# Junior PHP developer

*Начинающий php-разработчик*

## План обучения:

1. Повторить основы HTML и CSS - <http://htmlbook.ru/content/osnovy-html>
2. Основы PHP
   1. Пройти бесплатные курсы
      1. <https://php-start.com/course/php-start-theory>
      2. <https://php-up.com/lesson>  
         или <http://code-basics.ru/languages/php>
   2. [Сделать все практические задания](https://docs.google.com/document/d/1FHeyf5mIVXeAo1ucuH1kRp4mO8irgmcmQ1JOoL1kzvc/edit) и прислать на проверку на [hr@gootax.pro](mailto:hr@gootax.pro)
3. Стандарты кодирования
   1. Выучить стандарты кодирования PSR-0, PSR-1, PSR-2 ([перевод](https://svyatoslav.biz/misc/psr_translation/), [оригинал](https://www.php-fig.org/psr/))
   2. Выучить - <https://habrahabr.ru/company/mailru/blog/336788/>
   3. Встреча в офисе и сдача экзамена на знания стандартов кодирования
4. Познакомиться с IDE PhpStorm и настроить к работе
   1. [Quick Start](https://www.jetbrains.com/phpstorm/quickstart/), [видео-уроки](https://www.jetbrains.com/phpstorm/documentation/phpstorm-video-tutorials.jsp)
   2. [Настройте PhpStorm для работы с yii 2.0](http://yiiframework.ru/doc/cookbook/ru/ide)
   3. [Настройте PhpStorm для работы с PSR](http://laraveldaily.com/how-to-configure-phpstorm-for-psr-2/)
5. Изучить Yii 2.0
   1. [Пройти видео обучение](https://www.youtube.com/playlist?list=PLa9lO_Eq-vZWucOH1auan_m-POa3n7fV6)
   2. [Прочитать руководство пользования yii 2.0](https://www.yiiframework.com/doc/guide/2.0/ru)
   3. Научиться работать с [API yii2](https://www.yiiframework.com/doc/api/2.0)
   4. Выполнить практическое задание №1 (см. ниже) и прислать на проверку на [hr@gootax.pro](mailto:hr@gootax.pro)
6. [Познакомиться с популярными атаками на веб-приложения](https://habrahabr.ru/company/ua-hosting/blog/272205/)
7. Репозитории кода и системы контроля версий. Bitbucket + Mercurial.  
   Команды: pull, push, merge
   1. Mercurial ([1](http://habrahabr.ru/post/108443/), [2](http://habrahabr.ru/post/108658/), [3](http://habrahabr.ru/post/108904/), [4](http://habrahabr.ru/post/109074/), [5](http://habrahabr.ru/post/109203/), [6](http://habrahabr.ru/post/109428/))
   2. [Удачная модель ветвления](https://habrahabr.ru/post/106912/)
8. Повторить или изучить основы JS
   1. [Сделать практические задания](https://docs.google.com/document/d/1AYjxxmhMzbMPI6Cia-yziw1DDu8GIpyJOuIzE9A-WlM/edit#)
9. Выполнить финальное аттестационное задание (см. ниже) и прислать на проверку на [hr@gootax.pro](mailto:hr@gootax.pro)

## Практическое задание №1

Сделать блог по шагам <http://www.yiiframework.com/doc/blog/1.1/ru/start.overview> , но ипользуя yii 2.0 и PHP 7.2

* 1. Создать проект в репозитории  
     Использовать систему контроля версий git или mercurial
  2. После завершения каждой осмысленной части, нужно делать commit в репозиторий, чтобы мы видели как у вас шёл процесс разработки
  3. Подготовить миграции
  4. После выполнения задания прислать нам ссылку на проект в репозитории

**На заметку:**

* Работаем только с последней версией yii 2.0
* Работаем только с последней версией PHP (на текущий момент - 7.2)
* Nginx, Apache, Xdebug - последние стабильные версии
* Найти ответ или задать вопрос можно на форуме <http://yiiframework.ru/forum/>

## Финальное аттестационное задание

* На базе yii framework 2.0 создать модули: Города, Отзывы, Пользователи.
* Таблица Города (id, name, date\_create)
* Таблица Отзывы (id, id\_city, title, text, rating, img, id\_autor, date\_create)
* Таблица Пользователи (Авторы) (id, fio, email, phone, date\_create, password)
* Для верстки использовать <http://bootstrap-3.ru/index.php>

Мы заходим на сайт видим страницу с приветствием. Если мы можем определить город пользователя по IP, то спрашиваем: “Ижевск” ваш город? Да/Нет.  
Если не можем определить, то показываем список всех городов (для которых созданы отзывы) с сортировкой от А к Я.  
  
После выбора города в сессии запоминается выбор - время сессии 2 часа (и 2 часа нам сразу показываются отзывы выбранного города). После сессия очищается и опять сайт просит пользователя выбрать город.  
  
Выбираем город и видим тексты отзывов (относящиеся к этому городу), автора кто оставил отзыв, но не видим контакты автора, не можем создать отзыв и редактировать свой отзыв.

Отзывы могут создавать и редактировать только авторизованные пользователи.

При создании отзыва мы указываем поля:

* Название отзыва (до 100 символов)
* Текст отзыва (до 255 символов)
* Рейтинг (целое число от 1 до 5)
* Можем загрузить фото к отзыву (например, скан-копию отзыва на бумаге)
* Город выбираем из автокомплита, с возможностью выбрать несколько городов или не выбирать вовсе, тогда отзыв относится ко всем городам сразу

Авторизованный пользователь видит:

* Текст отзыва
* ФИО автора  
  и когда нажимает на ФИО автора, то всплывает окно, где мы видим email и телефон автора, а также ссылку посмотреть все отзывы автора. Нажимая на ссылку, видим страницу со всеми отзывами автора по всем города (но с пометкой какой отзыв в какому городу относится или ко всем городам)

Для создания и редактирования отзывов нам надо зарегистрироваться. При регистрации указываем поля:

* ФИО
* Email
* Телефон
* Придумываем пароль
* Повторяем пароль
* Код из каптчи

Пароль нельзя хранить в БД в открытом виде, нужно хранить его безопасно.

Далее нужно подтвердить введенный email через отправку на указанный email контрольной строки.  
После этого только можно авторизоваться.

Если зарегистрируемся и авторизуемся, то можем редактировать и добавлять отзывы.

Если при создании отзыва в нашей базе нет города, который хочет указать клиент, то надо через стороннее АПИ подгрузить город и сохранить в БД городов.

Прелоадер на создание, удаление, редактирование во время работы ajax.

**Дополнительно:**

* Использовать стандартные способы yii кеширования обращений в БД.
* Всю разработку вести через репозиторий и систему контроля версий. После завершения какой осмысленной части, нужно делать commit в репозиторий, чтобы мы видели как шёл процесс разработки
* Подготовить миграции
* После выполнения прислать нам ссылку на проект в репозитории

## Итог обучения

Начинающий php-разработчик хорошо разбирается в теории. Под руководством более опытных разработчиков может самостоятельно писать код без ошибок. Умеет самостоятельно находить свои ошибки и решать простые и средние задачи.

# Middle PHP developer

Опытный php-разработчик

## Задачи обучения:

1. Понять разницу между Nginx и Apache
   1. Научиться устанавливать и настраивать
2. Изучить Xdebug
   1. Установить и использовать
3. Прочитать все статьи <http://ruhighload.com/> и всё попробовать
4. [Читайте блог Дмитрия](http://www.elisdn.ru/blog)
5. Укрепление навыков программирования
   1. Сделать практические задания
6. Изучение нюансов php
   1. Какой код быстрее работает
7. Автоматизированное тестирование - PhpUnit. [основы](http://habrahabr.ru/post/56289/), [перевод 8 глав документации](http://www.smartyit.ru/php)
8. Детальное изучение безопасности веб-приложений
   1. SQL-инъекции
   2. XSS
   3. target=\_blank
   4. и другие
9. Изучение MySQL
   1. Написание сложных запросов
10. NoSQL
    1. MongoDb
    2. Redis
11. Кеширование
    1. Memcached
    2. Redis
12. Composer. [Основы](http://habrahabr.ru/post/258891/), [свой vendor](http://habrahabr.ru/post/248055/)
13. Выучить [TOP100-300](https://gist.github.com/jaytaph/807dda9b3d307702dc37) функций PHP
14. Прочитать книжки (по неделе на книжку, читая вечерами и на выходных):
    1. PHP+MySQL — [ozon.ru](http://www.ozon.ru/context/detail/id/31351020/)
    2. ООП и паттерны — [ozon.ru](http://www.ozon.ru/context/detail/id/5648968/)
    3. Совершенный код — [ozon.ru](http://www.ozon.ru/context/detail/id/5508646/)

Можно заменить на аналогичные, почти всё можно [скачать без смс](http://rutracker.org/forum/viewforum.php?f=1425).

1. Работа с очередями: RabbitMq
2. Совершенствовать свой английский, хотя бы на уровне чтения.