

Сетевой программный комплекс тестирования знаний

Ю.А. Луцик

Белорусский государственный университет информатики и радиоэлектроники
220013, г.Минск, ул. П. Бровки, 6, ual@tut.by

Луцик Ю.А.

Резюме. Рассматриваются варианты тестов, позволяющих оценивать знания, получаемые в процессе обучения, а также структура сетевого программного комплекса, предназначенного для объективной оценки полученных знаний.

Программы тестирования знаний являются на сегодняшний день востребованным инструментом в процессе обучения. Тестирование – быстрый и эффективный способ контроля и оценки знаний. Тест – это система тестовых заданий, специфичной формы, позволяющая качественно и эффективно оценить уровень и структуру знаний испытуемых. Тестовые задания представляют не набор произвольно объединенных заданий, а систему, т.е. обладают системообразующими свойствами. Преимущества такого метода контроля знаний обучаемого очевидны:

- отсутствует вероятность субъективной оценки знаний;
- преподавателю не требуется тратить дополнительное время на проверку (по сравнению с экзаменом в традиционной форме);
- сокращается общее время проведения экзамена;
- снижается стрессовая составляющая.

Сетевые тестирующие приложения могут быть использованы в среде дистанционного обучения, т.к. позволяют полностью реализовать весь процесс тестирования без участия преподавателя. Результаты тестов могут сохраняться автоматически и использоваться в дальнейшем. Этот принцип используют системы автоматического управления учебным процессом, создающие индивидуальные базы данных на каждого студента.

Недостатком является малая информативность ответа пользователя. Выбор правильного варианта ничего не говорит о том, как он был получен.

Результатом выполненной работы является создание клиент-серверного приложения тестирования знаний, обладающее следующими возможностями:

- выбор одного или более ответов из нескольких предложенных вариантов;
- установление связей между двумя контейнерами, содержащими варианты вопросов и ответов;
- выбор из множества предлагаемой информации необходимой для формирования ответа;
- ввод формулировки возможного ответа.

При разработке приложения был использован фреймворк Qt [1]. Он позволяет запускать написанное с его помощью программное обеспечение в большинстве современных операционных систем путём простой компиляции программы для каждой системы без изменения исходного кода. Включает в себя все основные классы, которые могут потребоваться при разработке прикладного программного обеспечения, начиная от элементов графического интерфейса и заканчивая классами для работы с сетью, базами данных и XML.

При реализации сетевого приложения выбрана клиент-серверная архитектура [2]. Приложение состоит из клиентской и серверной программ, которые могут быть расположены на разных компьютерах и взаимодействовать между собой через сеть. Программа-сервер ожидает от программы-клиента запрос и предоставляет ресурсы в виде данных или сервисных функций. Сервер должен обрабатывать запросы от множества клиентов, поэтому он размещается на производительном компьютере.

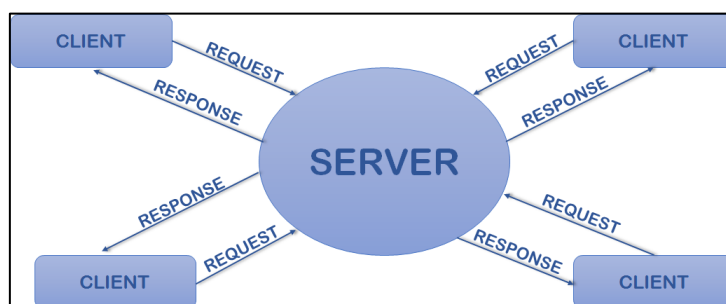


Рисунок 1 – Клиент-серверная архитектура приложения

Основной особенностью разрабатываемой системы является способ ее использования. Серверная часть данного приложения запускается в единственном экземпляре в рамках одного учебного учреждения или одной кафедры. Сервер используется множеством преподавателей, каждый из которых может дополнять базу данных вопросами по различным предметам.

В серверной части приложения можно выделить следующие блоки:

- базу данных;
- блок работы с базой данных;
- блок обработки запросов;
- блок работы с клиентом;

В клиентской части:

- блок работы с сервером;
- интерфейс преподавателя;
- интерфейс тестируемого;
- блок отображения некоторых видов вопросов;
- интерфейс администратора.

Каждый блок выполняет свою задачу. Каждый из перечисленных блоков взаимодействует с другими путем обмена данными.

Блок базы данных включает в себя СУБД, используемую серверной частью приложения. При реализации использовалась СУБД MySQL. Основными достоинствами MySQL, из-за которых была выбрана именно это СУБД, являются:

- быстродействие. СУБД MySQL является одной из самых быстрых баз данных из имеющихся на современном рынке;
- поддержка большинства функционала SQL;
- простота установки и начала использования. вместе с самой СУБД поставляется приложение MySQL workbench, позволяющее создать требуемую схему базы данных, не прибегая к SQL-запросам;
- широкий набор типов данных;
- наличие развернутой документации.

Схема базы данных включает в себя таблицы, в которых хранятся следующие данные:

- список студентов;
- список предметов;
- список вопросов, при этом каждый вопрос может относиться более чем к одному предмету;
- список тестов;
- список преподавателей и администраторов.

В проекте СУБД расположена на компьютере с серверной частью приложения.

Блок работы с базой данных обеспечивает взаимодействие с СУБД: формирование SQL-запросов, необходимых для поиска, добавления, удаления, изменения требуемых записей.

Блок работы с клиентом обеспечивает процесс авторизации клиента и передачу принятого запроса в блок обработки запросов.

Блок обработки запросов. Для обработки запроса необходимо определить его тип и выполнить десериализацию данных – перевод последовательности байтов в требуемую структуру данных, которая в дальнейшем передается блоку, осуществляющему работу с базой данных. Если в запрос к базе данных предполагает получение информации из нее, то перед передачей через сеть данные должны пройти обратный процесс – сериализацию. Далее данные передаются клиенту, после чего соединение закрывается.

Блок работы с сервером отвечает за установление соединения с сервером, проведение авторизации и проверку результата авторизации, получение вопросов для теста и отправку ответов на сервер, отправку запросов и прием результатов запросов.

Интерфейс тестируемого выполняет отображение вопросов и навигацию по ним, хранит ответы, введенные пользователем до момента завершения теста. По истечении времени, отводимого на тест, окно закрывается, ответы отправляются на сервер. Для просмотра отображается оценка, полученная тестируемым.

Блок отображения некоторых видов вопросов необходим для применения в тесте вопросов на упорядочивание и группировку.

Интерфейс преподавателя позволяет преподавателю выполнять следующие операции:

- осуществлять редактирование базы данных списка вопросов;

- формировать тест из уже имеющегося списка вопросов;
- редактировать список предметов (дисциплин);
- осуществлять проведение теста и наблюдать за получаемыми оценками;
- просматривать результаты проведенных тестов в виде отчета, который содержит текст вопроса, правильный ответ на вопрос и ответ тестируемого.

После окончания тестирования преподавателю доступна информация как о полученных оценках, так и об ответах каждого студента.

Интерфейс администратора позволяет производить следующие действия:

- поиск, добавление и удаление из базы данных пользователей данного приложения;
- смена паролей пользователей.

Ниже приведены некоторые скриншоты работы приложения

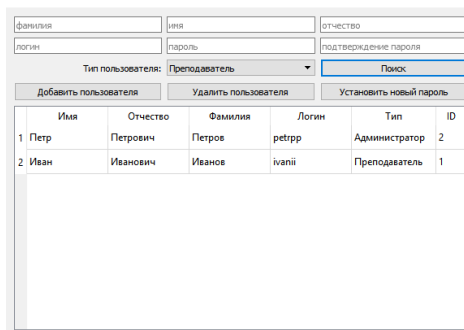


Рисунок 2 – Интерфейс администратора

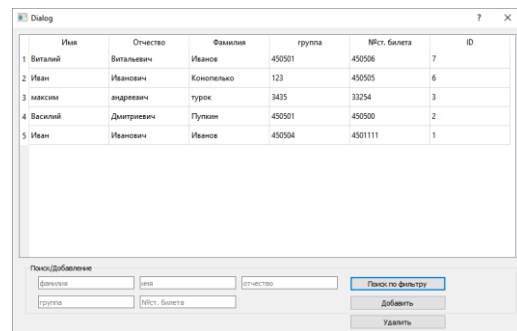


Рисунок 3 – Управление студентами

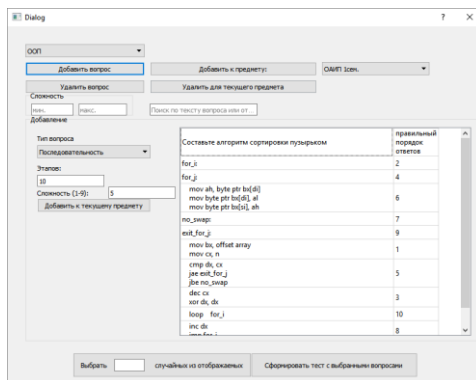


Рисунок 4 – Добавление вопроса

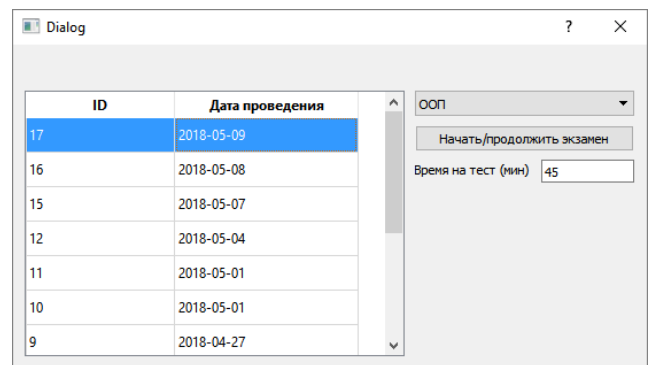


Рисунок 5 – Выбор теста

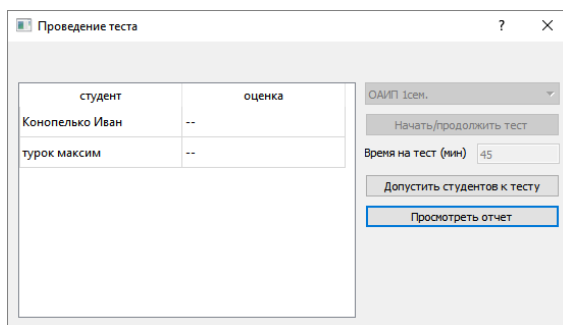


Рисунок 6 – Проведение теста

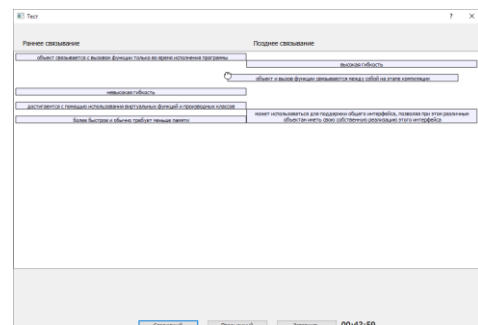


Рисунок 7 – Отображение вопроса теста

Литература

[1] Qt Documentation [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://doc.qt.io/>.

[2] Types of Client-Server Architecture [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://msatechnosoft.in/blog/tech-blogs/types-of-client-server-architecture/>.