

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ ПРОИЗВОДСТВЕННЫМИ ПРОЦЕССАМИ РЕСТОРАНА НА ОСНОВЕ MICROSOFT ACCESS, РАБОТАЮЩАЯ В РЕЖИМЕ КЛИЕНТ-СЕРВЕРНОГО ПРИЛОЖЕНИЯ

AUTOMATED SYSTEM FOR RESTAURANT PROCESS CONTROL BASED ON MICROSOFT ACCESS, OPERATING IN CLIENT-SERVER APPLICATION MODE

**A. Goncharov
D. Yashin
K. Brattsev**

Summary. Modern automated control systems allow solving complex problems in the management of production processes in public catering. New opportunities provided by the use of automation of processes in public catering enterprises make it possible to improve the quality of obtaining objective information about the changes taking place at management facilities, the quality of service, and the efficiency of the enterprise.

The proposed automated restaurant production process control system will improve the quality of customer service and production efficiency.

Keywords: automation, control system, client-service application, process, restaurant.

Гончаров Андрей Валерьевич

Кандидат технических наук, доцент, Московский государственный университет технологий и управления имени К.Г. Разумовского
a.goncharov@mgutm.ru

Яшин Денис Дмитриевич

Кандидат педагогических наук, доцент, Московский государственный университет технологий и управления имени К.Г. Разумовского
d.yashin@mgutm.ru

Братцев Кирилл Евгеньевич

Кандидат технических наук, доцент, Московский государственный университет технологий и управления имени К.Г. Разумовского
bk@fpv.ru

Аннотация. Современные автоматизированные системы управления позволяют решать сложные задачи в управлении производственными процессами в общественном питании. Новые возможности, которые дает применение автоматизации процессов на предприятиях общественного питания, позволяют повысить качество получения объективной информации об изменениях, происходящих на объектах управления, качество обслуживания, эффективность работы предприятия.

Предложенная автоматизированная система управления производственными процессами ресторана позволит повысить качество обслуживания клиентов и эффективность производства.

Ключевые слова: автоматизация, система управления, клиент-сервисное приложение, процесс, ресторан.

В настоящее время качественная система автоматизации ресторана позволит сделать управление рестораном эффективным, облегчит управление работой заведения. Основные задачи, которые решают системы автоматизации в ресторанном бизнесе: скорость и качество обслуживания гостей; формализация процесса взаимодействия между персоналом; минимизации потерь от воровства; минимизация потерь на баре; автоматизация складского учета; интеграция с тарифицируемым оборудованием; поддержка системы лояльности; управление персоналом; обеспечение финансовой подсистемы [3, 5].

Delphi — это RAD, или Rapid Application Development — среда быстрой разработки приложений. Иначе говоря, Delphi включает в себя не только все средства для работы с языком Object Pascal (тем более что, начиная с версии, 2005 в Delphi имеется поддержка других языков программирования, в частности, C#), но и ряд дополнительных средств, призванных максимально ускорить и упростить создание программ. К таковым средствам относятся, прежде всего, визуальный редактор форм, при помощи которого за считанные минуты и без лишних усилий можно создать полноценно выглядящую программу, а также прочие составные

части визуальной составляющей разработки программ [1, 2]. В результате время разработки программ существенно сокращается, поскольку отпадает необходимость просчитывать вручную расположение каждого элемента пользовательского интерфейса. Вкупе с развитыми средствами для написания и отладки кода — специализированным текстовым редактором, оптимизирующим компилятором и отладчиком, Delphi является средством быстрой разработки приложений [4]. При этом основой Delphi для конечного пользователя является IDE — Integrated Development Environment (интегрированная среда разработки), которая объединяет в себе редактор кода и средства визуальной разработки, а также связывает это с компилятором, средствами разработки баз данных и прочими составными частями Delphi.

MS Access — это система управления базами данных под управлением Windows. MS Access входит в пакет Microsoft Office. Запуск MS Access Пуск/Программы/MicrosoftAccess. Объектом обработки MS Access является файл базы данных, имеющий произвольное имя и расширение.mdb. Объекты MS Access: таблицы, формы, запросы и отчеты.

Постановка задачи

1. Организационно-экономическая сущность задачи. Для предприятий, работающих в сфере общественного питания очень важно оперативно и своевременно вести учёт продуктов и заказов. Поэтому задача автоматизации обработки данных ресторана является одной из приоритетных задач развития предприятия. Основной целью задачи является своевременный оперативный учёт заказов клиентов, а также решаются задачи составления отчётов и других необходимых документов. Автоматизация позволит снизить трудозатраты и число ошибок работников предприятия при обработке данных. Также она приведет к экономии времени сотрудников и повысит оперативность принимаемых решений. Централизованное хранение данных повышает точность и достоверность предоставляемых сведений. Вся информация необходимая для решения задачи «Автоматизация заказа блюд в ресторане» хранится в файлах баз данных на магнитном носителе. Автоматизация позволит получать отчёты по заданным условиям, с подсчётом итоговых значений, и организовать хранение данных для последующего анализа. Условием прекращения использования задачи «Автоматизация заказа блюд в ресторане» является прекращение деятельности предприятия [6].
2. Описание выходной информации. В процессе решения задачи могут быть получены следующие

выходные документы: «Меню ресторана», которое отражает информацию о блюдах, которые может заказать клиент; «Заказ», содержащий информацию о заказанных блюдах, передаваемый на кухню.

3. Описание входной информации.

Входные документы задачи «Автоматизация заказа блюд в ресторане» подразделяются на оперативно-учетную, условно-постоянную и нормативно справочную информацию. Для решения задачи оперативно-учётной информацией используется документ «перечень блюд на текущий день», в котором указаны тип блюда и его наименование.

Предметной областью называется часть реальной системы, представляющая интерес для данного исследования. При проектировании автоматизированных систем предметная область отображается моделями данных нескольких уровней. Число уровней зависит от сложности решаемых задач, но в любом случае включает концептуальный и логический уровни. Предметной областью созданной базы данных является деятельность ресторана [7]. Описаны три основных вида деятельности:

1. Кухня. Производится непосредственно приготовление или «набор» блюд из имеющихся ингредиентов, кроме того, выполняется заказ на необходимые ингредиенты, а также формирование меню или «набор» блюд из уже приготовленных.
2. Бухгалтерия. Производятся все необходимые расчеты.
3. Обслуживание клиентов. Производится обслуживание посетителей ресторана, которым предоставляется меню и выполняется заказ выбранных ими блюд.
4. В базе данных разрабатываемого программного продукта будут храниться данные об общем ассортименте блюд:
 - ◆ Код блюда;
 - ◆ Наименование блюда;
 - ◆ Объем;
 - ◆ Время приготовления;
 - ◆ Стоимость, руб.;

Персонал, который будет обслуживать посетителей ресторана, характеризуются следующими параметрами:

- ◆ Код работника;
- ◆ Фамилия;
- ◆ Имя;
- ◆ Отчество;
- ◆ Дата рождения;
- ◆ Номер/серия паспорта;
- ◆ Адрес;

- ◆ Телефон.

Кроме того, отдельно существует информация о должностях, занимаемых тем или иным работником:

- ◆ Код должности.
- ◆ Название должности.
- ◆ Оклад.
- ◆ Выполняемые функции.

Также в программе содержатся данные о постоянных клиентах:

- ◆ Код клиента;
- ◆ ФИО;
- ◆ Контактный телефон.

При формировании заказа учитываются следующие данные:

- ◆ Наименование блюда;
- ◆ Количество блюд;
- ◆ Цена блюда;
- ◆ Стоимость блюда;

Для удобства хранения блюд по разделам существует информация о видах блюд:

- ◆ Код вида;
- ◆ Наименование вида.

В создаваемом программном продукте могут работать группы пользователей:

1. Обслуживающий персонал ресторана.
2. Администратор ресторана.

Постановка задачи

Основные цели и задачи. Построить базу данных позволяющую повысить эффективность работы ресторана с партнерами и клиентами, и автоматизировать документооборот внутри предприятия ООО «Шик»:

1. Должности
2. Заказ
3. Меню
4. Склад
5. Сотрудники

Описание технологии функционирования БД

Редактированием (добавление, изменение, удаление) общего ассортимента блюд в ресторане занимается повар, он следит за наличием тех или иных продуктов для приготовления блюд.

Администратор на основе получаемых данных формирует меню на 6 различных листах (соответствующих 6 разделам блюд), а затем просматривает и выводит

на печать. Такие операции как фильтрация, поиск и сортировка данных о блюдах доступна и повару, и администратору.

Клиент, придя в ресторан, просматривает готовое меню и осуществляет заказ согласно собственным предпочтениям. Официант подает данные о заказываемых блюдах бухгалтеру и повару. А также информацию о карте постоянного клиента (если такая имеется) бухгалтеру, который, в свою очередь, формирует общий заказ. А затем чек заказа с учетом скидки. На чеке отображается фамилия, имя, отчество обслуживающего официанта, номер столика, дата и время заказа [6, 7].

Данный чек распечатывается и вкладывается в папку «Счёт» для предоставления клиенту. Администратор, занимающийся кадрами, формирует список персонала. Предусмотрено добавление, изменение, удаление записей, а также следит за наличием постоянных клиентов и размером их скидки.

Требования к базе данных

База данных предназначена для формирования меню и заказов клиентуры ресторана. Система должна вести учет первичной документации, учет работников и самой клиентуры, а также формировать необходимые отчеты. Главным назначением базы является автоматизированная деятельность ресторана.

База данных может работать на одном компьютере или в локальной сети. Основной режим работы — диалоговый, т.е. пользователь работает с базой данных, используя объекты форм поля, кнопки, списки, получая при этом необходимые отчеты в Rave Designer, используя для этого специальные шаблоны. Необходимо предусмотреть поиск, отбор и сортировку документов, используя и другие критерии к базе данных.

1. Реляционная модель базы нормализована и состоит из 6 таблиц.

2. Приложение БД имеет удобный для пользователя интерфейс, выполненный в одном стиле.

3. Для разработки приложения выбрана среда разработки Delphi. Delphi — это продукт Borland International для быстрого создания приложений.

Высокопроизводительный инструмент визуального построения приложений включает в себя настоящий компилятор кода и предоставляет средства визуального программирования, несколько похожие на те, что можно обнаружить в Microsoft Visual Basic или в других инструментах визуального проектирования. В основе

Delphi лежит язык Object Pascal, который является расширением объектно-ориентированного языка Pascal. В Delphi также входят локальный SQL-сервер, генераторы отчетов, библиотеки визуальных компонентов, и прочее хозяйство, необходимое для того, чтобы чувствовать себя совершенно уверенным при профессиональной разработке информационных систем или просто программ для Windows-среды. Преимущества Delphi по сравнению с аналогичными программными продуктами:

- ◆ создаваемые с помощью Delphi программы могут работать не только под управлением Windows, а сама она относится к классу инструментальных средств ускоренной разработки программ (RAD);
- ◆ быстрота разработки приложения;
- ◆ высокая производительность разработанного приложения;
- ◆ низкие требования разработанного приложения к ресурсам компьютера;
- ◆ наращиваемость за счет встраивания новых компонент и инструментов в среду Delphi;
- ◆ возможность разработки новых компонент и инструментов собственными средствами Delphi (существующие компоненты и инструменты доступны в исходных кодах);
- ◆ удачная проработка иерархии объектов.

4. Для реализации приложения по автоматизации обслуживания посетителей ресторана мы выбрали СУБД Paradox 7. Принцип хранения данных в Paradox сходен с принципами хранения данных в dBase — каждая таблица хранится в своем файле (расширение *.db), MEMO- и BLOB-поля хранятся в отдельном файле (расширение *.md), как и индексы (расширение *.px).

5. Однако, в отличие от dBase, формат данных Paradox не является открытым, поэтому для доступа к данным этого формата требуются специальные библиотеки. Так как в этой ситуации доступ к данным осуществляется только с помощью «знающих» этот формат библиотек, простое редактирование подобных данных по сравнению с данными открытых форматов типа dBase существенно затруднено. В этом случае возможны такие недоступные при использовании «открытых» форматов данных сервисы, как защита таблиц и отдельных полей паролем, хранение некоторых правил ссылочной целостности в самих таблицах — все эти сервисы предоставляются Paradox, начиная с:

- ◆ первых версий этой СУБД. Достоинства данной СУБД заключаются в том, что она содержит:
- ◆ Средства манипуляции данными Paradox и dBase.
- ◆ Средства создания форм, отчетов и приложений.
- ◆ Средства визуального построения запросов.
- ◆ Средства публикации данных и отчетов в Internet и создания Web-клиентов.

- ◆ Corel Web-сервер.
- ◆ ODBC-драйвер для доступа к данным формата Paradox из Windows-приложений.
- ◆ Средства для доступа к данным формата Paradox из Java-приложений.

6. Работа с приложением организована с помощью меню, расположенного на главном окне, обеспечив тем самым быстрый доступ пользователя ко всей информации, хранящейся в базе данных. Приложение позволяет выполнять следующие операции с данными:

- ◆ ввод данных; в отдельных случаях, с проверкой вводимых данных и выдачей соответствующих сообщений в случае ошибок при вводе данных;
- ◆ удобный просмотр, сортировку, поиск и фильтрацию данных;
- ◆ формирование запросов и получение с помощью них необходимой информации из БД (в том числе и запросы с группировкой);
- ◆ формирование печатной формы (отчётов).

Функционирование приложения ориентировано на конечного пользователя, удобно в работе и выполняет все необходимые для пользователя действия. Таким образом, пользователь может просматривать, добавлять и удалять записи о сотрудниках, клиентах и заказах. Также он может осуществлять поиск необходимой информации, составлять отчёты. Также пользователь может настраивать интерфейс программы согласно собственным предпочтениям, изменяя расположение компонентов на форме, шрифт надписей, подписи на кнопках и т.д. Кроме того, он может располагать окна формы по своему усмотрению.

Модель данных представляет объекты и их взаимосвязи без указания способов их физического хранения. Таким образом, концептуальная модель является, по существу, моделью предметной области. При проектировании концептуальной модели должна происходить структуризация данных и выявление взаимосвязей между ними без рассмотрения особенностей реализации и вопросов эффективности обработки. Проектирование концептуальной модели основано на анализе задач, стоящих перед рекламным агентством. Данная модель включает описания объектов и их взаимосвязей, представляющих интерес в рассматриваемой предметной области и выявляемых в результате анализа данных.

Чтобы построить необходимую нам модель, мы привели все имеющиеся данные к третьей нормальной форме, в результате чего получили следующие сущности:

- ◆ Блюда.
- ◆ Виды блюд.

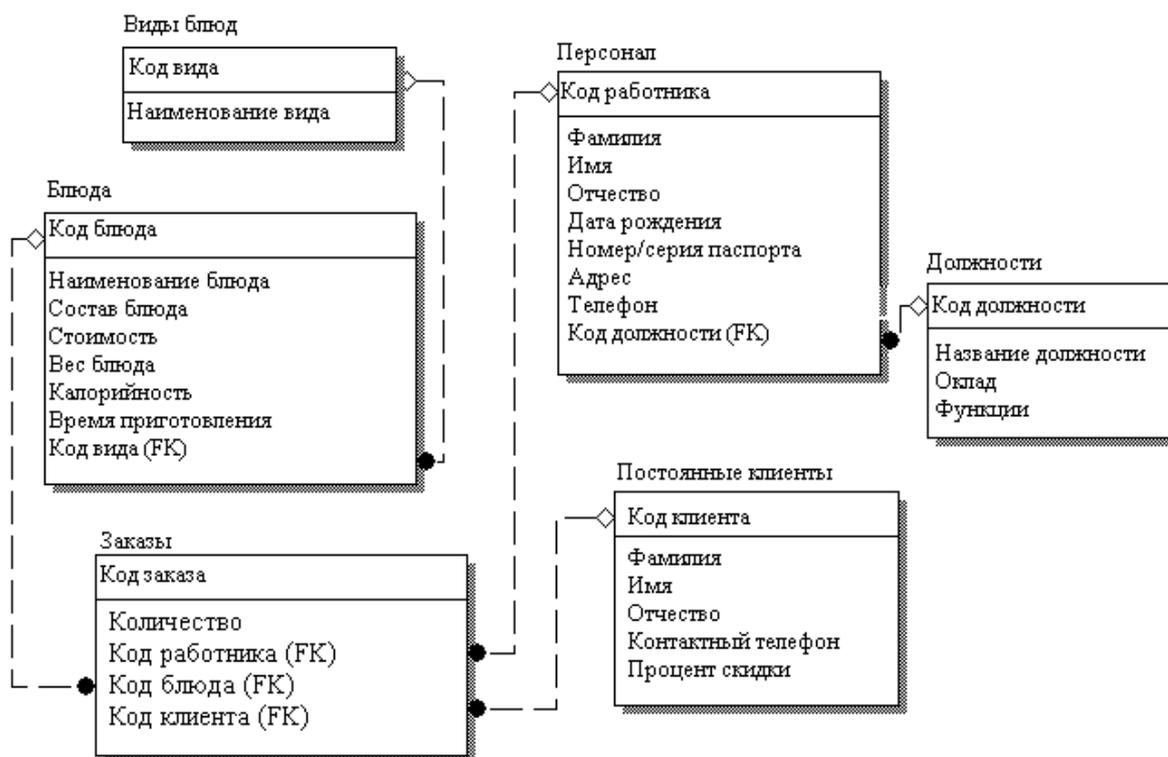


Рис. 1. Модель данных ресторана.

- ◆ Персонал.
- ◆ Должности.
- ◆ Постоянные клиенты.
- ◆ Заказы.

Модель строим на логическом уровне (см. рис. 1).

Из рисунка 1 видно, что в модели проставлены связи. Рассмотрим их подробнее (таблица 1).

7. Реляционная модель базы данных

Реляционная модель БД — модель логического уровня, представляющая собой отображение логических связей между элементами данных, независимо от их содержания и среды хранения.

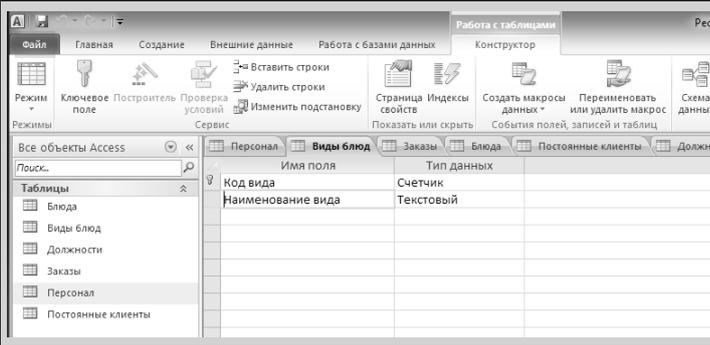
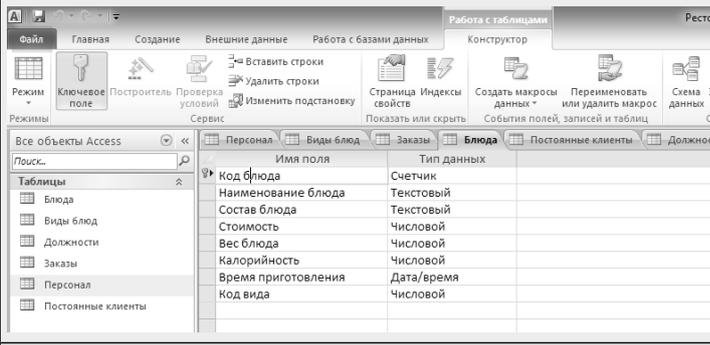
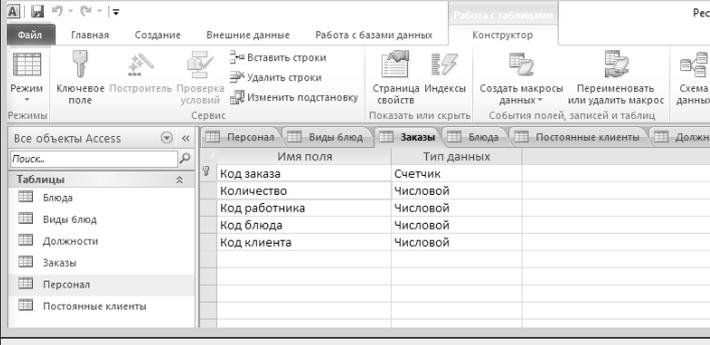
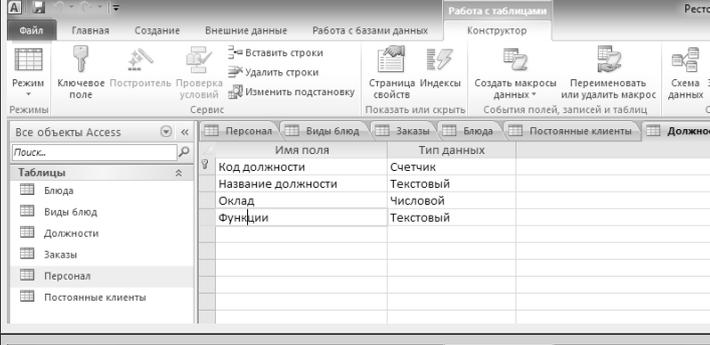
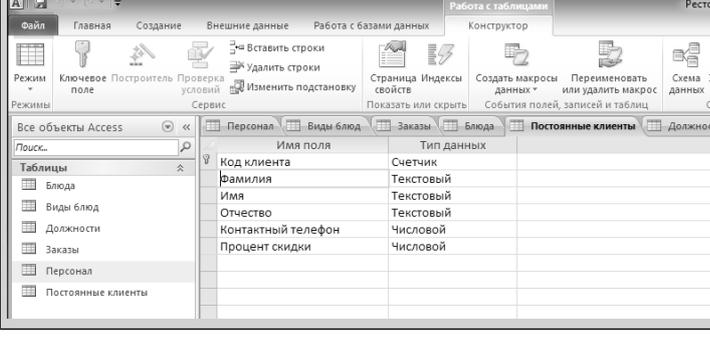
На этапе датологического проектирования строится логическая структура БД. При этом происходит преобразование исходной инфологической модели в модель данных, которая поддерживается конкретной СУБД. После этого производится проверка адекватности датологической модели, отображаемой предметной области. Конечным результатом проектирования является описание структуры БД на языке описания данных конкретных СУБД.

8. Физическое проектирование базы данных

Анализ сущностей концептуальной модели, их атрибутов и связей позволяет сделать вывод: каждая сущность может быть представлена отдельной таблицей, причем все они уже находятся в третьей нормальной форме. Таблицы базы данных создаются с помощью утилиты BDE Administrator в программе Database Desktop. Таблицы имеют тип Paradox 7. Создадим папку, где будут храниться созданные таблицы и пропишем путь к ней в BDE (Borland Database Engine) Administrator для создания алиаса. Структура таблиц проста, в ней создаются связи по первичному и по индексным полям между таблицами.

В таблице Menu представлен весь ассортимент блюд. Структура таблицы (поля, их типы и назначение) представлена на рисунке 2. Всем полям назначен атрибут Required Field (требование обязательного существования значения у поля на момент его запоминания в БД), кроме поля Kod_bluda, поскольку это поле автоинкрементное, заполнение его значением производится автоматически при запоминании новой записи. С помощью элемента Secondary Indexes задали необходимые индексы (Kod_vida, по данному полю будет

Таблица 1

<p>Таблица «Виды блюд» и таблица «Блюда» — установлена связь «один-ко-многим» при помощи первичного ключа «Код вида»</p>	
<p>Таблица «Должности» и таблица «Персонал» — установлена связь «один-ко-многим» при помощи первичного ключа «Код должности»</p>	
<p>Таблица «Блюда» и таблица «Заказы» — установлена связь «один-ко-многим» при помощи первичного ключа «Код блюда»</p>	
<p>Таблица «Персонал» и таблица «Заказы» — установлена связь «один-ко-многим» при помощи первичного ключа «Код работника»</p>	
<p>Таблица «Постоянные клиенты» и таблица «Заказы» — установлена связь «один-ко-многим» при помощи первичного ключа «Код клиента».</p>	

Field roster:				
	Field Name	Type	Size	Key
1	Kod_bluda	+		*
2	Naimen_bluda	A	30	
3	Soderganie	A	115	
4	Kalorinost	I		
5	Ves_bluda	I		
6	Vremya_prigot	A	20	
7	Stoimost_rub	I		
8	Kod_vida	I		

Table properties:

Validity Checks: [Dropdown]
 Define... [Button]
 1. Required Field
 2. Minimum value: [Text Box]
 3. Maximum value: [Text Box]
 4. Default value: [Text Box]

Рис. 2. Структура таблицы «Меню»

осуществлена фильтрация в таблице), а посредством элемента Refrential Integrity — ссылочную целостность, установив при этом возможность каскадных изменений и удалений.

В таблице Personal первичным ключом является поле Kod_rabotnika, а вторичный индекс присвоен полю Kod_dolgnosti (по нему происходит связь с таблицей Dolgnosti, а также будет осуществлена сортировка в таблице) и полю Familiya (по нему будет осуществлен поиск в таблице).

В таблице Dolgnosti первичным ключом является поле Kod_dolgnosti (по нему происходит связь с таблицей Personal).

В таблице Postoyanue_klientu первичным ключом является поле Kod_klienta (по нему будет осуществлен поиск в таблице).

В таблице Vidu_blud первичным ключом является поле Kod_vida (по нему происходит связь с таблицей Menu).

В таблице Zakazu первичным ключом является поле Nom_p/p, а вторичный индекс присвоен полю Kod_bluda (по нему происходит связь с таблицей Menu) и полю Kod_rabotnika (данное поле будет включено в отчет Rave).

9. Создание приложения для работы с базой данных. Разработка функциональной модели СУБД

Все описанные таблицы, составляющие основу базы данных, функционируют в рамках созданной си-

стемы управления базой данных "Restaurant". Приложение для работы с СУБД "Restaurant" создано средствами среды программирования Delphi 7.0. Модуль «Справочные таблицы» является основным модулем разрабатываемой программы. Его назначение — предоставление пользователю возможности выбора дальнейшего действия. Он позволяет загрузить другие модули, настроить СУБД согласно желаниям пользователя или осуществить выход из СУБД. Модули «Персонал», «Постоянные клиенты», «Ассортимент блюд», «Заказы» позволяют просматривать и редактировать соответствующие справочники, то есть осуществлять добавление, удаление, изменение и сохранение записей справочника. Из модуля «Ассортимент блюд» можно сформировать отчет (в виде меню), а из модуля «Заказы» формируется отчет в виде чека заказа. То есть это уже операционные таблицы Модуль «Файл» является справкой приложения. В модуле «О программе» можно просмотреть информацию о программном продукте. В модуле «Об авторе» содержится информация о разработчике приложения. Модуль «Выход» осуществляет завершение работы программы. Разработка экранных форм ввода-вывода и отчетов. При разработке СУБД "Restaurant" были разработаны экранные формы ввода/вывода исходных данных и результатов обработки, а также печатаемых отчётов. При запуске приложения мы видим, как процесс загрузки главной формы скрывает заставка. Она активна в течение 5 секунд. Затем открывается главное окно приложения.

Главное окно приложения содержит меню, с помощью которого можно начать работу. При выборе пункта меню Файл о программе открывается окно с информацией о программном продукте.

Управление отображением данных

С целью управления данными разрабатываем программу, которая будет работать с базой данных MS Access. Для разработки программы используем ADO:



— ADOConnection.

Создаем новый проект. На форме компонент ADOConnection с закладки ADO палитры компонентов, настраиваем соединение с сервером, которое прописано в свойстве ConnectionString.

Используем специальный файл (Use Data Link File); Используем строку подключения (Use Connection String).

На закладке Provider перечислены все доступные ADO драйверы доступа к базам данных.

Для доступа к базам данных MS Access используется драйвер «Microsoft Jet OLE DB Provider». Такой драйвер обязательно устанавливается на машину вместе с MS Office, а в последних версиях Windows он устанавливается по умолчанию.

После выбора базы данных, необходимо протестировать соединение (Test Connection).

В свойствах компонента ADOConnection необходимо отключить свойство LoginPrompt, выставив его в False для того, чтобы при каждом обращении к базе не загружалось окно ввода пароля.

Далее выставляем свойство Connected в True, чтобы произошло соединение с базой. Теперь необходимо получить доступ к созданной базе. Для этого находим форму компонент ADOTable в закладке ADO палитры компонентов



— ADOTable.

В данном компоненте свойство ConnectionString настраиваемое. Указываем у нашего компонента в свойстве Connection, созданный нами компонент соединения с базой данных.

В свойстве TableName нужно выбрать имя таблицы. После этого можно осуществлять подключение. Для этого выставляем свойство Active в true.— этот метод проверяет правильность закладки.

Для отображения данных из таблицы надо ещё установить на форму компонент DataSource



TDataSource с закладки Data Access палитры компонентов.

Далее отображаем данные. Устанавливаем компонент DBGrid



— DBGrid.

Это компонент-сетка, которая может отображать данные в виде таблицы. В этом же компоненте можно добавлять, удалять и редактировать строки таблицы.

И последний этап создания приложения — связывание компонента сетки с компонентом отображения таблицы. Для этого в свойстве DataSource компонента DBGrid указываем созданный нами компонент DataSource1.

Разработка формы

После размещения на форме компонент MainMenu категории Standard создаем меню программы. На главную форму устанавливаем компоненты: Image bitbtn с закладки additional.

Компонент отображает на форме графическое изображение и обеспечивает работу с ним. Основные свойства:

- ◆ содержит отображаемый графический объект;
- ◆ картинка изменяется пропорционально при изменении размеров;
- ◆ цвет фона изображения прозрачный, чтобы через него было видно нижележащее изображение;
- ◆ центрирование изображения в поле компонента, если его размеры меньше размеров поля. При значении False изображение располагается в верхнем левом углу поля. Свойство не действует, если AutoSize установлено в true или если Stretch установлено в true и Picture содержит не пиктограмму;
- ◆ отображает изменение автоматически размера компонента, подстраиваясь под размер изображения. По умолчанию False — не подстраивается.

Меню программы

Меню:

- ◆ мастер
- ◆ выход
- ◆ Отчеты

Документы:

- ◆ Прайс лист
- ◆ карточки клиентов
- ◆ карточки сотрудников

Действия:

- ◆ работа с клиентами
- ◆ редактировать сотрудников
- ◆ редактировать ассортимент
- ◆ редактировать предложение
- ◆ оформить покупку

Данная форма предоставляет возможность оформления и редактирования продаж товаров и услуг.

Создание отчетности в Quick Reports

Создания отчётов, которое входит в поставку Delphi — это Quick Reports. Следует отметить, что по скорости действия данный компонент не является достаточно быстрым в сети Internet, но его главное преимущество в том, что он установлен и готов к работе в Delphi.

Все компоненты Quick Reports находятся на закладке QReport палитры компонентов.

Головной компонент Quick Reports — TQuickRep. Этот компонент — основа любого отчёта. Он представляет собой холст листа будущего отчёта.

На форме (тест) установил компонент QuickRep. Выдели этот компонент и в объектном инспекторе включи параметры HasTitle и HasDetail свойства Bands.

Далее в этих секциях располагаем компоненты, которые будут отображать необходимую информацию. На закладке QReport палитры компонентов доступны следующие компоненты, которые можно располагать в этих разделах:

- ◆ надпись. Этот компонент похож на стандартный компонент TLabel и отображает нужные данные;
- ◆ данные. Этот компонент тоже похож на TLabel, только он предназначен для отображения значения какого-либо поля из базы данных.

Тип поля базы данных должен быть совместим с текстом, т.е. может быть целым числом, строкой, датой, но не может быть картинкой или бинарными данными;

- ◆ системная информация. Это опять копия TLabel только с возможностью отображать системную информацию;
- ◆ дату, время, номер страницы, номер строки в таблице, общее количество страниц и т.д.;
- ◆ набор строк. Этот компонент уже похож на TMemo и способен отображать Memo данные из базы данных;
- ◆ компонент для создания обрамлений.

Программный комплекс состоит из клиентской части, выполненной в среде программирования Delphi, и серверной части, выполненной в виде базы данных MS Access, хранимой на сервере и состоящей из таблиц, хранимых процедур и других компонентов базы данных.

ЛИТЕРАТУРА

1. 1С: Предприятие 8. Общепит. [Сайт]. — URL: <https://solutions.1c.ru/catalog/public-catering> (дата обращения: 27.02.2022).
2. Зуева А.Н. Бизнес-процессы: анализ, моделирование, управление [Текст]: учебное пособие / А.Н. Зуева. — М.: ПТУ МИРЭА, 2020. — 157 с.
3. Искусственный интеллект в ресторанном бизнесе. Основные направления и примеры. [Сайт] — URL: <https://evergreens.com.ua/ru/articles/ai-in-restaurants.html> (дата обращения: 25.05.2022).
4. К 2035 году мировой объем FoodNet может составить порядка 5,9 трлн. долларов. [Сайт] — URL: <http://biotech2030.ru/k-2035-godu-mirovoj-obem-foodnet-mozhet-sostavit-poryadka-5-9-trln-dollarov/> (дата обращения: 01.03.2022).
5. Как использовать Wallet на iPhone, iPod touch и Apple Watch. [Сайт] — URL: <https://support.apple.com/ru-ru/HT204003> (дата обращения: 20.05.2022).
6. Какой бизнес может увеличить прибыль с помощью цифровой экосистемы. [Сайт] — URL: <https://www.cossa.ru/special/ecosystems/271856/> (дата обращения: 21.05.2022).
7. Кисилевич Т.И. Автоматизация деятельности предприятий общественного питания: теория и опыт [Текст] / Т.И. Кисилевич, Я.Ю. Митрюшкин, Г.Р. Хвистани // Инновационное развитие экономики. — 2018. — № 6–1 (48). — С. 161–166.

© Гончаров Андрей Валерьевич (a.goncharov@mgutm.ru),

Яшин Денис Дмитриевич (d.yashin@mgutm.ru), Братцев Кирилл Евгеньевич (bk@fpv.ru).

Журнал «Современная наука: актуальные проблемы теории и практики»