

Разработка компьютерной игры для обучения языку SQL

А. Э. Потемкина, email: potemkina.ae@gmail.ru

М. В. Матвеева, email: marie.matveeva@gmail.ru

«Воронежский государственный университет»

***Аннотация.** В статье рассмотрен процесс разработки компьютерной игры для обучения языку SQL. Произведен анализ функциональности приложения. В ходе работы спроектированы модель данных и структура приложения. Выполнена практическая реализация компьютерной игры.*

***Ключевые слова:** Компьютерная игра, обучающая игра, модель данных, SQL.*

Введение

На рынке современных компьютерных и мобильных игр существует небольшое количество приложений, посвященных обучению языку SQL. Такие приложения предлагают пользователю возможность самостоятельно изучить язык SQL в игровом формате. Подобный подход является привлекательным для желающих приобрести или повысить навыки в сфере работы с базами данных.

Многие из существующих приложений реализуют фиксированные короткие образовательные программы, в которых отсутствует теоретический материал, позволяющий изучить язык SQL. Другие не отвечают современным требованиям компьютерной игры в сфере оформления визуальных интерфейсов и удобства в эксплуатации. Нигде не реализуется рейтинговая система, стимулирующая пользователей к решению задач. Так же большинство приложений не акцентируют внимание пользователя на грамотном написании SQL команд и запросов.

Из этого следует необходимость создания приложения, учитывающего недостатки существующих и поддерживающие их преимущества. Компьютерная игра должна позволять реализовывать программы для обучения студентов и начинающих изучать язык SQL, с которым было бы удобно работать как студенту, так и преподавателю.

1. Анализ существующих приложений

В настоящее время существует небольшое количество компьютерных и мобильных игр, обучающих языку SQL. Среди них наиболее известные:

- SoloLearn;
- SQL рецепты;
- DataCamp;

Сравнительный анализ их функциональности приведен в таблице 1:

Таблица 1

Сравнительный анализ существующих приложений

Критерий сравнения	SoloLearn	SQL рецепты	DataCamp
Личный кабинет	+	-	+
Рейтинг пользователей	-	-	-
Современный дизайн	+	-	+
Поддержка русского языка	+	+	-
Платный контент	+	-	+
Предоставление теоретического материала	+	+	-
Автоматическое исправление SQL команд	-	-	-
Поддержка оператора SELECT	+	+	+

Из представленных данных можно сделать вывод о необходимости создания игры, учитывающей недостатки существующих приложений и поддерживающее их преимущества.

2. Функциональность приложения

Требовалось спроектировать и реализовать приложение, обладающее следующей функциональностью:

- регистрация и авторизация пользователей;
- личный кабинет пользователя;
- предоставление пользователю возможности выбрать из списка доступных задач;
- решение пользователем представленных задач;
- вывод информации о правильности решенной задачи;
- рейтинговая система;

- предоставление пользователю возможности просматривать рейтинг;
- автоматическое исправление SQL команд на правильное написание;
- удаление и добавления пользователя администратором.

Автоматическое исправление SQL команд помогает пользователю акцентировать внимание на грамотности запросов с точки зрения написания. То есть команды языка SQL должны быть написаны прописными буквами, а названия столбцов и таблиц так, как указано в базе данных.

Приведенная выше функциональность реализована в приложении. На рис. 1 представлена функциональность приложения изображенная на диаграмме вариантов использования.

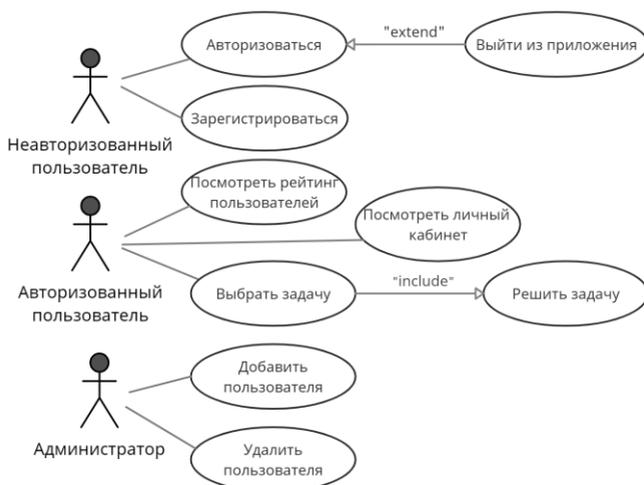


Рис. 1. Диаграмма вариантов использования

3. Анализ структуры приложения.

При проведении анализа структуры приложения, была составлена примерная схема компонентов приложения (рис. 2):

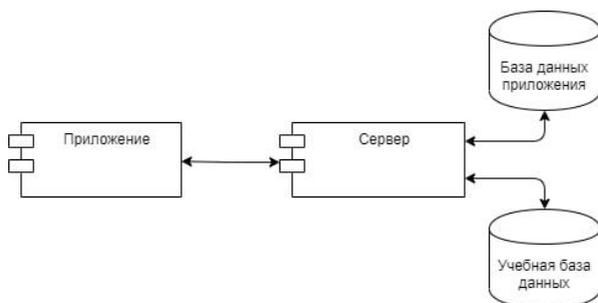


Рис. 2. Схема компонентов приложения

База данных приложения хранит информацию о пользователях и другие системные таблицы. Учебная база данных используется пользователем при решении задач.

Сервер осуществляет связь между базами данных и приложением.

В ходе работы была спроектирована модель данных приложения, которая включает следующие компоненты:

1. «Пользователи». Хранит: индивидуальный номер пользователя; логин; пароль; имя; фамилию; роль.
2. «Задания». Включает в себя: индивидуальный номер задания; текст задания; ответ к заданию.
3. «Пользователь-задача». Связующий элемент между задачей и пользователем. Включает в себя: индивидуальные номера пользователя и задачи, а так же баллы, которые пользователь получил за решение определенного задания. На основе этого будет формироваться рейтинг.
4. «Роль пользователя». Хранит данные о ролях пользователя в системе. Будет содержать в себе всего две записи, соответствующие роли пользователя и роли администратора.

4. Средства реализации

При реализации проекта использовались следующие программные средства:

1. Интерактивная среда разработки Microsoft Visual Studio 2019.
2. Язык разработки C#.
3. Реляционная система управления базами данных MySQL.
4. Программный инструмент моделирования StarUML.
5. Программное обеспечение для сборки веб-сервера XAMPP.

5. Структура приложения

Физические взаимосвязи между аппаратными компонентами приложения отображены на диаграмме развертывания на рис. 3.

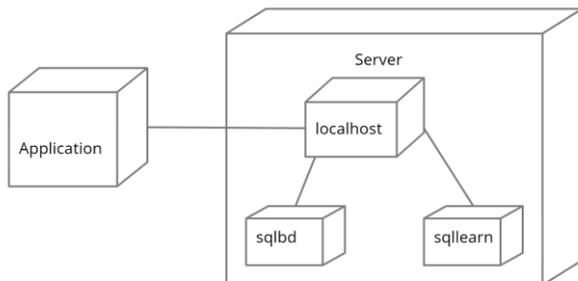


Рис. 3. Диаграмма развертывания

Детально узлы схемы описаны в таблице:

Таблица 2

Структура приложения

Узел	Описание
Application	Приложение.
Localhost	Сервер приложений. Осуществляет связь с базами данных и самим приложением.
Sqlbd	Системная база данных.
Sqllearn	Учебная база данных.

На рис. 4 представлена структура программных средств изображенных на диаграмме компонентов:

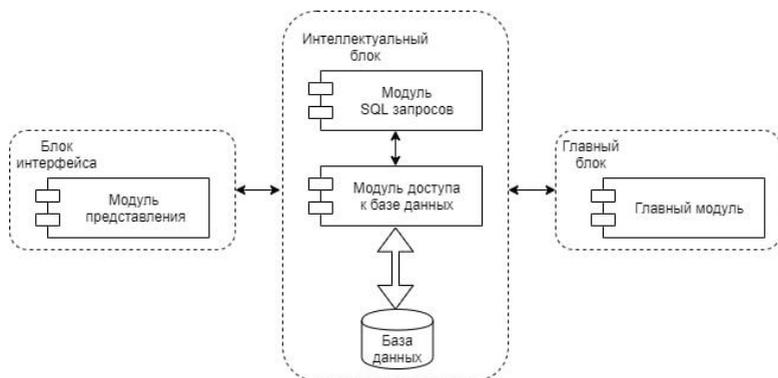


Рис. 4. Диаграмма компонентов архитектуры программных средств

Задачи и функции каждого модуля:

Главный модуль представляет собой класс Main, который содержит в себе метод, отвечающий за запуск приложения:

Модуль представления отвечает за создание и размещение элементов управления на форме.

Модуль доступа к базе данных – модуль отвечающий за передачу запросов в базу данных и обработку полученных от нее ответов.

Модуль SQL запросов – модуль отвечающий за грамотное написание команд и запросов. Содержит в себе метод CorrectionSqlQuery(). Этот метод реализует алгоритм исправления запросов, которые являются неправильными с точки зрения написания. Все команды языка SQL пишутся прописными буквами, а название столбцов и таблиц так, как указано в базе данных

Ниже представлены основные классы приложения и их главные задачи:

1. Главный класс Program, который отвечает за запуск приложения и загружает форму авторизации.
2. Вспомогательный класс DB, отвечающий за подключение приложения к базе данных. Содержит в себе методы, отвечающие за открытие соединения с базой данных и закрытие соединения, а так же метод, который устанавливает соединение с базой.
3. Вспомогательный класс DataBank. Данный класс содержит список переменных, которые необходимы для передачи информации из одного класса в другой.

4. Класс SQLquery. Содержит в себе метод, реализующий алгоритм исправления запросов, которые являются неправильными с точки зрения написания.

6. Модель данных

Описание таблиц базы данных приведено в таблице 3.

Таблица 3

Описание таблиц базы данных приложения

Таблица	Столбец	Описание
Users	id_user	Идентификационный номер.
	Login	Логин пользователя.
	password	Пароль пользователя.
	Name	Имя пользователя.
	lastName	Фамилия пользователя.
	id_role	Роль пользователя.
Roles	id_role	Идентификационный номер.
	Role	Имя роли.
Tasks	id_task	Идентификационный номер.
	Task	Условие задачи.
	Answer	Ответ к задаче.
users_tasks	id_ub	Идентификационный номер.
	id_user	Идентификационный номер пользователя, решившего задание.
	id_task	Идентификационный номер задания.
	Ball	Баллы, полученные пользователем.

7. Алгоритм исправления SQL команд

Для исправления SQL команд и вывода правильно отформатированного запроса, необходимо выполнить алгоритм, представленный ниже:

1. ввести команду или запрос;
2. нажать на кнопку проверки;
3. найти SQL команду;
4. если команда не найдена, то перейти к п. 7, иначе перейти к п. 5;
5. проверить на правильность написания, если команда корректна, то есть все названия таблиц и столбцов указаны так же, как и в базе данных, а команды языка SQL написаны прописными буквами, в этом случае перейти к п. 7, иначе к п. 6;

6. если команда отличается от заданного верного написания, удалить эту команду, на место удаленной вставить команду верного написания;
7. конец алгоритма.

Данный алгоритм, реализованный в программе позволит пользователю акцентировать внимание не только на составлении запроса, но и на правильности написания команд языка манипулирования данными и названий таблиц или столбцов.

Заключение

В результате работы была спроектирована и реализована компьютерная игра, позволяющая:

- зарегистрироваться и авторизоваться новым пользователям;
- решать пользователю задачи по обучению языка SQL;
- автоматически проверять задачи и выводить результаты их проверки;
- вести рейтинг пользователей;
- просмотреть пользователю свой личный кабинет и рейтинг;
- автоматически исправлять запросы и команды SQL языка, которые являются неправильными с точки зрения грамотного написания;
- возможность пользователя с правами администратора добавлять и удалять других пользователей.

Литература

1. Программирование на Microsoft ADO.NET 2.0 / Д. Сеппа; пер. с англ. – М.: Издательство «Русская редакция»; СПб: Питер, 2007. – 784 с.
2. MySql 5.0/ В. Гольцман; Текст предоставлен издательством [http://litres.ru/MySQL 5.0](http://litres.ru/MySQL_5.0). Библиотека программиста: Питер; Санкт-Петербург; 2010.
3. Куликов, С. С. Работа с MySQL, MS SQL Server и Oracle в примерах : практ. пособие. / С. С. Куликов. — Минск: БОФФ, 2016. — 556 с.