

BitTorrent

Profesor encargado:

Lic. Fernando Martínez González
frndmartinezglz@gmail.com

16 de octubre de 2018

FTP normalmente resulta una alternativa fácil a la hora de compartir y descargar archivos. Cuando un archivo se elimina del FTP, para recuperar espacio, se pierde la oportunidad de obtenerlo. En este caso el problema está en que el servicio se encuentra *centralizado*. FTP también es un estándar de los más antiguos de internet en capa de aplicación. Existen alternativas *descentralizadas* que garantizan que mientras alguna parte del sistema tenga el archivo este va a estar disponible para los demás clientes de la red. Esto se podría realizar utilizando el almacenamiento de los propios clientes como almacenamiento del sistema.

Índice

1. Especificaciones	1
1.1. Cliente	1
1.2. Tracker	1
2. Recomendaciones	2
3. Evaluación	2

1. Especificaciones

Para la confección de este sistema se requiere de implementar dos elementos, posteriormente descritos. Cualquier modificación o aclaración ver al profesor encargado de la orientación.

1.1. Cliente

Un cliente es un nodo de la red que *sirve* y *descarga* archivos. Un archivo en descarga o descargado debe estar disponible para el resto de los clientes. El almacenamiento solo puede estar del lado de los clientes, no se puede utilizar un servicio externo para esta tarea.

Se debe proveer una aplicación gráfica para una mejor interacción con el sistema.

1.2. Tracker

Un *tracker* es el elemento en el sistema que mantiene actualizado qué clientes poseen qué archivo. La información se publica y descarga a través de *archivos torrent*. Esto quiere decir que si un cliente quiere servir determinado archivo (o grupo de archivos) debe crear un archivo torrent que así lo declare y publicarlo en el tracker. Este elemento no puede representar un punto de falla única en su sistema.

2. Recomendaciones

Existen numerosas implementaciones de BitTorrent en varios lenguajes de programación. Sin ánimo de potenciar el fraude académico, sino las buenas prácticas, usted podría nutrirse de las ideas que se siguieron en estos proyectos.

Sobre los aspectos que siempre se tendrán en cuenta en la revisión son:

- tolerancia a fallas
- replicación.

El soporte de estas debe ser, preferentemente, parametrizado, para que el servicio ofrecido pueda escalar sin dificultad.

El trabajo a realizar es abundante, por este motivo es de vital importancia las buenas prácticas de programación. Conocimientos de virtualización, testing y patrones de diseño, pueden ayudarlo a franquear este desafío. Trazar metas pequeñas que progresivamente estructuren todo el proyecto, para esto pueden utilizar un gestor de tareas (Trello, Issues de Gitlab, etc).

3. Evaluación

Todos los proyectos poseen un alto contenido ingenieril. Si durante la realización del proyecto, el estudiante utiliza alguna herramienta de evaluación automática (*testing*), se incentiva a que lo comente y lo utilice durante la revisión.

La calificación final del proyecto es resultado de su trabajo y del desenvolvimiento del equipo durante la exposición. Presentar un informe escrito, si bien no es obligatorio, será bien considerado en el momento de evaluar.