

Литература

- 1 Ле-Бодик, Г. Мобильные сообщения: службы и технологии SMS, EMS и MMS: пер. с англ. / Г. Ле-Бодик. – М.: КУДИЦ-ОБРАЗ, 2005. – 448 с.
- 2 Таненбаум, Э. Архитектура компьютера / Э. Таненбаум. – СПб.: Питер, 2007. – 844 с.
- 3 Отправка СМС / NHSMS 1.6.0 [Электронный ресурс] // URL: http://otpravka/sms.ucoz.ru/blog/nsms_1_6_programma_dlja_sistemy_opoveshenija_cherez_sms/2010-02-23-6.
- 4 Отправка СМС / Simple SMS Lite [Электронный ресурс] // http://otpravkasms.ucoz.ru/blog/skachat_simplesms_lite_dlja_otpravki_ms_s_kompjutera_cherez_telefon/2010-01-29-23.
- 5 Хедвинд / SMS Коммуникатор [Электронный ресурс] / Хедвинд // URL: <http://www.headwind.ru/communicator.php>.

УДК 004.7

Е. Ю. Лукашов-Фурсиков, В. А. Короткевич

СИНХРОНИЗАЦИЯ БАЗ ДАННЫХ ПРОГРАММНОГО КОМПЛЕКСА «ПРИЕМНАЯ КОМИССИЯ ВУЗА»

Статья рассказывает об основных этапах и особенностях разработки приложения в среде C++ Builder, предназначенного для синхронизации баз данных программного комплекса «Приемная комиссия вуза». Разработанное приложение ориентировано на администратора программного комплекса. Реализовано два режима синхронизации. Первый режим позволяет выполнять копирование данных по запросу администратора программного комплекса. При работе во втором режиме приложение автоматически выполняет копирование данных по заданному расписанию.

Программный комплекс «Приемная комиссия вуза» использует два SQL-сервера. Первый используется оконными приложениями, которые предназначены для секретарей приемной комиссии и руководства вуза. На втором хранятся данные, используемые интернет-сервисом для абитуриентов – «Личный кабинет». Разделение данных на два сервера вызвано требованиями производительности, надежности и защищенности системы.

В процессе функционирования программного комплекса обеспечивается синхронизация данных между серверами. При этом анкетные данные, введенные абитуриентами в личном кабинете, копируются на сервер, доступный секретарям приемной комиссии, а сообщения для абитуриентов, данные таблиц-справочников, данные текущего конкурса и проходные баллы копируются на сервер интернет-сервиса для отображения абитуриентам. Схема синхронизации данных изображена на рисунке 1.



Рисунок 1 – Схема синхронизации данных

Для выполнения поставленной задачи в среде C++ Builder было разработано приложение для копирования данных. Интерфейс приложения представлен набором вкладок (страниц), каждая из которых содержит компоненты для настройки параметров копирования определенных данных (рисунок 2).

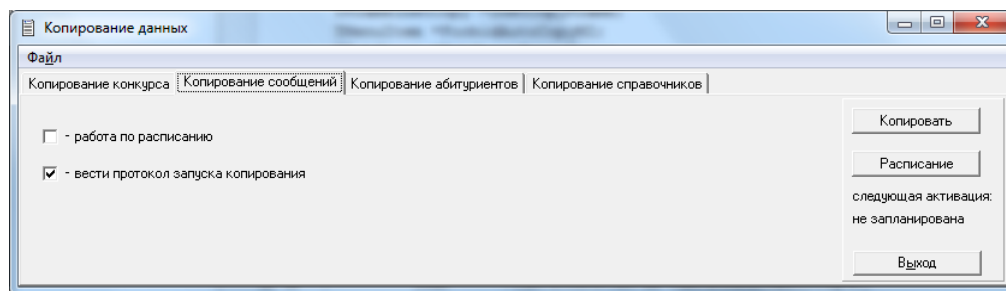


Рисунок 2 – Интерфейс приложения синхронизации данных

Для реализации многостраничного интерфейса был использован стандартный компонент C++ Builder – *TPageControl*, представляющий собой многостраничную панель. Каждая страница этого компонента является объектом типа *TTabSheet*. *TTabSheet* – это панель, на которой можно размещать любые управляющие компоненты, окна редактирования и т. п. В компоненте *TPageControl* имеется ряд методов, которые позволяют оперировать страницами, создавать их, уничтожать, переключать [1].

Во всех видах копирования имеется стандартный набор функций управления и настройки процесса синхронизации данных. За выполнение каждой функции и настройки отвечают определенные компоненты, которые представлены в виде кнопок (*TButton*), индикаторов с флажком (*TCheckBox*) и др. Все эти компоненты расположены на специальной панели, называемой фреймом. По своей сути фрейм представляет собой некий фрагмент окна приложения, способный переноситься на разные формы, в разные приложения и допускающий использование преимуществ наследования. Фрейм проектируется отдельно, как самостоятельное окно, имеет свой модуль .cpp, включается в Депозитарий и используется так же, как и форма, включая наследование. Однако фрейм не является самостоятельным окном Windows, может отображаться только на форме или другом контейнере и имеет свойства, методы и события, подобные панели, а не форме [2].

В процессе разработки приложения был создан родительский фрейм с базовым функционалом. Затем от базового фрейма были созданы наследники, расширяющие этот функционал. Этот подход позволяет значительно упростить разработку новых и дальнейшее сопровождение имеющихся модулей копирования. Ко всему прочему обеспечивается объектно-ориентированный стиль разработки приложения.

В приложении копирование осуществляется двумя способами:

- по запросу администратора программного комплекса;
- автоматически (по расписанию).

Для осуществления копирования данных вручную администратору программного комплекса необходимо нажать кнопку «Копировать» на вкладке, отвечающей за копирование нужных ему данных.

Для выполнения автоматического копирования администратору заранее необходимо задать расписание. Для этого на странице нужного модуля копирования необходимо нажать кнопку «Расписание». После этого отобразится окно настройки расписания, представленное на рисунке 3.

Работа по расписанию может осуществляться в одном из двух режимов: по периоду или по времени. Переключение между этими режимами обеспечивается стандартным компонентом C++ Builder для выбора одной из нескольких взаимоисключающих альтернатив – *TRadioButton*. Для первого режима с помощью полей ввода (компонент

TDateTimePicker) задается временной интервал с точностью до минут, а также периодичность синхронизации – количество минут, через которое будет производиться очередное копирование данных (компонент *TEdit*). Также предусмотрена возможность круглосуточного запуска. При выборе режима «по времени» необходимо задать список времен с точностью до минут, согласно которому будет выполняться копирование данных. Для отображения списка времен используется стандартный компонент *TComboBox*. При необходимости данный список можно редактировать, добавляя (кнопка «Добавить»), удаляя (кнопка «Удалить») или изменяя записи. Независимо от выбранного режима работы необходимо отметить дни недели, в которые будет запускаться синхронизация. Для этого используется стандартный компонент *TCheckBoxList*. Также имеется возможность в случае необходимости задавать конечную дату и время, при достижении которой процесс автоматического копирования данных прекращается.

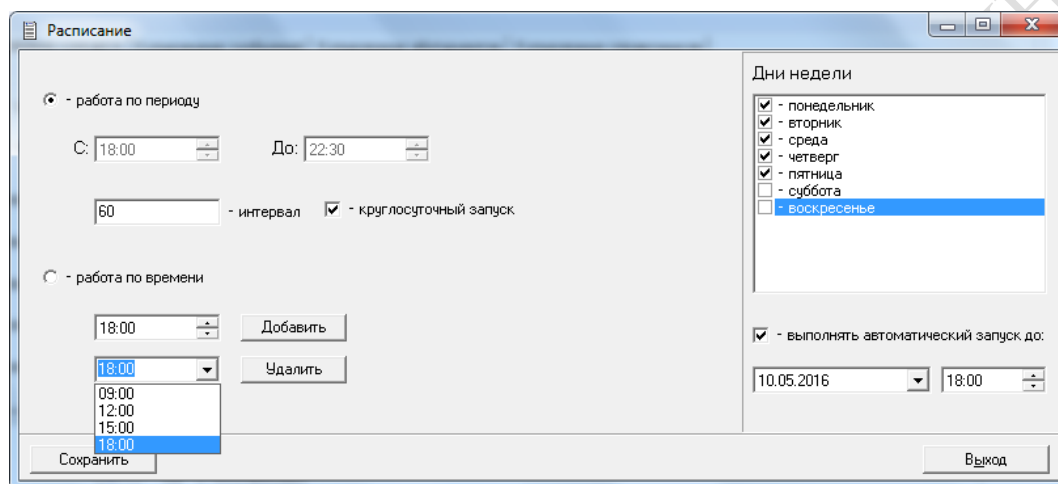


Рисунок 3 – Окно настройки расписания копирования

По завершении настройки расписания администратору необходимо отметить «работа по расписанию» на вкладке соответствующего модуля для запуска процесса копирования по расписанию. После этого будет вычислен промежуток времени (интервал) в миллисекундах до следующего запуска копирования данных. Дата и время следующего запуска отображается под кнопкой «Расписание». Функцию таймера берет на себя стандартный компонент C++ Builder – *TTimer*. Его свойство *Interval* задает период срабатывания таймера, а свойство *Enabled* – доступность таймера. Через заданный интервал времени таймер срабатывает, вызывая событие *OnTimer* [3]. В обработчике этого события вызывается метод копирования, переопределенный во фрейме-наследнике. Также вычисляется новый интервал времени до запуска следующего копирования данных.

Используя стандартный компонент *TDateTimePicker* для копирования данных текущего конкурса и проходных баллов, дополнительно реализована возможность задавать предельные дату и время копирования для каждой формы обучения. После достижения этой даты данные о текущем конкурсе и проходные баллы соответствующей формы обучения копироваться не будут. Страница модуля копирования текущего конкурса и проходных баллов изображена на рисунке 4.

Помимо основных компонентов на вкладке копирования справочников имеются компоненты для отображения, задания и редактирования списка таблиц-справочников, данные которых будут копироваться на сервер интернет-сервиса. Для отображения списка используется специальный компонент, созданный на основе стандартного компонента C++ Builder *TDBGrid*. Для связи с таблицей базы данных, содержащей список справочников, используется компонент, созданный на основе стандартного *TADOTable*. Страница копирования справочников изображена на рисунке 5.

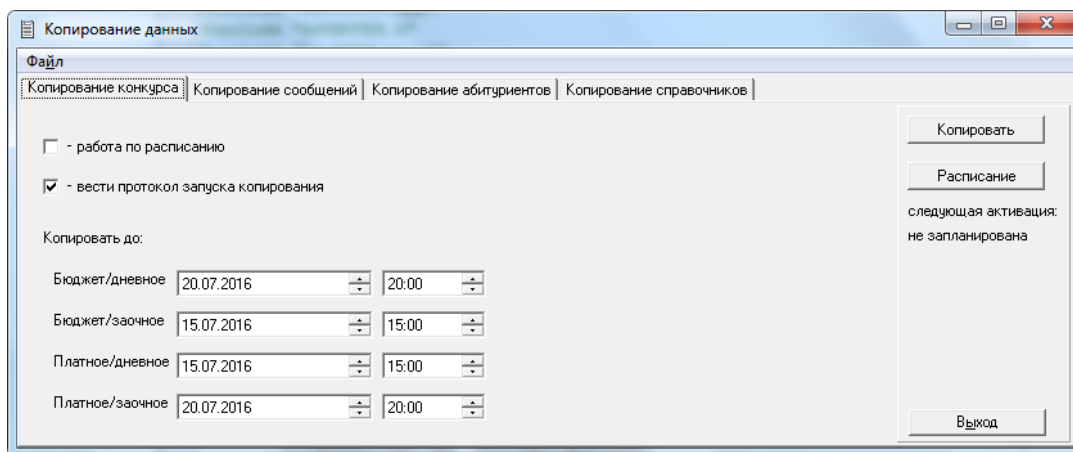


Рисунок 4 – Страница копирования текущего конкурса и проходных баллов

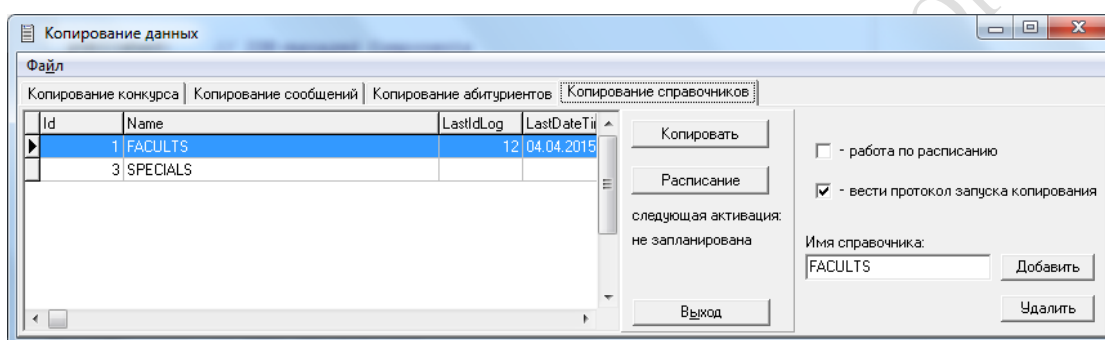


Рисунок 5 – Страница копирования таблиц-справочников

Разработанное приложение входит в состав программного комплекса «Приемная комиссия вуза», будет внедрено в эксплуатацию в ходе приемной компании 2016 года в ГГУ им. Ф. Скорины.

Литература

- 1 Архангельский, А. Я. Программирование в С++ Builder / А. Я. Архангельский. – СПб.: Бином, 2010. – 1304 с.
- 2 Холингворт, Дж. Borland С++ Builder 6. Руководство разработчика / Дж. Холингворт, Б. Сворт, М. Кэшмэн, П. Густавсон. – М.: Вильямс, 2003. – 976 с.
- 3 Шамис, В. Borland С++ Builder 6. Для профессионалов / В. Шамис. – СПб.: Питер, 2003. – 800 с.

УДК 372.853

В. Д. Мельниченко

СПОСОБЫ ОБОБЩЕНИЯ И СИСТЕМАТИЗАЦИИ ЗНАНИЙ УЧАЩИХСЯ ПО ФИЗИКЕ

В статье рассмотрены некоторые методики проведения урока, направленные на обобщение и систематизацию знаний учащихся. Представлен разработанный автором план-конспект урока обобщения и систематизации знаний по теме «Расчет количества теплоты при нагревании и охлаждении. Горение. Плавление». Использованная