

Приложение 9

Пример реализации системы электронной коммерции

Введение

Большую помощь при выполнении проекта с использованием Web-технологий может оказать анализ пяти полностью рабочих приложений-примеров от компании Microsoft под названием ASP.NET Starter Kits (www.asp.net/starterkits). Цель этих приложений - предложить разработчикам несколько полезных стартовых Web-сайтов и образцы хорошего ASP.NET-кода, пригодного для обучения и в качестве отправной точки. Все пять наборов поставляются с полным исходным кодом, работают с хранилищами данных SQL Server или Microsoft Data Engine (MSDE) и существуют в версиях на Visual Basic, C# и J#. В данном приложении будет кратко рассмотрено первое приложение IBuySpy.

Виртуальный магазин IBuySpy - полнофункциональное трехуровневое Web-приложение электронной коммерции. Оно объединяет покупательскую корзину, учетную запись пользователя и его регистрационное имя, высокопроизводительную систему кэширования данных и транзакции. Web-приложение содержит Web-службы для подачи заказов, и извлечения информации о состоянии. Чтобы установить и настроить приложение на локальном компьютере, где будет возможность модифицировать код и применять средства отладки, можно обратиться по адресу <http://www.IBuySpy.com/>. Программа загрузки запустит сценарий базы данных для SQL Server и создаст виртуальный каталог из папки, в который было установлено приложение IBuySpy. После этого можно запустить установленное приложение, запросив стартовую страницу <http://localhost/StoreVBVS/Default.aspx> или открыть приложение в Visual Studio .NET. Ниже будет дано более детальное описание магазина IBuySpy.

Описание виртуального магазина IBuySpy

Виртуальный магазин IBuySpy представляет собой пример использования одного из самых распространенных типов Web-приложений – сайтов электронной коммерции. Легенда состоит в том, что он принадлежит фиктивной компании, занимающейся продажей различного рода высокотехнологичных шпионских штучек. Сайт отвечает всем требованиям своего жанра: включает каталог продукции с возможностью поиска, список товаров, имеющих наибольший спрос, отзывы пользователей и покупательскую корзину. Архитектура приложения состоит из четырех уровней: уровня данных, хранимых процедур, бизнес-компонентов и Web-страниц.

Уровень данных

Приложение IBuySpy использует семь таблиц базы данных под названием Store, которые условно можно разделить на две части. Назначение первой – хранить информацию о покупателях и сделанных ими заказах. Назначение второй – хранить информацию о продаваемых товарах.

Каждая запись с информацией о покупателе (таблица Customers) может содержать ссылки на неограниченное количество дочерних записей и сделанных им заказах. Информация о заказах, в свою очередь, разделена на две таблицы – Orders и OrderDetails. Этот прием повышает производительность в случаях, когда для выполнения запроса требуется извлечь информацию только из одной таблицы.

Каждый товар (таблица Products) должен быть помещен в одну из категорий (к примеру, General, Tools или Travel), описанных в таблице Categories. Кроме того, для каждого товара может существовать неограниченное количество отзывов в таблице Reviews.

Таблица ShoppingCart содержит выбранные, но еще не заказанные пользователем товары. Эту информацию можно было поместить в состояние сеанса (session state), но использование таблицы базы данных предполагает большую гибкость и к тому же позволяет сохранять информацию до следующих посещений вместо того, чтобы уничтожать ее по истечении заданного промежутка времени. Чтобы узнать, какие товары находятся в определенной корзине, необходимо найти все записи с одинаковым идентификатором CartID.

Стоит отметить следующие особенности структуры базы данных.

- Пользовательские пароли хранятся в базе данных в незашифрованном виде. Исходя из этого, необходимо обеспечить хранение этой информации в тщательно защищенной базе данных SQL Server, недоступной для большинства пользователей.
- Изображения товаров хранятся в виде имен файлов (например, image.gif). Приложение может использовать это имя в качестве относительной ссылки, помещенной в элемент управления "изображение" (image control). Однако этот файл должен присутствовать в каталоге приложения на Web-сервере. В случае его отсутствия проверка ошибок выполнена не будет, и пользователь не будет предупрежден об этом. Альтернативой может быть хранение изображений в виде двоичных данных в самой базе данных. Этот подход может решить проблему отсутствующих изображений, но он понизит эффективность и сделает эти изображения недоступными для статичных страниц HTML. При запуске приложения IBuySpy локально, для всех товаров используется одно и то же шаблонное изображение. Если же запустить приложение в сети (или самостоятельно модифицировать таблицу Products), для каждого товара будет отображаться индивидуальное изображение.

- В таблице Customers записи о покупателях не привязаны к пользовательским отзывам. Это позволяет даже не сделавшим покупки клиентам добавлять свои комментарии. На многих Web-сайтах (например, на Amazon.com) публиковать отзывы могут только известные пользователи.
- Товары в покупательской корзине не привязаны к каким-либо записям о пользователях. При использовании этого подхода пользователь может начинать заполнять корзину до регистрации на сайте. Многие пользователи находят другие схемы (в том числе, с обязательной регистрацией перед покупкой) чрезвычайно неудобными.

Хранимые процедуры

Хранимые процедуры почти всегда являются наиболее эффективным способом извлечения информации из базы данных и изменения ее содержания. В приложении IBuySpru доступ к данным осуществляется с помощью 23 отдельных хранимых процедур. В качестве примера в таблице приведено описание хранимых процедур, связанных с обработкой корзины покупателя.

Хранимая процедура	Описание
ShoppingCartAddItem	Добавляет определенный товар в определенную покупательскую корзину, помещая новую запись в таблицу ShoppingCart. Если этот товар в ней уже есть, обновляется запись в поле Quantity.
ShoppingCartEmpty	Удаляет все записи с указанным идентификатором покупательской корзины.
ShoppingCartItemCount	Возвращает число элементов в определенной покупательской корзине.
ShoppingCartList	Возвращает список элементов в определенной покупательской корзине.
ShoppingCartMigrate	Перемещает данные из временной покупательской корзины (создаваемой для пользователей, не имеющих учетной записи или не прошедших процедуру регистрации) в окончательную учетную запись зарегистрированного пользователя. Это происходит непосредственно перед покупкой.
ShoppingCartRemoveAbandoned	Удаляет покупательские корзины, существующие в системе дольше одного дня.
ShoppingCartRemoveItem	Удаляет определенный товар из определенной покупательской корзины.
ShoppingCartTotal	Рассчитывает общую стоимость всех элементов, находящихся в определенной покупательской корзине.
ShoppingCartUpdate	Изменяет поле Quantity определенного товара и покупательской корзины.

Бизнес-компоненты

Web-страницы приложения IBuySpru никогда не осуществляют прямой доступ к базе данных. Они работают при посредничестве пяти компонентов (CustomersDB, OrdersDB, ProductsDB, ReviewDB и ShoppingCartDB), инкапсулируя весь код доступа к базе данных. Такая технология позволяет вносить изменения в код доступа к базе данных, не затрагивая кода страницы, а также предоставляет более логичный интерфейс. Чтобы просмотреть методы, содержащиеся в том или ином классе компонента приложения, можно использовать окно Solution Explorer IDE Visual Studio .NET.

Например, компонент CustomersDB содержит два класса: обслуживающий класс CustomersDB и класс данных CustomerDetails, который отвечает за информацию в одиночной записи о покупателе. Класс CustomersDB содержит три метода: AddCustomer, GetCustomerDetails и Login.

Метод AddCustomer принимает имя, адрес электронной почты и пароль нового пользователя и создает соответствующую запись в таблице Customers с помощью хранимой процедуры CustomerAdd. Чтобы гарантировать успешное завершение этого процесса, уникальный идентификатор покупателя (генерируемый в базе данных) возвращается в виде строки. Чтобы применить этот компонент, клиент должен создать экземпляр класса CustomersDB и предоставить соответствующую информацию. Сказанное выше можно обнаружить на странице register.aspx. Метод GetCustomerDetails, возвращает информацию в виде объекта CustomerDetails. Метод Login следует подобной логике, но запускает исключение, если соответствующая запись не найдена.

Все остальные компоненты выполняют схожие действия – сначала создают необходимые объекты параметров, затем вызывают хранимую процедуру и, наконец, извлекают информацию из выходных параметров, чтобы впоследствии, если потребуется, использовать ее в качестве возвращаемого значения.

Web-страницы

Виртуальный магазин IBuySpry состоит из тринадцати страниц aspx, пяти пользовательских элементов управления ascx и одной Web-службы asmx. Так как в этом приложении значительная роль принадлежит бизнес-компонентам и хранимым процедурам базы данных, код Web-страниц лаконичен, прост и понятен.