Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский государственный технический университет»  
Кафедра «ИИТ»

Лабораторная работа №1  
Тема: «Web-сервер Apache. Стек WAMP»

Выполнил: студент ФЭИС

Группы ПО-4  
Яковчик И.А.  
Проверил: Михняев А.Л.

Брест 2022

ЦЕЛЬ РАБОТЫ:

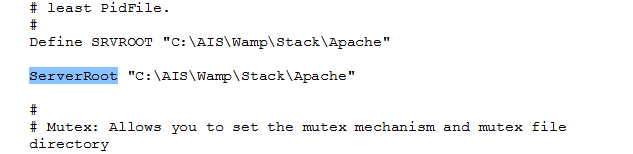
* Освоить сборку стека WAMP (Windows+Apache+MySQL+PHP) из отдельных  
  компонентов.
* Изучить основные принципы развертывания web-приложений на примере  
  wordpress.
* Организовать нагрузочное тестирование сайта.

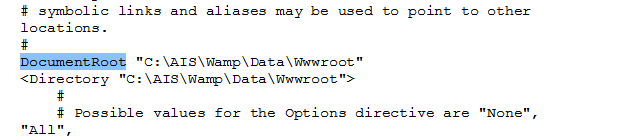
ХОД РАБОТЫ  
  
Создаём следующую структуру каталогов:

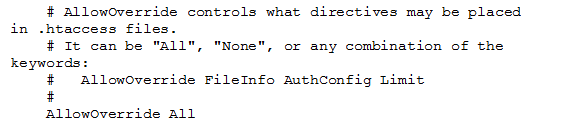
* C:\AIS\Wamp
  + Data - данные веб-приложения
    - Db - файлы базы данных mysql
    - Wwwroot - корень web-сервера
* Distrib - дистрибутивы с web-приложениями, доп. Компоненты
* Stack – компоненты стека web-приложений
* Apache - веб-сервер
* Mysql – СУБД
* Php - компоненты языка программирования webприложений

НАСТРОЙКА APACHE

В файл C:\AIS\Wamp\Stack\Apache\conf\httpd.conf были внесены изменения, требуемые в ходе лабораторной работы. Примеры изменений:

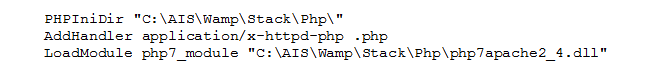


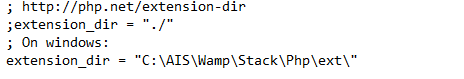
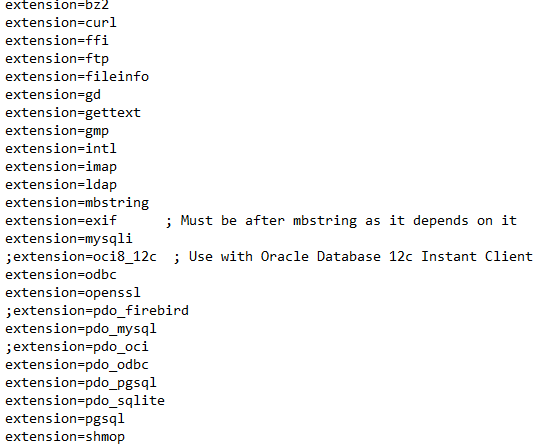


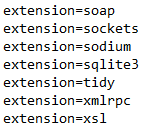


ПОДКЛЮЧЕНИЕ МОДУЛЯ PHP К APACHE

В файл c:\ais\wamp\stack\Apache\conf\httpd.conf в самый конец были добавлены  
следующие строчки:

  
В папке c:\wamp\stack\php переименовали php.ini-development в php.ini. В данный файл  
внесли следующие изменения:



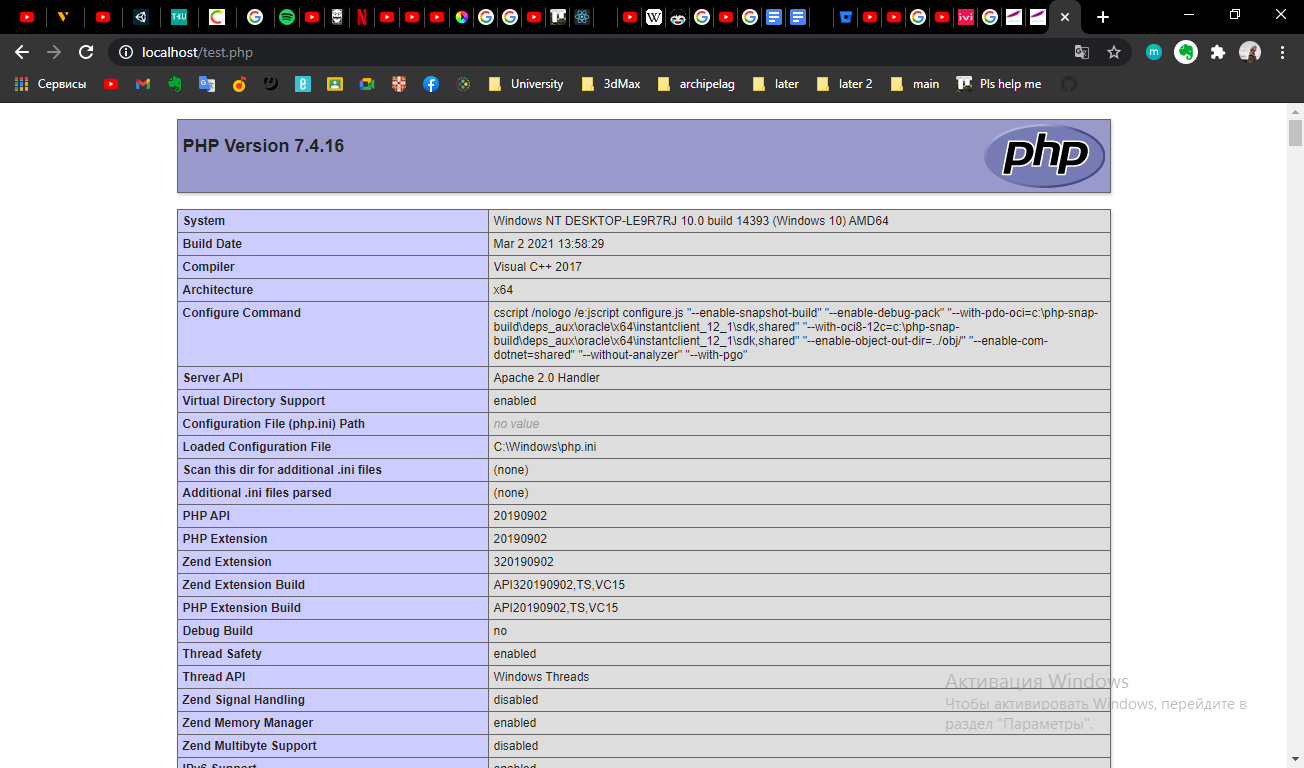
Запустим Apache следующей командой:



Создадим файл test.php и скопируем в него следующее:

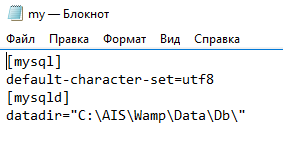
<?php  
phpinfo();  
?>

При переходе в браузере по ссылке **http://localhost/test.php** выдает следующее:

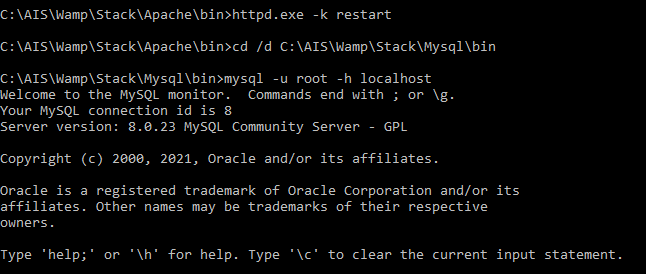


Значит, PHP и Apache настроены.

НАСТРОЙКА MySQL  
Заходим в папку C:\AIS\Wamp\Stack\Mysql и создаем файл my.ini со следующим  
содержанием:



Перезапустили Apache и ввели в cmd.exe следующее:



Mysql сервер работает.

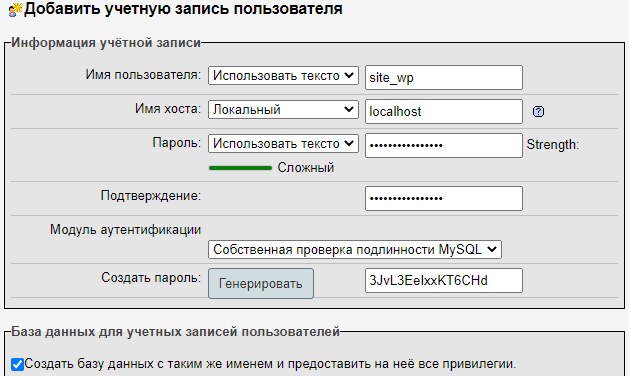
УСТАНОВКА PHPMyAdmin

С помощью данного приложения будут осуществляться все настройки баз данных  
mysql.  
В папку C:\AIS\Wamp\Data\Wwwroot\Phpmyadmin распакуем архив C:\AIS\Wamp\Distrib\phpmyadmin.zip и добавим файл config.inc.php со следующим содержанием:



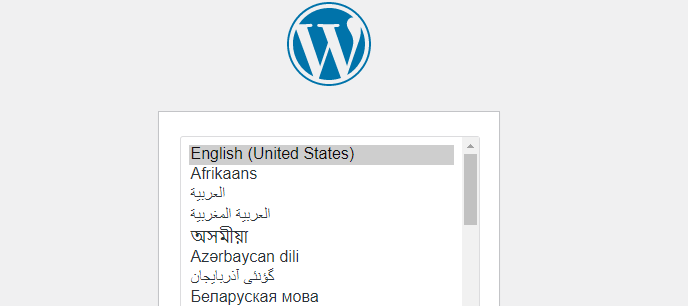
Теперь через адрес **http://localhost/phpmyadmin/** можно попасть на интерфейс веб-приложения phpMyAdmin.

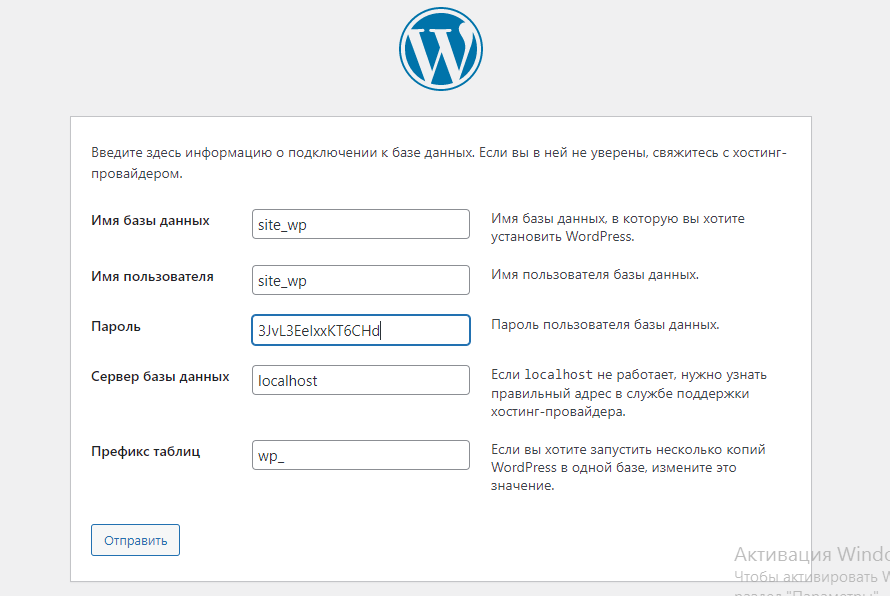
УСТАНОВКА WORDPRESSС помощью phpmyadmin создали нового пользователя site\_wp и базу данных с тем же именем. Пароль сгенерировал сам phpMyAdmin:

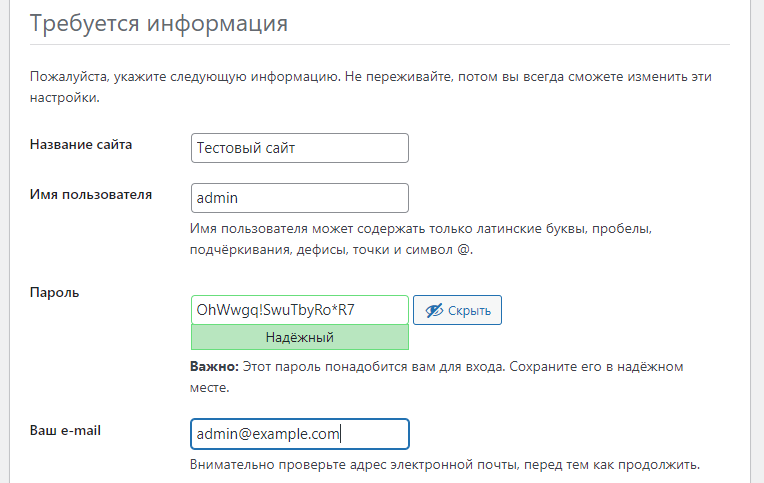


Распаковали содержимое C:\AIS\Wamp\Distrib\Wordpress.zip в папку  
C:\AIS\Wamp\Data\Wwwroot.

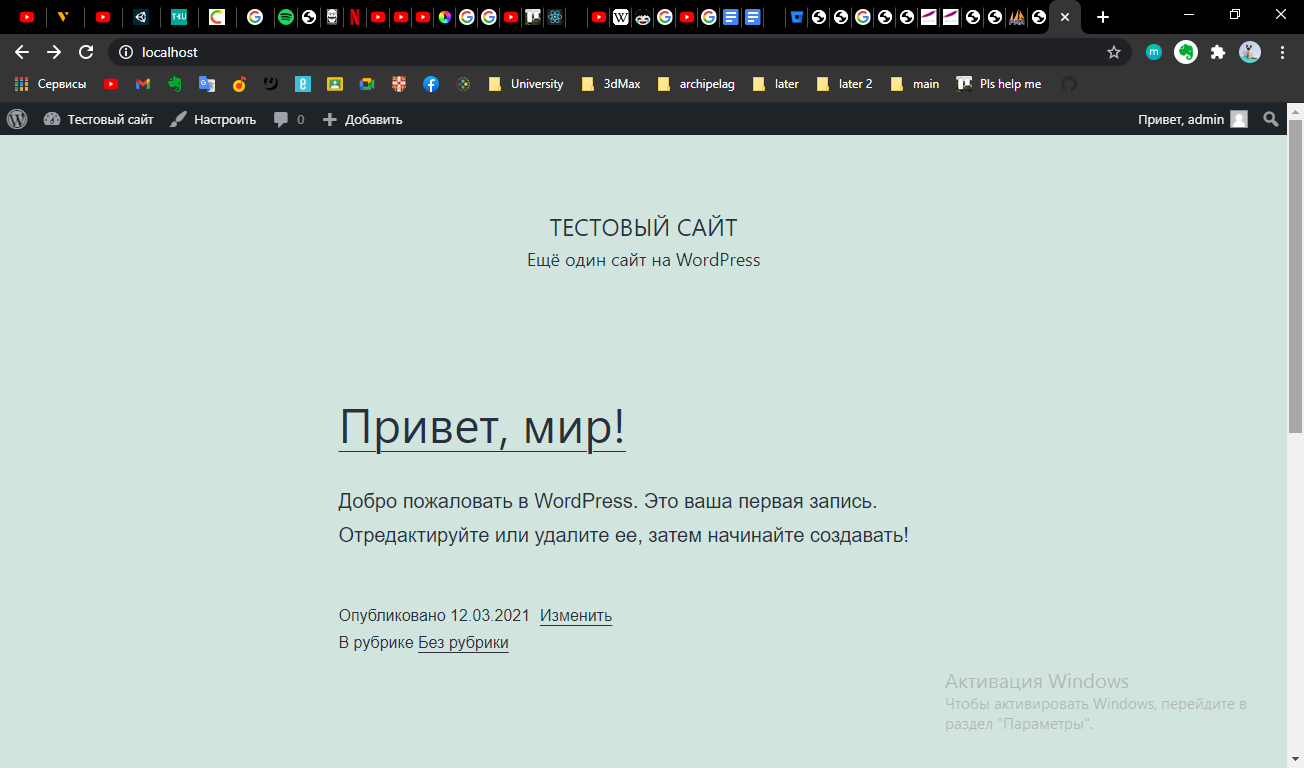
При переходе по адресу https://localhost запускается мастер настройки wordpress.  
Произвели его настройку:







При переходе на http://localhost теперь открывается базовый сайт wordpress.



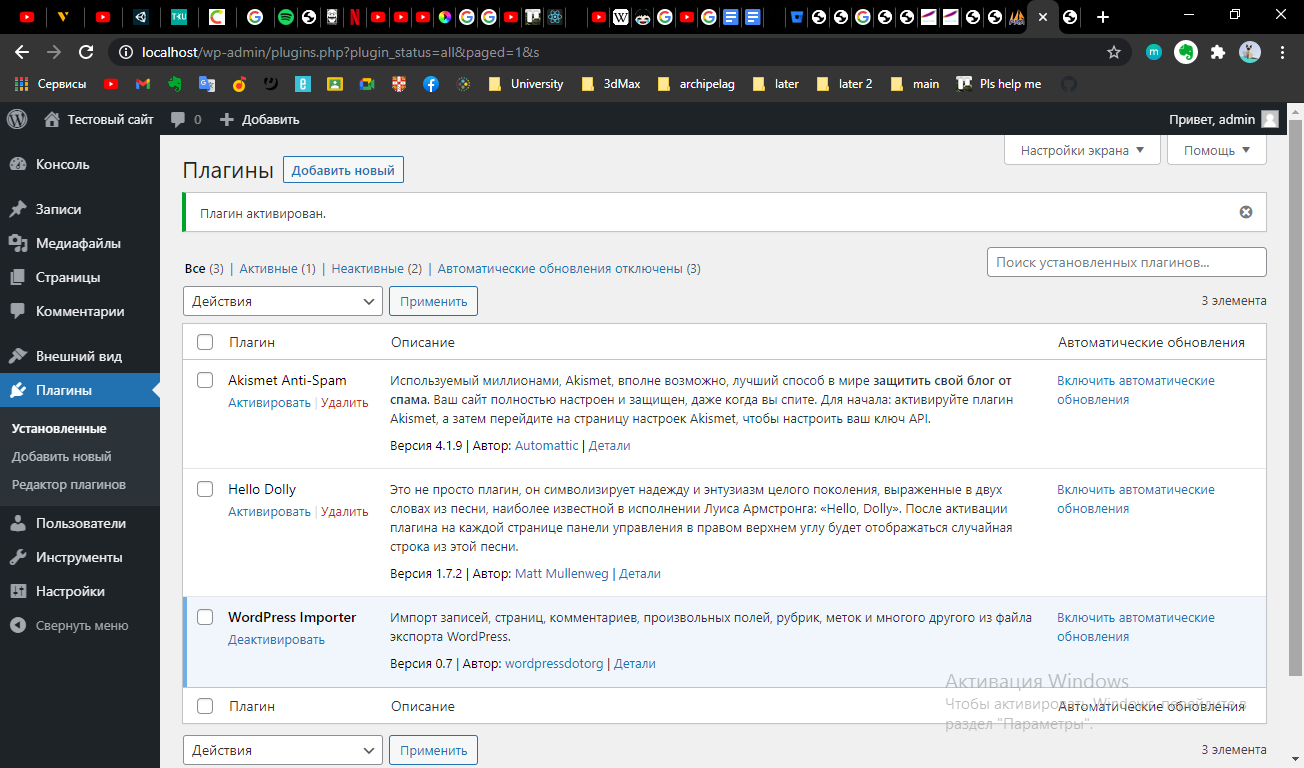
Административная панель открывается по адресу <http://localhost/wp-admin>.

НАПОЛНЕНИЕ САЙТА ТЕСТОВЫМИ ДАННЫМИ

Зайдим в административную панель. Запускаем http://localhost/wp-admin, заходим под пользователем admin.

Устанавливаем wordpress-importer:

* Плагины->Добавить новый->Загрузить плагин
* Загрузите файл плагин D:\wamp\distrib\wordpress-importer.0.6.4.zip
* Активировать плагин



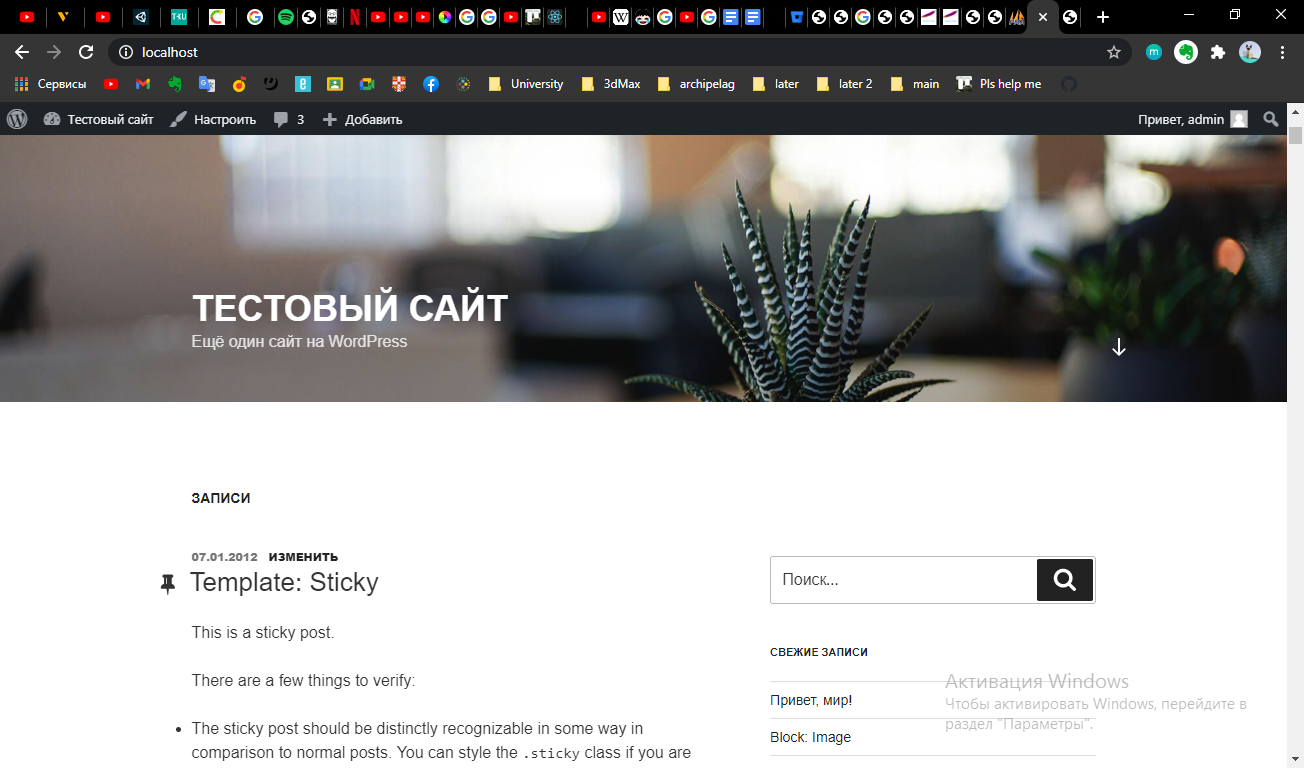
Импортируем тестовые данные:

* Инструменты->Импорт->Wordpress
* Загрузите данные из файла D:\wamp\distrib\themeunittestdata.wordpress.xml
* Загрузить и импортировать файл
* Импорт вложений: Скачать и импортировать файлы вложений
* Отправить

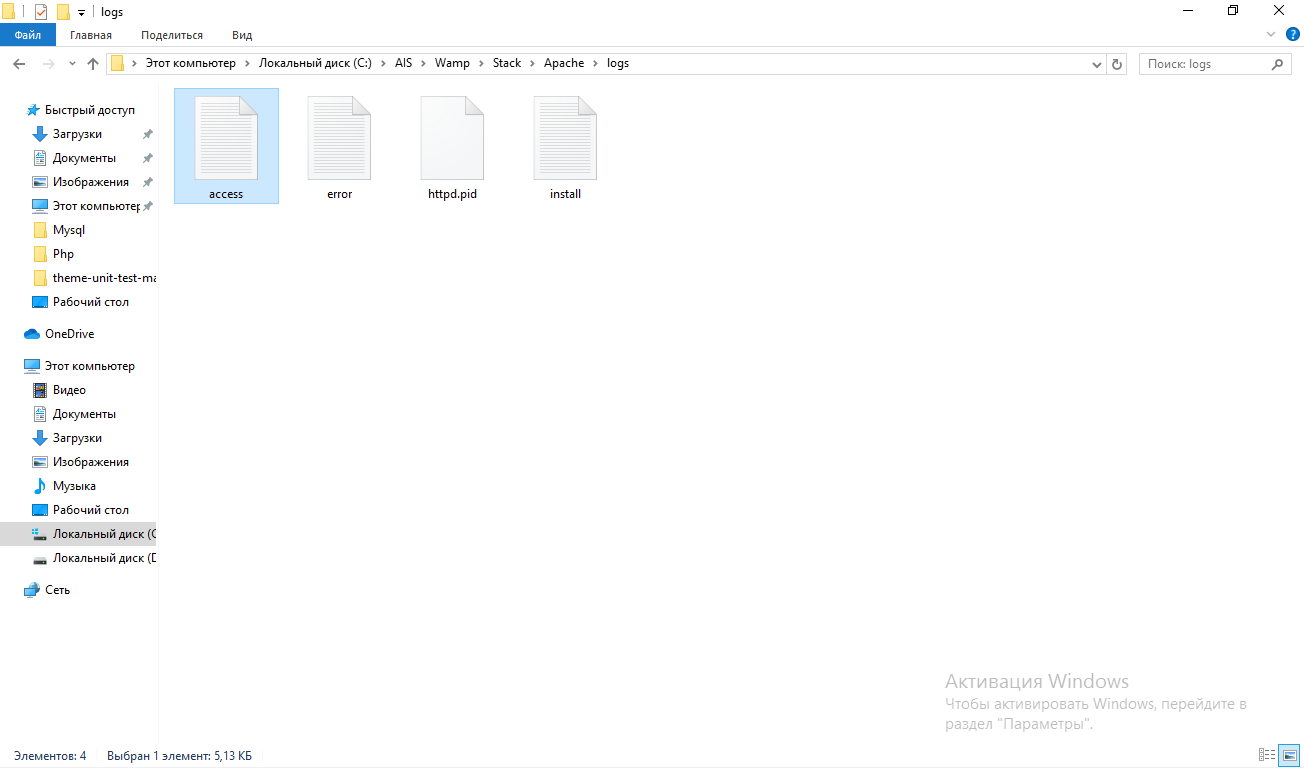
Переключаем тему на TwentySeventeen:

* Внешний вид->Темы
* Активируйте тему TwentySeventeen

После экспорта данных на сайте появилось наполнение: разделы, текст, изображения и т.п.



НАГРУЗОЧНОЕ ТЕСТИРОВАНИЕ САЙТАДля нагрузочного тестирования будем использовать JMeter. Для получения log-файла: остановим Apache, удалим существующий access.log файл, запустим Apache, создадим симуляцию активности на сайте, переходя по различным ссылкам, остановим Apache, скопируем access.log файл в новое место, чтобы он не затерся при перезапуске сервера, запустим Apache.  
В итоге у нас получился чистый, длинный лог-файл:



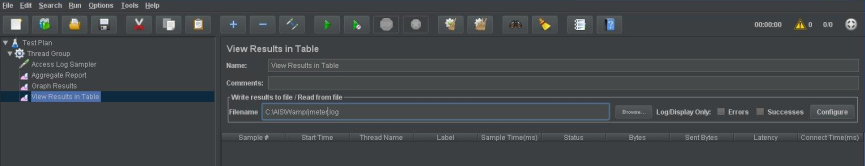
Запустив программу, слева видим — Test Plan. Добавим Thread group в Test plan. Правым кликом на Thread Group добавляем Access Log Sampler. Настраиваем адрес сайта, порт и путь к копии лога сервера Apache и добавляем в тест средства отображения:

Thread Group->Add->Listener->View Results in Table

Thread Group->Add->Listener->Graph Results

Thread Group->Add->Listener->Aggregate Report

Тест-план готов:



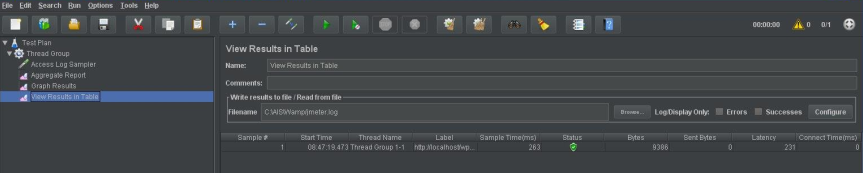
Приступаем к его тестированию и отладке.

File -> Save

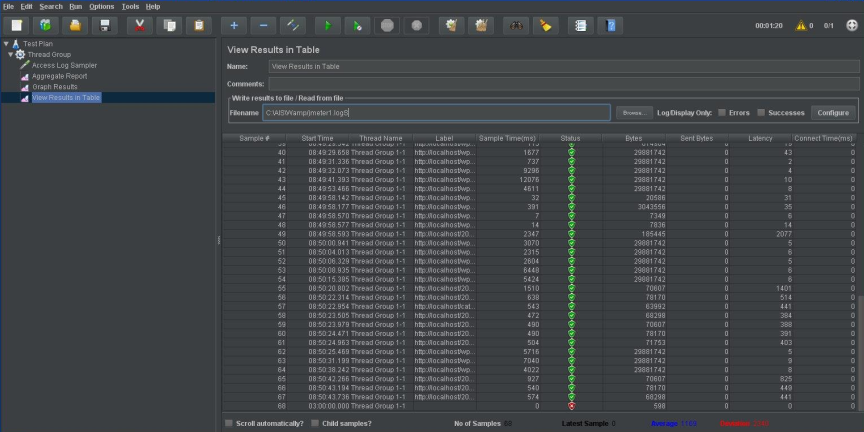
Run -> Clear All

Run-> Start

View Results in Table сообщает, что тест прошел без ошибок:

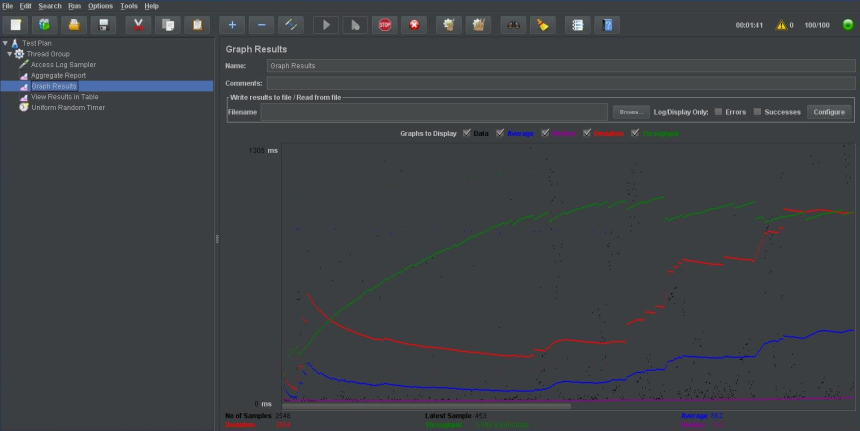


При установке Loop Count в infitie:



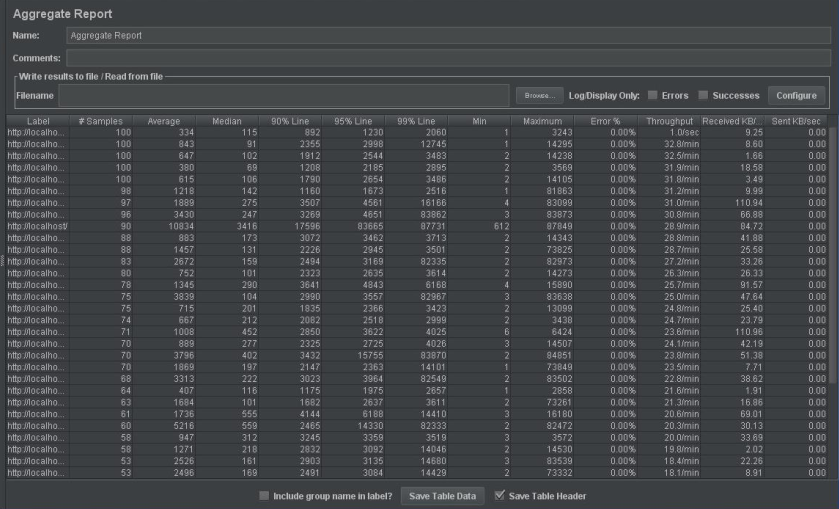
При установке количества юзеров в 100 со случайными интервалами между их запросами от 0 до 1000 миллисекунд получим:

GraphResults:

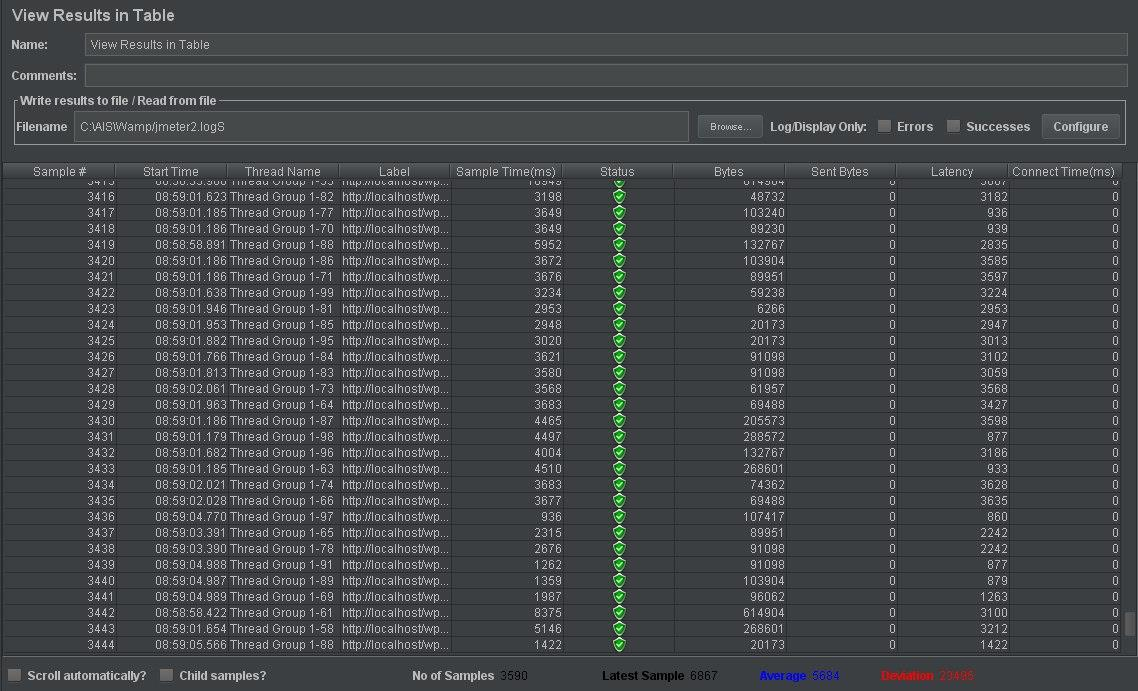


Значения предоставлены в миллисекундах.  
Data — время отклика каждой отдельной единицы данных т.е. каждого проверенного url.  
Average — усредненное время отклика, объективный график изменения нагрузки.  
Median — значение медианы.  
Deviation — погрешность, стандартное отклонение.  
Throughput — пропускная способность выполняемых запросов.  
Если усредненное время отклика (синий график) становится выше пропускной способности (зеленый график), сайт не справляется с нагрузкой. Из графика видно, что сайт с нагрузкой справляется и наблюдается адекватный рост времени отклика (синий график) по мере роста числа посетителей.

Aggregate Report отобразит статистику по каждому индивидуальному url отдельно:



View Result in Table отобразит результат в виде таблицы, здесь указано время, а также статус (успешно/не успешно):



На этом тестирование завершено. В ходе него мы проверили wordpress на нагрузку, и  
получили очень наглядные результаты. Сайт выдержал нагрузку в 100 пользователей в 100 секунд, т.к. запускался на достаточно мощном аппаратном обеспечении.

ВЫВОД  
В ходе выполнения данной лабораторной работы были:

* Освоена сборка стека WAMP (Windows+Apache+MySQL+PHP) из отдельных  
  компонентов.
* Изучены основные принципы развертывания web-приложений на примере  
  wordpress.
* Организовано нагрузочное тестирование сайта.