

УДК 004.9

РАЗРАБОТКА МОДУЛЯ ОТСЛЕЖИВАНИЯ ДЕЙСТВИЙ СИСТЕМЫ ОЦЕНКИ И КОНТРОЛЯ ЗНАНИЙ СТУДЕНТОВ LMSDOT**Сиденко Кирилл Артурович¹,
Дедович Татьяна Григорьевна², Смирнов Даниил Павлович³**¹Студент;

Государственный университет «Дубна»;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: Dangeronika@yandex.ru.

²Кандидат физико-математических наук, доцент;

Государственный университет «Дубна»;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;

Старший научный сотрудник;

Объединенный институт ядерных исследований;

Россия, 141980, Московская обл., г. Дубна, ул. Жолио-Кюри, д. 6;

e-mail: tdedovich@jinr.ru.

³Аспирант;

Государственный университет «Дубна»;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: smirnov@lmsdot.com.

В статье исследованы функционал и применяемые технологии в модулях отслеживания действий популярных LMS-систем. Проанализирована система контроля и оценки знаний студентов Lmsdot. Сформулированы требования к автоматизации процесса отслеживания действий, разработана концептуальная модель, различные сценарии взаимодействия и дизайн соответствующего модуля для системы Lmsdot. Выполнена реализация программного обеспечения. Модуль внедрен в эксплуатацию и протестирован преподавателями Смирновым Д.П. и Дедович Т.Г. во время преподавания курса «Математическая логика и теория алгоритмов».

Ключевые слова: электронное обучение, LMS системы, образование, проверка списывания, история событий.

Для цитирования:

Сиденко К. А., Дедович Т. Г., Смирнов Д. П. Разработка модуля отслеживания действий системы оценки и контроля знаний студентов Lmsdot // Системный анализ в науке и образовании: сетевое научное издание. 2023. № 4. С. 129-141. EDN: ZFYRFG. URL : <https://sanse.ru/index.php/sanse/article/view/592>.

DEVELOPMENT OF THE ACTIVITY TRACKING MODULE OF CONTROL AND ASSESSMENT OF STUDENTS' KNOWLEDGE SYSTEM LMSDOT**Sidenko Kirill A.¹, Dedovich Tatyana G.², Smirnov Daniil P.³**¹Student;

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;

e-mail: Dangeronika@yandex.ru.

²PhD in Physical and Mathematical Sciences, associate professor;

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;

Senior Researcher;



Статья находится в открытом доступе и распространяется в соответствии с лицензией Creative Commons «Attribution» («Атрибуция») 4.0 Всемирная (CC BY 4.0) <https://creativecommons.org/licenses/by/4.0/deed.ru>

Joint Institute for Nuclear Research;
6 Joliot-Curie Str., Dubna, Moscow region, 141980, Russia;
e-mail: tdedovich@jinr.ru.

³PhD student;
Dubna State University;
19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;
e-mail: smirnov@lmsdot.com

The article investigates the functionality and technologies used in the activity tracking modules of popular LMS systems. The control and assessment of students' knowledge system (Lmsdot) is analyzed. The requirements for automating the process of tracking actions were formulated, a conceptual model, various interaction scenarios and the design of the corresponding module for the Lmsdot system were developed. Software implementation completed. The module was put into operation and tested by teachers D.P. Smirnov. and Dedovich T.G. while teaching the course "Mathematical Logic and Theory of Algorithms".

Keywords: e-learning, LMS systems, education, write-off check, history of events.

For citation:

Sidenko K. A., Dedovich T. G., Smirnov D. P. Development of the activity tracking module of control and assessment of students' knowledge system Lmsdot. *System analysis in science and education*, 2023;(4):129-141 (in Russ). EDN: ZFYRFG. Available from: <https://sanse.ru/index.php/sanse/article/view/592>.

Введение

В современном мире в образовательной сфере важную роль играют системы управления обучением *LMS* (англ. *Learning management system*). Такие системы позволяют создать единую базу электронных курсов: видеуроки, лекции, презентации, книги и задания. Процессом обучения управляет преподаватель. Он назначает пользователям проверочные работы и выполняет их проверку. Как правило *LMS* системы позволяют автоматически проверять тесты и фиксировать, сколько времени учился каждый пользователь. Данные формируются в виде сводного отчета, что упрощает оценивание уровня подготовки обучающихся.

LMS системы используются не только для дистанционного, но и очного обучения. Как правило, предмет ведут несколько преподавателей, которые совместно создают единый блок заданий. В этом случае важно иметь модуль, который позволит упростить коммуникацию преподавателей и обмен информацией между ними. Такую возможность может предоставить история действий при создании заданий и работ к курсу.

Во время заполнения курса преподаватель выполняет много действий, и он должен иметь актуальную информацию о выполненных изменениях в системе. Такую возможность предоставляет история действий внутри учебных групп.

Важно отметить, что качественная проверка знаний студентов - одна из самых важных задач в обучении. Задача является сложной, так как любой студент может использовать интернет ресурс для получения дополнительной информации во время проверочных работ. Все эти проблемы можно «смягчить», предоставив преподавателю возможность отслеживания действий студентов в рамках *LMS* системы.

В университете «Дубна» преподавателями Дедович Т.Г., Смирнов Д.П., Зинченко Д.А. используется система оценки и контроля знаний студентов *Lmsdot*. В системе, только разработчик может видеть действия, которые совершают пользователи системы. В случае необходимости получения информации о произошедших событиях преподаватели обращаются к разработчикам, выполняющим сопровождение функционирования системы. Поэтому возникла необходимость усовершенствовать модуль отслеживания действий. Модуль должен позволять преподавателям получать информация о действиях преподавателей в рамках курса и групп, а также действиях студентов при выполнении работ и занятий.

В статье проведен анализ модулей отслеживания действий в популярных *LMS* системах, показаны этапы разработки модуля и представлен его интерфейс.

Анализ модуля отслеживания действий в LMS-системах

Проведено исследование модулей отслеживания действий пользователей в разных LMS-системах. Для анализа были выбраны наиболее популярные в России системы *Moodle* [1], *iSpring Learn* [2], *Mirapolis* [3] и *Teachbase* [4]. Система *Mirapolis* не содержит модуля отслеживания действий пользователей. Для остальных систем представим сравнительные характеристики в таблице и опишем особенности технической реализации модуля.

Табл. 1. Характеристики модуля отслеживания действий в LMS системах

LMS	Moodle	iSpring Learn	Teachbase
Создание и редактирование курса, группы или работ	1	1	1
Активация и деактивация работ	1	0	0
Вставка контента и другие действия студентов при решении заданий	0	0	0
Вход и выход	0	0	1
Добавления или удаления участников в курс и группу	1	1	0
Отправки задания на проверку	1	1	0
Активность студентов	1	0	1

В табл. 1 рассмотрены типы событий, которые существенны для выполнения поставленных задач: создание и редактирование курса, группы или проверочных работ; активация и деактивация работ; отслеживание вставки контента и других действий студентов при решении заданий. Рассмотрен общий класс отслеживаемых событий: добавление или удаление участников в курс или группу; вход и выход студентов; активность студентов; отправка задания на проверку. Все события оцениваются по двухбалльной системе, где «0» обозначает отсутствие функционала, а «1» — наличие.

В системе *Moodle* используется собственный сервис отслеживания действий, который разделен на функциональные модули для разных типов событий. *Moodle* предоставляет многофункциональный инструмент по отображению всех событий. При получении произошедших событий можно указать параметры: тип, дата и название события, имя и *ip*-адрес инициатора события, сущность с которой произошло событие.

В *iSpring Learn* просмотр всех событий находится на главной странице как один из контентных блоков в виде прокручиваемого списка. Все сущности, участвующие в событии, являются ссылками.

В системе *Teachbase* отправляется запрос на получение списка произошедших событий, в котором либо отсутствует тип, либо указано определенное количество типов событий. Если тип событий не указан, то отображаются все произошедшие события. Типы событий связаны с бизнес-логикой, а именно: курсы, активности пользователей и кураторов, результаты встреч (например, лекция), результаты событий.

Результаты анализа показали, что большинство современных LMS систем имеют собственный модуль отслеживания действий, который обладает функционалом по отслеживанию различных изменений в курсе, группе и активностей пользователей. В некоторых модулях LMS систем разработан функционал по фильтрации запрашиваемых событий. Важно отметить, что ни одна из рассмотренных LMS систем не отслеживает событие, связанное с действиями студентов во время выполнения работы.

Перечислим функциональные требования, определяющие возможности пользователя в рамках бизнес-требований. Модуль должен позволять преподавателям получать информация о действиях всех преподавателей в рамках курса, своих действиях в группах и действиях студентов при выполнении работ и занятий. Истории курсов и групп должны находиться на страницах «Курс» и «Группа».

Последовательность действий студентов должна отображаться при нажатии на кнопку «История изменений» на страницах конкретных работ и занятий.

Представим нефункциональные требования, описывающие характеристики системы и её окружения. Система должна вносить изменения в таблицу логирований, согласно спецификации каждого события, в момент его совершения. Интерфейс модуля должен быть интуитивно понятным, иметь масштабируемую верстку и предоставлять информацию о совершенных действиях.

Список отслеживаемых событий

Составим список событий для истории изменений курсов и групп, а также последовательности действий студентов, после которых вносятся изменения.

В разделе «история» на страницах «Курс» и «Группа» отображаются следующие события: создание и редактирование курса, группы, шаблонов заданий и работ; активация и деактивации работы; изменение заданий в конкретной работе. Шаблон задания представляет собой отдельную сущность, у которой есть различные свойства: название, теги, форма ответа, описание, варианты заданий и настройки приватности. Используя шаблоны заданий, можно составлять шаблон работы. Шаблон работы – это последовательность заданий, а также описание и настройки приватности. Шаблоны заданий и работ используются для быстрого составления проверочных работ.

Информация о действиях студентов отображается в «Истории изменений» решения работ и заданий. Изменения вносятся, когда студент перестает набирать ответ в течение 30 секунд, нажимает на кнопку «сохранить» или «отправить» решение, а также использует вставку контента из поля с решением или внешнюю вставку.

Изменения базы данных

В системе *Lmsdot* [5-8] используется база данных *Postgres* [9]. Диаграмма *ERD* таблицы логирований *EventLog* базы данных представлена на рис. 1. Ниже указаны поля таблицы, используемые в клиентской части при отображении информации о действиях пользователей. Обязательные поля: идентификатор *id* совершенного события (тип *integer*, первичный ключ РК в таблице), тип произошедшего события *eventType* (строка, например, «CourseCreated»), тип пользователя *initiatorType* совершаемого действия, значение текущей даты *createdAt*. Необязательные поля («?»): идентификатор учебной группы *studyGroupId* (описывает действие в учебной группе и является вторичным ключом *FK* из таблицы учебных групп), идентификатор курса *courseId* (описывает действие в курсе и является вторичным ключом из таблицы курсов), поле *oldValue* описывает состояния объекта до внесения изменений в него (тип *javascript object notation*), поле *newValue* описывает состояния объекта при его создании или после редактирования, идентификатор *entityId* создаваемой сущности (курса, группы, работы или шаблона работы), идентификатор инициатора совершаемого действия *initiatorId* (внешний ключ из таблицы пользователей).

На рис. 2 показаны связи созданной таблицы *EventLog*, с другими таблицами в базе данных. Отметим, что база данных *Lmsdot* состоит из большого количества таблиц. Таблицы *StudyGroup*, *Course* и *User* описывают группы, курсы и пользователей, соответственно. Для изменения схемы таблиц базы данных был создан файл на синтаксисе библиотеки *prisma* [10], который генерировал файл с *sql* [11] командами. Сгенерированный файл содержит команды, выполнение которых вносит изменения в базу данных.

EventLog		
id	int	PK
eventType	varchar(255)	
studyGroupId	cuid?	FK
courseId	cuid?	FK
oldValue	jsonb?	
newValue	jsonb?	
entityId	varchar(255)?	
initiatorId	int?	FK
initiatorType	varchar(255)	
createdAt	datetime	

Рис. 1. Раздел Диаграмма ERD таблицы логирований EventLog

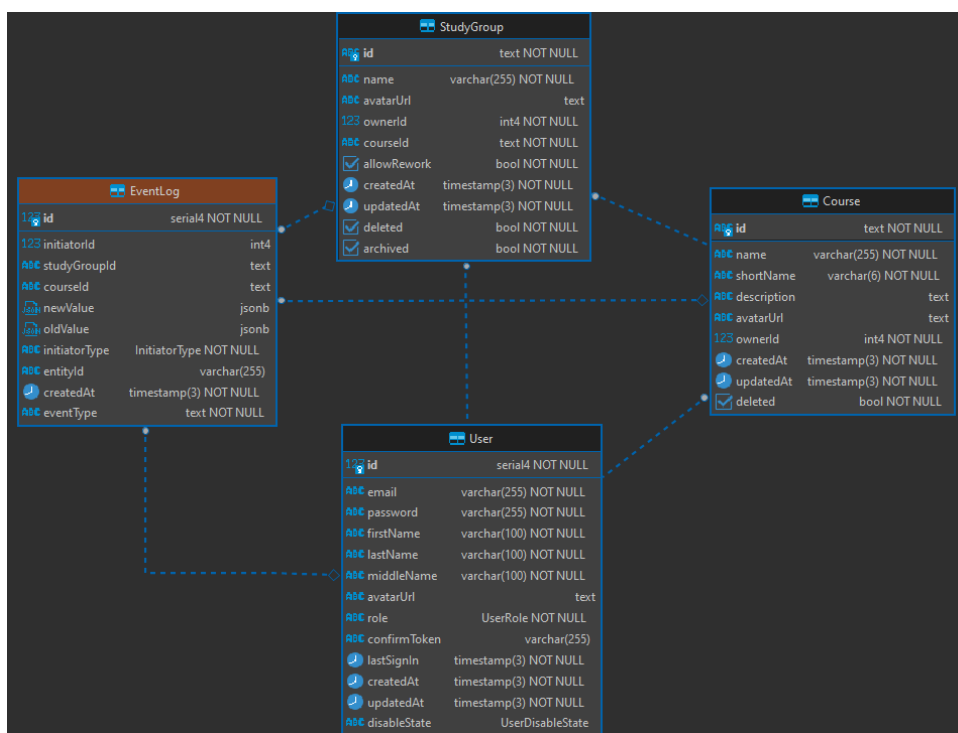


Рис. 2. Связи таблиц EventLog, StudyGroup, Course и User в базе данных

Изменения серверной части

Серверная часть системы *Lmsdot* написана на языке программирования *Javascript* [12], с надстройкой *Typescript* [13]. Структура серверной части декларируется фреймворком *nest* [14], поэтому разработан сервис для внесения данных в таблицу логирований, интерфейс серверного приложения (*api*) модуля и валидатор данных для устойчивой работы сервера.

Сервис, для внесения в таблицу *EventLog* информации о произошедшем действии, реализован с помощью функции, использующей методы библиотеки *CLS* [15]. При выполнении запроса в ней храниться уникальный идентификатор и тип пользователя.

Интерфейс серверного приложения (*api*) реализован посредством функции, которая обеспечивает получение данных из таблицы *EventLog*. Входные параметры функции: идентификаторы курса и группы, номер позиции, с которой необходимо взять количество строк, определяющих объем данных, возвращаемых сервером на один запрос;

Для валидации данных в фреймворке *nest* используется объект *dto* (*data transfer object*). Этот объект содержит описание сущности таблицы логирований и сравнивает их с входными данными. Если типы данных совпадают, то обработка запроса продолжается, иначе возвращается ошибка.

Изменения клиентской части

При разработке модуля в клиентской части, были реализованы компоненты и функции с использованием библиотеки *solid* [16].

При открытии раздела «История» на страницах «Курс» и «Группа» или при нажатии на кнопку «История изменений» на страницах конкретных работ и занятий, загружается компонент *EventLogs*, позволяющий получать информацию из таблицы логирований. Далее, отправляется запрос с параметрами фильтрации, чтобы получать информацию о событиях, связанных с сущностями на открытой странице. Отправляемые запросы реализованы с помощью библиотеки *graphql* [17]. Компонент *EventLogs* является оберткой для компонента *EventLog*, который используется для отображения события вместе с функцией *useEventLogInfo*. Эта функция принимает данные о событиях и возвращает объект с полями: название события, иконка сущности, цветовая иконка (характеризует тип события) и информации об изменениях сущности. Каждое поле объекта содержит в себе верстку, зависящую от типа события.

Модуль отслеживания действий в структуре *Lmsdot*

Система *Lmsdot* имеет несколько микро-сервисов: серверная часть, клиентские части для студентов и преподавателей, а также общая клиентская часть для распределения входящего потока пользователей по ролям. На рисунке 3 представлена схема обработки запроса на создание курса в системе *Lmsdot*.

Как только преподаватель аутентифицируется в систему, микро-сервис общей клиентской части перенаправит его в преподавательскую клиентскую часть. После заполнения формы на создание курса и нажатия кнопки «создать», на серверную часть отправляется запрос, построенный с использованием библиотеки *GraphQL*.

Как только запрос приходит на сервер, то вызывается соответствующий ему интерфейс *api*, а также вызывается метод библиотеки *CLS* для сохранения контекста запроса в краткосрочном хранилище (объекте). Далее происходит проверка данных валидатором на наличие всех необходимых полей в запросе, чтобы создать курс. Если проверка заканчивается успешно, то вызывается соответствующий сервис по созданию курса. Такой сервис, используя библиотеку *prisma*, вызывает метод по работе с таблицей курсов из базы данных. Как только информация о создаваемом курсе будет внесена в таблицу, вызывается событие с соответствующим типом, и передается модулю отслеживания действий (логирования), который создает запись в таблице *EventLog* на основе данных курса и *id* пользователя, полученного из *CLS*.

Созданный курс необходимо вернуть на клиентскую часть, чтобы отобразить в интерфейсе пользователя актуальную информацию обо всех курсах. Перед возвращением ответа от серверной части происходит вторая проверка на соответствие возвращаемых данных. При успешной проверке, на запрос возвращается ответ со статусом 203. Как только клиентская часть получает созданный курс, вызывается перерисовка страницы для отображения обновленных данных.

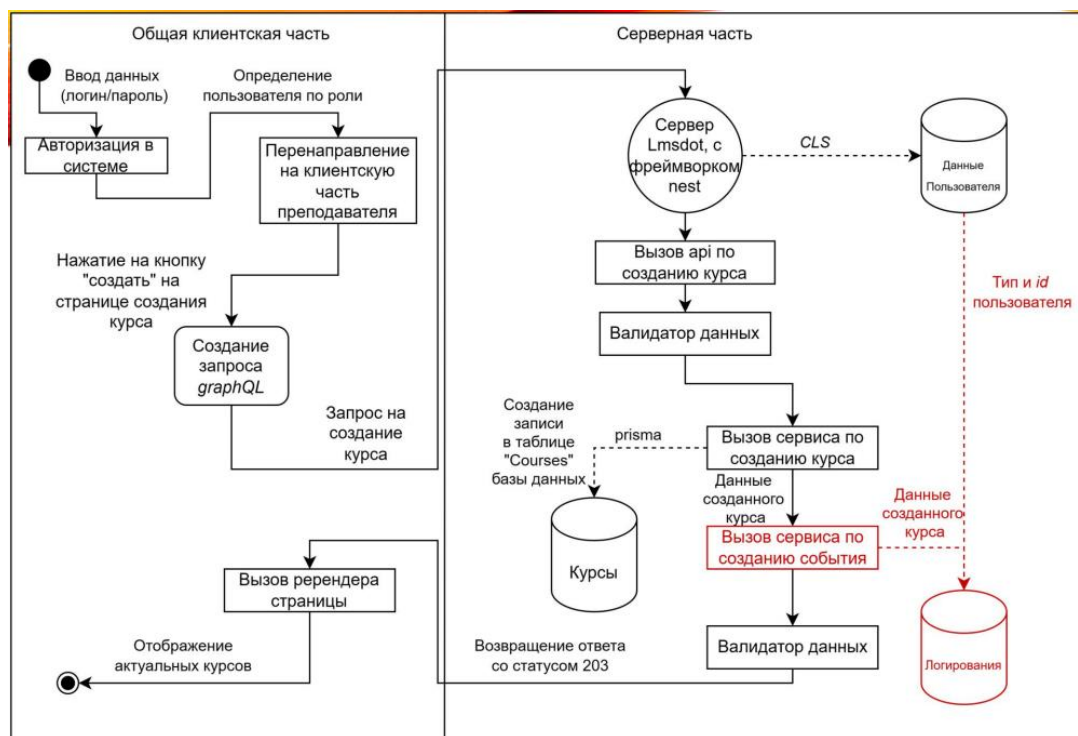


Рис. 3. Схема обработки запроса на создание курса, включающая модуль логирования, в системе Lmsdot

Интерфейс для отслеживания действий преподавателей

На страницах «курс» и «группа» добавлен новый раздел «История». На рисунке 4 приведён скриншот со страницы курса «ТВМС» с добавлением раздела «История».

В разделе «История» предоставляемая информация имеет шапку и тело. Шапка одинакова для страниц «курс» и «группа» и отображает: время совершения действия, инициатора совершенного действия, сущность (курс, группа, шаблон, работа) над которой произошло действие, тип события (создание, редактирование, активация и деактивация), цветовую иконку (характеризует тип) и иконку сущности (курс, группа, шаблон, работа).

Тело отображает информацию о свойствах сущности, над которой произошло действие. Информация отличается для страниц «курс» и «группа». Опишем каждую из них по отдельности.

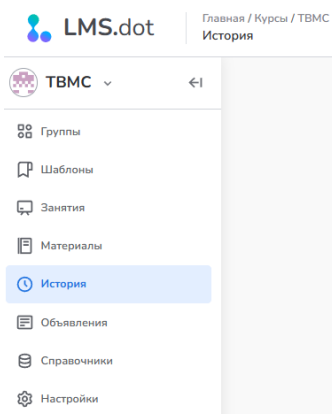


Рис. 4. Раздел «История» на странице курса «ТВМС»

Интерфейс на странице «Курс»

На странице «курс» в разделе «история» отображаются создание и редактирование курса, групп, а также шаблонов заданий и работ в этом курсе.

На рисунке 5 показан интерфейс событий, отображающий создание и редактирование курса «ТВМС». Видно, что курс был создан Сиденко Кириллом 5 июля 2023 года в 16:34. Редактирование краткого названия курса проводилось 5 июля 2023 года в 16:42.

На рисунке 6 показан интерфейс событий, отображающий создание группы «2015» и ее редактирование. Из рисунка видно, что группа была создана Сиденко Кириллом 5 июля 2023 года в 17:02. Редактирование проводилось 5 июля 2023 года в 17:03 и было изменено название группы.

На рисунках 7 и 8 показаны интерфейсы событий, отображающие создание и редактирование шаблона задания «Теоретический вопрос», соответственно. Из рисунка 7 видно, что шаблон задания был создан Сиденко Кириллом 5 июня 2023 года в 17:12 и включал один вариант. На рисунке 8 показано, что преподаватель Яровой Данила 5 июня 2023 года в 17:17 при редактировании добавил варианты «2» и «3» в задание.

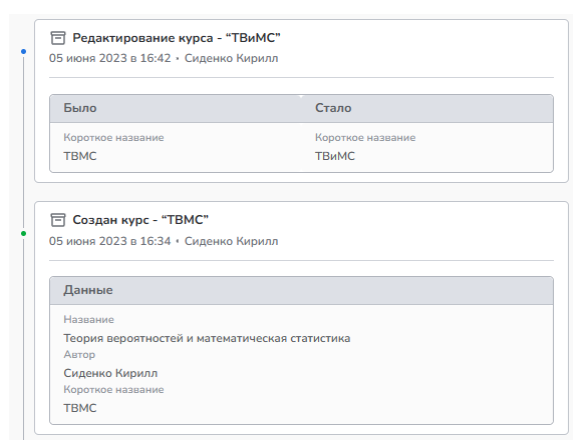


Рис. 5. Интерфейс событий. Создание и редактирование курса «ТВМС»

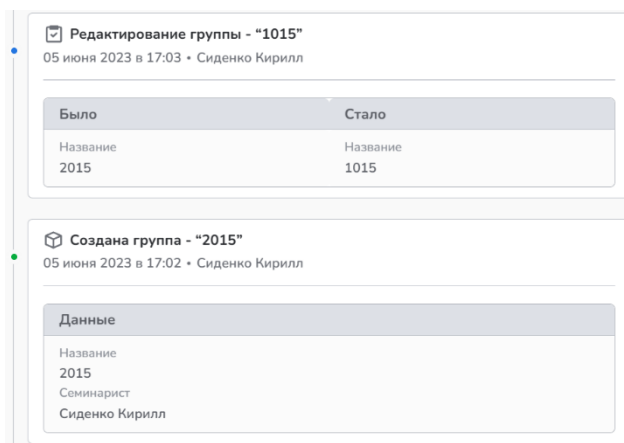


Рис. 6. Интерфейс события. Создание и редактирование группы «2015»

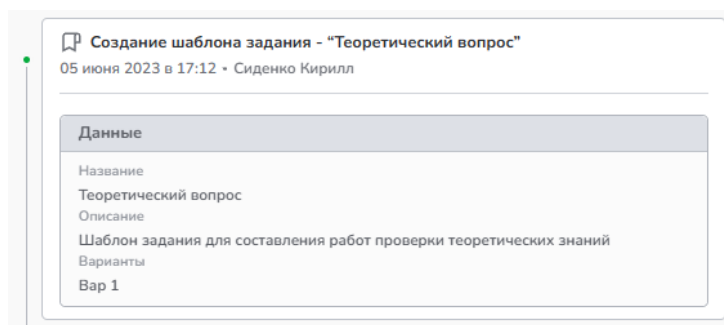


Рис. 7. Интерфейс события. Создание шаблона задания «Теоретический вопрос»

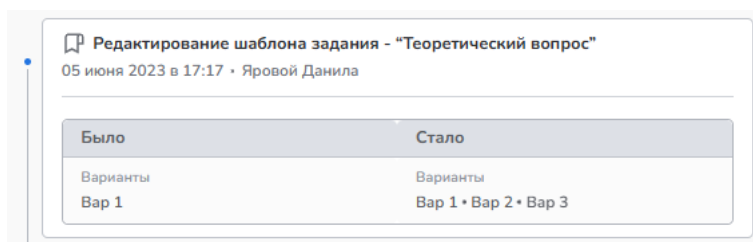


Рис. 8. Интерфейс события. Редактирование шаблона задания «Теоретический вопрос»

На рисунках 9 и 10 показаны интерфейсы событий, отображающих создание и редактирование шаблона работы «Теоретический опрос 1». Из рисунка 9 видно, что шаблон работы создан Сиденко Кириллом 5 июня 2023 в 17:41 и включал два задания (регулярные выражения и ДКП) с выбором первого варианта. На рисунке 10 показано, что преподаватель Яровой Данила 5 июня 2023 в 17:49 при редактировании изменил варианты заданий в работе.

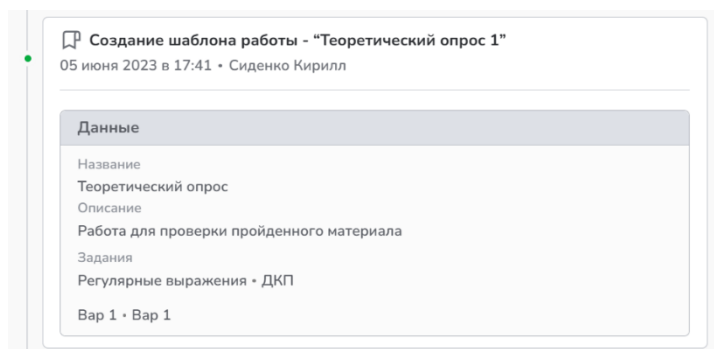


Рис. 9. Интерфейс события. Создание шаблона работы «Теоретический опрос 1»

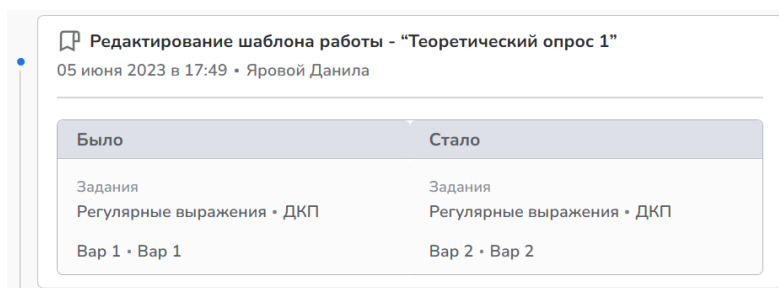


Рис. 10. Интерфейс события. Редактирование шаблона работы «Теоретический опрос 1»

Интерфейс на странице «Группа»

На странице «Группа» в разделе «История» повторяются события, связанные с созданием и редактированием группы, а также отображаются события, связанные с созданием, редактированием, активацией и деактивацией работ в этой группе.

На рисунке 11 отображены все события, связанный с работой. Из рисунка видно, что инициатором всех событий был преподаватель Яровой Данила. Он создал работу «Контрольная 12» 5 июня 2023 в 18:20, а также активировал работу для студентов 5 июня 2023 в 19:23. Преподаватель деактивировал работу для студентов 5 июня 2023 в 19:37 и редактировал варианты заданий в ней. Время активации и деактивации работы показывает, сколько времени продолжалась данная работа.

The screenshot displays a vertical timeline of events for a task titled "Контрольная 12". Each event is marked with a colored dot on the left and a checkmark icon. The events are as follows:

- Redaction of work - "Контрольная 12"** (05 июня 2023 в 19:47 • Яровой Данила): A table comparing the state before and after editing.

Было	Стало
Задания 1 • 2 • 3 • 4 • 5	Задания 1 • 2 • 3 • 4 • 5
Вар 1 • Вар 1 • Вар 1 • Вар 1 • Вар 1	Вар 2 • Вар 2 • Вар 2 • Вар 2 • Вар 2
- Deactivation of work - "Контрольная 12"** (05 июня 2023 в 19:37 • Яровой Данила)
- Activation of work - "Контрольная 12"** (05 июня 2023 в 19:13 • Яровой Данила)
- Creation of work - "Контрольная 12"** (05 июня 2023 в 18:20 • Яровой Данила): A table showing the initial data of the task.

Данные
Название Контрольная 12
Описание Алгоритмы Эрли
Задания 1 • 2 • 3 • 4 • 5
Вар 1 • Вар 1 • Вар 1 • Вар 1 • Вар 1

Рис.11. Интерфейс событий, отображающий действия с работой «Контрольная 12»

Интерфейс для отслеживания действий студентов

На страницах проверочной работы и занятий, можно посмотреть последовательности действий студентов при решении заданий. Действия можно просмотреть подробнее при нажатии кнопки «История изменений». В этом случае открывается раздел, где отображается состояния ответа студента. Информация с поля решения задания поступает через двадцать секунд с последнего нажатия клавиши. Если студент использовал функцию «вставка» из буфера обмена операционной системы, то отображаемая информация будет сопровождаться специальным сообщением. Если копируемая информация взята из решения задания, то появится сообщение «Вставка контента». В иных случаях копирования появится сообщение «Вставка внешнего контента». Отслеживание действий студентов прекращается, при отправке задания на проверку. На рисунке 12 изображена последовательность решения задачи студента «Ярославов Петр».

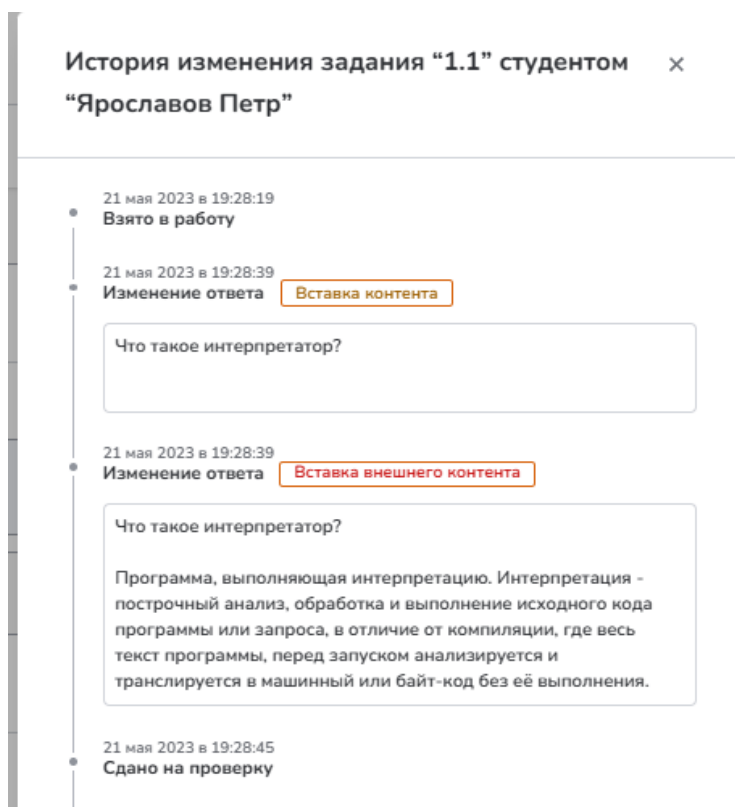


Рис. 12. Последовательность решения задания студентом Ярослав Петр

Из рисунка видно, что в процессе решения задания, студент использовал копирование два раза. В первом случае студент скопировал условие задачи и появилось сообщение «Вставка контента». Во втором случае произошло внешнее копирование текста, которое отмечено соответствующим сообщением «Вставка внешнего контента». В этом случае преподаватель получает однозначную информацию о списывании.

Проверка на практике

Внедрение модуля отслеживания действий было выполнено весной 2023 года. Эксплуатация разработанного модуля происходила в группах 2251–2255, 2221–2222 и 2181. Выполнялось ручное тестирование преподавателями Смирновым Д.П. и Дедович Т.Г. Были выражены пожелания к изменению интерфейса. Впоследствии доработки, интерфейс был улучшен и получен положительный отзыв.

Были выявлены случаи списывания, определены часто встречающиеся ошибки в действиях преподавателей, которые впоследствии помогут модернизировать систему *Lmsdot*. История в разделе «Курс» позволила упростить коммуникацию преподавателей при совместном создании шаблонов.

Заключение

В статье представлены результаты анализа модулей отслеживания действий в популярных *LMS* системах.

Для разработки модуля в системе *Lmsdot* составлены функциональные и нефункциональные требования, а также список всех отслеживаемых событий. Спроектирована диаграмма *ERD* таблицы логирования базы данных. Для реализации модуля была использована комбинация технологий, включая хранение информации пользователя в краткосрочном хранилище, объектно-реляционное связывание и визуализацию результатов. Разработан интерфейс для отслеживания действий преподавателей на страницах «Курс» и «Группа», а также действий студентов на страницах заданий и контрольных работ.

Модуль был разработан и интегрирован в существующую систему *Lmsdot*. Преподавателями Смирновым Д.П. и Дедович Т.Г. во время преподавания курса «Математическая логика и теория алгоритмов» проведено тестирование модуля. Получен положительный отзыв.

Список источников

1. Moodle – Open-Source learning platform | Moodle.org. – URL: <https://moodle.org/> (дата обращения: 22.04.2023).
2. Программы для онлайн-обучения iSpring. – ООО «Ричмедиа», 2001-2023. – URL: <https://www.ispring.ru/> (дата обращения: 22.04.2023).
3. Система дистанционного обучения Mirapolis LMS. – 2 Mirapolis, 2023. – URL: <https://www.mirapolis.ru/lms/> (дата обращения: 22.04.2023).
4. Платформа для обучения Teachbase | Тичбейс. – ООО «Интернет Школа», 2009 – 2023. – URL: <https://teachbase.ru/> (дата обращения: 22.04.2023).
5. Система Дистанционного Обучения. – URL: <http://lmsdot.com> (дата обращения: 15.03.2023).
6. Кочешков А. Д., Смирнов Д. П., Дедович Т. Г. Разработка информационной системы контроля и оценки знаний студентов по математическим дисциплинам в университете «Дубна» // Системный анализ в науке и образовании. – 2021. – № 2. – С. 140-150.
7. Смирнов Д. П., Дедович Т. Г. Анализ и выбор современных инструментов и технологий для разработки системы управления обучением // Системный анализ в науке и образовании. – 2022. – № 1. – С. 105-116.
8. Кузнецов М. М, Смирнов Д. П., Дедович Т. Г. Разработка модуля импорта заданий системы контроля и оценки знаний студентов Lmsdot // Системный анализ в науке и образовании. – 2023. – № 2. – С. 97-107.
9. PostgreSQL: The World's Most Advanced Open Source Relational Database. –The PostgreSQL Global Development Group, 1996-2023. URL: <https://www.postgresql.org/> (дата обращения: 15.03.2023).
10. Prisma | Simplify working and interacting with databases. – Prisma Data, Inc., 2023. – URL: <https://www.prisma.io/> (дата обращения: 17.03.2023).
11. Ефимова М., Зуйкова А. Как устроен язык SQL и почему он так востребован // Блог Яндекс Практикума. – АНО ДПО «Образовательные технологии Яндекса», ООО «Яндекс», 2022. – Дата публикации: 27 июля 2022. – URL: <https://practicum.yandex.ru/blog/что-такое-sql/> (дата обращения: 15.03.2023).
12. JavaScript | MDN. – URL: <https://developer.mozilla.org/en-US/docs/Web/JavaScript> (дата обращения: 18.02.2023).
13. Typescript: JavaScript with syntax for types. – Microsoft, 2012-2023. – URL: <https://www.typescriptlang.org/> (дата обращения: 18.02.2023).
14. NestJS – A progressive Node.js framework. Kamil Mysliwiec, 2017-2023. URL: <https://nestjs.com/> (дата обращения: 17.03.2023).
15. cls-ts npm : Continuation-Local Storage (Typescript). – URL: <https://www.npmjs.com/package/cls-ts> (дата обращения: 20.03.2023).
16. solidjs.com : [проект с открытым исходным кодом]. – URL: <https://www.solidjs.com/> (дата обращения: 01.04.2023).

17. GraphQL | A query language for your API. – The GraphQL Foundation, 2023. – URL: <https://graphql.org/> (дата обращения: 17.03.2023).