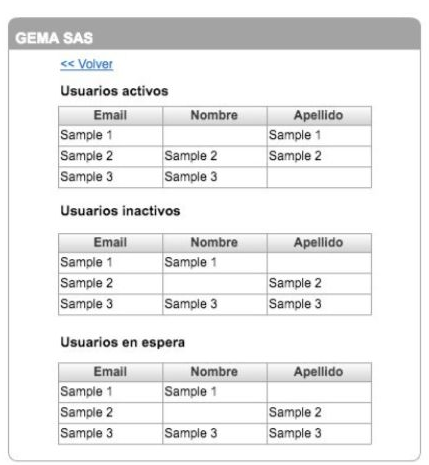
Nombre: Fabio Andrés Ortiz

Cédula: 1032506144

Problema:

La empresa de datos GEMA SAS necesita fortalecer uno de sus procesos más importantes. Para ello necesita un Sitio web que le permita cargar un archivo de texto plano (Adjunto) usando un formulario, para su posterior análisis. La idea es que después de cargar en el formulario el archivo de texto (Adjunto) se puedan ver los datos de manera organizada en 3 diferentes tablas. Una correspondiente a usuarios activos, otra a inactivos y otra en proceso de espera. Para ello GEMA SAS envió unos Mockups con el resultado esperado.

Mock Ups:

Planteamiento Solución:

Implementar una interconexión entre PHP y MYSQL, primero debe solucionarse la subida del archivo, posteriormente en MYSQL completar la tabla y a través de la extensión seleccionada, realizar los respectivos filtros mostrando la información requerida.

Tiempo Planteado de Solución: 4 horas (2:30 horas programación, 1 hora en estilo y 30 minutos de documentación)

Desarrollo Solución:

Se selecciona XAMPP para habilitar un localhost, para la interconexión entre MYSQL y PHP se utilizará PDO. En el htdocs se crean 3 documentos iniciales, un index como pantalla principal, “dbconnect” y “connection” para la conexión con la base de datos. En “connection” están descritos los valores variables entre computador a computador, como son el host, dbname, username, password; han de reemplazarse acorde a la configuración de la máquina.

“dbconnect” instancia la variable “conn” que servirá a las futuras conexiones que deban ser realizadas para las consultas, es decir las querys.

Se utiliza MySQL Workbench para la base de datos, se crea una base de datos (llamada “test” en este caso) y una tabla conformada por: email, name (Nombre), lastname (Apellido) y state (estado). Revisar .sql anexo.

Comprobar si la conexión con MYSQL es exitosa.

El archivo .txt se carga en un formulario con un input tipo “File”, utilizando el método post. Se crea una nueva página donde serán mostrados los resultados “visualization.php”, a través de un isset de comprobación del formulario se guarda el archivo en una ruta independiente de cada PC para evitar el error 1290. Para mostrar donde se encuentra tal ruta utilizar el comando “SHOW VARIABLES LIKE 'secure\_file\_priv'” en MYSQL (explicado en el manual de instalación). El siguiente comando borra la información de la tabla, si existe o no, para evitar inconvenientes de duplicados cada vez que se suba un nuevo archivo.

La función “uploadData” a partir de la dirección donde fue guardado el documento la cargará en la tabla “datos”, dividiendo los campos por ‘,’ y las filas con '\n'. Más adelante se explicará la función “check”.

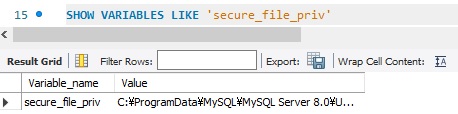
Ya “completada” la tabla, se procede a realizar las consultas correctas en la construcción de las mismas, cada tabla está estructurada en sentido similar. Se realiza una consulta que devolverá los resultados acordes al estatus, así la primera de “Activos” tiene al query especificando el seleccionar únicamente a quienes tienen de estado el 1, es decir, son activos.

Una excepción es si el estado tiene algún valor no permitido, es decir que no sea ni 1,2 o 3. Para cumplir el requerimiento se desarrolla la función “check”. En está se realiza una consulta donde se cuenta cuántos valores no pertenecen a tales menciones, si es superior a 0 se redirige al index con una alerta señalando el error del formato.

Para los estilos fue utilizado “Bulma” una librería de CSS.

Manual de Instalación:

* Tener instalado XAMPP y MYSQL WORKBENCH para la interconexión MYSQL/PHP.
* Copiar todos los archivos en la carpeta “htdocs” de XAMPP.
* Ejecutar el test.sql en MYSQL WORKBENCH.
* En “conection.php” cambiar los datos según el equipo que esté utilizando.
* Ejecutar la instrucción “SHOW VARIABLES LIKE 'secure\_file\_priv'” en MYSQL WORKBENCH, la dirección que aparezca reemplazar en “visualization.php” en $route y uploadData.



Tiempo real de Solución: 5:30 horas (4:00 horas programación, 50 minutos en estilos y 40 minutos de documentación)