

ЗМІСТ

Скорочення та умовні позначки	3
Вступ.....	4
1 Дослідження та аналіз технології хмарових обчислень.....	7
1.1 Застосування хмарових технологій агенціями нерухомості	7
1.2 Теоретичні аспекти створення web-сервісів	8
1.3 Огляд стандартів для реалізації web-сервісів.....	10
1.4 Аналіз сервіс-орієнтованої архітектури для реалізації web-сервісів.....	14
1.5 Огляд технологій реалізації сервіс-орієнтованих додатків	16
1.6 Аналіз та вибір моделі пошукового механізму для збору інформації web-сервісом	18
1.7 Застосування технології парсингу.....	21
1.8 Постановка завдання.....	22
2 Аналіз та вибір програмних засобів реалізації web-сервісу	25
2.1 Вибір протоколу реалізації web-сервісу методом аналізу ієрархій.....	25
2.2 Особливості застосування та вимоги до web-сервісів	32
2.3 Визначення бізнес-процесів хмарного сервісу	34
2.4 Вибір системи керування базами даних	34
2.5 Вибір мови програмування	36
3 Проектування web-сервісу агенції нерухомості	38
3.1 Функціональні можливості користувачів сервісу	38
3.2 Моделювання бізнес-процесів пошукового web-сервісу.....	39
3.3 Проектування сервісу за методологією Workflow Diagramming	43
3.4 Проектування сервісу за допомогою діаграм потоків даних DFD	46
3.5 Проектування бази даних.....	48
4 Програмна реалізація web-сервісу агенції нерухомості	57
4.1 Засоби PHP для розробки web-сервісу SOAP	57
4.2 Створення класу SoapClient	58
4.2 Створення класу SoapServer.....	60

	2
4.3 Результати застосування web-сервісу	61
4.4 Додаток адміністратора web-сервісу.....	65
Висновки	70
Перелік джерел посилання	71

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

БД – база даних;

ІС – інформаційна система;

MAI (Метод аналізу ієрархій) – математичний інструмент системного підходу до складних проблем прийняття рішень.

ОС – операційна система;

СУБД – система управління базами даних;

ЦОД – центрах обробки даних;

CSS – Cascading Style Sheets – каскадні таблиці стилів;

PDA – (Personal Digital Assistant) – портативний пристрій, що володіє широкими функціональними можливостями;

SADT (Structured Analysis and Design Technique) – методологія структурного аналізу і проектування

SOA (Service Oriented Architecture) – сервіс-орієнтована архітектура

SOAP (Simple Object Access Protocol) – простий протокол доступу до об'єктів;

XML (eXtensible Markup Language) – розширювана мова розмітки;

WSDL (Web Services Description Language) – мова опису web-сервісів

UDDI ((Universal Description, Discovery and Integration) – стандарт для індексації web-сервісів;

UML (Unified Modeling Language) – мова візуального моделювання

CRM (Customer Relationship Management) – програмне забезпечення для організації та автоматизації роботи з замовниками;

ERP (Enterprise Resource Planning) – програмне забезпечення для управління ресурсами компанії;

ВСТУП

Сьогодні інформаційні технології відіграють одну з ключових ролей в ефективності бізнесу. Найбільш оперативно реагувати на ринкові зміни дозволяють хмарні технології, які стрімко впроваджуються в корпоративну IT-середовище. Завдяки своїм перевагам хмари є одним з найбільш динамічно зростаючих напрямів IT-ринку. Хмарні технології все більше проникають в усі галузі економіки, бізнес-процеси і сфери людського життя. Останнім часом, у зв'язку зі стрімким розвитком глобальної мережі Інтернет, збільшилася потреба у появі послуг, що надаються в реальному часі. На сьогоднішній день існує безліч сервісів, що надають послуги з обміну інформацією в мережі Internet, які стали невід'ємною і важливою частиною ефективності роботи багатьох фірм, підприємств і організацій.

В хмарах сьогодні розгортають безліч інтерактивних сервісів для користувачів, наприклад, онлайн-перекладачі, онлайн-ігри, миттєві платежі, навігаційні сервіси, онлайн-бібліотеки, музичні агрегатори, держпослуги, а також звичні програми для роботи з текстами, таблицями, фотографіями та т.п. Уже сьогодні в корпоративних хмарах розміщують call-центри, поштові сервіси, сайти, форми замовлень, а також програмні продукти для електронного документообігу, автоматизації виробництва, логістики та торгівлі, бухгалтерські та податкові системи компанії, і навіть самі робочі столи співробітників для віддаленої роботи [1]¹⁾.

Останнім часом активно зростає кількість онлайн веб-сервісів, щодня з'являються нові онлайн ресурси здатні взаємодіяти з користувачами без значних трудових витрат з боку компанії. Насправді онлайн сервіс є основним співробітником, який працює безвідмовно, цілодобово, без вихідних і абсолютно безкоштовно. Веб-сервіс надає нові можливості для розвитку бізнесу, залучення клієнтів і збільшення доходів. Онлайн веб-сервіси розробляються

¹⁾ [1] Міжнародний досвід застосування інтернет-маркетингу. Л. С. Ринейська. Ефективна економіка, №3 2015. URL: <http://www.economy.nayka.com.ua/?op=1&z=3895> (дата звернення: 15.05.2019).

для різних тематик і сфер детальності компанії. Багато компаній, все частіше, приходять до вирішення застосування веб-сервісу в своїй діяльності, оскільки звичайний сайт вже не відповідає їхнім потребам або завданням нового проекту. Можливості розробки веб-сервісу присутні для будь-якої сфери діяльності, наявність такого каналу взаємодії з відвідувачами позитивно позначиться на розвитку компанії і дозволить вийти бізнесу на абсолютно новий рівень [2]¹⁾.

Співробітники агенцій нерухомості – ріелтори щодня обробляють безліч сайтів або дощок оголошень про нерухомість, тому що для успішної конкуренції необхідно стежити за новими оновленнями і регулярно «обдзвонювати» старі. Знайти вигідне оголошення від власника квартири за запитом клієнта – складне завдання.

Тому актуальним є завдання проектування та розробки веб-сервісу, що здійснює пошук серед визначених інформаційних ресурсів необхідних об'єктів нерухомого майна за обраними критеріями.

Метою магістерської роботи є дослідження технології хмарних обчислень та реалізація хмарного сервісу підтримки діяльності агенції нерухомості, що здійснює швидкий пошук об'єктів нерухомого майна серед зазначених ресурсів за необхідними критеріями.

У даній магістерській роботі розглядається питання створення web-сервісів і їх використання у складі розподіленої Internet-системи.

Для досягнення мети необхідно вирішити наступні завдання:

- провести дослідження технології хмарних обчислень та предметної області для створення веб-сервісу;
- сформулювати вимоги та визначити функції, що буде реалізовувати веб-сервіс;
- здійснити проектування функціональної моделі взаємодії компонентів веб-сервісу;

¹⁾ [2] Введення в веб-сервіси EasyCode. URL: <http://easy-code.com.ua/2011/03/vvedennya-v-veb-servisi/> (дата звернення: 15.05.2019).

- провести вибір апаратно-програмної платформи для реалізації хмарного сервісу;
- виконати програмну реалізацію сервісу.

Хмарний веб-сервіс підтримки діяльності агенції нерухомості повинен задовольняти наступним вимогам: надійність, простота використання та доступність для користувачів, легкість впровадження для агенцій нерухомості, конфіденційність даних. Використання такого веб-сервісу агенціями нерухомості набагато покращить якість та швидкість надання послуг пошуку необхідного об'єкту нерухомого майна. Розроблений веб-сервіс підтримки діяльності агенції нерухомості може бути використано інформаційними ресурсами, що надають інформацію про об'єкти нерухомості.

Магістерська робота містить в собі 77 сторінок, 13 таблиць, 35 рисунків, 14 посилань.

1 ДОСЛІДЖЕННЯ ТА АНАЛІЗ ТЕХНОЛОГІЇ ХМАРОВИХ ОБЧИСЛЕНЬ

1.1 Застосування хмарових технологій агенціями нерухомості

Під хмарними обчисленнями (від англ. cloud computing), або ж використовується термін «хмарна обробка даних», зазвичай розуміється надання користувачу комп'ютерних ресурсів і потужностей у вигляді Інтернет-сервісу. Таким чином, обчислювальні ресурси надаються користувачу в «чистому» вигляді, і користувач може не знати, які комп'ютери обробляють його запити, під керуванням якої операційної системи це відбувається і т.д. Термінал в хмарі є потужним обчислювальним пристроєм, здатним не тільки накопичувати проміжну інформацію, але і безпосередньо управляти глобальною системою обчислювальних ресурсів. В даний час великі обчислювальні хмари складаються з тисяч серверів, розміщених в центрах обробки даних (ЦОД). Вони забезпечують ресурсами десятки тисяч додатків, які одночасно використовують мільйони користувачів. Хмарні технології є зручним інструментом для підприємств, яким занадто дорого утримувати власні ERP, CRM або інші сервери, що вимагають придбання і налаштування додаткового обладнання. За моделлю розгортання хмари поділяють на приватні, загальнодоступні (публічні) та гібридні. Приватні хмари – це внутрішні хмарні інфраструктура і служби підприємства. Ці хмари знаходяться в межах корпоративної мережі. Організація може керувати приватною хмарою самостійно а. Загальнодоступні (публічні) хмари – це хмарні послуги, що надаються постачальником. Хмари знаходяться за межами корпоративної мережі. Їх користувачі не мають можливості управляти даними хмарою або обслуговувати її – вся відповідальність покладена на власника хмари [3]¹⁾.

Стрімке поширення хмарних технологій ставить завдання інтеграції хмарних сервісів в сферу нерухомості. Хмарні обчислення мають широкі пе-

¹⁾ [3] Хмарні технології. Переваги і недоліки. Валтек системний інтегратор – ІТ послуги системної інтеграції в Україні. URL: <https://valtek.com.ua/ua/system-integration/it-infrastructure/clouds/cloud-technologies> (дата звернення: 10.07.2019).

спективи використання в сфері нерухомості. Ігнорувати сьогодні хмарні технології, не можливо. Навіть якщо сьогодні споживачу не потрібно перенести робочі процеси і бази даних в «хмару», не варто зовсім відмовлятися від можливостей хмарних технологій. Забезпечення безпеки і протидія кіберзагрозам, back-up заявок, даних і контактів, економія на програмному забезпеченні є значними факторами, які можуть не грати важливого значення сьогодні, але можуть виявитися критичними завтра, коли локальні сервера переважать і перестануть функціонувати, і на їх відновлення знадобиться час, за яке конкуренти можуть перехопити клієнтів. Придбання та обслуговування різного роду програмного забезпечення і комп'ютерної техніки регулярно вимагає істотних фінансових вкладень і залучення кваліфікованих фахівців [4]¹⁾.

Сьогодні організації в сфері нерухомості застосовують послуги хмарних технологій, отримують їх безкоштовно або за невелику плату, причому нерідко подібні послуги надаються більш надійними і доступними в порівнянні з їх розміщенням або супроводом в самій агенції.

1.2 Теоретичні аспекти створення web-сервісів

Web-сервіси являють собою технологію інтеграції додатків, яка може використовуватися в інформаційних системах, побудованих з використанням технологій Internet. Web-сервіс не є продуктом для кінцевого користувача. Він являє собою заснований на компонентах додаток, дозволяючи багаторазово використовувати свою функціональність в різних середовищах і на клієнтах різних типів. Користувачем web-сервісу завжди є інший додаток [4]¹⁾.

Web-служби можуть використовуватися для вирішення різного роду завдань. З метою здійснення B2B (business-to-business) транзакцій або з'єд-

¹⁾ [4] Хмарні обчислення в правовому полі України « Взгляд юриста». URL: <http://jurblog.com.ua/2016/08/hmarni-obchislennya-v-pravovomu-poli-ukrayini/> (дата звернення: 10.07.2019)

нання внутрішніх систем окремих компаній. Web-сервіси засновані на використанні загальноприйнятих, відкритих технологій Internet, що не залежать від постачальників ПЗ, за допомогою чого досягається головна перевага web-сервісів як концепції побудови розподілених ІС – їх універсальність.

Таким чином, розподілені додатки, побудовані з використанням технології web-сервісів, забезпечують інтеграцію різнорідних додатків різних учасників єдиного бізнес-процесу. Подібна інтеграція називається додатками B2B. На сьогоднішній день це найбільш широко поширені завдання, які вирішуються із застосуванням web-сервісів. Такі сервіси дозволяють спільно використовувати інформацію або можуть інтегруватися з іншими службами. У подібних випадках в якості користувача web-служби, швидше за все, буде виступати програмне забезпечення, встановлене в такій компанії. Також web-сервіс можна використовувати як готові модулі для розробників. Припустимо, незалежний розробник спроектував web-службу аутентифікації, призначену для застосування в середовищі ASP.NET. Однак даний процес буде абсолютно прозорим для кінцевого користувача, який вирішить, що вказані засоби аутентифікації є частиною вашої програми. Такі готові компоненти можна використовувати в web-додатках, а також в настільних і мобільних програмах. Існує можливість використовувати web-сервіс в якості компонентних бібліотек DLL для багаторазового використання коду. Найпростіший спосіб багаторазового використання певних функціональних можливостей в Інтернет-додатках полягає в проектуванні web-сервісу, до якого могли б звертатися різні клієнти, в тому числі настільні додатки, PDA і браузері. При цьому не важливо, де розташовуються web-служби і клієнти, необхідно лише наявність Інтернет-з'єднання між клієнтом і службою [5]¹⁾.

¹⁾ [5] Технологии и средства связи. URL: lib.tssonline.ru/articles2/fix-corp/oblastnye-vozmozhnosti-preimushchestva-i-perspektivy-ispolzovaniya-oblastnyh-tehnologiy-v-sovremennom-mire (дата звернення: 18.08.2019).

1.3 Огляд стандартів для реалізації web-сервісів

Web-служба або web-сервіс – це програмна система, що ідентифікується рядком URI, чиї загальнодоступні інтерфейси визначені на мові XML. Опис цієї програмної системи може бути знайдено іншими програмними системами, які можуть взаємодіяти з нею відповідно до цього опису допомогою повідомлень, заснованих на XML, і переданих за допомогою інтернет-протоколів. Web-служба є одиницею модульності при використанні сервіс-орієнтованої архітектури додатку. Сервіс – це ресурс, який реалізує якусь функцію з виконання будь-яких дій і має такі властивості [6]¹⁾:

- є функціонально самостійним об'єктом;
- є повторно використовуваним;
- функціонує незалежно від інших ІС;
- являє собою «чорний ящик» для будь-яких зовнішніх додатків;
- визначається відповідно до певних правил;
- може бути викликаний за допомогою комунікаційних протоколів;
- володіє незалежністю від МП, на якому реалізований сервіс, від платформи, на якій він функціонує;
- володіє прозорістю розташування.

Таким чином, з функціональної точки зору, Internet-додаток розпадається на сукупність взаємодіючих між собою web-сервісів.

Оскільки web-сервіс є функціонально самостійним і повторно використовуваним об'єктом, його можна застосовувати неодноразово в різних web-додатках або web-системах. Оскільки web-сервіс функціонує незалежно від інших ІС (інформаційна система), його можна застосовувати в різних незалежних один від одного web-додатках або web-системах.

Оскільки web-сервіс є «чорний ящик», то для будь-яких зовнішніх додатків все, що повинно бути відомо зовнішнім додаткам це те, що необхідно

¹⁾ [6] Веб-сервисы в теории и на практике для начинающих / Хабр. URL: <https://habr.com/ru/post/46374/> (дата звернення: 19.08.2019).

подати на вхід сервісу і що слід очікувати на виході. В основу web-сервісів покладено ряд технологій і стандартів, і, виходячи з цього він визначається відповідно до певних правил. Web-сервіси покликані працювати в мережевому середовищі, тому одним з їхніх властивостей є те, що вони можуть бути викликані за допомогою комунікаційних протоколів, зокрема за допомогою протоколу HTTP.

Основу Web-сервісів складають платформи-незалежні технології World Wide Web (WWW), тому вони мають незалежність від мови програмування, на якому реалізовані сервіси, і від платформи, на якій вони функціонують. Користувачеві, який взаємодіє з додатком, що використовує результат роботи web-сервісу, немає необхідності знати про місцезнаходження сервісу – він може перебувати як на тому ж комп'ютері, що і сам додаток, так і на віддаленому сервері. Тому можна стверджувати, що web-сервіси мають прозорість розташування [7]¹⁾. Важлива властивість web-сервісу полягає в тому, що він не залежить від провайдера, комп'ютера або браузера з його використанням користувач може працювати зі своїми даними в будь-якій точці світу, де у нього є доступ до Internet. Доступ до web-сервісів здійснюється по протоколу HTTP, а обмін даними відбувається у форматі XML (рис.1.1).



Рисунок 1.1 – Схема роботи web-сервісу

¹⁾ [7] Веб-сервисы. Сетевые технологии. URL: <http://www.4stud.info/networking/web-services.html> (дата звернення: 19.08.2019).

Якщо розділити Інтернет на кілька шарів, ми зможемо виділити, як мінімум, два концептуальних типу додатків – обчислювальні вузли, які реалізують нетривіальні функції та прикладні web-ресурси. При цьому другий, найчастіше зацікавлені в послугах перших. Але й сама мережа Інтернет різнорівня, тобто різні додатки на різних вузлах мережі функціонують на різних апаратно-програмних платформах, і використовують різні технології і мови. Щоб зв'язати все це і надати можливість одним додаткам обмінюватися даними з іншими, і були придумані web-сервіси. Web-сервіси – це реалізація чітких інтерфейсів обміну даними між різними додатками, які написані не тільки на різних мовах, але і розподілені на різних вузлах мережі.

Web-сервіси базуються на трьох основних web-стандартах:

- SOAP – протокол для посилки повідомлень по протоколу HTTP та іншим Internet-протоколам;
- WSDL – мова для опису програмних інтерфейсів web-сервісів;
- UDDI – стандарт для індексації web-сервісів.

Web-сервіси надають інтерфейси для передачі компонентних даних і бізнес-логіки по HTTP. Для того щоб програми могли використовувати web-сервіси, програмні інтерфейси останніх повинні бути детально описані. Опис може включати таку інформацію, як протокол, адреса сервера, номер використовуваного порту, список доступних операцій, формат запиту та відповіді і т.п. Існуючі web-сервіси описуються в WSDL-документах, які розташовуються або на сервері додатків, або в спеціальних XML-сховищах. WSDL-документ може посилатися на інші WSDL-документи та документи XSD (XML Schema), в яких описані типи даних, використовувані web-сервісами. XML-сховища використовуються для управління WSDL-документами. У середині WSDL-документа знаходиться адреса (URL) web-сервісу. Web-сервіси описані і проіндексовані в бізнес-реєстрі, що містить адреси (URL) WSDL-документів [7]¹⁾.

¹⁾ [7] Веб-сервисы. Сетевые технологии. URL: <http://www.4stud.info/networking/web-services.html> (дата звернення: 19.08.2019).

SOAP – це стандарт для відсилання та отримання повідомлень по мережі Internet. Специфікація SOAP визначає XML-конверт для передачі повідомлень, метод для кодування програмних структур даних у форматі XML, а також засоби зв'язку по протоколу HTTP. SOAP-повідомлення бувають двох типів: запит (Request) і відповідь (Response). Запит викликає метод віддаленого об'єкта, відповідь повертає результат виконання даного методу.

UDDI (Universal Description, Discovery and Integration) – інструмент для розміщення описів web-сервісів. Завдання UDDI – це надати механізм для виявлення web-сервісів. UDDI задає бізнес-реєстр, в якому провайдери web-сервісів можуть реєструвати сервіси, а розробники – шукати необхідні їм сервіси. Бізнес-реєстр UDDI сам є SOAP web-сервісом. Він підтримує операції створення, модифікації, видалення та пошуку web-сервісів [3]¹⁾. Схема роботи web-сервісів (рис. 1.2).

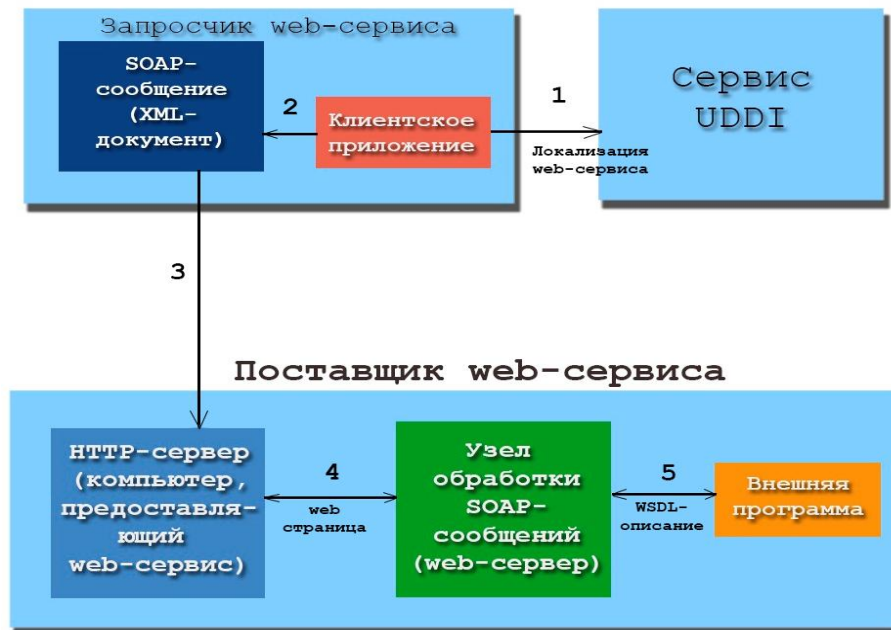


Рисунок 1.2 – Схема функціонування web-сервісів

¹⁾ [3] Хмарні технології. Переваги і недоліки. | Валтек системний інтегратор – ІТ послуги системної інтеграції в Україні. URL: <https://valtek.com.ua/ua/system-integration/it-infrastructure/clouds/cloud-technologies> (дата звернення: 10.07.2019).

Програма-клієнт повинна знати: які методи сервісу доступні, які параметри потрібні і тип, що повертається результатом. Web-сервіс повертає ці дані в SOAP-форматі. Для того, щоб передати web-сервісу вказівки зробити що-небудь, ми повинні очевидним чином запакувати наші параметри в XML, причому обов'язково в тому вигляді, в якому web-сервіс зрозуміє. Після цього за допомогою HTTP SOAP переслати ці параметри web-сервісу, і після отримання результату від web-сервісу теж у вигляді SOAP-пакета, дешифрувати ці SOAP-пакети і відобразити їх в браузері.

Кожне SOAP-повідомлення поміщається в SOAP-конверт – одиницю обміну повідомленнями. SOAP-запит – це звичайний HTTP POST запит.

У першому рядку знаходиться команда POST, за нею – URI сервісу. Крім цього, зазначений Content-Type – це вказівка браузеру про тип документу, який повинен бути обов'язково в форматі XML, тому що в противному випадку SOAP-пакет не буде правильно декодований назад.

1.4 Аналіз сервіс-орієнтованої архітектури для реалізації web-сервісів

Для здійснення проектування хмарного web-сервісу підтримки діяльності агенції нерухомості, який здійснює пошук об'єктів нерухомості, обрана архітектура SOA. Ця архітектура являє собою модульний підхід до розробки програмного забезпечення, заснований на використанні сервісів (служб) зі стандартизованими інтерфейсами. У сьогоdnішній трактуванні SOA під сервісами розуміють web-сервіси, в основі яких лежать загальноприйняті Internet-технології та розвинена інфраструктура.

В основі SOA лежать принципи багатократного використання функціональних елементів, ліквідації дублювання функціональності в програмному забезпеченні, уніфікації типових операційних процесів, забезпечення переведення операційної моделі компанії на централізовані процеси і функціональну організацію на основі промислової платформи інтеграції.

Компоненти програми можуть бути розподілені по різних вузлах мережі, і пропонуються як незалежні, слабо пов'язані, замінні сервіси-додатки. Програмні комплекси, розроблені відповідно до SOA, часто реалізуються як набір web-сервісів, інтегрованих за допомогою відомих стандартних протоколів (SOAP, WSDL, UDDI і т. п.). Важлива властивість web-сервісу полягає в тому, що він не залежить від провайдера, комп'ютера або браузера, а з його використанням, користувач може працювати зі своїми даними в будь-якій точці світу, де у нього є доступ до Internet. Доступ до web-сервісів здійснюється по протоколу HTTP, а обмін даними відбувається у форматі XML.

Базова архітектура SOA складається з провайдера сервісів, сервісу і (необов'язкового) каталогу сервісів. Для обміну інформацією використовується механізм обміну повідомленнями типу "додаток до додатка". Подібність між цією моделлю і чистими web-сервісами цілком очевидно, оскільки в обох випадках застосовується WSDL-документ (web-services description language), що є контрактом з активізації, що зберігаються в каталозі сервісів, з якого цей сервіс може бути запитаний і витягнутий завдяки механізму UDDI. Web-сервіси насправді є реалізацією архітектури SOA на самому базовому рівні.

Спочатку провайдер сервісу створює сервіс, приймає рішення відкрити цей сервіс і публікує його. Публікація виконується шляхом відправки інформації про сервіс в каталог сервісів. З іншого боку, ініціатор запитів до сервісу, маючи потребу в певному сервісі, переглядає каталог web-служб в пошуках тієї з них, який задовольняє необхідному критерію.

Після виявлення такого сервісу і використання доступної в каталозі сервісів інформації ініціатор запитів сервісу може напряму звернутися до провайдера сервісів належним способом для задоволення бізнес-потреби.

Кожну архітектуру, яка реалізується цими засобами, цілком можна назвати орієнтованою на сервіси, але при цьому кожна з них визначає свої власні формати і протоколи, механізми викликів, інтерфейси для прикладних програм. Схема запитів представлена на рис. 1.3.

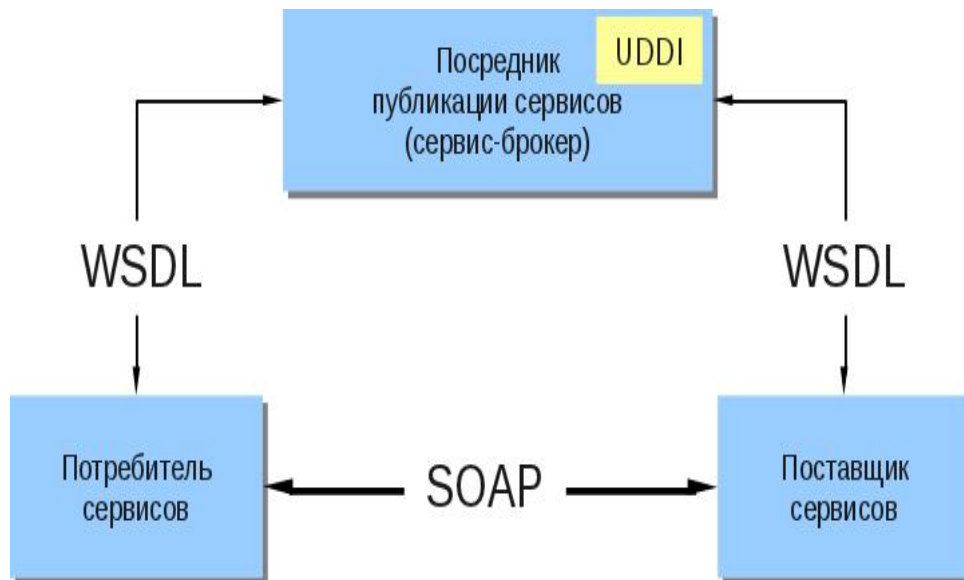


Рисунок 1.3 – Базова архітектура SOA

Ідея сервісів в інформаційних системах не нова. Добре відомі такі підходи до її реалізації:

- Java RMI від компанії Sun Microsystems (Java Remote Method Invocation);
- CORBA консорціуму Open Management Group (Common Object Request Broker Architecture);
- DCE запропонована асоціацією Open Group (Distributed Computing Environment);
- Microsoft DCOM (Distributed Component Object Model).

Зараз настав момент, коли компанії повинні серйозно розглядати можливість використання web-сервісів. Проблеми безпеки та стандартизації web-сервісів активно вирішуються, тому продукти, засновані на технології web-сервісів, все активніше входять в повсякденну роботу компаній.

1.5 Огляд технологій реалізації сервіс-орієнтованих додатків

Існує три основні технології реалізації сервіс-орієнтованого Internet-додатку, заснованого на застосуванні web-сервісів: віддалений виклик проце-

дур за допомогою XML-RPC; передача стану (Representation State Transfer (REST) за допомогою RESTful web-сервісів; використання web-сервісів на основі SOAP. Для реалізації web-сервісу підтримки діяльності агенції нерухомості, що здійснює пошук за обраними критеріями, доцільно обрати технологію на основі SOAP. Це найпопулярніший і широко поширений спосіб реалізації web-сервісів, проте існують і інші архітектури.

XML-RPC – XML-виклик віддалених процедур – це протокол виклику віддалених процедур, заснований на XML. Він є прабатьком SOAP, відрізняється винятковою простотою застосування. У ньому визначаються виклики процедур, закодовані на XML і передані по HTTP. Як і будь-який інший інтерфейс, RPC визначає набір стандартних типів даних і команд, які можна використовувати для доступу до функціональності іншої програми, що знаходиться на іншому комп'ютері в мережі. Після декількох циклів по розширенню функціональності, з'явилася система, відома як SOAP. Пізніше компанія Microsoft почала широко рекламувати і впроваджувати SOAP, а початковий XML-RPC був відкинутий компанією-розробником (все тією ж Microsoft). Але, незважаючи на це, стандарт XML-RPC набув популярності серед програмістів через свою простоту і, за рахунок цього, існує донині. Однак, технологія XML-RPC має ряд істотних недоліків, включаючи дуже примітивний контроль типів даних і відсутність підтримки кодування символів. REST – це стиль архітектури програмного забезпечення для розподілених систем, таких як World Wide Web, який, як правило, використовується для побудови web-сервісів. Термін REST був введений у 2000 році Роем Філдінгом, одним з авторів HTTP-протоколу. Системи, що підтримують REST, називаються RESTful-системами [8]¹⁾.

У загальному випадку REST є дуже простим інтерфейсом управління інформацією без використання якихось додаткових внутрішніх прошарків. Кожна одиниця інформації однозначно визначається глобальним ідентифіка-

¹⁾ [8] UBS. Публикации о технологиях веб-сервисов. URL: http://www.ubs.ru/ws/ws_soap1.html (дата звернення: 26.09.2019).

тором, таким як URL. Передбачається відмовитися від розробки нових протоколів, а використовувати кілька добре перевірених, вважаючи, що для роботи з об'єктами цілком достатньо вміти виконувати чотири типи дій: створення (Creation), відновлення (Retrieval), зміна (Update) і знищення (Destruction). З цих дій виходить так званий «шаблон проектування» CRUD. Протокол HTTP визначає методи GET/PUT/POST/DELETE, які й реалізують шаблон CRUD. Для успішної роботи в Web-сервісу необхідно підтримувати складний механізм, що забезпечує взаємодію на випадок внесення системних змін. Розглянемо більш детально архітектуру, що базується на використанні протоколу SOAP для реалізації web-сервісу. Веб-сервіс – це базова однонаправлена модель з'єднання, що забезпечує узгоджену передачу повідомлення від відправника до одержувача, потенційно допускає наявність посередників, які можуть обробляти частину повідомлення або додавати до нього додаткові елементи. SOAP є найголовнішою частиною технології Web-сервісів, здійснює перенесення даних по мережі з одного місця в інше [8]¹⁾.

SOAP забезпечує доставку даних веб-сервісів. SOAP дозволяє відправнику і одержувачу XML-документів підтримувати загальний протокол передачі даних, що забезпечує ефективність мережевого зв'язку.

1.6 Аналіз та вибір моделі пошукового механізму для збору інформації web-сервісом

Мережа Інтернет є готовою платформою для створення і використання розподілених машинно-орієнтованих систем на основі web-сервісів. Технологія web-сервісів виступає в якості механізму сервера додатків, до якого звертаються не кінцеві користувачі, а сторонні додатки. Це дозволяє багаторазово використовувати функціональні елементи, усунути дублювання коду, спростити вирішення завдань інтеграції додатків.

¹⁾ [8] UBS. Публикации о технологиях веб-сервисов. URL: http://www.ubs.ru/ws/ws_soap1.html (дата звернення: 26.09.2019).

Web-сервіси (або веб-служби) – це технологія, яка досить чітко окреслену середу застосування. Якщо подивитися на web-сервіси в розрізі стека мережевих протоколів, то їх можна визначити як ще одну надбудову над протоколом HTTP. З іншого боку, в мережі можна виділити два концептуальних типу додатків – обчислювальні вузли, які реалізують нетривіальні функції та прикладні веб-ресурси. Прикладні веб-ресурси часто потребують послуги перших. Але різноманітність мережі Інтернет вимагає забезпечення функціонування різних додатків на різних вузлах мережі, на різних апаратно-програмних платформах, на різних технологіях і мовах. Застосування технології web-сервісів забезпечує можливість одним додаткам обмінюватися даними з іншими. По суті, web-сервіси – це реалізація абсолютно чітких інтерфейсів обміну даними між різними додатками, які написані не тільки на різних мовах, але і розподілені на різних вузлах мережі. Сучасні web-сервіси використовують наступні протоколи: SOAP (Simple Object Access Protocol); REST (Representational State Transfer), XML-RPC (XML Remote Procedure Call). Для використання різноманітних Інтернет-послуг мережі Інтернет існує безліч web-сервісів. Вони діляться на такі основні типи: пошукові системи, поштові сервіси, Інтернет-форуми, соціальні мережі. Як правило, такі сайти складаються з сторінок користувачів, груп і безлічі інших сервісів. Дані сайти зручні для знайомства, переписки та подальшого спілкування. Для реалізації web-сервісу підтримки діяльності агенції нерухомості доцільно здійснити розробку пошукової системи для пошуку тієї чи іншої інформації з продажу нерухомості серед зазначеної будь-якої кількості визначених агентств. Для цього були розглянуті загальні принципи роботи пошукових систем. Пошукова система складається з наступних основних компонентів [6]¹⁾:

- Spider (павук) – програма, яка завантажує веб-сторінки;
- Crawler (краулер) – програма, яка автоматично проходить по всіх посиланнях, знайденим на сторінці;

¹⁾ [6] Веб-сервисы в теории и на практике для начинающих / Хабр. URL: <https://habr.com/ru/post/46374/> (дата звернення: 19.08.2019).

- Indexer (індексатор) – програма, яка аналізує веб-сторінки;
- Database – сховище викачаних і оброблених сторінок;
- Search engine results engine (система видачі результатів) – витягує результати пошуку з бази даних;
- Web-server (веб-сервер) – здійснює взаємодію між користувачем і іншими компонентами пошукової системи.

Детальна реалізація пошукових механізмів може відрізнятися одна від одної, проте всім пошуковим системам притаманні спільні риси. Реалізація web-сервісу для підтримки діяльності агентства нерухомості вимагає розробки модулів, які реалізують функції зазначених компонентів.

Реалізація компонента Spider забезпечує завантаження web-сторінки тим же способом, що і браузер користувача. Відмінність полягає в тому, що браузер відображає інформацію, що міститься на сторінці (текстову, графічну), а спайдер не має жодних візуальних компонент і працює з html-текстом сторінки.

Crawler забезпечує виділення всіх посилань, які виявлені на аналізованій сторінці для визначення подальших пошукових дій компонента Spider. Список адрес ресурсів які в подальшому мають бути проаналізовані визначаються заздалегідь.

Механізм пошуку Spider для агенції нерухомості, слідуючи по знайдених посиланнях, здійснює пошук нових документів, ще невідомих пошуковій системі. Компонент Indexer в подальшому здійснює розбір сторінки на складові частини і аналізує їх. Виділяються і аналізуються різні елементи сторінки, такі як текст, заголовки, структурні та стильові особливості, спеціальні службові html-теги та інші компоненти коду. Безпосередньо Indexer і здійснює пошук необхідного об'єкту нерухомості по заданих у пошуку критеріях.

Реалізація компонента Database для забезпечення функціонування web-сервісу підтримки діяльності агенції нерухомості – це сховище всіх даних, які пошукова система отримує і аналізує.

Система видачі результатів та ранжируванням сторінок забезпечується компонентом Search Engine Results. Цей компонент здійснює аналіз сторінок на запит користувача, і вирішує в якому порядку вони повинні бути відсортовані. Це відбувається за алгоритмами ранжирування пошукової системи. Розробка Web-server для зазначеного web-сервісу забезпечує присутність сторінки з полем введення, в якому користувач може задати свій пошуковий запит. Веб-сервер також відповідає за видачу результатів користувачу у вигляді html-сторінки.

1.7 Застосування технології парсингу

Трудомістку роботу зі збору інформації можна і потрібно автоматизувати за допомогою технології парсинг. Парсинг інформаційних ресурсів – це швидка обробка інформації в мережі за заданим алгоритмом. Виконання механізму парсингу у системі, задаються спеціальним виразом на мові програмування, званім регулярним виразом. Дії парсеру у системі розробляє розробник та оговорює у масці парсеру. Модуль, що реалізує механізм парсингу створюються за допомогою різних мов програмування (наприклад, мовою програмування PHP та HTML) [9]¹⁾.

Алгоритм роботи парсингу передбачає виконання наступних дій:

- здійснення пошуку в мережі Інтернет ресурсів (сторінок) з потрібною інформацією;
- витяг тексту згідно алгоритму, прописаного з командного рядка;
- формування тексту, включаючи аналіз контенту, відсів непридатного матеріалу, збереження сторінки;
- перехід на інші веб-сторінки для подальшого збору даних.

Парсинг – це синтаксичний аналіз сайтів, який автоматично проводиться парсером – спеціальною програмою або скриптом. Характер парсинга ви-

¹⁾ [9] Литвин В.В., В.А. Висоцька, Д.Г. Досин. Методи та засоби опрацювання інформаційних ресурсів на основі онтологій. Львів: ЛА «Піраміда», 2016. 404 с.

значається завданням отримати певну інформацію зі сторінок сайту, параметри аналізу заздалегідь задаються. Зібрана інформація надається в певному виді, і проводиться на одному з мов програмування.

Перевагами застосування технології парсенгу є надання наступних можливостей користувачу:

- робота парсера відбувається в автоматичному режимі, що звільняє користувача від необхідності постійно здійснювати пошук, а вимагає лише прийняти результати збору даних;
- дозволяє заощадити на створенні контенту, запозичуючи його з інших джерел;
- дозволяє заощадити час на пошук і аналіз інформації за необхідними критеріями, автоматично та швидко обробляючи великий масив даних;
- використання динамічної інформації на сайті, або в системі за рахунок постійного оновлення даних в автоматичному режимі.

1.8 Постановка завдання

Для здійснення розробки та проектування хмарного web-сервісу підтримки діяльності агентства нерухомості, а зокрема створення пошукового механізму web-сервісу, що здійснює пошук об'єктів нерухомого майна за обраними критеріями було використано технологію парсеру. Для реалізації поставленої мети та коректного функціонування web-сервісу достатньо використувати парсер, який з частотою один раз в годину збирає інформацію з п'яти обраних сайтів нерухомості.

Для здійснення пошуку у web-сервісі підтримки діяльності агенції нерухомості були обрані наступні критерії:

- назва оголошення;
- тип нерухомості;
- розділ нерухомості;

- регіон або область;
- місто.

Коли парсер знаходить всі критерії, вони порівнюються з відповідними полями у базі даних, і якщо немає подібних, то створюється нове оголошення, та записується у базу даних. Парсер працює до тих пір, поки не перевірить всі оголошення з сайтів, і якщо немає нових, то він відключається, до наступного запуску.

Для здійснення проектування і програмної реалізації хмарного сервісу підтримки діяльності агенції нерухомості потрібно вирішити наступні завдання:

- провести дослідження основних принципів сервіс-орієнтованої архітектури, як основи для реалізації web-сервісів;
- провести огляд областей застосування, теоретичних основ і способів створення web-сервісів;
- розглянути застосування сервіс-орієнтованої архітектури як основи для реалізації web-сервісів;
- провести аналіз та здійснити вибір моделі пошукового механізму для збору інформації про оголошення, з сайтів нерухомості;
- сервіс підтримки діяльності агентства нерухомості повинен підтримувати три категорії користувачів: зареєстрований користувач, який має право, на розміщення оголошень, про продаж нерухомості, та видалення своїх оголошень, якщо вони вже не актуальні; адміністратор web-сервісу, який стежить за оголошеннями, а також є адміністратором бази даних.

У рамках даної магістерської роботи потрібно створити сервіс-орієнтовану систему, яка була б частиною розподіленої Internet-системи, призначена для підтримки інформаційного ресурсу агенції з нерухомості, який забезпечує швидкий пошук нових оголошень, розміщених користувачами за останній час.

Web-сервіс підтримки діяльності агенції з нерухомості повинен забезпечувати виконання наступних функцій:

- пошук інформації про продаж об'єктів нерухомості за допомогою технології парсеру;
- надання останньої інформації з ринку нерухомості;
- надання доступу до статей та новин;
- для адміністратора сервісу повинна підтримуватися можливість повного управління базою даних;
- адміністратор повинен мати можливість додавати нових користувачів в систему та розподіляти права доступу до інформації;
- для адміністратора сервісу повинна підтримуватися можливість управління всієї інформації в системі.

Реалізація і подальше застосування такого web-сервісу дозволить агентству з нерухомості швидко реагувати на новини ринку нерухомого майна, володіти найактуальнішою інформацією що до об'єктів нерухомості, і як наслідок – поліпшення якості надання послуг з купівлі та продажу нерухомого майна для своїх клієнтів.

2 АНАЛІЗ ТА ВИБІР ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ WEB-SERVICES

2.1 Вибір протоколу реалізації web-сервісу методом аналізу ієрархій

Для здійснення проектування та розробки web-сервісу підтримки діяльності агенції з нерухомості необхідно здійснити вибір протоколу для реалізації web-сервісу. Враховуючи всі можливі переваги застосування протоколів, скористаємося методом аналізу ієрархій. Саме цей метод допоможе вирішити проблему, яку саме архітектуру використовувати при розробці web-сервісу. Проведемо аналіз наступних протоколів: SOAP, XML-RPC, REST (технологія).

Метод аналізу ієрархій (MAI) – математичний інструмент системного підходу до складних проблем прийняття рішень. MAI дозволяє особі, що приймає рішення, знайти такий варіант (альтернативу), який найкращим чином узгоджується з його розумінням суті проблеми та вимогами до її вирішення. MAI дозволяє зрозумілим і раціональним чином структурувати складну проблему прийняття рішень у вигляді ієрархії, порівняти і виконати кількісну оцінку альтернативних варіантів рішення. Метод аналізу ієрархій використовується у всьому світі для прийняття рішень у різноманітних ситуаціях: від управління на міждержавному рівні до рішення галузевих і приватних проблем у бізнесі, промисловості, охороні здоров'я та освіті. Аналіз проблеми прийняття рішень в MAI починається з побудови ієрархічної структури, яка включає мету, критерії, альтернативи і інші фактори, що впливають на вибір. Ця структура відображає розуміння проблеми особою, яка приймає рішення [10]¹⁾.

Кожен елемент ієрархії може представляти різні аспекти розв'язуваної задачі, причому до уваги можуть бути прийняті як матеріальні, так і нематеріальні чинники, вимірювані кількісні параметри та якісні характеристики,

¹⁾ [10] Подлиновский В.В. Введение в теорию важности критериев в многокритериальных задачах принятия решений. М.: Физматлит, 2007. 64 с.

об'єктивні дані і суб'єктивні експертні оцінки. Наступним етапом аналізу є визначення пріоритетів, які представляють відносну важливість або перевагу елементів побудованої ієрархічної структури, за допомогою процедури парних порівнянь. На заключному етапі аналізу виконується синтез (лінійна згортка) пріоритетів на ієрархії, в результаті якої обчислюються пріоритети альтернативних рішень щодо головної мети. Кращою вважається альтернатива з максимальним значенням пріоритету.

Порядок застосування методу аналізу ієрархій:

- побудова якісної моделі проблеми у вигляді ієрархії, що включає мету, альтернативні варіанти досягнення цілі і критерії для оцінки якості альтернатив;
- визначення пріоритетів всіх елементів ієрархії з використанням методу парних порівнянь;
- синтез глобальних пріоритетів альтернатив шляхом лінійної згортки пріоритетів елементів на ієрархії;
- перевірка суджень на узгодженість;
- прийняття рішення на основі отриманих результатів.

Перший крок МАІ – побудова ієрархічної структури, що об'єднує мету вибору, критерії, альтернативи і інші фактори, що впливають на вибір рішення. Побудова такої структури допомагає проаналізувати всі аспекти проблеми і глибше вникнути в суть завдання. Ієрархічна структура – це графічне представлення проблеми у вигляді перевернутого дерева, де кожен елемент, за винятком самого верхнього, залежить від одного або більше вище розташованих елементів. Ієрархічні структури, які використовують в МАІ, являють собою інструмент для якісного моделювання складних проблем [10]¹⁾.

Вершиною ієрархії є головна мета; елементи нижнього рівня представляють безліч варіантів досягнення мети (альтернатив); елементи проміжних рівнів відповідають критеріям або факторів, які пов'язують ціль з альтерна-

¹⁾ [10] Подлиновский В.В. Введение в теорию важности критериев в многокритериальных задачах принятия решений. М.: Физматлит, 2007. 64 с.

тивами. Після побудови ієрархії учасники процесу використовують МАІ для визначення пріоритетів всіх вузлів структури. Інформація для розстановки пріоритетів збирається з усіх учасників і математично обробляється.

Отже, для здійснення вибору протоколу реалізації веб-сервісу підтримки діяльності агенції нерухомості потрібно порівняти три протоколи SOAP, XML-RPC і REST за такими критеріями:

- простота та легкість реалізації;
- незалежність від транспорту передачі
- безпека.

Перший крок процедури МАІ полягає в попарному порівнянні протоколів за кожним критерієм. Для цього використовуємо стандартну шкалу порівняння, наведену у табл. 2.1.

Таблиця 2.1 – Шкала порівняння

Рейтинг	Опис
1	Однакова перевага
3	Помірна перевага
5	Явна перевага
7	Очевидна перевага
9	Виняткова перевага

Можливо привласнювати значення рейтингу 2, 4, 6 і 8, які визначаються як середні від найближчих рейтингів.

Розпочнемо дослідження з першого критерію – простота, і як наслідок, легкість реалізації. Таблицю слід читати таким чином: вказаний в рядку протокол порівнюється з протоколом, вказаним в стовпці. Якщо вказаний в рядку протокол прийнятніший, то відповідне число від 1 до 9 записується в чарунку на перетині рядка і стовпця. Якщо ж прийнятніший протокол, вказаний в стовпці, то 1 ділиться на відповідне число від 1 до 9, і результат записується в чарунку на перетині рядка і стовпця. Очевидно, що оскільки будь-який

протокол однаково переважає в порівнянні з самим собою, то в усі діагональні чарунки заноситься значення 1. По показнику «простота та легкість реалізації» протоколу 1 віддається середнє між помірним і протокол 1 має явну перевагу порівняно з протоколом 2, тому в чарунку другого стовпця першого рядка заноситься число 4. Протоколу 3 (XML-RPC) віддається перевага від однакового до помірного перед протоколом 1 (REST), тому в чарунку третього стовпця першого рядка записано число 1/2 (табл. 2.2).

Таблиця 2.2 – Попарне порівняння за критерієм «простота та легкість реалізації»

	SOAP	REST	XML-RPC
SOAP	1	4	0,5
REST	0,25	1	0,14
XML-RPC	2	7	1
<i>Сумма</i>	3,25	12	1,64

Після виконання усіх попарних порівнянь матрицю необхідно нормалізувати. Це виконується шляхом підсумовування чисел в кожному стовпці і подальшого ділення кожного елемента стовпця на отриману для цього стовпця суму. Результати цієї операції представлені в табл. 2.3.

Таблиця 2.3 – Нормалізована матриця для критерію простота та легкість реалізації

	SOAP	REST	XML-RPC	Среднее
SOAP	0,308	0,33333	0,304348	0,315124
REST	0,077	0,08333	0,086957	0,082404
XML-RPC	0,615	0,58333	0,608696	0,602471

Наступний крок полягає в обчисленні балу для кожного протоколу за критерієм простота та легкість реалізації. Завершивши нормалізацію матриці, необхідно вчислити коефіцієнт узгодженості і перевірити його значення.

Мета цієї операції полягає в тому щоб переконатися в узгодженості завдання переваг в вихідній таблиці. Наприклад, якщо за критерієм загальної вартості задана явна перевага протоколу 1 перед протоколом 2 і помірна перевага протоколу 2 в порівнянні з протоколом 3, то при порівнянні протоколів 1 і 3 завдання однакової переваги приведе до неузгодженості, ще більша неузгодженість виникне при вказівці, що 3 краще 1. Обчислення коефіцієнта узгодженості складається з трьох етапів:

- обчислюється міра узгодженості для кожної архітектури;
- визначається індекс узгодженості ІС;
- обчислюється коефіцієнт узгодженості як відношення ІС/ІР, де ІР – індекс рандомізації.

Для обчислення міри узгодженості можна скористатися функцією множення матриць у табличному редакторі Excel. Як показано на рис. 2.1, для протоколу 1 (SOAP) середній рейтинг кожного протоколу множиться на відповідну кількість балів в першому рядку ці добутки підсумовуються, і сума ділиться на середній рейтинг першого протоколу .

	SOAP	REST	XML-RPC	Среднее	Мера согласованности
SOAP	0,307692	0,333333	0,304348	0,315124	3,001916
REST	0,076923	0,083333	0,086957	0,082404	3,000483
XML-RPC	0,615385	0,583333	0,608696	0,602471	3,003547
	Индекс согласованности:			0,000991	
	Индекс рандомизации:			0,58	
	Козфициент согласованности:			0,001709	

Рисунок 2.1 – Коефіцієнт узгодженості для критерію простота та легкість реалізації

Аналогічні обчислення здійснюються для 2 і 3 протоколів. У ідеальному випадку міри узгодженості мають дорівнювати числу можливих альтернативних рішень (у нашому випадку є 3 рішення, тобто 3 протоколи). Для обчислення індексу узгодженості визначається середня міра узгодженості усіх трьох протоколів, з неї віднімається кількість можливих варіантів рішення p і результат ділиться на $p-1$. Індекс узгодженості ІС показаний на рис. 2.2 в чарунці, і його значення дорівнює 0,001. Останній етап визначення коефіцієнта узгодженості полягає в діленні ІС на індекс рандомізації ІР, який в свою чергу дорівнюють: 0,00; 0,58; 0,90; 1,12; 1,24; 1,32; 1,41; 1,45; 1,51, значення якого для різних значень обчислюються в методі. Коефіцієнт узгодженості дорівнює 0,002.

	SOAP	REST	XML-RPC			
SOAP	1	0,5	6			
REST	2	1	8			
XML-REST	0,166667	0,125	1			
Сумма	3,166667	1,625	15			
Нормалізація:						
	SOAP	REST	XML-RPC	Среднее	Мера согласованности	
SOAP	0,315789	0,307692	0,4	0,341161	3,020042	
REST	0,631579	0,615385	0,533333	0,593432	3,031534	
XML-RPC	0,052632	0,076923	0,066667	0,065407	3,003439	
Індекс согласованности:				0,009169		
Індекс рандомізації:				0,58		
Коефіцієнт согласованности:				0,015809		

Рисунок 2.2 – Коефіцієнт узгодженості за критерієм «незалежність транспорту передачі»

В разі абсолютної узгодженості переваг міра узгодженості дорівнюватиме 3, отже, ІС дорівнюватимуть нулю, і коефіцієнт узгодженості також дорівнюватиме нулю. Якщо цей коефіцієнт дуже великий (більше 0,10 за оцін-

кою Сааті), то експерт був недостатньо послідовний в своїх оцінках, тому слід повернутися назад і переглянути результати попарних порівнянь (в більшості випадків виявляється елементарна помилка, коефіцієнт узгодженості сигналізує про її наявність. Тепер необхідно виконати те ж саме для останніх трьох критеріїв. У всіх випадках значення коефіцієнта узгодженості поміщені в межах від 0 до 0,047, це означає, що ми були досить послідовні в своїх оцінках (рис. 2.3).

	SOAP	REST	XML-RPC			
SOAP	1	1	5			
REST	1	1	5			
XML-RPC	0,2	0,2	1			
Сумма	2,2	2,2	11			
Нормалізація:						
	SOAP	REST	XML-RPC	Среднее	Мера согласованности	
SOAP	0,454545	0,454545	0,454545	0,454545	3	
REST	0,454545	0,454545	0,454545	0,454545	3	
XML-RPC	0,090909	0,090909	0,090909	0,090909	3	
Индекс согласованности:				0		
Индекс рандомизации:				0,58		
Козфициент согласованности:				0		

Рисунок 2.3 – Коефіцієнт узгодженості за критерієм безпека

На цьому перший етап роботи закінчується. На другому етапі здійснюються аналогічні попарні порівняння для визначення ваги критеріїв. Процес аналогічний попередньому в тому, що знову виконується порівняння, проте тепер порівнюються не протоколи, як це було на етапі 1, а критерії. При підрахунках виявилось, що показник складності і надійності математичних алгоритмів має найбільшу вагу (52,5%), за ним йде вартість (30,4%). Приємно,

що заходи узгодженості виявилися близькі до 4, тому індекс узгодженості і коефіцієнт узгодженості близькі до нуля.

Останній крок полягає в обчисленні зважених середніх оцінок для кожного варіанту рішення і застосуванні отриманих результатів для ухвалення рішення про те, який з протоколів обрати для здійснення проектування та реалізації хмарного сервісу підтримки діяльності агенції нерухомості.

На підставі отриманих результатів можна зробити висновок, що протокол SOAP (показник 0,378) дещо перевершує технологію REST (0,376), а протокол XML-RPC від них помітно відстав. Отже, здійснювати програмну реалізацію web-сервісу для підтримки діяльності агенції нерухомості будемо використовуючи протокол SOAP. Перевагами застосування протоколу SOAP для реалізації web-сервісу можливо відзначити наступні: підтримка формату XML, можливість роботи з безліччю протоколів (а не тільки HTTP), працює з операціями, які реалізують будь-яку бізнес-логіку за допомогою декількох інтерфейсів, підтримка WS-security.

Для здійснення проектування та розробки web-сервісу підтримки діяльності агенції нерухомості було проведено аналіз та вибір протоколу SOAP, використовуючи математичний апарат теорії прийняття рішень, а саме метод аналізу ієрархій.

2.2 Особливості застосування та вимоги до web-сервісів

За своєю суттю web-сервіс не має графічного інтерфейсу, який би зміг оцінити простий користувач цього сервісу, так як web-сервіс оперує лише інформацією у форматі XML. Фактично клієнтом web-служби виступає додаток, а не людина. З іншої сторони, web-сервіс має бути побудований таким чином, щоб кожен web-застосування зміг по своєму визначати дизайн системи, яка використовує даний web-сервіс. Тобто стилізація та розмітка даних, що повертаються web-сервісом до додатку ні в якому разі не повинна мати місце. Тому для того, щоб були виконані ці вимоги потрібно чітко визначити

формат даних, що повертаються. Наступною вимогою до web-сервісу є те, що всі мережеві зв'язки між сервісом та додатком повинні забезпечуватись з допомогою глобальної мережі Internet. Тому повинен використовуватись протокол передачі гіпертексту HTTP [3]¹⁾.

Також у WSDL-описі web-сервісу не повинні існувати двунправлені циклічні посилання між файлами WSDL1 та WSDL2, тому, що це призведе до поганих наслідків. Наступною, однією з найголовніших вимог, є перехоплення помилок, та їх опрацювання. Це дуже важливою вимогою у програмуванні в цілому. Ще однією важливою вимогою до web-сервісу є документування методів та функцій web-сервісу. Це потрібно для того, щоб людина-розробник програмного забезпечення, при використанні даного web-сервісу могла з легкістю зрозуміти призначення того чи іншого методу або функції. Тобто всі методи та функції повинні прокоментовані у форматі JavaDOC, для подальшого оформлення документації [8]²⁾.

Що стосується web-ресурсу, який буде використовувати інформацію web-сервісу, то він повинен відповідати всім сучасним стандартам та нормам. Якісний web-ресурс, на технічному рівні, – це гарний зміст, хороша структура, приємний дизайн. Хороший сайт повинен швидко завантажуватися, не напружуючи відвідувача. Він повинен мати просту і прозору структуру, в якій легко орієнтуватися.

Інформаційна система, що знаходиться в глобальній мережі Internet, та орієнтована на інформаційну підтримку діяльності агенції нерухомості повинна приносити прибуток. Тобто клієнти web-сервісу підтримки діяльності агенції нерухомості при додаванні оголошення купівлі або продажу об'єктів нерухомості повинні заплатити визначений комісійний збір.

¹⁾ [3] Хмарні технології. Переваги і недоліки. Валтек системний інтегратор – ІТ послуги системної інтеграції в Україні. URL: <https://valtek.com.ua/ua/system-integration/it-infrastructure/clouds/cloud-technologies> (дата звернення: 10.07.2019).

²⁾ [8] UBS. Публикации о технологиях веб-сервисов. URL: http://www.ubs.ru/ws/ws_soad1.html (дата звернення: 26.09.2019).

2.3 Визначення бізнес-процесів хмарного сервісу

Бізнес-процес – це сукупність взаємопов'язаних заходів або завдань, спрямованих на створення певного продукту або послуги для споживачів. Існують три види бізнес-процесів:

- управляючі – бізнес-процеси, які управляють функціонуванням системи;
- операційні – бізнес-процеси, які складають основний бізнес компанії і створюють основний потік доходів;
- підтримуючі – бізнес-процеси, які обслуговують основний бізнес.

У предметній області web-сервісу агенції нерухомості підтримуються операційні бізнес-процеси, спрямовані на збільшення доходу власника web-сервісу об'єктів нерухомості. Розрізняють також основні і допоміжні процеси. Основні процеси – це ті, які додають якість, допоміжні процеси формують інфраструктуру організації. Основним таким процесом є розміщення оголошень, а також розміщення оголошень на головній сторінці, а допоміжними бізнес-процесами є ведення бази даних.

Web-сервіс, що проектується підтримує такі бізнес-процеси:

- надання інформації про продаж нерухомості;
- розміщення оголошень про продаж нерухомості;
- надання інформації про останні новини з ринку нерухомості;
- надання адміністратору web-сервісу можливості додавати, видаляти та редагувати всю інформацію в базі даних з підсистеми адміністратора.

2.4 Вибір системи керування базами даних

Для розробки web-сервісу підтримки діяльності агенції нерухомості доцільно використовувати СУБД середнього масштабу і продуктивності. На

сьогоднішній день СУБД MySQL є однією з найвідоміших, надійних і швидких з усього сімейства існуючих СУБД. Вона найкраще підходить для проєктів, що публікуються в мережі Internet. СУБД MySQL написана під десятки видів операційних систем, починаючи від UNIX-сумісних операційних систем та закінчуючи сімейством операційних систем Windows. Принцип роботи СУБД MySQL аналогічний принципу роботи будь-якої реаліційної СУБД, що використовує SQL (Structured Query Language – мова структурованих запитів), як командної мови для створення чи заповнення баз даних, таблиць, для поповнення таблиць даними, для здійснення вибірки даних. MySQL, як і будь-яка інша СУБД являє собою програму-сервер, яка знаходиться в пам'яті комп'ютера і обслуговує TCP порт. У випадку з MySQL, номером порту буде число 3306, а клієнтська програма, будь то CGI-додаток або програмний продукт на C, з'єднується з СУБД з цього порту і посилає йому рядки на SQL. Таким способом відбувається спілкування сервера баз даних з клієнтськими програмами. Для запуску MySQL-сервера необхідно виконати файл `mysqld.exe`, а сервер запускається як, безвіконний фоновий процес. При цьому він залишається в пам'яті і обробляє запити від клієнтських додатків. Для зупинки сервера слід виконати команду `mysqladmin – u root shutdown`. Якщо сервер не був зупинений коректно, то при наступному запуску у файлі `mysql.err` буде додано запис про збої. Коректна зупинка сервера необхідна для збереження всіх даних, які знаходяться в кешах MySQL. MySQL має розвинену систему доступу до баз даних. Користувачу бази даних може бути наданий доступ до всієї бази даних, окремих таблиць і окремих стовпців таблиць, а також є розмежування на дії, які може виконувати користувач із записами [11]¹⁾.

Для організації такої, складною структури доступу використовується декілька таблиць в системній базі даних. На підставі значень цих таблиць налаштовується політика надання доступу. База даних, яку сервер MySQL ви-

¹⁾ [11] Люк Веллінг, Лора Томсон. Разработка веб-приложений с помощью PHP и MySQL. М.: Вильямс, 2010. 848 с.

користовує для зберігання внутрішньої інформації про користувачів, за замовчуванням має ім'я `mysql`. У цій базі даних використовуються певні таблиці для зберігання інформації облікових записів користувачів.

Для реалізації всього функціоналу хмарного сервісу підтримки діяльності агенції нерухомості необхідно на першому етапі здійснити проектування бази даних, що забезпечить зберігання всієї необхідної інформації та доступу до даних.

2.5 Вибір мови програмування

Для здійснення програмної реалізації `web-сервісу` підтримки діяльності агенції нерухомості була обрана об'єктно-орієнтована мова програмування `PHP`. Такий вибір обґрунтовується тим, що мова `PHP` має дуже зручні характеристики для реалізації поставленої задачі. Тобто для реалізації сервіс-орієнтованої системи на базі `SOAP` в `PHP` існує `DLL`-бібліотека, яка містить у собі важливі класи та методи, які допомагають у розробці і об'єднують розробку. Ефективність є винятково важливим фактором при програмуванні для багатокористувацьких середовищ, до числа яких належить і `WWW`.

У `PHP` реалізований механізм виділення ресурсів і забезпечена поліпшена підтримка об'єктно-орієнтованого програмування, засобів управління сеансом і механізмом підрахунку посилань, що запобігає виділенню зайвої пам'яті. Ще одна перевага полягає в тому, що вихідний текст сценаріїв `PHP` не можна переглянути в браузері, оскільки сценарій компілюється до його відправлення за запитом користувача. Немає проблем і з залежністю від браузерів, оскільки перед відправкою клієнту сценаріїв `PHP` повністю інтерпретуються на стороні сервера. Основні переваги застосування мови `PHP` для програмування це доступність і простота. `PHP` має ряд особливостей, які добре працюють разом, вони включають збір інформації, динамічне введення і низький рівень абстракції, що робить доступнішою дану мову. У тому числі доступністю дешевих і широко доступних хостинг-провайдерів, що є важливим

чинником, який сприяє поширенню відкритого вихідного коду PHP. PHP це особлива мова, створена спеціально, щоб використовувати її для реалізації веб-інтерфейсів Інтернет-систем. Відсутність розгортання або необхідності компілювати робить швидкі цикли зворотного зв'язку і легкі шляхи розвитку. Велике поширення і застосування означає підтримку і доступну документацію. PHP обробляється на стороні сервера і є скриптовою мовою, що вбудується до HTML-документу, це означає, що при виборі PHP в якості мови реалізації, вона дозволяє динамічно, швидко і легко генерувати web-сторінки. Крім того, вона доступна для більшості операційних систем і веб-серверів, а також легко співпрацює з базами даних, включаючи MySQL. Оскільки PHP не містить коду, орієнтованого на конкретний web-сервер, користувачі не обмежуються певними серверами. Apache, Microsoft IIS, Netscape Enterprise Server, Stronghold і Zeus – PHP працює на всіх перерахованих серверах. Оскільки ці сервери працюють на різних платформах, PHP в цілому є платформо-незалежною мовою й існує на таких платформах, як UNIX, Solaris, FreeBSD і Windows [11]¹⁾.

¹⁾ [11] Люк Веллинг, Лора Томсон. Разработка веб-приложений с помощью PHP и MySQL. М.: Вильямс, 2010. 848 с.

3 ПРОЕКТУВАННЯ WEB-SERVISY АГЕНЦІЇ НЕРУХОМОСТІ

3.1 Функціональні можливості користувачів сервісу

Визначення функціональних можливостей користувачів є важливою складовою розробки будь-якої системи. В web-сервісі підтримки діяльності агенції нерухомості передбачено кілька категорій користувачів, кожна з яких визначає вимоги по відношенню до додатка бази даних, тобто визначає, які дії і над якими даними повинен виконувати той чи інший користувач. Набір функцій тієї чи іншої категорії користувачів може відноситися тільки до даної категорії або частково збігатися з набором функцій інших категорій користувачів. Для пошукового web-сервісу об'єктів нерухомості було передбачено два типи користувачів: адміністратор та клієнт, який в свою чергу має своїх користувачів.

В якості клієнта виступає web-додаток, в який вбудовується частини сервіс-орієнтованої системи. За допомогою WSDL-опису та документації методів і функцій пошукового web-сервісу об'єктів нерухомості клієнт має можливість і всі дані для того, щоб без проблем вбудовувати частину функціоналу веб-сервісу у свій web-додаток, тобто розміщати оголошення та новини, у своїй системі .

Користувач даної web-системи – це звичайний користувач мережі Internet, який завантажив сторінку клієнта у своєму браузері. Для користувача web-ресурсу, який він відвідує, немає значення звідки беруться ті чи інші дані, так як він переглядає цілісну web-сторінку без яких небуť змін, в той час як на сервер був посланий запит на збір та представлення інформації. Також клієнт може розмістити своє оголошення, звернувшись до адміністратора, та залишивши заявку на оголошення. Після проходження модерації оголошення клієнта розміщується у web-системі.

Інший тип користувачів системи – це адміністратор web-сервісу підтримки діяльності агенції нерухомості. Він ідентифікується системою за допомогою унікального ім'я та підтверджує свою особистість за допомогою па-

ролю. Адміністратор володіє наступними правами: авторизація та аутентифікація у системі; додавання, редагування та видалення оголошень, новин; додавання нових користувачів, адміністраторів у систему з іншим набором привілеїв.

3.2 Моделювання бізнес-процесів пошукового web-сервісу

Для здійснення розробки хмарного сервісу агенції нерухомості необхідно виконати моделювання бізнес-процесів. В даний час для моделювання бізнес-процесів застосовується велика кількість різних методологій, мов опису бізнес-діяльності організації. Технології SADT – методологія структурного аналізу і проектування (Structured Analysis and Design Technique), IDEF0 – методологія функціонального моделювання, DFD – методологія моделювання потоків даних, IDEF3 – методологія моделювання потоків робіт, IDEF1X – методологія опису даних, IDEF4 – об'єктно-орієнтована методологія, яка відображає взаємодію об'єктів, UML (Unified Modeling Language) – мова візуального моделювання, заснована на об'єктно-орієнтованому підході [12]¹⁾.

Для проектування web-сервісу агенції нерухомості була обрана методологія функціонального моделювання SADT (Structured Analysis and Design Technique) – стандарт IDEF0. Для найзагальнішого опису системи була розроблена контекстна діаграма. Яка описує взаємодію сервісу з зовнішнім середовищем. Після того, як була створена та описана головна робота пошукового web-сервісу об'єктів нерухомості її розбивають на більш менші частини, тобто отримують її декомпозицію. Цей процес називається функціональна декомпозиція, а діаграми, які описують кожен фрагмент і взаємодію фрагментів, називаються діаграмами декомпозиції.

Після декомпозиції головної роботи (контекстної діаграми) проводиться подальша декомпозиція дрібних блоків. Цей процес продовжується до тих

¹⁾ [12] Карпова Т.С. Базы данных: модели, разработка, реализация. СПб.: Питер, 2001. 304 с.

пір доки не буде досягнуто найбільш потрібного рівня деталізації системи. Декомпозиція дрібних блоків дозволяє більш детально описати функції системи. Після кожного сеансу декомпозиції проводяться сеанси експертизи предметної області. Знайдені невідповідності виправляються, і тільки після проходження експертизи без зауважень можна приступати до наступного сеансу декомпозиції. Так досягається відповідність моделі реальним процесам на кожному рівні декомпозиції моделі. Контекстна діаграма web-сервісу підтримки діяльності агенції нерухомості представлена на рис. 3.1.

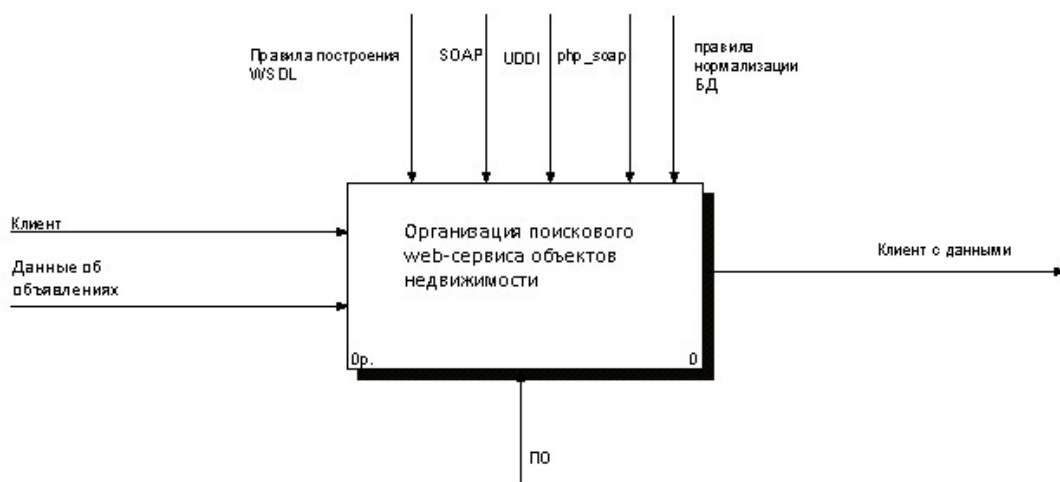


Рисунок 3.1 – Контекстна діаграма головної роботи «Організація web-сервісу об'єктів нерухомості»

На даній контекстній діаграмі відображена головна робота сервіс-орієнтованої системи «Організація пошукового web-сервісу об'єктів нерухомості». Входом головної роботи є клієнт, який хоче отримати інформацію про оголошення об'єктів нерухомості і для цього робить запит до web-сервісу та дані про продаж нерухомості, які надходять у базу даних web-сервісу. Ця визначена робота керується правилами створення WSDL-документу, протоколом SOAP, протоколом UDDI, правилами нормалізації бази даних та можливостями бібліотеки php_soap. Виходом є клієнт з даними, які він хотів отримати згідно з обраними критеріями пошуку. Механіз-

мом, тобто тим, за допомогою чого створюється та функціонує вся система є програмне забезпечення. Після декомпозиції головної роботи системи отримуємо більш детальний опис роботи. Декомпозиція другого рівня містить у собі три роботи: розробка web-сервісу, розміщення web-сервісу в каталогах та запуск web-сервісу. Перший блок-робота «Розробка web-сервісу» включає в себе розробку бази даних, розробку сервера web-сервісу, а також розробку WSDL-документа. Входом цієї роботи є дані про продаж нерухомості, а виходом є готовий WSDL-документ. Паралельно робота керується правилами нормалізації бази даних, так як розробка та нормалізація бази даних входить до складу цієї роботи, протоколом SOAP та правилами побудови WSDL-документу. Наступною роботою за розробкою web-сервісу являється «Розміщення web-сервісу в каталогах», забезпечує розміщення web-сервісу у каталогах, для того, щоб сервісом могли користуватись клієнти. Наступна, не менш важлива робота це «Запуск web-сервісу», яка націлена на запуск, а також його тестування та налагодження для подальшої безперебійної роботи. Діаграма декомпозиції представлена на рис. 3.2.

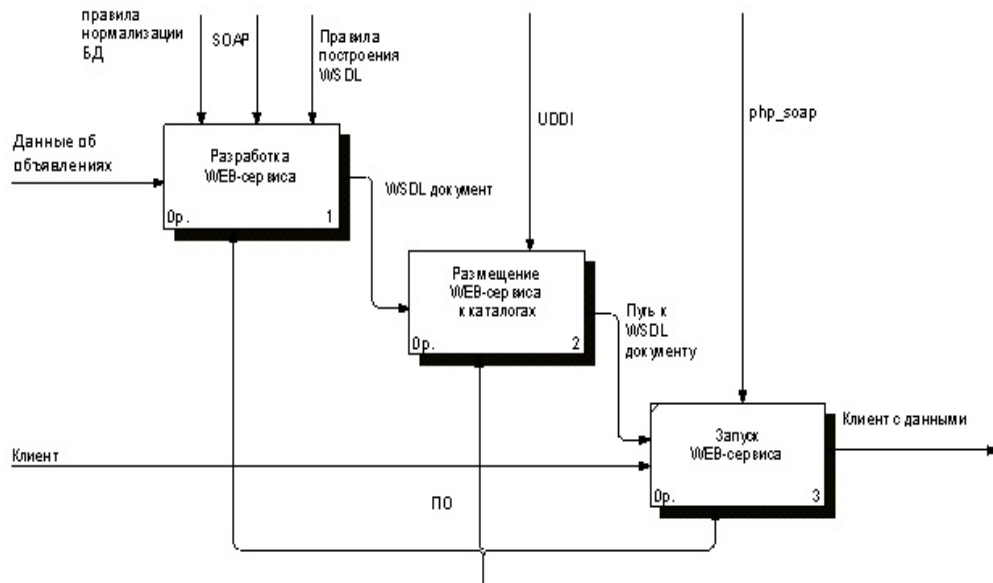


Рисунок 3.2 – Діаграма декомпозиції головної роботи «Організація web-сервісу об'єктів нерухомості»

Після декомпозиції першого блоку другого рівня декомпозиції отримаємо такі роботи: розробка бази даних, розробка сервера та розробка WSDL-документа.

Входом роботи «Розробка бази даних» є дані про продаж об'єктів нерухомості, які будуть зберігатись в базі даних. На виході отримуємо готову базу даних з інформацією, яку потребує клієнт. Даний блок керується правилами нормалізації бази даних, для того, щоб уникнути надмірності у базі.

Наступна робота – «Розробка серверу» отримує на вхід готову базу даних, з якої сервер буде отримувати інформацію. Керується вона протоколом SOAP, а виходом є сервер, що забезпечує доступ до бази даних та обробку запитів від клієнтів. Наступний блок – це робота «Розробка WSDL-документу». Ця робота отримує на вхід працюючий сервер web-сервісу, за яким створюється WSDL-документ. Виходом являється саме WSDL-файл, що описує всі можливості web-сервісу. Керується дана робота правилами створення та синтаксисом WSDL-документу. Діаграма декомпозиції представлена на рис. 3.3.

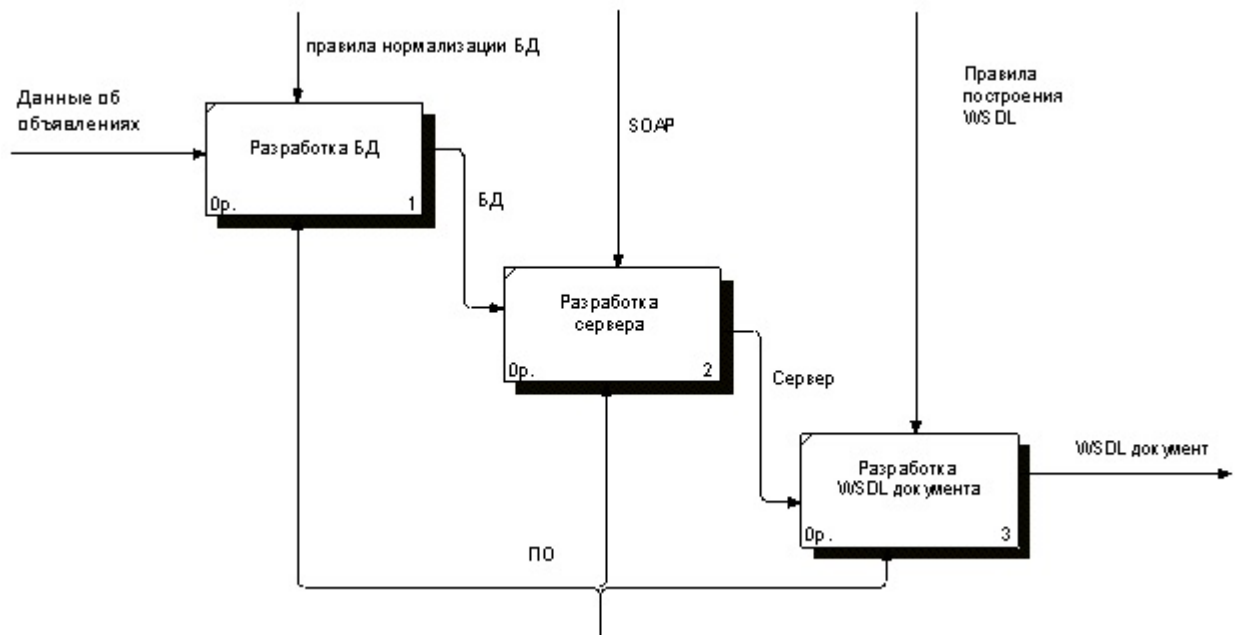


Рисунок 3.3 – Діаграма декомпозиції роботи «Розробка web-сервісу»

Декомпозиція третього рівня роботи «Розміщення web-сервісу» представлена на рис. 3.4.

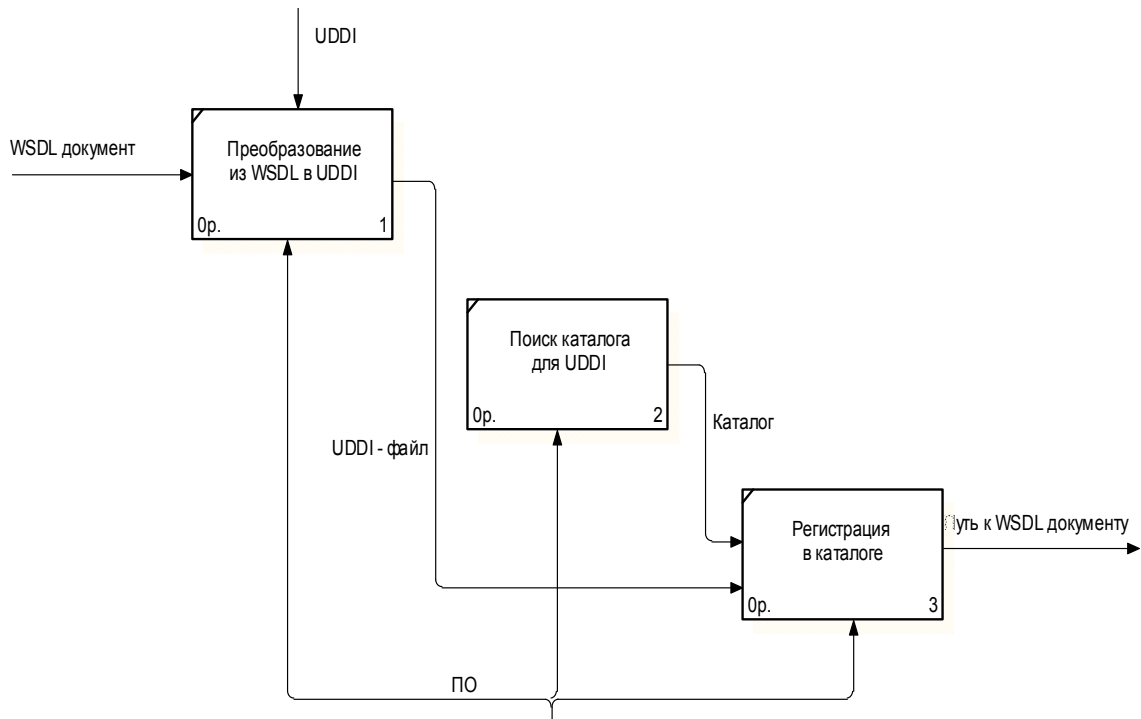


Рисунок 3.4 – Діаграма декомпозиції роботи «Розміщення web-сервісу»

Для виконання розміщення web-сервісу необхідно виконати наступні кроки: виконати представлення WSDL у UDDI, здійснити пошук каталогу UDDI, а також виконати реєстрацію в каталозі. На вході робота отримує WSDL-документ, а після проходження через всі рівні робіт отримуємо шлях до WSDL-документу web-сервісу, що знаходиться в мережі Internet.

3.3 Проектування сервісу за методологією Workflow Diagramming

Контекстна діаграма IDEF3 відображає основну функцію хмарного web-сервісу підтримки діяльності агенції нерухомості для виконання пошуку об'єктів нерухомості. Вона являє собою єдину роботу – «Організація web-сервісу об'єктів нерухомості», ця діаграма наведена на рис. 3.5.

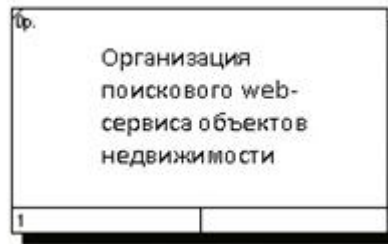


Рисунок 3.5 – Контекстна діаграма сервісу за стандартом IDEF3

Після декомпозиції першого рівня роботи «Організація пошукового web-сервісу об'єктів нерухомості» отримуємо наступні блоки-роботи: «Розробка web-сервісу», «Розміщення web-сервісу», «Запуск web-сервісу». Всі ці три роботи розміщені одна за одною та з'єднані зв'язком «звичайна стрілка», або старший зв'язок. Це означає, що перелічені роботи не можуть виконуватись паралельно, а виконуються лише послідовно, тобто наступна робота починається лише після завершення попередньої. Що зумовлено тим, що наступна робота використовує результат попередньої роботи. Діаграма декомпозиції представлена на рис. 3.6.

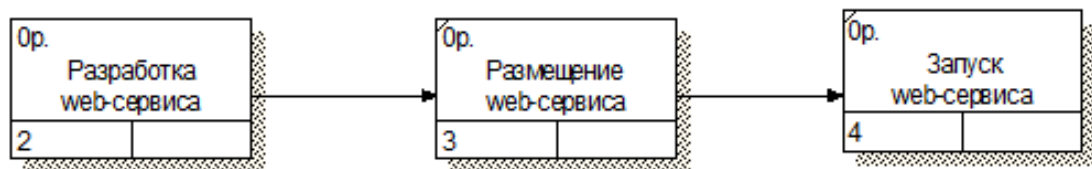


Рисунок 3.6 – Діаграма декомпозиції першого рівня сервісу

При декомпозиції наступного рівня роботи «Розробка web-сервісу» отримуємо наступні роботи: «Розробка бази даних», «Розробка сервера», «Розробка WSDL-документа». Ці роботи, як і попередні зв'язані між собою старшим зв'язком, тобто наступна робота буде виконуватись лише тоді, коли всі попередні будуть завершені. Тому, робота «Розробка сервера» пов'язана з роботою «Розробка бази даних» старшим зв'язком, так як при розробці сер-

вера використовується база даних, яка повинна бути спроектована та розроблена до початку роботи над сервером. Такий самий зв'язок з'єднує роботи «Розробка WSDL-документа» та «Розробка серверу». Так як розробка WSDL-документа може бути виконана тільки після реалізації серверної частини, то звідси випливає те, що робота над сервером повинна завершитись до початку роботи над створенням WSDL-документу. Діаграма декомпозиції блоку «Розробка web-сервісу» представлена на рис. 3.7.

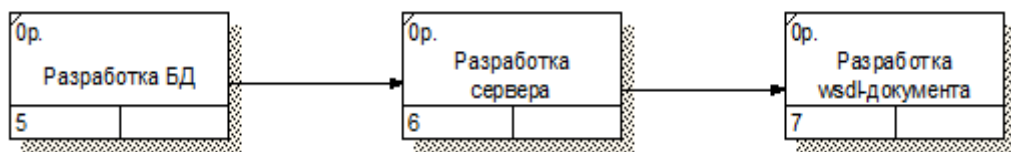


Рисунок 3.7 – Діаграма декомпозиції блоку «Розробка web-сервісу»

При декомпозиції роботи «Розміщення web-сервісу» виділені наступні роботи: «Перетворення WSDL в UDDI», «Пошук каталогу для web-сервісу», «Реєстрація в каталозі». Зв'язок між роботами «Пошук каталогу» та «Перетворення WSDL в UDDI» означає те, що робота-приймач може завершитись ще до початку або закінчення роботи-джерела, а робота «Реєстрація у каталозі» має старший зв'язок, тобто всі попередні роботи повинні завершитись до її початку. Діаграма декомпозиції блоку «Розміщення web-сервісу» наведена на рис. 3.8.

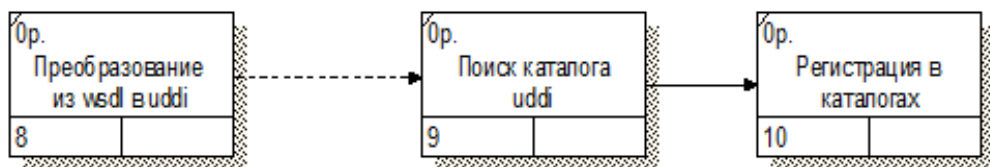


Рисунок 3.8 – Діаграма декомпозиції блоку «Розміщення web-сервісу»

3.4 Проектування сервісу за допомогою діаграм потоків даних DFD

Діаграми потоків даних DFD використовуються для опису руху документів і обробки інформації як додаток до методології функціонального моделювання IDEF0. На відміну від методології IDEF0, стрілки на діаграмах DFD показують лише те, як об'єкти (включаючи дані) рухаються від однієї роботи до іншої. Діаграма потоків даних DFD – це граф, на якому показано рух значень даних від їх джерел через перетворюючі їх процеси до їх споживачів в інших об'єктах. Діаграми найвищого верхнього рівня ієрархії відображає зв'язок пошукового web-сервісу об'єктів нерухомості з зовнішніми сутностями, які взаємодіють з роботою і визначаються зовнішніми входами та виходами. У стандарті DFD також існує декомпозиція контекстної діаграми. Вона проводиться до тих пір, доки не буде досягнутий потрібний рівень декомпозиції, при якому процеси стають елементами і далі деталізувати їх вже не можливо [12]¹⁾. На контекстній даграмі головним процесом сервісу є «Організація пошукового web-сервісу об'єктів нерухомості». Контекстна діаграма представлена на рис. 3.9.



Рисунок 3.9 – Контекстна діаграма сервісу, стандарт DFD

¹⁾ [12] Карпова Т.С. Базы данных: модели, разработка, реализация. – СПб.: Питер, 2001. – 304 с.

Існують також зовнішні сутності. Зі сховища даних в систему передаються дані про продаж об'єктів нерухомості, новини та найвигідніші пропозиції ринку нерухомості. З боку замовника у систему надходить бюджет, який необхідний для створення самого сервісу. До блоку «Каталог» передається WSDL-документ, який являється вхідною точкою до web-сервісу, а клієнт отримує дані, що запросив.

Після декомпозиції блоку головного процесу, що знаходиться на контекстній діаграмі, отримуємо наступні процеси: «Розробка web-сервісу», «Розміщення сервісу в каталозі», а також «Запуск web-сервісу». Перший та другий процеси приймають потік вхідних даних з зовнішньої сутності «Замовник». Це бажання замовника та його бюджет. Також процес «Розробка web-сервісу» приймає потік даних зі сховища даних, а процес «Розміщення web-сервісу» приймає потік з попереднього процесу – WSDL-документ. Після запуску web-сервісу клієнт має можливість отримувати дані, що йому потрібні. Діаграма декомпозиції головного процесу контекстної діаграми представлена на рис. 3.10.

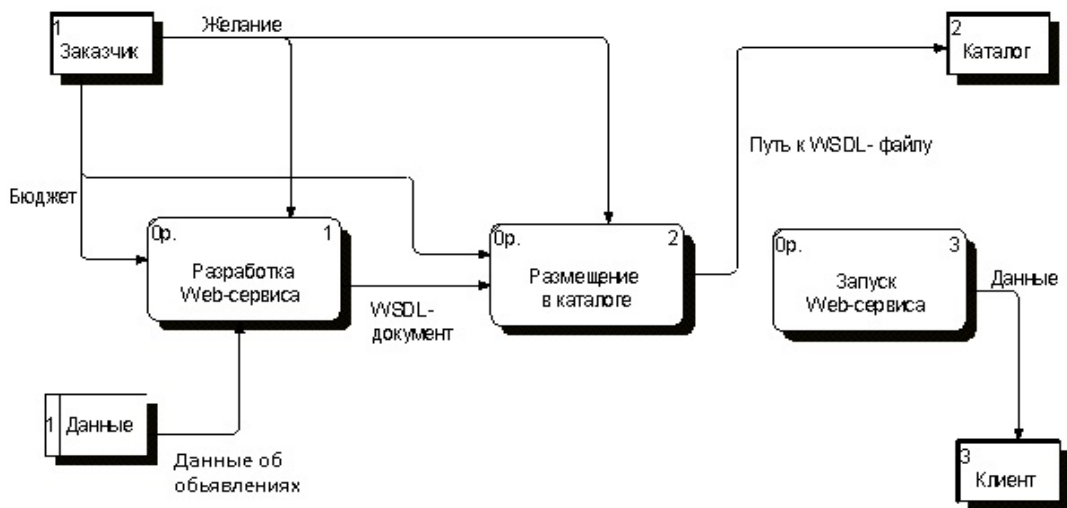


Рисунок 3.10 – Діаграма декомпозиції процесу «Організація пошукового web-сервісу об'єктів нерухомості» за стандартом DFD

3.5 Проектування бази даних

Для представлення логічної структури бази даних хмарного сервісу підтримки діяльності агенції нерухомості була обрана модель представлення даних «сутність-зв'язок». Модель «сутність-зв'язок» призначена спеціально для логічного представлення даних, яка визначає значення даних в контексті їх взаємозв'язку з іншими даними. Сутність є об'єктом, який може бути однозначно ідентифікований серед будь якого набору інших об'єктів. Сутність фактично являє собою безліч атрибутів, які описують властивості всіх членів набору сутностей. Безліч значень атрибуту є доменом. Зв'язок – це асоціація, встановлена між кількома сутностями. Кожна сутність повинна містити атрибут або групу атрибутів, які будуть однозначно ідентифікувати кожен екземпляр сутності. Такий атрибут називають первинним ключем [13]¹⁾.

Виходячи з постановки завдання, а саме для реалізації функціональних можливостей, які має надавати сервіс клієнту, web-сервіс підтримки діяльності агенції нерухомості, повинен містити базу даних, де повинна міститися наступна інформація:

- інформація що до переліку оголошень;
- новини ринку нерухомості;
- тип операції;
- перелік міст;
- перелік грошової валюти;
- перелік регіонів оголошень;
- перелік типів нерухомості;
- перелік розділів нерухомості;
- перелік позицій при виводі оголошень;
- про користувачів, які відвідували оголошення;
- про список адміністраторів web-сервісу об'єктів нерухомості.

¹⁾ [13] Томасон Лаура, Люк Веллинг. Разработка Web-приложений на PHP и MySQL: пер. с англ. 2-е изд. СПб.: ООО ДиаСофтЮП, 2003. 672 с.

Визначимо основні сутності бази даних web-сервісу підтримки діяльності агенції нерухомості: оголошення; новини нерухомості; бронювання позиції оголошення. В базі даних пошукового web-сервісу об'єктів нерухомості необхідно зберігати такі дані:

- про адміністраторів: прізвище, ім'я, логін, пароль, фотографія та IP-адреса останнього візиту;
- про новини: мета-опис, ключові слова, заголовок, короткий опис новини, повний текст новини, дату публікації, зображення, кількість переглядів, регіон, до якої належить новина, а також тип нерухомості, до якого належить новина;
- про оголошення: назва, регіон до якого належить оголошення, місто, ціна нерухомості, валюта, в якій вказана ціна на нерухомість, поверх, на якому розміщено об'єкт нерухомості, максимальна кількість поверхів, житлова площа, загальна площа, тип стін, тип операції з об'єктом нерухомості, посилання на джерело оголошення, позицію при показі на екран, про тип нерухомості, інформація про розділ нерухомості, рік будівництва, дату публікації оголошення, телефон агентства нерухомості, зображення об'єктів нерухомості.

Оскільки в базі даних повинна бути виключена надмірність інформації, а також не допускаються аномалії оновлення, додавання і видалення, необхідно зробити декомпозицію вихідного набору таблиць і привести базу даних до нормальної форми. Нормалізація – це розбиття таблиці на дві або більше, що володіють кращими властивостями при внесенні, зміні і видаленні даних. Остаточна мета нормалізації зводиться до отримання такого проекту бази даних web-сервісу підтримки діяльності агенції з нерухомості, в якому кожен факт з'являється лише в одному місці, тобто виключена надмірність інформації. Це робиться з метою економії пам'яті та для виключення можливої суперечливості збережених даних. Після проведення процедури нормалізації, БД буде приведена до третьої нормальної форми, тобто всі не ключові атрибути не будуть транзитивно залежати від ключових атрибутів сутностей. Для

виключення аномалій оновлення, додавання і видалення, а також надмірності інформації в базі даних web-сервісу агенції нерухомості були виділені наступні сутності:

- сутність «Список користувачів»;
- сутність «Оголошення»;
- сутність «Валюта»;
- сутність «Регіон»;
- сутність «Місто»;
- сутність «Тип операції»;
- сутність «Позиція»;
- сутність «Розділ нерухомості»;
- сутність «Тип нерухомості»;
- сутності «Бронювання»;
- сутність «Новини».

Наступним кроком є визначення всіх атрибутів для сутностей бази даних сервісу підтримки діяльності агенції нерухомості.

Сутність «Регіон» призначена для того, щоб зберігати дані про регіон пошуку оголошень та містить наступні атрибути: ідентифікатор регіону, назва регіону (табл. 3.1).

Таблиця 3.1 – Сутність «Регіон»

№ п/п	Атрибут	Тип
1	id_redion	int(3)
2	name	varchar(50)

Сутність «Список користувачів» призначена для того, щоб зберігати дані про всіх користувачів-адміністраторів, що мають доступ до підсистеми адміністратора та можливість здійснення операцій що до внесення інформації

ції або змін у базу даних. Дана сутність містить атрибути: ідентифікатор користувача, логін, пароль, фото, ім'я, прізвище та IP-адреса (таблиця 3.2).

Таблиця 3.2 – Сутність «Список користувачів»

№ п/п	Атрибут	Тип
1	id_user	int(2)
2	username	varchar(255)
3	userPass	varchar(255)
4	name	varchar(50)
5	surname	varchar(50)
6	img	varchar(255)
7	ip	varchar(15)

Сутність «Валюта» містить наступний перелік атрибутів ідентифікатор валюти, та її назву (табл. 3.3).

Таблиця 3.3 – Сутність «Валюта»

№ п/п	Атрибут	Тип
1	id_cur	int(8)
2	name	varchar(15)

Сутність «Оголошення» зберігає всю інформацію про оголошення щодо об'єктів нерухомості та містить наступні атрибути: ідентифікатор оголошення, назва, ідентифікатор регіону, посилання на сайт оголошення, ціна об'єкту нерухомості, ідентифікатор валюти, поверх, максимальну кількість поверхів, житлову площу, загальну площу, тип стін, ідентифікатор типу операції з нерухомістю, ідентифікатор позиції при виводі оголошення, тип нерухомості, розділ нерухомості, рік будівництва, дату розміщення оголошення, телефон, та зображення об'єктів нерухомості (табл. 3.4).

Таблиця 3.4 – Сутність «Оголошення»

№ п/п	Атрибут	Тип
1	id_annonce	int(5)
2	name	varchar(200)
3	id_reion	int(5)
4	id_city	int(3)
5	price	int(15)
6	currency	int(3)
7	floor	int(5)
8	max_floor	int(3)
9	total_area	int(5)
10	living_space	int(5)
11	type_of_wall	varchar(25)
12	type_operation	int(3)
13	link_of_annonce	varchar(255)
14	position	int(15)
15	section_realty	int(3)
16	type_realty	int(3)
17	description	varchar(255)
18	year_of_construction	date
19	date_of_annonce	date
20	phone	int(15)
21	img	varchar(255)

Сутність «Новини» містить наступний набір атрибутів: ідентифікатор новини, мета-опис новини, ключові слова новини, заголовок, аліас, короткий опис новини, повний текст, дата публікації, зображення, кількість переглядів, чи є новина важливою, ідентифікатор регіону, ідентифікатор типу нерухомості, та поле, яке визначає чи являється дана новина статтею чи ні.

Визначені атрибути цієї сутності, їх типи та наведені у табл. 3.5.

Таблиця 3.5 – Сутність «Новини»

№ п/п	Атрибут	Тип
1	id_news	int(5)
2	meta_d	varchar(255)
3	meta_k	varchar(255)
4	title	varchar(255)
5	alias	varchar(255)
6	description	text
7	full_text	text
8	date	date
9	img	varchar(255)
10	views	int(5)
11	main	tinyint(1)
12	id_region	int(3)
13	id_type_real	int(5)
14	is_article	tinyint(1)

Сутність «Тип нерухомості» містить наступний перелік атрибутів: ідентифікатор типу нерухомості та назву об'єкту нерухомого майна.

Визначені атрибути цієї сутності, їх типи та наведені у табл. 3.6.

Таблиця 3.6 – Сутність «Тип нерухомості»

№ п/п	Атрибут	Тип
1	id_type_realty	int(5)
2	name	varchar(25)

Сутність «Розділ нерухомості» містить наступні атрибути: ідентифікатор розділу нерухомості, назву розділу.

Визначені атрибути цієї сутності, їх типи та наведені у табл. 3.7.

Таблиця 3.7 – Сутність «Розділ нерухомості»

№ п/п	Атрибут	Тип
1	id_sect_real	int(3)
2	name	varchar(25)

Сутність «Бронювання» має наступні атрибути: ідентифікатор бронювання, прізвище, ім'я та по-батькові замовника, e-mail замовника, телефон, дата замовлення, ідентифікатор оголошення та позиції (табл. 3.8).

Таблиця 3.8 – Сутність «Бронювання»

№ п/п	Атрибут	Тип
1	id_bron	int(5)
2	name	varchar(255)
3	email	varchar(255)
4	date	Date
5	id_annonce	int(5)
6	id_pos	int(5)
7	phone	varchar(20)

Сутність «Позиція» призначена для зберігання даних про позиції оголошень, та містить такі атрибути: ідентифікатор позиції, номер (табл. 3.9).

Таблиця 3.9 – Сутність «Позиція»

№ п/п	Атрибут	Тип
1	id_pos	int(10)
2	number	int(20)

Сутність «Місто» призначена для зберігання даних про міста дислокації (знаходження) об'єктів нерухомості, та містить такі атрибути: ідентифікатор міста та його назву(табл. 3.10).

Таблиця 3.10 – Сутність «Місто»

№ п/п	Атрибут	Тип
1	id_city	int(3)
2	name	varchar(45)

Після того, як всі сутності бази даних web-сервісу підтримки діяльності агенції нерухомості були описані, визначені всі атрибути і їх типи потрібно визначити та описати зв'язки між ними.

Між сутностями «Оголошення» та «Регіон» існує зв'язок один-до-багатьох: багато оголошень належать до одного регіону.

Між сутностями «Оголошення» та «Місто» існує зв'язок один-до-багатьох: багато оголошень належать до одного міста. Між сутностями «Оголошення» та «Регіон» існує зв'язок один-до-багатьох: багато оголошень належать до одного регіону.

Між сутностями «Оголошення» та «Позиція» існує зв'язок один-до-одного, тобто одному оголошенню належать одна позиція. Між сутностями «Оголошення» та «Розділ нерухомості» існує зв'язок один-до-багатьох: багато оголошень належать до одного розділу нерухомості.

Між сутностями «Оголошення» та «Тип нерухомості» існує зв'язок один-до-багатьох: багато оголошень належать до одного типу нерухомості.

Між сутностями «Новини» та «Регіон» існує зв'язок один-до-багатьох: багато новин належать до одного регіону.

Між сутностями «Новини» та «Тип нерухомості» існує зв'язок один-до-багатьох: багато новин належать до одного типу нерухомості.

Модель «сутність-зв'язок» бази даних web-сервісу підтримки діяльності агенції нерухомості визначає всі зв'язки між сутностями. Тип зв'язку, що використовується «один-до-багатьох» та «один-до-одного».

Після виконання етапу проектування бази даних хмарного сервісу підтримки діяльності агенції нерухомості, після опису всіх сутностей та їх атрибутів наведемо діаграму «сутність-зв'язок» бази даних системи (рис. 3.11).

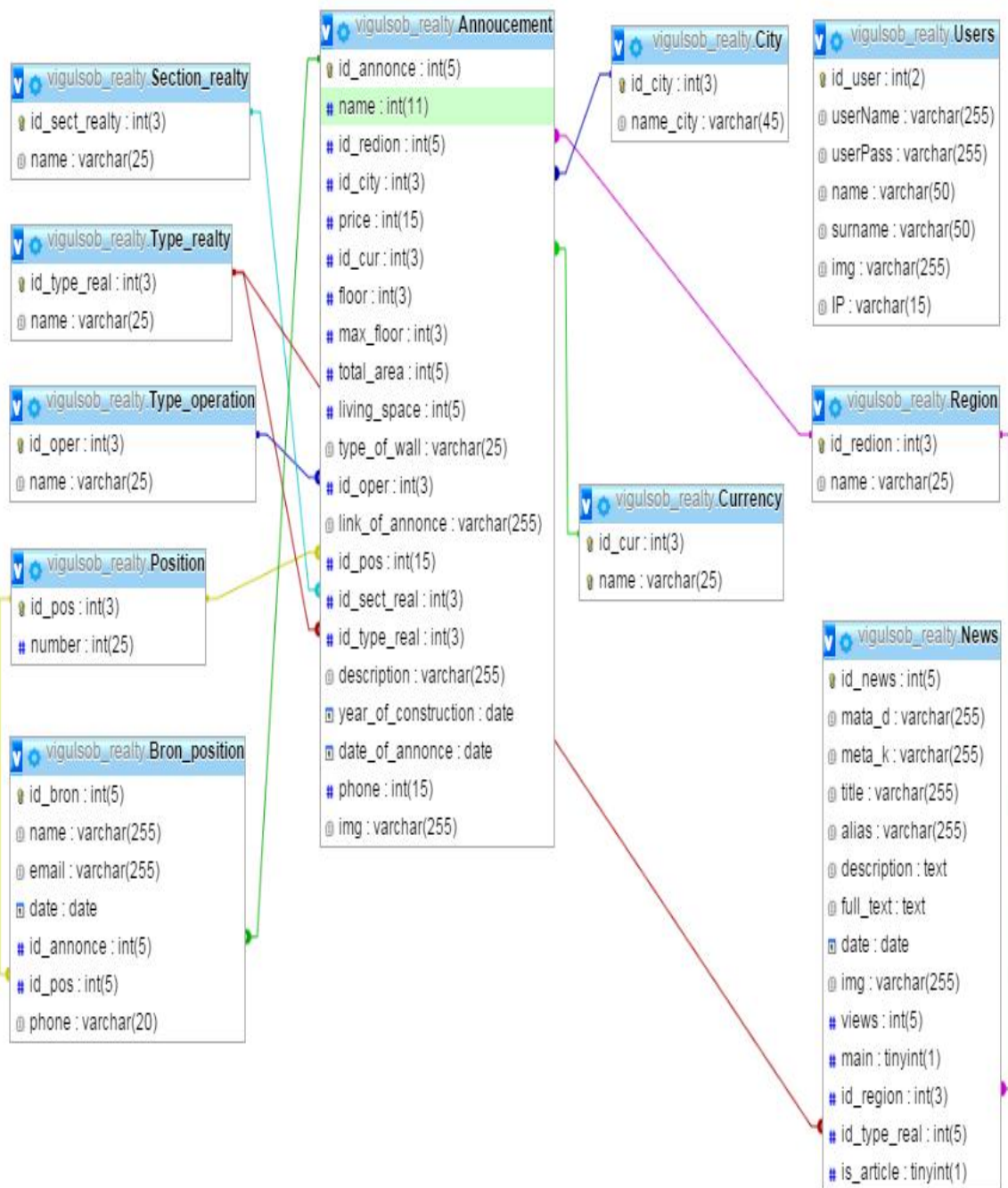


Рисунок 3.11 – Модель «сутність-зв'язок» бази даних сервісу

Для розробки web-сервісу підтримки діяльності агенції нерухомості доцільно використовувати СУБД середнього масштабу і продуктивності, а саме СУБД MySQL, яка є однією з надійних і швидких СУБД.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ WEB-SERВІСУ АГЕНЦІЇ НЕРУХОМОСТІ

4.1 Засоби PHP для розробки web-сервісу SOAP

Для здійснення програмної реалізації web-сервісу підтримки діяльності агенції нерухомості була обрана об'єктно-орієнтована мова програмування PHP, яка забезпечує розробнику доступ до зручних інструментів розробки для реалізації поставленої задачі. Для програмної реалізації web-сервісу агенції нерухомості використана існуюча в PHP DLL-бібліотека, що забезпечує зручні класи та методи, які стають у нагоді розробнику та забезпечують швидку розробку. Для розробки web-сервісу за протоколом SOAP, в PHP існує спеціальний модуль `php_soap`, який має низку необхідних функцій, які налаштовуються в конфігураційному файлі інтерпретатора PHP `php.ini` [14]¹⁾.

Функція `soap.wsdl_cache_enabled = 1` – вмикає/вимикає кешування WSDL-файлу. Для нормальної роботи, на період розробки проекту, потрібно вимикати кешування WSDL-файлу, так як зміни в даному файлі повинні відображатись миттєво. Функція `soap.wsdl_cache_dir = /tmp` – встановлює каталог кешованих файлів, який з часом очищується від кешу, а час життя кешованих файлів в каталозі `tmp` задається іншою командою – `soap.wsdl_cache_ttl = 86400`. Також PHP забезпечує легкий до освоєння та функціональний інтерфейс для роботи з web-сервісами за допомогою розширення модулем SOAP, що дозволяє створювати сценарії, які обмінюються інформацією з іншими додатками за допомогою XML-пакетів [8]. Існує три стандартних класи для роботи з SOAP:

- клас `SoapClient` – створює SOAP-клієнта, який співпрацює з SOAP-сервісом;
- клас `SoapServer` – створює SOAP-сервер, для роботи клієнта;
- клас `SoapFault` – обробляє помилки та виводить про них інформацію.

¹⁾ [14] У. Мерсер, А. Кент, Д. Новицький и другие. PHP 5 для начинающих. М.: Дилектика, 2006. 846 с.

4.2 Створення класу SoapClient

Клас SoapClient створює SOAP-клієнта для web-сервісу підтримки діяльності агенції нерухомості (рис. 4.1). Даний клас виконує наступні функції: створює новий об'єкт класу SoapClient; в параметрі wsdl вказується URI WSDL-документу web-сервісу; всі методи стають методами об'єкту.

```

class SoapClient {
    __construct ( mixed wsdl [, array options] )
}
<?php
    $link = 'http://www.webservice.net/CurrencyConvertor.asmx?wsdl';
    $client = new SoapClient ( $link );
    $usd = $client->ConversionRate ( array ( 'FromCurrency' => 'USD',

    $euro = $client->ConversionRate ( array ( 'FromCurrency' => 'EUR',
    echo 'USD: ' . $usd->ConversionRateResult .
        '<br>EUR: ' . $euro->ConversionRateResult;

?>

```

Рисунок 4.1 – Створення класу SoapClient

Наступним завданням є розробка коду, який повертає список SOAP-функцій, що підтримуються web-сервісом (рис. 4.2).

```

class SoapClient {
    array __getFunctions ( void )
}
<?php
    $client = new SoapClient ( 'some.wsdl' );
    foreach ( $client->__getFunctions () as $type ) {
        echo $type . '<br>';
    }

?>

```

Рисунок 4.2 – Визначення функцій SOAP

В подальшому було розроблено код, який повертає SOAP-запит. Цей код є працездатним при умові, що тільки якщо SOAP-клієнт був створений із увімкненою опцією trace (рис. 4.3).

```
class SoapClient {
    string __getLastRequest ( void )
}
<?php
    $client = SoapClient ("some.wsdl",array('trace' => 1));
    $result = $client->SomeFunction();
    echo "Запрос: " . $client->__getLastRequest();
?>
```

Рисунок 4.3 – Повернення останнього SOAP-запиту

Отримання SOAP-відповіді забезпечує наступний фрагмент коду (рис. 4.4). Працює лише в тому випадку, якщо об'єкт класу був створений з увімкненою опцією trace.

```
class SoapClient {
    string __getLastResponse ( void )
}
<?php
    $client = SoapClient ("some.wsdl",array('trace' => 1));
    $result = $client->SomeFunction();
    echo "Ответ: " . $client->__getLastResponse();
?>
```

Рисунок 4.4 – Повернення останньої SOAP-відповіді

4.2 Створення класу SoapServer

SOAP-сервер створює новий об'єкт класу SoapServer. У параметрі wsdл вказується URI WSDL-документу, що використовується для створення web-сервісу (рис. 4.5).

```
class SoapServer {  
    __construct ( mixed wsdл [, array options] )  
}  
$server = new SoapServer ("stock.wsdл");
```

Рисунок 4.5 – Створення SOAP-серверу

Створення SOAP-серверу забезпечує реєстрацію методів web-сервісу. У параметрі functions вказується ім'я функції, що реєструється або масив, що містить у собі імена декількох функцій, що реєструються. Функція handle() забезпечує виконання запуску процесу обробки web-серверу. У параметрі soap_request вказується SOAP-запит для обробки. Проте, якщо параметр soap_request опущений, то використовується значення задане за замовчуванням у змінній \$HTTP_RAW_POST_DATA (рис. 4.6).

```
class SoapServer {  
    void addFunction (mixed functions)  
    void handle ( [string soap_request] )  
}  
$server->addFunction("getStock");  
$server->handle();
```

Рисунок 4.6 – Додавання методу серверу та обробка запиту

4.3 Результати застосування web-сервісу

Результати роботи створеного web-сервісу підтримки діяльності агенції нерухомості та виконання сервісом всього необхідного функціоналу були протестовані у web-системі, яка була створена для тестування роботи web-сервісу. Вигляд головної сторінки тестової web-системи наведено на рис. 4.7.

The screenshot displays the main interface of a real estate web service. At the top, there is a navigation bar with the logo 'web-сервіс агенції нерухомості' and menu items: 'Продаж житла', 'Оренда житла', 'Новобудови', and a green button '+ Додати оголошення'. A user profile icon labeled 'Увійти' is also present.

The search section features a large orange filter box with the following options:

- Пошук:** Квартири (dropdown)
- Операція:** Продаж (dropdown)
- В області:** Виберіть область (dropdown)
- В місті:** Виберіть місто (dropdown)
- Усі типи:** (dropdown)
- Кімнат:** вид, до
- Загальна:** вид, до
- Ціна:** вид, до
- Тип Валюти:** \$ (dropdown)
- Тільки з фото
- Розширений пошук** (button)
- Пошук** (button)

Below the search filters, there are several property listings:

- ТОП оголошення:**
 - Продаж приміщення вільного призначення 6128 кв. м. **договірна**
 - Продаж квартири, 2-к квартира 33 кв. м. **725 806** грн / 27 000 \$
 - Новобуд: г. Київ, ул. Ясногорська, ЖК Ясногорський. **від 15 000** грн / м²
 - Новобуд: пгт. Гостомель, ул. ЖК Покровський. **від 8 800** грн / м²
 - Продаж будинку, Одеса 2-3т дом 240 кв. м. **1 881 720** грн / 70 000 \$
- Новобудови:** Львов (73) | Київ (331) | Дніпропетровск (30) | Одеса (120)
- Усі новобудови** (link)
- Grid of apartment complexes:
 - ЖК Дома Каркашадзе (г. Одеса, ул. Французский бульвар, 2)
 - Бизнес-центр Нежилые (г. Одеса, р-н Таирово, пр. ... від 8 200 грн/м²)
 - ЖК на Щорса (г. Одеса, ул. Щорса, 148/2 від 11 500 грн/м²)
 - ЖК Левитана (г. Одеса, ул. Левитана, 11... від 13 800 грн/м²)
 - ЖК Радужный 2 (г. Одеса, р-н Таирово, пр. ... від 14 091 грн/м²)
 - ЖК Набережный (г. Одеса, ул. Жаботинског... від 12 000 грн/м²)
 - ЖК Третья Жемчужина (г. Одеса, ул. Архитекторская)
 - ЖК "Акапулько - 2" (г. Одеса, ул. Педагогическ... від 19 900 грн/м²)
- Типи нерухомості:**
 - Квартири
 - Будинки
 - Комерційна
 - Офіси
 - Земельні ділянки
 - Гаражі
- Останні новини нерухомості:**
 - Обустройство ниш: где, как и зачем?
 - Самые красивые и необычные здания мира: Норвежский театр оперы и балета вчера
 - Виды межкомнатных дверей вчера

At the bottom, there is a footer with the text: 'Реклама на сайті © 2019 Всі права захищені'.

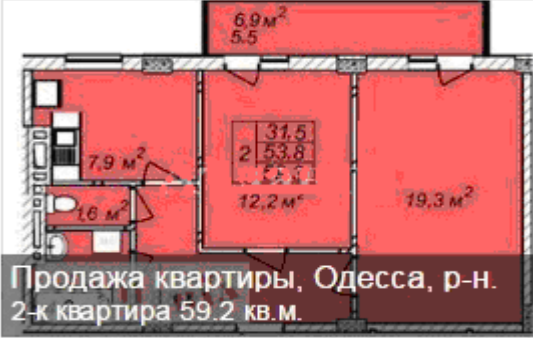
Рисунок 4.7 – Головна сторінка тестової web-системи

Також на сторінки розміщено перелік оголошень та розділ останніх новин ринку об'єктів нерухомості. При виборі оголошення, користувач здійснює перехід на сторінку детальної інформації про об'єкт нерухомості

На сторінці представлена вся інформація, яка завантажується з відповідних таблиць бази даних, а саме назва оголошення, ціна об'єкту нерухомості, контактні данні, кількість кімнат, площа об'єкту, як загальна, так і житлова. Також вказано тип оголошення, тип операції з об'єктом нерухомості, загальна кількість поверхів, тип стін, опис об'єкту та рік в якому був побудований об'єкт.


Ліворуч на сторінки представлений блок «Схожі оголошення», де наведено оголошення, які відносяться до того ж самого регіону або місту або вони схожі по типу нерухомості. Коли хоч одна з трьох вимог виконується, то на сторінці з'являються ці оголошення (рис. 4.9).

Схожі оголошення
Квартира, продаж, Одесса



Продажа квартиры, Одесса, р-н.
2-к квартира 59.2 кв.м.

843 548 грн / 31 380 \$ [Подробнее →](#)



Продажа квартиры, Одесса, р-н.
2-к квартира 40 кв.м.

1 411 290 грн / 52 50... [Подробнее →](#)

Рисунок 4.9 – Сторінка «Схожі оголошення»

Крім того, після сторінки детального перегляду всієї наявної інформації про оголошення, розташований розділ «Вчора ви дивились». Цей розділ зберігає IP зареєстрованого користувача в системі, та при здійсненні переходів по оголошенням, спеціально розроблений скрипт запам'ятовує дії користувача та зберігає історію переглядів, відображаючи незабаром переглянуті оголошення. Вигляд розділу «Ви вчора дивились» представлений на рис. 4.10.

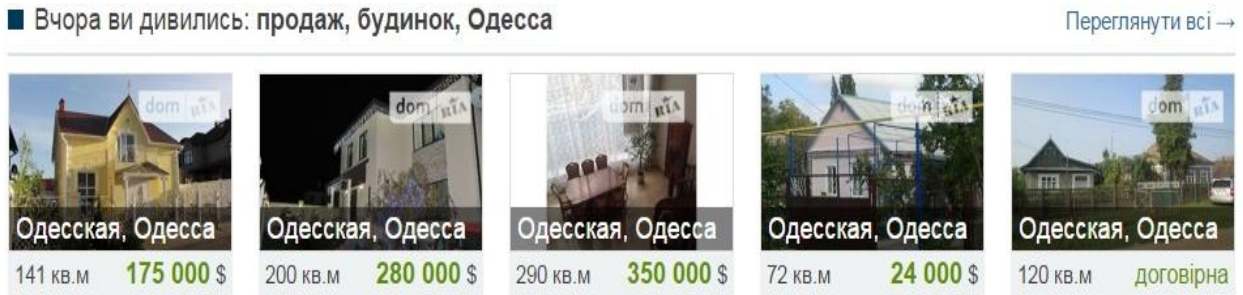


Рисунок 4.10 – Вигляд розділу сторінки «Вчора ви дивились»

На головній сторінці системи наведено розділ «Останні новини нерухомості» (рис. 4.11). В цьому блоці виводяться останні три новини, в залежності від вашого регіону, а якщо для вашого регіону немає новин, то виводяться будь-які три новини.

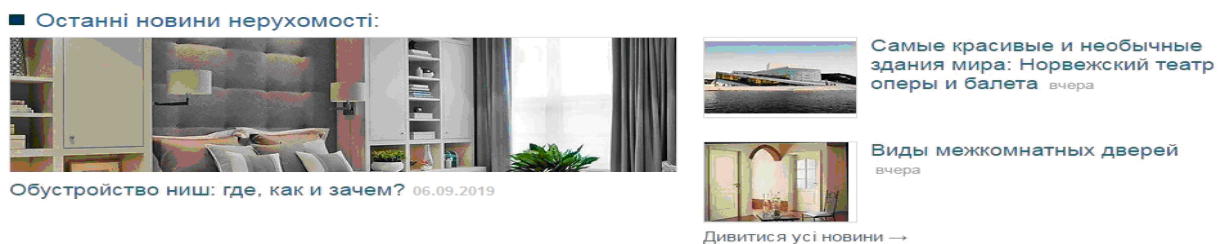
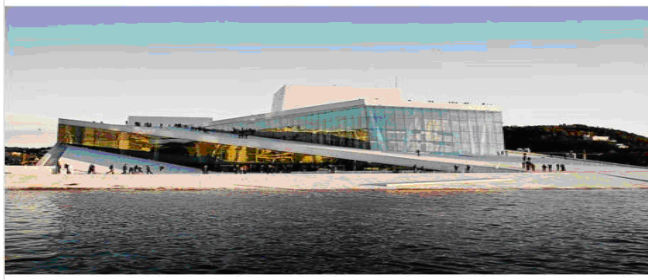


Рисунок 4.11 – Сторінка «Останні новини нерухомості»

На сторінці системи наведено розділ «Новини» (рис. 4.12). В цьому розділі реалізована можливість ознайомитися з більш детальною інформацією про новини ринку нерухомості.

Самые красивые и необычные здания мира: Норвежский театр оперы и балета



Дата публикации: 06.09.2019

Автор: Admin

[Другие новости недвижимости](#)

Ультрасовременное здание театра оперы и балета поражает не только своими белоснежными фасадами и асимметричным дизайном, но и самыми известными постановками и премьерами мира.

Государственный театр оперы и балета в Осло является самым крупным общественным зданием, построенным в Норвегии исключительно на средства государственного бюджета. Идея строительства возникла еще в конце XIX века, однако определение места для строительства, а также открытие тендера на разработку проекта осуществились только в 1999 году. На открытый конкурс было прислано более двухсот работ со всех уголков мира, но победителем стало архитектурное бюро Snøhetta.

Рисунок 4.12 – Сторінка систем из розділом «Новини»

4.4 Додаток адміністратора web-сервісу

Для здійснення управління web-сервісом було реалізовано додаток адміністратора, що дозволяє виконувати налаштування, додавання, видалення, редагування інформації у системі, що зберігається у базі даних. Додаток адміністратора являє собою інтерфейс, що дозволяє виконувати всі зазначені операції користувачем з наданими правами. Для здійснення доступу до панелі адміністратора необхідно здійснити авторизацію у системі, ввести унікальний логін та пароль (рис. 4.13).

Необходима авторизация ✕

Для доступа на сервер `http://webservice:80` нужно указать имя пользователя и пароль.
Сообщение сервера: Admin Page.

Имя пользователя:

Пароль:

Рисунок 4.13 – Форма авторизації у додатку адміністратора

Після введення логіну та паролю в системі відбувається аутентифікація користувача. Якщо ж логін та пароль не співпадають з тими, які існують в базі даних, то користувач отримає відповідне повідомлення про помилку. Після входу в систему користувач-адміністратор бачить сторінку представлену на рис. 4.14.

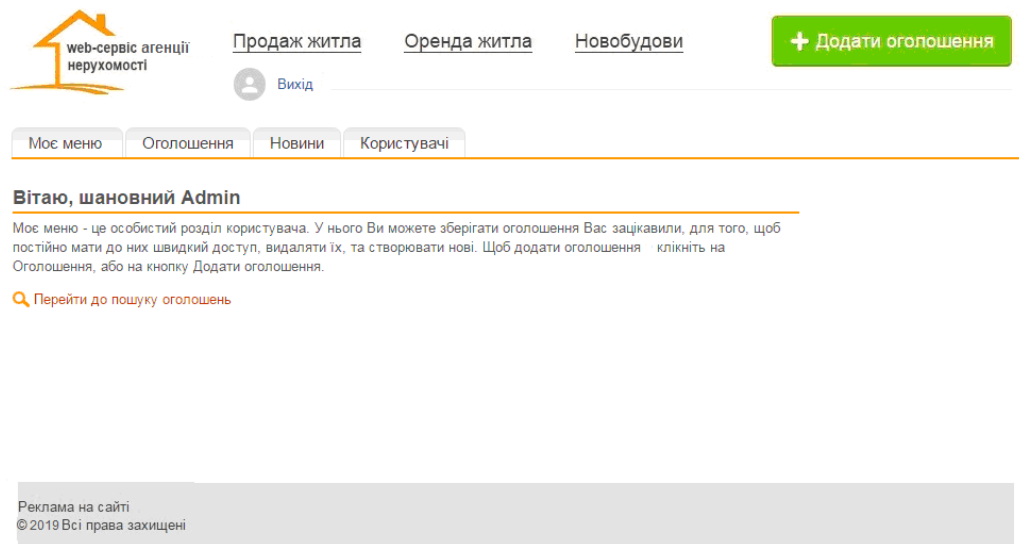


Рисунок 4.14 – Головна сторінка додатку адміністратора

На головній сторінці додатку адміністратора, в верхній її частині знаходиться поле пошуку і панель з наступною інформацією: зображення користувача, ім'я, під яким він здійснив вхід до системи та пункти основного меню: продаж житла, оренда житла, новобудови, додавання оголошення. Далі розташоване наступне горизонтальне меню, яке надає доступ до панелі адміністратора з основними функціями. Меню дозволяє здійснювати управління наступними розділами у системі: новини, оголошення, моє меню, користувачі. При виборі пункту меню «Оголошення», чи на кнопку «Додати оголошення», адміністратор перейде на сторінку додавання оголошень власноруч, що передбачає виконання наступних двох кроків. На першому кроці, адміністратору необхідно заповнити поля форми: категорія, тип нерухомості, тип операції, область, місто, район, вулиця. Поля, які є обов'язковими помічені,

переконавшись у вірності внесеної інформації необхідно натисну кнопку «Далі», що забезпечить перехід до наступного кроку (рис. 4.15).

The screenshot shows the first step of a web form for adding a real estate listing. The header includes the logo 'web-сервіс агенції нерухомості' and navigation links: 'Продаж житла', 'Оренда житла', 'Новобудови', and a green '+ Додати оголошення' button. A 'Вихід' button is also present. Below the header is a navigation bar with 'Моє меню', 'Оголошення', 'Новини', and 'Користувачі'. The main content area is divided into two steps: 'Крок 1 Основна інформація' (highlighted in orange) and 'Крок 2 Додаткова інформація'. The form fields for Step 1 are:

- Категорія:* (dropdown menu with 'квартири' selected)
- Тип нерухомості :* (dropdown menu with 'квартира' selected)
- Тип операції:* (dropdown menu with 'продаж' selected)
- Область:* (dropdown menu with 'Оберіть область' selected)
- Місто:* (dropdown menu with 'Спочатку оберіть область' selected)
- Район: (dropdown menu with 'Спочатку оберіть місто' selected)
- Вулиця: (text input field)

 A note at the bottom right states: 'Всі поля з * обов'язкові для заповненя'. An example address is provided: 'Приклад: Шевченку вул., Перемоги вул., Мичуріна пер.'

Рисунок 4.15 – Перший крок додавання оголошення

Перехід до другого наступного кроку додавання оголошення надає можливість доповнити оголошення більш детальною інформацією: додати фото та опис об'єкта, а потім натиснути «Опублікувати» (рис. 4.16).

The screenshot shows the second step of the web form, titled 'Добавление объявления'. The header and navigation bar are identical to the previous step. The main content area is divided into two steps: 'Крок 1 Основна інформація' and 'Крок 2 Додаткова інформація' (highlighted in orange). The form fields for Step 2 are:

- 'Добавьте фото объекта:' section with a green '+ ' button and instructions: 'Щоб вибрати кілька фото утримуйте клавішу Ctrl. Максимальні розміри одного фото: 5 Мб. Формати фото: JPEG, JPG, PNG. Спрощена система завантаження фотографій. Як правильно сфотографувати нерухомість.'
- 'Опишіть об'єкт' section with a text input field and a note: 'В описаних заборонено вводити номер телефону. Ви можете вказати його в своїй анкеті.'
- A green 'Опублікувати' button at the bottom.

Рисунок 4.16 – Другий крок додавання оголошення

У додатку адміністратора реалізована можливість додавання новин у систему. За допомогою пункту меню «Новини» передбачається заповнення адміністратором форми, яка наведена на рисунку 4.17. Форма має вбудований текстовий редактор та файловий менеджер. Правильність введення даних у поля перевіряється на коректність.

The screenshot shows a web application interface for adding news. At the top, there is a logo for 'web-сервіс агенції нерухомості' and navigation links for 'Продаж житла', 'Оренда житла', and 'Новобудови'. A green button labeled '+ Додати оголошення' is visible. Below the navigation, there are tabs for 'Моє меню', 'Оголошення', 'Новини', and 'Користувачі'. The main form contains several input fields: 'Заголовок новини', 'Мета-опис', 'Ключові слова', 'Аліас', and 'Зображення'. To the right, there are dropdown menus for 'Тип нерухомості' (set to 'квартира') and 'Область' (set to 'Оберть область'). Below these fields are two rich text editors, each with a toolbar containing various text formatting options like bold, italic, underline, and font color. At the bottom of the form, there are two green buttons: 'Додати' and 'Відхилити'.

Рисунок 4.17 – Сторінка додавання новин

Для того щоб додати новину необхідно натиснути кнопку «Додати» або в разі необхідності, «Відхилити».

Для забезпечення додавання зображень в систему у файловому менеджері передбачена можливість налаштування властивостей зображення, що дозволяє зручно додавати зображення необхідної якості в систему (рис. 4.18).

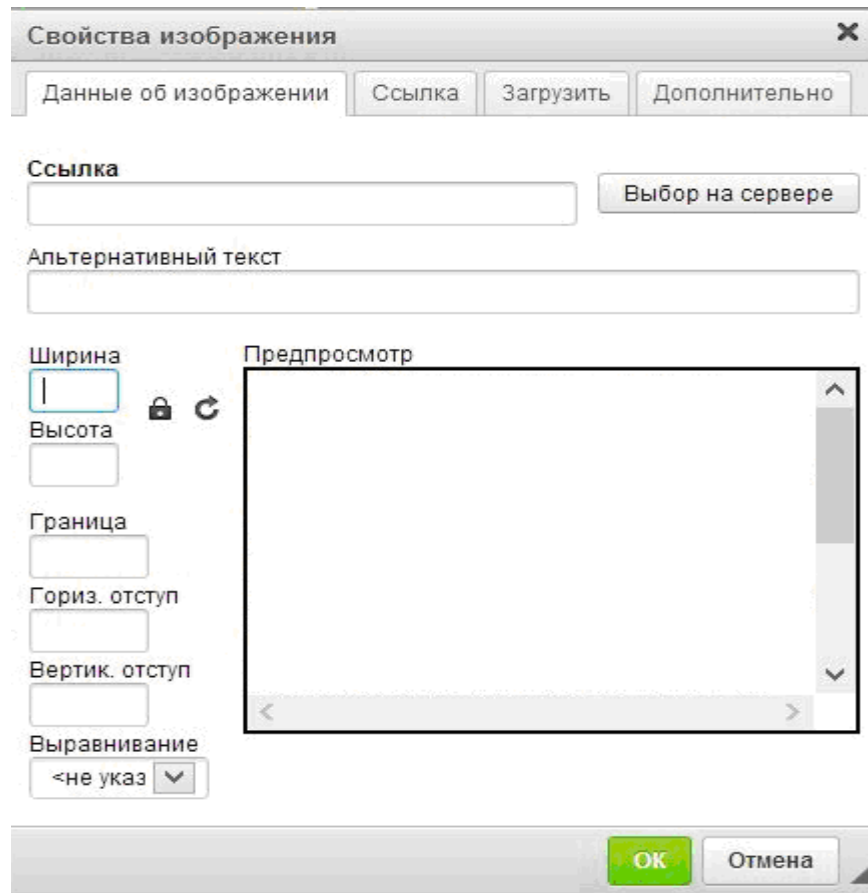


Рисунок 4.18– Файловий менеджер для додавання зображень та файлів

Розроблений веб-сервіс підтримки діяльності агенції нерухомості відповідає вимогам з забезпечення простого та зручного використання користувачами будь-якого агентства нерухомості, може бути використано для впровадження у будь-яку вже існуючу інформаційну систему. Використання такого веб-сервісу агенціями нерухомості набагато покращить якість та швидкість надання послуг пошуку необхідного об'єкту нерухомого майна. Сервіс забезпечує можливість перегляду переліку оголошень, які за допомогою технології парсеру, додаються до бази даних, а після запиту користувача виводиться на екран.

ВИСНОВКИ

В результаті виконання магістерської роботи було виконано дослідження і аналіз технологій хмарових обчислень, та згідно з обраним протоколом, виконано проектування і програмна реалізація web-сервісу підтримки діяльності агенції нерухомості. Для здійснення проектування та розробки web-сервісу підтримки діяльності агенції нерухомості було проведено аналіз та вибір програмних засобів реалізації, зокрема протоколу SOAP, використовуючи математичний апарат теорії прийняття рішень, а саме метод аналізу ієрархій.

Основним призначенням розробленого сервісу є допомога інформаційним ресурсам, діючим у мережі Інтернет, використовувати функціональні можливості сервісу, вбудовуючи його в функціонуючі інформаційні системи. Web-сервіс підтримки діяльності агенції нерухомості функціонує так, що ним можуть користуватись розробники або адміністратори вже існуючих web-ресурсів, які зацікавлені в інформації, яку надає цей сервіс. Реалізована можливість вбудовувати окремі модулі web-сервісу підтримки діяльності агенції нерухомості в інші web-системи, а також є можливість використовувати даний web-сервіс іншими web-сервісами. До окремих модулів, що можуть бути вбудовані в web-системи відносяться такі: модуль новин, модуль оголошень. Ці модулі можуть вбудовуватись в інші web-системи та існувати незалежно від них, так як вся інформація зберігається в базі даних web-сервісу. При розробці сервісу були використані сучасні програмні засоби: СУБД MySQL, сервер Apache, об'єктно-орієнтовані мови програмування PHP та Javascript, бібліотека jQuery, мова розмітки гіпертексту HTML та каскадні таблиці стилів CSS.

Результатом магістерської роботи є розроблений web-сервіс підтримки діяльності агенції нерухомості, на основі протоколу SOAP, який є готовим програмним забезпеченням для впровадження в існуючі інформаційні ресурси.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Міжнародний досвід застосування інтернет-маркетингу. Л. С. Ринейська Ефективна економіка №3 2015. URL: <http://www.economy.nayka.com.ua/?op=1&z=3895> (дата звернення: 15.05.2019).
2. Введення в веб-сервіси. EasyCode. URL: <http://easy-code.com.ua/2011/03/vvedennya-v-veb-servisi/> (дата звернення: 15.05.2019).
3. Хмарні технології. Переваги і недоліки. Валтек системний інтегратор – ІТ послуги системної інтеграції в Україні. URL: <https://valtek.com.ua/ua/system-integration/it-infrastructure/clouds/cloud-technologies> (дата звернення: 10.07.2019).
4. Хмарні обчислення в правовому полі України «Взгляд юриста». URL: <http://jurblog.com.ua/2016/08/hmarni-obchislennya-v-pravovomu-poli-ukrayini/> (дата звернення: 10.07.2019).
5. Технологии и средства связи. URL: lib.tsonline.ru/articles2/fix-corp/oblachnye-vozmozhnosti-preimuschestva-i-perspektivy-ispolzovaniya-oblachnyh-tehnologiy-v-sovremennom-mire (дата звернення: 18.08.2019).
6. Веб-сервисы в теории и на практике для начинающих / Хабр. URL: <https://habr.com/ru/post/46374/> (дата звернення: 19.08.2019).
7. Веб-сервисы. Сетевые технологии. URL: <http://www.4stud.info/networking/web-services.html> (дата звернення: 19.08.2019).
8. UBS. Публикации о технологиях веб-сервисов. URL: http://www.ubs.ru/ws/ws_soad1.html (дата звернення: 26.09.2019).
9. Литвин В.В., В.А. Висоцька, Д.Г. Досин. Методи та засоби опрацювання інформаційних ресурсів на основі онтологій. Львів: ЛА «Піраміда», 2016. 404 с.
10. Подлиновский В.В. Введение в теорию важности критериев в многокритериальных задачах принятия решений. М.: Физматлит, 2007. 64 с.

11. Люк Веллинг, Лора Томсон. Разработка веб-приложений с помощью PHP и MySQL. М.: Вильямс, 2010. 848 с.
12. Карпова Т.С. Базы данных: модели, разработка, реализация. СПб.: Питер, 2001. 304 с.
13. Томасон Лаура, Люк Веллинг. Разработка Web-приложений на PHP и MySQL: пер. с англ. 2-е изд. СПб.: ООО ДиаСофтЮП, 2003. 672 с.
14. У. Мерсер, А. Кент, Д. Новицкий и другие. PHP 5 для начинающих. М.: Диалектика, 2006. 846 с.