**Installation**

1. На первом шаге, нам нужно установить VirtualBox, Vagrant и Homestead:  
   <https://laravel.com/docs/6.x/homestead#first-steps>

**Time**: 2 hours.

**Быстрый старт**

1. Изучаем основные элементы Laravel. Нужно для понимания возможности фреймворка: <https://laravel.ru/docs/v5/quickstart>

**Time**: 1 hour.

1. Более глубокое понимание для практического использования: <https://laravel.ru/docs/v5/quickstart-intermediate>

**Time**: 2 hours.

**Маршрутизация**

1. простейшая маршрутизация: [link](https://laravel.ru/docs/v5/routing#%D0%BF%D1%80%D0%BE%D1%81%D1%82%D0%B5%D0%B9%D1%88%D0%B0%D1%8F);
2. параметры маршрутов: [link](https://laravel.ru/docs/v5/routing#%D0%BF%D0%B0%D1%80%D0%B0%D0%BC%D0%B5%D1%82%D1%80%D1%8B);
3. именованные маршруты: [link](https://laravel.ru/docs/v5/routing#%D0%B8%D0%BC%D0%B5%D0%BD%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5);
4. middleware:

* что такое middleware: [link](https://laravel.ru/docs/v5/middleware);
* применение в routes: [link](https://laravel.ru/docs/v5/routing#%D0%BF%D0%BE%D1%81%D1%80%D0%B5%D0%B4%D0%BD%D0%B8%D0%BA%D0%B8);

1. префиксы: [link](https://laravel.ru/docs/v5/routing#%D0%BF%D1%80%D0%B5%D1%84%D0%B8%D0%BA%D1%81%D1%8B);
2. получение текущего маршрута: [link](https://laravel.ru/docs/v5/routing#%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5);
3. ошибки 404: [link](https://laravel.ru/docs/v5/routing#404);
4. связь роутеров и контроллеров: [link](https://laracasts.com/series/laravel-6-from-scratch/episodes/8).

**Time**: 3 hours.

**Контроллеры**

1. определение контроллеров: [link](https://laravel.ru/docs/v5/controllers#%D0%BE%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5);
2. middleware для контроллеров: [link](https://laravel.ru/docs/v5/controllers#%D0%BF%D0%BE%D1%81%D1%80%D0%B5%D0%B4%D0%BD%D0%B8%D0%BA);
3. контроллеры ресурсов: [link](https://laravel.ru/docs/v5/controllers#%D1%80%D0%B5%D1%81%D1%83%D1%80%D1%81);
4. внедрение зависимостей и контроллеры: [link](https://laravel.ru/docs/v5/controllers#%D0%B2%D0%BD%D0%B5%D0%B4%D1%80%D0%B5%D0%BD%D0%B8%D0%B5);

**Time**: 2 hours.

**Requests**

1. получение экземпляра запроса: [link](https://laravel.ru/docs/v5/requests#%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5_%D1%8D%D0%BA%D0%B7%D0%B5%D0%BC%D0%BF%D0%BB%D1%8F%D1%80%D0%B0);
2. методы и путь запроса: [link](https://laravel.ru/docs/v5/requests#%D0%BC%D0%B5%D1%82%D0%BE%D0%B4%D1%8B);
3. получение данных ввода: [link](https://laravel.ru/docs/v5/requests#%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B2%D0%B2%D0%BE%D0%B4%D0%B0);
4. старый ввод: [link](https://laravel.ru/docs/v5/requests#%D1%81%D1%82%D0%B0%D1%80%D1%8B%D0%B9);
5. cookies: [link](https://laravel.ru/docs/v5/requests#cookies);
6. получение загруженных файлов: [link](https://laravel.ru/docs/v5/requests#%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B6%D0%B5%D0%BD%D0%BD%D1%8B%D1%85);
7. хранение загруженных файлов: [link](https://laravel.ru/docs/v5/requests#%D1%85%D1%80%D0%B0%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5);

**Time**: 2 hours.

**Сессии**

1. использование сессий: [link](https://laravel.su/docs/5.4/session#using-the-session);
2. хранение данных: [link](https://laravel.su/docs/5.4/session#storing-data);
3. флеш данные: [link](https://laravel.su/docs/5.4/session#flash-data);
4. удаление данных: [link](https://laravel.su/docs/5.4/session#deleting-data);
5. обновление ID сессии: [link](https://laravel.su/docs/5.4/session#regenerating-the-session-id);

**Time**: 1 hour.

**Валидация**

1. написание логики валидации в контроллере: [link](https://laravel.su/docs/5.4/validation#quick-creating-the-controller);
2. создание Form Requests: [link](https://laravel.com/docs/6.x/validation#creating-form-requests);
3. валидационные правила. Их очень много: начиная с accepted и заканчивая URL: [link](https://laravel.su/docs/5.4/validation#rule-accepted);
4. отображение ошибок валидации: [link](https://laravel.su/docs/5.4/validation#quick-displaying-the-validation-errors);

**Time**: 3 hours.

**Responses**

1. создание responses: [link](https://laravel.ru/docs/v5/responses#%D1%81%D0%BE%D0%B7%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5);
2. view response: [link](https://laravel.ru/docs/v5/responses#%D0%B4%D1%80%D1%83%D0%B3%D0%B8%D0%B5);
3. json response: [link](https://laravel.ru/docs/v5/responses#json-%D0%BE%D1%82%D0%BA%D0%BB%D0%B8%D0%BA%D0%B8);
4. file response: [link](https://laravel.ru/docs/v5/responses#%D0%BE%D1%82%D0%BA%D0%BB%D0%B8%D0%BA);

**Time**: 2 hours.

**Http-исключения и логирование**

1. логирование: [link](https://laravel.su/docs/5.4/errors#logging);
2. http-исключения: [link](https://laravel.su/docs/5.4/errors#http-exceptions);

**Time**: 1 hour.

**Views**

1. создание представлений: [link](https://laravel.ru/docs/v5/views#%D1%81%D0%BE%D0%B7%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5);
2. передача данных в представление: [link](https://laravel.ru/docs/v5/views#%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D1%87%D0%B0);

**Time**: 1 hour.

**Миграции**

1. что такое миграции: [link](https://laravel.su/docs/5.4/migrations#introduction);
2. создание миграции: [link](https://laravel.su/docs/5.4/migrations#generating-migrations);
3. структура миграции: [link](https://laravel.su/docs/5.4/migrations#migration-structure);
4. выполнение и откат миграции: [link](https://laravel.su/docs/5.4/migrations#running-migrations);
5. создание таблицы: [link](https://laravel.su/docs/5.4/migrations#creating-tables);
6. удаление таблицы: [link](https://laravel.su/docs/5.4/migrations#renaming-and-dropping-tables);
7. создание столбцов: [link](https://laravel.su/docs/5.4/migrations#creating-columns);

**Time**: 2 hours.

**Базы данных**

1. настройка: [link](https://laravel.su/docs/5.4/database#configuration);
2. выполнение сырых запросов: [link](https://laravel.su/docs/5.4/database#running-queries);
3. конструктор запросов (полностью): [link](https://laravel.su/docs/5.4/queries);
4. страничный вывод из базы данных: [link](https://laravel.su/docs/5.4/pagination#paginating-query-builder-results);

**Time**: 4 hours.

**Eloquent**

1. работа с моделью (полностью): [link](https://laravel.su/docs/5.4/eloquent);
2. отношения: [link](https://laravel.su/docs/5.4/eloquent-relationships);

**Time**: 7 hours.

**Задание**

1) Дано:

· Модель User без кастомных полей.

· Модель Article с полем text.

· Относятся как многие ко многим.

· Написать свойство articles в первой модели, которое вернёт все статьи пользователя.

· Написать метод users во второй, который вернёт всех авторов статьи.

2) Работа с формами и запросами:

· Расширить модель User, чтобы она имела поля, соответствующие форме index.html.

· Доработать приложение, чтобы данные с этой формы создавали объект User в базе данных на сервере. После сохранения данных на сервере происходит редирект на точно такую-же страницу, где данные уже введены и их нельзя изменить.

3) Мягкое удаление и миграции:

· Добавить возможность удалять статьи в мягком режиме . При этом сделать возможность обновить систему со старой версии (из первого задания) на новую без полного сброса БД.

4) Методы и строгая типизация:

· Теперь надо создать функцию в классе статей, которая: принимает пользователя и только пользователя – выдает mismatch в противном случае, возвращает  истину, если указанный человек автор статьи,  ложь – если это не так – null если статья удалена. Если функция пытается вернуть что – то иное – mismatch.

5) ORM :

· Добавим новое поле пользователю user – experience  Нужно создать функцию, в которой будет извлекаться пользователь и сохраняться в переменную $user = User::find(1). Дальше функция выводит experience. Параллельно с работой функции асинхронный метод меняет опыт на случайное число каждые несколько секунд.  В первой функции еще раз выводится спустя промежуток времени опыт пользователя.  Каким будет этот вывод?

**Time**: 7 hours.