

Координатами вектора X могут быть выбраны среднеквадратичные отклонения содержания $C(x_1)$, $Si(x_2)$, $Mn(x_3)$.

Используя общий вид классифицирующего правила (9), может быть проведен расчет порогового значения и коэффициентов дискриминантной функции, позволяющий идентифицировать факторы нестабильности технологического процесса плавки – низкое качество входных параметров процесса (качество шихтовых компонентов) или неэффективность существующего процесса управления плавкой. Первый вариант соответствует попаданию вектора X в класс A , второй соответствует попаданию вектора X в класс B .

На основе обработки результатов серийных плавок было получено классифицирующее правило вида (9):

$$\begin{aligned} x^j \in A \text{ если } [-127,375x_1 + 56,328x_2 - 16,513x_3 \geq (-11,9514)] \\ x^j \in B \text{ если } [-127,375x_1 + 56,328x_2 - 16,513x_3 < (-11,9514)] \end{aligned} \quad (10)$$

На основе правила (10) может быть принято решение о том, является ли отклонение в качестве сплава причиной некачественной шихты, или несовершенством процесса плавки.

Выводы.

1. Для выбора критерия эффективности технологического процесса электродуговой плавки необходим расчет статистических характеристик, описывающих процесс, непосредственно в конкретных промышленных условиях.

2. Разработанная методика анализа технологического процесса плавки, основанная на расчете вероятности получения брака готовой продукции в зависимости от величин полей рассеивания параметров шихты, позволяет выявлять некачественные шихтовые материалы и вносить соответствующие коррективы в процессе управления плавкой.

3. Для идентификации параметров нестабильности технологических процессов, влияющих на качество управления плавкой, эффективной является байесовская стратегия.

4. Полученное классифицирующее правило позволяет по результатам химического анализа чугуна установить, является ли причиной отклонения в качестве сплава от заданного некачественная шихта или отклонения от заданных режимов в самом процессе управления плавкой.

Поступила в редколлегию 18.03.2010

УДК 004.055

Л.И. МАРЧЕНКО, ст.преподаватель, ХНУРЭ, г. Харьков

В.О. СИНЯВИН, студент, ХНУРЭ, г. Харьков

ПРОГРАММНЫЙ ПАКЕТ ДЛЯ ЛОКАЛЬНОГО ТЕСТИРОВАНИЯ

Створення локальної системи тестування із простими і комфортними інтерфейсами для викладача та студента. Система редагування дозволяє мобільно змінювати кількість тем, питань з кожної теми, відповідей з кожного питання, тривалість тестування.

Creation of the in-plant system of testing with simple and comfort interfaces for a teacher and student. The system of editing allows mobile to change a quantity by that, questions on every theme, answers on every question, duration of testing.

Введение.

В современном мире все большее количество людей стремится получить высшее образование. При этом количество преподавателей увеличивается не пропорционально количеству студентов, что приводит к существенному увеличению нагрузки на преподавателя и возможному снижению качества обучения.

Проверка знаний студентов является одним из важных вопросов процесса обучения. Применение компьютерных технологий позволяет организовать этот процесс на высоком уровне.

На современном этапе отдается предпочтение системам тестирования. Процесс проверки знаний и самоконтроля должен быть простым и комфортным как для студента, так и для преподавателя, создающего проверочные тесты.

Исходя из этого была разработана данная система компьютерного тестирования, с отдельным интерфейсом для преподавателя. Учитывая, что персональный компьютер уже не является чем-то необычным, можно с полной уверенностью заявлять, что любой студент технически способен пройти тестирование с использованием компьютера. Также использование персонального компьютера не дает студенту возможности заявлять о предвзятости преподавателя, так как персональный компьютер, по определению, не может относиться предвзято к кому бы то ни было.

Отдельный интерфейс для преподавателя позволяет создавать задания для курса, группы и, если необходимо, отдельного студента. Создание такого сервиса приводит к проблеме обеспечения невозможности влияния студента на результаты проверки.

Достаточное число студентов уже при поступлении в ВУЗ имеют знания по обходу систем информационной безопасности. Это приводит нас к следующему выводу: создавать Web-тестирование попросту опасно. Общеизвестным фактом является то, насколько просто получить несанкционированный доступ к серверу и насколько сложно защитить его от атак. Исходя из этого, целесообразно использовать локальные системы тестирования. Также при использовании локальных систем не приходится зависеть от устойчивой работы сервера, что во многих случаях так же является одним из основополагающих факторов. И еще несколько аргументов в пользу локального тестирования: проводить его можно всего лишь на одном компьютере; программный пакет можно выдать домой студентам для ознакомления и подготовки к проверке знаний; приступить к тестированию, при полной неподготовленности помещения (присутствуют только включенные компьютеры и сама программа у преподавателя на носителе) можно уже в течение 15 минут, что не слишком отличается от обычной контрольной с раздачей вопросов.

Основная часть.

Разработка данного программного пакета для локального тестирования велась в среде программного продукта компании Microsoft Visual Studio 2008 на объектно-ориентированном языке программирования C# с использованием NetFramework 3.5. Основной задачей являлась идея создания приложения,

которое было бы универсально для тестирования студентов и просто в редактировании и добавлении непосредственно разделов, подлежащих для изучения студентами.

Программный пакет **Solution TestingSystem** состоит из трех проектов: **TestingSystem**, **Client** и **SymmetricEncryptionUtility**.

TestingSystem. Главная форма проекта содержит четыре допустимых опции (рис.1). При нажатии кнопки “Добавить Раздел” появляется форма, представленная на рис.2.

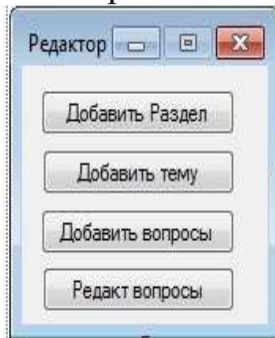


Рис.1. Вид главного окна редактора

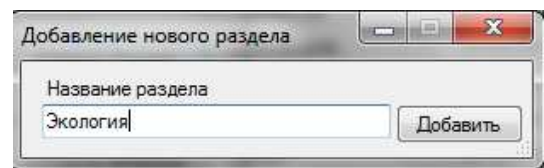


Рис. 2. Опция добавления раздела

Эта форма позволяет преподавателю создавать новый раздел вопросов для последующего тестирования. Нажимая на кнопку “Добавить тему” преподаватель видит форму, показанную на рис.3. В этом окне создается название темы и определяется количество вопросов, предполагаемое для тестирования (может изменяться в последствии).

И собственно еще две основные формы, которые позволяют создавать, добавлять и редактировать вопросы (рис.4, рис.5).

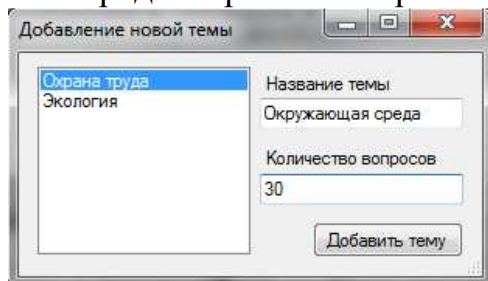


Рис. 3. Опция создания /добавления темы

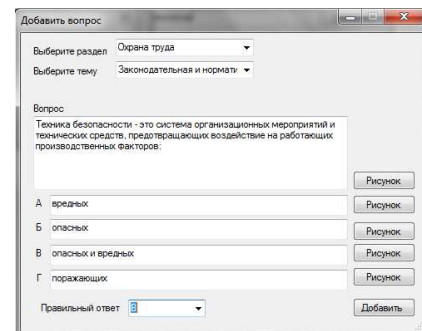


Рис.4. Опция создания /добавления вопросов

В этом окне преподаватель выбирает в выпадающем списке раздел, затем тему и добавляет вопросы. Возможность добавления рисунков в тексты вопросов улучшает качество составленных тестов.

Форма «Redactor» позволяет полностью подготовить (настроить) выбранную тему для тестирования студента: назначить раздел и тему в разделе, откорректировать содержание и количество вопросов, задать продолжительность времени для прохождения теста.

Данные формы являются очень простым решением, которое предоставляется собственно средствами Visual C#. Основная логика данной части проекта заключена в файле GetOneQuestion.cs, который отвечает за получение и расшифровку вопросов из базы данных.

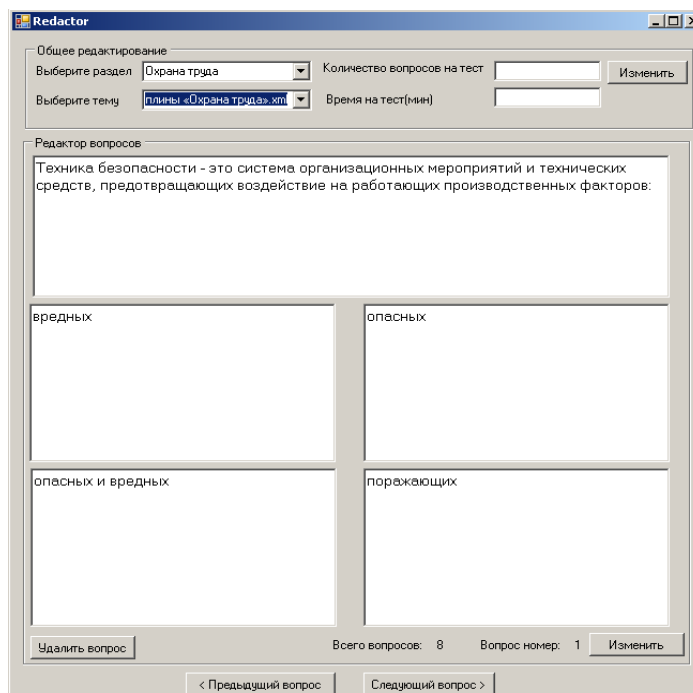


Рис.5. Форма «Redactor» - опция редактирования вопросов

SymmetricEncryptionUtility. Данная часть проекта реализована как .dll библиотека, состоящая из одного класса, которая подключается в дальнейшем к системе тестирования и редактору.

В среде .Net Framework поддерживается три типа шифрования: симметричное, ассиметричное и однонаправленное. Симметричные типы используют всегда один и тот же ключ. Для данного проекта был применен DES- абстрактный симметричный алгоритм, с необходимой длиной ключа 64, что по умолчанию реализовывает DESCryptoServiceProvider. Важным замечанием является то, что надежность шифрования пропорциональна длине ключа. Таким образом, чем больше длина ключа, тем меньше вероятность успешного взлома методом “грубой силы”, поскольку нужно проверить намного больше возможных значений ключа. Значительным преимуществом выбранного алгоритма шифрования является весьма значительная производительность.

Client. Проект Client, созданный для тестирования студентов, реализован на основе двух форм (рис.6,рис.7), позволяет выбрать раздел, тему и начать тестирование.

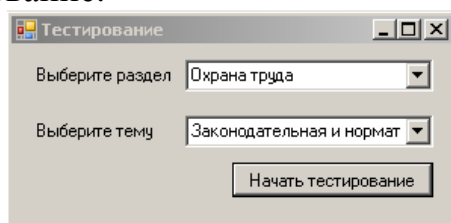


Рис.6. Главное окно тестирования

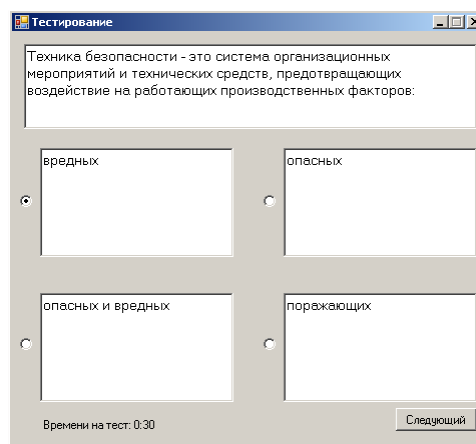


Рис.7 Окно для введения ответов

В окне для ответов (рис.7) можно увидеть текст вопроса, варианты ответа и время отведенное на тест. Эта форма отвечает за моделирование и правильное отображение вопросов, которые получают в дальнейшем посредством класса `GetOneQuestion`.

Все поля этого класса (`Fields`) являются атрибутами присущими каждому вопросу, созданными по двум основным методам: `Decrypt` и `GetOneQuestion`.

Функция `Decrypt` отвечает за правильную обработку и расшифрование вопроса получаемого из файла, средствами библиотеки `SymmetricEncryptionUtility`. А метод `GetOneQuestion` работает с файлом, выбирает нужный вопрос для отображения на экране и передает его как структуру данных в форму тестирования, где при помощи полученных данных моделирует информацию и отображает ее на экране.

Оценивание результатов тестирования отображается в процентном отношении, что наиболее адекватно отражает реальную картину успешности прохождения теста.

Заключение.

Созданный программный пакет для локального тестирования `Solution TestingSystem` позволяет решить следующие задачи:

- независимость от работы компьютерной сети;
- защиту информации от несанкционированного доступа;
- возможность работы с небольшими аппаратными средствами (достаточно одного компьютера);
- независимость от компьютерной подготовки преподавателей и студентов;
- время установки и подготовки к проведению тестирования не более 15 мин.

Программный пакет `Solution TestingSystem` построен по принципу «прозрачного прохождения», и является простым, комфортным, не требующим специальной подготовки, инструментом для создания тестов различной длины, наполнения, оформления и т.д..

Процесс тестирования для студентов с помощью данного пакета не представляет собой никаких трудностей.

Системные требования к компьютеру для данного проекта являются минимальными. За счет того, что программа полностью оптимизирована и не ресурсо-требовательна, единственным условием является установка пакета `Net Framework` на ПК, где будет проходить тестирование.

Возможности пакета для локального компьютерного тестирования позволяют использовать его при любой форме обучения практически всех дисциплин и для самостоятельной работы студентов.

Список литературы: 1.Тролсен Э. Язык программирования C# 2005 и платформа .Net 2.0 2005.-1700с.2.Шпушта М, Мак-Доналд М. Microsoft ASP.NET 2.0 с примерами на C# 2005 для профессионалов 2005.-2200с.3.Седжвик Роберт. Фундаментальные алгоритмы на C++. 2003.-800с. 4.Б. Эккель, Ч Эллисон. Философия C++. 2004.-400с.

Поступила в редколлегию 13.04.2010