



UNIVERSIDAD VERACRUZANA
MAESTRÍA EN SISTEMAS INTERACTIVOS CENTRADOS
EN EL USUARIO

Lab 04 - Visualizador de archivos

SERVICIOS DE RED ORIENTADOS A SISTEMAS
INTERACTIVOS

Autor:

Generación 8 MSICU

2020

Contenido

Descripción de actividad	3
Análisis del problema	4
Personas o perfil de usuario	5
Flujo de sistema	7
Prototipo de baja fidelidad	8
Prototipo de alta fidelidad:	10
Tecnología utilizada	11
Diseño de la interfaz	12
Colores	12
Propuesta	14

Descripción de actividad

Lab 04 - Visualizador de archivos

Vence el 22 de abril de 2020 13:00

Instrucciones

Realizar un sistema que visualice los archivos que se encuentran distribuidos en los equipos de una red.

1. Cada equipo deberá estar corriendo el servicio (servidor) que proporciona un listado de los archivos ubicados en un directorio local (p.e. C:\recursos\), especificar en la documentación pantalla del directorio con su contenido.
2. Cualquier equipo podrá ejecutar un cliente GUI que permitirá visualizar los archivos distribuidos en la red. Suponiendo que:
 - Equipo:192.168.1.22, tiene los archivos: Archivo1, Archivo2, Archivo3. Entonces el cliente mostraría algo similar a -->(Imagen ilustrativa anexa)
3. Si se agrega contenido nuevo al directorio, la lista en el cliente se debe actualizar.

El diseño de la interfaz del cliente debe ser visualmente atractivo, respetando el diseño centrado en el usuario. Suponiendo que estará dirigido a personas que no son del área de las ciencias de la computación y que utilizan la computadora sólo para escribir documentos y presentaciones en power point.

Pueden utilizar cualquiera de las herramientas investigadas previamente.



Análisis del problema

De acuerdo con la actividad solicitada es necesario programar un servicio que estará alojado del lado del servidor, el cual gestionará la información que se mostrará en el cliente. Del lado del cliente se deberá programar una interfaz que solicitará el servicio y mostrará los archivos de todos los clientes conectados a dicho servicio, como se menciona en la descripción de la actividad, esta interfaz podrá ser utilizada por 1 o más clientes. Adicional a esto, al crear un nuevo archivo en cualquier de los clientes activos, la lista debe ser actualizada. Dicha lista deberá mostrar algunos detalles tales como el nombre y alguna referencia del propietario. La interfaz deberá ser lo suficientemente entendible para un usuario con pocos conocimientos computacionales.

Para resolver esta actividad se decide dividir las actividades a realizar entre los 13 integrantes del equipo, dividido de la siguiente manera:

- Servicio server API (Francisco Sánchez-Jesús Rosas)
- Servicio cliente que conecte al servidor (Óscar - Ramiro)
- Flujo de sistema (Ana)
- GUI CSS (Santiago - Fidel)
- Diseño prototipo (Luis Abraham)
- GUI diseño HTML (Francisco Galicia)
- GUI implementando el servicio cliente AJAX (Luis Méndez)
- GUI (lista de los archivos en el diseño) (Román)
- GUI (lista de los detalles) (Alicia)
- Documentación (Rubén)

Personas o perfil de usuario

Esta herramienta nos permite crear un personaje (arquetipo) ficticio para representar los diferentes tipos de usuarios que pueden utilizar tu servicio, producto, sitio o marca.

La creación de personas nos ayudará a entender las necesidades, experiencias, comportamientos y metas de nuestros usuarios. Igualmente nos ayuda a tener otros puntos de vista diferentes y salir de la idea de uno mismo, es decir nos ayuda a empatizar con diferentes usuarios.


El objetivo de la técnica de Persona es el de recoger información cualitativa o cuantitativa acerca del cliente que nos permita identificar arquetipos o patrones de comportamiento. Proporciona una información detallada y una comprensión consistente de un tipo de audiencia concreta. Estas fichas son una representación detallada de gente real que nos ayudan a:

- Investigar sus acciones, costumbres y comportamientos.
- Entender las principales necesidades y tareas principales.
- Priorizar los grupos de usuarios más tangibles o accesibles.
- Identificar las nuevas características y/o funcionalidades que debería tener el producto/servicio.

De acuerdo con la información proporcionada, el usuario objetivo serán personas ajenas del área de las ciencias de la computación y que utilizan la computadora principalmente para actividades de ofimática. Utilizando esta herramienta de personaje ficticio se describen algunos datos demográficos, habilidades, metas, una descripción básica del usuario, motivaciones, frustraciones y tecnología utilizada. (ver Figura 1)

NOMBRE

Bertha González



Demografía

Female

50

years

Mexico

Ocupación: Apoyo administrativo

Escolaridad: Secundaria

Estado civil: casada

Metas

- Estabilidad económica.
- Una vida tranquila.

Frase personal

La familia es lo mas importante.

Acerca de Bertha

Bertha es una persona que ha laborado por varios años en una oficina ayudando en papeleo y cosas administrativos. Normalmente su día comienza a las 7 de la mañana ya que debe viajar una hora para llegar a su trabajo. Le gusta visitar a sus hermanas los fines de semana y su hobby es salir a bailar.

Es una persona honrada, trabajadora, que vive bajo un alto nivel de estrés y preocupación debido al alto costo de la vida. También se desmotiva que personas mucho mas jóvenes aprenden mas rápido sobre lo administrativo y saben mas que ella. Su nivel socio económico es bajo, por lo que no tiene mucho acceso a tecnología.

Habilidades

Word

0

25

50

75

100

Excel

0

25

50

75

100

Power Point

0

25

50


75

100

Motivaciones



- La familia.
- Dar buena educación a sus hijos.

Navegador



Google chrome

Tecnología

Frustraciones

- Alto costo de los productos básicos.
- Inseguridad.
- Aprender nuevas cosas en la computadora.

Servicios utilizados












Figura 1. Personaje ficticio del usuario de la aplicación.

Flujo de sistema

De acuerdo con los requisitos solicitados para la aplicación, se analizan los datos y se genera un diagrama en donde se representa el flujo del sistema (ver Figura 2).

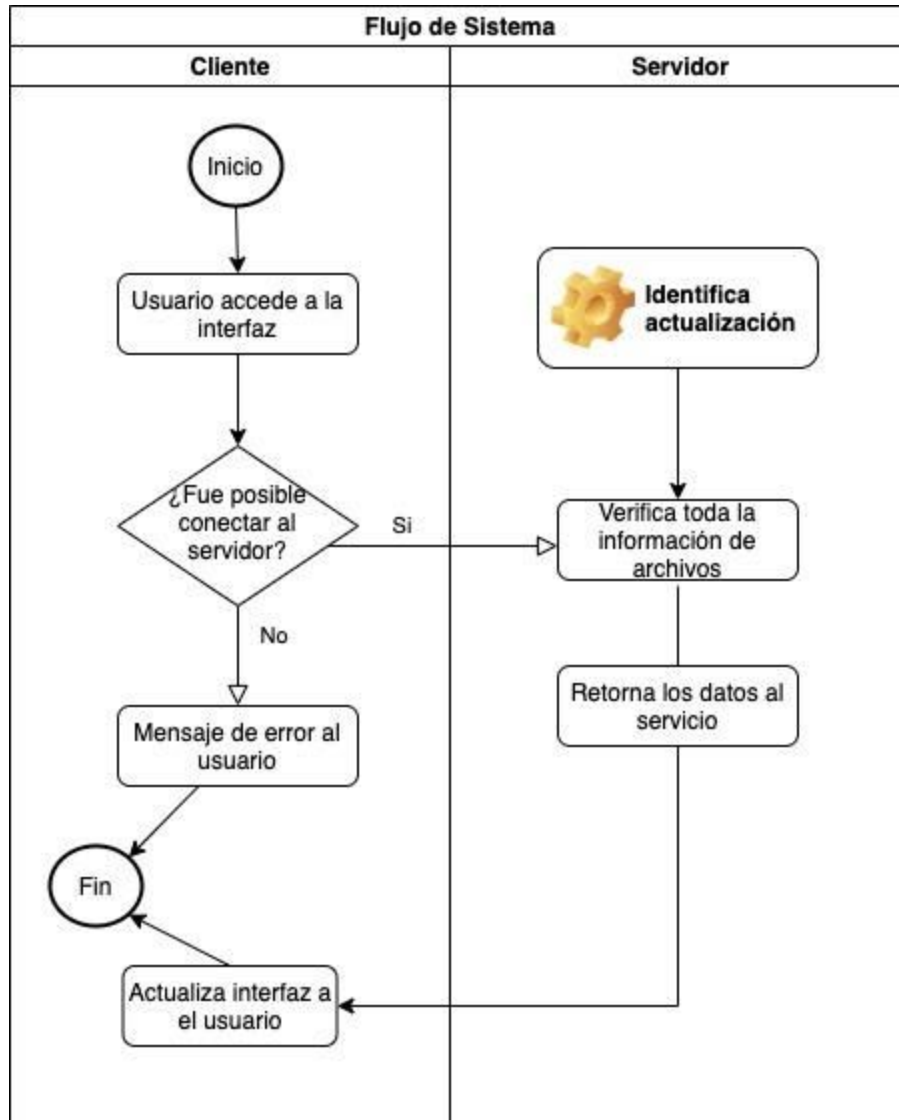


Figura 2. Flujo del sistema.

El flujo inicia con la conexión del usuario a la interfaz. Posteriormente se envía una petición de conexión al servidor, para que retorne los detalles relacionados a los archivos presentes en las carpetas de los servidores distribuidos. En caso de no ser posible realizar la conexión, un mensaje de error será presentado al usuario, que deberá recargar la pantalla para nueva conexión.

El servidor además de responder con los detalles de los archivos, también enviará una señal al cliente con la lista actualizada siempre que se agregará/crear un documento (proceso “Identifica actualización”).

Prototipo de baja fidelidad

Con base en el flujo del sistema se identifica la información necesaria, se procede a realizar la primera versión. (ver Figura 3). Entre todo el equipo de trabajo se revisa detenidamente y se realizan algunas correcciones y modificaciones y se desarrolla una segunda versión (ver Figura 4), la cual con los últimos detalles se llega a la tercera y última versión para pasar a la siguiente etapa. (ver Figura 5)

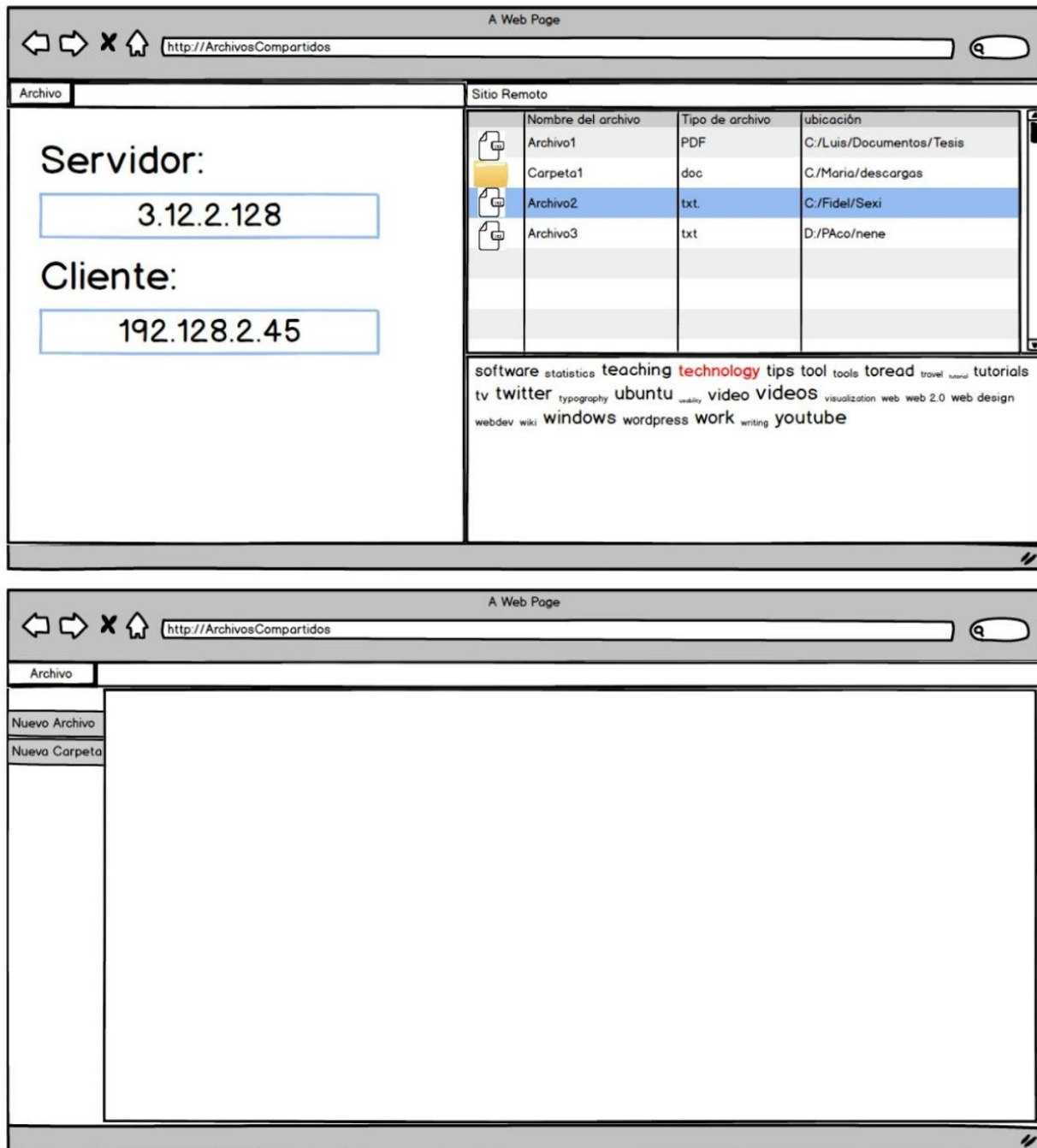


Figura 3. Prototipo de baja fidelidad (Versión 1).

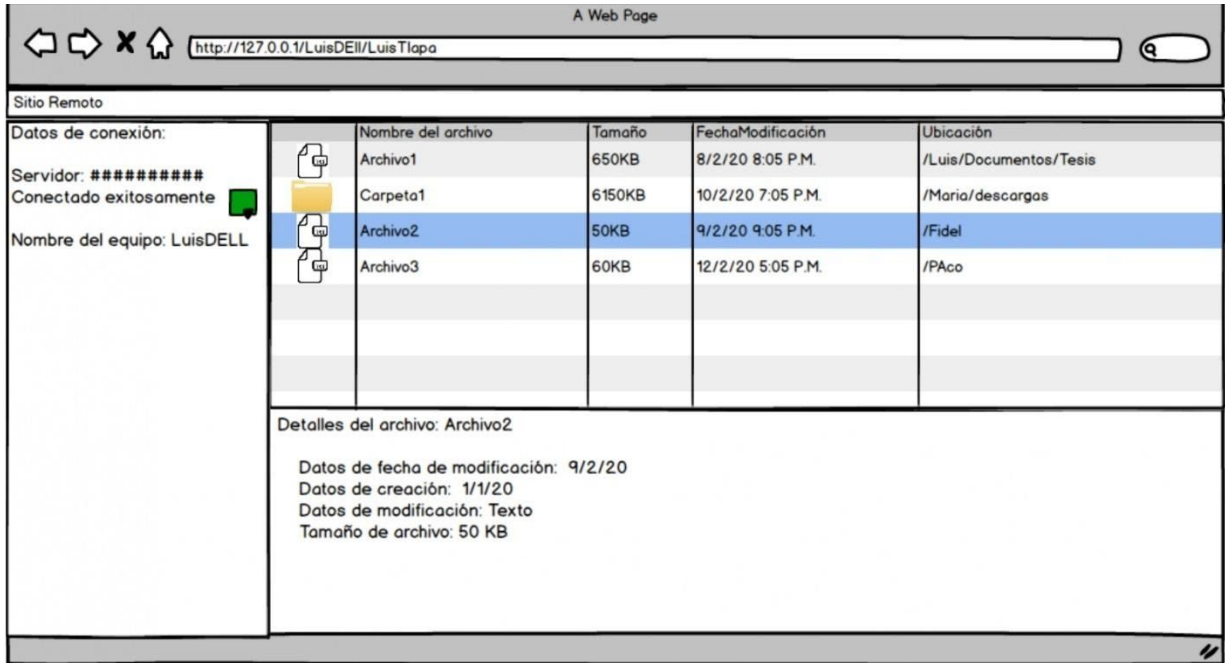


Figura 4. Prototipo de baja fidelidad (Versión 2).

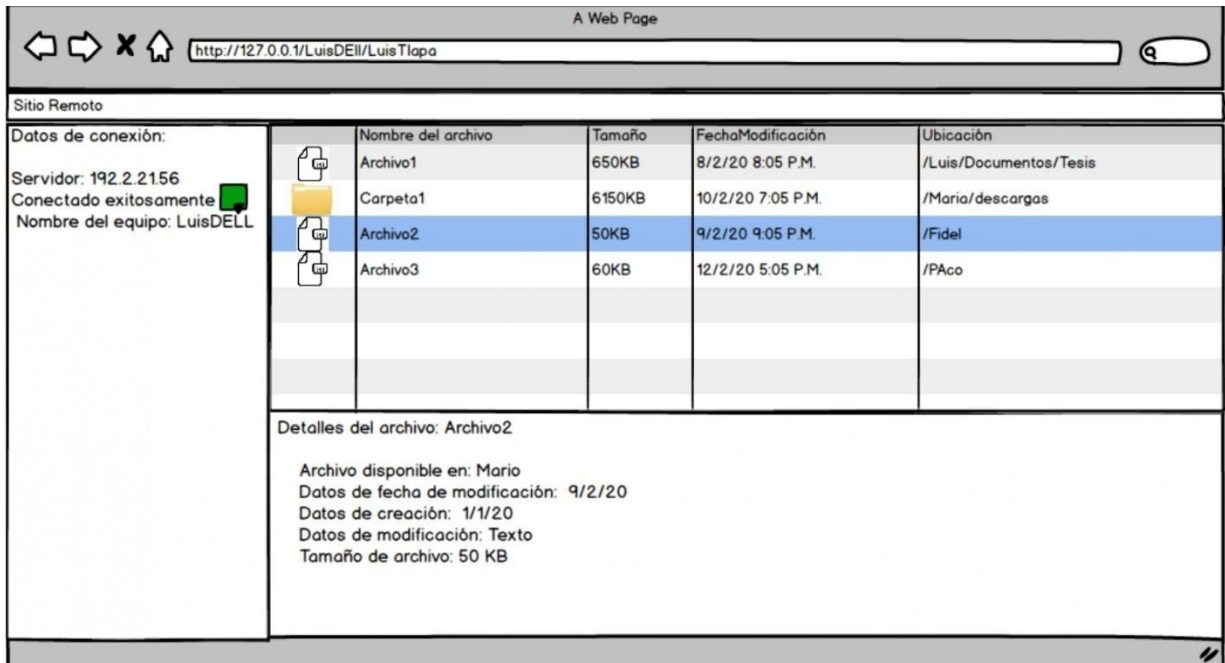


Figura 5. Prototipo de baja fidelidad (Versión 3).

Prototipo de alta fidelidad

Posterior a los prototipos de baja fidelidad se continúa con los prototipos de alta fidelidad donde se hacen las modificaciones y mejoras sugeridas entre todo el equipo para estos prototipos, los cuales se presentan a continuación.

Información de conexión



FILEMSICU
Visualizador de archivos

Sistema desarrollado como parte de la actividad Lab 04.

Servidor

Estado de la conexión

Equipo conectado:

Nombre del equipo

Lista de archivos compartidos en la conexión

Icono	Nombre del archivo	Tamaño del archivo	Fecha de modificación	Ubicación del archivo
1,001	Lorem	ipsum	dolor	sit
1,002	amet	consectetur	adipiscing	elit
1,003	Integer	nec	odio	Praesent

Detalles del archivo seleccionado:

Nombre de archivo:

Nombre del archivo

Fecha de creación:

DD-MM-AAAA

Fecha de última modificación:

DD-MM-AAAA

Datos de modificación:

Modificación

Disponible en:

Dirección del equipo

Tamaño del archivo:

Información de conexión



FILEMSICU
Visualizador de archivos

Sistema desarrollado como parte de la actividad Lab 04.

Servidor

Estado de la conexión

Equipo conectado:

Nombre del equipo

☰

Lista de archivos compartidos en la conexión

Icono	Nombre del archivo	Tamaño del archivo	Fecha de modificación	Ubicación del archivo
	Lorem	ipsum	dolor	sit
	amet	consectetur	adipiscing	elit
	Integer	nec	odio	Praesent
	Integer	nec	odio	Praesent

Detalles del archivo seleccionado:

Nombre de archivo:

Nombre del archivo

Fecha de creación:

DD-MM-AAAA

Fecha de última modificación:

DD-MM-AAAA

Datos de modificación:

Figura 6. Prototipos de alta fidelidad.

Tecnología utilizada

Por la naturaleza de la actividad se decide utilizar las siguientes tecnologías para el desarrollo de esta aplicación.

Back-end:

- Python 2.7 o superior
- Websocket Python

Front-end:

- Bootstrap 4
- HTML 5
- CSS
- Javascript
- Fontawesome

Diseño de la interfaz

Colores

Para definir los colores utilizados para la interfaz de usuario se recurrió al manual de identidad proporcionado por la universidad veracruzana, el cual se encuentra disponible en el la siguiente URL: <https://www.uv.mx/comunicacionuv/manual-de-identidad/>

De acuerdo con el manual de identidad de la universidad veracruzana los colores escogidos para la interfaz son los siguientes con su correspondiente paleta de colores.

Azul: #18529d



Figura 7. Paleta de colores, tonalidad azul

Verde: #28AD56

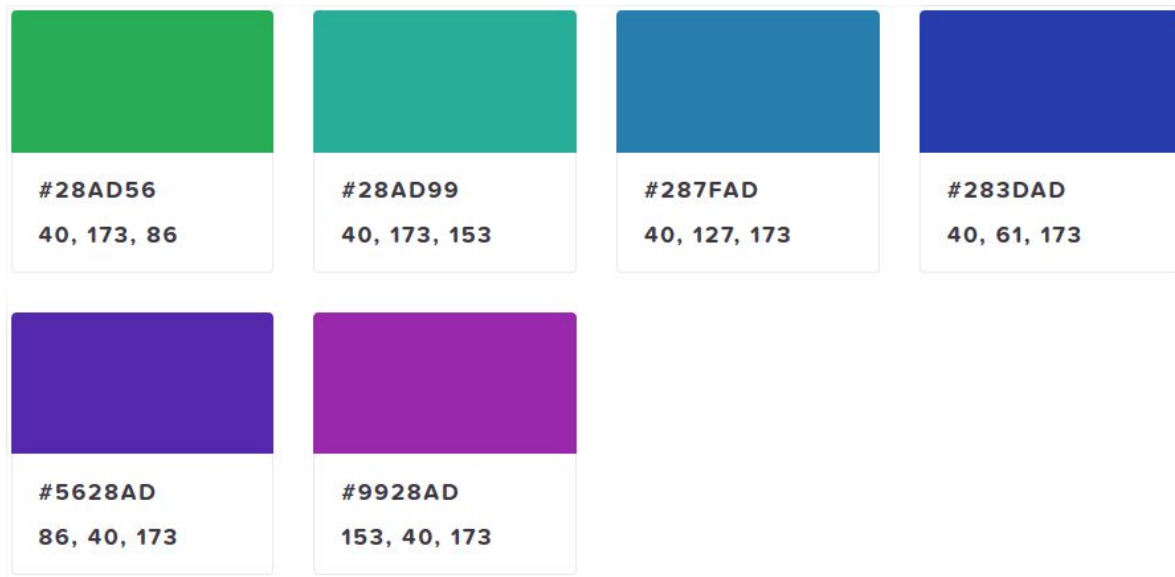


Figura 8. Paleta de colores, tonalidad verde.

Negro: #000000

Blanco: #FFFFFF

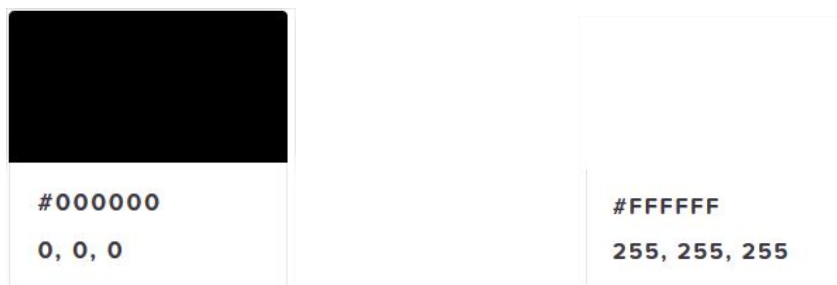
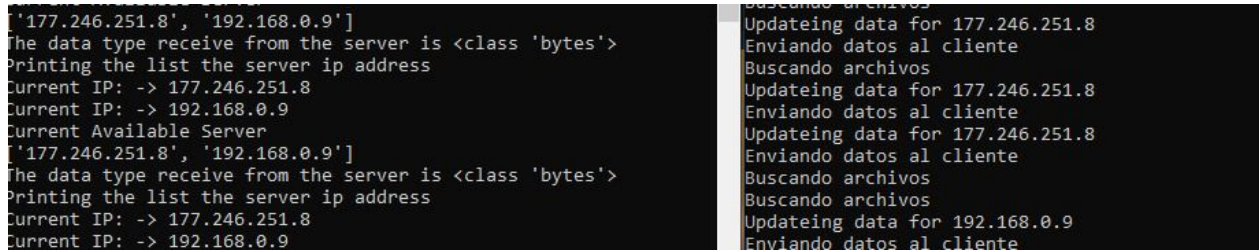


Figura 9. Paleta de colores, tonalidades extra.

Propuesta

Tomando en cuenta todos los aspectos de diseño, la aplicación por parte del backend al correrse el servicio web dentro de cada servidor, este escuchará las peticiones de los demás que estén conectados y mandarán estos archivos a través del servicio, para ser consumidos por el cliente.

Una muestra de esto en la consola es lo siguiente:



```

['177.246.251.8', '192.168.0.9']
The data type receive from the server is <class 'bytes'>
Printing the list the server ip address
Current IP: -> 177.246.251.8
Current IP: -> 192.168.0.9
Current Available Server
['177.246.251.8', '192.168.0.9']
The data type receive from the server is <class 'bytes'>
Printing the list the server ip address
Current IP: -> 177.246.251.8
Current IP: -> 192.168.0.9

Buscando archivos
Updateing data for 177.246.251.8
Enviando datos al cliente
Buscando archivos
Updateing data for 177.246.251.8
Enviando datos al cliente
Updateing data for 177.246.251.8
Enviando datos al cliente
Buscando archivos
Buscando archivos
Updateing data for 192.168.0.9
Enviando datos al cliente
  
```

Figura 10. Backend de consola

EL servidor al momento de crear el servicio web, mostrará en consola las IP's de los servidores corriendo este servicio, después el cliente desde la interfaz podrá consumir el servicio y mostrar los archivos de cada servidor.

Del lado de la interfaz, al estar escuchando el servicio web de los servidores, el cliente podrá explorar la lista de archivos de los servidores para poder ver sus detalles al momento de dar clic

Lista de archivos compartidos en la conexión

ID	Nombre del archivo	Tamaño del archivo	Fecha de modificación	Ubicación del archivo
9673	.git	4096	April 24, 2020, 5:39 p.m.	C:\Users\theme\Desktop\redes
9674	.gitattributes	66	April 24, 2020, 5:39 p.m.	C:\Users\theme\Desktop\redes
9675	db.sqlite3	151552	April 24, 2020, 5:39 p.m.	C:\Users\theme\Desktop\redes
9676	inicio	4096	April 24, 2020, 5:39 p.m.	C:\Users\theme\Desktop\redes
9677	localFiles.txt	106	April 24, 2020, 5:39 p.m.	C:\Users\theme\Desktop\redes
9678	localJsonFiles.txt	0	April 24, 2020, 5:39 p.m.	C:\Users\theme\Desktop\redes
9679	manage.py	646	April 24, 2020, 5:39 p.m.	C:\Users\theme\Desktop\redes
9680	README.md	18	April 24, 2020, 5:39 p.m.	C:\Users\theme\Desktop\redes

Figura 11. Tabla de archivos

El código de programación, utilizó el framework django, apoyado de websockets para la comunicación y API rest para consumir la información del server y mostrarla en la interfaz.

El enlace al proyecto se encuentra en: <https://github.com/bugideveloper/redes/>