

## РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ЗАПИСИ НА РЕМОНТ

Мальчуков П.И., студент гр. ИТ-162, IV курс  
Сыркин И.С., Доцент кафедры ИиАПС КузГТУ  
Кузбасский государственный технический университет имени Т.Ф.Горбачева  
г. Кемерово

Буквально каждая компания стремится увеличивать получаемую прибыль и сокращать расходы. И путей достижения этой цели абсолютное множество, повышение цен, экономия на материалах, зарплатах рабочих и прочее, но далеко не каждое предприятие выбирает путь автоматизации целевых процессов и сокращения временных и финансовых затрат на их выполнение. Однако именно такое направление развития было выбрано руководством СТО «Автория». В качестве автоматизируемого процесса был определён процесс предпродажного взаимодействия с клиентами.

По итогу передо мной встала задача о разработке веб-системы для записи на ремонт авто с дополнительным функционалом позволяющим увеличить LTV (life-time value – прибыль за всё время взаимодействия с клиентом) предприятия.

В качестве минимальных требований были выдвинуты следующие пункты:

- Возможность записи на осмотр
- Возможность записи на ремонт
- Выбор проблемного узла авто при записи
- Выбор удобного СТО на карте при записи
- Регистрация
- Авторизация
- Добавление автомобиля в профиль
- Развёртываемая панель с информацией об автомобиле
- Связь сервиса с БД для хранения данных

А также наличие административной панели для внутреннего управления веб-сервисом и модерирования данных клиентов сотрудником компании

В качестве программного обеспечения для разработки описанной веб-системы было принято решение использовать IDE PhpStorm и OS Panel, включая консоль и систему взаимодействия с базами данных PhpMyAdmin. А также, несмотря на меньшую профессиональную адаптированность, было решено использовать Microsoft Windows 10 в связи с большей надёжностью в сравнении с, казалось бы, более подходящей операционной системой Linux Ubuntu.

Язык программирования для разработки был выбран Php laravel - бесплатный веб-фреймворк с открытым кодом, оптимизированный под разработку

ку веб-приложению на основе архитектурной модели MVC (модель-представление-контроллер). Laravel был издан под лицензией MIT.

В декабре 2013 года Laravel занял место самого многообещающего проекта на 2014 год по версии сайта [sitepoint.com](http://sitepoint.com).

В 2015 году в результате опроса пользователей о используемых PHP-фреймворках на сайте [sitepoint.com](http://sitepoint.com) Laravel занял первое место в следующих номинациях:

- Фреймворк корпоративного уровня
- Фреймворк для личных проектов
- Кладёзь антипаттернов

Ключевые особенности архитектуры Laravel, по большей части, основываясь на которых, я принял решение использовать данный фреймворк:

- Использование пакетов (англ. *packages*), позволяющих создавать и подключать модули в формате Composer к приложению на Laravel. Многие дополнительные возможности уже доступны в виде таких модулей.

- Eloquent ORM — реализация шаблона проектирования ActiveRecord на PHP. Позволяет строго определить отношения между объектами базы данных.

- Удобная логика приложения.

- Обратная маршрутизация, связывающая между собой генерируемые приложением ссылки и маршруты, что позволяет изменять последние с автоматическим обновлением связанных ссылок. При задании ссылок с помощью именованных маршрутов, Laravel автоматически добавляет конечные URL.

- REST-контроллеры, представляющие из себя дополнительный слой разделения логики обработки GET- и POST-запросов HTTP.

- Автозагрузка классов — механизм автоматической загрузки классов PHP без необходимости подключать файлы их определений в `include`. Загрузка по требованию предотвращает загрузку ненужных компонентов; загружаются только те из них, которые действительно используются.

- Составители представлений (англ. *view composers*) — блоки кода, облегчающие развёртывание UX составляющей веб-страниц.

- Инверсия управления (англ. *Inversion of Control*) — обеспечивает получение экземпляров объектов методом обратного управления.

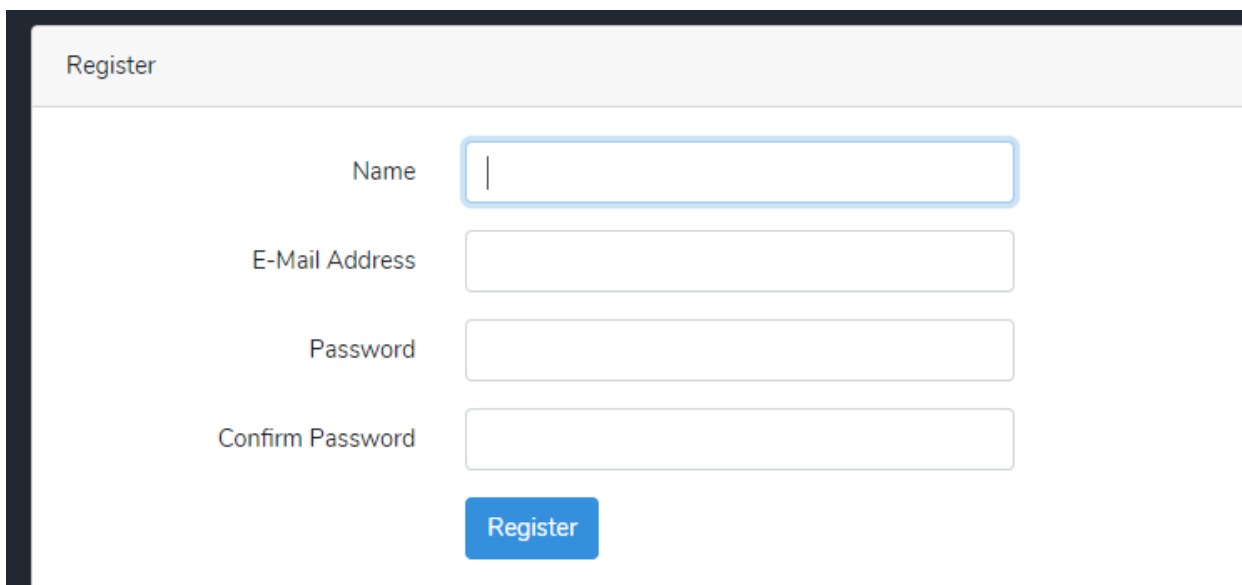
- Миграции — система управления версиями для баз данных. Обеспечивает связь изменений в коде приложения со структурными изменениями баз данных, что существенно упрощает развёртывание и обновление приложения.

- Модульное тестирование (юнит-тесты) — передовой способ необходимого тестирования, предотвращающего регрессии.

- Страничный вывод — функционал упрощающий генерацию страниц путём использования базового функционала, встроенного в Laravel.

После получения полного списка требований была организована вёрстка следующих каркасов:

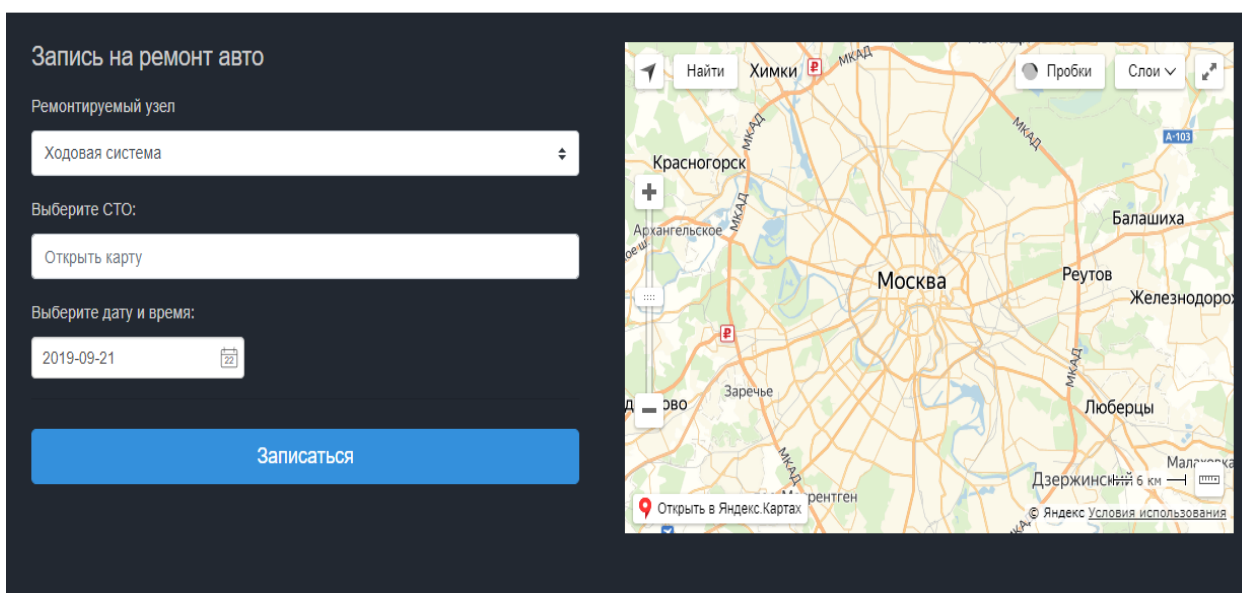
Форма для регистрация представлена на рисунке 1.



The image shows a web form titled "Register". It contains four input fields: "Name", "E-Mail Address", "Password", and "Confirm Password". Below these fields is a blue button labeled "Register". The "Name" field is currently active, indicated by a blue border and a vertical cursor.

*Рис. 1. Форма регистрации*

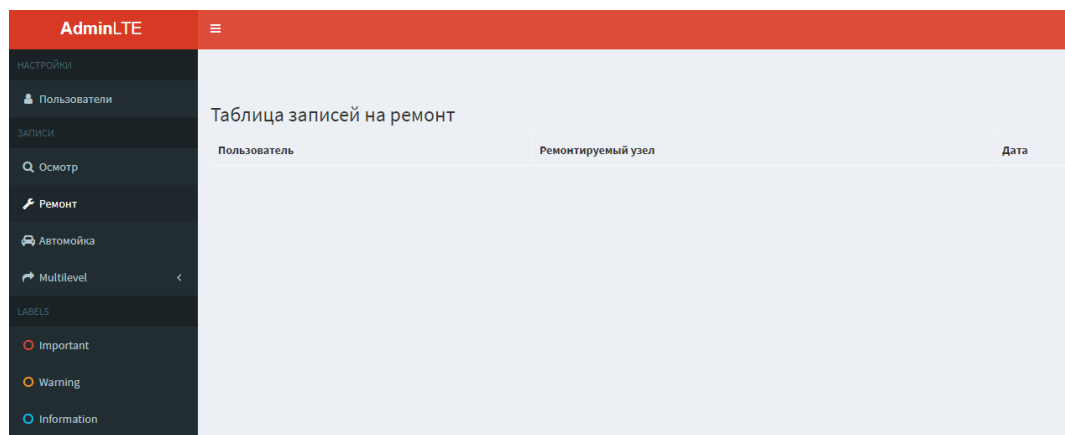
Форма записи на ремонт представлена на рисунке 2.



The image shows a web form titled "Запись на ремонт авто" (Car Repair Booking). It includes a dropdown menu for "Ремонтируемый узел" (Repairable part) with "Ходовая система" (Suspension system) selected. Below it is a button "Открыть карту" (Open map). There is a date and time selector showing "2019-09-21". At the bottom is a blue button "Записаться" (Book). To the right of the form is a map of Moscow showing various districts like Красногорск, Балашиха, and Люберцы.

*Рис. 2. Форма записи*

Форма админ-панели, позволяющая модератору взаимодействовать с базами данных через удобный интерфейс, удалять и изменять данные, а также работать с базовой статистикой, представлена на рисунке 3.



*Рис. 3. Форма админ-панели*

### Список литературы

1. Сайт о программировании [Электронный ресурс]. – <https://laravel.ru>
2. Сайт о программировании [Электронный ресурс]. – <https://ru.stackoverflow.com>