

РАЗРАБОТКА И РЕАЛИЗАЦИЯ ПРИЛОЖЕНИЯ ДЛЯ ТЕСТИРОВАНИЯ СТУДЕНТОВ

Хлебников В.В., Старцев М.В.

Россия, Тамбовский государственный университет имени Г. Р. Державина
khlebnikovvv@mail.ru, starcevmv@yandex.ru

Аннотация. В статье описывается разработанная автономная тестовая система, позволяющая проводить автоматизированный контроль полученных знаний в ходе обучения. Также реализованное приложение позволяет осуществлять сбор информации, полученной во время тестирования, для ее дальнейшего анализа.

Ключевые слова: тестовая система, конструктор тестов, язык программирования.

На сегодняшний день использование новейших информационных технологий является неотъемлемой частью процесса обучения как в средней, так и в высшей школе. Широкое использование тестового метода стало возможно благодаря информационным технологиям, позволившим автоматизировать процесс тестирования обучающихся. Тестовые системы в настоящее время существуют как в автономном варианте, так и бывают встроены в автоматизированный учебный комплекс. Автономные тестовые системы, как правило, имеют по сравнению с интегрированными более расширенный функционал. Они позволяют не только учитывать количество правильных ответов, но и проводить более детальный анализ динамической информации о степени усвоения материала тем или иным обучающимся, отслеживать психофизические параметры прохождения теста, предоставлять агрегированную информацию по классу (группе, потоку) [1,2].

В рамках существующих на данный момент времени, автоматизированных тестирующих систем решается ряд важных задач обучения, подробно описанных в учебной литературе [3]. Эти задачи можно разделить на несколько групп. К первой группе можно отнести задачи проверки уровня знаний, умений и навыков обучающихся до и после освоения материала. В результате такого тестирования легко можно проследить динамику по приращению так называемых ЗУН. К этой же группе относятся задачи проверки показателей работоспособности обучающихся, которая осуществляется путем регистрации таких психофизиологических показателей, как скорость реакции, уровень внимания и т.д. В частности, к этой группе можно отнести всевозможные IQ-тесты, которые позволяют выявить существующий уровень интеллекта обучаемого.

Вторая группа задач связана с регистрацией и статистическим анализом показателей усвоения учебного материала: ведение индивидуальных разделов для каждого обучающегося, определение времени решения задач, определение общего числа ошибок и т.д.

Таким образом, цель разработки описываемой в данной статье системы заключалась в программной реализации такой автоматизированной тестирующей системы, которая была бы способна осуществлять автоматизированный сбор информации об уровне усвоения изученного материала.

Для разработки тестирующей системы нами была использована доступная нам программная среда Delphi XE [4]. Для облегчения разработки индивидуального задания, был создан «Конструктор тестов», внешний вид которого изображен на рисунке 1.

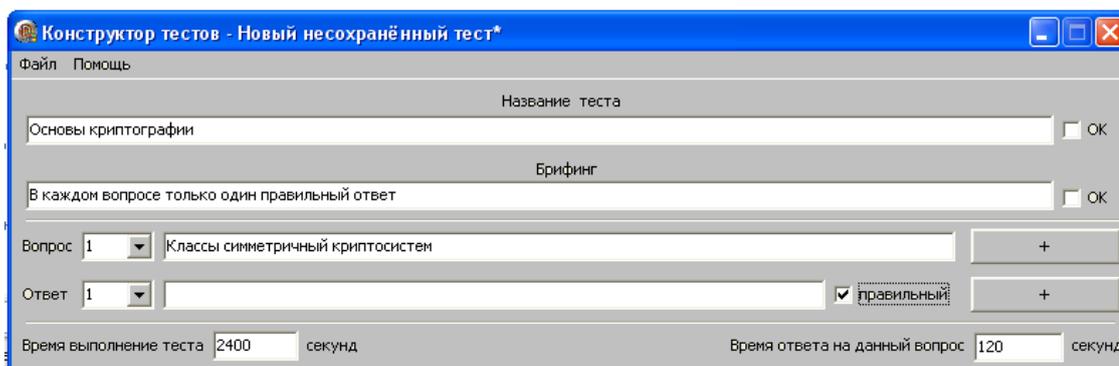


Рисунок 1 – интерфейс конструктора тестов

Возможности реализованного конструктора заключаются в варьировании следующих параметров: изменение времени прохождения теста, выбор количества ответов на вопрос, выбор количества правильных ответов. Последний параметр может варьироваться от одного до трех.

Внешний вид приложения имеет дружелюбный интерфейс и интуитивно понятен для использования, что, несомненно, является неоспоримым достоинством. В меню «Файл» расположены стандартные пункты меню, позволяющие открыть, закрыть, сохранить или создать новый тест.

В основе манипулирования данными, которые используются при создании и прохождении теста лежит трехмерный динамический массив. При закрытии приложения все данные из трехмерного массива записываются в файл, имеющий собственный формат. Данный формат может распознать только тестирующая оболочка или конструктор тестов.

Для создания нового вопроса в раскрывающемся списке необходимо выбрать его номер, затем ввести текст вопроса и нажать кнопку «+», расположенную в правой части экрана. В результате этих действий вся введенная информация запишется в динамический массив. Внешний вид окна, предназначенного для создания вопросов приведен на рисунке 2.



Рисунок 2 – Сохранение нового вопроса

Таким образом, если при создании вопроса теста форма заполнена не корректно, то система выдаст окно с соответствующим предупреждением. Внешний вид данного предупреждения приведен на рисунке 3.

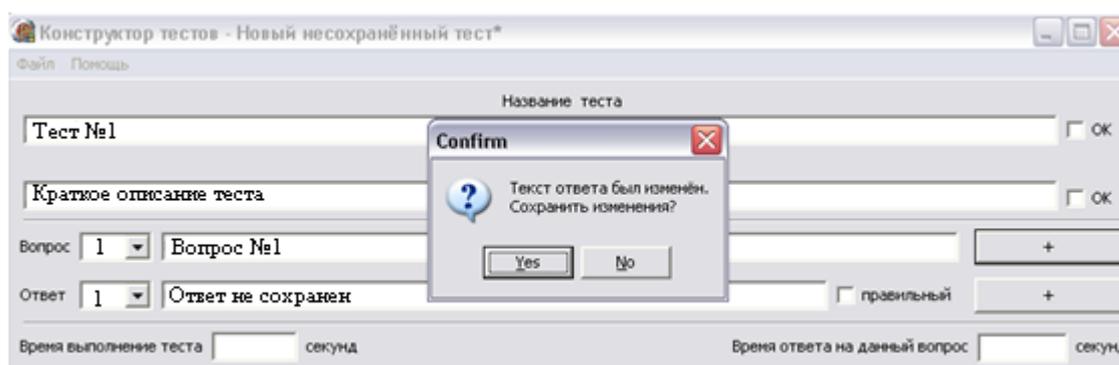


Рисунок 3 – Запрос на сохранение ответа

Для создания теста вначале указывается номер вопроса. Затем в раскрывающемся списке выбирается количество вариантов ответа на данный вопрос. В качестве правильных ответов можно указать варианты от одного до трех. Для этого устанавливается флаг «правильный» напротив верного ответа. Данный флаг располагается в правой части окна реализованного приложения.

В нижней правой части формы можно указать общее время, отведенное на тест. В этом случае время отведенное на ответ на каждый вопрос будет распределено равномерно. Но можно и указывать время в секундах на каждый вопрос отдельно, в зависимости от его уровня сложности. В такой ситуации время отводимое на тест сложится из времени отводимое на каждый вопрос.

При завершении теста вся информация о тестируемом и результаты записываются в текстовый файл, формат которого был разработан отдельно. Данный формат имеет расширение tst. С этим же форматом работает тестовая оболочка, представленная на рисунке 4.

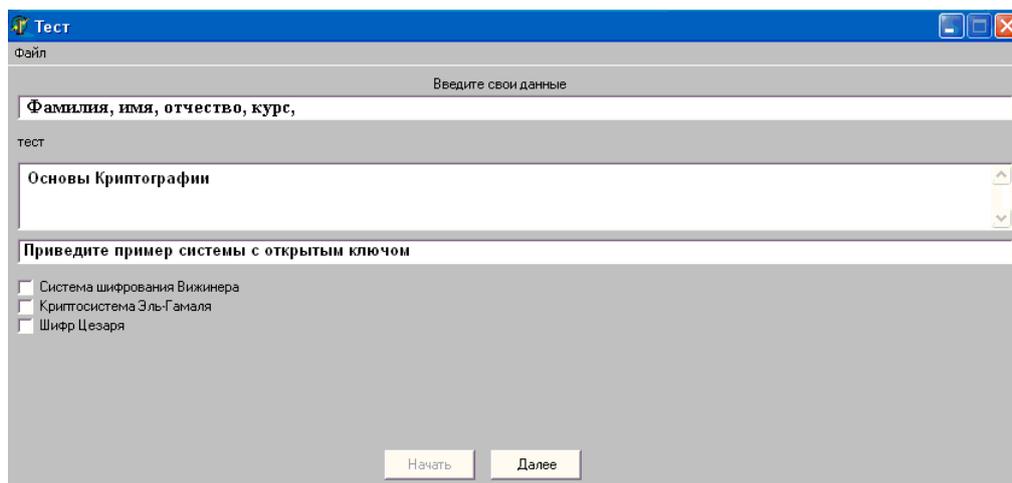


Рисунок 4 – Интерфейс приложения для тестирования студентов

Для оптимальной работы реализуемого приложения, его необходимо размещать на удаленном сервере. А для тестирования запускать с компьютеров – клиентов. Именно такая организация процесса тестирования позволяет наиболее эффективно использовать тестовую систему, сохраняя полученные данные в едином месте, что удобно и важно для проведения последующего анализа результатов тестирования.

Начало работы клиентской части приложения начинается с открытия пользовательской тестовой оболочки, запускаемой двойным щелчком мыши по ехе-файлу. В этом окне выводятся основные данные, такие как название теста и краткая информация о нем.

Далее в форме вводятся личные данные тестируемого, которые включают в себя фамилию, имя, отчество, курс и группу сохраняются в текстовый файл, вместе с ответами на вопросы теста.

При прохождении теста происходит контроль времени на каждый вопрос. Время можно распределять равномерно между вопросами, или варьировать в зависимости от уровня сложности. Если же время, отведенное на вопрос заканчивается, то система информирует об этом тестируемого и происходит автоматический переход на следующий вопрос, при этом если был получен правильный ответ, то система это учитывает. При окончании времени отведенного на целый тест реализованное приложение автоматически сохраняет полученные данные в файл и завершает свою работу.

В результате, формируется файл отчета, в котором находятся основные сведения о тестируемом, его личные данные и результаты прохождения теста. Для более эффективного процесса контроля усвоенного материала в файл также выводятся количество правильных ответов и их процентное соотношение. Для более адекватной оценки уровня усвоения изученного материала в файл также выводятся номера правильных и не правильных ответов.

В реализованном приложении для тестирования студентов присутствуют следующие ограничения на создание тестов: количество правильных ответов варьируется от одного до трех, количество ответов в одном вопросе не должно превышать десяти, количество вопросов в тесте не ограничено.

Таким образом, разработанная тестовая система включает в себя конструктор тестов и клиентскую оболочку, которые представлены двумя ехе-файлами. Конструктор тестов без каких-либо дополнительных знаний позволяет педагогу, даже плохо знакомому с информационными технологиями, создать свой тест и выполнить все необходимые настройки теста. Клиентская оболочка запускается на компьютере студента. Она позволяет выбирать и запускать сформированный в конструкторе тестов файл, содержащий тестовые задания. По прохождении теста испытуемому на экран выводится полученный результат, а также данные записываются в единый log-файл, который находится на компьютере-сервере и доступен к просмотру педагогу для проведения анализа результатов тестирования.

Литература

1. Волков В.С. Концепции современного естествознания. Интернет-тестирование базовых знаний. СПб.: Лань П, 2016. 208 с.
2. Дмитренко В.П., Сотникова Е.В., Черняев А.В. Интернет-тестирование базовых знаний. СПб.: Лань П, 2016. 160 с.

3. Чесноков А.В., Филимонова Е.М., Поляков А.Е. Теоретические положения и тестирование базовых знаний. М.: Форум, 2018. 512 с.
4. Архангельский А.Я. Программирование в Delphi. Учебник по классическим версиям Delphi. М.: Бином-Пресс, 2016. 816 с.