

УДК 658.513

Программированный контроль для мониторинга знаний студентов по инженерной графике

Воронов С.С., студент

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Проектирование и технология производства электронной аппаратуры»*

Кострица О.К., студент

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Проектирование и технология производства электронной аппаратуры»*

Научный руководитель: Суфляева Н.Е., к.т.н., доцент

Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана

bauman@bmstu.ru

Введение

В условиях острого дефицита времени, отведенного для освоения студентами общеинженерных дисциплин, требуется оптимизация учебных программ и активное использование современных арсеналов технических средств обучения и контроля знаний. При постоянно увеличивающейся учебной нагрузке на преподавателя, что выражается во все возрастающем количестве обучаемых им студентов (на кафедре инженерной графики число студентов, приходящихся на одного преподавателя, доходит до 100 и более), ему все сложнее отследить пробелы в текущих знаниях учащихся, которые в конечном итоге приводят к низкому итоговому уровню знаний по предмету.

Одним из эффективных приемов повышения качества обучения является последовательный мониторинг процесса усвоения материала по отдельным темам дисциплины, а именно - текущий программированный контроль на основе компьютерных технологий. Компьютерные системы никогда не смогут заменить квалифицированного творческого преподавания, но они могут с большой эффективностью брать на себя некоторые задачи контроля полученных знаний. В частности, компьютер может проверять корректность введенных ответов, если они являются однозначно определёнными. К этому виду контроля знаний относятся тесты с одним или несколькими ответами по каждому заданию.

Отсюда вытекает идея создания автоматизированной системы, которая позволяла бы контролировать весь цикл тестирования: создание теста, его реализация и, наконец, проверка с возможностью подробного изучения результатов. Введение подобного метода проверки знаний студентов позволяет ускорить и облегчить процесс контроля преподавателем качества обучения.

1 Описание общей концепции работы программированного контроля

Сформулируем общую задачу, решение которой мы реализовали в своей работе. Необходимо создать программное обеспечение, которое позволяло бы выполнять три основные функции:

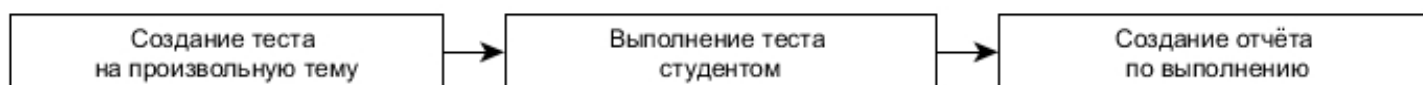


Рис.1. Схема работы системы

В нашей работе мы стремились сделать весь процесс тестирования студента как можно более гибким. Рассмотрим некоторые тонкости:

Создание теста:

- Создание теста должно включать в себя не только добавление абсолютно нового объекта, но и возможность редактирования старого.
- Важно не забыть, что каждый тест обычно имеет различные варианты с вопросами одной тематики.
- Должна иметься возможность вставить графическое изображение, чтобы затем задавать по нему вопросы или иметь его как комментарий к вопросу.
- Количество вопросов и вариантов ответа в тесте может каждый раз быть различным.
- Правильных ответов может быть несколько.

Выполнение теста:

- Студент должен иметь возможность отвечать на вопросы в любом порядке.
- Преподаватель по желанию может ограничивать прохождение теста по времени.
- Возможность вернуться к ещё не отвеченным вопросам должна быть реализована легко и понятно.
- Студент должен иметь возможность отслеживать прогресс выполнения теста.

Просмотр результатов:

- Должна иметься единая база по результатам всех тестов.
- Для лёгкости работы с данными необходима возможность их фильтрации.
- Преподаватель должен иметь возможность не просто узнать, правильно или неправильно ответил студент, а ещё и увидеть, какой ответ он выбрал, чтобы в дальнейшем работать с проблемными местами.

Отдельной проблемой стоит вопрос безопасности. Для качественного тестирования студент не должен иметь доступ к каким-либо данным теста до начала его выполнения. Должна быть исключена возможность прохождения теста за другого человека. В то же время студент не должен иметь возможности правильно выполнить тест, если до этого он узнал верную последовательность порядковых номеров ответов на задания теста.

2 Архитектура тестовой системы

На последовательном решении указанных подзадач был построен целостный продукт. Опишем принципы взаимосвязи её частей. Разработаны три отдельных проекта: программа создания тестов, программа тестирования, программа проверки результатов. К первой и последней имеет доступ только преподаватель, со второй же работает студент.

Особенностью осуществленной реализации является централизованное хранение как самих тестов, так и результатов их выполнения на удаленном сервере. После создания/редактирования теста его содержимое отправляется в базу данных на удалённом сервере посредством сети интернет. При выполнении теста студентом он загружается с сервера в момент подтверждения выбора теста, исключая хранение важных данных на локальном компьютере. После выполнения теста результаты вновь отправляются на сервер. При необходимости просмотра результатов преподавателем они загружаются из соответствующей базы данных.

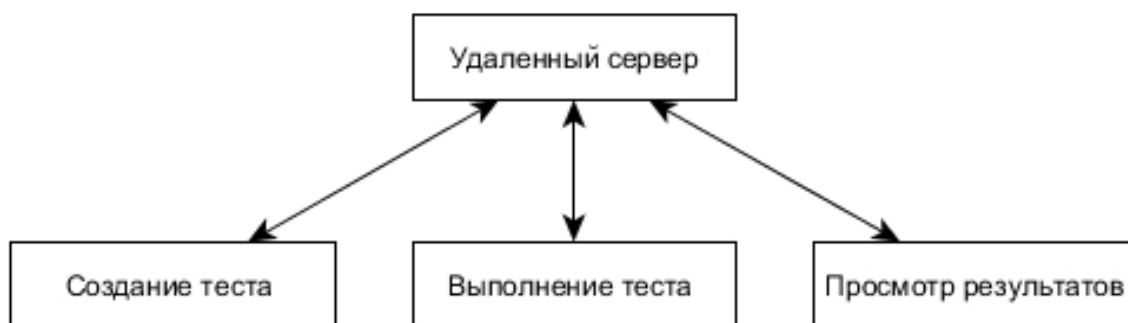


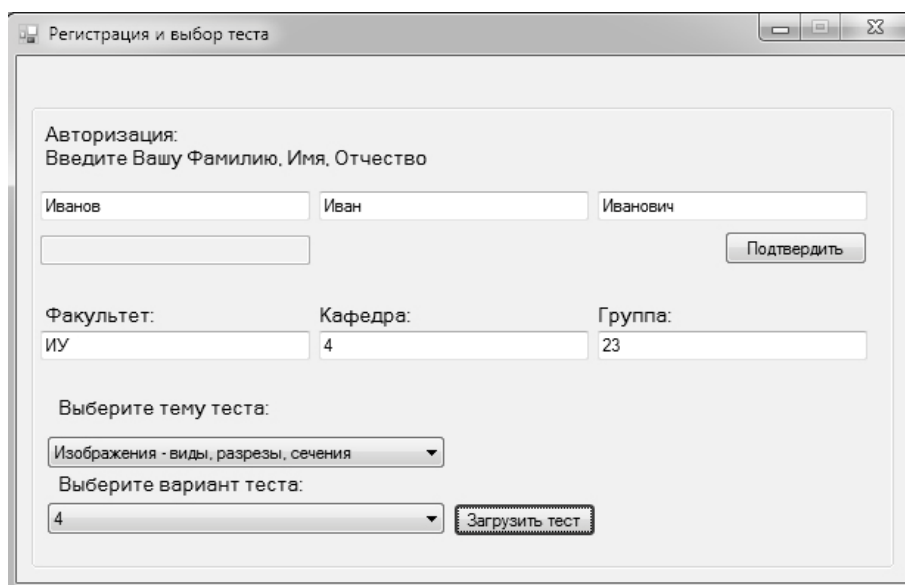
Рис. 2. Работа с удаленным сервером

3 Работа компонентов тестовой системы

В ходе работы было создано три приложения, написанных на языке С# на базе платформы Microsoft .NET Framework с использованием стандартных элементов интерфейса Windows. Данные приложения могут быть выполнены на любом компьютере операционной системы Microsoft Windows большинства версий, поддерживающих работу с библиотеками .NET Framework. Под эти задачи подходит любой университетский компьютер.

Выполнение теста:

Прежде всего, на усмотрение преподавателя, может быть составлен список студентов группы с уникальными логинами и паролями. Это исключает возможность прохождения теста за студента кем-либо ещё. В противном случае студент сам вбивает в окно регистрации свои данные и выбирает необходимый тест и его вариант (рис. 3). Все поля кроме отчества необходимы к заполнению.



Регистрация и выбор теста

Авторизация:
Введите Вашу Фамилию, Имя, Отчество

Иванов Иван Иванович

Подтвердить

Факультет: Кафедра: Группа:

ИУ 4 23

Выберите тему теста:

Изображения - виды, разрезы, сечения

Выберите вариант теста:

4

Загрузить тест

Рис. 3. Окно регистрации

Во время прохождения теста студент может переходить между заданиями с помощью кнопок «Предыдущий» и «Следующий». В любой момент времени студент может нажать на кнопку «Не отвеченные вопросы», чтобы перейти к вопросу, в котором ещё не выбран ответ, или на кнопку «Закончить», чтобы отправить результаты в базу данных. Внизу находится информационная строка, показывающая основную информацию.

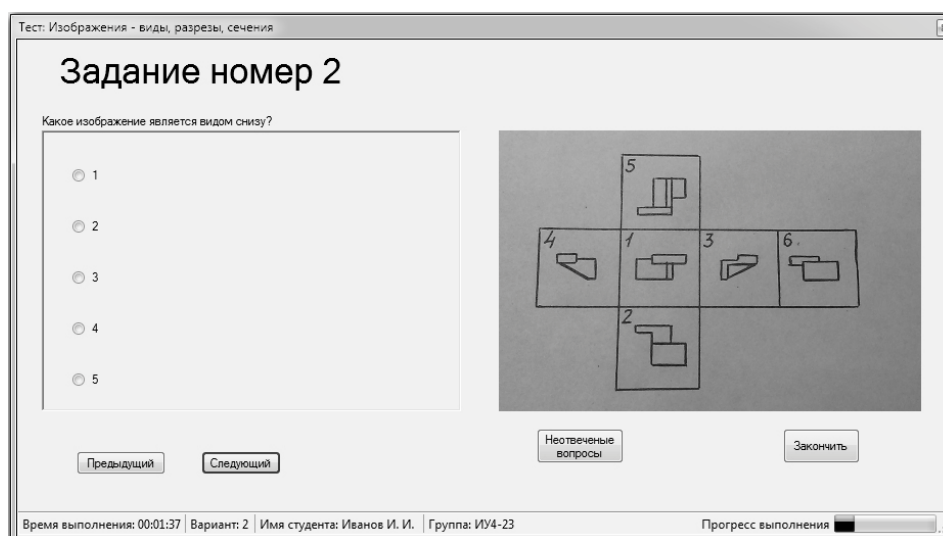


Рис. 4. Окно выполнения теста

Просмотр результатов:

Основную часть окна занимает таблица результатов. Имеется возможность упорядочивать записи по любому полю таблицы. При нажатии на конкретную запись в таблице ответов отображаются выбранные варианты тестируемого с указанием задания и правильности этого ответа. В случае выбора нескольких правильных ответов будет указан каждый.

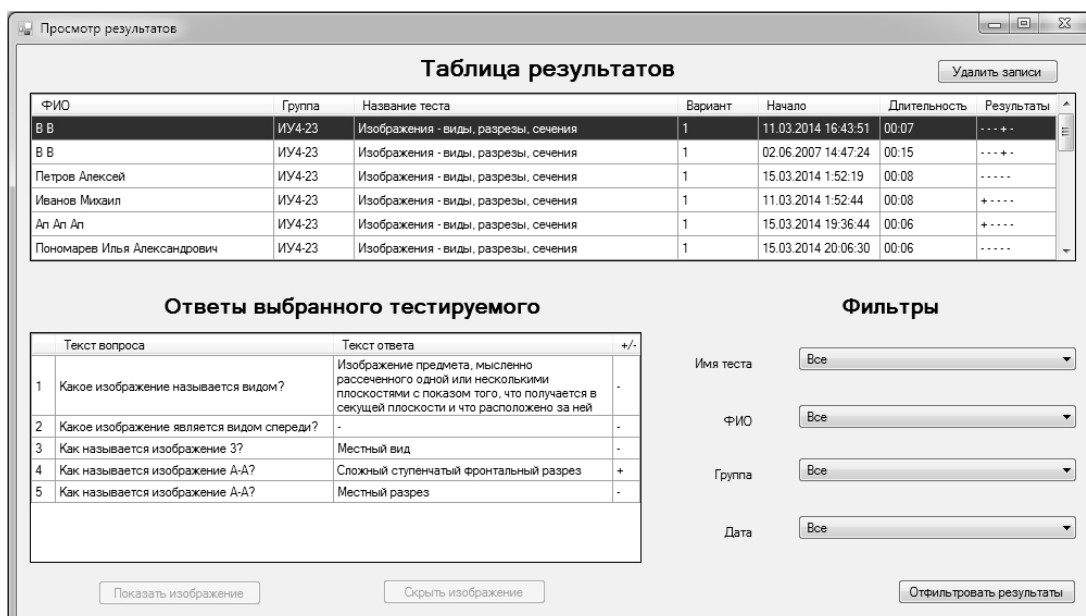


Рис. 5. Окно просмотра результатов

Если к заданию прилагалось изображение, то просмотреть его можно выбрав задание и нажав кнопку «Показать изображение». С помощью фильтров можно вывести в

таблицу результатов только те записи, которые удовлетворяют выбранному запросу. Наконец, имеется возможность удалить одну или сразу несколько записей из таблицы результатов, а значит и из удалённой базы данных.

Создание теста:

Соответствующий компонент был также разработан для добавления новых тестов в базу и редактирования уже существующих. В соответствии с реализованной структурой теста, преподаватель может добавить произвольное количество вопросов и вариантов ответа для каждого из них, отметить один или более ответов как правильный. Для каждого из вопросов устанавливается изображение, показываемое при его просмотре. Также в данную программу включена возможность загрузки любого существующего в базе теста для его редактирования. Для удобства использования и исключения возможных ошибок именования, выбор тестов происходит с помощью выпадающего списка с возможностью редактирования.

Рис. 6. Окно редактора тестов

Заключение

Тестирование – одно из проверенных средств контроля обучения студентов. В то же время программируемые компьютеры позволяют решать определённые, конкретно сформулированные задачи. Объединение двух этих мощных средств – верный шаг на пути к повышению эффективности обучения студентов.

Данный проект может быть расширен для решения большого класса практических задач. Авторы проекта планируют добавление некоторых функций для того, чтобы сделать тестирование более гибким.

Список литературы

1. Петцольд Ч. Программирование для Microsoft Windows на C#. М.: Русская редакция, 2002. 564 с.
2. Стиллмен Э., Грин Д. Изучаем C#. М.: Питер, 2012. 689 с.
3. ГОСТ 2.301-68. Форматы. М.: Изд-во стандартов, 2009. 215 с.
4. ГОСТ 2.302-68. Масштабы М.: Изд-во стандартов, 2009. 215 с.
5. ГОСТ 2.303-68. Линии. М.: Изд-во стандартов, 2009. 215 с.
6. ГОСТ 2.304-81. Шрифты вертикальные. М.: Изд-во стандартов, 2009. 215 с.
7. ГОСТ 2.305-2008. Изображения - виды, разрезы, сечения. М.: Изд-во стандартов, 2009. 215 с.