

АВТОМАТИЗИРОВАННАЯ ВЕБ-ОРИЕНТИРОВАННАЯ СИСТЕМА КОНТРОЛЯ ЗНАНИЙ УЧАЩИХСЯ

Дранников А.В.¹, Похил И.А.², Колесников А.В.², Воронов А.Э.²

*¹Воронежский государственный университет инженерных технологий,
Воронеж;*

²Луганский государственный университет имени Владимира Даля, Луганск

Ключевые слова: автоматизация тестирования, веб-ориентированная система, контроль знаний, Vue.js, TypeScript, типы тестовых заданий, парсинг данных, импорт тестов, административный интерфейс.

Аннотация. В статье рассматривается процесс разработки и реализации веб-ориентированного программного комплекса, предназначенного для автоматизации контроля знаний учащихся. Обосновывается выбор технологического стека (Vue.js, TypeScript) для создания масштабируемого и надежного клиентского приложения. Подробно описывается ключевая функциональность системы: поддержка пяти различных типов вопросов, что позволяет проводить многоаспектную оценку компетенций. Особое внимание уделяется архитектуре системы, механизмам разграничения доступа по уровням образования и защите контента от копирования. Представлен инновационный модуль административного интерфейса – алгоритм-парсер, который автоматизирует импорт тестовых заданий из текстовых документов (например, Microsoft Word), значительно ускоряя наполнение базы данных и минимизируя ошибки. Делается вывод о готовности комплекса к внедрению в учебный процесс вузов и обозначаются перспективы его дальнейшего развития.

AUTOMATED WEB-ORIENTED STUDENT KNOWLEDGE MONITORING SYSTEM

Drannikov A.V.¹, Pokhil I.A.², Kolesnikov A.V.², Voronov A.E.²

¹Voronezh State University of Engineering Technologies, Voronezh;

²Lugansk State University named after Vladimir Dahl, Lugansk

Keywords: test automation, web-based system, knowledge assessment, Vue.js, TypeScript, test item types, data parsing, test import, administrative interface.

Abstract. This article examines the development and implementation of a web-based software suite designed to automate student knowledge assessment. The choice of a technology stack (Vue.js, TypeScript) for creating a scalable and reliable client application is justified. The key functionality of the system is described in detail: support for five different question types, enabling a multifaceted competency assessment. Particular attention is paid to the system architecture, mechanisms for access control by educational level, and content protection from copying. An innovative administrative interface module is presented a parser algorithm that automates the import of test items from text documents (e.g., Microsoft Word), significantly accelerating database population and minimizing errors. A conclusion is made about the readiness of the complex for implementation in the educational process of universities and prospects for its further development are outlined.

Современный этап развития образовательной системы характеризуется интенсивной интеграцией цифровых технологий во все этапы педагогического процесса. Одной из наиболее значимых задач в этом контексте является автоматизация процедуры контроля знаний, обеспечивающая как оперативность обратной связи, так и снижение административной нагрузки на преподавательский

состав. В ответ на этот вызов был разработан и внедрен специализированный программный комплекс, представляющий собой не просто инструмент для проведения тестирования, а целостную среду для оценки учебных достижений. Фундаментальным принципом, заложенным в основу разработки, являлось создание гибкой, масштабируемой и удобной в эксплуатации системы, способной адаптироваться к различным дисциплинам и уровням подготовки. «...К таким разработкам относится создание адаптивных интеллектуальных обучающих и тестирующих систем, важным компонентом которых является модель знаний студентов» [1].

Выбор технологического стека для клиентской части пал на Vue.js в связке с TypeScript [2]. Это решение продиктовано требованиями к поддержке и развитию сложной кодовой базы. Реактивная модель данных Vue.js обеспечивает мгновенное и синхронизированное обновление интерфейса, что критически важно для динамического взаимодействия пользователя с тестом. Компонентный подход позволил разделить интерфейс на переиспользуемые и логически завершённые блоки (например, отображение вопроса или модуль результатов), значительно упрощая разработку и сопровождение. Использование TypeScript добавляет в проект статическую типизацию, которая повышает надёжность и предсказуемость кода. Этот подход позволяет выявлять ошибки, связанные с несоответствием типов, на этапе компиляции, что особенно ценно при работе с большим количеством сущностей и в условиях коллективной разработки.

Ядром функциональности комплекса является подсистема тестирования, поддерживающая пять различных типов вопросов. Такое многообразие не является произвольным, а подчинено цели многоаспектной оценки компетенций учащегося. Классический вопрос с выбором единственного правильного ответа эффективен для проверки репродуктивных знаний и узнавания фактов. Однако его познавательная ограниченность компенсируется вторым типом вопроса – вопросом на сопоставление, который требует от тестируемого установления структурных и причинно-следственных связей между понятиями, тем самым проверяя ассоциативное мышление и системность знаний. Третий тип вопроса требует от тестируемого выставить ответы в правильной последовательности. Также были введены два типа заданий со свободной формой ответа (четвертый и пятый типы вопросов). Первый требует ввода единственной верной формулировки, что направлено на оценку точности и однозначности знаний. Второй, более гибкий, допускает несколько синонимичных или вариативных правильных ответов, что позволяет оценить понимание сути явления, а не его заученное определение. Подобная комплексная типология позволяет преподавателю конструировать сбалансированные тесты, направленные на проверку разных уровней познавательной деятельности – от узнавания до продуктивного воспроизведения.

Система визуализации результатов была разработана с ориентацией на принципы наглядности и удобства. Формируемый по завершении теста отчет включает не только количественные данные (общее количество вопросов, число верных и ошибочных ответов), но и детальную качественную разбивку. Цветовое кодирование (зеленый для верных ответов, красный – для неверных)

предоставляет учащемуся мгновенную и интуитивно понятную карту его успехов и затруднений (рис. 1).

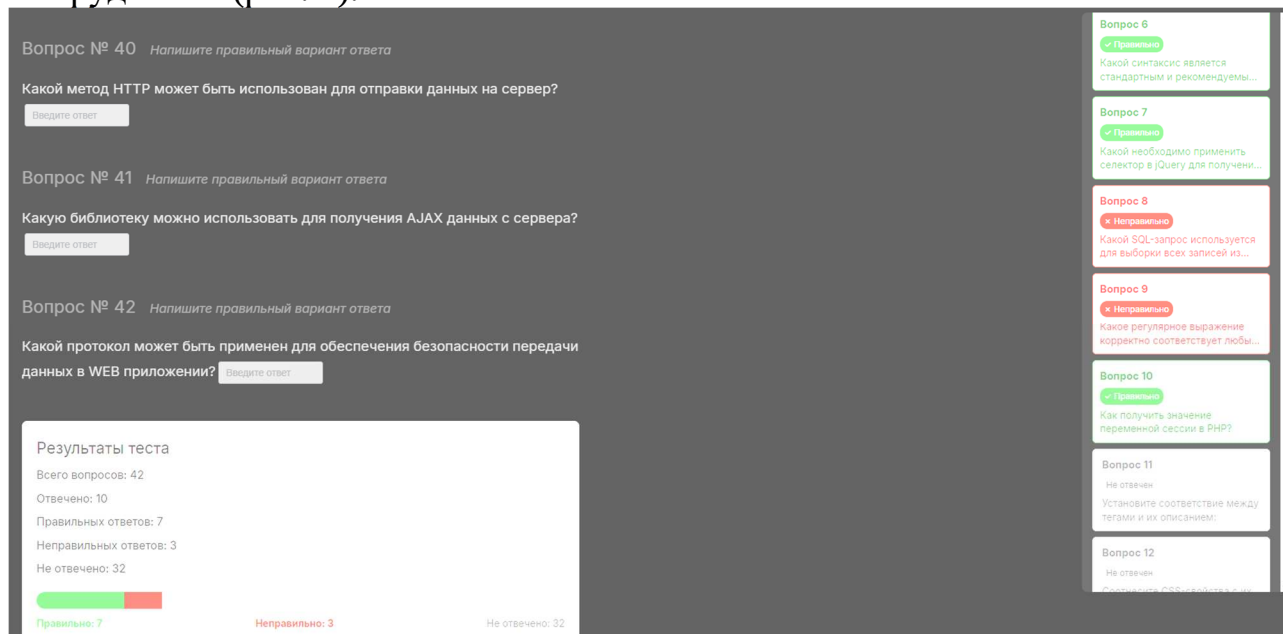


Рис. 1. Вывод результатов тестирования

«С помощью программного обеспечения проблема обработки и вывода результатов может быть решена путём улучшения обработки данных результатов, а также упрощения вывода конечного результата с помощью метода представления данных» [3].

Архитектура системы изначально проектировалась с учетом необходимости разграничения контента по уровням образования. Начальный этап взаимодействия пользователя с комплексом предполагает выбор между бакалавриатом и магистратурой (рис. 2).

Это организационное решение позволяет программно разделить бланки тестовых заданий по степени сложности и глубине охвата материала, обеспечивая актуальность предъявляемых заданий уровню подготовки обучающегося. Важным аспектом, направленным на сохранение академической честности и достоверности оценки, является реализация механизмов защиты тестового контента от несанкционированного копирования. Это предотвращает практику простого механического запоминания правильных ответов без осмысления учебного материала, тем самым мотивируя студентов на его глубокое усвоение.



Рис. 2. Выбор уровня тестирования

Серверная часть системы базируется на реляционной базе данных, которая выступает в роли централизованного и структурированного хранилища всей информации. В ней поддерживаются связи между сущностями: учебные курсы, тесты, вопросы различных типов, варианты ответов, а также результаты прохождений. Для управления системой был реализован специализированный административный интерфейс – защищенная веб-страница, предоставляющая авторизованным пользователям (преподавателям, администраторам) полный спектр операций CRUD (Create, Read, Update, Delete) для работы с контентом.

Наиболее инновационным элементом административного модуля является разработанный механизм, отвечающий за импорт тестовых заданий из внешних источников. Предложенное решение позволяет администратору копировать форматированный текст из документа Microsoft Word и вставлять его в специальное поле интерфейса. Запускаемый затем алгоритм-парсер проводит автоматический анализ текстовой структуры, идентифицирует паттерны, соответствующие формулировкам вопросов и ответов, производит их сегментацию на логические блоки и на основе полученных данных автоматически генерирует новые записи в базе данных (рис. 3).

Данный функционал не только на порядок снижает временные затраты на наполнение системы, но и минимизирует количество ошибок, неизбежных при ручном вводе большого массива информации, обеспечивая тем самым высокое качество контента.

Рис. 3. Быстрый импорт вопросов с файлов Word

Таким образом, разработанный веб-ориентированный программный комплекс представляет собой готовое к внедрению решение для автоматизации контроля знаний в вузе. В основе системы лежит современный технологический

стек (Vue.js, TypeScript), обеспечивающий надежность, масштабируемость и удобство сопровождения. Ключевым достижением является реализация пяти типов вопросов, что позволяет проводить многоаспектную оценку компетенций – от простого узнавания до проверки системного мышления и понимания предмета. Особую практическую ценность представляет разработанный модуль автоматического импорта тестовых заданий из текстовых документов, который кратно сокращает время наполнения базы данных и исключает ошибки ручного ввода. Архитектура системы учитывает требования безопасности и разграничения доступа по уровням подготовки, а также оставляет возможности для дальнейшего развития, включая внедрение адаптивного тестирования и интеграцию с образовательными платформами.

Список литературы

1. Сологуб Г.Б. Построение и использование байесовской сети для моделирования знаний студента в интеллектуальной системе тестирования // Компьютерные инструменты в образовании. – 2012. – №2. – С. 40-48.
2. The Progressive JavaScript Framework [Электронный ресурс]. – Режим доступа: <https://vuejs.org>.
3. Гавзова Е.В., Щенев Е.С., Щенева Ю.Б. Разработка программного и методического обеспечения электронного дистанционного учебного курса с применением элементов искусственного интеллекта контроля знаний // Информатика и прикладная математика. – 2022. – №28. – С. 27-32.

Сведения об авторах:

Дранников Алексей Викторович – д.т.н., профессор, профессор кафедры машин и аппаратов пищевых производств;

Похил Илья Алексеевич – студент;

Колесников Андрей Валерьевич – к.т.н., доцент, заведующий кафедрой автоматизации и компьютерно-интегрированных технологий;

Воронов Артур Эдуардович – к.т.н., доцент, доцент кафедры автоматизации и компьютерно-интегрированных технологий.