Шаг 1. Установка веб-сервера Apache и утилит для его работы.

Выполняем в терминале команду:

**sudo apt install apache2 apache2-utils**

После установки Apache должен автоматически запуститься, проверим его работу командой:

**systemctl status apache2**

Сделаем запуск Apache автоматическим при загрузке Debian:

**sudo systemctl enable apache2**

Теперь можете проверить открывается ли стартовая страница Apache на 80 порту, для этого в браузере введите IP-адрес компьютера, на котором установлен Debian, должна открыться стандартная веб-страница Apache:

Далее сделаем пользователя www-data (пользователя Apache) владельцем корневой директории, в которой у нас будут лежать файлы сайта:

**sudo chown www-data:www-data /var/www/html/ -R**

Шаг 2. Установка сервера баз данных MardiaDB (ответвление от системы управления базами данных MySQL)

Выполняем команду:

**sudo apt install mariadb-server mariadb-client**

Сделаем запуск MariaDB автоматическим при загрузке Debian:

**sudo systemctl enable mariadb**

Запустим мастер настройки безопасности:

**mysql\_secure\_installation**

MariaDB просит нас ввести пароль root:  
Enter current password for root (enter for none):  
Нажимаем Enter, так как текущий пароль root в MariaDB у нас пустой.

Далее вопрос:  
Switch to unix\_socket authentication [Y/n]  
Вводим Y, если мы хотим оставить возможность аутентификации через unix-сокет — это нововведение появилось именно в 10.4, плагин позволяет использовать присутствующие в системе учётные записи для подключения к СУБД при помощи локального unix-сокета. Делайте так, как вам удобней, лично я выбрал N.

Далее вопрос:  
Change the root password? [Y/n]  
Вводим Y, да мы хотим поменять пароль root и далее придумываем СЛОЖНЫЙ пароль root.

Далее вопрос:  
Remove anonymous users? [Y/n]  
Вводим Y, мы хотим удалить анонимного пользователя.

Далее вопрос:  
Disallow root login remotely? [Y/n]  
Вводим Y, да мы хотим запретить подключаться под пользователем root с удаленных машин.

Далее вопрос:  
Remove test database and access to it? [Y/n]  
Вводим Y, да мы хотим удалить тестовую базу test.

И последний вопрос:  
Reload privilege tables now? [Y/n]  
Вводим Y, да мы хотим перезагрузить таблицу привилегий для вступления их в силу.

После этого мы выполнили минимальные меры по защите нашего экземпляра MariaDB.

Шаг 3. Устанавливаем поддержку компонентов PHP

Выполняем команду:

**sudo apt install php libapache2-mod-php php-mysql php-common php-cli php-common php-json**

Узнаем версию установленного PHP:

**php --version**

В нашем случае установился PHP версии 7.4. **У вас 8.2 должна быть!!!**

Далее включаем поддержку PHP в Apache: sudo a2enmod php7.4

После этого нам необходимо перезапустить Apache: sudo systemctl restart apache2

Далее проверим работу PHP, для этого создадим PHP файл в корневой директории Apache:

**sudo nano /var/www/html/info.php**

В данный файл необходимо вставить следующий код:

**<?php phpinfo(); ?>**

Сохраняем файл комбинацией Ctrl+X и выходим из редактора.

Перейдем по IP-адресу вашего Debian и укажем путь к файлу info.php, проверим работоспособность PHP.

Шаг 4. Установка Moodle на Debian 12.

Загрузите Moodle из официального репозитория и установите правильные разрешения:

**Apt install git -y**

**cd /var/www/html**

**sudo git clone https://github.com/moodle/moodle.git**

Для Moodle требуется каталог для файлов курса:

**sudo mkdir /var/www/moodledata**

Правильные разрешения имеют решающее значение для безопасности и функциональности:

**sudo chown -R www-data:www-data /var/www/html/moodle**

**sudo chown -R www-data:www-data /var/www/moodledata**

**sudo chmod -R 755 /var/www/html/moodle**

**sudo chmod -R 777 /var/www/moodledata**

Шаг 5. Настройка MySQL для Moodle.

Сначала язарегистрировался в MySQL как пользователь root:

**sudo mysql -u root -p**

Создайте выделенную базу данных и пользователя для Moodle:

**CREATE DATABASE moodle CHARACTER SET utf8mb4 COLLATE utf8mb4\_unicode\_ci;**

**CREATE USER 'moodleuser'@'localhost' IDENTIFIED BY 'your-strong-password';**

**GRANT ALL PRIVILEGES ON moodle.\* TO 'moodleuser'@'localhost';**

**FLUSH PRIVILEGES;**

Шаг 6. Настройка Apache для Moodle.

Теперь нам надо изменить конфигурацию PHP, открываем файл:

**nano /etc/php/версия/apache2/php.ini**

И вносим следующие строки:

**extension=mysql.so  
extension=gd.so**

**max\_input\_vars = 5000**

Так же нужно изменить значения параметров:

**memory\_limit = 80M  
post\_max\_size = 80M  
upload\_max\_filesize = 80M**

Сохраняем изменения комбинацией Ctrl+X и перезапускаем Apache:

**/etc/init.d/apache2 restart**

Внесем некоторые изменения и в файл конфигурации MariaDB:

**nano /etc/mysql/my.cnf**

Добавляем строки:

**[client]  
default-character-set = utf8mb4  
[mysql]  
default-character-set = utf8mb4  
[mysqld]  
character-set-server = utf8mb4  
collation-server = utf8mb4\_unicode\_ci**

Сохраняем конфигурацию Ctrl+X и перезапускаем MariaDB:

**/etc/init.d/mariadb restart**

Далее нам нужно установить следующие пакеты:

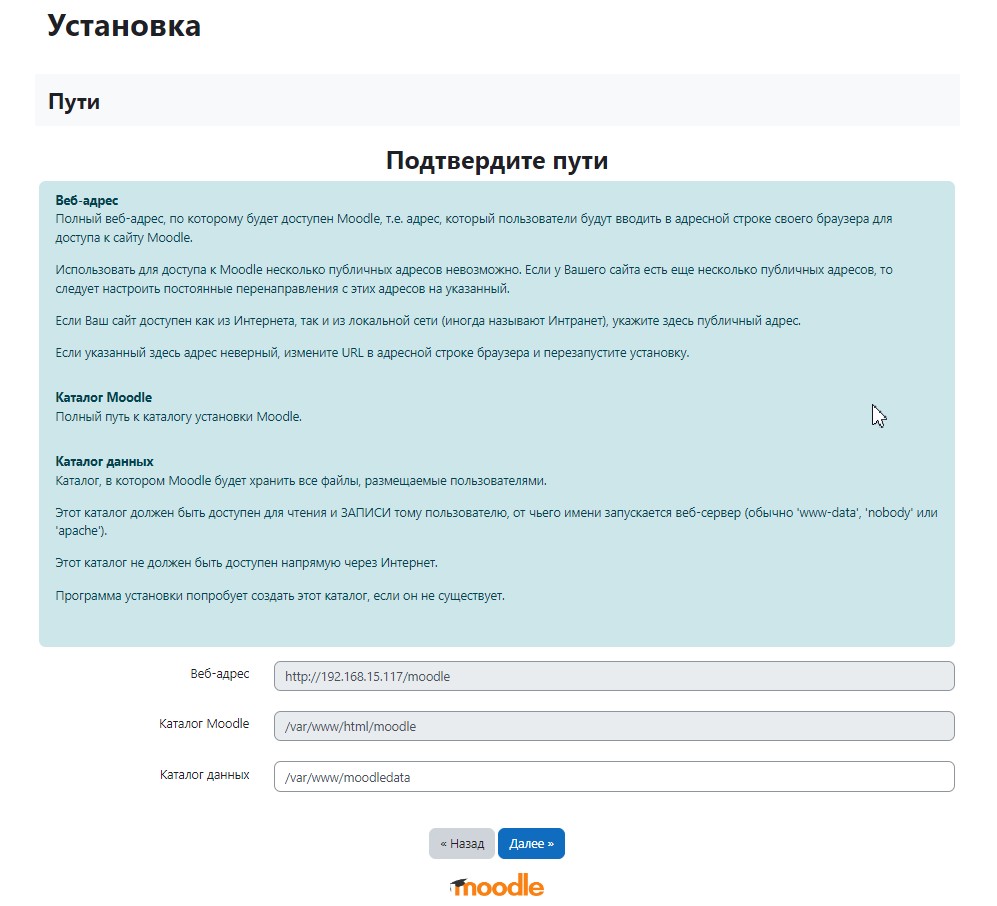
**sudo apt-get install php-curl php-zip php-xml**

Перезапускаем Apache:

**/etc/init.d/apache2 restart**

Переходим по адресу http://ваш\_IP/moodle/ и начинаем установку в веб-интерфейсе.

Переходим по адресу http://ваш\_IP/moodle/ и начинаем установку в веб-интерфейсе.



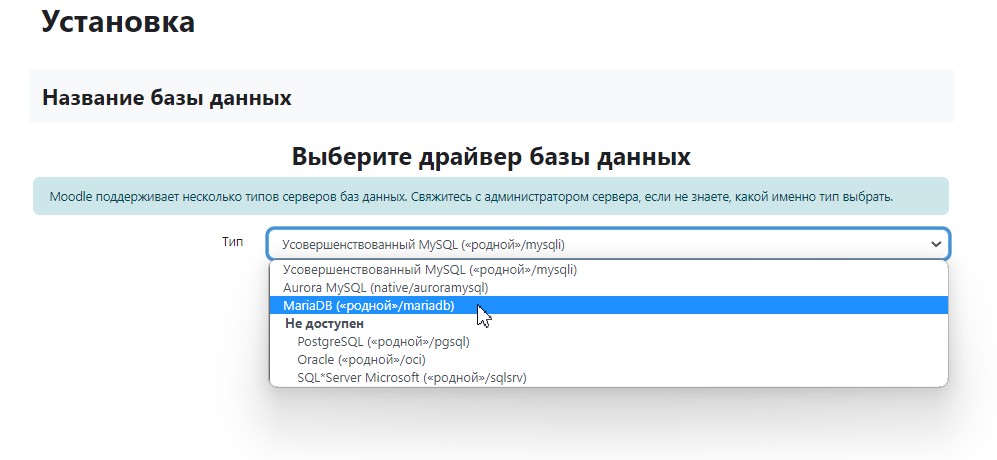
В процессе установки нам предложат указать путь к каталогу moodledata и предлагают разместить его в /var/www/moodledata. Можно оставить как есть или прописать свой собственный путь. Я оставлю предложенный по умолчанию вариант. Изначально Moodle попытается сам создать каталог moodledata, но скорее всего начнет ругаться, что директория www не доступна для записи. Облегчим ему задачу и создадим каталог за него:

**mkdir /var/www/moodledata**

Назначим пользователя Apache владельцем данной директории:

**sudo chown www-data:www-data /var/www/moodledata/ -R**

Теперь нам предстоит выбрать систему управления базами данных. Так как мы используем MariaDB, выбираем этот вариант:



Теперь нам предлагается указать параметры базы данных. Базу мы еще не создавали, самое время этим заняться!

Подключаемся к MariaDB и вводим пароль root: **ЭТО ДЕЛАЛОСЬ В ШАГЕ 5(ЕСЛИ ЧТО ПОВТОРИТЬ)**

**mysql -u root -p**

Создаем пользователя moodle для взаимодействия с базой данных (вместо password укажите свой надежный пароль):

**CREATE USER 'moodle'@'localhost' IDENTIFIED BY 'password';**

Создаем базу данных moodle:

**CREATE DATABASE moodle CHARACTER SET utf8 COLLATE utf8\_general\_ci;**

Предоставляем пользователю moodle все привилегии для работы с базой moodle:

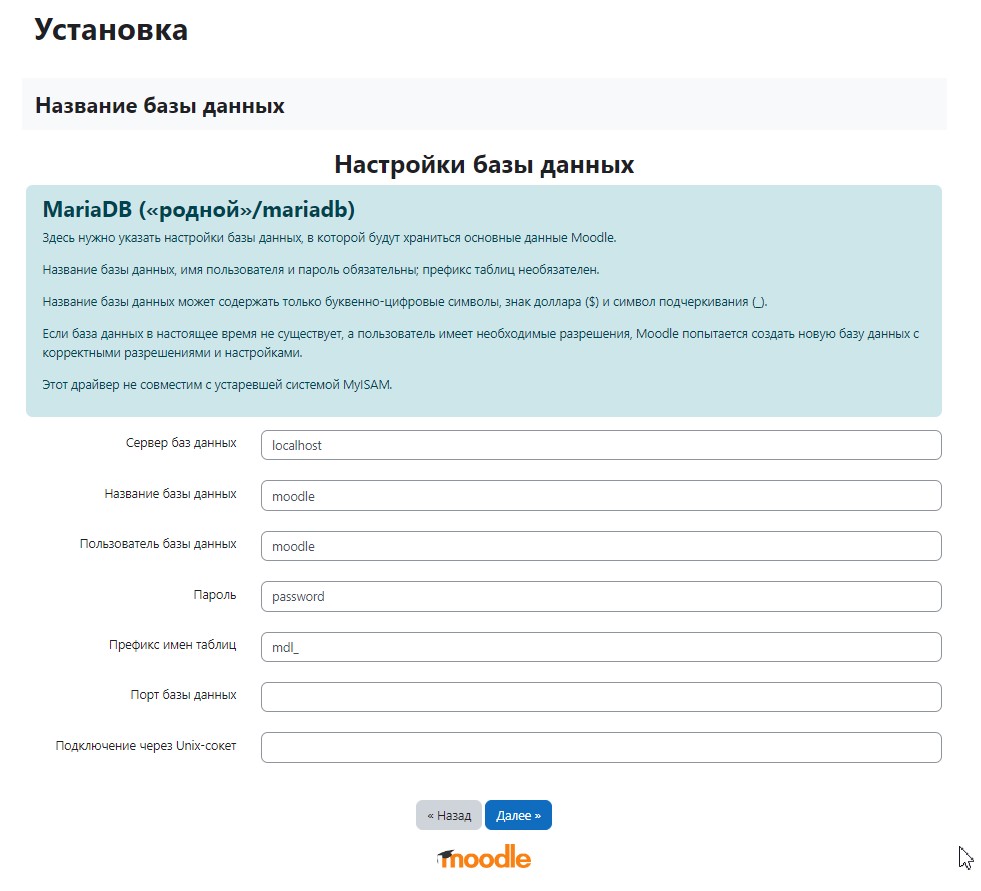
**GRANT ALL PRIVILEGES ON moodle.\* TO 'moodle'@'localhost';**

Обновляем привилегии:

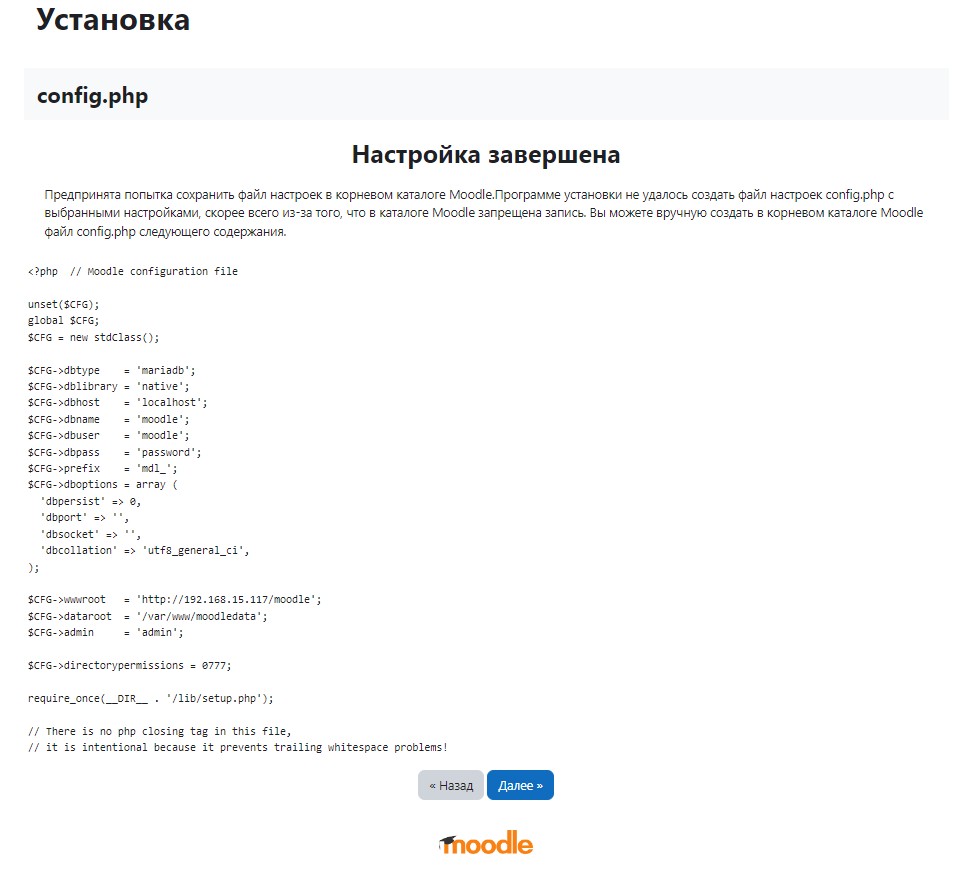
**FLUSH PRIVILEGES;**

Выходим из MariaDB – команда exit.

Заполняем параметры подключения к базе данных:



Программа установки уведомила нас, что у нее нет прав записи в корневой каталог html и она не смогла создать файл config.php:

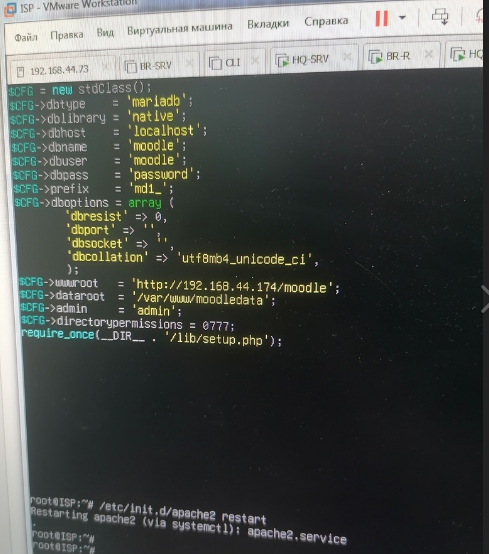


Не беда, поможем ей. Создадим файл:

**touch /var/www/html/moodle/config.php**

Откроем его в редакторе:

**nano /var/www/html/moodle/config.php**

И вставим туда тот текст, который выдал установщик. Для вставки скопированного текста в nano используем комбинацию Shift+Insert. Сохраняем файл Ctrl+X.  примерно так **ВАМ ПРИДЕТСЯ ПЕРЕПИСАТЬ ЕГО РУКАМИ(СМ РИС ВЫШЕ)**

После завершения установки Moodle я вместо страницы увидел сообщение: Moodle requires the mbstring PHP extension. Please install or enable the mbstring extension.

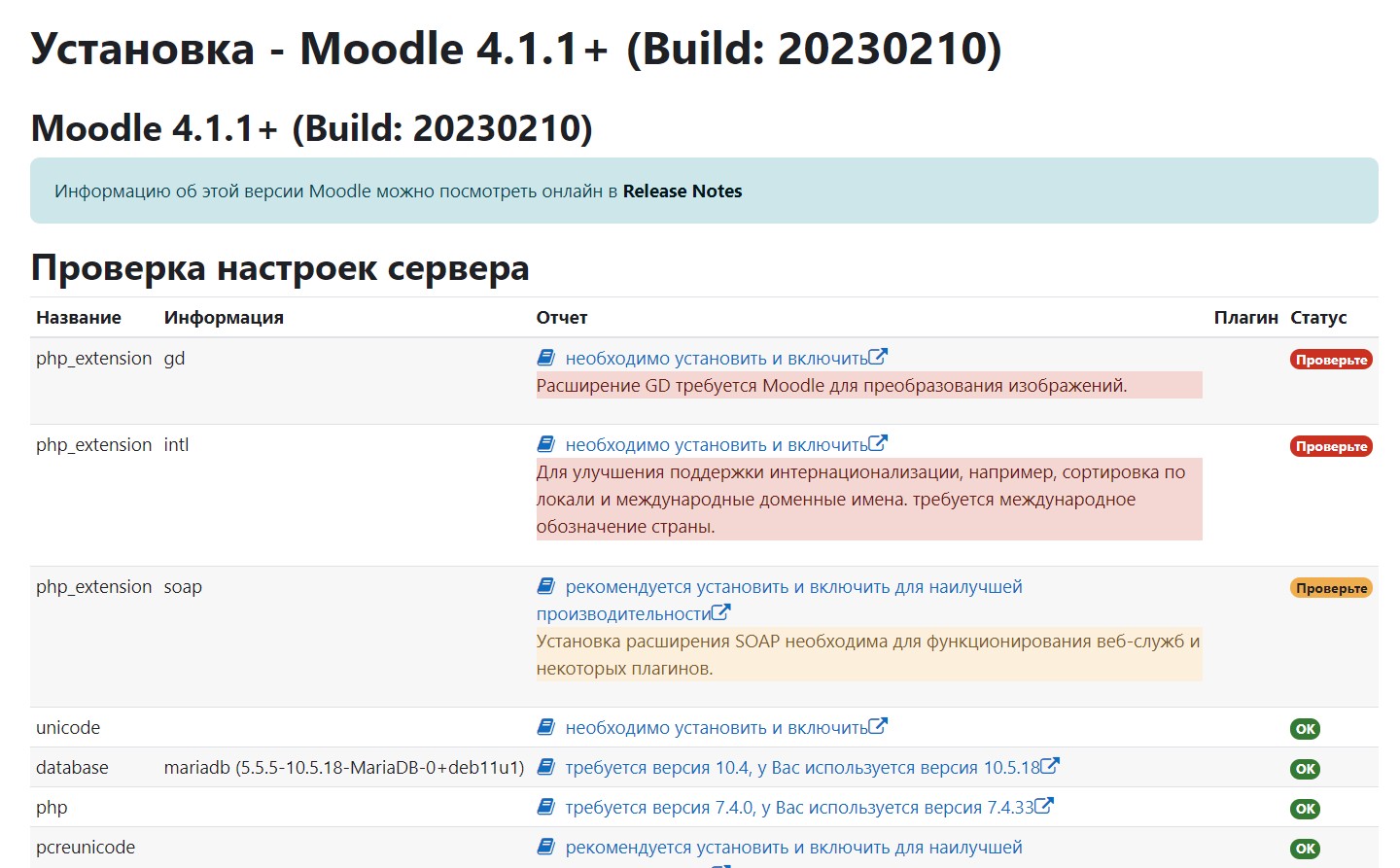
ОК, установим этот пакет:

**sudo apt-get install php-mbstring**

После обязательно перезагрузим Apache:

**/etc/init.d/apache2 restart**

Ура! Ошибка ушла и мы видим первую страницу диагностики Moodle:



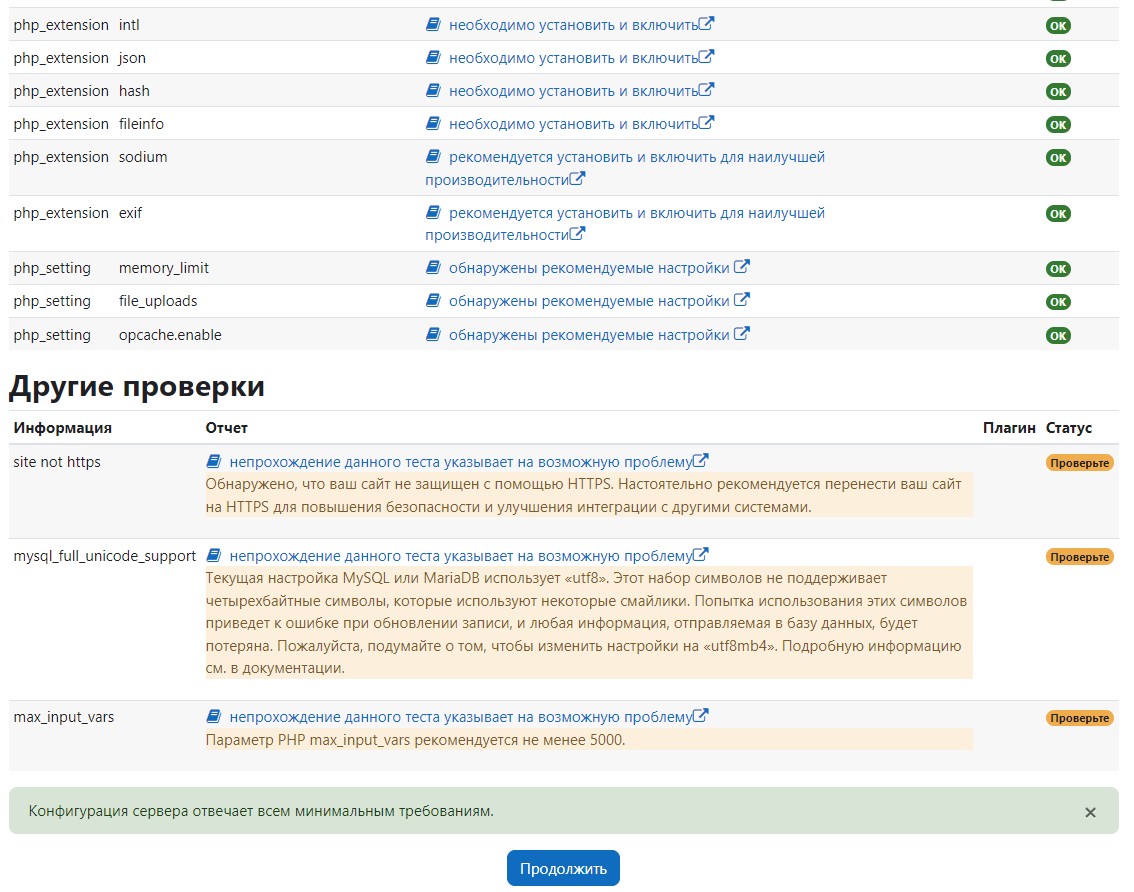
Следуя подсказкам на странице необходимо установить недостающие пакеты и внести иные коррективы, после чего страницу можно обновлять и проверять, чтобы никаких предупреждений больше не было. Интерфейс тут на русском, все подсказки написаны более менее понятно, думаю разберетесь.

Например, первое предупреждение с требованием включить расширение GD, решается командой:

**sudo apt-get install php-gd**

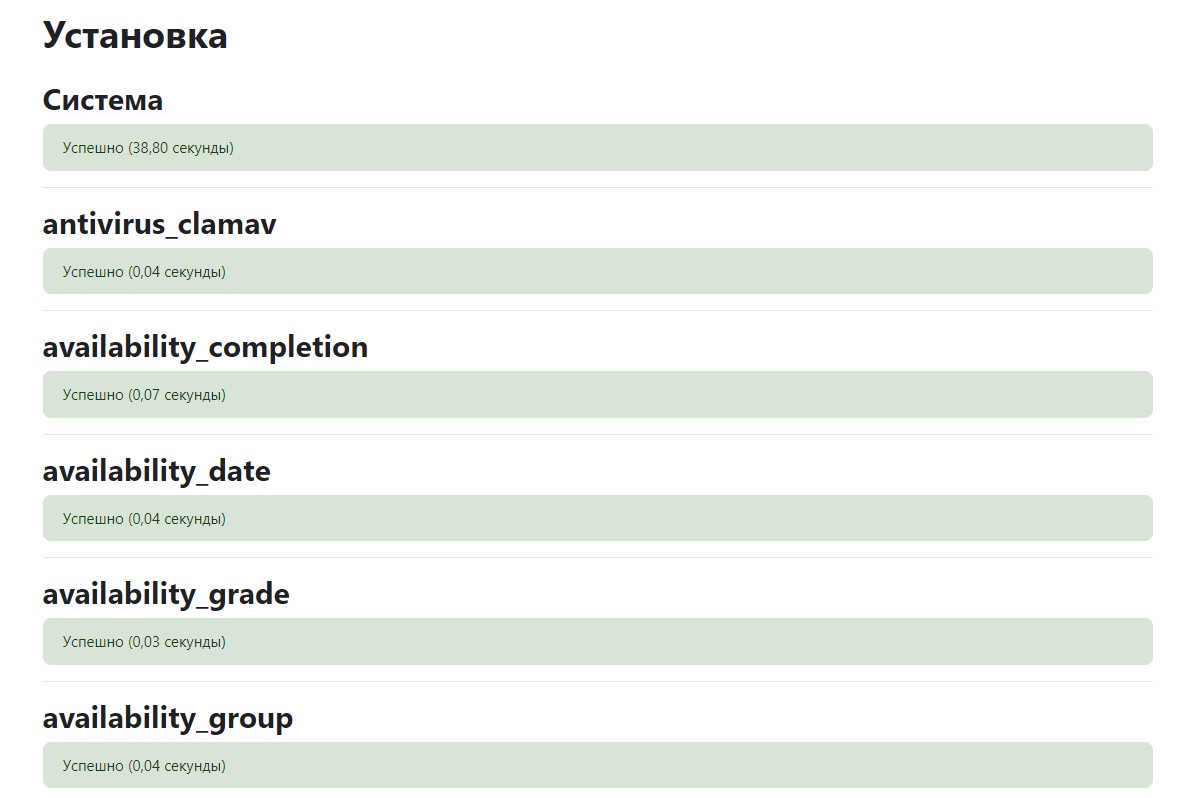
Информацию об этом я узнал когда перешел по ссылке в отчете данного предупреждения. Следуя подобным действием, произведите адаптацию сервера под Moodle. Не забывайте после каждой установки пакета или каких ни будь изменений перезапускать Apache, чтобы он смог обработать эти изменения.

После того как внизу появится зеленый прямоугольник с текстом: “Конфигурация сервера отвечает всем минимальным требованиям” вы можете нажать кнопку “Продолжить”.

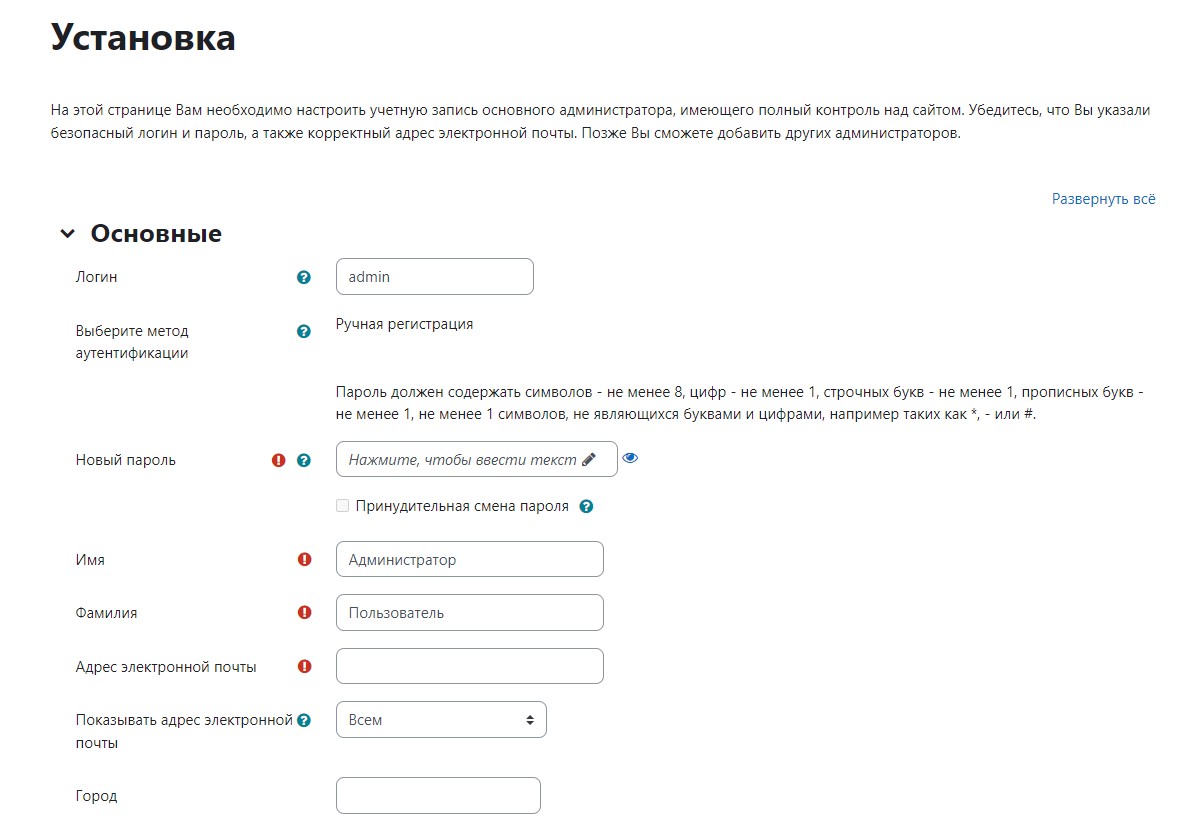


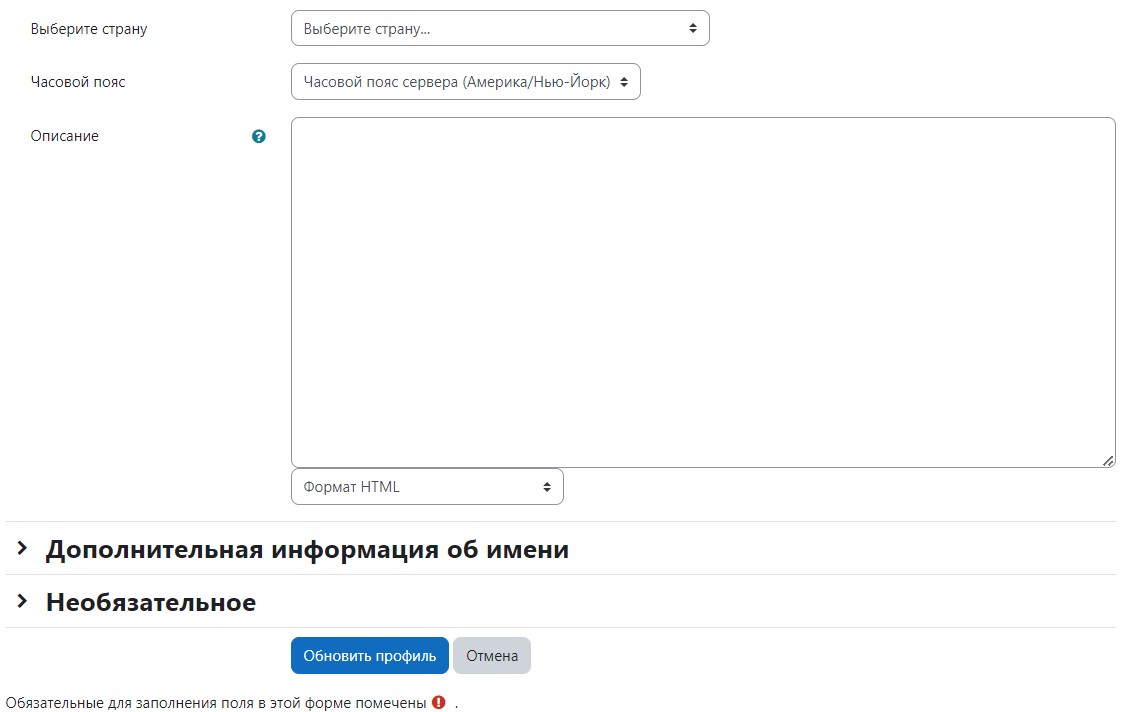
Начнется завершающий процесс установки, окно какое то время будет просто белым в состоянии долгой загрузки. Не перезагружайте и не закрывайте окно браузера, обязательно дождитесь завершения установки.

Когда установка завершится, мы увидим примерно такую страницу:



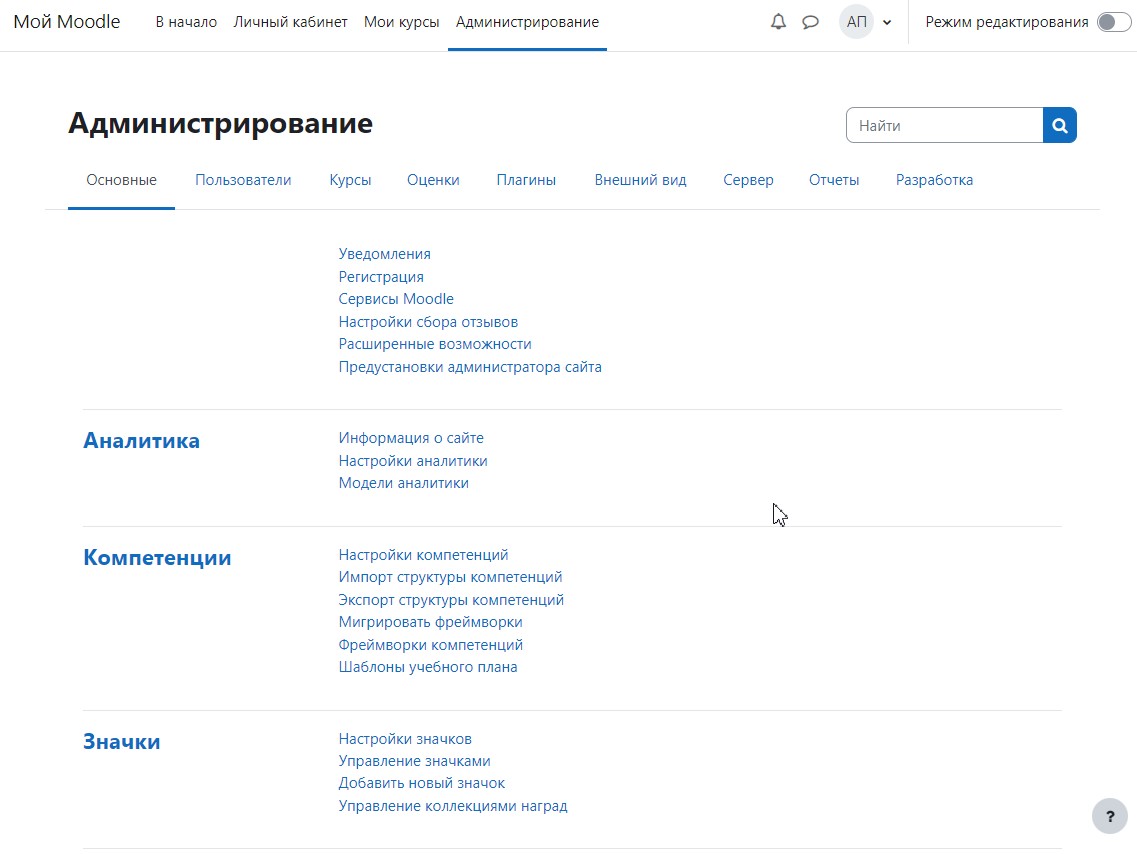
Нажимаем “Продолжить” и нам будет предложено создать администратора Moodle, указав его логин, пароль и прочие учетные данные:





Далее предлагается задать имя нашей системе Moodle, указать контакты службы поддержки и прочее. Заполняем и завершаем установку.

Так выглядит страница администратора:



Казалось бы все, установка окончена, наша система работает. Да, действительно так, но есть еще один момент, так сказать последний штрих. Нам необходимо настроить cron на автоматическое обслуживание Moodle. Запустим crontab от имени пользователя Apache: