

Julián Andrés Barrios Corredor

Prueba técnica GEMA SAS

JerreJerre

- Tiempo planeado de solución: **4-5 horas**
- Tiempo real de solución: **5 horas y 30 minutos**

Documentación de la solución:

Para la elaboración de la prueba técnica tuve en cuenta el patrón de diseño MVC, basándose en ciertas partes como el inicio de la aplicación y el manejo de vistas del framework Laravel. La aplicación cuenta con dos carpetas principales: App y public. La primera contiene todo el código que permite el funcionamiento de la parte lógica de la aplicación, tales como los modelos, la conexión a la base de datos, los controladores, e incluso la clase que se encarga de inicializar la aplicación. Por otro lado, la carpeta public contiene las vistas y los assets de la aplicación, tales como los archivos css y javascript.

En el archivo index.php, localizado en la raíz del proyecto, se incluye la clase Kernel.php, la cual se encarga de inicializar la aplicación, proveer configuración básica para esta, y realizar el proceso de redirección a los controladores y métodos indicados por medio de los parámetros 'c' y 'm' (controlador y método, respectivamente) pasados por medio de la url. Esto se hace a través de un objeto, llamado App Dispatcher, el cual se encarga de manejar dicha redirección. Si ninguno de los parámetros ('m' y 'c') es proveído, el AppDispatcher hará una instancia del controlador por defecto, llamado IndexController, y llamará a su método por defecto, el cual es método index.

En dichos controladores se realiza toda la parte lógica de la aplicación, de manera que es en estos en donde se recibe la información proveída por el usuario y se da una respuesta, la cual puede ser la renderización de una vista o la redirección a otro controlador. Cabe aclarar que cada controlador debe heredar la clase BaseController e implementar el método index. Esta clase provee además acceso a la renderización de vistas, mediante la clase de utilidad View, la cual busca la vista especificada en la carpeta public/views y se encarga de mostrarla al usuario, pasando a su vez las variables que se hayan especificado.

Por otro lado, se encuentran los modelos (los cuales son usados en el controlador). Estos modelos heredan de la clase BaseModel, la cual les provee el acceso a la base de datos, para que cada uno de estos pueda realizar las operaciones correspondientes sobre esta.

Manual de instalación:

1. Para el correcto funcionamiento de la solución se requiere la instalación de php (versión $\geq 7.0.0$), un servidor apache, y la base de datos MySQL o MariaDB. Estos tres componentes pueden ser encontrados en la siguiente dirección

<https://www.apachefriends.org/index.html>

Por otro lado, se requiere la instalación de la herramienta de manejo de dependencias para php, llamada composer, ya que mediante esta se gestionan los namespaces de la aplicación. Esta herramienta se puede encontrar en la siguiente dirección:

<https://getcomposer.org/>

Con los anteriores componentes instalados, se prosigue entonces a mover la carpeta de la aplicación al directorio del servidor apache. Dicho directorio varía dependiendo de la distribución usada:

- Windows: C:\xampp\htdocs
 - Linux: /var/www/html (para la mayoría de distribuciones)
 - macOS: /Library/WebServer/Documents
2. Luego de esto, es necesario abrir una terminal en el directorio de la aplicación, y ejecutar el siguiente comando: `composer dump-autoload`. De esta manera, todas las clases podrán ser cargadas sin problemas.
 3. Como acto seguido, es necesario crear la base de datos al servidor local. Para esto, hay diferentes formas: por un lado, se puede realizar mediante la herramienta PHPMysqlAdmin, proveída por XAMPP. Una vez en esta, nos dirigimos a la pestaña 'import', le damos click al botón 'choose file' y cargamos el archivo 'script.sql' localizado en la raíz del proyecto. Para finalizar damos click en el botón 'Go' localizado al final de la página.
 4. Finalmente, para ver la aplicación en funcionamiento, nos dirigimos a la URL en el navegador de elección: `http://localhost/proyectojerrejerre/` y veremos entonces la pantalla inicial esperada según los requerimientos.