**Instalación**

En la carpeta del proyecto debe colocarse en el directorio raíz del servidor que se esté utilizando ya sea WAMP Server o XAMPP.

Luego para poder ejecutar el proyecto es necesario cambiar la ruta de ubicación de los archivos ya que se ha realizado utilizando direcciones absolutas, se debe dirigir a la carpeta del proyecto en la cual se encuentra la carpeta llamada Config con dos archivos, acceder al archivo “config.php” y en el reemplazar la constante llamada “define('URL', 'http://localhost:8080/MVC-TPFinalWebII/');” con el valor definido de su directorio raíz utilizado normalmente seguido del nombre de la carpeta donde esta ubicado el proyecto.

Con estos pasos ya es posible ejecutar la aplicación web.

Se debe considerar que esta aplicación fue realizada con la versión de PHP 7.1 o superior y se habilito el uso de SHOT\_OPEN\_TAG en el servidor PHP para aumentar la eficiencia a la hora de escribir código.

El servidor utilizado para realizar esta aplicación es el de WAMP Server.

**Problemas**

La aplicación de Gestión de Turnos Para Una Clínica fue realizada utilizando el patrón de arquitectura de software MVC (Model-View-Controller) .

**Obtener Fechas de Turnos**: uno de los principales problemas que surgieron durante el desarrollo fue la forma en la que se podía obtener el rango de fechas y horas en las que un doctor podía atender a los pacientes. El mismo se solucionó utilizando objetos provistos por el lenguaje PHP de tipo DateTime (objeto que puede almacenar fechas y horas, además de otras configuraciones), DatePeriod(dado una fecha de inicio, un intevalo de tiempo y una fecha de fin retorna un array recorriendo desde la fecha de inicio hasta la fecha de fin en el intervalo dado) y DateInterval(genera un objeto con intervalos de tiempo) las cuales combinadas se utilizaron para crear una función la cual retorna un rango de fechas en los periodos de tiempo indicados los cuales en este caso son 20min, 30min, 40min y 1hs.

**Manejo de las vistas según el rol**: a la hora de iniciar sesión con un usuario era necesario saber que vista se iba a mostrar ya que existían distintos tipos de usuarios. Este problema se solucionó mediante el uso de roles de usuario en la base de datos los cuales son administrador, admisión y paciente, por lo que a la hora de iniciar sesión con un usuario y contraseña la aplicación verifica en la base de datos que rol posee el usuario que está iniciando y luego lo guarda en una variable de sesión la cual lo mantiene mientras el usuario este activo. Mediante esta variable de sesión que contiene el rol del usuario y su número de documento se utiliza para permitir o denegar el acceso a las distintas vistas de la aplicación las cuales están asignadas en un archivo de configuración .JSON el cual contiene los roles y las paginas a las que puede acceder cada usuario.

**Contraseñas seguras**: a la hora de guardar las contraseñas en la base de datos era necesario utilizar un sistema de encriptación para aumentar la seguridad de la aplicación, el mismo se solucionó utilizando la función generadora de hash de PHP (password\_hash) la cual genera un hash a partir de la contraseña del usuario.

**Enviar Emails y Generar PDF**: para poder imprimir los turnos que los usuarios sacaban era necesario generar un archivo pdf con la información del mismo, este problema fue solucionado utilizando la librería de php FPDF la cual permite generar archivos pdf de manera fácil y rápida. A la hora de que un usuario sacaba un turno el sistema debía enviarle un email con los datos del mismo, este problema se solucionó utilizando la librería de php llamada PHPMailer la cual se debe configurar para poder en enviar emails a través del HOST de Gmail (smtp.gmail.com).