**ГЛАВА 2. ТЕХНИЧЕСКАЯ РЕАЛИЗАЦИЯ.**

1. **Структура сайта**

В ходе изучения задания на создания системы и общения с заказчиком было составлено и согласованно техническое задание, согласно которому реализация системы будет представлять из себя многостраничный сайт. Общение с заказчиком и согласование работ было зафиксировано в облачной программе для управления проектами “Trello”

Первоначально система выглядела как административная панель, состоящая из таких страниц как:

1. Авторизация – данная страница содержит предоставляет функцию входа на сайт и является первой страницей сайта.
2. Профиль – данная страница содержит информацию о пользователе с возможностью редактирования данных авторизации (пароля, почты).
3. Оповещения – данная страница, в зависимости от пользователя, предоставляет актуальную информацию по всем назначенным задолженностям. Их статус, кем назначена, кому назначена, дату урока на который назначена, причину, учебный материал по задолженности и тест для её закрытия.
4. Решить/проверить тест – данная страница предоставляет возможность для пользователей “Ученик” решить присланный тест для закрытия задолженности. Учителям данная страница предоставляет возможность проверить решенный тест.
5. Журнал задолженностей по классам – данная страница предоставляет возможность увидеть список задолженностей по классам который ведет авторизованный преподаватель, либо задолженности по всем классам для пользователя “Администратор”.
6. Журнал задолженностей по предметам – данная страница предоставляет возможность увидеть список задолженностей по предметам который ведет авторизованный преподаватель. Так же данная страница предоставляет возможность выставлять оценку либо задолженностей за урок, прикрепляя тест и учебный материал для её закрытия.
7. Тесты – данная страница содержит все когда-либо созданные тесты с возможностью фильтровать их по предмету.
8. Создание тестов – данная страница предоставляет возможность создания нового теста.
9. Редактирование теста – данная страница предоставляет возможность для пользователя кто создал данный тест и администрации редактировать данный теста.
10. Учебные материалы – предоставляет возможность просматривать все созданные учебные материалы.
11. Определенный учебный материал – данная страница содержит необходимые ссылки и информацию по учебе.
12. Редактирование учебных материалов – данная страница доступна для преподавателей и администрации и предоставляет возможность редактировать данные учебного материала.
13. Создание учебного материала – данная страница предоставляет возможность преподавателям создавать новые учебные материалы.
14. Административная панель – предоставляет информацию о доступных функция для администрации сайта.
15. Добавление новых пользователей – предоставляет возможность для администрации добавлять новых пользователей.
16. Добавление новые классы– предоставляет возможность для администрации создавать новые классы.
17. Добавление новые уроки– предоставляет возможность для администрации создавать новые уроки.

Данная структура сайта представлена на рисунке 3.

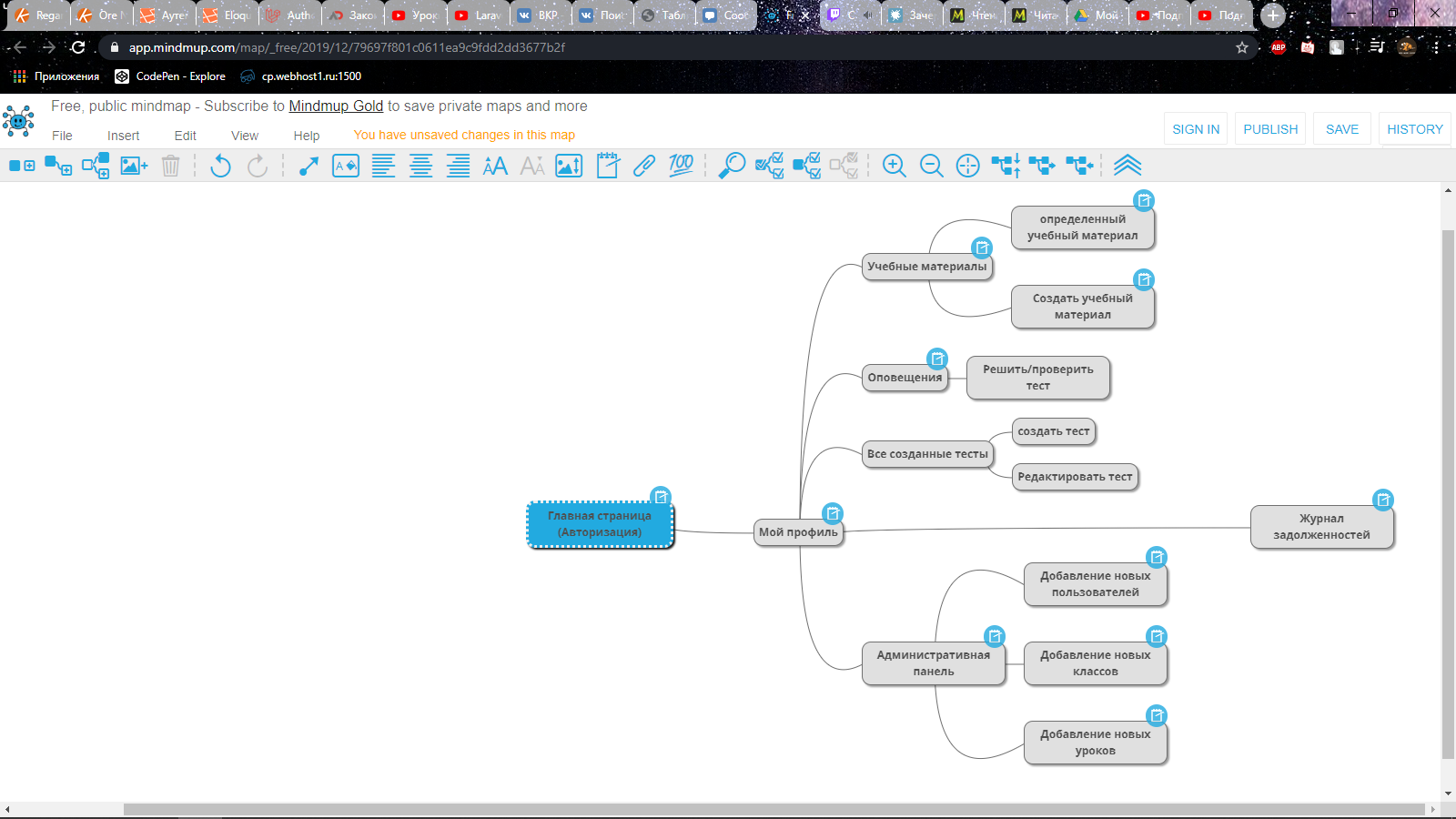


Рисунок 3. Первоначальная структура сайта.

Так же после общения с основными пользователями данной системы были выведены дополнительные функции для сайта. Сбор информации был зафиксирован через платформу Google Формы. Все выводы по данным опросам приведены в таблицах 10, 11 и 12. Ответы из которых были сделаны данные выводы прикреплены к данному документу (Приложение 2, 3). Так же благодаря данным опросам было сделано окончательное решение с выбором способа закрытия задолженностей.

|  |  |
| --- | --- |
| Вопрос | Ответ |
| Нужна ли возможность просматривать информацию об учениках? | Да |
| Нужна ли возможность на сайте создавать и записываться на дополнительные занятия? | Да |
| Нужна ли статистика по успеваемости учеников | Нет |

Таблица 10. Сбор информации с представителей пользователей “Ученики”.

|  |  |
| --- | --- |
| Вопрос | Ответ |
| Нужна ли возможность просматривать информацию об учениках? | Да |
| Нужна ли возможность на сайте создавать и записываться на дополнительные занятия? | Да |
| Как ученики должны закрывать задолженности? | Онлайн тестирование |
| Нужна ли статистика по успеваемости учеников | Нет |

Таблица 11. Сбор информации у представителей пользователей “ученики”.

|  |  |
| --- | --- |
| Вопрос | Ответ |
| Нужна ли на сайте система отчетов? | Да |
| Как ученики должны закрывать задолженности? | Онлайн тестирование |
| Нужна ли выгрузка отчетов | Да |

Таблица 12. Сбор информации у представителей пользователей “Администрация”.

Исходя из мнения опрошенных пользователей и администрации сайты было поставлено решение увеличить функционал сайта и добавить ему несколько страниц:

1. Отчетность – данная страница сайта содержит в себе отчёты об причинах задолженностях и данные о сложности тестов.
2. Карточка пользователя – данная страница предоставляет информацию об пользователе (ученик, учитель) его задолженностях и их закрытии для учителей и предмете и классе, который ведет преподаватель для учеников.
3. Создание дополнительного занятия – данная страница предоставляет возможность создать запись на дополнительно занятие для учителя.
4. Все дополнительные занятия – данная страница сайта предоставляет возможность ученикам и родителям записываться на дополнительные занятие создание учителями, а учителям просматривать все созданные дополнительные занятия.
5. Landing page – страница сайта, на которой случайные пользователи (пользователи которые не являются не учениками, не администрацией школы) смогут ознакомится с данной школой и её администраций.

В итоге структура сайта выглядит следующим образом (рис. 4):

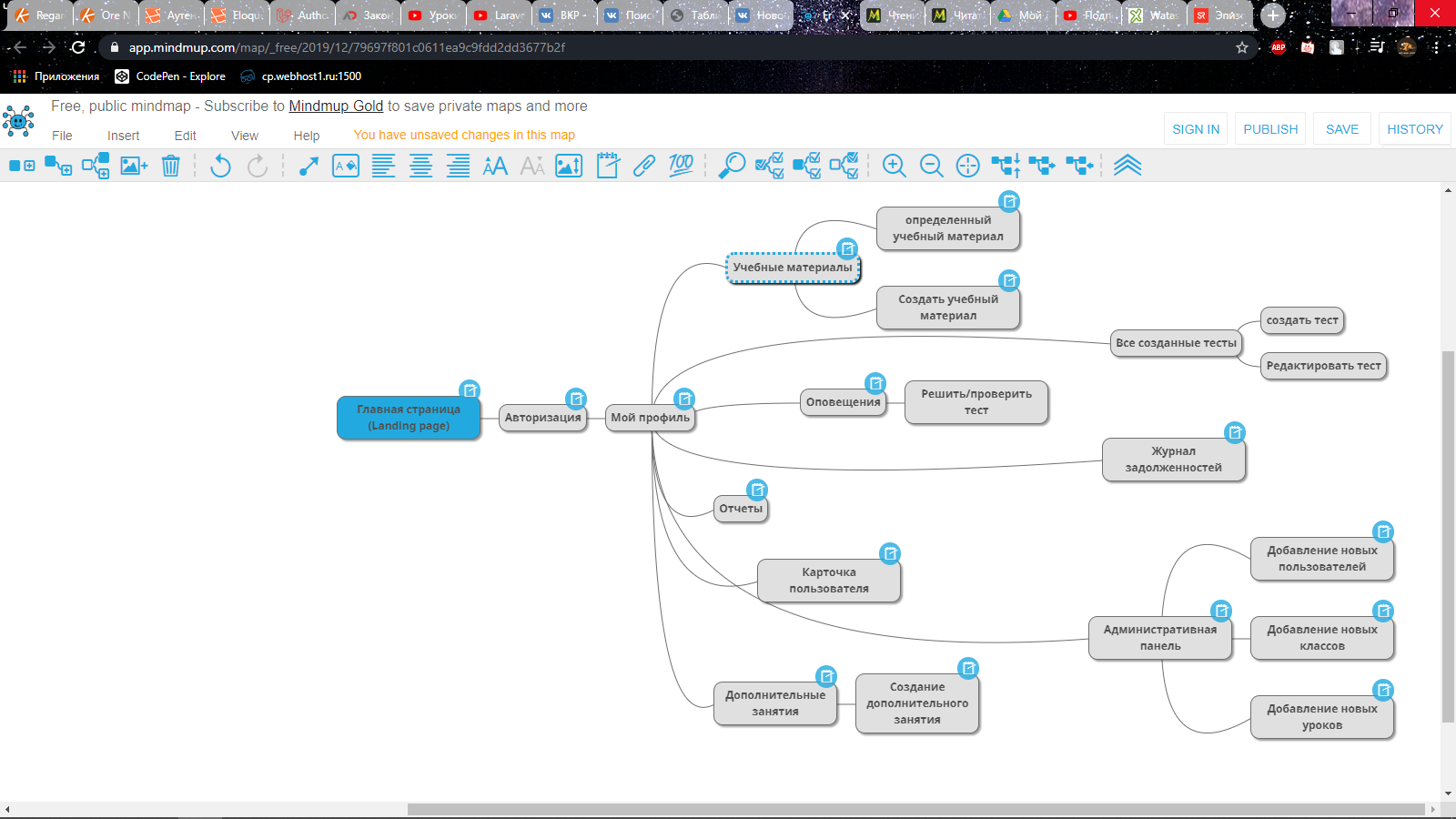


Рисунок 4. Конечная структура сайта.

В итоге, после работы над сайтом и определением структуры и функциональности сайта была выработана методика испытаний, по который необходимо тестировать и проверять на работоспособность данную систему (Приложение 4).

1. **Структура и работа с данными**

База данных содержит 14 таблиц, необходимых для хранения информации о пользователях, оценках, тестах, материалов и дополнительных занятиях. Структура базы показана на рисунке 3 и имеет следующие таблицы:

* user - пользователи сайта. Содержит информацию о логине, пароле, ФИО и типе пользователя;
* students - все студенты. Содержит информацию о ФИО ученика, дате рождения и классе, в котором он учится;
* class – классы. Предоставляет информацию об номере и букве класса.
* object – предметы. Предоставляет информацию о всех предметах, преподаваемых в школе и их названии;
* teacher – Содержит информацию об учителе ФИО и дате рождения.
* mark - оценки за контрольные и самостоятельные, в зависимости от поля type(оценка), выводится в оповещениях. В зависимости от поля state выводится данная стадия (исправлена, не исправлена);
* control - день работы. Содержит информацию о типе работы, дате, предмете и классу, по которому работа;
* type - тип работы(control). Все имеющиеся типы работ и их названия.
* test - все тесты. Содержит информацию об название тестов, по какому предмету и кем был создан;
* question - все вопросы к квесту. Содержит информацию об вопросах, ответах и тесте, к которому они прикреплены;
* answer - все ответы к вопросам. Содержит информацию об ответах на вопросы и правильности ответа.
* completeTests - решенные тесты учениками. Содержит информацию о всех решённых учениками тестах;
* material - все загруженные учебные материалы.
* AddLesson – Дополнительные занятие созданные учителями.
* WhoLesson – Информация о тех студентах, кто записан на занятие.



Структура базы данных. Рисунок 3.

1. **Интерфейс платформы**

После составления структуры сайта, технического задания и после общения с заказчиком был создан интерфейс веб приложения. Основными цветами были выбраны белый, серый и синий цвета, так как данные цвета подчеркивают официальность сайта и его принадлежность к образовательному ресурсу.

Красный цвета используются для выделения ошибок в тестах, авторизации и неудовлетворительных оценок или задолженностях. Это подчеркивает внимание преподавателей и студентов т.к. красный цвет является стандартным в учебных заведениях для выставления оценок или подчеркивания ошибок в работах.

После проведенных исследований дизайн страниц сайта выглядит следующим образом:

1. Меню личного кабинет было разделено на 2 части для более удобной навигации по сайту и большему пространству для основных функций сайта. Левая часть меню содержит все ссылки для навигации по сайту. Пользователей имеет возможность скрыть данную часть меню через кнопку в верхней части меню сайта. Это сделано для более удобной работы с сайтом на мобильных телефонах. Верхнее меню, помимо кнопки, содержит в себе информацию о том под каким аккаунтом пользователей авторизован на данный момент и кнопку выхода из системы. Пример данного дизайна представлен на рисунке 4.

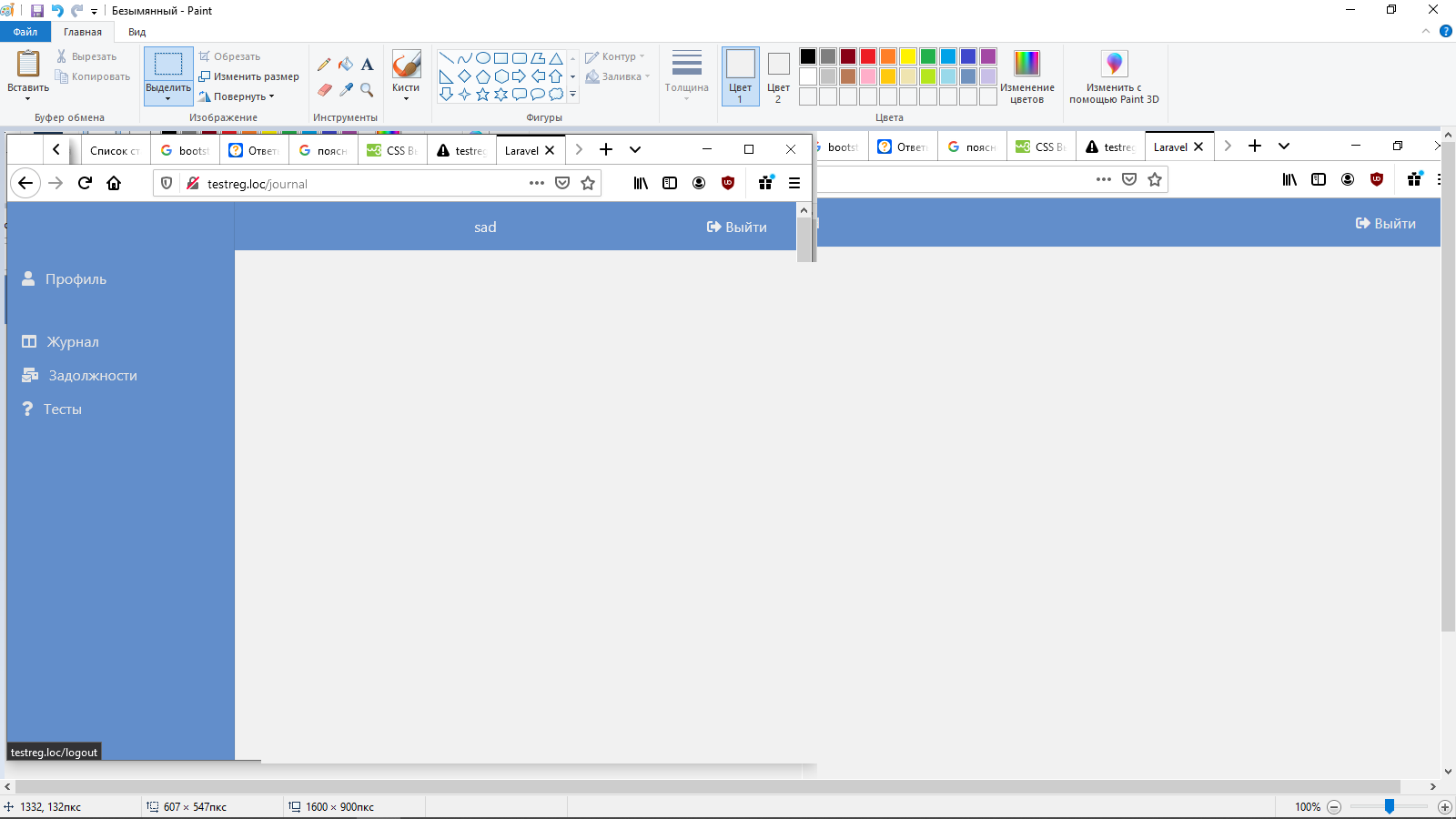


Рисунок 4. Меню личного кабинета.

1. Журнал задолженностей. Внешний вид журнала предполагает стандартный дизайн бумажного или электронного журнала, это сделано для более легкого освоения работы с журналом учителями. Журнал по классам разделен по классам, внутри которых он разделен по предметам. Все оценки “2”, а также пропуски или задолженности “н” имеют, как говорилось выше, красный цвет. Дизайн данной функции можно наблюдать на рисунке 5.

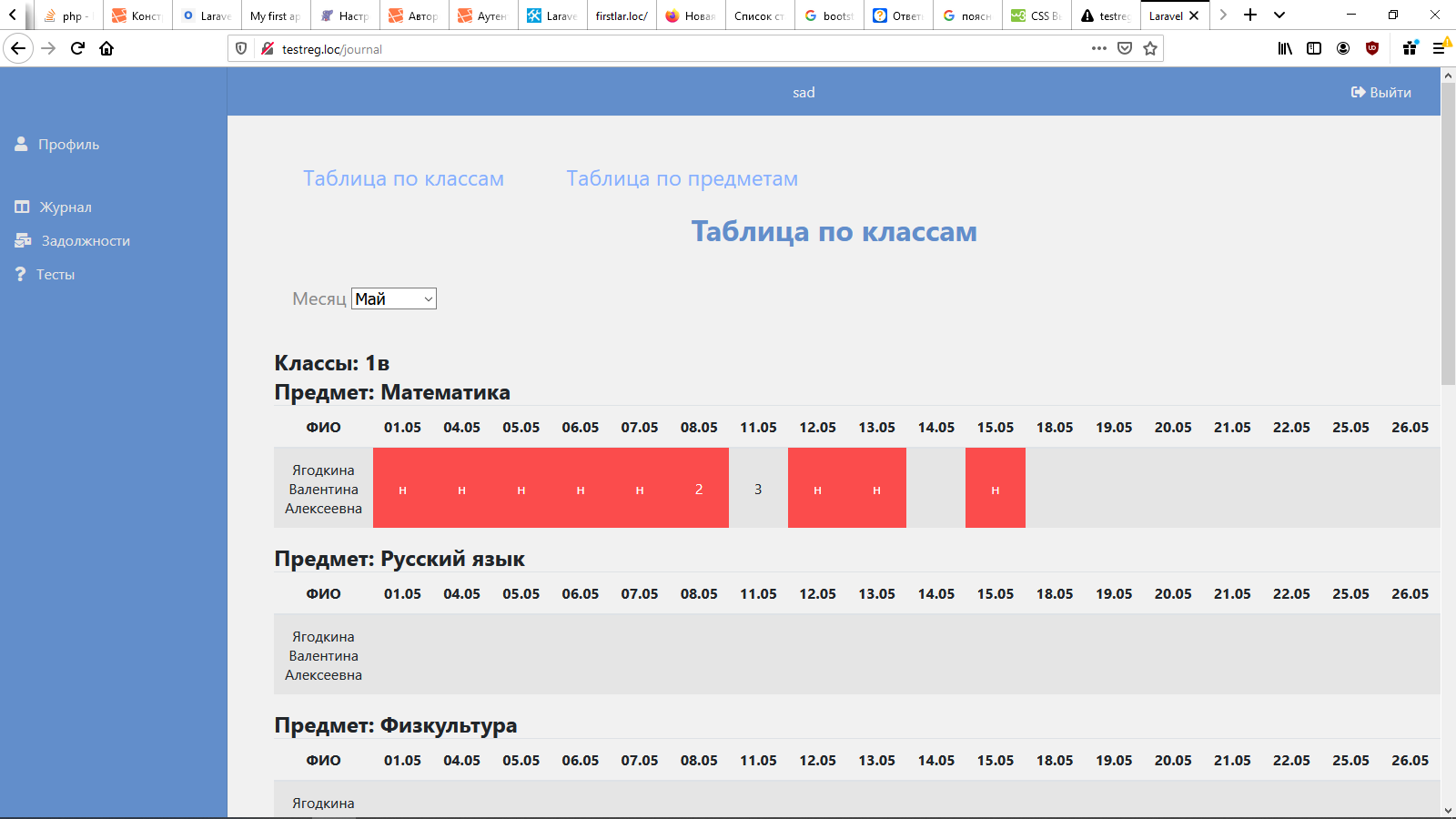


Рисунок 5. Журнал задолженностей.

1. Страница оповещений. Дизайн страницы оповещений представляет из себя таблицу с выполняемыми задолженностями. Данный список должен быть отфильтрован по дате последнего взаимодействия с заложенностью. Данная функция принимает такой вид так как она должна быть похоже на стандартное окно электронной почты, чтобы упростить работу для учителей и учеников с этой функцией. Данный дизайн можно наблюдать на рисунке 6.

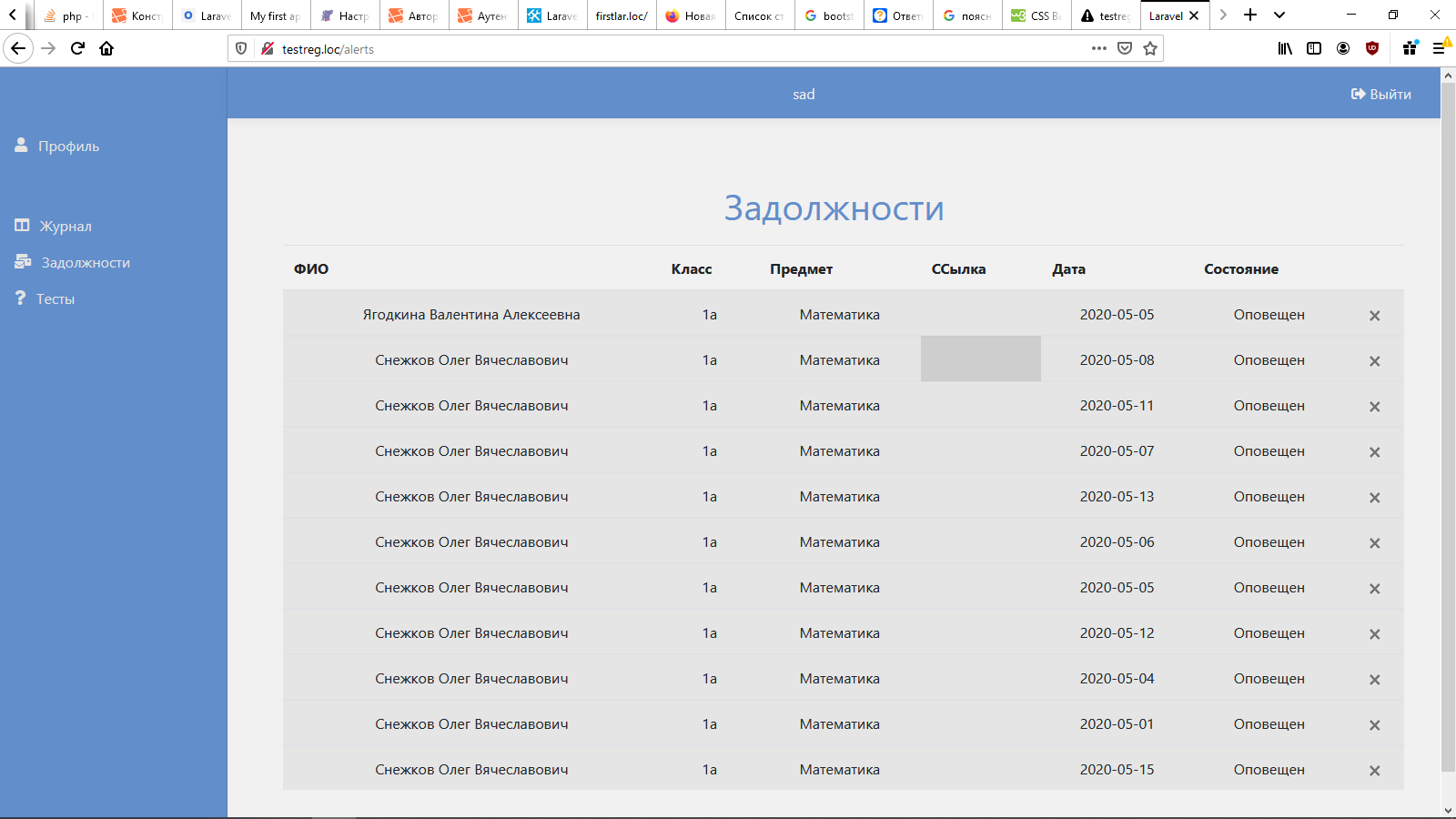


Рисунок 6. Страница оповещений.

1. Страница с тестами. Страница с тестами представляет собой список с краткой информацией о тесте и, если есть, прикрепленным материалом к теме теста. Данная страница имеет пагинацию с 10 тестами на одной странице. Дизайн данной функции можно наблюдать на рисунке 7.

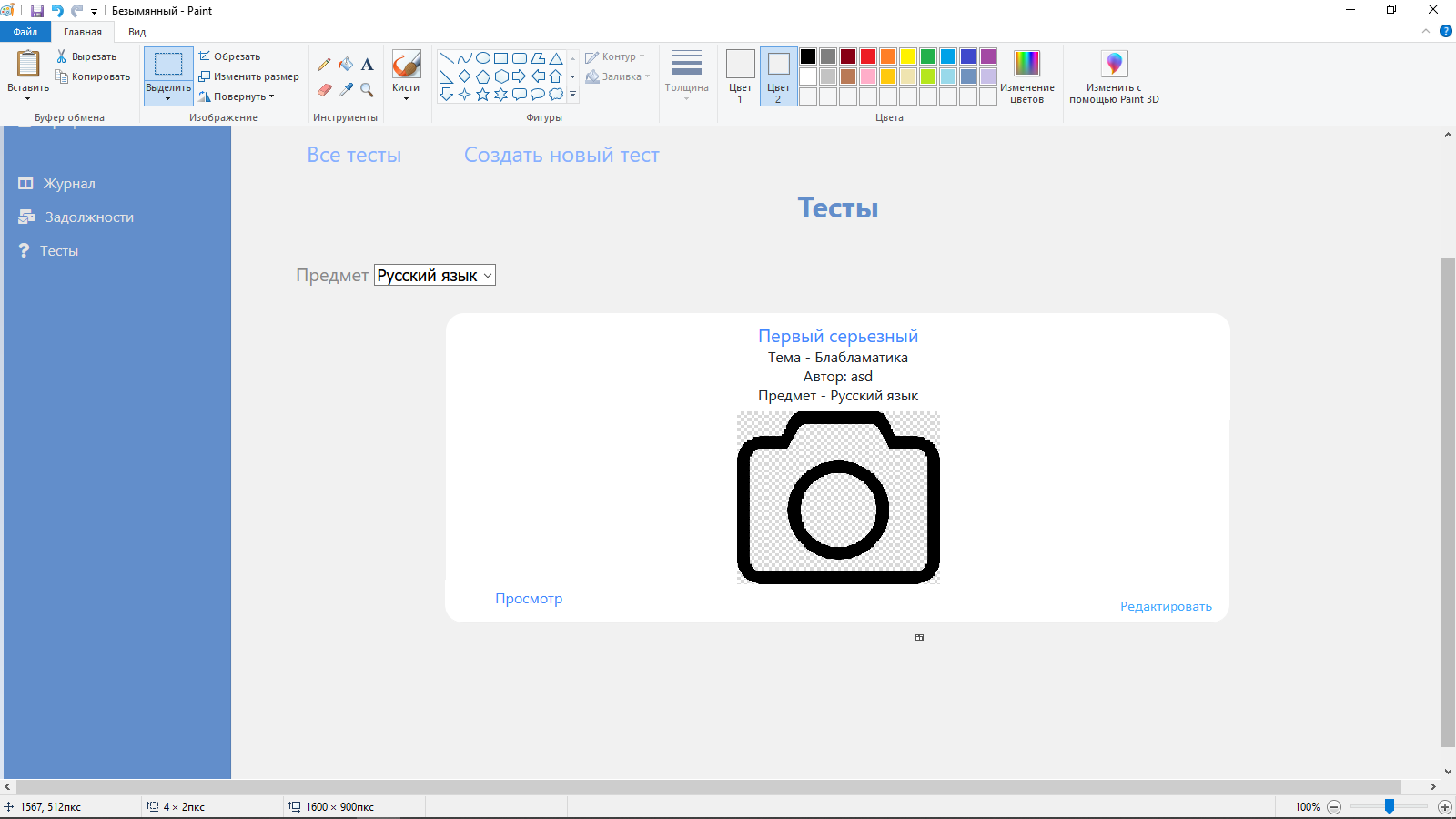


Рисунок 7. Страница с тестами.

1. Авторизация. Страница имеет стандартный вид и серо-синюю расцветку. Дизайн данной страницы можно наблюдать на рисунке 8.

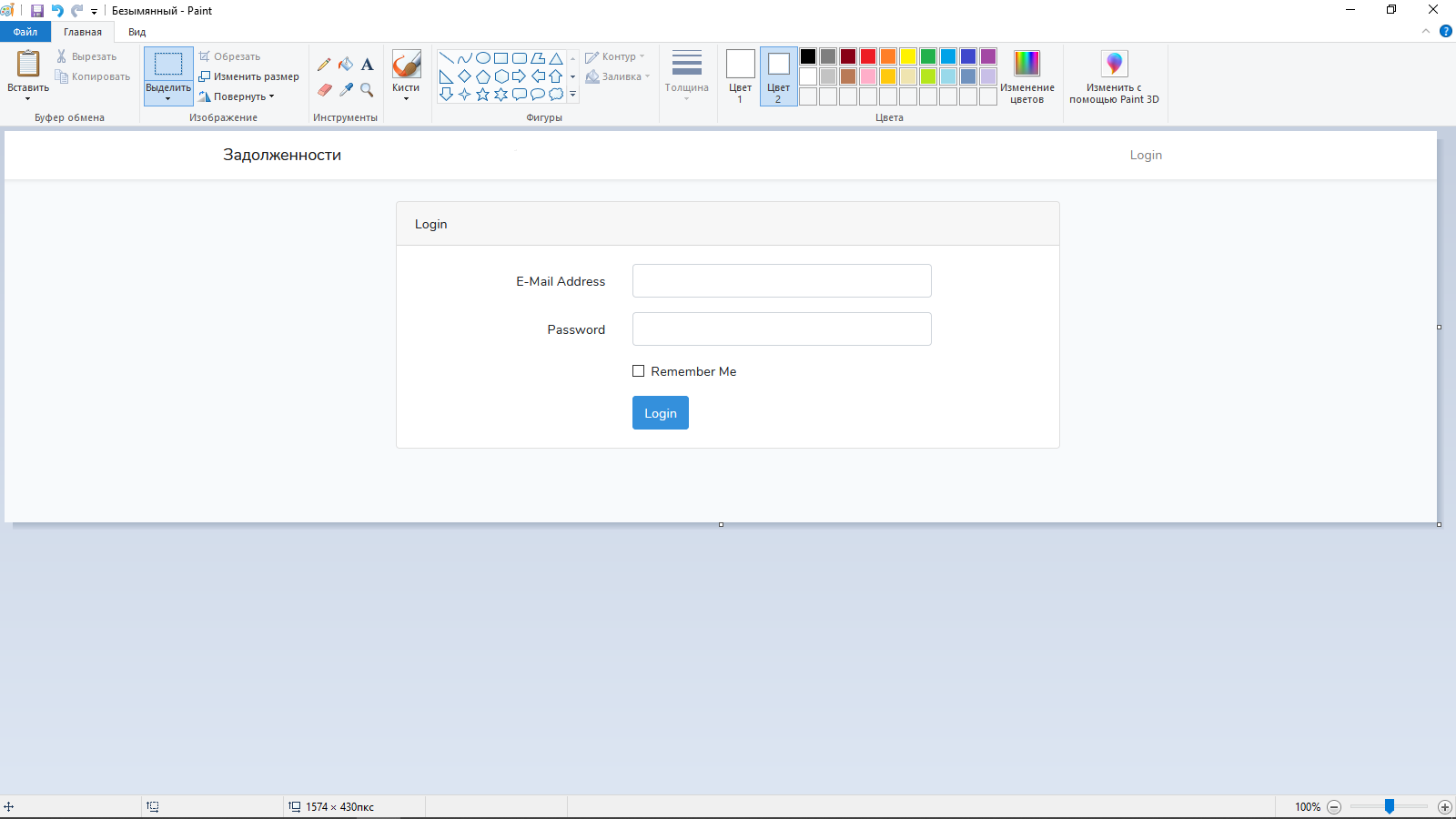


Рисунок 8. Авторизация.

1. **Реализация проекта**

Таким образом данная система представляет возможность решать все необходимые процессы для мониторинга и закрытия задолженностей по учебному плану. Так как эта система реализована в виде сайта, то языки программирования были выбраны адаптированные и использующееся в данных видах приложений.

Реализация кода проекта разделена на 2 части:

1. Клиентская часть.
2. Серверная часть.
   1. Клиентская часть.

Клиентская часть была реализована с помощью языка программирования JavaScript. Выбор данного языка связан с тем, что данный язык наиболее известен для исполнителя данного проекта и более адаптирован к интеграции с серверной частью сайта, реализованной при помощи фреймворка Laravel. Так же для реализации кода была использована библиотека JQuery.

На клиентской части сайта решаются задачи с визуальным отображением частей сайта и отправкой http запросов без загрузки страницы.

На клиентской части реализованы такие функции как:

1. Вывод модального окна при написании в поле определенного символа.

Данная функция необходима сайту для обработки запроса задолженности. В модальном окне выставляется причина задолженности возможность исправления задолженности. Так же в данном модальном окне используются функция, выводящая по флагу форме “Можно исправить задолженность” дополнительные поля с учебными материалами и тестами связанные с этим предметом и прикрепляемые к задолженности. Реализация данной функции показана в приложение 5.

1. Контроль вводимых символов в поле.

Данная функция необходима сайту для контроля введения оценок и задолженностей в журнале. Это поможет избегать лишних ошибок при контроле журнала. Данная функция отображена на рисунке 10.

function filterInputValue(value) {

            return /^[2345Нн]$/i.test(value) || (value === "");

        }

Рисунок 10. Контроль заполнения поля.

1. Обновление оценки и отправка Ajax запроса.

Данная функция необходима для передачи данных на базу данных без перехода на другую страницу или через обновление страницы. Она используется на странице журнала для отправки оценок и задолженностей в базу данных. Это ускорит заполнение журнала учителями. Данная функция реализована с использованием библиотеки JQuery. Код данной функции можно наблюдать в приложении 5.

1. Слайдер.

Данная функция используется на главной странице сайта и отображает главную информацию об учебном заведении. Данная функция реализована с использованием библиотеки JQuery. Код данной функции можно наблюдать в приложении 5.

1. Добавление вопросов и ответов.

Данная функция используется при создании тестов для добавления новых ответов и вопросов для составления теста. Реализация данной функции показана в приложении 5.

* 1. Серверная часть.

Серверная часть реализованна на языке программирования php с использование фреймворка Laravel. Данный фреймворк был использован для более безопасной работы сайта, оптимизации написания функционала и изучения новых технологий исполнителем проекта.

Серверная часть сайта нужна для взаимодействия с базой данных, и чтобы динамично отображать различные данные из базы данных.

На серверной части сайта реализованные такие основные части сайта как:

1. Создание журнала задолженностей.

Этот журнал используется для просмотра и выставления оценок и задолженностей учителями и администраторами.

1. Авторизация на сайте.

Данная функция реализована с помощью внутренних команд фреймворка Laravel и отредактирован под нужды конкретного сайта. Была убрана функция регистрации на сайте, так как данная функция не нужна, потому что все пользователи данная сайта должны добавляться через административную панель.

1. Создание строк в таблице журнала базы данных.

Данная функция используется для добавления оценок и задолженностей для передачи информации в базу данных. Данная функция реализована через контроллер Laravel и использует данные, которые посылаются через ajax запрос. Реализация данной функции показа в приложение 5.

1. Проверка на существующие оценки и задолженности по дате.

Данная функция проверяет есть ли по ячейки определенного ученика, предмету и дате задолженности или оценки. Реализации данной функции показана на рисунке 11.

function findValue($date, $user, $data, $lesson) {

        foreach($data as $d) {

            if( $d->date != $date ) continue;

            if( $d->lesson\_id != $lesson ) continue;

            if( $d->student\_id != $user ) continue;

            return $d->value;

        }

        return ' ';

    }

Рисунок 11. Проверка на оценки и задолженности.