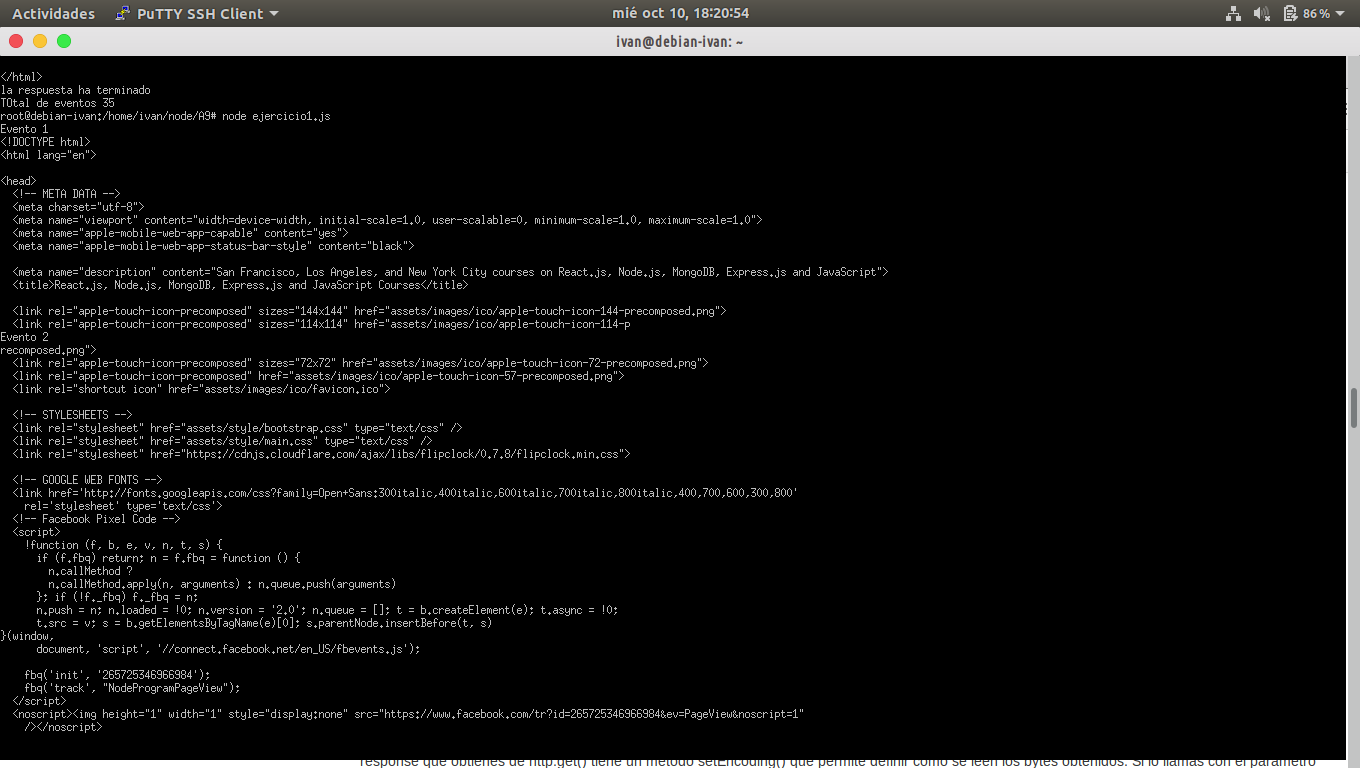
## A9. Cliente HTTP

Ejercicio 1

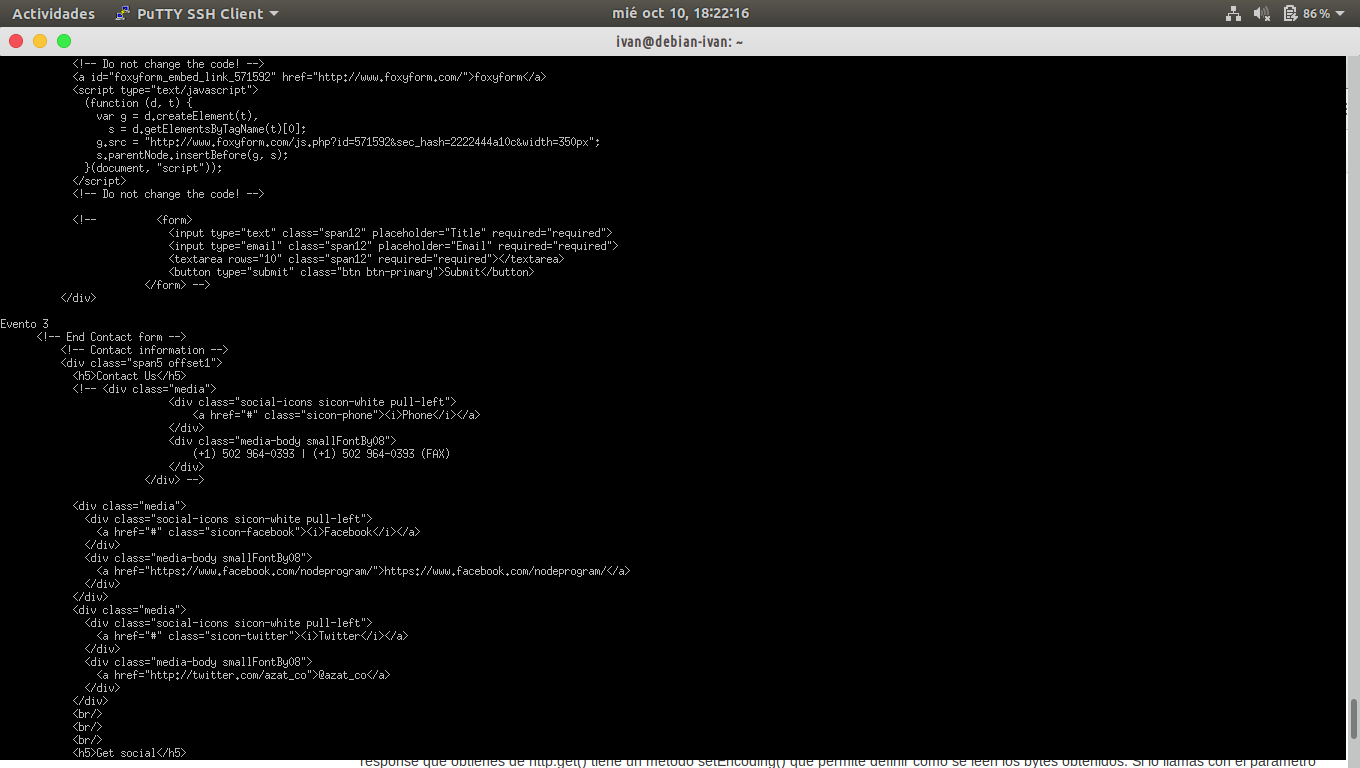
Escribe un programa que reciba como argumento una URL y realice una petición HTTP GET a la misma. Luego, deberá imprimir por consola el contenido de cada evento "data" de la petición. Antes de imprimir cada chunk debe incluir el texto "Evento X", donde X es el número de evento.



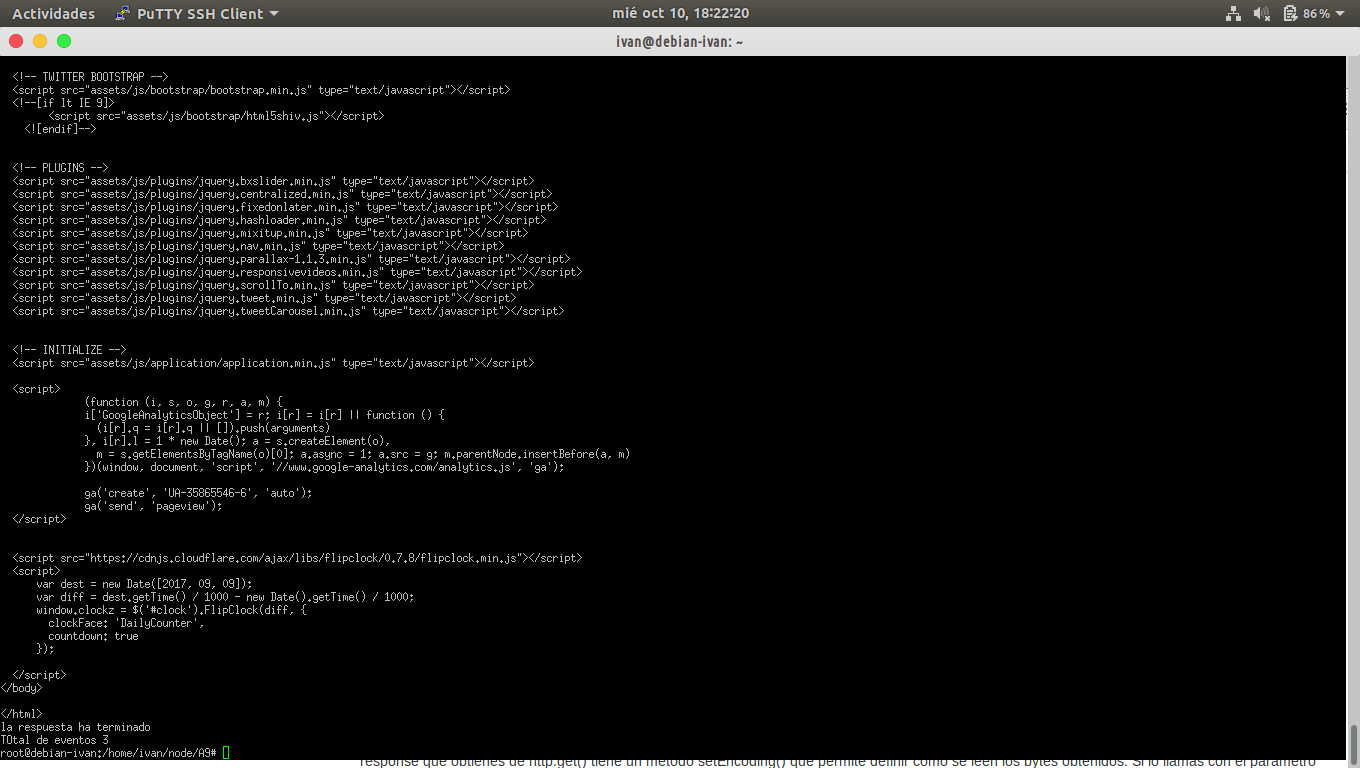
Observamos como nos devuelve el número de eventos.



Evento 3



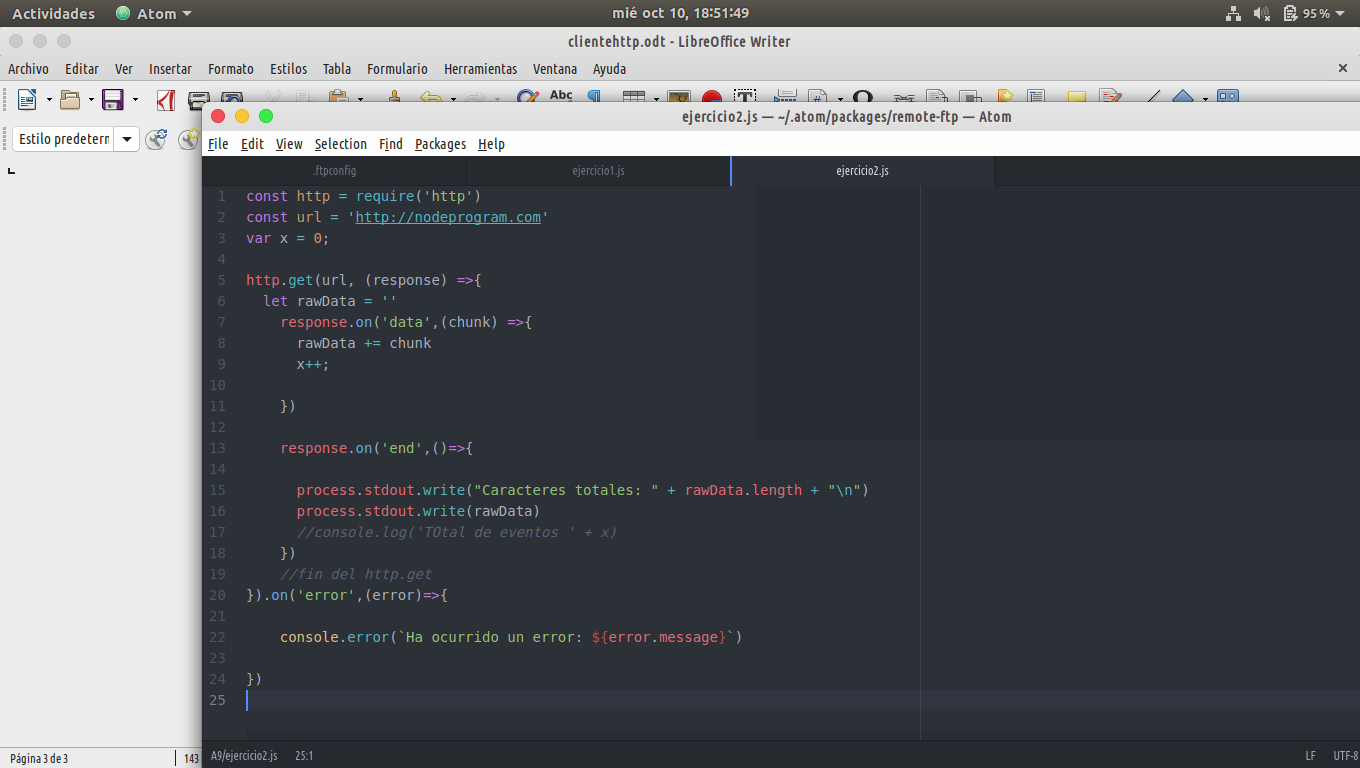
Total de eventos

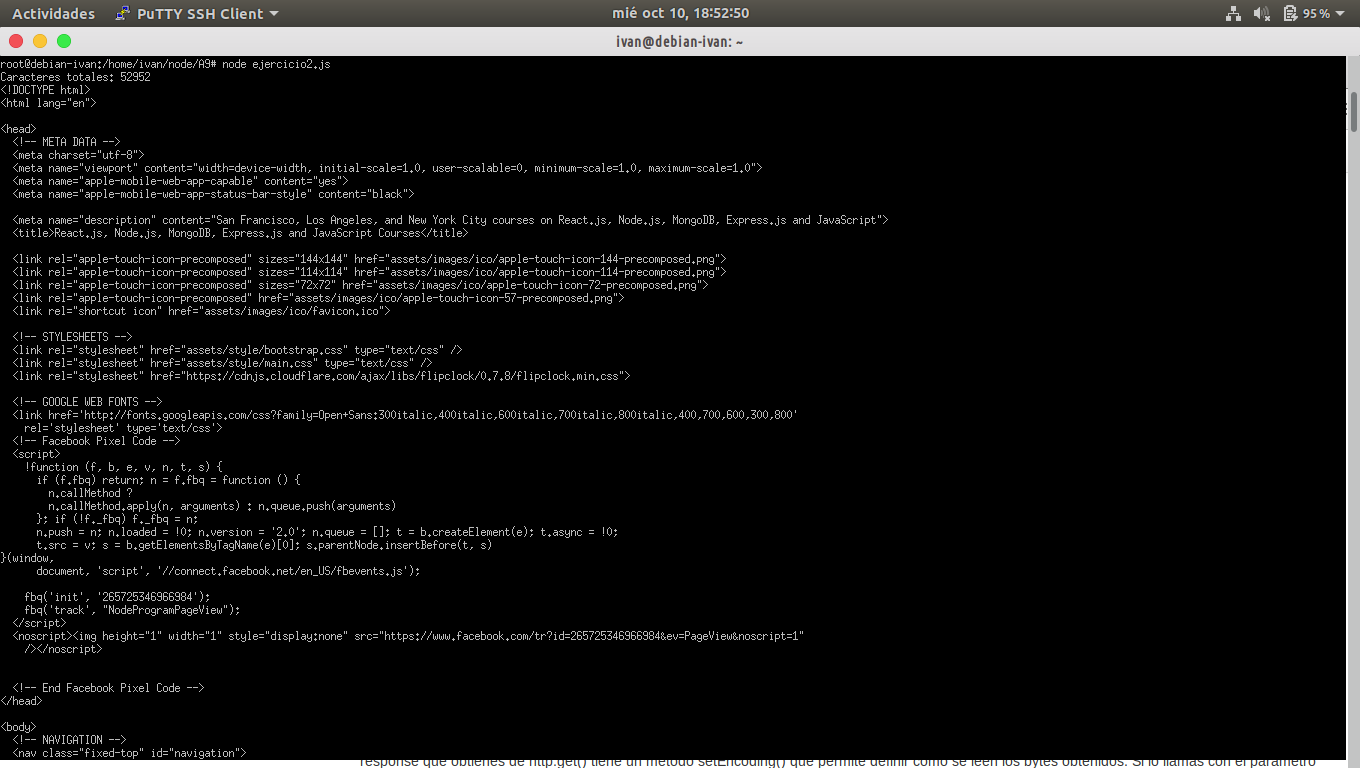


Ejercicio 2

Escribe un programa que realice una solicitud HTTP GET a una URL proporcionada como el primer argumento de línea de comando. Recoge todos los datos del servidor y luego escribe dos líneas en la consola (utiliza stdout y no console.log):

* La primera línea debe ser un número entero que muestre el número de caracteres recibidos del servidor.
* La segunda línea debe contener la cadena completa de caracteres enviados por el servidor.





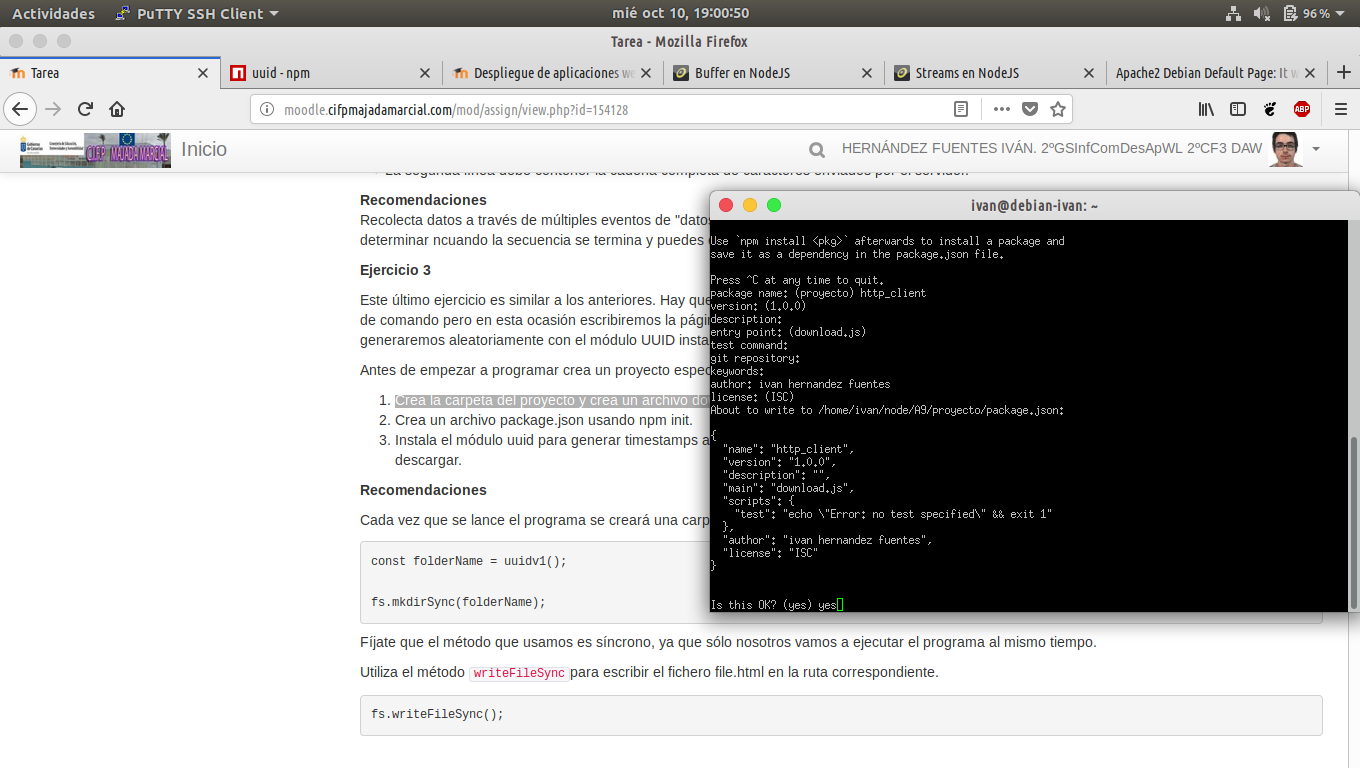
Ejercicio 3

Este último ejercicio es similar a los anteriores. Hay que escribir un programa que realice una solicitud HTTP GET a una URL proporcionada por línea de comando pero en esta ocasión escribiremos la página obtenida en un fichero que se llamará file.html, dentro de una carpeta cuyo nombre generaremos aleatoriamente con el módulo UUID instalable mediante npm.

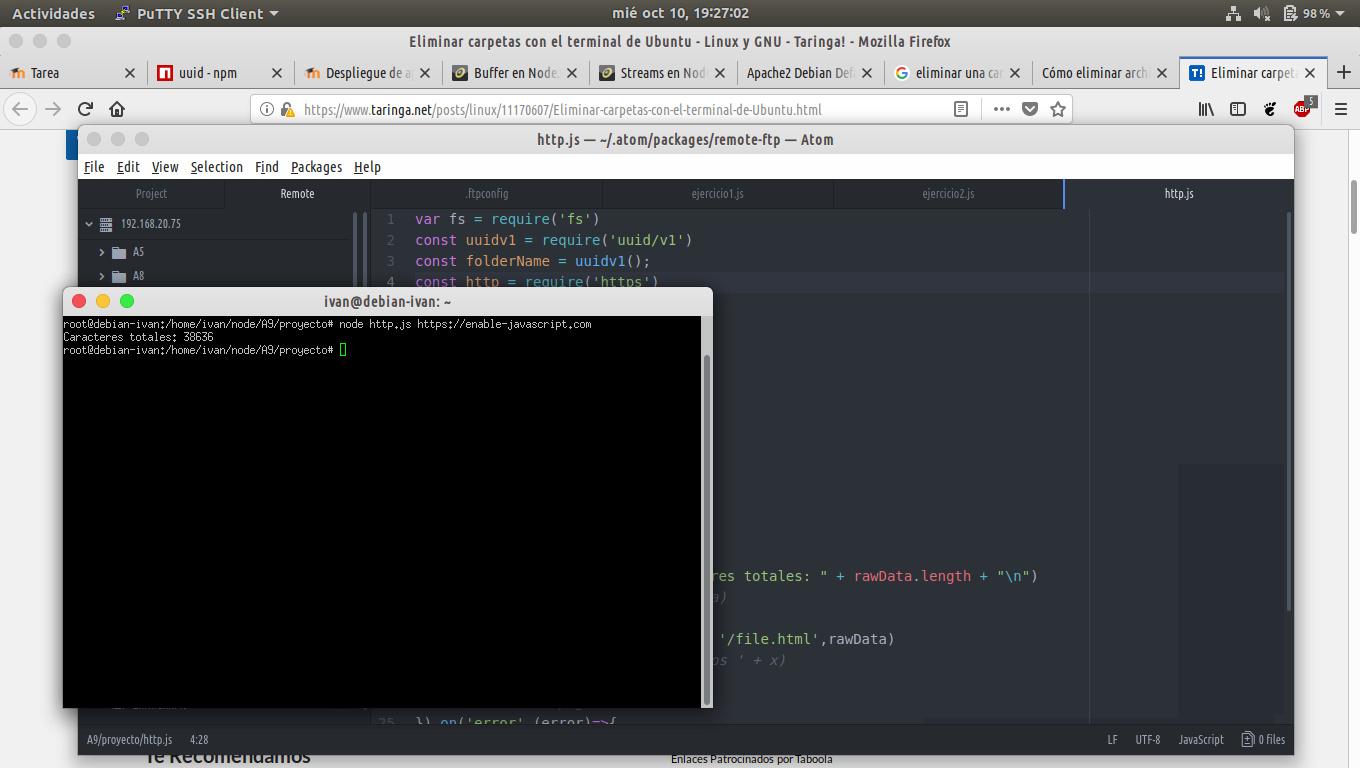
Antes de empezar a programar crea un proyecto específico para este ejercicio siguiendo los siguientes pasos:

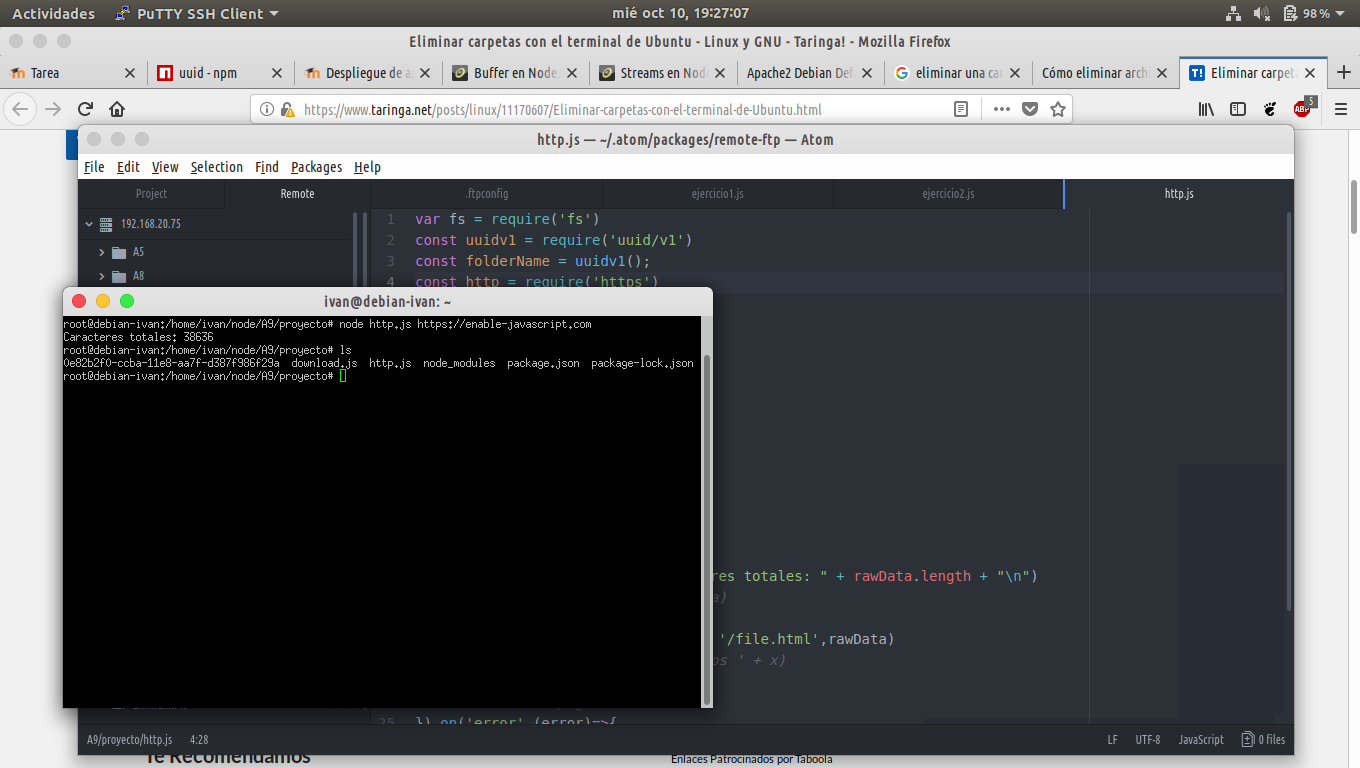
1. Crea la carpeta del proyecto y crea un archivo download.js en ella.
2. Crea un archivo package.json usando npm init.
3. Instala el módulo uuid para generar timestamps aleatorios que usaremos como nombres de carpeta distintos donde almacenaremos la página a descargar.

**Creamos nuestro package.json**

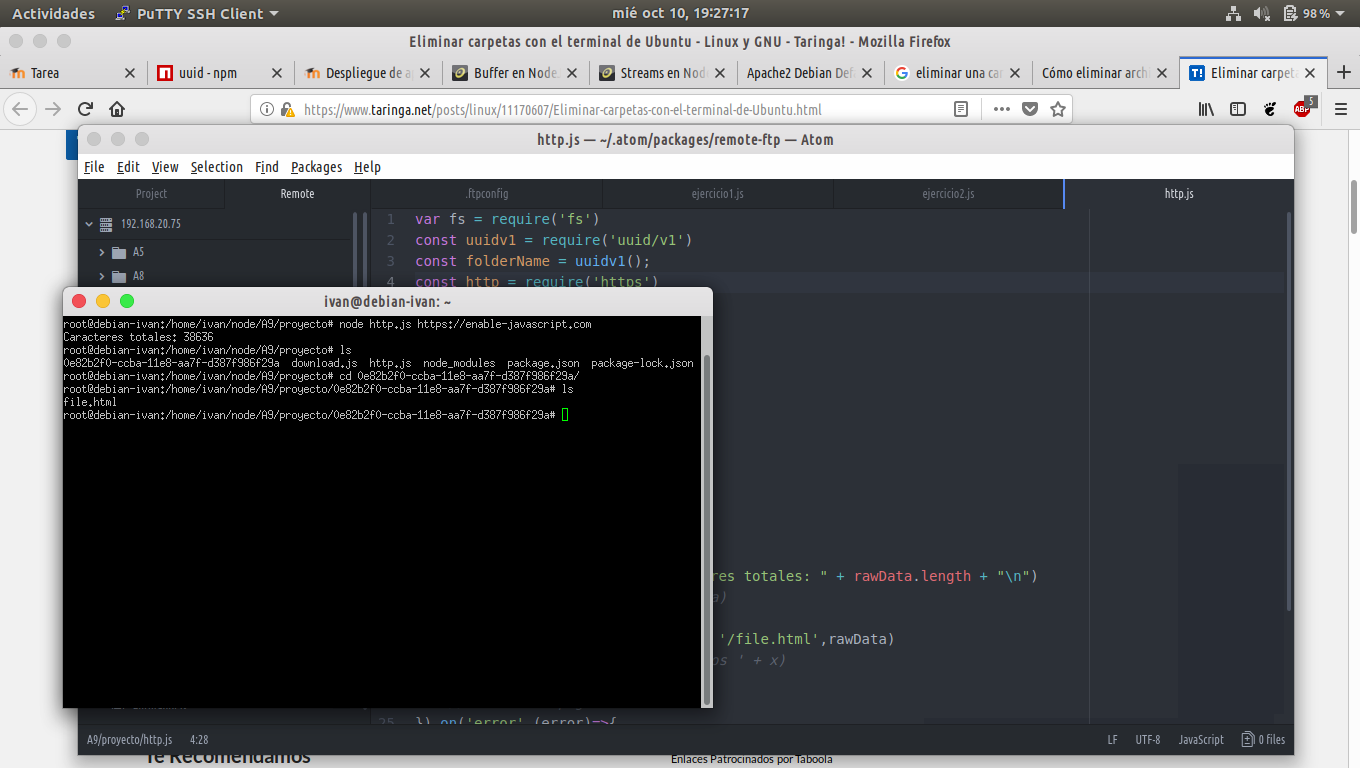


**Ejecutamos el script y agregamos el parámetro url**

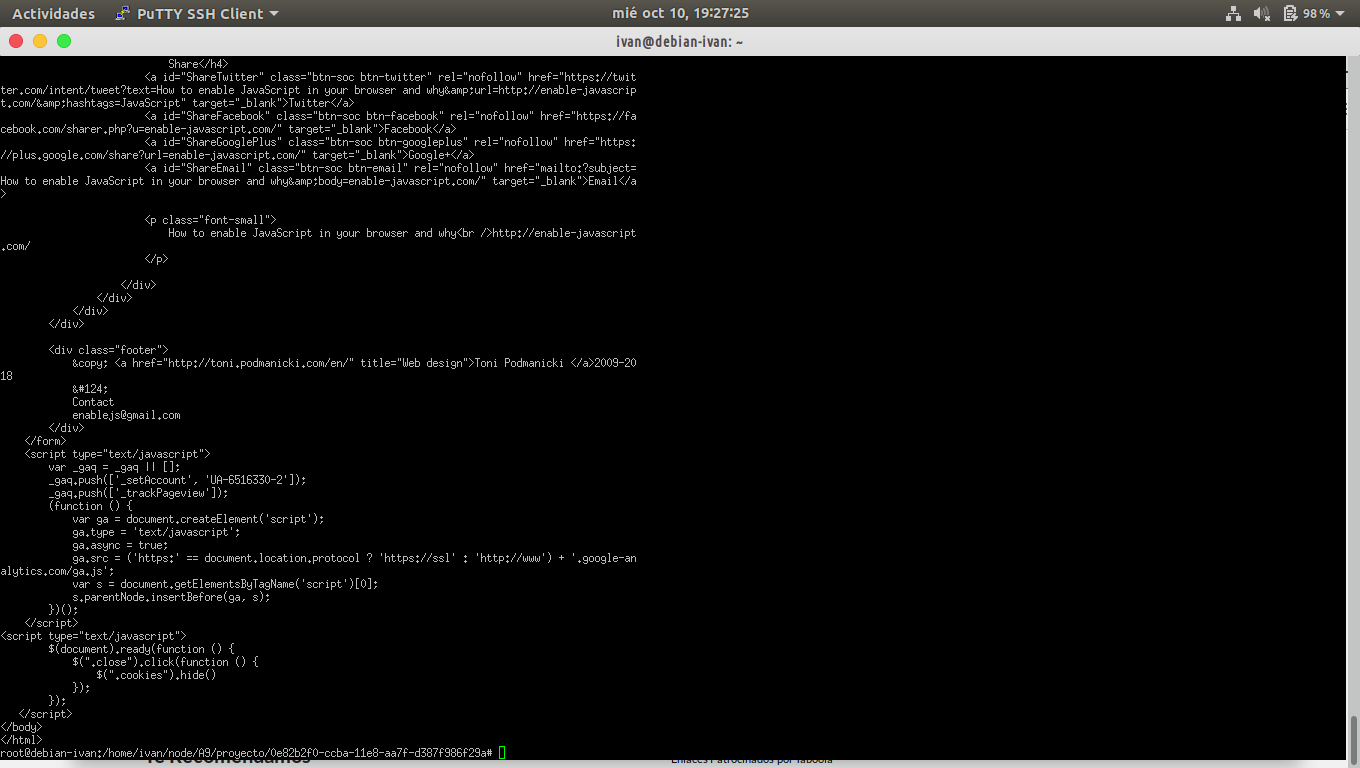


**Mostramos los archivos que existen en el proyecto y vemos como se ha creado la carpeta aleatoria.**

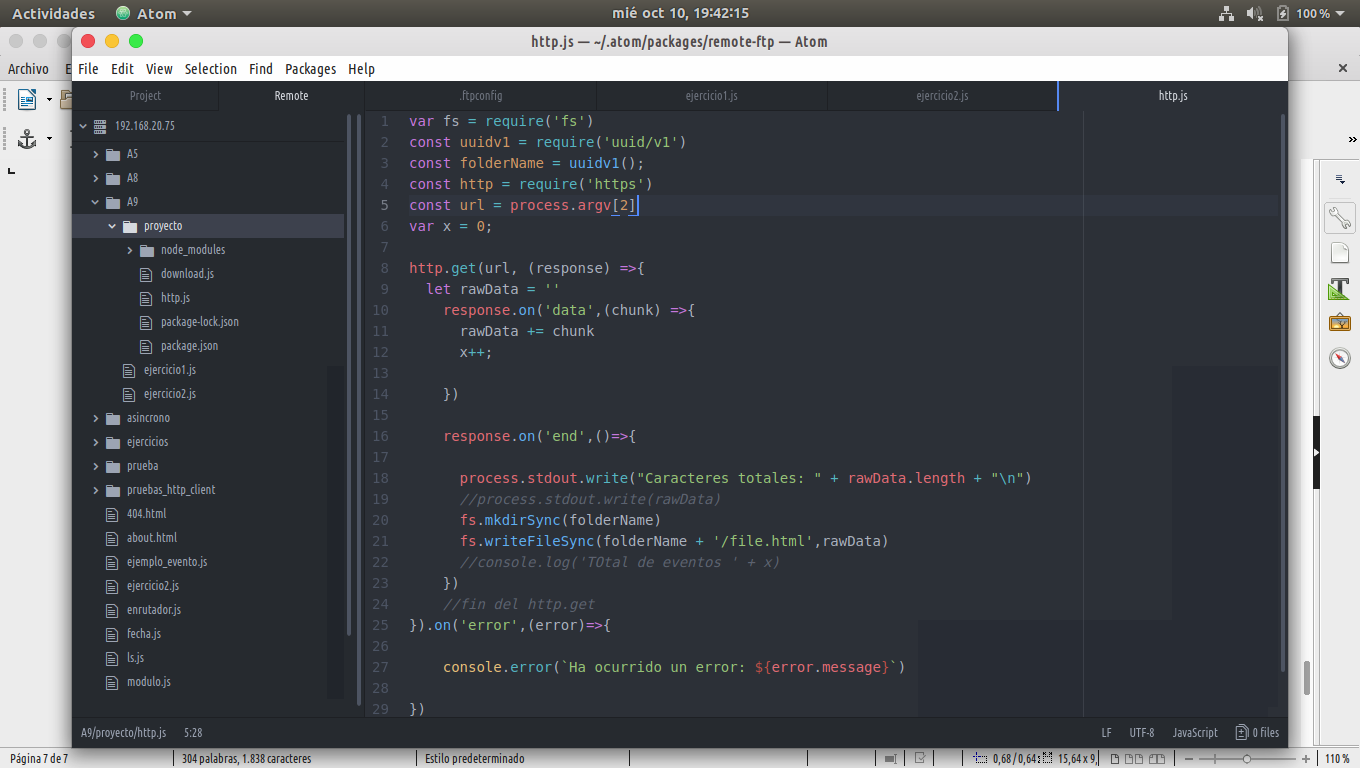
**Accedemos a la carpeta y vemos el archivo “file.html”**



**Mostramos la información del archivo**



**Código**

****