

DOI: 10.32517/0234-0453-2021-36-9-55-62

ОПЫТ РАЗРАБОТКИ ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ МОДЕЛИРОВАНИЯ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ С ФУНКЦИЕЙ ПРЯМОГО ИНЖИНИРИНГА ДЛЯ ОБУЧЕНИЯ СТУДЕНТОВ ТЕХНИЧЕСКИХ СПЕЦИАЛЬНОСТЕЙ

М. В. Смирнов¹ ✉, В. М. Поленок¹

¹ МИРЭА — Российский технологический университет, г. Москва, Россия

✉ smirnov_m@mirea.ru

Аннотация

В статье актуализируется необходимость разработки программного обеспечения для моделирования реляционных баз данных для использования в процессе обучения студентов технических специальностей по дисциплинам, связанным с базами данных.

Проблема рассматривается с точки зрения оценки современного программного обеспечения, используемого в процессе обучения студентов навыкам проектирования баз данных. На основании выявленных в ходе обзора программного обеспечения недостатков определен ряд требований к актуальному программному обеспечению. Сформированные ключевые требования — это мобильность, доступность, универсальность и открытость платформы разработки.

В статье описывается процесс решения ключевых задач, возникших в ходе реализации проекта разработки веб-приложения для моделирования реляционных баз данных в соответствии с сформированными требованиями. Последовательно рассмотрена практическая реализация следующих функций: создание логической реляционной модели данных, создание физической модели данных, прямой инжиниринг в программное обеспечение реляционных баз данных. Описаны основные технологические решения, примененные в ходе разработки веб-приложения для обеспечения заданных в условии качества.

Итогом работы является успешная апробация результатов разработки в процессе создания реального веб-приложения — как в рамках лабораторных и практических работ дисциплин «Проектирование и администрирование баз данных» и «Управление данными», так и на этапе написания дипломных работ для технических направлений обучения.

Ключевые слова: реляционные базы данных, веб-приложение баз данных, веб-моделирование, проектирование баз данных, прямой инжиниринг, нотация Чена, нотация DEF1x, ИТ-проект.

Для цитирования:

Смирнов М. В., Поленок В. М. Опыт разработки веб-приложения для моделирования реляционных баз данных с функцией прямого инжиниринга для обучения студентов технических специальностей. *Информатика и образование*. 2021;36(9):55–62. DOI: 10.32517/0234-0453-2021-36-9-55-62

EXPERIENCE IN DEVELOPING A WEB APPLICATION FOR RELATIONAL DATABASES MODELING WITH A FORWARD ENGINEERING FUNCTION FOR TRAINING STUDENTS OF TECHNICAL SPECIALTIES

M. V. Smirnov¹ ✉, V. M. Polenok¹

¹ MIREA — Russian Technological University, Moscow, Russia

✉ smirnov_m@mirea.ru

Abstract

The article actualizes the need to develop software for modeling relational databases for use in the process of teaching students of technical specialties in disciplines related to databases.

The problem is considered from the point of view of assessing modern software used in the process of teaching students database design skills. Based on the shortcomings identified during the software review, a number of requirements for the actual software were determined. Formed key requirements are mobility, accessibility, versatility and openness of the development platform.

The article describes the process of solving key problems that arose during the implementation of a project to develop a web application for modeling relational databases in accordance with the generated requirements. The practical implementation of the following functions is sequentially considered: creation of a logical relational data model, creation of a physical data model, direct engineering into relational database software. The main technological solutions used in the development of a web application to ensure the qualities specified in the condition are described.

The result of the work is the successful testing of the development results in the process of creating a real web application, both within the framework of laboratory and practical work in the disciplines “Design and administration of databases” and “Data management”, and at the stage of writing graduate works for technical directions of training.

Keywords: relational databases, web database application, web modeling, database design, direct engineering, Chen notation, IDEF1x notation, IT project.

For citation:

Smirnov M. V., Polenok V. M. Experience in developing a web application for relational databases modeling with a forward engineering function for training students of technical specialties. *Informatics and Education*. 2021;36(9):55–62. DOI: 10.32517/0234-0453-2021-36-9-55-62 (In Russian.)

1. Актуальность проблемы разработки веб-приложения для моделирования баз данных для обучения студентов технических специальностей

Учебные дисциплины, связанные с проектированием информационных систем и баз данных, входят во все учебные планы по подготовке бакалавров технических специальностей. Неотъемлемой частью преподавания этих дисциплин является разработка студентами прикладного программного обеспечения для различных предметных областей.

Для классического проектирования традиционных реляционных баз данных обычно достаточно знания основ языка SQL, а также основных принципов и правил реляционной модели данных. Вместе с этим сам процесс написания кода, реализации структуры хранения данных зачастую оказывается для студента излишне трудоемким не только на этапе создания физической модели, но даже в большей степени на этапе исправления ошибок и оптимизации модели.

Данная проблема актуальна не только для процесса обучения разработке программного обеспечения, но и для процесса реализации реальных проектов по созданию приложений баз данных и информационных систем. Поэтому еще в 90-х годах прошлого века начали появляться CASE-средства моделирования данных, призванные максимально автоматизировать и облегчить процесс создания модели данных для последующей интеграции ее с соответствующим программным обеспечением. Такие CASE-средства активно используются как в университетских программах обучения, так и в реальных проектах разработки программного обеспечения баз данных и в настоящее время. Вместе с тем существует ряд ограничений и проблем, связанных с их эксплуатацией, особенно в программах обучения университетов. В первую очередь это высокая стоимость программного обеспечения, привязка к определенным операционным системам и средам, ограниченный бесплатный функционал, возможность инжиниринга результатов моделирования только в определенное ПО БД.

В данной статье будут сформулированы основные проблемы современных CASE-средств моделирования данных с точки зрения применения их в процессе обучения студентов в рамках направления «Прикладная информатика», а также описаны процесс

формулирования требований к веб-приложению логического моделирования данных с функцией прямого инжиниринга физической модели данных и реализация такого приложения.

Проблематика упрощения процесса проектирования баз данных и создания рабочих физических моделей баз данных при реализации проектов разработки программного обеспечения, в том числе в ходе обучения студентов технических специальностей, в последние годы поднималась в работах А. В. Гаврилова [1, 2], Н. В. Титовской, С. Н. Титовского [3], Y. Zuoyou [4], Katz A. [5], N. Roy-Hubara, A. Sturm [6] и других ученых.

Тематикой перечисленных работ являются проблемы, связанные с дальнейшим изучением реляционной модели хранения данных [4, 6–8] и баз данных, построенных с использованием этой модели. Отдельное внимание в трудах ученых уделяется современному программному обеспечению моделирования баз данных. Также поднимаются вопросы автоматизации создания реляционных моделей данных и разработки моделей данных для решения задач в разных предметных областях [6, 8].

2. Анализ актуального программного обеспечения для моделирования баз данных

Проблема подбора программного обеспечения для проектирования баз данных подробно рассматривается в статье А. В. Гаврилова «Анализ функциональных возможностей бесплатных CASE-средств проектирования баз данных» [2]. Автор приходит к мнению, что в условиях чрезвычайной дороговизны качественного промышленного программного обеспечения проектирования баз данных (ErWin Data Modeler) актуальными являются поиск и применение свободного (free software) и бесплатного (freeware) программного обеспечения. А. В. Гаврилов приводит подробный обзор и сравнительный анализ функциональных возможностей актуального программного обеспечения по теме:

- MySQL Workbench;
- Valentina Studio;
- dbForge Studio for MySQL;
- SQLyog Community;
- EMS SQL Management Lite Studio;
- Heidi SQL.

Перечисленное программное обеспечение было опробовано авторами настоящей статьи в ходе проведения практических и лабораторных работ в дисциплинах, связанных с проектированием баз данных. Это программное обеспечение в целом обладает достаточным для моделирования баз данных функционалом (соединение с серверами баз данных, инструментарий для написания инструкций SQL на создание и изменение элементов базы данных, инструменты логического моделирования данных, средства импорта/экспорта данных) [9, 10].

Однако в ходе практического применения указанного программного обеспечения был выявлен ряд недостатков, создающих неудобства в процессе обучения студентов. Так, перечисленное программное обеспечение, за исключением EMS SQL Management Lite Studio, не имеет русскоязычного интерфейса. Важной для осуществления эффективной работы по проектированию баз данных опцией создания физической модели базы данных на сервере (forward engineering) обладают только три из перечисленных программных средств: MySQL Workbench [11], Valentina Studio [12], dbForge Studio [13]. Такими ключевыми для процесса обучения проектированию баз данных функциями, как сохранение схемы базы данных в графическом формате (или в любом другом универсальном формате для передачи и удобного чтения) и отображения модели данных в различных нотациях обладает лишь MySQL Workbench. В свою очередь, при всех очевидных преимуществах применения при проектировании программного решения MySQL Workbench Community Edition выявило три существенных недостатка:

- невозможность расширения функционала программного средства;
- ориентация данного программного продукта на работу в среде традиционных операционных систем (вариации систем Windows, Linux);
- ограничение возможности создания и администрирования баз данных лишь программным обеспечением Oracle MySQL.

Помимо перечисленного выше программного обеспечения для создания моделей данных добавим к рассмотрению современное веб-приложение Pony Object Relational Mapper [14]. Данное приложение ближе всего соответствует требованиям, предъявляемым к ПО для обучения студентов базам данных. Pony Object Relational Mapper — это бесплатное приложение, оно обладает кроссплатформенностью и мобильностью, позволяет строить физические модели данных и сохранять их на сервере, а также предоставляет средства прямого инжиниринга моделей в ключевые базы данных. Вместе с тем интерфейс программного средства представлен на английском языке, а инструментарий для построения моделей ограничен средствами физического моделирования нотации IDEF1x.

По итогам проведенного исследования было принято решение о разработке собственной версии программного обеспечения для проектирования баз

данных, обладающего необходимыми свойствами для проведения эффективных практикумов и лабораторных работ студентов бакалавриата (направление «Прикладная информатика») в области проектирования баз данных.

3. Постановка задачи на разработку веб-приложения для моделирования реляционных баз данных

Для разрабатываемого программного обеспечения для проектирования баз данных были сформулированы следующие ключевые требования:

- наличие ключевых функций существующих программных средств проектирования баз данных:
 - возможность создания логических моделей баз данных;
 - возможность хранения и изменения логических моделей баз данных;
 - функция прямого инжиниринга логической модели в физическую модель данных;
- удобный минималистичный графический интерфейс программного средства, реализованного в виде веб-приложения (универсализм, мобильность и простота доступа к работе — это одни из ключевых требований к современному программному обеспечению, используемому в процессе обучения);
- обеспечение возможности легкого доступа к дальнейшему расширению функционала базового программного обеспечения;
- реализация функции автоматического сохранения и удобной визуализации результатов моделирования (для удобства создания студентами отчетов о проведенной в рамках проектирования работе);
- реализация нескольких типов (нотаций) логического моделирования данных;
- увеличение количества программного обеспечения СУБД, с которыми может взаимодействовать программное обеспечение проектирования баз данных (в рамках функции прямого инжиниринга);
- использование в технологическом стеке разработки программ и технологий, допускающих использование конечного продукта в формате Open Source.

4. Технические аспекты реализации веб-приложения для моделирования реляционных баз данных

В ходе выполнения технического задания по разработке программного обеспечения для проектирования баз данных потребовалось решить ряд задач, связанных со спецификой реализации веб-приложения.

Выделим четыре наиболее трудоемкие и ключевые задачи:

- 1) Создание в приложении инструментария, соответствующего выбранной пользователем нотации моделирования баз данных.
- 2) Приложение должно предоставлять возможность пользователю создавать и загружать на веб-сервер (сохранять в своем рабочем пространстве) готовую модель базы данных в одной из выбранных им нотаций.
- 3) Приложение должно иметь возможность по запросу пользователя выгрузить для продолжения работы ранее созданную и сохраненную на сервере модель (предварительно предоставив выбор пользователю среди моделей, с которыми он когда-либо работал).
- 4) Приложение должно предоставлять пользователю возможность осуществить функцию прямого инжиниринга созданной модели в скрипт одного из SQL-диалектов и исполнить этот скрипт на сервере баз данных, создав физическую модель базы.

Далее будут последовательно раскрыты аспекты решения этих четырех задач.

4.1. Создание инструментария моделирования

При инициализации модели, с которой будет работать пользователь (после задания имени и типа диаграммы), веб-приложение выгружает с сервера набор интерактивных объектов в виде шаблонов HTML5 [15] в соответствии с выбранной пользователем нотацией. Данный набор интерактивных объектов, хранимый на сервере, легко обновляется и дополняется, что позволит в перспективе усложнить существующие и добавить новые варианты нотаций моделирования данных. Каждый интерактивный объект олицетворяет собой один из элементов нотации (сущность, атрибут, связь и т. д.) и в зависимости от своего типа и логики реляционной модели данных, соответственно, имеет собственный набор атрибутов, например, таких как:

- имя;
- позиция на холсте;
- тип данных (для атрибутов);
- длина (для атрибутов) и т. д.

После выбора объекта его атрибуты заполняются пользователем веб-приложения в соответствии с поставленной задачей моделирования данных.

При создании информационной модели из этих выгруженных с сервера объектов формируется HTML-документ, по структуре напоминающий традиционный документ языка XML. Наглядный пример формирования объекта Test (сущность) и неключевого атрибута этой сущности Phone с типом данных varchar показан на рисунке 1.

4.2. Загрузка готовой модели на веб-сервер (сохранение модели)

При сохранении модели на сервере полученная «схема» модели баз данных при помощи скриптов, написанных на языке JavaScript, конвертируется в формат документа JSON [16], готового к передаче данных на хранение серверу. С помощью технологии AJAX [17] подготовленные данные динамически передают созданную схему на хранение серверу. Таким образом реализуется незаметная и удобная пользователю функция динамического сохранения модели. Это особенно актуально для случаев, когда модель большая по объему или когда с моделью работает коллектив разработчиков (студенческая команда). На рисунке 2 представлен фрагмент «схемы» модели данных test, содержащей сущности, атрибуты и связи, в формате документа JSON.

На веб-сервере приложения полученный документ JSON средствами языка PHP преобразуется в ассоциативный массив, после чего помещается на хранение в базу данных на сервере.

4.3. Выгрузка модели в веб-приложение с сервера

Выгрузка модели осуществляется в порядке, обратном ее сохранению. Схематично весь процесс показан на рисунке 3.

4.4. Прямой инжиниринг логической модели данных в физическую модель

На основании сформированной «схемы» средствами языка JavaScript [18] на стороне клиента происходит синтаксическое формирование SQL-запроса.

```
▶<div class="box main ui-draggable ui-draggable-handle" id="1001570544210037" style="top: 297px; left: 444px;">...</div>
▼<div class="box main ui-draggable ui-draggable-handle" id="6541570544323813" style="top: 80px; left: 196px;">
  <div class="main_text">Test</div>
  ▶<div class="options">...</div>
</div>
▼<div class="box attribute ui-draggable ui-draggable-handle" id="4821570544293134" parent="1001570544210037" data_type="varchar" len_data="250" primary_key="false" style="top: 140px; left: 601px;">
  <div class="attribute_text">Phone</div>
  ▶<div class="options">...</div>
</div>
```

Рис. 1. Фрагмент «схема» модели базы данных. Описание сущности и ее атрибута
Fig. 1. Fragment "scheme" of the database model. Description of the entity and its attribute

```
"6541570544323813": {
  "position": {
    "top": 75,
    "left": 19},
  "name": "Test"}},
"attributes": {
  "4821570544293134": {
    "parent": "1001570544210037",
    "data_type": "varchar",
    "len_data": "250",
    "primary_key": "false",
    "name": "Phone",
    "position": {
      "top": 135,
      "left": 424.0000305175781}}...
```

Рис. 2. Документ — описание сущности и атрибутов таблицы

Fig. 2. Document — description of the entity and table attributes

Формирование SQL-запроса происходит в несколько этапов:

- 1) выборка из схемы всех сущностей;
- 2) выборка из схемы атрибутов для каждой сущности;
- 3) выборка из схемы описания связей.

При нахождении в схеме требуемой сущности из объекта извлекается имя сущности и создается начало SQL-инструкции CREATE по шаблону: «CREATE TABLE имя_сущности».

Далее происходит выборка всех столбцов, связанных с описанием сущности. Формируется строка столбца вида: «название_столбца тип_данных (пользовательское ограничение)». В случае если столбец отмечается пользователем как первичный ключ, в строку добавляется описание ограничения первичного ключа вида «название_столбца тип_данных (пользовательское ограничение) NOT NULL CONSTRAINT PK_название_столбца PRIMARY KEY (название_столбца)».

На последнем этапе инжиниринга инструкции SQL происходит выборка бинарных связей между таблицами. Формируется ограничение внешнего ключа вида «название_внешнего_ключа тип_данных (пользовательское ограничение) CONSTRAINT FK_название_родительской_таблицы_название_дочерней_таблицы FOREIGN KEY (название_внешнего_ключа _FK) REFERENCES название_родительской_таблицы (название_ключевого_атрибута_родительской_таблицы)».

Сформированные элементы запроса конкатенируются в одну строку инструкции. Конечный набор инструкций в совокупности представляет собой скрипт, реализующий структуру физической модели данных на сервере СУБД.

5. Демонстрации результатов работы с веб-приложением для моделирования реляционных баз данных

В рамках мероприятий по тестированию разработанного программного обеспечения draw.db [19] был осуществлен полный цикл проектирования фи-

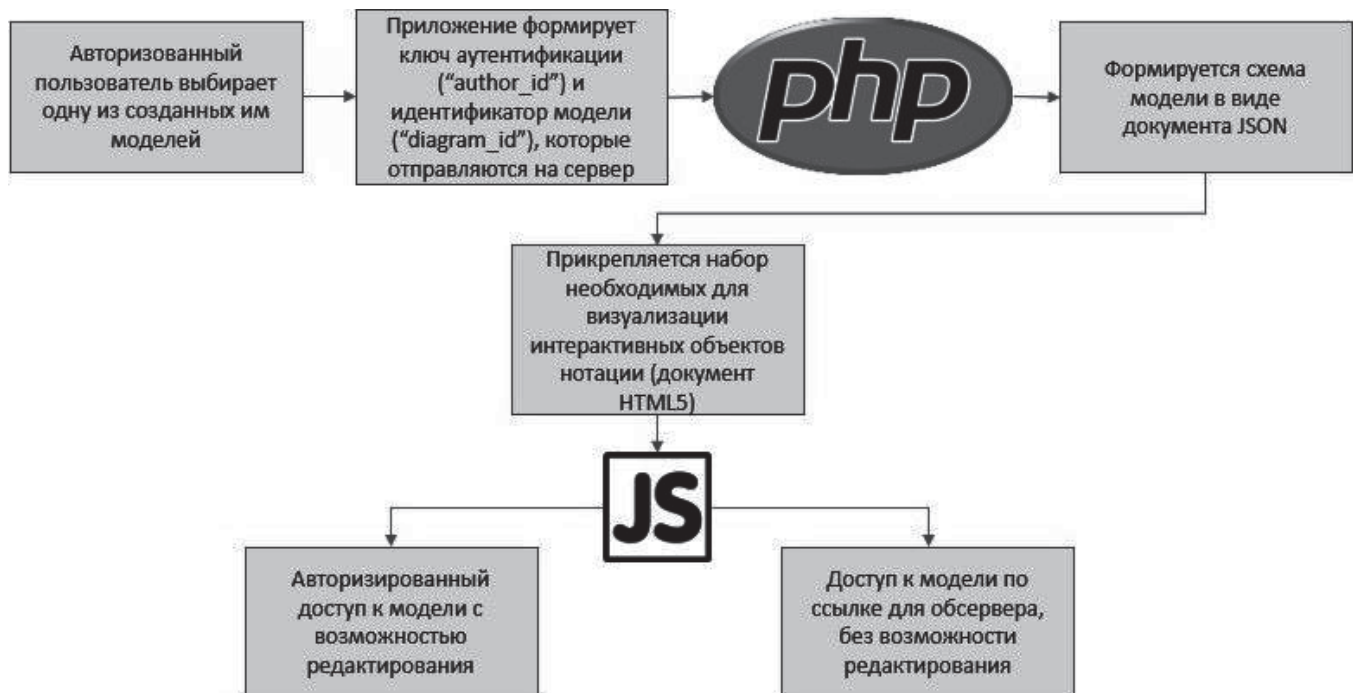


Рис. 3. Процесс загрузки модели данных с сервера веб-приложения

Fig. 3. The process of loading the data model from the web application server

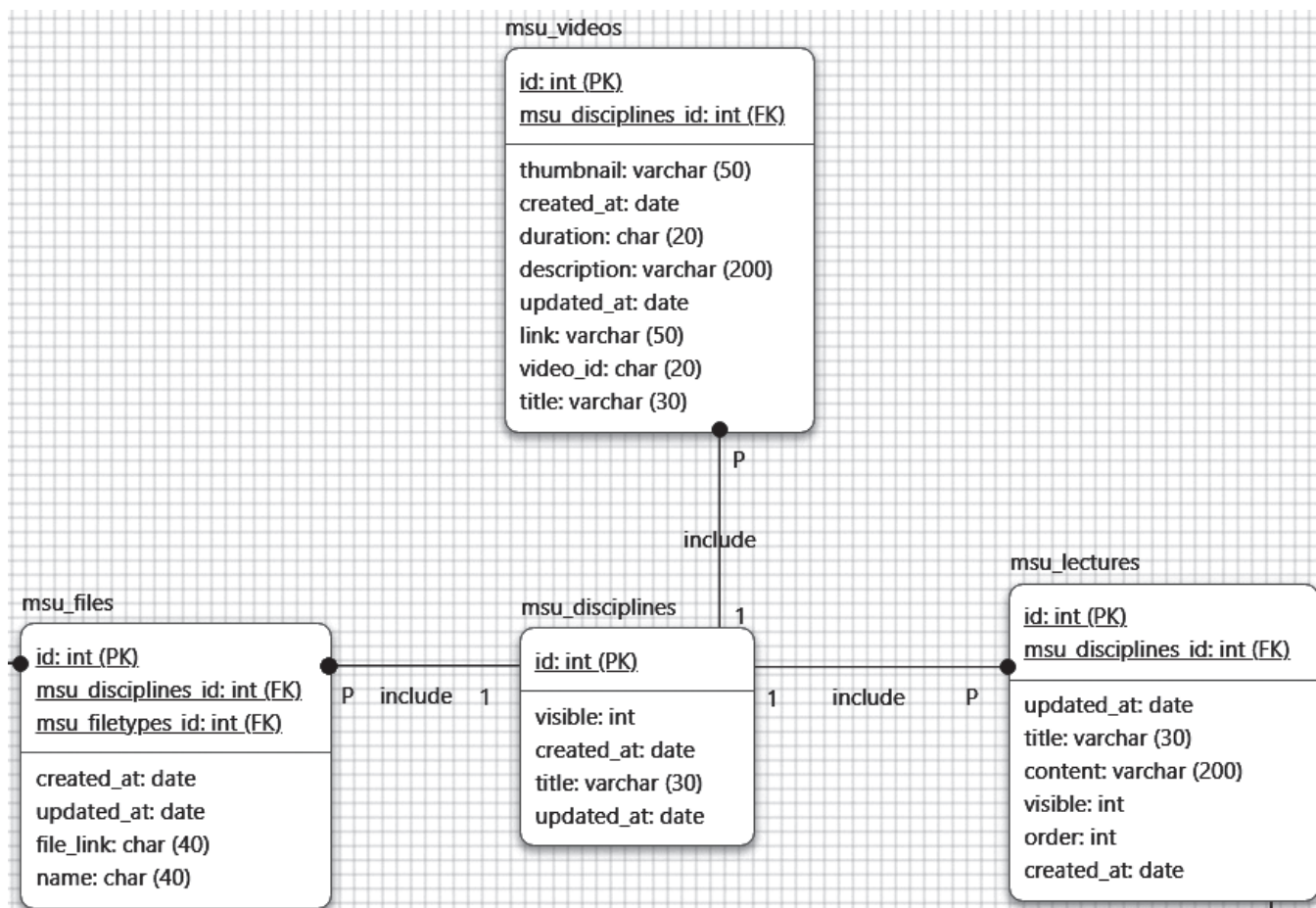


Рис. 4. Физическая модель данных приложения, реализованная в веб-приложении draw.db
 Fig. 4. Physical data model of the application, implemented in the web application draw.db

зической модели реляционной модели данных для веб-приложения MSUniversity [20]. Модель данных была составлена в приложении в нотации IDEF1x. Ключевой фрагмент модели показан на рисунке 4. Приведенная модель содержит:

- контейнеры таблиц (msu_videos, msu_files, msu_disciplines, msu_lectures);
- выделенные кластеризованные индексы первичного ключа;
- ограничения внешнего ключа;
- описания столбцов таблиц с их типами данных и пользовательскими ограничениями;
- графическую визуализацию кардинальностей связей между таблицами с пояснением логики связи в глагольном наклонении.

Как было описано выше, с помощью JSON-документа схема при сохранении ее пользователем переносится на сервер веб-приложения и в любой момент может быть загружена в любой браузер для дальнейшего редактирования и уточнения ее параметров.

Также в качестве проверки качества приложения было осуществлено логическое моделирование того же веб-приложения в нотации Чена. Результат моделирования в виде фрагмента модели показан на рисунке 5.

Этап прямого инжиниринга был проведен для физической модели приложения, показанной на рисунке 4. Фрагмент сгенерированного скрипта кода T-SQL с инструкцией CREATE для таблицы базы данных msu_files показан на рисунке 6.

Дальнейшая успешная реализация сформированного скрипта в СУБД MS SQL Server 2019 доказала эффективность и качество реализованной функции прямого инжиниринга в разработанном веб-приложении.

6. Выводы и дальнейшие перспективы разработки

В 2019 году прототип разработанного программного средства был использован студенческой группой МИРЭА в ходе подготовки к участию и в ходе участия во Всероссийской олимпиаде студентов образовательных организаций высшего образования по направлению подготовки «Прикладная информатика» (ВСО ПИ-2019) на базе Российского экономического университета имени Г. В. Плеханова. Итогом выступления стали призовые места.

Разработанное программное обеспечение в течение 2020/2021 учебного года успешно применялось в дисциплинах «Проектирование и администриро-

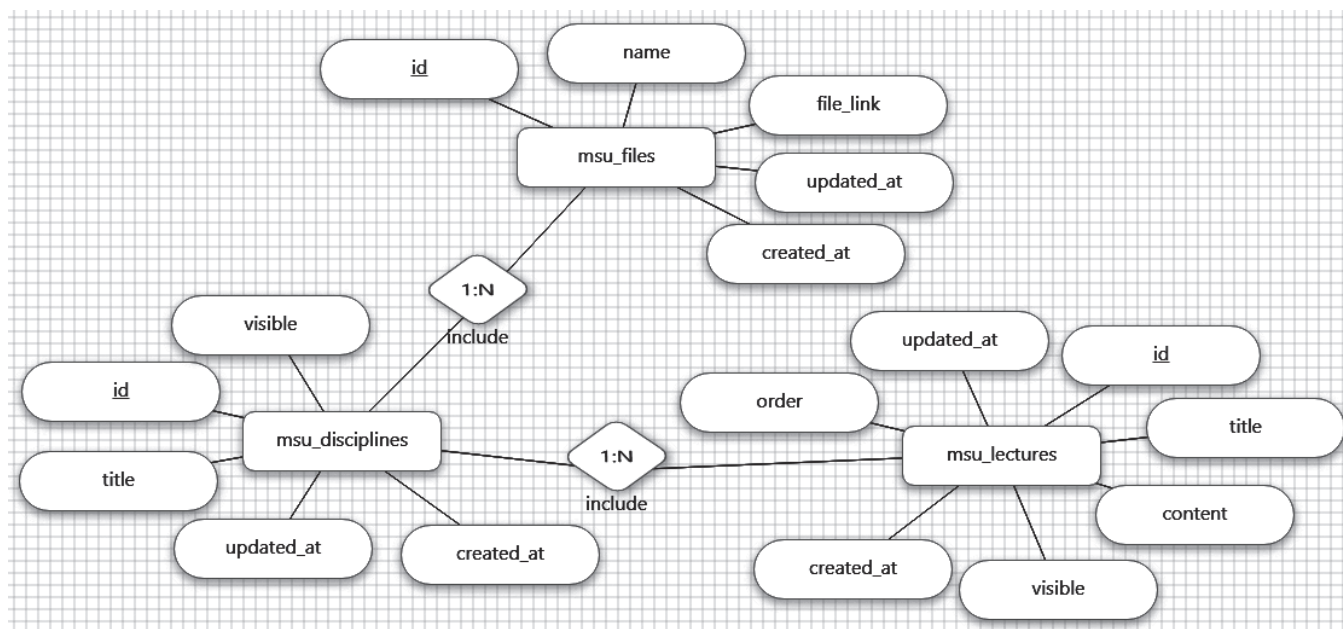


Рис. 5. Логическая модель данных, реализованная в веб-приложении draw.db

Fig. 5. Logical data model implemented in draw.db web application

Запрос для выполнения на сервере:

```

62 );
63
64
65 CREATE TABLE msu_files (
66   id int NOT NULL CONSTRAINT PK_id PRIMARY KEY(id),
67   msu_disciplines_id int,
68   msu_filetypes_id int,
69   name char(40),
70   file_link char(40),
71   created_at date,
72   updated_at date,
73   id_FK int,
74   CONSTRAINT FK_msu_disciplines_msu_files FOREIGN KEY (id_FK) REFERENCES msu_disciplines
75     (id),
76   id_FK int,
77   CONSTRAINT FK_msu_filetypes_msu_files FOREIGN KEY (id_FK) REFERENCES msu_filetypes(id)
78 );
79
80 CREATE TABLE msu_filetypes (
81   id int NOT NULL CONSTRAINT PK_id PRIMARY KEY(id),
82   name char(40),
83   file_link char(40),
84   created_at date,
85   updated_at date,
86   id_FK int,
87   CONSTRAINT FK_msu_disciplines_msu_filetypes FOREIGN KEY (id_FK) REFERENCES msu_disciplines
88     (id),
89   id_FK int,
90   CONSTRAINT FK_msu_filetypes_msu_filetypes FOREIGN KEY (id_FK) REFERENCES msu_filetypes(id)
91 );

```

Результат выполнения запроса на сервере:

1

Рис. 6. Демонстрация функции прямого инжиниринга в веб-приложении draw.db

Fig. 6. Demonstration of the direct engineering function in the draw.db web application

вание баз данных» и «Управление данными» в МИ-РЭА — Российском технологическом университете. Удобство применения и соответствие поставленным требованиям были доказаны в ходе лабораторных и самостоятельных работ студентов. В 2021/2022 учебном году программное обеспечение апробируется в ходе практических работ магистерского курса «Технологии проектирования информационных систем» в качестве CASE-средства разработки фи-

зической модели данных для курсового проекта. Разработанное программное обеспечение показало свою эффективность на этапе написания и защиты выпускных квалификационных работ бакалавров в 2020 и 2021 годах.

В ходе эксплуатации разработанного программного средства был выявлен недостаток, связанный с недостаточной аппаратной мощностью серверной части программного продукта. В настоящее время

серверное аппаратное обеспечение позволяет запускать ограниченное количество параллельных сессий (не более 17–20 одновременных сессий). В связи с этим в 2022 году планируется переход на более мощный сервер МИРЭА для существенного увеличения количества возможных одновременных подключений.

Поскольку открытая архитектура программного средства создает предпосылки для дальнейшей модернизации функционала, исследовательским коллективом уже ведутся работы по проектированию и разработке модуля с типовыми паттернами реляционных моделей баз данных для разных предметных областей разработки программного обеспечения. Также ведутся научные исследования в области моделирования документных NoSQL баз данных.

Список источников / References

1. Гаврилов А. В. Использование современных CASE-средств структурного проектирования при обучении студентов по направлению подготовки «прикладная информатика». *Открытое образование*. 2015;(4):22–27. DOI: 10.21686/1818-4243-2015-4(111-22-27)
2. Гаврилов А. В. The use of modern case-tools of structural design in teaching students on the bachelor program “Applied informatics”. *Open education*. 2015;(4):22–27. DOI: 10.21686/1818-4243-2015-4(111-22-27)
3. Гаврилов А. В. Анализ функциональных возможностей бесплатных CASE-средств проектирования баз данных. *Открытое образование*. 2016;20(4):39–43. DOI: 10.21686/1818-4243-2016-4-39-43
4. Гаврилов А. В. Analysis of functionality free CASE-tools databases design. *Open education*. 2016;20(4):39–43. DOI: 10.21686/1818-4243-2016-4-39-43
5. Титовская Н. В., Титовский С. Н. Методика обучения будущих IT-специалистов проектированию и разработке баз данных на основе интерактивного подхода. *Вестник Красноярского государственного педагогического университета им. В. П. Астафьева*. 2019;(4):75–87. DOI: 10.25146/1995-0861-2019-50-4-164
6. Titovskaya N. V., Titovsky S. N. Technique of teaching design and development of databases to future information technology experts. *The bulletin of the Krasnoyarsk State Pedagogical University named after V. P. Astafev*. 2019;(4):75–87. DOI: 10.25146/1995-0861-2019-50-4-164
7. Zuoyou Yin. Database design on rotating classroom teaching resources system based on DB2. *Proc. 2nd Int. Conf. on Materials Science, Machinery and Energy Engineering (MSMEE 2017)*. Atlantis Press; 2017:1584–1588. DOI: 10.2991/msmee-17.2017.285
8. Katz Adi. Improved teaching of database schema modeling by visualizing changes in levels of abstraction. *Journal of Information Systems Education*. 2020;31(4):294–311. Available at: <http://jise.org/Volume31/n4/JISEv31n4p294.html>
9. Roy-Hubara N., Sturm A. Design methods for the new database era: a systematic literature review. *Software and System Modeling*. 2020;19:297–312. DOI: 10.1007/s10270-019-00739-8
10. Yubei Lin. Student-centered teaching design and practice of database course. *Advances in Social Science, Education and Humanities Research*. 2020;450:149–153. DOI: 10.2991/assehr.k.200727.033
11. Li Yang, Li Cao. The effect of MySQL Workbench in teaching entity-relationship diagram (ERD) to relational schema mapping. *International Journal of Modern Education and Computer Science (IJMECS)*. 2016;8(7):1–12. DOI: 10.5815/ijmeecs.2016.07.01
12. Дейт К. Дж. Введение в системы баз данных. 8-е изд. М.: Вильямс; 2006. 1328 с.
13. Date C. J. Introduction to database systems. Moscow, Williams; 2006. 1328 p.
14. Коннолли Т., Бегг К. Е. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. 3-е изд. М.: Вильямс; 2003. 1436 с.
15. Connolly T., Begg C. E. Database systems: A practical approach to design, implementation, and management. Moscow, Williams; 2003. 1436 p.
16. MySQL Workbench. Available at: <https://www.mysql.com/products/workbench/>
17. Introducing Valentina Studio best free database management software. Available at: <https://valentina-db.com/en/valentina-studio-overview>
18. dbForge Studio for SQL Server. Available at: <https://www.devart.com/ru/dbforge/sql/studio/>
19. PonyORM — Python ORM with beautiful query syntax. Available at: <https://ponyorm.org/>
20. HTML Standard. Available at: <https://html.spec.whatwg.org>
21. JSON. Available at: <http://json.org/json-ru.html>
22. Маклафлин Б. Изучаем Ajax. СПб.: Питер; 2007. 443 с.
23. McLaughlin B. Head Rush Ajax. St. Petersburg, Piter; 2007. 443 p.
24. Макфарланд Д. JavaScript и jQuery: исчерпывающее руководство. М.: Диалектика; 2015. 880 с.
25. Mcfarland D. JavaScript & jQuery: The Missing Manual. Moscow, Dialektika; 2015. 880 p.
26. DBLogic Assistant. Available at: <http://msuniversity.ru/dba/>
27. MSUniversity. Available at: <http://msuniversity.ru>

Информация об авторах

Смирнов Михаил Вячеславович, канд. экон. наук, доцент кафедры «Предметно-ориентированные информационные системы», Институт кибербезопасности и цифровых технологий, МИРЭА — Российский технологический университет, г. Москва, Россия; *ORCID*: <http://orcid.org/0000-0001-5018-2963>; *e-mail*: smirnov_m@mirea.ru

Поленок Вячеслав Михайлович, магистрант кафедры практической и прикладной информатики, Институт информационных технологий, МИРЭА — Российский технологический университет, г. Москва, Россия; *ORCID*: <http://orcid.org/0000-0003-2494-9803>; *e-mail*: ya.slavar@yandex.ru

Information about the authors

Mikhail V. Smirnov, Candidate of Sciences (Economics), Associate Professor at the Department “Subject-oriented information systems”, Institute for Cybersecurity and Digital Technologies, MIREA — Russian Technological University, Moscow, Russia; *ORCID*: <http://orcid.org/0000-0001-5018-2963>; *e-mail*: smirnov_m@mirea.ru

Vyacheslav M. Polenok, a master’s student at the Department of Practical and Applied Informatics, Institute of Information Technologies, MIREA — Russian Technological University, Moscow, Russia; *ORCID*: <http://orcid.org/0000-0003-2494-9803>; *e-mail*: ya.slavar@yandex.ru

Поступила в редакцию / Received: 20.08.2021.

Поступила после рецензирования / Revised: 29.10.2021.

Принята к печати / Accepted: 09.11.2021.