

# **Estandar HTTP**

**Macedo Borbolla Eduardo Sai <sup>1</sup>Grupo: 3CM5**

<sup>1</sup>Escuela Superior de Cómputo <sup>1</sup> Web Application Development

{mordrirsai}@gmail.com

Este escrito tiene como objetivo describir de manera general la funcionalidad y utilidad del protocolo HTTP.

## **1. Estandar HTTP**

El estandar HTTP define un protocolo de comunicaciones de transferencia de hipertexto basado en peticiones y respuestas, el cual es stateless o sin estado haciendo esto referencia a que a pesar de gestionar mediante los formatos de peticion/respuesta bajo una arquitectura de cliente-servidor el protocolo no tiene algún tipo de memoria que permita/ le importe "recordar" o guardar en una memoria inmediata las acciones recientemente realizadas.

Dentro de las distintas necesidades que se encontraron entre versiones de este protocolo, se descubrió que las funcionalidades básicas podrían extenderse para un marco de referencia de comunicaciones más complejo que el que se pensó originalmente.

Desde el protocolo original se define la correcta sintaxis y definición semántica con la que debe estar estructurada y apegada la comunicación de información dentro del hipertexto a ser transferido, de tal manera que se conforman 9 tipos de peticiones a recursos usualmente explotadas por los programadores para el manejo de negociación de contenido así como su representación de acuerdo a su contexto por parte de un cliente o más generalmente un user-agent hacia un servidor al lanzar la primera petición , ya sea para acceder a sus recursos, modificarlo, eliminarlos, etc.. llamadas transacciones que permiten el intercambio de información entre los distintos tipos de roles que toman los dispositivos (usuarios, clientes) pudiendo también incluso controlar el flujo de mensajes discriminando la cantidad de tiempo de tiempo que lleva en la red o los saltos que ha dado para llegar a su destino pudiendo gestionar de manera ordenada, transparente y concisa una buena comunicación al seguir una buena implementación..

Mediante la necesidad de generar transacciones más complejas se generó la necesidad de mantener un estado o controlar las conexiones provenientes de un user-agent de tal manera que los cuerpos de los mensajes, encabezados, mensajes de estado, cookies, así como otras configuraciones de comunicación establecidas un poco más elaboradas, han aportado una gran flexibilidad en las funcionalidades ofrecidas para las siguientes versiones del protocolo que nos proveen herramientas cotidianas hoy en día como:

1. Contenido cacheable: Brinda facilidad y mejora de procesamiento de información anteriormente manejada en una transacción
2. Proxies: Entidades que permiten anonimato en la red y control de transparencia de envío de datos
3. Control de permisos sobre retransmisiones.

## **2. Conclusiones**

El protocolo HTTP de ser bien implementado se compromete a brindar una plataforma para la transferencia de datos muy sencilla, poderosa y robusta con la cual las comunicaciones pueden usarse eficientemente tanto en su uso común como con una buena configuración y explotación de características ventajosas según el contexto de nuestra aplicación particular sin salir del marco de referencia estandar.