

7. Software and systems engineering — Software testing. Part 1: General concepts, ISO/IEC/IEEE 29119-2:2021 Software and systems engineering — Software testing. Part 2 : Test processes : ISO/IEC/IEEE 29119-1:2022. — 47 p. // International Organization for Standardization. — URL: <https://www.iso.org/standard/81291.html> (date of access: 05.05.2025).

8. Менеджмент проектов, программ и портфелей. Руководство по менеджменту проектов : СТБ ISO 21502-2024 ; введ. 01.08.2025. — Минск : Госстандарт : Белорус. гос. ин-т стандартизации и сертификации, 2024. — 52 с.

9. Система стандартов по информации, библиотечному и издательскому делу. Информация и документация. Управление документами. Логическая структура, состав метаданных документов и требования к их содержанию : ГОСТ Р 7.0.109-2024 : утв. и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 29 февраля 2024 г. № 265-ст. — М. : Российский институт стандартизации, 2024. — 114 с. //Электронный фонд правовых и нормативно-технических документов. — URL: <https://docs.cntd.ru/document/1305118388> (дата обращения: 03.05.2025).

10. Качество программных средств. Термины и определения : ГОСТ 28806-90 ; введ. 01.01.1992. — Минск : Госстандарт : Белорус. гос. ин-т стандартизации и сертификации, 2011. — 12 с.

11. Оценка качества программных средств. Общие положения : ГОСТ 28195-99 : введ. 01.03.2000. — Минск : Госстандарт : Белорус. гос. ин-т стандартизации и сертификации, 2024. — 52 с.

12. Информационные технологии. Оценка программной продукции. Характеристики качества и руководства по их применению: СТБ ИСО/МЭК 9126-2003 ; введ. 01.11.2003. — Минск : Госстандарт : Белорус. гос. ин-т стандартизации и сертификации, 2024. — 16 с.

13. Разработка программного обеспечения. Требования к качеству и оценка программного продукта (SQuaRE). Руководство по SQuaRE: СТБ ISO/IEC 25000-2009 ; введ. 01.01.2010. — Минск : Госстандарт : Белорус. гос. ин-т стандартизации и сертификации, 2009. — 44 с.

14. Разработка программного обеспечения. Требования к качеству и оценка программного продукта (SQuaRE). Планирование и управление : СТБ ISO/IEC 25001-2009 ; введ. 01.01.2010. — Минск : Госстандарт : Белорус. гос. ин-т стандартизации и сертификации, 2009. — 20 с.

УДК 378.14

Э. Н. Прошунин

*Учреждение образования «Барановичский государственный университет»,
Барановичи, Республика Беларусь*

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УЧЕТА ОБУЧЕНИЯ И АТТЕСТАЦИИ СОТРУДНИКОВ «558 АВИАЦИОННЫЙ РЕМОНТНЫЙ ЗАВОД»

Введение. Автоматизация процессов учёта обучения и тестирования сотрудников является важной задачей для предприятий, стремящихся обеспечить высокий уровень профессиональной подготовки персонала и соответствие требованиям безопасности. Особенно это актуально на предприятиях авиационной отрасли, таких как ОАО «558 Авиационный ремонтный завод», где квалификация сотрудников напрямую влияет на безопасность и надёжность выполняемых работ по ремонту и обслуживанию авиационной техники.

В условиях постоянно меняющихся стандартов и нормативных требований необходимо иметь систему, позволяющую централизованно управлять процессами обучения и контроля знаний. Внедрение автоматизированной информационной системы обеспечит удобное назначение тестов и курсов, ведение базы вопросов, фиксацию результатов и формирование отчётности. Также важно реализовать разграничение прав доступа в зависимости от роли пользователя (администратор, сотрудник), что обеспечит безопасность и чёткость управления.

Для реализации системы используются современные технологии: серверная часть разработана на платформе ASP.NET Core, клиентская — с применением Windows Forms на языке C#, а хранение данных осуществляется в СУБД SQLite. Обмен данными между компонентами происходит через HTTP с использованием формата JSON. Внедрение данной системы позволит повысить эффективность работы кадровых и административных подразделений, сократить количество ошибок и обеспечить прозрачный контроль за обучением сотрудников.

Основная часть. C# — современный язык программирования, активно используемый для разработки десктопных и серверных приложений. Он позволяет создавать надёжные и производительные приложения для операционной системы Windows, что особенно актуально для корпоративных решений, используемых внутри предприятия. В рамках разработки информационной системы учёта обучения и тестирования сотрудников C# был выбран как основной язык для клиентской и серверной частей системы [1].

Windows Forms — библиотека для создания графических интерфейсов в приложениях на платформе .NET. Она используется для реализации клиентской части приложения, обеспечивая удобное взаимодействие пользователей с системой. Интерфейс позволяет сотрудникам проходить тесты, просматривать свои результаты, а администраторам — назначать курсы, управлять вопросами и отслеживать успеваемость персонала [2].

SQLite — встраиваемая реляционная СУБД, выбранная для хранения информации о сотрудниках, тестах, результатах, ролях и назначениях. Она проста в развертывании, не требует отдельного сервера и отлично подходит для приложений, работающих в локальной сети предприятия [3].

Среди основных функциональных требований можно выделить следующие:

- проведение тестирования сотрудников (отображение списка доступных тестов, прохождение теста, отправка результатов);
- управление тестами (создание, редактирование и удаление тестов, добавление и редактирование вопросов и вариантов ответов);
- мониторинг уровня продаж (отображение статистики продаж).
- назначение тестов (администратор может назначить тест определённому сотруднику с указанием срока прохождения);

- формирование отчётности (отображение результатов по каждому сотруднику и тесту, экспорт отчётов при необходимости);
- хранение результатов (сохранение данных о прохождении тестов, количестве попыток, времени выполнения и баллах);
- авторизация и аутентификация пользователей (вход по логину и паролю, проверка уровня доступа).

Если рассматривать структуру приложения, то можно выделить следующие компоненты:

1. Серверная часть — разработана на ASP.NET Core и отвечает за взаимодействие с базой данных, обработку запросов от клиента, реализацию бизнес-логики, включая назначение тестов, проверку прав доступа и ведение учёта пользователей и результатов.

2. Клиентская часть — реализована на WinForms и обеспечивает взаимодействие пользователей с системой: прохождение тестов, управление тестами, создание отчётов и отображение назначений. Интерфейс адаптирован под обе роли: администратора и сотрудника.

3. Система управления базой данных — используется SQLite. В базе хранятся сведения о тестах, вопросах, вариантах ответов, сотрудниках, ролях, назначениях и результатах. SQLite выбрана за простоту, надёжность и достаточную функциональность для работы в локальной среде.

В ходе работы было разработано настольное приложение для администратора, позволяющее управлять сотрудниками, их ролями, отделами и учётными данными, как показано на рисунке 1

При открытии выбранного теста отображается окно редактирования вопросов (рисунок 2).

На рисунке 3 показано окно назначения теста сотруднику.

На рисунке 4 показана панель сотрудника.

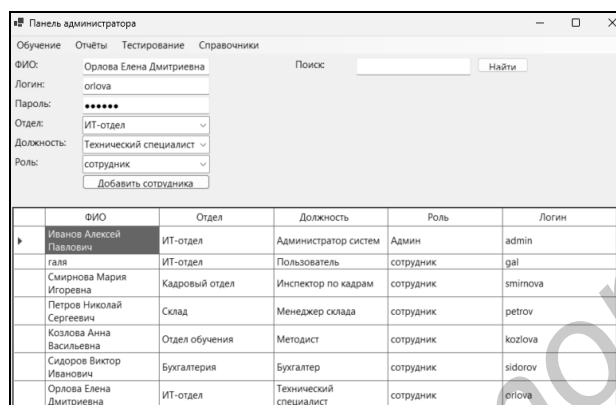


Рисунок 1 — Панель администратора

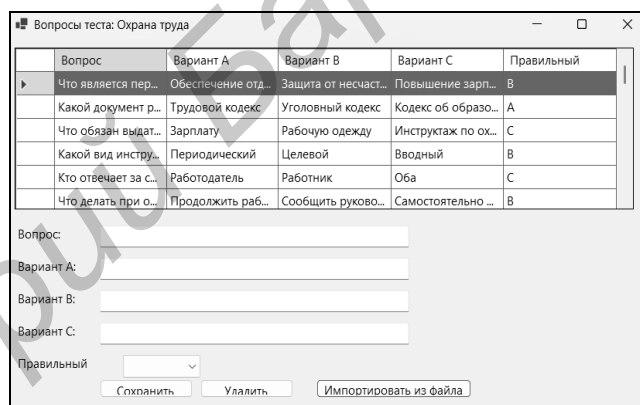


Рисунок 2 — Добавление вопросов

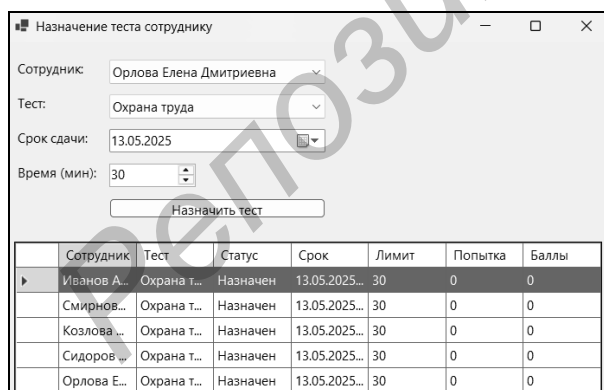


Рисунок 3 — Назначение тестов

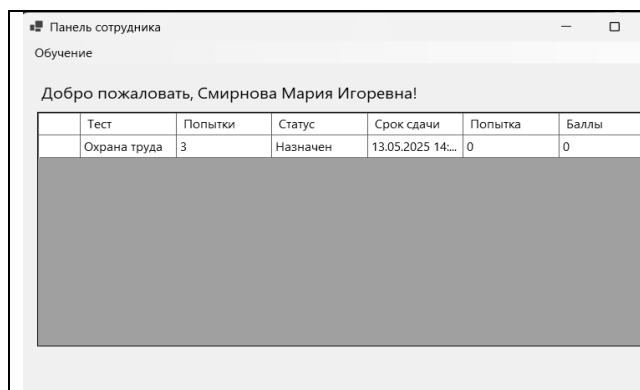


Рисунок 4 — Панель сотрудника

Заключение. В рамках выполнения дипломного проекта была разработана информационная система для автоматизации процесса тестирования сотрудников на предприятии. Основной целью проекта стало повышение эффективности оценки знаний персонала, минимизация влияния человеческого фактора и создание удобного инструмента для контроля обучения и сертификации работников.

Реализованное программное обеспечение включает в себя модули для создания и редактирования тестов, назначения тестов сотрудникам, прохождения тестирования, а также сбора и анализа результатов. Система предоставляет гибкую ролевую модель доступа, где сотрудники могут только проходить тесты, а администраторы — управлять всеми сущностями системы. Архитектура приложения разделена на

клиентскую и серверную части, что повышает его масштабируемость и сопровождаемость. В качестве СУБД была выбрана SQLite, обеспечивающая надёжность хранения и быструю работу в локальной среде.

Разработка системы позволила обеспечить централизованный подход к обучению и оценке знаний персонала, повысить прозрачность процессов и сократить время на проведение проверок. Внедрение такого ПО является актуальным шагом на пути цифровизации внутренних процессов организации

Список цитируемых источников

1. Microsoft Docs. Что такое ASP.NET Core?. — URL: <https://learn.microsoft.com/ru-ru/aspnet/core/> (дата обращения: 06.05.2025).
2. SQLite Documentation. About SQLite. — URL: <https://sqlite.org/about.html> (дата обращения: 06.05.2025).
3. Microsoft Docs. WinForms overview. — URL: <https://learn.microsoft.com/en-us/dotnet/desktop/winforms/> (дата обращения: 06.05.2025).

УДК 378.14

В. Д. Сакович

*Учреждение образования «Барановичский государственный университет»,
Барановичи, Республика Беларусь*

РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ УЧЕТА И МОНИТОРИНГА ПРОДАЖ ЛЕСОПРОДУКЦИИ

Введение. Автоматизация процессов продаж представляет собой внедрение программных решений, направленных на снижение доли ручного труда, сокращение временных затрат, повышение точности и прозрачности операций. Основной целью автоматизации в данном контексте является повышение эффективности управления продажами, ускорение обработки заказов, повышение удовлетворенности клиентов, а также оперативный контроль ключевых показателей реализации.

Современные предприятия стремятся внедрять такие решения, которые обеспечивают не только прирост производительности, но и улучшение качества управленческих решений. Автоматизированные системы сбора, хранения и анализа информации о продажах позволяют минимизировать влияние человеческого фактора, повысить оперативность реагирования на изменения спроса и своевременно выявлять отклонения от плановых показателей.

В рамках реализации проекта было решено разработать веб-приложение на основе современных веб-технологий, включая JavaScript, с использованием Node.js для серверной части, React и библиотеки компонентов Bootstrap для клиентской части, а также PostgreSQL в качестве системы управления базами данных. Такой технологический стек обеспечивает надёжность, масштабируемость, кросс-платформенность и удобство в сопровождении разрабатываемого программного продукта.

Основная часть. JavaScript — один из наиболее популярных языков программирования для разработки веб-приложений. Он активно используется для создания интерактивных интерфейсов и функционала на стороне клиента. JavaScript позволяет реализовать динамическое взаимодействие пользователя с веб-приложением, обеспечивая плавную и быструю работу интерфейса [1].

React — популярная JavaScript-библиотека для создания пользовательских интерфейсов. Она была выбрана для разработки фронтенда веб-приложения, так как обеспечивает высокую производительность, повторное использование компонентов и простоту в интеграции с другими инструментами. React позволяет эффективно управлять состоянием приложения и обновлять только те части интерфейса, которые изменяются, что существенно ускоряет работу приложения [2].

Node.js — серверная платформа, основанная на JavaScript, используется для создания серверной части приложения. Node.js обеспечивает асинхронную обработку запросов, что позволяет серверу эффективно обрабатывать большое количество подключений одновременно. Это решение особенно важно для работы с большими объёмами данных, таких как информация о продажах, остатках продукции на складе и заказах от клиентов [3].

Среди основных функциональных требований можно выделить следующие:

1. Формирование и контроль состава продукции (ввод и хранение данных о видах лесопроductии, возможность редактирования состава товара).
2. Управление продажами (добавление, редактирование и хранение данных о продаже продукции, учёт информации о покупателях и деталях сделок).
3. Мониторинг уровня продаж (отображение статистики продаж).
4. Управление складскими запасами (отображение актуального количества товара на складе).
5. Генерация отчетов (формирование отчетов по продажам и остаткам на складе).
6. Разграничение доступа и авторизация (реализация ролевой модели с различными правами доступа, система аутентификации пользователей).
7. Интуитивно понятный графический интерфейс (создание удобного и понятного интерфейса для пользователей, с возможностью простого взаимодействия с приложением).