

МЕТОД ПОБУДОВИ МЕРЕЖІ ВІТРИН ІНТЕРНЕТ-МАГАЗИНІВ НА ОСНОВІ АРХІТЕКТУРИ MVC *

© 2017 ПУРСЬКИЙ О. І., МАЗОХА Д. П.

УДК 339.1;330.4;658.6

Пурський О. І., Мазоха Д. П. Метод побудови мережі вітрин інтернет-магазинів на основі архітектури MVC

Розроблено метод побудови мережі вітрин інтернет-магазинів, що забезпечує підтримку великої кількості замовлень і відвідувань покупців з урахуванням відповідності сучасним стандартам продуктивності та надійності інтернет-рішень у сфері електронної торгівлі. Метод передбачає створення типової вітрини та реалізацію для інформаційної системи управління мережею вітрин власної бази даних на основі даних центральної управляючої інформаційної системи та двосторонньої реплікації даних. Запропоновано механізм «клонування» інтернет-вітрин, що входять в мережу, та їх прискореної інтеграції з бізнес-процесами підприємства та системою управління на базі типової вітрини. Розробку типових вітрин інтернет-магазинів здійснено на основі концепції MVC (Model – View – Controller), технології ASP.NET MVC Framework і візуальних шаблонів Web-сторінок, що дозволяє забезпечити незалежність алгоритмів поведінки об'єктів від самих об'єктів та їх візуального представлення. Це значно підвищує ефективність розробки інтернет-проектів електронної торгівлі, прискорює процес реалізації та забезпечує високу гнучкість і функціональність вітрин інтернет-магазинів.

Ключові слова: електронна торгівля, мережа інтернет-вітрин, технологія ASP.NET MVC Framework, концепція MVC.

Рис.: 1. **Бібл.:** 19.

Пурський Олег Іванович – доктор фізико-математичних наук, професор кафедри економічної кібернетики, Київський національний торговельно-економічний університет (вул. Кіото, 19, Київ, 02156, Україна)

E-mail: Pursky_O@ukr.net

Мазоха Дмитро Павлович – аспірант кафедри економічної кібернетики, Київський національний торговельно-економічний університет (вул. Кіото, 19, Київ, 02156, Україна)

УДК 339.1;330.4;658.6

UDC 339.1;330.4;658.6

Пурський О. И., Мазоха Д. П. Метод построения сети витрин интернет-магазинов на основе архитектуры MVC

Разработан метод построения сети витрин интернет-магазинов, который обеспечивает поддержку большого количества заказов и посещений покупателей с учетом соответствия современным стандартам производительности и надежности интернет-решений в области электронной торговли. Метод предусматривает создание типичной витрины и реализацию для информационной системы управления сетью витрин собственной базы данных на основе данных центральной управляющей информационной системы и двусторонней репликации данных. Предложен механизм «клонирования» интернет-витрин, которые входят в сеть, и их ускоренной интеграции с бизнес-процессами предприятия и системой управления на базе типичной витрины. Разработка типичных витрин интернет-магазинов осуществлена на основе концепции MVC (Model – View – Controller), технологии ASP.NET MVC Framework и визуальных шаблонов Web-страниц, что позволило обеспечить независимость алгоритмов поведения объектов от самих объектов и их визуального представления. Это значительно повышает эффективность разработки интернет-проектов электронной торговли, ускоряет процесс реализации и обеспечивает высокую гибкость и функциональность витрин интернет-магазинов.

Ключевые слова: электронная торговля, сеть интернет-витрин, технология ASP.NET MVC Framework, концепция MVC.

Рис.: 1. **Библ.:** 19.

Пурський Олег Іванович – доктор фізико-математических наук, профессор кафедры экономической кибернетики, Киевский национальный торгово-экономический университет (ул. Киото, 19, Киев, 02156, Украина)

E-mail: Pursky_O@ukr.net

Мазоха Дмитрий Павлович – аспирант кафедры экономической кибернетики, Киевский национальный торгово-экономический университет (ул. Киото, 19, Киев, 02156, Украина)

Pursky O. I., Mazoha D. P. The Method of Building a Network of Online Showcases on the Basis of the MVC Architecture

A method to build a network of online showcases that support a large number of customer orders and visits, which meets the current performance standards and the reliability of Internet solutions in the sphere electronic commerce, has been developed. The method involves the creation of a typical showcase and the implementation for the information management system of a showcases network of an own database operating on the data from the central management information system with the two-way data replication. A mechanism for «cloning» the online showcases, which are part of the network, and their quick integration with the business processes of enterprise and a management system based on a typical showcase, has been proposed. The development of typical online showcases is implemented on the basis of MVC concept (Model-View-Controller), the ASP.NET MVC Framework Technology, and the visual templates of web pages, thus ensuring that the algorithms for the behavior of objects are independent of both the objects themselves and their visual representation. This enhances the development of e-commerce projects significantly, speeds up the implementation process, and provides a high degree of flexibility and functionality of the online showcases.

Keywords: e-commerce, network of online showcases, ASP.NET MVC Framework Technology, MVC concept.

Fig.: 1. **Bibl.:** 19.

Pursky Oleg I. – D. Sc. (Physics and Mathematics), Professor of the Department of Economic Cybernetics, Kyiv National University of Trade and Economics (19 Kioto Str., Kyiv, 02156, Ukraine)

E-mail: Pursky_O@ukr.net

Mazoha Dmytro P. – Postgraduate Student of the Department of Economic Cybernetics, Kyiv National University of Trade and Economics (19 Kioto Str., Kyiv, 02156, Ukraine)

Електронна торгівля сприяє не тільки глобалізації та прискоренню інформаційного обміну, але також є однією з головних світових тенденцій економічного розвитку. Економічний ефект від викори-

стання технологій електронної торгівлі є надзвичайно позитивним, і на сьогодні даний напрямок економічної діяльності став невід'ємною частиною суспільного життя. Електронна торгівля, як суб'єкт торговельного ринку, приводить до загального зниження цін на товари і послуги, що сприяє подальшому збільшенню обсягів електронної торгівлі.

* Дослідження здійснено за фінансової підтримки Міністерства освіти і науки України, проект № 0117U000507, тема «Модельовання механізмів функціонування міжнародної електронної торгівлі».

Процеси торговельної глобалізації спричиняють створення великими операторами ринку мультибрендових мереж магазинів і гіпермаркетів з метою збільшення обсягів торговельних трансакцій і зниження витрат за рахунок спільного використання загальних ресурсів. Згадані тенденції знаходять своє відображення і в електронній торгівлі, викликаючи розширення підприємств електронної торгівлі, появу мереж інтернет-магазинів (ІМ) і гіпермаркетів, що дозволяє нарощувати обсяги торговельних операцій і диференціювати торговельні стратегії на основі мережі вітрин ІМ.

Дослідженням механізмів електронної торгівлі займалася значна кількість вітчизняних і закордонних науковців, зокрема: Геєць В. М. [1], Вітлінський В. В. [2], Базилевич В. Д. [3], Мазаракі А. А. [4], Порохня В. М. [5], Ситник В. Ф. [6], Хейг М. [7], Копитко Б. І. [8], Дюбанов О. С. [9], Лаудон К. С. [10], Козьє Д. [11], Хартман А. [12]. Водночас завдання впровадження ефективних, інтегрованих у Web-середовище інформаційних систем (ІС) управління електронною торгівлею залишається на сьогодні однією з найбільш актуальних і складних проблем для великих підприємств електронної торгівлі. Відомі теоретичні та практичні програмні рішення для малого і середнього бізнесу не відповідають вимогам великих компаній щодо можливостей створення інтернет-вітрин, їх функціям і інструментарію розвитку, а також до рівня інтеграції Web-середовища і системи управління підприємством електронної торгівлі [13]. Має місце недостатнє використання можливостей електронних закупівель, оптимізації та систематизації логістичних процесів для автоматизації формування торговельного асортименту і логістики поставок товарів за рахунок інтеграції інформаційних систем підприємств електронної торгівлі і оптових інтернет-майданчиків. Ця проблема має особливе значення для управління товарними потоками мережі вітрин ІМ з асортиментом у сотні тисяч позицій.

Головна *мета* статті полягає в розробці методу побудови мережі взаємопов'язаних вітрин великих ІМ на основі концепції MVC (*Model – View – Controller*: Модель – Представлення – Контролер).

Розробка методу побудови мережі вітрин ІМ здійснювалася із урахуванням відповідності до вимог з функціональності, продуктивності та надійності Web-додатків [14] і передбачає такі етапи:

1. Розробка ядра Web-додатків контура управління вітринами (УВ) [15] на основі архітектури MVC.
2. Розробка MVC-патерна та реалізація в його рамках типової вітрини ІМ, яка є основою для бізнес-логіки для Web-додатків всієї мережі вітрин ІМ.
3. Використання шаблонів Web-сторінок для реалізації графічного дизайну і реалізація на основі шаблонів рівня Web-додатків за допомогою технології ASP.NET MVC Framework [16] (технологія для створення Web-додатків, яка генерує шаблон *Model – View – Controller*).
4. Реалізація функцій і додатків для СУБД, що абстрагують дані від моделей патерна MVC таким чином, щоб перенести логіку обробки даних на рівень СУБД.
5. Реалізація для СУБД контурів УВ і управління підприємством (УП) [15] двосторонньої асинхронної

реплікації з метою забезпечення цілісності інформаційної структури, даних і алгоритмів їх обробки.

Головне завдання, яке потребує вирішення при розробці архітектури ІС УВ, – це проблема організації взаємодії між логічним рівнем додатків, рівнем інтерфейсів, рівнем бізнес-логіки та рівнем збереження даних. Якщо всі основні сутності, представлені такими об'єктами, як «кошик», «каталог», «товар» і т. ін., будуть унікальними, а алгоритми їх функціонування незалежними від вітрини, де вони застосовуються, та зовнішнього представлення, яке повинне їм відповідати, то можна на їхній основі створити програмне ядро функцій контуру УВ. Для цього необхідно реалізувати три основні рівні: базовий рівень функцій, рівень сценаріїв та рівень адаптерів.

Принцип уніфікації та абстрагування основних алгоритмічних об'єктів у складних системах не є новим, тому достатньо використовувати сучасні підходи і технології, орієнтовані на вирішення подібного класу задач. Для розробки ІС управління мережею вітрин у представленому методі пропонується використовувати архітектурну концепцію MVC [17] (*рис. 1*). Як відзначалося раніше, вітрина ІМ включає велику кількість різних об'єктів («товар», «кошик», «каталог» і т. п.), які відображають набір сутностей предметної області. Важливою особливістю таких об'єктів є їхня взаємодія і вплив один на одного, на користувача, а також вплив користувача на них. У результаті формується замкнена система, зв'язки в якій вибудовуються на основі зовнішніх впливів користувачів, з одного боку, і процесів контуру УП, з іншого.

Призначення більшості ІС у середовищі Інтернет полягає в отриманні даних з деякої СУБД і надання їх користувачеві. Після виконання користувачем певних дій система зберігає дані або модифікує їх. Через те, що основний обмін даними відбувається між сховищем даних та інтерфейсом користувача, ці функції часто поєднують, при цьому, як правило, підвищується загальна продуктивність системи і зменшується загальний об'єм програмного коду. Але досить часто виникає необхідність незалежної модифікації користувальницького інтерфейсу або бізнес-логіки. Подальше ускладнення Web-додатка вимагає створення складної об'єктно-орієнтованої моделі та постійної її модифікації.

Класичне вирішення задачі було запропоновано програмістами модулів для підтримки графічного інтерфейсу на мові Smalltalk у вигляді патерна проектування Model-View-Controller [17], який прийнятий за основу графічного інтерфейсу додатків розроблювальної системи. Розглянемо організацію елементів MVC з огляду на задачі, що вирішуються в електронній торгівлі.

Модель (*Model*) містить у собі функціональну логіку додатка, тобто рівень бізнес-логіки. Спосіб реалізації даного рівня залежить від обраної інженерної технології реалізації. Основна мета застосування даного патерна (MVC) – зробити інформаційну модель повністю незалежною. Це дозволить змінювати і контролер, і представлення моделі, не змінюючи саму модель, а також дозволить забезпечити функціонування декількох

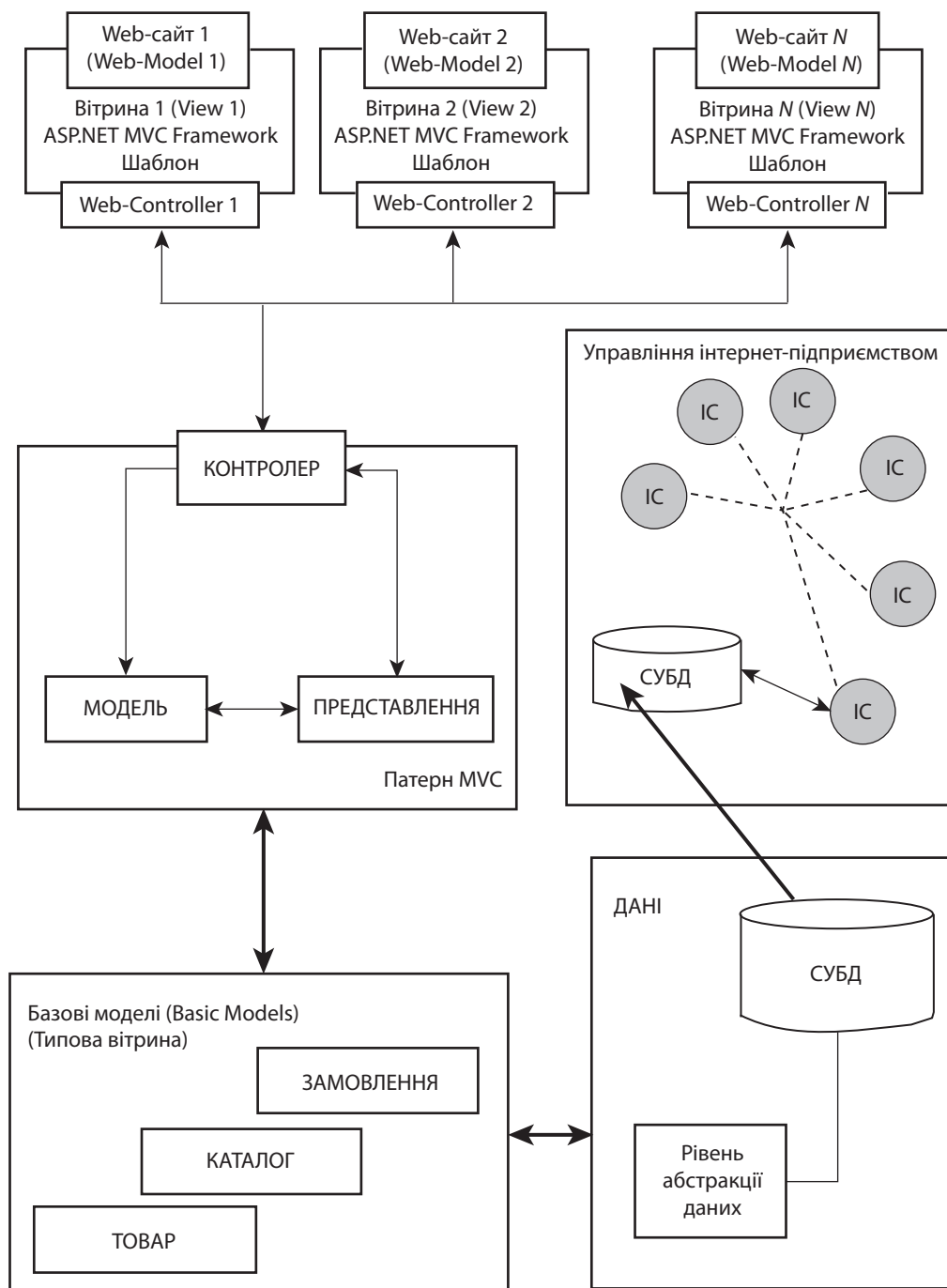


Рис. 1. Модель структури мережі вітрин ІМ на основі архітектури MVC

варіантів представлення і контролерів для однієї моделі одночасно (див. рис. 1). Таким чином, з'являється можливість різної обробки моделі на різних вітринах ІМ. Перевага очевидна – при зміні моделі автоматично перебудовуються всі пов'язані з нею об'єкти на всіх вітринах ІМ. Важливо, що модель при жодних умовах не може містити посилань на об'єкти представлення або контролера. В іншому випадку утворюється прямий зв'язок між незалежними об'єктами на різних вітринах, і зміна бізнес-логіки на одній із вітрин спричинить відповідні зміни на незалежній від неї, іншій вітрині.

До функцій представлення (View) входить відображення даних, отриманих від моделі. Зазвичай представлення має вільний доступ до моделі та може отримувати

від неї дані, але це доступ тільки на зчитування даних. Представлення нічого не змінює в моделі та не може ініціювати методи, що приводять до зміни її внутрішнього стану. У випадку активної моделі представлення може зчитувати події зміни моделі та змінювати формат представлення, зчитавши змінені дані при отриманні відповідного оповіщення. Стосовно прикладної задачі побудови вітрин ІМ, варто звернути увагу на зручність зміни представлення моделі на всіх вітринах ІМ (див. рис. 1). Припустимо, що модель «кошик» через зміни була модифікована і тепер повинна відображати не тільки назву і вартість товару, але й вартість його доставки. Модифікація моделі «кошик» у базовому патерні спричинить наслідкові зміни на всіх вітринах ІМ (див. рис. 1). Модифі-

фікація представлення моделі в базовому патерні надає можливості відображення нового поля даних в інтерфейсі як окремої вітрини, так і всіх вітрин ІМ.

Для взаємодії з контролером (*Controller*) представлення, як правило, реалізує певний інтерфейс, відомий контролеру, що дозволяє змінювати представлення незалежно і мати декілька представлень на контролер. Таким чином, якщо окремі вітрини ІМ вимагають особливої обробки реалізацій моделі «кошик» на сайті, то вона може бути закладена в окреме представлення. Як приклад, варто навести різні способи переміщення товарів у кошик. По одному контролеру (посиланню на сайті «відкласти в кошик») можуть спрацьовувати різні представлення кошика: коротке – у вигляді спливаючого вікна, розширене – у вигляді повноцінно відображеного кошика.

Підміна або зміна представлення – це задача, яку найчастіше доводиться вирішувати при створенні різних режимів відображення об'єктів. По суті це і є та причина, по якій створюються різні патерни розподілу моделі та представлення. У задачі контролера входить реакція на зовнішні події та зміна моделі і/або представлення відповідно до закладеної в нього логіки. Один контролер може працювати з декількома представленнями, залежно від ситуації, взаємодіючи з ними через деякий задалегідь відомий інтерфейс, який ці представлення реалізують. Такий підхід дозволяє використовувати переваги патерна MVC для проектування і розробки мережі вітрин ІМ, які полягають у тому, що новий режим відображення для всіх або для окремих вітрин ІМ може бути внесений в контролер і буде використаний вітринами незалежно від структури і логіки їх інтерфейсу та його візуалізації.

Таким чином, шаблон MVC дозволяє розділити бізнес-логіку, представлення і обробку дій користувача на три частини:

- ✦ контролер приймає та інтерпретує вхідні дані, а потім інформує модель і представлення про необхідність відповідної реакції;
- ✦ модель обробляє отримані дані відповідно до задач, які повинен виконати додаток, надає дані (для представлення), а також реагує на запити зміни свого стану (від контролера);
- ✦ представлення виводить на екран результат обробки даних.

Важливо відзначити, що як представлення, так і контролер залежать від моделі. Але модель не залежить ні від представлення, ні від контролера. Це одна із ключових переваг подібного розподілу, яка дозволяє побудувати модель незалежно від візуалізації представлення (див. рис. 1).

Додатково до розглянутих особливостей і переваг архітектури MVC стосовно задачі побудови мережі вітрин ІМ варто звернути увагу на можливість централізації управління такими прикладними компонентами, як забезпечення безпеки, авторизація, робота з даними та моніторинг станів. Архітектура MVC забезпечує виявлення й усунення можливих помилок, а також сприяє підключенню до Web-додатка нових джерел даних, нових типів клієнтів, чітко визначаючи при цьому обов'язки

класів. У цілому, вона максимально сприяє успішному супроводу Web-додатка протягом усього життєвого циклу, що вкрай важливо для мережі вітрин ІМ.

Безсумнівною перевагою архітектури MVC є також можливість стандартизації модулів контролера, моделей і представлень у рамках розв'язуваної задачі, забезпечуючи в такий спосіб побудову базового патерна розробки. На основі таких стандартних модулів можна реалізувати типову вітрину ІМ. Типова вітрина повинна включати всі базові функції, відповідно до прийнятої бізнес-логіки електронної торгівлі. Створення шаблону (див. рис. 1), основою якого є типова вітрина ІМ, забезпечує підвищення якості та швидкості розробки мережі вітрин, відповідність усіх нових вітрин ІМ єдиним стандартам функціональності, продуктивності та надійності, що прийняті для проекрованої інтегрованої торговельної інформаційної системи (ТІС), а також сприяє підвищенню ефективності підтримки і механізмів розвитку мережі вітрин. Для настільних додатків шаблон проектування MVC став стандартом. Об'єктно-орієнтовані методи набувають все більшого значення при побудові інтернет-систем, оскільки Web-додатки стають складнішими та інтегруються з комплексними ІТ-рішеннями.

Всесвітню мережу Інтернет характеризує різноманіття Web-ресурсів, розробники яких прагнуть надати їм унікальний графічний інтерфейс. Ресурси електронної торгівлі, такі як вітрини ІМ – не виключення. Більше того, до них висуваються підвищені вимоги по дизайну Web-сторінок з таких причин. Насамперед, Web-сторінки таких ресурсів постійно змінюються, покращуються, доповнюються спеціальними графічними елементами та модифікуються як графічно, так і інформаційно. Окрім того, часто потрібна наявність функції зміни зовнішнього вигляду Web-сторінок залежно від поведінки користувача, активованого об'єкта, його конкретної реалізації в умовах конкретної вітрини ІМ.

Для вирішення цієї задачі досить використовувати шаблони Web-сторінок. Шаблон – це HTML-документ, який містить спеціалізовані теги, що і дозволяє на етапі виконання програми перетворювати базову візуалізацію по заданому об'єктом алгоритму в необхідний зовнішній вигляд на основі цих тегів. Шаблон може містити також підпрограми перетворення дизайну сторінок і об'єктів на основі оперативних впливів користувача. Алгоритм зміни задається об'єктами на основі шаблонів візуалізації реакцій на дії користувачів. Таким чином, алгоритм самого об'єкта і представлення його даних не змінюється при необхідності змінювати візуалізацію Web-сайту. Завдяки використанню шаблонів можна змінювати зовнішній вигляд і представлення, залишаючи алгоритми недоторканими, що дозволяє реалізовувати мережу вітрин на основі загальних об'єктів і сукупностей даних. Саме в шаблоні Web-сторінки можливо реалізувати контролери об'єктів на основі платформи ASP.NET MVC Framework [16], надавши Web-сторінці незалежність від ядра контуру. За зворотний зв'язок на JSON (*Javascript Object Notation*), запити на сервері відповідають контролери ядра.

Вітринам ІМ не потрібен доступ до надлишкових даних прикладних ІС контура УП. Необхідні оперативні дані в агрегованому представленні містяться в центральній ІС контура УП, безпосередньо в її БД [15]. Для забезпечення цілісності даних на рівні контура УВ необхідно створити «рівень» абстракції даних (див. рис. 1), під яким розуміється набір функцій і алгоритмів, що забезпечують роботу СУБД шляхом абстрагування самих даних від реалізації процедур їх зберігання. Тобто, запис у СУБД, що визначає конкретний товар, зчитується із СУБД алгоритмом такого «рівня» і передається в структурованому вигляді інформаційної моделі «об'єкта» базового патерна MVC (див. рис. 1). Додатками такого рівня повинні бути функції та збережені процедури самої СУБД. Моделі базового патерна MVC не повинні мати можливості взаємодіяти з полями і структурами самої бази даних.

Стосовно вимоги до функціонування контуру УВ і контуру УП [15] в умовах повної відсутності зв'язку, то вона означає, що мережа вітрин ІМ і ІС контуру УП повинні бути забезпечені незалежними копіями даних. При цьому дублювання даних неприпустимо з погляду теорії нормалізації баз даних через можливе порушення їх цілісності та відповідних негативних наслідків. Імовірність серйозної помилки при цьому тим вища, чим більший об'єм даних і частота їх модифікації. Для ТІС класу В2С імовірність виникнення помилок через порушення цілісності та дублювання даних надзвичайно висока. Тому необхідно забезпечити синхронність даних за допомогою технології двосторонньої реплікації даних між СУБД контурів. Потрібно звернути увагу не тільки на реплікацію даних, але й на реплікацію пакетів функцій і додатків СУБД. Ґрунтуючись на вимозі підтримки розподіленості механізмів обробки інформації при низькій якості зв'язку, потрібно внести в механізм реплікації підтримку архівації та механізму відкладеного проведення процедур синхронізації, а також механізму автоматичного відновлення зв'язку після обриву з'єднання. Реалізацію згаданих вимог і підтримку взаємозв'язку інтерактивних елементів ТІС [13] із СУБД забезпечує використання при розробці сучасної технології ADO.NET Entity Framework [18; 19] – об'єктно-орієнтованої технології доступу до даних.

ВИСНОВКИ

Застосування багаторівневої архітектури на основі патерна MVC, безсумнівно, ускладнює Web-додаток, але водночас забезпечує такі його якості, як зручність розробки, тестування, легкість подальшої модифікації, структурованість програмного коду, абстрагованість від інформаційних моделей і рівнів даних. Архітектура MVC створює умови для комплексної розробки додатків, дозволяючи розробникам зосередитися на виконанні функцій, що вимагають глибокої спеціалізації, забезпечуючи можливість високого рівня декомпозиції проекту. Прискорення розробки і зручність підтримки за рахунок використання типових функцій вітрин, стандартизованих у рамках проекту інформаційних моделей і множини різноманітних контролерів вітрин ІМ (див. рис. 1) є також важливим фактором доцільності застосування архітектури MVC для задач розробки Web-

систем електронної торгівлі. Таким чином, усі вітрини реалізують унікальний графічний дизайн, орієнтований на різні групи покупців, і залишають логіку сторінок, об'єктів і саму архітектуру стабільною і незалежною в цілому. Побудова вітрин на основі технологій MVC [17] і ASP.NET MVC Framework [16] у комбінації з технологіями шаблонізації Web-сторінок дозволяє забезпечити незалежність алгоритмів поведінки об'єктів від самих об'єктів, їх зовнішнього вигляду і поведінки інтерфейсів об'єктів та сайтів при використанні на тій або іншій вітрині ІМ, а також дозволяє отримати високу гнучкість і функціональність мережі вітрин ІМ при значних переважах їхньої реалізації та підтримки для розробників. ■

ЛІТЕРАТУРА

1. Економіка України: стратегія і політика довгострокового розвитку/за ред. акад. НАН України В. М. Гейця. Київ: Інститут економічного прогнозування, 2003. 1008 с.
2. Вітлінський В. В., Мельник Г. В., Скіцько В. І. Моделювання збутової електронної логістичної підсистеми з використанням нечіткої мережі Петрі. *Бізнес Інформ*. 2014. № 8. С. 82–87.
3. Базилевич В. Д. Формування ринку електронної комерції в Україні. *Вісник Інституту економічного прогнозування*. 2002. № 2. С. 32–36.
4. Внутрішня торгівля в Україні: економічні умови ефективного розвитку: монографія/А. А. Мазараки, І. О. Бланк, Л. О. Лігоненко, Н. М. Гуляєва та ін. Київ: Київ. нац. торг.-екон. ун-т, 2006. 195 с.
5. Порожня В. М. Моделювання економіки. Запоріжжя: Запорізь. держ. інж. акад. 2001. 382 с.
6. Ситник В. Ф., Козак І. А. Телекомунікації в бізнесі: навч.-метод. посіб. Київ: КНЕУ, 1999. 204 с.
7. Хейг М. Основы электронного бизнеса/пер. с англ. С. Косихина. М.: ФАИР-ПРЕСС, 2002. 208 с.
8. Копитко Б. І., Юрчук Г. В. Поняття електронного бізнесу та класифікація його основних бізнес-моделей. Соціально-економічні дослідження в перехідний період. *Механізми регулювання регіонального ринку праці*. 2002. Вип. 3. С. 349–364.
9. Дюбанов О. С., Рамазанов С. К. Моделі поведінки суб'єктів ринку електронної торгівлі в сучасних умовах. *Бізнес Інформ*. 2011. № 6. С. 104–105.
10. Laudon, K. C., Traver, C. G. E-commerce: business, technology, society. USA: Addison Wesley, 2002. 762 p.
11. Козье Д. Электронная коммерция. М.: ИТД «Русская редакция», 1999. 228 с.
12. Хартман А. Стратегии успеха в Интернет-экономике. М.: ЛОРИ, 2001. 272 с.
13. Мазоха Д. П. Програмно-апаратні рішення для інтегрованих інформаційних Web-систем електронної торгівлі. *Економіка. Фінанси. Право*. 2017. № 8 (1). С. 20–24.
14. Пурський О. І., Мазоха Д. П., Баннікова С. О. Проблематика побудови інтегрованих інформаційних систем електронної торгівлі // Актуальні проблеми прогнозування розвитку економіки України: монографія/за ред. О. І. Черняка, П. В. Захарченка. Бердянськ: Вид-во Ткачук О. В., 2017. С. 131–139.
15. Пурський О. І., Мазоха Д. П., Жарій О. І. Функціональна модель Web-підприємства з мережею Інтернет-магазинів. *Проблеми економіки*. 2015. № 2. С. 166–171.
16. Фримен А. ASP.NET MVC 5 с примерами на C# 5.0 для профессионалов. ИД «Вильямс», 2015. 736 с.
17. Yates, C., Ladd S., Davison D., Devijver S. Expert Spring MVC and Web Flow. Apress, 2006. 424 p.

18. Mueller, J. P. Microsoft ADO.NET Entity Framework Step by Step. Microsoft Press, 2013. 448 p.

19. Jennings, R. Professional ADO.NET 3.5 with LINQ and the Entity Framework. Wrox, 2009. 672 p.

REFERENCES

Bazylevych, V. D. "Formuvannya rynku elektronnoi komertsii v Ukraini" [Formation of the e-commerce market in Ukraine]. *Visnyk Instytutu ekonomichnoho prohnozuvannia*, no. 2 (2002): 32-36.

Diubanov, O. S., and Ramazanov, S. K. "Modeli povedinky subiektiv rynku elektronnoi torhivli v suchasnykh umovakh" [Models of behavior of subjects of the e-commerce market in modern conditions]. *Biznes Inform*, no. 6 (2011): 104-105.

Ekonomika Ukrainy: stratehiia i polityka dovrostrokovooho rozvytku [Economy of Ukraine: strategy and policy of long-term development]. Kyiv: Instytut ekonomichnoho prohnozuvannia, 2003.

Frimen, A. *ASP.NET MVC 5 s primerami na C# 5.0 dlia profesionalov* [ASP.NET MVC 5 with examples in C# 5.0 for professionals]. Moscow: Vilyams, 2015.

Jennings, R. *Professional ADO.NET 3.5 with LINQ and the Entity Framework*. Wrox, 2009.

Khartman, A. *Strategii uspekha v Internet-ekonomike* [Strategies for success in the Internet economy]. Moscow: LORI, 2001.

Kheyg, M. *Osnovy elektronnoho biznesa* [Fundamentals of e-business]. Moscow: FAIR-PRESS, 2002.

Kopytko, B. I., and Yurchuk, H. V. "Poniattia elektronnoho biznesu ta klasyfikatsiia yoho osnovnykh biznes-modelei" [The concept of e-business and the classification of its main business models]. *Sotsialno-ekonomichni doslidzhennia v perekhidnyi period. Mekhanizmy rehuliuвання rehionalnoho rynku pratsi*, no. 3 (2002): 349-364.

Koze, D. *Elektronnaya kommertsiya* [E-commerce]. Moscow: Russkaya redaktsiya, 1999.

Laudon, K. C., and Traver, C. G. *E-commerce: business, technology, society*. USA: Addison Wesley, 2002.

Mazaraki, A. A. et al. *Vnutrishnia torhivlia v Ukraini: ekonomichni umovy efektyvnoho rozvytku* [Domestic Trade in Ukraine: Economic Conditions for Effective Development]. Kyiv: KNTEU, 2006.

Mazokha, D. P. "Prohramno-aparatni rishennia dlia intehrovanykh informatsiinykh Web-system elektronnoi torhivli" [Software and hardware solutions for integrated information Web-systems of e-commerce]. *Ekonomika. Finansy. Pravo*, no. 8 (1) (2017): 20-24.

Mueller, J. P. *Microsoft ADO.NET Entity Framework Step by Step*. Microsoft Press, 2013.

Porokhnia, V. M. *Modeliuвання ekonomiky* [Modeling the economy]. Zaporizhzhia: ZDIA, 2001.

Purskyi, O. I., Mazokha, D. P., and Bannikova, S. O. "Problematyka pobudovy intehrovanykh informatsiinykh system elektronnoi torhivli" [Problem of building integrated information systems of e-commerce]. In *Aktualni problemy prohnozuvannia rozvytku ekonomiky Ukrainy*, 131-139. Berdiansk: Vyd-vo Tkachuk O. V., 2017.

Purskyi, O. I., Mazokha, D. P., and Zharii, O. I. "Funktsionalna model Web-pidpriemstva z merezheiu Internet-mahazyniv" [Functional model of Web-enterprise with a network of Internet-shops]. *Problemy ekonomiky*, no. 2 (2015): 166-171.

Sytnyk, V. F., and Kozak, I. A. *Telekomunikatsii v biznesi* [Telecommunications in business]. Kyiv: KNEU, 1999.

Vitlinskyi, V. V., Melnyk, H. V., and Skitsko, V. I. "Modeliuвання zbutovoi elektronnoi lohistychnoi pidsystemy z vykorystanniam nechitkoi merezhi Petri" [Simulation of the electronic logistics subsystem with the use of a fuzzy Petri network]. *Biznes Inform*, no. 8 (2014): 82-87.

Yates, C. et al. *Expert Spring MVC and Web Flow*. Apress, 2006.