МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

ФАКУЛЬТЕТ ЕЛЕКТОРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА КОМП’ЮТЕРНИХ НАУК

СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

**ЗВІТ**

**з науково-дослідної роботи**

за освітньою програмою 8.122.00.02 “Інформаційні технології проектування”

**Виконавець роботи** *Тиченко Ярослав Михайлович*

(прізвище, ім’я, по батькові)

(підпис, дата)

**Звіт захищено з оцінкою**  « » 2018 р.

**Науковий керівник практики**  к. т. н. доц. Шендрик В.В.

(підпис) (наукова ступінь, вчене звання, прізвище та ініціали)

**Суми-2018**

# ЗМІСТ

[ВСТУП 2](#_Toc503000176)

[1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ 4](#_Toc503000177)

[1.1 Системи управління взаємовідносинами з клієнтами 4](#_Toc503000178)

[1.2 Формальний опис простору даних 6](#_Toc503000179)

[1.3 Дослідження аналогів 17](#_Toc503000180)

[2 ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ 20](#_Toc503000181)

[2.1 Мета та задачі дослідження 20](#_Toc503000182)

[2.2 Вибір методів 21](#_Toc503000183)

[3 МОДЕЛЮВАННЯ СИСТЕМИ УПРАВЛІННЯ ВІДНОСИН З КЛІЄНТАМИ 23](#_Toc503000184)

[3.1 Бізнес процеси агенства нерухомості 23](#_Toc503000185)

[3.2 UML діаграми 27](#_Toc503000186)

[ВИСНОВОК 29](#_Toc503000187)

[СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 31](#_Toc503000188)

# ВСТУП

В наш час інформаційних технологій широко розповсюджені cистеми управління взаємовідносинами з клієнтами (CRM-системи). Вони використовуються майже в усіх сферах бізнесу надання послуг та торгівлі для вирішення багатьох задач. За допомогою CRM-систем можна аналізувати дані, приймати рішення та виконувати багато інших завдань, пов’язаних з керуванням відносин з клієнтами.

CRM-системи набули широкого впровадження у сферах діяльності підприємств, котрі спеціалізуються на операціях з нерухомістю. Це пов’зано з тим, що агентства нерухомості є, найчастіше, клієнтоорієнтованими, тому успішність їх діяльності насамперед залежить від ефективних відносин з клієнтами.

Дана галузь передбачає взаємодію з великою кількістю даних у розрізнених ресурсах, де не завжди дотримуються єдиних стандартів подання інформації, а її збирання також проходить із джерел з наперед не відомими моделями даних .

У той же час обсяг потрібної інформації для успішної діяльності зростає і тому багато CRM-систем вже нині відчувають проблеми з опрацюванням даних із різних ресурсів. Наявна для зручного опрацювання інформація є або неповна, або є такою, що не дає конкурентної переваги.

Тому агентствам нерухомості та компаніям у сфері операцій з нерухомістю є необхідним можливість опрацьовувати велику кількість інформації з різних ресурсів, які не притримуються стандартів подання інформації. На сьогоднішній день існує велика кількість систем управління взаємовідносинами з клієнтами, але дані рішення не задовольняють потреб по отриманню і обробці інформації із великої кількості різних ресурсів.

**Об'єкт дослідження** – модель простору даних в галузі агентства нерухомістю.

**Предмет дослідження** – cистема управління взаємовідносинами з клієнтами для агентства нерухомості з використанням моделі простору даних, що дасть змогу оброблювати велику кількість різнотипних джерел інформації.

**Гіпотеза дослідження** – впровадження простору даних надасть змогу підвищення якості консолідованих даних, отриманих з різнотипних джерел, що покращить можливість для їх ефективного опрацювання.

****1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ****

1.1 Системи управління взаємовідносинами з клієнтами

CRM-система (Customer Relationship Management або система управління відносинами з клієнтами) — це прикладне програмне забезпечення для організацій, призначене для автоматизації стратегій взаємодії з замовниками (клієнтами), зокрема, для підвищення рівня продажів, оптимізації маркетингу і поліпшення обслуговування клієнтів шляхом збереження інформації про клієнтів і історію взаємин з ними, встановлення і поліпшення бізнес-процесів і подальшого аналізу результатів.

Сучасні CRM направлені на вивчення ринку і конкретних потреб клієнтів. На основі цих знань розробляються нові товари або послуги і таким чином компанія досягає поставлених цілей і покращує свій фінансовий показник.

Існує три CRM-підходи, кожен з яких може бути реалізованим окремо від інших:

* оперативний — автоматизація споживчих бізнес-процесів, що допомагає персоналу з роботи з клієнтами виконувати свої функції;
* співробітницький — програма взаємодіє зі споживачами без участі персоналу з роботи з клієнтами;
* аналітичний — аналіз інформації про споживачів із різноманітними цілями;

Також, при розробці і виборі CRM-системи, їх розділяють за трьома принципами:

* наявність єдиного сховища інформації, звідки в будь-який момент доступні усі відомості про усі випадки взаємодії з клієнтом;
* синхронізація управління множинними каналами взаємодії;
* постійний аналіз зібраної інформації про клієнтів та прийняття відповідних організаційних рішень — наприклад, «сортування» клієнтів на основі їхньої значимості для компанії.

Серед функцій CRM-систем, треба виділити можливість управління відділом продажу та контроль роботи менеджерів, оперативність отримання інформації і точність контрольованих даних.

Управління відділом продажу та контроль роботи менеджерів. Деякі менеджери продажу здатні до самостійної роботи, проте здебільшого за роботою співробітників потрібен постійний контроль. Менеджер, який знає, що вся його робота контролюється керівництвом, як правило, працює краще.

Оперативність отримання інформації. Якщо взяти ситуацію, коли від менеджера потрібно терміново отримати оперативну інформацію, як у нього йдуть справи з найважливішим і ключовим клієнтом прямо зараз – які є способи отримання цих відомостей від менеджера? При використанні CRM-системи, менеджерів найчастіше взагалі не потрібно відволікати від роботи. CRM дозволяє отримати повний контроль над роботою менеджерів в режимі реального часу, при цьому вони продовжують займатися своєю справою.

Точність контрольованих даних. Іншою проблемою при ручній підготовці звітів стає точність одержуваних даних. Якщо робота з клієнтами поставлена на потік, а кількість угод в роботі більше десятка, то ручне складання звітів, які б містили точні і безпомилкові дані стає вельми не простим завданням. До моменту написання звіту, якась інформація забудеться, якась загубиться, іншу безслідно понесе з собою менеджер, що раптово звільнився. До того ж складання звітів, які б містили не один-два, а скажімо десяток ключових параметрів – потребують від співробітників витратити забагато часу, якого і так мало. Очевидно, що керівник, який при прийнятті рішень про розвиток бізнесу спирається на невірні відомості, буде приймати неправильні рішення.

1.2 Формальний опис простору даних

Отже, простір даних DS – це множина даних, поданих у різних моделях (баз даних DB, сховищ даних DW, статичних веб-сторінок Wb, неструктурованих даних Nd, графічних та мультимедійних даних Gr), локальних сховищ та індексів (ODW), а також засобів інтеграції (Int), пошуку (Se) та опрацювання інформації (Wo), об’єднаних середовищем управління моделями (EM). Як ключова задача робіт у області управління даними використовується платформа підтримки просторів даних (DataSpace Support Platforms, DSSP). Алгебраїчне представлення простору даних є таким:

DS=<DB, DW, ODW, Wb, Nd, Gr, Int, Se, Wo, EM> (1.1)

Моделі даних, що підтримуються у просторі даних, утворюватимуть ієрархію відповідно до їх виразної потужності [1]: реляційна, багатовимірна, об’єктно-реляційна моделі, розширена мова розмітки інформації (Extensible Markup Language — XML) зі схемою, середовище опису ресурсів (Resource Description Framework – RDF), стандартний засіб опису зв’язків між об’єктами даних – онтології, описані за допомогою Web Ontology Language – OWL, структурований текст (у тому числі HTML), неструктурований текст, це показано на рисунку 1.1.

Кожен учасник простору даних підтримує деяку модель даних і деяку мову запитів, відповідну цій моделі. "Запит" до такої моделі даних відповідає тому, що звичайно підтримується у файлових системах відносно їх директорій: зіставлення імен, пошук в діапазоні дат, сортування за розміром файла і т.д. На наступному рівні DSSP повинна підтримувати модель даних мультимножини слів для ефективного пошуку необхідної інформації за ключовими словами, отже, ми отримаємо деяку можливість бачення вмісту учасників простору даних. Нижче рівня моделі мультимножини слів в ієрархії може розташовуватися модель напівструктурованих даних, заснована на позначених графах.

За наявності деякого середовища ключова проблема полягає в знаходженні методів інтерпретації запитів різними мовами для учасників, що підтримують деякі моделі. Точніше, проблема полягає в переформулюванні запиту, поданому складною мовою, для джерела, яке підтримує слабкішу модель даних, і навпаки, переформулюванні запиту, поданого простою мовою, для джерела, яке підтримує виразніші модель даних і мову запитів (наприклад, запит за ключовими словами до реляційної бази даних).

Однією з основних служб простору даних є каталогізація елементів даних учасників. Каталог CG – це реєстр ресурсів даних, що містить основну інформацію про кожний з них: джерело, ім'я, місцеположення в джерелі, розмір, дата створення і власник і т.д. Каталог є інфраструктурою для більшості інших сервісів простору даних, але він також може підтримувати базовий, призначений для користувача, інтерфейс проглядання простору даних:

DB, DW, Wb, Nd, Gr⇒CG. (1.2)



Рисунок 1.1 – Придатність моделей даних до підтримки мов запитів та до використання в глобальній мережі

Він не тільки містить описову інформацію (тобто виконує роль метаданих), але й зберігає для кожного учасника схему джерела, статистичні дані, швидкість зміни, точність, можливості відповідей на запити, інформацію про власника і дані, про політику доступу і підтримку конфіденційності. Оскільки джерела простору даних фізично не переносять у нього інформацію та можуть обмінюватись між собою інформацією, то у каталозі необхідно зберігати дані і про зв’язки між джерелами.

Відмінності між поданням джерел даних у метаданих та каталозі схематично подані на рисунку 1.2.



Рисунок 1.2 – Подання даних у метаданих у сховищі даних та каталозі простору даних

На рис. 2 джерелами даних сховища даних є бази даних (реляційні, об’єктно-реляційні або багатовимірні) та оперативні сховища даних, а джерелами простору даних є об’єкти, подані у довільній моделі. Тому у каталозі необхідно також вказувати тип джерела та засоби його опрацювання (програмні продукти, стандарти передачі тощо).

Зв'язки у каталозі можуть зберігатися у вигляді:

* метаданих,
* перетворень запитів,
* графів залежності,
* текстових описів,
* тощо.

Приклад схеми каталогу поданий на рисунку 3. Залежно від усієї реалізації простору даних для каталогу можна використовувати відношення реляційної моделі, XML-файли, програмні модулі тощо.

Поверх каталога розміщене середовище управління моделями, яке дає змогу створювати нові зв'язки і маніпулювати існуючими зв'язками (наприклад, об'єднувати або інвертувати відображення, зливати схеми і створювати єдині представлення декількох джерел).

Для ідентифікації та роботи з неоднорідними колекціями в просторі даних можна використовувати глобальну схему імен (Uniform Resource Identifiers – URI ) як механізм посилань на глобальні константи, щодо яких є деяка угода між декількома постачальниками даних.

Важливою компонентою простору даних є компонента зберігання і індексування (ODW) для досягнення таких цілей:

* для створення асоціацій між об'єктами даних від різних учасників;
* для вдосконалення доступу до джерел з обмеженими власними засобами доступу;
* для забезпечення можливості виконання деяких запитів без доступу до реального джерела даних;
* для підтримки високого рівня доступності і відновлення.

Засоби індексування повинні володіти високим рівнем адаптивності до неоднорідних середовищ. Результатом локального зберігання та індексування є запит, що може повернути, наприклад, рядок в текстовому файлі, елемент шляху до файла, значення в базі даних, елемент схеми або тег в XML-файлі. Важливими аспектами індексу є те, що, по-перше, він визначає інформацію для всіх учасників, коли деякі значення входять до декількох джерел даних (у деякому розумінні це узагальнює ідею індексів з'єднання). По-друге, індекс повинен справлятися з різноманітністю посилань на об'єкти предметної області, наприклад, з різними способами опису адміністративної одиниці.



Рисунок 1.3 – Схема каталогу простору даних

Отже, зв’язок між каталогом CG, середовищем управління моделями EM та локальним сховищем та індексами ODW можна подати як функцію:

EM(CG)⇒ODW (1.3)

Чим більше моделей здатне «розрізнити» середовище управління, тим точнішою буде інфорація в ODW і тим ефективніше можна буде проводити процедури інтеграції, пошуку та опрацювання даних у просторі даних DS.

Оскільки одним із ключових питань простору даних є питання інтеграції, то розглянемо стандарти інтеграції. Інтеграція інформаційних систем на основі веб-служб Int пов'язана з використанням чотирьох ключових стандартів [4]:

* Розширена мова розмітки інформації — Extensible Markup Language (XML). Описує інформацію, що пересилається по Інтернету. Запит на одержання яких-небудь даних чи виконання певних дій іншим застосуванням вимагає наявності способів передачі параметрів і одержання назад певних результатів. При використанні веб-служб ця інформація описується за допомогою мови XML, що є міжнародним загальноприйнятим стандартом для опису довільних даних, якими, своєю чергою, можуть обмінюватися інформаційні системи.
* Простий протокол доступу до об'єкта — Simple Object Access Protocol (SOAP). Цей стандарт описує протокол виклику веб-служби (віддалений процес доступу до послуг/інформації деякої прикладної системи). У типовій ситуації взаємодії система однієї організації може викликати систему іншої організації, використовуючи протокол SOAP. Запит, що зазвичай містить ту чи іншу форму бізнес-документа, посилається ініціатором до запитуваної системи. Остання приймає запит, і вхідний документ, який міститься в запиті, обробляється. У результаті запитана система генерує відповідь, що повертається ініціатору взаємодії. Ініціатор також інформується про статус (успіх або інше) запиту.
* Мова опису веб-служб — Web Services Description Language (WSDL). Це мова, яка ґрунтується на стандарті XML, що визначає спосіб доступу до веб-служб. Вона описує функціональні можливості веб-служб і групує операції взаємодії у певні інтерфейси, що задають способи виконання операцій і ті параметри, які повинні бути на вході і виході.

Інтеграція на основі веб-сервісів має декілька рівнів. На рівні даних програмні застовування можуть обмінюватись інформацією. Цей рівень передбачає інтеграцію даних і є найпростішим. Наступний рівень – об’єктна взаємодія. Тут йдеться про те, що програмне застосування, розташоване на одному сервері, може запускати програмні процеси на іншому. Третій рівень інтеграції – інтеграція на рівні стандартної семантики. На цьому рівні сервіси можуть “спілкуватися спільною мовою”, обходячи технологічні розбіжності. Один сервіс може звертатись до іншого із “запитом на виконання покупки”, “запитом на виконання пошуку”, “запитом на отримання статистики” та ін. На цьому рівні інтеграції сервіси потребуватимуть лише стандартизації семантики, тобто, під словами “покупка”, “пошук” і “статистика” вони повинні розуміти одне й те саме. Якщо семантичних розбіжностей між ними немає, інтеграція не має особливих труднощів. Тобто, використовуючи специфікацію WSDL, програмне застосування може “говорити” системно-незалежною мовою. З одного боку, системна незалежність застосувань постає з використання мови XML при створенні WSDL-описів, а з іншого – специфікація SOAP дає змогу взаємодіяти серверному та клієнтському застосуванням. Потрібно лише надати вхідні дані, а турботи про те, яким чином доставити їх додатку на обробку та повернути її результати назад, протокол SOAP повністю бере на себе.

Водночас існують і деякі недоліки технології веб-сервісів. У [16] зазначено основні недоліки: неоднозначність специфікації SOAP, недостатня безпечність та недостатня швидкість роботи веб- сервісів.

Проте простори даних не є підходом до інтеграції даних; швидше, це підхід співіснування даних. Мета підтримки простору даних полягає в забезпеченні базового набору функцій між всіма джерелами даних, а не в їх інтеграції. Наприклад, DSSP може забезпечити між всіма своїми джерелами даних пошук за ключовими словами, аналогічно тому, що забезпечують існуючі пошукові системи в десктопах. При потребі в складніших операціях, таких як запити в реляційному стилі, аналіз даних (data mining) або моніторинг яких-небудь джерел, можна докласти додаткові зусилля до тіснішої інтеграції цих джерел в інкрементній манері "оплати поточних рахунків" ("pay-as-you-go").

DSSP повинні працювати з даними і застосуваннями в різноманітних форматах, доступних від багатьох систем через різні інтерфейси. Від DSSP потрібна підтримка всіх даних простору даних, без яких-небудь винятків (як це буває при використанні СУБД). Тому однієою із ключових задач побудови простору даних є визначення виразної потужності запитів із Se. Цей компонент повинен забезпечувати наступні можливості.

Запит про довільні дані. У користувачів повинна бути можливість запиту будь-якого елемента даних, незалежно від його формату і моделі даних. Спочатку DSSP повинні підтримувати для кожного учасника запити за ключовими словами. У міру того, як ми одержимо більше інформації про учасника, ми повинні поступово почати підтримувати складніші запити. Система повинна підтримувати плавне перемикання між запитами за ключовими словами, переглядом і структурованими запитами. Зокрема, при видачі відповідей на запит за ключовими словами (або на структурований запит) повинні пропонуватися додаткові інтерфейси запитів, що дають змогу користувачу удосконалити свій запит.

Стуктуровані запити. Запити в стилі баз даних повинні підтримуватися на основі загальних інтерфейсів (тобто схем-посередників), що забезпечують доступ до декількох джерел, чи можуть адресуватися до конкретного джерела даних (з використанням його власної схеми) з наміром отримання відповідей і від інших джерел (як в системах управління одноранговими даними — Peer-Data Management System) [6]. Запити можуть формулюватися різноманітними мовами (і на основі різних моделей даних) і повинні, по можливості, якнайкраще переформулювати на інші моделі даних і схеми, забезпечуючи точні і наближені семантичні відображення.

Запити до метаданих. У системі повинен підтримуватися широкий спектр запитів до метаданих. Повинні забезпечуватися можливості:

* отримання даних про джерело відповіді або про те, як ця відповідь була виведена або обчислена;
* забезпечення тимчасових позначок на елементах даних, які брали участь в обчисленні відповіді;
* визначення елементів даних у просторі даних, що можуть залежати від заданого елемента даних, і підтримка гіпотетичних запитів (тобто Що б змінилося, якби я видалив елемент даних X?);
* запити джерел і рівня невірогідності відповіді.

DSSP повинні також підтримувати запити на встановлення місцеположення даних, відповідями на які є джерела даних, а не конкретні елементи даних. Наприклад, система повинна бути в стані відповідати на запити Де я можу знайти дані про Чернівецьку область? або В яких джерелах є атрибут "призначення"?. Аналогічно, за наявності XML-документа повинна бути можливість вибрати XML-документи зі схожою структурою і відповідні XML-перетворення. Нарешті, за наявності фрагмента схеми або опису Web-сервісу повинно бути можливо знайти в просторі даних схожі фрагменти.

Моніторинг. Всі перераховані служби пошуку і запиту даних повинні також підтримуватися в інкрементній формі, що застосовується у реальному часі до потокових або змінних джерел даних. Моніторинг може бути організований у вигляді процесу без стану, в якому елементи даних розглядаються окремо, або у вигляді процесу із станом, в якому аналізується декілька елементів даних. Наприклад, фільтрація повідомлень – це процес без станів, а віконне агрегатне обчислення – це процес із станами. Служба інкрементного моніторингу може забезпечити додаткові функції виявлення складних подій і генерації сигналів.

Хоча DSSP забезпечує засоби інтегрованого пошуку, запиту, оновлення і адміністрування просторів даних, ті самі дані часто можуть бути доступні для читання і оновлення через власний інтерфейс системи, що безпосередньо управляє даними. Тому, на відміну від СУБД, DSSP не має повного контролю над своїми даними.

Засоби опрацювання даних Wo повинні підтримувати:

* Видобування даних (Data mining) – асоціативні правила, дерева рішень, генетичні алгоритми тощо;
* Засоби аналізу даних (Online Analytical Processing – OLAP) – реляційний OLAP (Relational OLAP – ROLAP), багатовимірний OLAP (Multidimensional OLAP – MOLAP), гібридний OLAP (Hybrid OLAP – HOLAP), динамічний OLAP (Dynamic OLAP – DOLAP);
* Засоби природномовного пошуку – побудова нечітких запитів, запитів у вигляді природних питань, запитів до метаданих;
* Засоби підбору контенту на основі аналізу характеристик користувача;

Засоби миттєвого аналізу даних (наприклад, визначення причин підвищення тиску у котлах за значеннями давачів приладів та пропонування методів усунення неполадок).

Схему зв’язку між елементами сховища даних подано на рисунку 

Рисунок 1.4 – Схема зв’язку між елементами сховища даних

Отже, треба виділити такі особливості просторів даних [7]:

* Простори даних складаються з широкої різноманітності форматів та інтерфейсів і усі без винятку формати даних повинні підтримуватися;
* Дані у просторі даних повністю не контролюються;
* Передбачається інтеграція тексту, даних, коду і потоків;
* Підтримка структурованих, текстових, просторових, темпоральних, мультимедійних, процедурних даних; тригерів; потоків і черг даних як рівноправних компонентів;
* Простори даних повинні забезпечувати вбудовану підтримку неточних даних. Повинна бути можливість задання неточних запитів, і процесор запитів повинен відноситися до цього як до додаткового джерела неповноти і неточності;
* Відповіді на запити повинні залежати від профілю користувача.
* Відповідь на запит експерта повинна відрізнятися від відповіді на запит новачка. Релевантність відповіді теж повинна залежати від користувача і від контексту;
* Система повинна знати точні взаємозв'язки між елементами, що використовуються у кожній схемі;
* DSSP пропонує рівні обслуговування та методи отриманні приблизних відповідей;
* DSSP повинен запропонувати інструменти і шляхи створення щільнішої інтеграції даних в просторі в міру необхідності.

Можуть забезпечуватися різні рівні послуг з обробки запитів до DSSP, і в деяких випадках вони можуть повертати якнайкращі з можливих приблизні відповіді. Наприклад, якщо деякі джерела даних стають недоступними, DSSP може забезпечити найкращий з можливих результат на основі даних, доступних під час виконання запиту.



Рисунок 1.5 – Об’єкти простору даних та його задачі

Простір даних підпорядковується загальним методам адміністрування.

1.3 Дослідження аналогів

На сьогоднішній день, на ринку доступна безліч різноманітних систем управління відносинами з клієнтами. Розглянемо найпопулярніші з них на місцевому ринку, а саме: «Бітрікс 24» та «AmoCRM». Інші аналоги представлені на ринку мають «скромніший» набір функціоналу, або є невідповідними для нашої предметної області, тому не можуть вирішити поставлену задачу

Бітрікс 24 – з’явившись відносно недавно, ця система стрімко підкорює ринок. І можна не сумніватися в тому, що таке динамічне зростання багато в чому стало можливим завдяки гарним маркетинговим бюджетам, оскільки творці CRM – власники CMS «Бітрікс».



Рисунок 1.6 – Вигляд Бітрікс 24

Платформа надає широкий набір інструментів для організації ефективної роботи практично будь-якого бізнесу:

* створення проектів, постановка і відстеження завдань за ними;
* управління продажами, комерційними пропозиціями та іншими операціями;
* воронка продажів та обробка лідів;
* “хмарний” диск для зберігання документації та файлів;
* розділення прав доступу і ролей;
* інтеграція з сайтом та інтернет-магазином;
* живий чат між співробітниками компанії;
* контакти, запис дзвінків, звіти і багато іншого.великий ряд можливостей

Але дана система має великі обмеження щодо корегування або розширення системи, настройки системи під індивідуальні потреби бізнесу. Також більшість функцій заточені під CMS «Бітрікс», що також накладає обмеження.

AmoCRM — система, орієнтована насамперед на менеджерів із продажу або інших фахівців, які активно працюють переважно в цьому сегменті. Кожна угода та клієнт тут мають власну картку, що містить всю необхідну інформацію:

* етап операції (воронка продажів);
* відповідальний менеджер;
* інформація про клієнта і т. д.

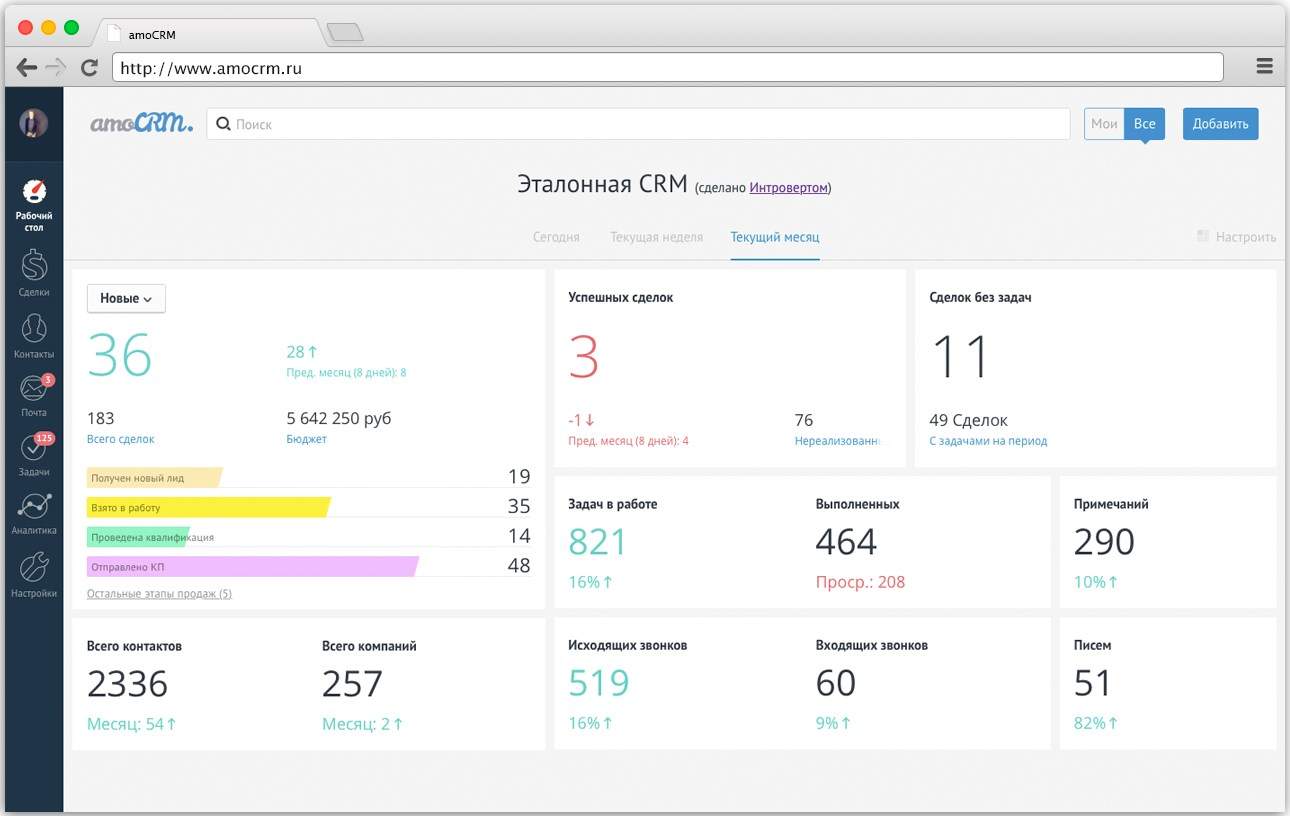


Рисунок 1.7 – Інтерфейс «amoCRM»

AmoCRM — є досить зручним продуктом, але система налічує істотну кількість багів, не має змогу настройки зручності інтерфейсу, додаткові функції є досить дорогими в ціні та важки в експлуатації. Також потребує додаткового навчання користувачів.

# 2 ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

2.1 Мета та задачі дослідження

Метою роботи є розроблення інформаційної системи управління взаємовідносинами з клієнтами , з використанням моделі простору даних, що дасть змогу обробляти велику кількість різних джерел інформації. Результатом використання такої системи має стати підвищення якості консолідованих даних, отриманих з різнотипних джерел, що покращить можливість для їх ефективного опрацювання.

Для досягнення поставленої мети поставлені наступні задачі дослідження:

* аналіз процесів діяльності агентств нерухомості;
* визначення вимог агентства нерухомості до інформації отриманої із різних джерел;
* визначення критеріїв оцінювання даних із різнотипних джерел, до яких звертається система;
* визначення критеріїв, необхідних для створення платформи підтримки простору даних в галузі агентства нерухомості;
* побудова оптимальної моделі простору даних в галузі агентства нерухомості .

# 2.2 Вибір методів

Дотримуючись сучасних принципів розробки інформаційних систем було вирішено розділити проект на дві складові: серверна частина та користувацькі інтерфейси. Даний підхід надає такі переваги:

• можливість подальшого використання даних із серверної частини у інших системах;

• розділення коду за функціоналом, що полегшить процес розробки, тестування та підтримки проекту;

• розділення клієнтської частини та серверу, що дає можливість розробки та редагування окремо кожної частини,.

Для реалізації серверної частини було вирішено створити з використання мови програмування PHP. PHP — скриптова мова програмування, яка була створена для генерації HTML-сторінок на стороні веб-сервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок. Вона є універсальною й гнучкою, простою та традиційною, має багато різноманітних розширень та доповнень, велику спільноту та наявність різноманітних бібліотек. Окрім того PHP є дуже потужною та швидкою, а також має вбудовані засоби безпеки.

Також необхідно обрати СУБД, яка буде використовуватися для збереження даних. Основними вимогами до неї є безкоштовність, надійність, швидкодія та підтримка реляційної моделі даних. Проаналізувавши ринок програмного забезпечення було обрано СУБД MySQL, яка відповідає всі вимогам, а також має цілий ряд додаткових переваг, а саме:

• простота в роботі - MySQL встановити досить просто. Існують різноманітні програми, наприклад графічний інтерфейс, дозволяє досить легко працювати з БД; має простий інтерфейс для роботи з PHP;

• багатий функціонал - MySQL підтримує більшість функціоналу SQL;

• безпека - велика кількість функцій забезпечують безпеку, по Які замовчуванням підтримується;

• масштабованість - MySQL легко працює з великими обсягами даних і легко масштабується;

• швидкість - спрощення деяких стандартів дозволяє MySQL значно збільшити продуктивність.

Для реалізації клієнтських інтерфейсів було вирішено використовувати мова розмітки веб-сторінок HTML та каскадні таблиці стилів, а для реалізації інтерактивних елементів буде використовуватися динамічна, об'єктно-орієнтованa мова програмування JavaScript. Для створення якісного продукту було вирішено використовувати клієнтський фреймворк призначений для створення веб-сайтів Bootstrap.

# 3 МОДЕЛЮВАННЯ СИСТЕМИ УПРАВЛІННЯ ВІДНОСИН З КЛІЄНТАМИ

# 3.1 Бізнес процеси агенства нерухомості

Бізнес-процес — це стійка,цілеспрямована сукупність взаємопов’язаних видів діяльності (послідовність роботи),котра за конкретною технологією перетворює входи на виходи, надаючи цінності для споживача.

Результатом моделювання бізнес-процесів є модель, модель, котра відноситься до одного з двох типів:

• модель AS-IS (як є) — модель поточної організації бізнес-процесів підприємствами;

• модель TO-BE (як буде)— модель ідеальної організації процесів.

SADT (IDEF0) — методологія структурного проектування, яка інтегрує процес моделювання, управління конфігурацією проекту, використання додаткових мовних засобів й керівництво проектом з власною графічною мовою. Ця мова моделювання була запропонована більш ніж 25 років тому Д. Россом (SoftTech, Inc.). Згідно цієї технології аналізований процес представляється у вигляді сукупності безлічі взаємопов’язаних дій, робіт (Activities), які взаємодіють між собою на основі певних правил (Control), з урахуванням споживаних інформаційних, людських та виробничих ресурсів (Mechanism), що мають чітко визначений вхід (Input) й не менш чітко визначений вихід (Output).

AllFusionProcessModeler 7 (раніше BPwin) — інструмент для моделювання, аналізу, документування й оптимізації бізнес-процесів. AllFusionProcessModeler 7 можно використовувати для графічного представлення бізнес-процесів. Графічне викладення цієї інформації дозволяє перевести задачі управління організацією з області важкого ремесла в сферу інженерних технологій.

Основні можливості системи:

• підтримка різноманітних технологій моделювання

• аналіз показників витрат та продуктивності

• Інтеграція процесів/даних

• Підтримка стандартних нотацій

• Експорт об’єктів й властивостей в інші моделі

• Документування інформації в межах всієї моделі

• Масштабованість звітності без втрати якості графіків

Будь-яка IDEF0 діаграма складається з прямокутників, які мають назву роботами (activity), й стрілками (arrow). Роботи уявляють собою деяку конкретну функцію в рамках розглянутої системи. За вимогами стандарту назва кожної роботи має бути виражена віддієслівним іменником. Кожна з чотирьох сторін прямокутника має своє конкретне значення.

Контекстна діаграма є вершиною деревовидної структури діаграм і являє собою найзагальніший опис системи та взаємовідносин в зовнішньому середовищі.

Вхід — це споживча, або змінювана роботою інформація або матеріал.

Вихід — інформація або матеріал, який виробляється роботою.

Управління — процедура, правила, стратегії або стандарти, скими керується робота.

Механізми — ресурси, які виконують роботу (наприклад співробітники, обладнання, пристрої, тощо).

Для підприємства, що розглядається в даній роботі:

- на вході — запит замовника;

- на виході — готова система;

- стрілки управління: функціональні й не функціональні вимоги, технічна документація, специфікація й бізнес правила;

- стрілки механізмів: програміст, менеджер й адміністратор системи.

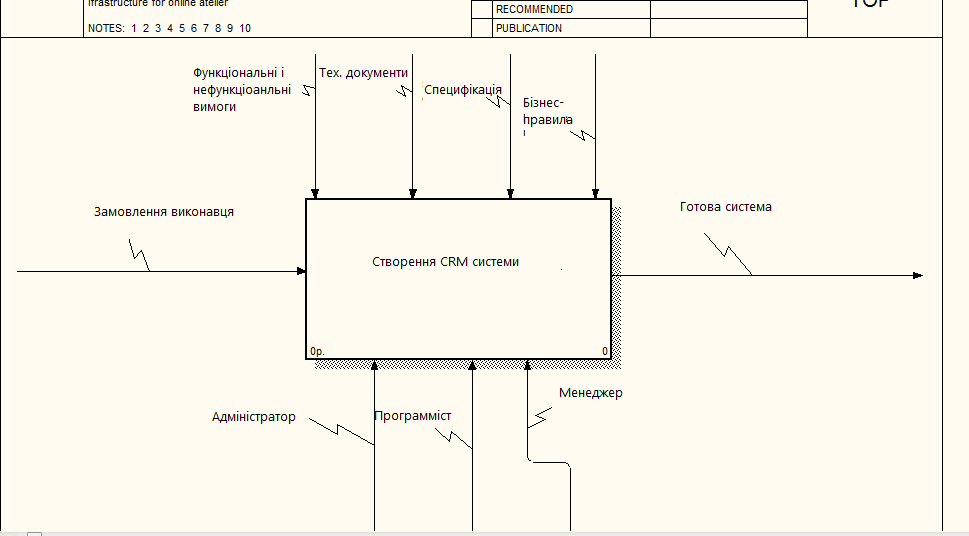


Рисунок 3.1-Контекстна діаграма IDEF0

IDEF3 — технологія збору даних, необхідних для проведення структурного аналізу системи, доповнює технологію IDEF0. За допомогою цієї технології ми маємо можливість уточнювати картину процесу, залучаючи увагу аналітика до чергування виконування функцій та бізнес процесів в цілому. Логіка цієї технології дозволяє будувати та аналізувати альтернативний сценарій розвитку досліджуваних бізнес-процесів ( модель типу «А що — якщо?»).

IDEF3 — методологія моделювання, яка використовує графічний опис інформаційних потоків, взаємовідносини між процесами обробки інформації та об’єктів, які є частиною цих процесів. IDEF3 дає можливість аналітикам описати ситуацію, в якій процеси виконуються за визначеною послідовністю, а також, описати об’єкти які беруть спільну участь в одному процесі. Будь-яка IDEF3-діаграма може містити роботи, зв’язки, перехрестя і об’єкти посилань.

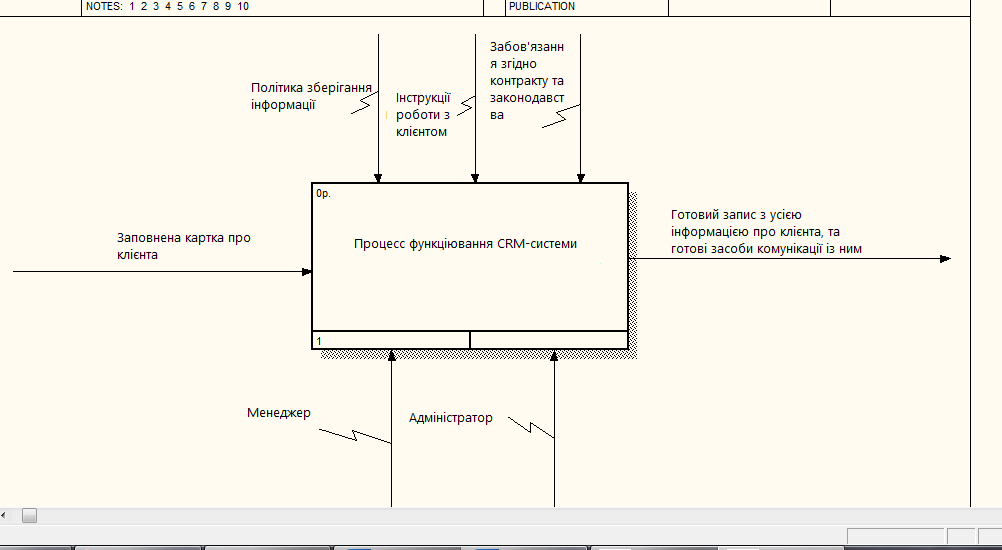


Рисунок 3.2-Контекстна діаграма IDEF3

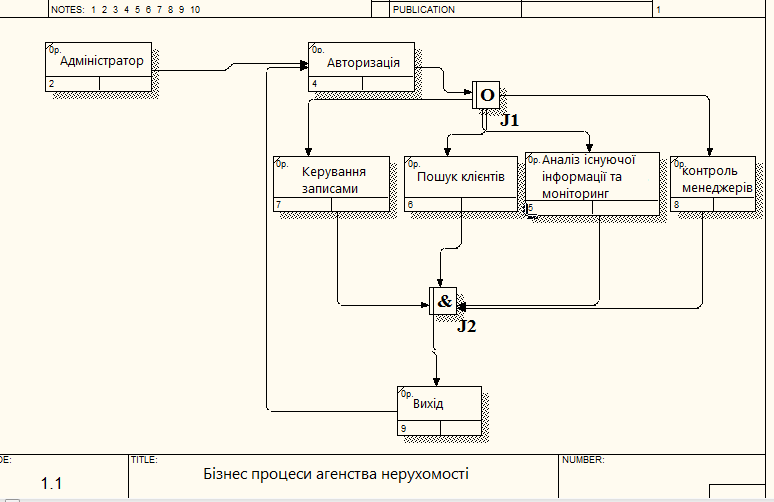


Рисунок 3.3-— Перший рівень декомпозиції діаграми IDEF3 для адміністратора

# 3.2 UML діаграми

UML (англ. UnifiedModelingLanguage – уніфікована мова моделювання) – мова графічного опису для об’єктного моделювання в області розробки програмного забезпечення. UML являє собою мову широкого профілю, це відкритий стандарт, який використовує графічні позначення для створення абстрактної моделі системи, що має назву UML - модель.

1. Діаграма об’єктів – це знімок об’єктів системи, в якійсь момент часу. Оскільки вона показу екземпляри, а не класи, то діаграму об’єктів часто називають діаграмою екземплярів.
2. Діаграма використання – діаграма, яка відображає відносини між акторами та прецедентами, що є складовою частиною моделі прецедентів, дозволяючи описувати систему на концептуальному рівні.

Інструментом для будування UML діаграм є IBM RationalRoseEnterprise

IBM RationalRoseEnterprise надає набір функцій, керованих моделлю, для розробки цілого ряду програм, в тому числі мовами Ada, ANSI C++, C++, CORBA, Java, Java EE, Visual C++ і VisualBasic. Ці програмні забезпечення дозволяють прискорити розробку таких додатків завдяки створенню коду на основі візуальних моделей з використанням UML (UnifiedModelingLanguage).

RationalRoseEnterprise пропонує загальний інструмент моделювання й мову для спрощення робочого середовища й можливості скорішого створення якісного програмного забезпечення.

* Моделювання найпоширеніших програм — надає функції візуального моделювання для розробки програм багатьох типів.
* Розробка веб-додатків — пропонує інструмент XML та веб-моделювання веб-додатків.
* Інтеграція проектування й розробки програм уніфікує роботу колективу проекту шляхом надавання загальних засобів виконання й нотацій модель UML.

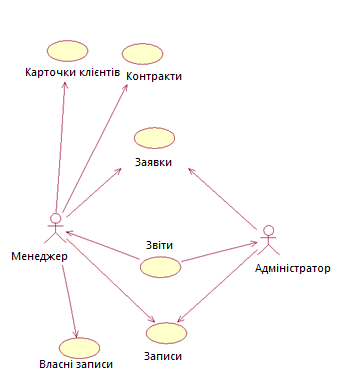


Рисунок 3.3 – Діаграма об’єктів

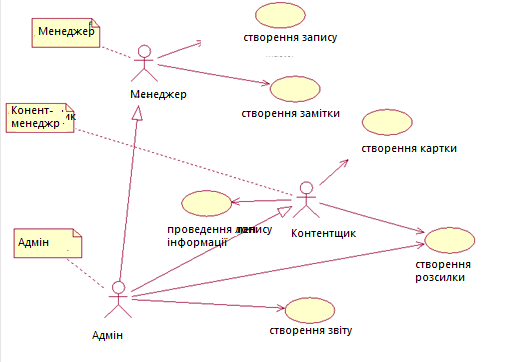


Рисунок 3.4 – Діаграма варіантів використання

# ВИСНОВОК

В сучасних реаліях ведення бізнесу, компаніям в галузі бізнесу надання послуг та продажу неможливо обійтися автоматизації та стандартизації управління відносин з клієнтами, тобто впровадження CRM-системи. Для побудови оптимального системи керування відносинами з клієнтами необхідно визначити основні вимоги до системи, мати на увазі потреби користувачів системи та специфіку предметної області. Агентство нерухомості є клієнтоорієнтованою галуззю, тому ефективно побудовані відносини з клієнтом є одним з найголовніших критеріїв успішності.

CRM-системи для агентства нерухомості має специфічні вимоги, а саме необхідність ефективно обробляти велику кількість інформації із різних джерел. На основі запропонованого підходу до використання простору даних можливо створити систему, що забезпечить підвищення якості консолідованих даних, отриманих з різнотипних джерел, що покращить можливість для їх ефективного опрацювання.

Найважливішим етапом є побудова концептуальної моделі простору даних у галузі агентств нерухомості, для цього необхідно:

1. Треба визначити об’єкти простору даних, яку інформацію вони надають, у якій моделі даних вони передаються та з яких джерел надходять.
2. Треба визначити учасників простору даних. Простір даних повинен містити всю інформацію, необхідну для в галузі агентств нерухомості, незважаючи на формат і місцерозташування цієї інформації, а також моделювати розвинений набір зв'язків між репозиторіями даних. Отже, ми моделюємо простір даних як набір учасників і зв'язків.
3. Зазначити особливості простору даних у галузі агентств нерухомості. В деяких простори даних агентств нерухомості для різних адміністративних одиниць можуть вкладатися одне в інше, саме тому треба вказати правила розмежування доступу для учасників в просторі даних.

Як показує досвід, система з високою якістю та точністю отриманих даних із різних джерел є вагомою конкурентною перевагою в сфері агентств нерухомості, тому вміло побудована модель простору даних має велику цінність і актуальність.

# СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Кузнецов С. От баз данных к пространствам данных: новая абстракция управления информацией. [Електронний ресурс] – 2006, http://www.citforum.ru/database/articles/from\_db\_to\_ds..

2. Дрюэк К. (Katherine Drewek). "Хранилища данных: сходство и различия подходов Билла Инмона и Ральфа Кимболла",2005, [Електронний ресурс] / The TeleManagement Forum. – 2003. – Режим доступу: http://www.b-eye-network.com/view/743

3. Dan Linstedt. Data Vaulttm overview the next evolution in data modeling. [Електронний ресурс] – 2005, http://www.tdan.com/i021hy01.htm.

4. Огляд технологій інтеграції інформаційних систем, [Електронний ресурс] – 2006, – Режим доступу: http://www.microsoft.com/Ukraine/Government/Analytics/Integration

Technologies/Overview.msp.

5. Кузнецов С. Пространства данных: исследовательский полигон или путь к новому поколению систем управления данными? [Електронний ресурс] – Режим доступу: http://synthesis.ipi.ac.ru/sigmod/seminar/s20060420

6. Donald Kossmann, Jens-Peter Dittrich. Personal Data Spaces. [Електронний ресурс]– Режим доступу: еtp://www.inf.ethz.ch/news/focus/res\_focus/feb\_2006/index\_DE.

7. Garretts Summary of Principles of Dataspace Systems, [Електронний ресурс] – Режим доступу: http://aravaipa.eas.asu.edu/wiki/index.php/Garretts\_Summary\_of\_Principles\_of\_Dataspace\_Systems#Overview

8. Donald Kossmann, Jens-Peter Dittrich. Personal Data Spaces. [Електронний ресурс] – Режим доступу: http://www.inf.ethz.ch/news/focus/res\_focus/feb\_2006/index\_DE.9. Cruchten, Ph. The Rational Unified Process: An Introduction / Philippe Cruchten. – 3rd ed. – .Addison-Wesley Professional, 2004. – 310 p.

9. Processing of natural language queries to a relational database. Samsonova

[Електронний ресурс] – M, Pisarev A, Blagov M, Режим доступу: <http://www.cs.dartmouth.edu/~brd/Teaching/AI/Lectures/Summaries/natlang.html>

10. Особенности построения хранилищ. [Електронний ресурс] – Режим доступу: <http://citforum.uar.net/seminars/sch.shtml/>

11. Особенности построения хранилищ данных. [Електронний ресурс] – Режим доступу: <http://citforum.uar.net/seminars/cis99/sch.shtml/>

12. Райлі Дж. NGOSS: Побудова ефективних систем підтримки та експлуатації мереж оператора зв'язку / Дж. Райлі, М.Кринер. – пер. с англ. – М.: Альпіна Бізнес Букс, 2007. – 192 с.

13. Kacprzyk J.,Ziolkowski A. Database Queries with Fuzzy Linguistic Quantifiers // IEEE Transactions on Systems, Man, and Cybernetics. SMC-16, 1996. – P. 512-529.