de usuario en la plataforma.

#### **5.1.1 Criterio A. Facilidad de implementación**

- [3] Alta
- [2] Media
- [1] Baja

#### **5.1.2 Criterio B. Bajo costo de implementación**

- [3] Bajo
- [2] Medio
- [1] Alto

#### **5.1.3 Criterio C. Acceso**

- [3] Rapido y trivial
- [2] Rapido
- [1] Demorado

#### **5.1.4 Criterio D. Disponibilidad de cuentas**

- [3] Alta
- [2] Media
- [1] Baja

**5.2** Criterios de evaluación para soluciones al perfil de usuario

#### **5.2.1 Criterio A. Sencillez y minimalismo**

- [2] Sencillo y atractivo
- [1] Sencillo

#### **5.2.2 Criterio B. Enfoque del ambito**

- [2] Conocimiento compartido
- [1] Red Social

### 5.2.3 Criterio C. Similitud de acuerdo con la interfaz necesitada

- [3] Alta
- [2] Media
- [1] Baja

**5.3** Criterios para la evaluación de soluciones de notificaciones para usuarios de la plataforma AssistMe.

#### 5.3.1 Criterio A. Visibilidad

- [2] Sencillo y atractivo
- [1] Sencillo

#### 5.3.2 Criterio B. Accebilidad

- [3] Fácil
- [2] Medio
- [1] Difícil,

### Evaluación

#### LOGIN

Criterio	Criterio A	Criterio B	Criterio C	Criterio D	Total
Facebook	3	3	3	2	11
E-mail Globant	3	3	3	3	12
Touch -ID	1	1	3	1	6
Face -ID	1	1	3	1	6
Google	2	3	3	3	11

De acuerdo con el punto de “LOGIN” las mejores opciones como solución al registro e inicio de sesión de usuario en la plataforma son por medio del E-mail Globant y el inicio de sesión a través de google, ya que, permiten una alta facilidad al momento de implementarla, bajo costo de implementación porque todo depende de la codificación, rápido acceso y de manera muy trivial (facilidad de uso) y la cantidad de cuentas que puede tener un usuario necesita un E-mail y la gran mayoría tienen estos correos vinculados con Google.

#### PERFIL

Criterio	Criterio A	Criterio B	Criterio C	Total
StackOverflow	2	2	3	7
Facebook	2	1	2	5
Linkedin	2	1	2	5

Para el perfil de usuario la mejor opción para su solución es teniendo en cuenta la interfaz de usuario y los servicios que presta la reconocida página StackOverFlow, ya que, su sencille y fácil interacción permite al usuario comprender mucho más fácil el uso de la plataforma. Además de que permite tener un conocimiento compartido junto a otros usuarios. Por otro lado, los demás criterios criterios (Facebook y linkedin) presentan algunas funciones y servicios interesantes. Sin embargo, no están lo suficientemente relacionados con el contexto de este proyecto.

#### NOTIFICACIONES

Criterio	Criterio A	Criterio B	Total
Notificaciones flotantes	1	3	4
Correo	2	2	4
Plataforma	2	3	5

Teniendo en cuenta las notificaciones que pueden tener los usuarios la mejor opción es hacerlo por medio de la plataforma, ya que, es mucho más sencillo -teniendo en cuenta la comodidad del usuario para ver sus notificaciones- y atractivo y mucho más fácil para usar a comparación de las otras dos opciones.