



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ  
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»  
(МАИ)**

Кафедра 319 «Системы интеллектуального мониторинга»

Дисциплина «Web-программирование»

Система управления контентом «LMS»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

02335 – 01 81 01

Листов 24

Нормоконтроль

\_\_\_\_\_ / Коновалов К.А.

«\_\_»\_\_\_\_\_ 2020 г.

Принял

\_\_\_\_\_ / Коновалов К.А.

«\_\_»\_\_\_\_\_ 2020 г.

Задание принял к исполнению

студент группы

МЗО-335Б-17

\_\_\_\_\_ / Аншуков М.А.

«\_\_»\_\_\_\_\_ 2020 г.

## Оглавление

Введение.....	3
1. Постановка задачи.....	3
2. Описание приложения.....	4
3. Организация данных.....	18
3.1. Организация входных данных.....	18
3.2. Организация выходных данных.....	18
4. Технические и программные средства.....	19
4.1. Системные требования при разработке.....	19
4.2. Системные требования при эксплуатации.....	19
5. Результаты работы программы и их оценка.....	21
5.1. Объект исследования.....	21
5.2. Цель испытания.....	21
5.3. Требования к программе.....	21
5.4. Методы испытания.....	21
5.5. Оценка результатов тестирования.....	22
Заключение.....	23
Источники литературы.....	24

## Введение

Веб-приложение для работы с информационной системой «LMS» может быть использовано как программное обеспечение для автоматизации тестирования и экзаменовки студентов как части учебного процесса введённым дисциплинам.

### 1. Постановка задачи

Реализовать веб-приложение для работы с информационной системой «LMS» используя php, для реализации бэкэнда, и стек HTML5, CSS3, JS, для реализации фронтенда. Веб-приложение должно уметь обращаться с БД lms\_db информационной системы и обеспечивать функционал, представленные на рис. 1.

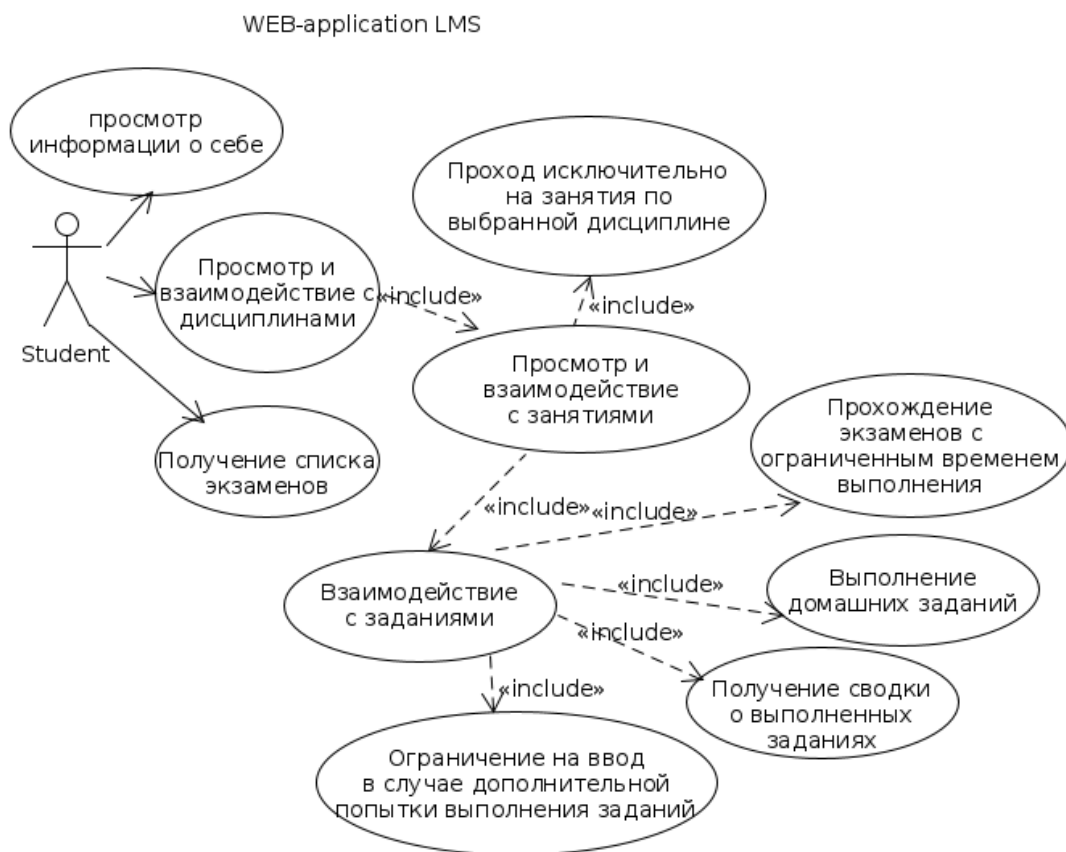


Рис. 1. Требования к функционалу веб-приложения

## 2. Описание приложения

Для хранения информации в ИС «LMS» используется СУБД PostgreSQL с базой данных «lms\_db». В базе данных для хранения и структурирования информации реализованы пять связанных между собой таблиц. Схема базы данных изображена на рисунке 2.

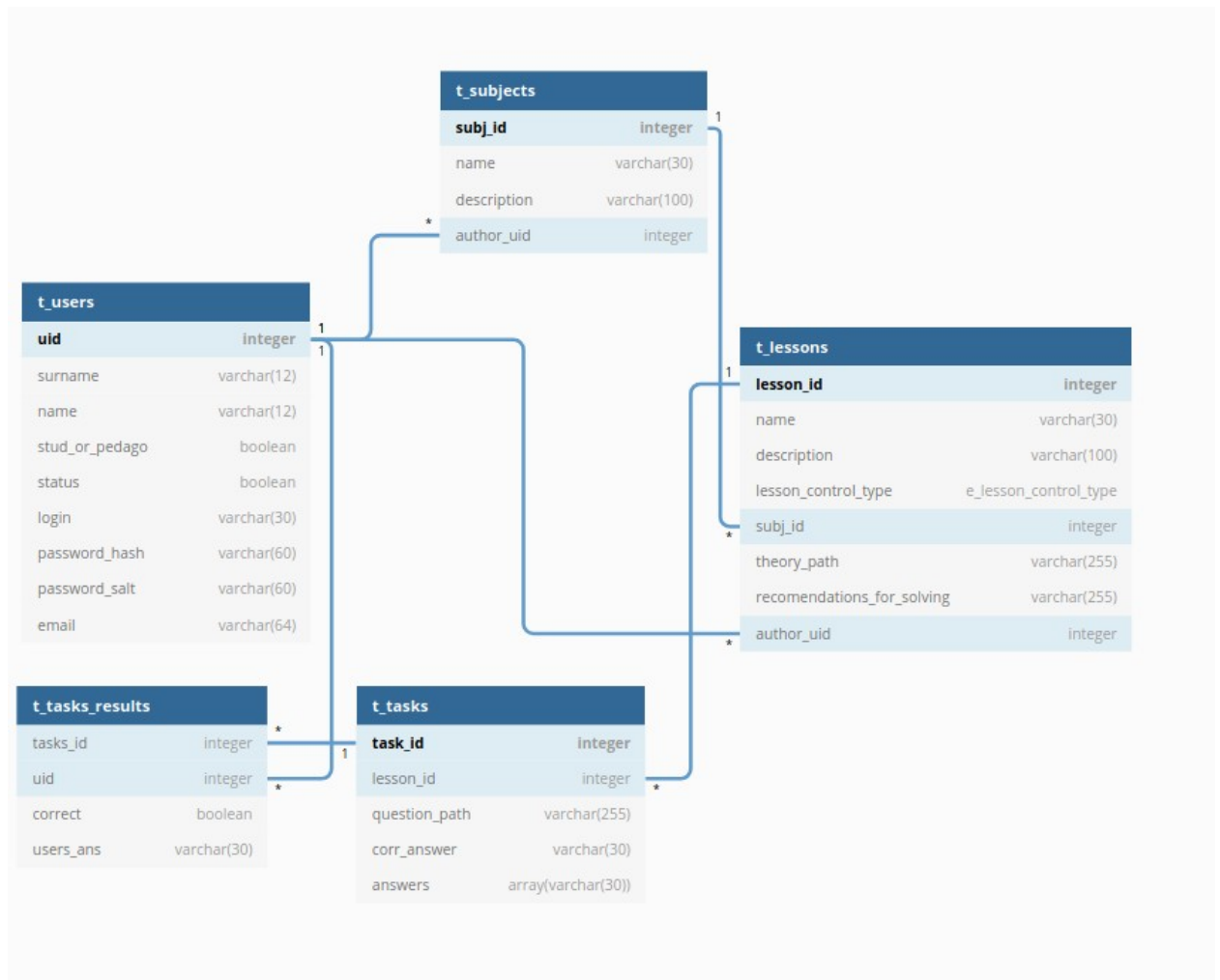


Рис. 2. Схема базы данных lms\_db

Таблица t\_users содержит информацию о пользователях, её структура представлена в таблице 1. Поле Id является ключевым в данной таблице.

Таблица 1 – описание таблицы t\_users

Поле	Тип	Ключ	Описание
uid	Int Increment NOT NULL	PK	Идентификатор пользователя
surname	varchar(12) NOT NULL		Фамилия пользователя
name	varchar(12) NOT NULL		Имя пользователя
stud_or_pedago	boolean(50) NOT NULL		Значение студент(true) или преподаватель(false)
login	Varchar(30) NOT NULL UNIQUE		Логин пользователя
password_hash	varchar(60) NOT NULL		Хэш пароля
password_salt	varchar(60) NOT NULL		Соль пароля
email	varchar(64)		Электронная почта пользователя

Таблица t\_subjects содержит информацию о введенных дисциплинах. Имеет связь один ко многим с таблицей t\_users через поле author\_uid и с таблицей t\_lessons через поле subj\_id . Структура таблицы представлена в таблице 2.

Таблица t\_lessons содержит информацию о введенных занятиях по дисциплинам. Имеет связь один ко многим с таблицей t\_users через поле author\_uid, с таблицей t\_subjects через поле subj\_id и с таблицей t\_tasks через поле lesson\_id, структура таблицы представлена в таблице 3.

Таблица t\_tasks содержит информацию о заданиях по занятиям. Имеет связь один ко многим с таблицей t\_lessons через поле lesson\_id и с таблицей t\_tasks\_results через поле task\_id, структура таблицы представлена в таблице 4.

Таблица t\_tasks\_results содержит информацию результатах работы студентов. Имеет связь один ко многим с таблицей t\_tasks через поле task\_id, структура таблицы представлена в таблице 5.

Таблица 2 – описание таблицы t\_subjects

Поле	Тип	Ключ	Описание
subj_id	Int increment NOT NULL	PK	Идентификатор дисциплины
name	varchar(30) NOT NULL		Название дисциплины
description	varchar(255) NOT NULL		Описание дисциплины
author_uid	Int NOT NULL	FK	Идентификатор автора (пользователя)

Таблица 3 – описание таблицы t\_lessons

Поле	Тип	Ключ	Описание
lesson_id	Int increment NOT NULL	PK	Идентификатор занятия
name	varchar(30) NOT NULL		Название занятия
description	varchar(255) NOT NULL		Описание занятия
lesson_control_type	enum e_lesson_control_type NOT NULL		Режим прохождения заданий (возможные значения: Homework, Exam)
subj_id	Int NOT NULL	FK	Идентификатор дисциплины
theory_path	varchar(255) NOT NULL		Теоретическая справка
recomendations_for_solving	varchar(255) NOT NULL		Рекомендации к выполнению заданий
author_uid	Int NOT NULL	FK	Идентификатор автора (пользователя)

Таблица 4 – описание таблицы t\_tasks

Поле	Тип	Ключ	Описание
task_id	Int increment NOT NULL	PK	Идентификатор задания
lesson_id	Int NOT NULL	FK	Идентификатор занятия
question_path	varchar(255) NOT NULL		Вопрос/задание
corr_answer	varchar(30) NOT NULL		Верный ответ
answers	array(varchar(30)) NOT NULL		Варианты ответа

Таблица 5 – описание таблицы t\_tasks\_results

Поле	Тип	Ключ	Описание
task_id	Int increment NOT NULL	PK	Идентификатор задания
uid	Int NOT NULL	FK	Идентификатор пользователя
correct	Boolean NOT NULL		Критерий правильности
users_ans	varchar(30) NOT NULL		Ответ пользователя

Для разработки веб-приложения были выбраны php, для реализации бэкэнда, и стек HTML5, CSS3, JS, для реализации фронтенда., с использованием паттерна MVC.

Концепция паттерна MVC предполагает разделение приложения на три компонента:

- **Модель (model):** описывает используемые в приложении данные, а также логику, которая связана непосредственно с данными, например, логику валидации данных. Как правило, объекты моделей хранятся в базе данных.

В MVC модели представлены двумя основными типами: модели представлений, которые используются представлениями для отображения и передачи данных, и модели домена, которые описывают логику управления данными.

Модель может содержать данные, хранить логику управления этими данными. В то же время модель не должна содержать логику взаимодействия с пользователем и не должна определять механизм обработки запроса. Кроме того, модель не должна содержать логику отображения данных в представлении.

- **Представление (view):** отвечают за визуальную часть или пользовательский интерфейс, нередко html-страница, через который пользователь взаимодействует с приложением. Также представление может содержать логику, связанную с отображением данных. В то же время представление не должно содержать логику обработки запроса пользователя или управления данными.
- **Контроллер (controller):** представляет центральный компонент MVC, который обеспечивает связь между пользователем и приложением, представлением и хранилищем данных. Он содержит логику обработки запроса пользователя. Контроллер получает вводимые пользователем данные и обрабатывает их. И в зависимости от результатов обработки отправляет пользователю определенный вывод, например, в виде представления, наполненного данными моделей.

Отношения между компонентами паттерна представлены схемой на рис.3.



Рис. 3. Схема отношений между элементами MVC

В этой схеме модель является независимым компонентом - любые изменения контроллера или представления никак не влияют на модель. Контроллер и представление являются относительно независимыми компонентами. Так, из представления можно обращаться к определенному контроллеру, а из контроллера генерировать представления, но при этом нередко их можно изменять независимо друг от друга.



Такое разграничение компонентов приложения позволяет реализовать концепцию разделение ответственности, при которой каждый компонент отвечает за свою строго очерченную сферу. В связи с чем легче построить работу над отдельными компонентами. И благодаря этому приложение легче разрабатывать, поддерживать и тестировать отдельные компоненты.

## 2.1. Описание алгоритмов

При первом запуске приложения Студент должен получить логин и пароль у своего Преподавателя (так как lms не является полностью открытой системой). Что бы войти в приложение Студент должен авторизоваться, Преподаватель авторизоваться не сможет. Авторизация происходит каждый раз при загрузке какой-либо из страниц сайта (реализовано с помощью «cookie»). Это сделано, чтобы случайные пользователи не могли воспользоваться ресурсами сайта. Алгоритм авторизации представлен диаграммой последовательности рис. 4

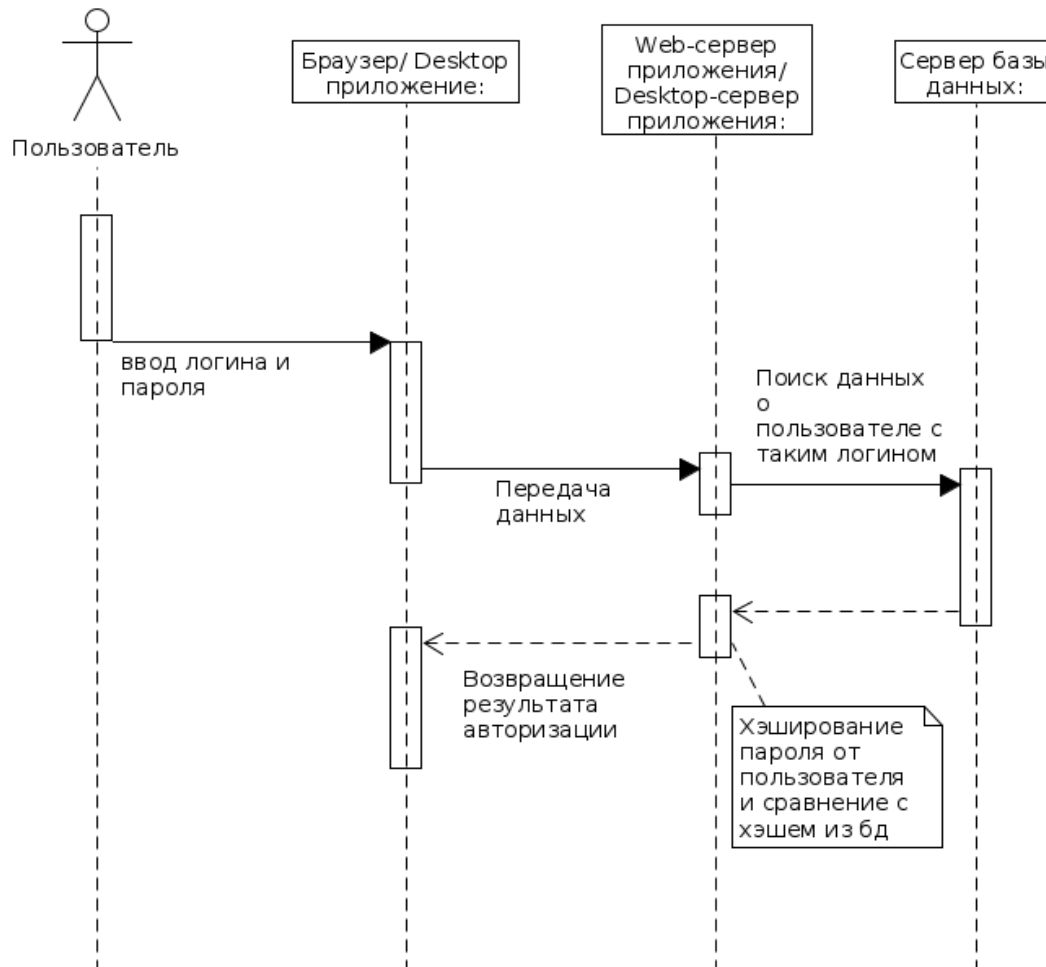


Рис. 4. Диаграмма последовательности алгоритма авторизации

После чего открывается главная страница, на которой отображается информация о Студенте, существующих экзаменах и выполненных заданиях, если таковые имеются. Алгоритм отображения выше упомянутых данных представлен диаграммой последовательности рис. 5.

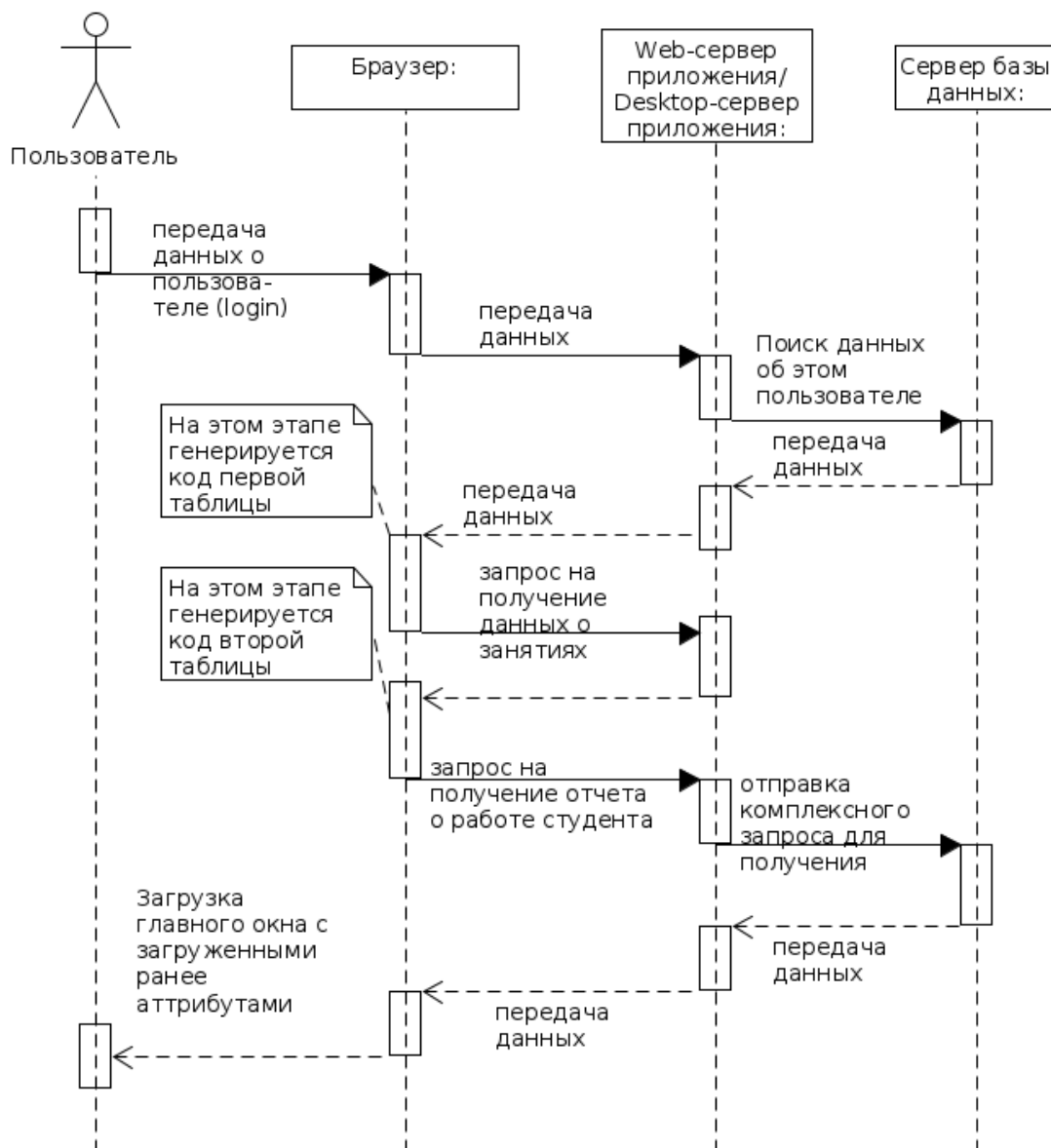


Рис. 5. Диаграмма последовательности алгоритма отображения информация о студенте, существующих экзаменах и выполненных заданиях

Чтобы посмотреть список существующих предметов студенту необходимо перейти в окно «subjects», после чего нажать на один из выбранных предметов и нажать на кнопку «Subjects lessons». После это

отобразятся занятия именно по этому предмету. Диаграмма последовательности алгоритма представлена на рис.6.

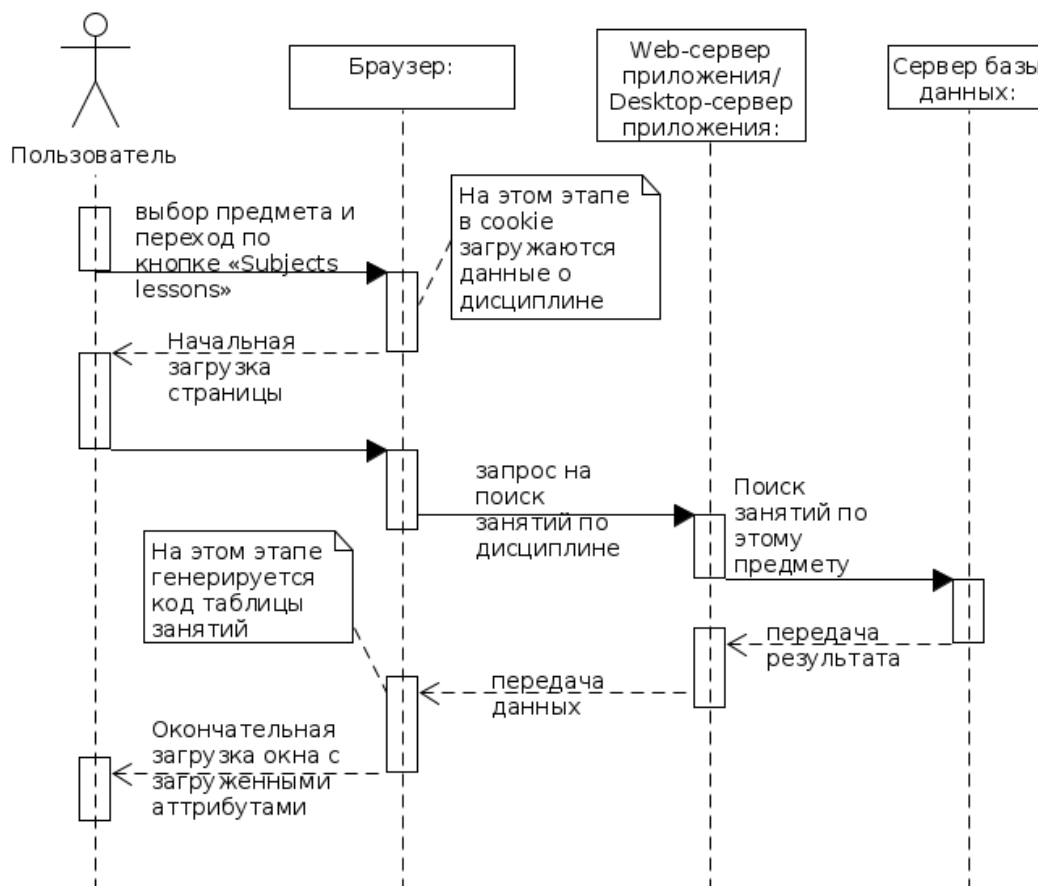


Рис. 6. Диаграмма последовательности алгоритма отображения занятий по предмету

Далее Студент может выбрать задания по какому занятию выполнять. Алгоритм выбора и загрузки тот же, что при отображении занятий по предмету. Студенту останется ответить на поставленные вопросы к занятию. Если тип контроля “Exam”, то на ответить необходимо в срок (по умолчанию, 15 минут), по истечению которого, ответы, независимо от желания Студента будут отправлены на сервер, а студент будет отправлен на главную страницу. Алгоритм последовательности внесения и отправки ответов представлен на рис.7.

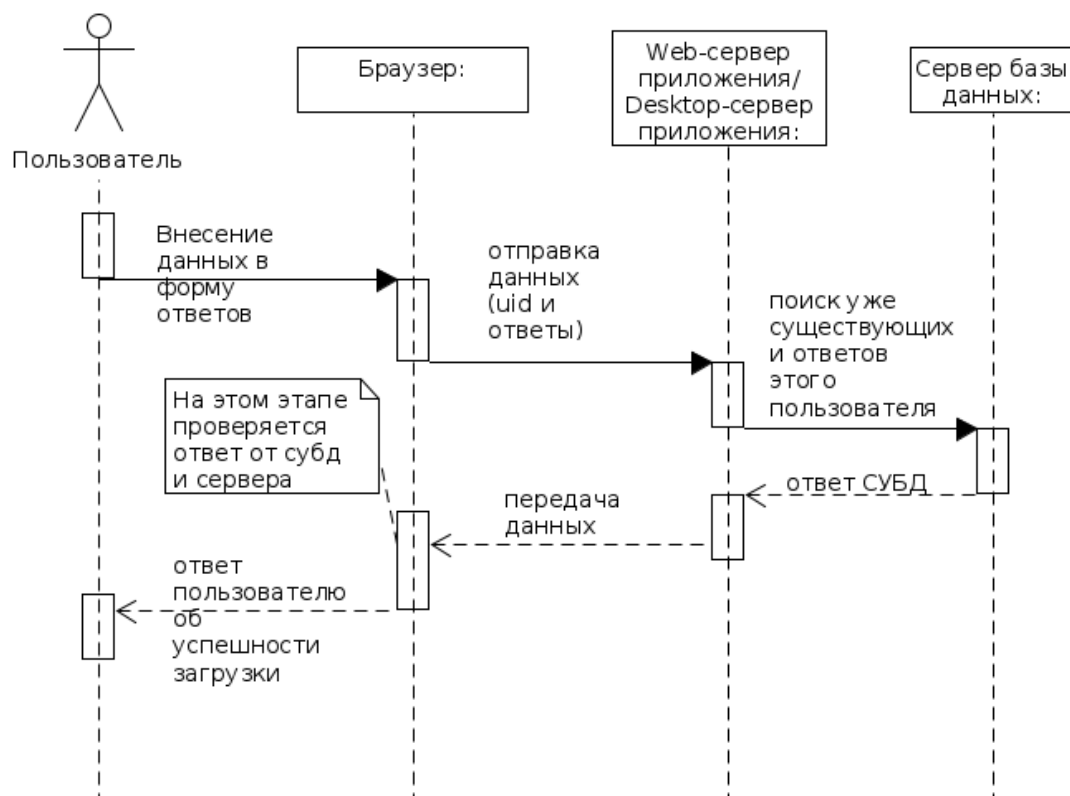


Рис. 7. Диаграмма последовательности алгоритма внесения и отправки ответов

Далее Студент перемещается в главное окно, где может увидеть результаты выполненной работы. Также он может выйти из своего аккаунта нажатием на кнопку “Exit”.

## 2.2. Результаты работы программы

### Запуск программы

Выводится окно авторизации (рис. 8), в котором находятся поля для ввода данных, кнопка авторизации и кнопка вывода контактов.

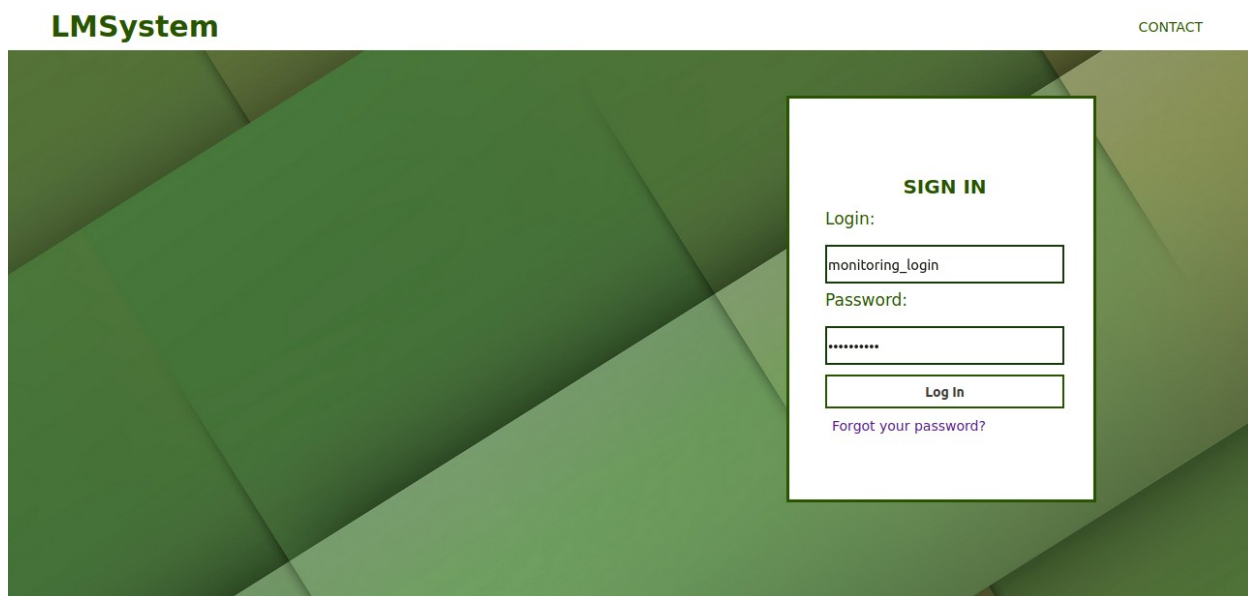


Рис. 8. Окно авторизации

### Главная страница

После авторизации открывается главная страница (рис. 9) в которой есть кнопки: «Home», которая перенаправляет на главную страницу; «Subjects», которая перенаправляет на страницу просмотра и выбора дисциплины; «Lessons», которая перенаправляет на страницу просмотра и выбора занятий; «Exit», которая позволяет выйти из аккаунта и перенаправляет на страницу авторизации. На этой же странице, вы можете видеть информацию о пользователе, таблицу экзаменов и выполненные на данный момент задания. Если последних двух нет, то они будут заменены на сообщения об отсутствии экзаменов и выполненных заданий, соответственно.

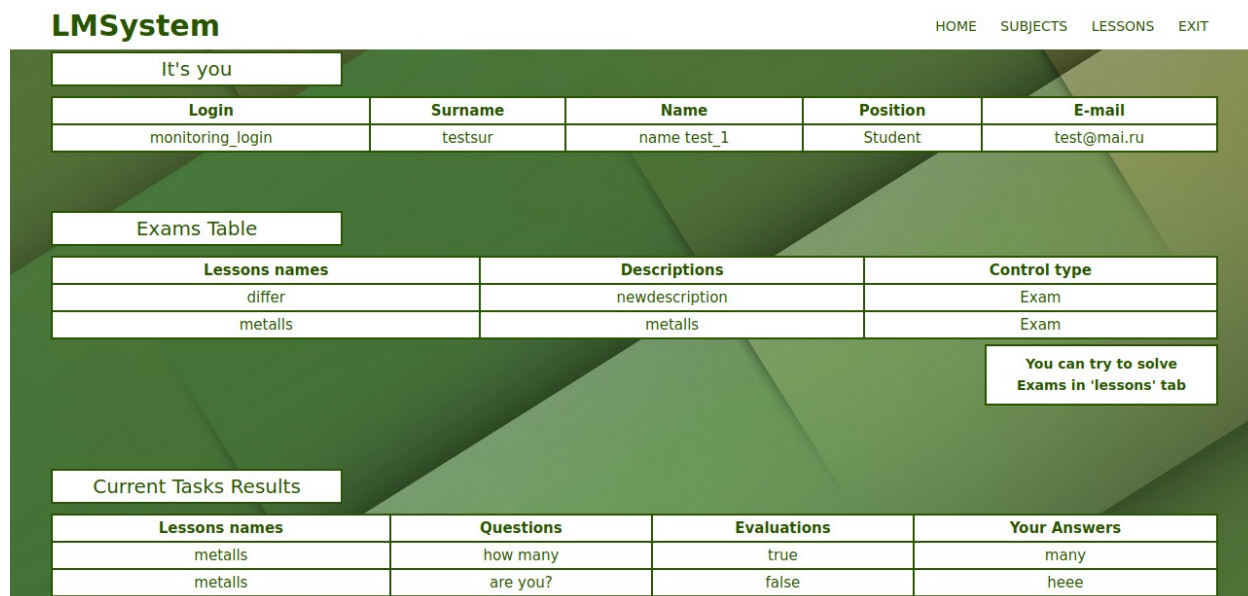


Рис. 9. Главная страница

### Выбор дисциплины

При нажатии кнопки «Subjects» открывается страница (рис.10) , в которой нужно выбрать дисциплину(сейчас выбран «russian») и нажать «Subjects lessons», после чего вы будете перенаправлены в окно «Lessons».

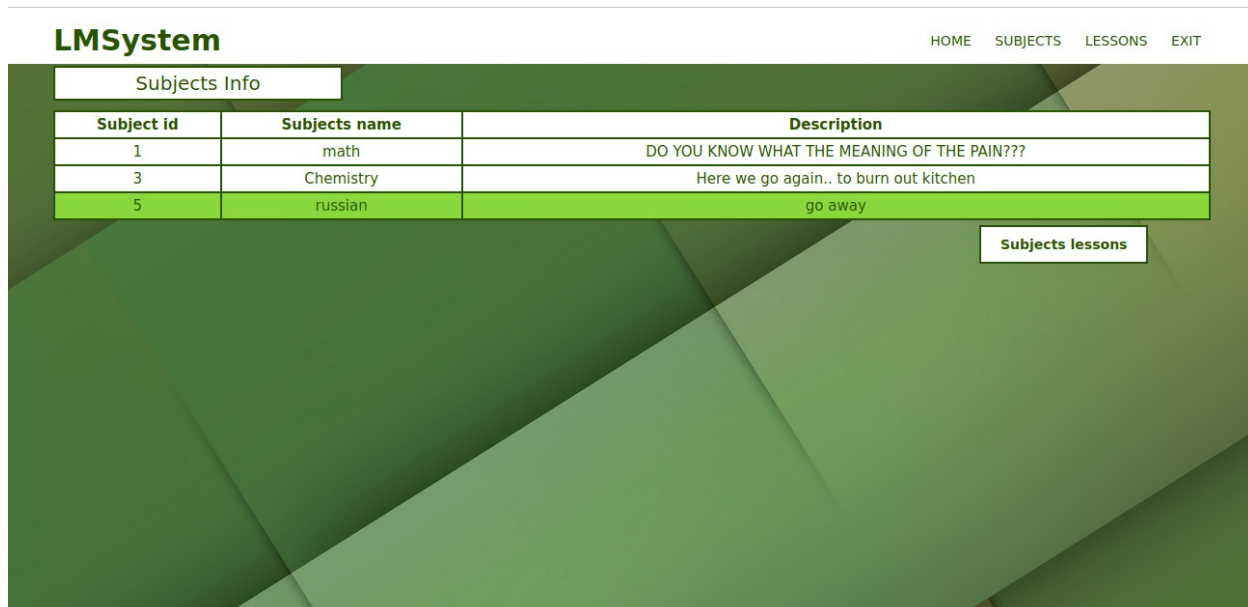


Рис. 10. Страница просмотра и выбора дисциплин

## Выбор занятия

При нажатии кнопки «Lessons» или выбора дисциплины в окне «Subjects» открывается страница (рис.11) , в которой нужно выбрать занятие(сейчас выбран «syntax») и нажать «Try to pass», после чего вы будете перенаправлены в окно выполнения заданий. Также стоит перед входом в режим выполнения заданий смотреть на столбец Control type. В нем указывается режим выполнения заданий, с возможными вариантами «Homework» и «Exam». Если выбрать второй, то установится дедлайн выполнения заданий (по умолчанию 15 минут), по завершению которого ответы будут отправлены на сервер.

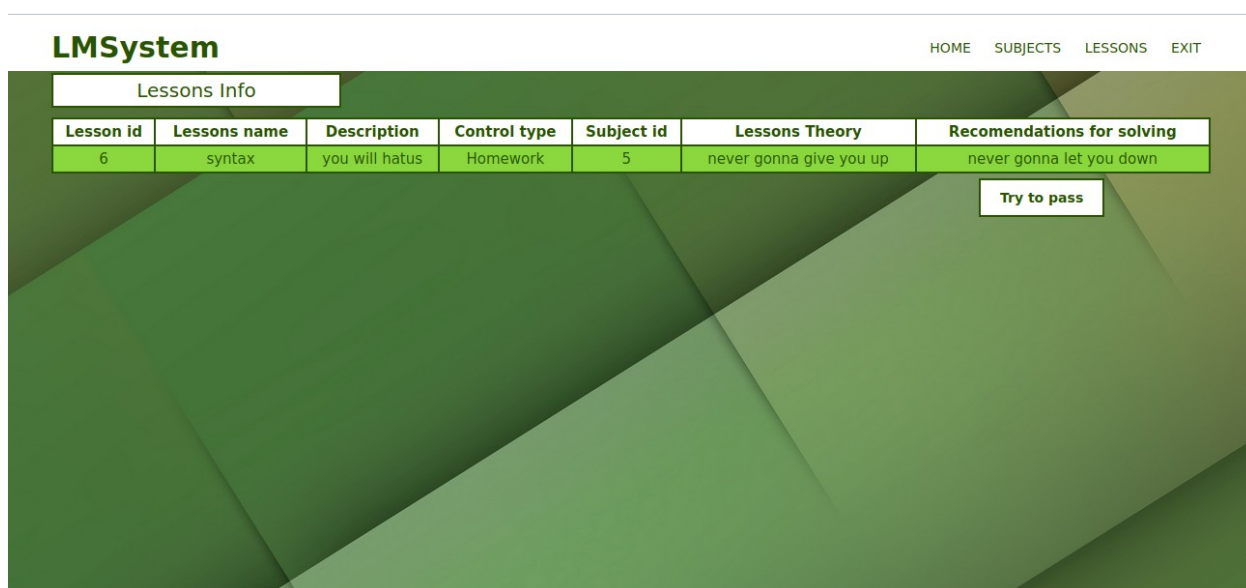


Рис. 11. Страница просмотра и выбора занятий

## Окно выполнения заданий

При открытии окна выполнения заданий в режиме «Exam»(рис.12), запустится таймер, за отсчётом которого пользователь может следить, чтобы не тратить время на подсчёт времени с других источников. По истечению таймера все ответы будут отправлены на сервер независимо от желания пользователя.

В режиме же «Homework» (рис.13) таймер отсутствует, чтобы пользователь мог выполнять задание, сколько ему нужно времени.



**LMSystem**

[HOME](#)
[SUBJECTS](#)
[LESSONS](#)
[EXIT](#)

Exam

Question	Answer
how many	many
are you?	heee
hello world?	фывфыв

End

hours: 0

minutes: 14

seconds: 43

Рис. 12. Окно выполнения заданий в режиме «Exam»

**LMSystem**

[HOME](#)
[SUBJECTS](#)
[LESSONS](#)
[EXIT](#)

Homework

Question	Answer
how are you!	

End

Рис. 13. Окно выполнения заданий в режиме «Homework»

В обоих случаях используются разного типа варианты ответов. Среди вариантов: выпадающий список ответов и поле ввода.

### **3. Организация данных**

#### **3.1. Организация входных данных**

##### **Входные данные для программы:**

- нажатие на кнопки, с помощью левой клавиши мыши;
- ввод данных в текстовые поля.

#### **3.2. Организация выходных данных**

##### **Выходные данные для программы:**

- результаты работы команд;
- перечень диагностических сообщений;
- текстовый отчет по результатам запроса.

## **4. Технические и программные средства**

### **4.1. Системные требования при разработке**

При разработке использовались Visual Studio Code, php (интерпретатор), PostgreSQL(СУБД), NGINX(http-сервер) Mozilla Firefox(браузер) и js (интерпретатор Node.js для тестирования js-скриптов), требования:

- Процессор с тактовой частотой не ниже 600 МГц; предпочтительно 1 ГГц и больше; Рекомендуется использовать как минимум двухъядерный процессор;
- 256 МБ ОЗУ; рекомендуется 512 МБ и больше ОЗУ;
- Место на жестком диске: до 100 ГБ (минимум 4 ГБ) свободного места в зависимости от установленных компонентов; обычно для установки требуется от 10 до 20 ГБ свободного места;
- Скорость жесткого диска: для повышения производительности установите операционную систему;
- Видеоадаптер с минимальным разрешением 720p (1280 на 720 пикселей);
- NGINX версии не ниже 1.16.1, без дополнений;
- манипулятор типа «мышь» и клавиатура;
- Windows 7 с пакетом обновления 1 (SP1) (с последними обновлениями Windows) и выше или (K)Ubuntu Linux 16.04 (64-bit) и выше.

### **4.2. Системные требования при эксплуатации**

#### **4.2.1. Системные требования для клиента**

Для клиента потребуется браузер. Требования:

- Любая операционная система с GUI;
- ОЗУ 512 Мб.

#### **4.2.2. Системные требования для сервера**

Для сервера требуется PostgreSQL, NGINX, операционная система из семейства Linux (взята за основу Ubuntu Server 19.04):

- Для Ubuntu Server 19.04 требуется как минимум 10 ГБ свободного места на диске.
- Для Ubuntu Server 19.04 требуется монитор Super VGA с разрешением 1024x768 пикселей или более высоким.
- Для поддержки функциональных средств Интернета требуется доступ к Интернету (могут применяться дополнительные тарифы).
- Быстродействие процессора: **Минимум:** процессор x64 с тактовой частотой 1,4 ГГц;  
**Рекомендуется:** 2,0 ГГц и выше;
- Минимум 1 ГБ ОЗУ; Рекомендуется 2 ГБ ОЗУ и больше;
- Процессор x64: AMD Opteron, AMD Athlon 64, Intel Xeon с поддержкой Intel EM64T, Intel Pentium IV с поддержкой EM64T.

## 5. Результаты работы программы и их оценка

### 5.1. Объект исследования

Объектом испытания является веб-приложение, для работы с системой управления контентом «LMS».

### 5.2. Цель испытания

Целью испытания является проверка работоспособности веб-приложения.

### 5.3. Требования к программе

Во время испытания необходимо проверить следующий функционал программы:

- корректная реакция на нажатие кнопок пользователем;
- корректная реакция на ввод пользователем данных;
- вывод сообщений в случае ошибок.

### 5.4. Методы испытания

Тестирование приложения проводилось методом “Белого ящика”, таблица тестов представлена в таблице 6.

Таблица 6 – Тесты

Назначение	Входные данные	Эталон	Результат
Вывод сообщений об ошибках при авторизации	Отсутствие входных данных	«Error! Wrong values of login of password fields»	Совпадает
	Несовпадение паролей	«Error! Wrong values of login of password fields»	Совпадает
	Верные данные	Переход на главную страницу	Совпадает

Продолжение таблицы 6

Нажатие на кнопку «Subjects lessons»	Без выбранного предмета	«You need to choose a subject»	Совпадает
	С выбранным предметом с существующими уроки по предмету	Переход на страницу уроков по выбранному предмету	Совпадает
	С выбранным предметом с не существующими уроки по предмету	«No such lessons»	Совпадает
Запуск режима выполнения заданий	Запуск выполнения домашней работы по уроку без заданий	«No available tasks for this lesson. Contact with your Educator. Do you want to stay here?»	Совпадает
	Провал по дедлайну	«Time is up! Your current answers were sent»	Совпадает
	Попытка закрытия страницы или перехода на другую страницу	«This page is asking you to confirm that you want to leave - data you have entered may not be saved.»	Совпадает
	Повторная попытка выполнения задания	«Your answers wasn't accepted. Maybe you were trying to solve before»	Совпадает

### 5.5. Оценка результатов тестирования

В тестовых примерах, приведенных в таблице 7, ожидаемая реакция совпадает с фактической, что доказывает работоспособность данной программы.

## **Заключение**

В соответствии с поставленной задачей было разработано веб-приложение для работы с системой управления контентом «LMS». Необходимый функционал полностью выполнен.

Веб-приложение работает стабильно и корректно обрабатывает команды пользователя, оснащена дополнительными комментариями к коду программы и может быть изменена другими программистами.

## Источники литературы

- Сайт с руководствами по JS, HTML5, CSS3: <https://developer.mozilla.org/ru/>;
- Руководство по php и встроенным библиотекам: <https://www.php.net/>;
- Руководство по HTML5, CSS3: <http://htmlbook.ru/>;
- Форум специализирующийся на JS: <https://javascript.ru>;
- Сайт с руководствами по JS, HTML5, CSS3: <https://www.w3schools.com/>.