

(повна назва факультету)

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

(назва освітнього ступеня)

на тему:

Виконав(ла): студент(ка) _____ курсу, групи _____
спеціальності _____

(шифр і назва спеціальності)

(підпис)

(прізвище та ініціали)

Керівник

(підпис)

(прізвище та ініціали)

Нормоконтроль

(підпис)

(прізвище та ініціали)

Завідувач кафедри

(підпис)

(прізвище та ініціали)

Рецензент

(підпис)

(прізвище та ініціали)

РЕФЕРАТ

Кваліфікаційна робота магістра. Тернопільський національний технічний університет імені Івана Пулюя, кафедра програмної інженерії, спеціальність 121 «Інженерія програмного забезпечення». ТНТУ, 2022. Сторінок - 64, рисунків - 26, додатків – 3, презентація.

Тема: Розробка CRM системи для контролю та управління взаємодією з потенційними клієнтами.

Метою кваліфікаційної роботи магістра є проектування та реалізація веб-орієнтованої системи для залучення нових та покращення відносин з існуючими клієнтами.

В роботі здійснено аналіз предметної області, розглянуто доцільність використання CRM системи для різних видів підприємств, а також визначено функціональні вимоги до програмного продукту та проаналізовано обрані веб-технології для реалізації програмного забезпечення. У роботі здійснено проектування архітектури системи, моделі бази даних, моделювання варіантів використання.

Результатом виконання кваліфікаційної роботи магістра є CRM система для зберігання інформації про клієнтів компанії, обробки та аналізу даних задля покращення процесу комунікації та налагодження співпраці, а також для збільшення прибутку підприємства.

Ключові слова: CRM, JAVA, SPRING BOOT, POSTGRESQL, BOOTSTRAP, JAVASCRIPT, THYMELEAF, БІЗНЕС, ПІДПРИЄМСТВО, КЛІЄНТ.

ABSTRACT

Master's qualification work. Ivan Pulyuy Ternopil National Technical University, department of software engineering, specialty 121 "Software engineering". TNTU, 2022. Pages - 64, figures - 26, appendices - 3, presentation.

Topic: Development of a CRM system for controlling and managing interaction with potential customers.

The purpose of the master's paper work is to design and implement a web-based system for attracting new and improving relations with existing customers.

The work analyzed the subject area, considered the expediency of using the CRM system for various types of enterprises, as well as defined the functional requirements for the software product and analyzed the selected web technologies for software implementation. In the work, the design of the system architecture, the database model, and the modeling of use cases were carried out.

The result of the master's qualification work is a CRM system for storing information about the company's customers, processing and analyzing data to improve the communication process and establish cooperation, as well as to increase the company's profit.

Keywords: CRM, JAVA, SPRING BOOT, POSTGRESQL, BOOTSTRAP, JAVASCRIPT, THYMELEAF, BUSINESS, ENTERPRISE, CLIENT.

ЗМІСТ

ВСТУП	5
1 АНАЛІЗ ВИМОГ ДО ПЗ	7
1.1 Аналіз предметної області	7
1.2 Постановка задачі.....	18
1.3 Обґрунтування вибору середовища розробки програмної системи та мови програмування.....	21
2 РОЗРОБКА МОДЕЛІ ТА ПРОЕКТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ ...	27
2.1 Вибір процесу розробки	27
2.2 Розробка моделі предметної області.....	34
2.3 Розробка бізнес моделі	35
2.4 Проектування архітектури	37
3 КОНСТРУЮВАННЯ СИСТЕМИ	39
3.1 Реалізація ключових класів.....	39
3.2 Розробка GUI	41
3.3 Тестування та оцінка якості ПЗ	45
4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКИ ЖИТТЄДІЯЛЬНОСТІ.....	55
4.1 Охорона праці.....	55
4.2 Планування заходів цивільного захисту на об'єкті у випадку надзвичайних ситуацій	57
ВИСНОВКИ.....	62
СПИСОК ВИКОРИАНИХ ДЖЕРЕЛ.....	64
ДОДАТКИ.....	66

Вступ

Потреба в автоматизації різноманітних процесів у сучасному бізнесі стала звичним явищем. Тепер нам все складніше проводити облік, здійснювати аналіз тієї чи іншої інформації, зберігати дані без використання спеціалізованого програмного забезпечення. Оскільки клієнт постає центром бізнесу, для нас важливо побудувати хороші відносини з ним і у цьому може допомогти автоматизована система, що призначена для роботи з клієнтами.

У країнах, де економіка найбільш розвинена, за результатами досліджень виявлено, що компанії, які працюють із значним відсотком клієнтів, не отримують значного прибутку через погану взаємодію з ними. Зменшення числа клієнтів, а також велика кількість негативних відгуків погано впливає на імідж компанії. Тому, що клієнти, які незадоволені обслуговуванням, зазвичай схильні більш розширено розповідати про свій досвід ніж задоволені. Тому для компаній будь-якого виду діяльності важливо залучати клієнтів та правильно взаємодіяти з ними.

Для вирішення подібного роду проблем призначені CRM системи, які можуть збирати та обробляти величезний обсяг інформації щодо взаємовідносин з клієнтами. Такі системи надають можливість автоматизованого управління, а також є чудовим інструментом для аналізу даних, необхідних для побудови ефективної взаємодії та отримання прибутку.

Зовсім недоцільно вважати, що CRM системи призначені лише для великих підприємств, оскільки це не так. Не важливо у якій галузі працює компанія і якого розміру, автоматизована система стане незамінною частиною на шляху до розвитку та росту компанії.

Отож, враховуючи актуальність проблеми, у мене виникла ідея щодо розробки програмного забезпечення, яке б дало можливість за допомогою зручного користувацького інтерфейсу виконувати основні функції CRM системи.

Об'єктом дослідження є застосування новітніх інформаційних технологій у бізнесі для кращого залучення нових клієнтів та ефективнішої взаємодії з вже існуючими.

Предметом дослідження є безпосередньо програмна система для автоматизації обліку, документації, управління взаємовідносинами з клієнтами.

Метою кваліфікаційної роботи магістра є проектування та реалізація веб-орієнтованої системи для залучення нових та покращення відносин з існуючими клієнтами.

Аби досягнути даної мети потрібно виконати наступні задачі:

- здійснити аналіз існуючих програмних систем для управління взаємовідносинами з клієнтами;
- провести дослідження новітніх технологій для веб розробки CRM систем;
- спроектувати та реалізувати CRM систему для управління взаємовідносинами з клієнтами та провести тестування.

Виконуючи список поставлених задач, в результаті даної роботи буде розроблено CRM систему для зберігання інформації про клієнтів компанії, обробки та аналізу даних задля покращення процесу комунікації та налагодження співпраці, а також для збільшення прибутку підприємства.

1 АНАЛІЗ ВИМОГ ДО ПЗ

1.1 Аналіз предметної області

Успішне функціонування будь-якого підприємства неможливо уявити без ефективного управління, адже воно є цінним ресурсом, таким як інформація, яку можна отримати і використати у правильних цілях. Сучасні технології частково змінюють процеси, що відбуваються у бізнесі, відкриваючи при цьому нові можливості та створюючи свіжі ідеї щодо того, як зберегти конкурентоспроможність і підвищити продуктивність, як зменшити витрати та покращити результат роботи. Впровадження інформаційних технологій є досить важливим не тільки для підвищення конкурентоспроможності компанії, але й для формування належних взаємовідносин з клієнтами, оскільки саме клієнти приносять прибуток.

Переважає більшість компаній сьогодні має принаймні інтегровану систему управління. Така система орієнтована на продукт. Вона записує, відстежує та аналізує інформацію про продукти, доходи та витрати на логістику. Проте існує велика кількість інформації та даних, що надходить про клієнтів щодня, яку компанії взагалі не використовують. Програмне забезпечення для виставлення рахунків точно не розповість компанії про те, що сталося до продажу. Хто є менеджером із закупівель, особою, яка приймає рішення чи впливає на вибір клієнта? Незрозуміло, які аргументи могли б переконати їх у тому, що наш продукт чи послуга – правильний вибір. Системи виставлення рахунків не відображають ані задоволеності клієнтів послугами чи продуктом компанії, ані того, чи була якась скарга та як її розглядали. Без цієї інформації підприємства можуть бучити лише їхні доходи та витрати. Керівництво не цікавиться питаннями, які значною мірою впливають на успіх та ефективність роботи.

Одним із мінімальних очікувань від системи управління взаємовідносинами з клієнтами є можливість отримати окрему інформацію, що пов'язана з клієнтами, функціонувати як єдине ціле, а не бути ізольованою [1].

Така система підтримує спільну базу даних, щоб співробітники з різних відділів могли отримати доступ до необхідної інформації. Система повинна надавати можливість кільком користувачам одночасно отримувати доступ до бази даних і щоб внесені зміни негайно зберігались та інформація залишалась актуальною.

Можна помітити, що безпрецедентний потік товарів і послуг обрушився на громадськість в останні роки через глобалізацію ринків, чи то корпорацій, чи домашніх господарств. Оскільки пропозиція перевищує обсяг платоспроможного попиту, то конкуренція між однотипними товарами та послугами різко зростає. Останніми роками можна спостерігати збільшення зусиль підприємств для підвищення конкурентоспроможності. Компанії мають очевидну потребу в кращому розумінні кожного сегменту ринку, а також в ефективній маркетинговій комунікації. Маркетинг, який базується на традиційних засобах масової інформації все менше й менше може вирішити це завдання. Споживачу все важче розрізнити схожі продукти та їхню придатність для особистих потреб. Підприємствам важко вибірково спілкуватися з кожним шаром і сегментом споживачів. Зобов'язання компаній щодо реклами зростають на ринку, а разом з ними зростають і витрати на маркетинг. Бажання рентабельного маркетингу та більш вражаючої комунікації звільнили простір для прямого маркетингу. Прямий маркетинг використовує канал зв'язку, який ініціює особистий контакт і зв'язок з потенційним клієнтом через обрані медіа. Розробка маркетингової бази є важливою для цього типу маркетингової комунікації.

Ефективна CRM система має маркетинговий і орієнтований на клієнта підхід. Вона містить дані про стан ринку, інформацію про конкурентів та будь-яких інших зацікавлених сторін цільового ринку. Окрім того, така система розширює можливості компанії шляхом включення історичних записів передпродажної діяльності, прогнозів продажів, підтримки маркетингових кампаній і надання допомоги для підходу до сегментації ринку з різних можливих аспектів.

Опитування показали, що залучення нових клієнтів обходиться в рази дорожче, ніж утримання існуючих:

- 20% існуючих клієнтів компанії приносять 80% прибутку;
- клієнти, які вносять решту 20% доходу компанії, зазвичай використовують 60-70% ресурсів.

Ці загально визнані факти підтверджують, що ми повинні приділяти більше уваги існуючим клієнтам.

Хороша CRM система повинна допомагати компанії в обслуговуванні клієнтів. Після продажу бізнес має прагнути отримати зворотній зв'язок від клієнтів, вміло керувати будь-якими скаргами та забезпечити максимальне задоволення клієнтів. Завдяки такому підходу компанія може:

- утримати клієнтів;
- відновити зв'язок із втраченими клієнтами;
- продавати нові продукти існуючим клієнтам;
- залучати нових клієнтів.

CRM – це скоординоване управління усіма внутрішніми бізнес-процесами у свідомий та організований спосіб з фокусом на клієнтах. Якщо підійти до цього з дещо ширшої точки зору, то це скоріше свого роду філософія, якої компанія повинна дотримуватись у стосунках з клієнтами. Крім цього, CRM є орієнтованим на клієнта керівним принципом, який об'єднує діяльність компанії від першого контакту з клієнтом і до тих пір, поки він не стане клієнтом, що залишається. Відповідно, система CRM визначається як інтегрована ІТ-система, яка підтримує згадані вище функції за допомогою бази даних. Така система не є продуктом, а орієнтована радше на клієнта і належним чином вписується в організаційну та внутрішню структуру компанії, насамперед у процеси маркетингу та продажів.

CRM системи призначені для тих підприємств, що:

- мають справу із замовниками, клієнтами;
- бажають усунути проблеми з потоком інформації шляхом звільнення працівників;
- бажають точніше побачити інформацію про прогнозований дохід;
- хотіли б отримувати детальну інформацію про діяльність і продуктивність відділу продажів;

- хочуть підтримувати довгострокову задоволеність і лояльність своїх клієнтів;
- бажають запобігти втраті існуючих клієнтів;
- хотіли б ефективно проводити маркетингові кампанії;
- хотіли б розуміти та бути в курсі ринкових та споживчих тенденцій.

Як і будь-яка інша система, CRM система має ряд недоліків, серед яких:

- оптимізація та зміна бізнес-процесів;
- співробітникам не завжди подобається той факт, що їхню діяльність контролюватимуть чи оцінюватимуть її ефективність;
- дані клієнта потрібно записувати та оновлювати, вони не з'являються самі по собі.

Основними модулями системи являється реєстрація інформації клієнта, дані про продажі, дані про замовлення за певний період часу, управління кампанією та задоволеність клієнта.

Більшість функцій CRM системи базуються на збережених даних клієнтів. Підсистема, що відповідає за реєстрацію інформації про клієнта, записуватиме основні відомості про клієнтів, такі як ім'я, адреса, контактна інформація, пов'язані контакти, а також типові маркетингові дані: інтереси, чисельність працівників, річний дохід тощо. Ця інформація може бути отримана від різних постачальників або записана вручну.

Підсистема, що відповідає за дані про продажі, а також за процес зацікавленості клієнта в продукції чи послугі до моменту оформлення замовлення, веде облік продажів та проектів. Процес зацікавленості клієнта можна назвати керуванням потенційними клієнтами. Під час цього процесу буде здійснюватися багато нецифрових комунікацій і взаємодії між бізнесом і клієнтом. Загалом, веденням та оновленням реєстру продажів займається декілька працівників. Керівники компанії можуть у будь-який час побачити прогнози продажів та їх поточний рівень. Це дозволяє оцінити успішність продажів і завдяки цьому дохід стає більш передбачуваним. У випадку невдалого продажу причини невдачі можуть бути досліджені та проаналізовані, що має

дати компанії певне розуміння того, як уникнути повторення помилок у майбутньому.

Підсистема, що відповідає за збереження даних про замовлення, дозволяє компанії слідкувати за точними витратами клієнтів, а також надає інформацію, на що саме витрачались клієнти. Кмітливі фірми приділятимуть трохи більше уваги тим клієнтам, які витратили найбільше. За допомогою цієї підсистеми ми можемо відстежувати витрати клієнтів, кількість закуплених товарів, періоди часу і тому подібні дані. Окрім цього, підсистема, що аналізує дані про замовлення, також надає можливість сприяти перехресним продажам. Це може стати основою для розрахунку прибутковості протягом життєвого циклу клієнта.

Керівництво кампанією в системі CRM підтримує різноманітні кампанії прямого маркетингу, а саме: поштову розсилку, події, електронну пошту, телефон тощо. Це дає змогу бізнесу визначити точну цільову групу кампанії та записувати відгуки, отримані під час і після кампанії. Цю інформацію можна обробляти, фільтрувати, сортувати, організовувати та аналізувати. Це допоможе надавати клієнтам персоналізовані послуги та продукт і може стимулювати розробку продукту за ініціативою клієнта. Слід зазначити, що добре побудована система CRM повинна давати точний аналіз і розуміння ресурсів, типу управління чи підходу, що необхідні для охоплення конкретного ринку чи його сегменту. Однак ця тема належить до сфери інтелектуального аналізу даних, яку я не розглядатиму у даній роботі.

Щоб зберегти клієнтів та зменшити кількість уже втрачених, важливо розуміти що таке задоволеність клієнта, знати яким чином її можна виміряти після використання, споживання наших послуг [2]. Клієнти зазвичай не скаржаться, а просто вирішують піти. Ось чому вкрай важливо активно відстежувати рівень задоволеності клієнтів, і де це необхідно, звертати увагу на факт незадоволеності до того, як клієнти підуть і знайдуть інше рішення своєї проблеми. Опитування щодо задоволеності клієнтів можуть просуватись у двох напрямках:

- активне дослідження та застосування своєчасного конкретного засобу правового захисту для конкретної скарги клієнта;

- внутрішні бізнес-процеси та їх ресурси.

Під час досліджень скарг клієнтів важливо побачити точну причину невдоволення та зафіксувати точний характер рішення і його кінцевих термін. Важливо також визначити особу, яка буде відповідальною за виконання. У другому випадку, коли джерело невдоволення всередині організації, все трішки складніше. Завдання довгострокового мислення менеджменту полягає в аналізі результатів і у підході до цього питання раціоналістично, а також, якщо це можливо, слід бути готовими дещо змінити внутрішні бізнес процеси.

Отже, CRM рішення можна розуміти як комбінацію або набір інструментів, за допомогою яких організації можуть аналізувати минулі комунікації та планувати майбутню взаємодію з клієнтами. Важливість управління взаємовідносинами з клієнтами підкреслюється тим фактом, що такі рішення дозволяють організаціям збирати відгуки користувачів з різних каналів, таких як канали соціальних мереж, електронна пошта, веб-сайти, живий чат тощо [3]. Це допомагає компаніям зрозуміти, як клієнти взаємодіють із їхніми послугами та які очікування у даних клієнтів.

Програмне забезпечення CRM стало необхідним для будь-якого бізнесу, який хоче істотно розвиватись. CRM рішення допомагає створити привабливу взаємодію з потенційними клієнтами та перетворює їх на клієнтів. Цей процес перетворення від потенційного клієнта до повноцінного клієнта може бути досить складним і обтяжливим для команди з продажів. Тому багато організацій впроваджують у свій бізнес такого роду програмну систему для максимального підвищення продуктивності та конкурентоздатності. Більшість CRM рішень можна адаптувати практично до будь-якої галузі та повністю адаптувати до специфікацій власника бізнесу. Існують різні переваги такої програмної системи для різних галузей промисловості. У готельній індустрії CRM буде корисна тим, що надаватиме можливість автоматично призначати запити з кількох джерел певним командам, миттєво сповіщати на мобільний телефон про потенційних клієнтів, а також надсилати електронні листи, переважно із нагадуваннями, за попередньо встановленими шаблонами і, звичайно ж, створювати рахунки-

фактури. Туристична індустрія також стає у чергу із необхідністю консолідувати потенційних та наявних клієнтів.

З усіх секторів, які були змінені та вдосконалені завдяки CRM системі, можливо жоден не зазнав таких глибоких змін, як галузь охорони здоров'я. Частково причиною цього у США є оцифровка, яка була необхідною як ключовий компонент Закону про захист пацієнтів і доступне лікування 2010 року. Масштабні програми субсидій одним махом дозволили приватним медичним практикам, лікарням і клінікам по всій країні застосовувати цифрове ведення медичних записів і історію лікувань пацієнтів. Сьогодні компанії в сфері здоров'я використовують CRM рішення як універсальний набір інструментів для управління всіма потребами пацієнтів. Вони використовують такі системи, щоб керувати зустрічами та розкладами, координувати декілька одночасних видів догляду та навіть надсилати нагадування своїм пацієнтам, коли їм, власне, потрібно пройти профілактичні візити. Окрім того, дані, які генерує CRM система у сфері охорони здоров'я, використовуються для живлення цілого нового покоління рішень автоматизації на базі штучного інтелекту, які можуть робити усе, починаючи від обробки складних попередніх авторизацій страхування до планування персоналу для притоку пацієнтів [4]. Коротше кажучи, технології революціонізують лікування пацієнтів і водночас знижують витрати.

Агентствам нерухомості потрібне програмне забезпечення CRM для нерухомості, аби краще обслуговувати своїх клієнтів, включно з орендодавцями, орендарями та продавцями. Для таких цілей досить важливим є своєчасне розміщення списку об'єктів нерухомості та своєчасній розсилці нагадувань зацікавленим покупцям. Отож, для агенцій нерухомості можна виділити декілька причин, чому слід використовувати дану програмну систему:

- перехід від введення вручну до автоматизованого введення та збереження даних;
- швидші відповіді на клієнтські запити, адже програмна система CRM для нерухомості зазвичай має реалізовані автовідповідачі, що оперативно відповідають на поширені запити;

- взаємодія з наявними клієнтами, адже програма здатна надсилати електронні листи, інформаційні бюлетені та повідомляти про нові послуги і пропозиції;
- інтеграція з декількома каналами, оскільки CRM дає можливість відстежувати клієнтські запити у кількох каналах, включаючи соціальні мережі, а також сповіщати компанію, коли приходить запит;
- розширені можливості пошуку програмного забезпечення допомагають торговцям нерухомістю знаходити відповідні будинки для різних клієнтів із гігантської купи даних про доступні об'єкти;
- маркетингові кампанії, які дуже легко створювати в соціальних мережах і електронній пошті.

CRM для банківської діяльності є незамінним інструментом у тому випадку, якщо потрібно підтримувати теплі і продуктивні відносини з клієнтами. Банківські послуги необхідні кожному, коли справа доходить до управління своїми фінансами. Комплексна оцінка клієнтського програмного забезпечення CRM у банківській справі добре інтегрується з іншими банківськими рішеннями та дає змогу вести облік кожної діяльності власника рахунку. Це стає в нагоді особливо, коли особа подає заявку на отримання позики й потрібен аналіз її фінансових звичок.

Відповідне рішення CRM допомагає надавати клієнтам персоналізовані банківські послуги прямо біля їхнього порогу. Це включає в себе такі послуги, як внесення або зняття грошей, подання заявки на позику та багато іншого. Також програмне забезпечення CRM зберігає всю інформацію в одному місці, звідки до неї можна отримати доступ і змінити.

Значення CRM можна зрозуміти з того факту, що дана система надає корисну інформацію, пов'язану з успіхом ваших рекламних кампаній, поведінкою клієнтів і сферою вдосконалення. Відстеження маркетингових стратегій завдяки CRM стає легшим, оскільки ця система допомагає банкам стежити за каналами залучення та створювати докладні звіти для вивчення поведінки клієнтів.

Єдина причина, чому електронна комерція не є кращою за індустрію охорони здоров'я з точки зору змін, спричинених CRM системою та автоматизацією, полягає у тому, що це цифрова галузь, яка легше адаптується до технологій. Наприклад, веб-сайти B2C завжди збирали дані про клієнтів, тому найбільша зміна полягала в тому, як ці дані використовуються і як ними керують. На сьогоднішній день, операції електронної комерції використовують платформу CRM для обробки всього: інформації про повторне залучення клієнтів до звітності до аналітики на сайті. Вони також можуть надати складні моделі сегментації клієнтів на основі безлічі точок даних, які можуть, наприклад, ідентифікувати особу як відвідувача сайту, навіть якщо така особа маскує свою IP-адресу, намагаючись залишитись анонімним. Одна з основних функцій електронної комерції полягає у тому, аби звертатись до клієнтів у потрібний час і створювати унікальний досвід. Цей підхід передбачає побудову надійної між каналної комунікації для залучення нових клієнтів та взаємодії з уже наявними. CRM у роздрібному бізнесі дозволяє клієнтам стежити за своїми замовленнями, їх рухом і датою доставки в режимі реального часу. CRM у цьому випадку дозволяє вивчати історію покупок клієнтів, а потім надавати їм знижки та пропозиції. Завдяки наявності такої програмної системи в електронній комерції відділи продажів можуть автоматизувати повторювані завдання, такі як відповіді на запити клієнтів. Це залишає їм більше часу для продуктивніших завдань. Використовуючи CRM у роздрібній торгівлі, ми можемо отримати ноу-хау про найпопулярніші товари та їх наявність на складі. Це допомагає компанії поповнити їх запаси, якщо кількість цих товарів падає нижче певного критичного рівня. Справжня важливість даної програмної системи виходить на перший план, коли організації потрібно задовольнити скарги клієнтів щодо конкретних замовлень. Якщо робити це за допомогою CRM, то в певному сенсі це допоможе підвищити лояльність клієнтів до бренду організації.

Важливість CRM також усвідомлюється, коли керівники кол-центру мають клієнта на дзвінку і їм важко отримати відповідну інформацію. Програмна система CRM допомагає отримати повне уявлення про попередню взаємодію клієнта. Потреба в такій системі також виникає, коли компанія хоче

автоматизувати свої повторювані завдання та зосередитися на кращому обслуговуванні клієнта. Рішення CRM надає менеджерам декілька інструментів для моніторингу продуктивності їхньої команди за такими параметрами, як обсяг оброблених викликів та багато іншого. Також є можливість розділяти запити клієнтів на основі пріоритету. Це допомагає оптимізувати бізнес-процес і спочатку вирішувати критичні запити. Головне значення CRM системи у кол-центрах полягає у тому, що вона автоматично направляє вхідні запити до наступного доступного керівника. Один керівник, звичайно що, може передавати запити іншому. Така система також дозволяє керівникам кол-центру ініціювати різноманітні дзвінки, включно із запитами перед продажем, авторизацією повернення товарів, обробкою замовлень та обслуговуванням, що здійснюється після продажу продукту.

Іншим сектором, який використовує CRM та автоматизацію з великим ефектом, є юридична галузь. Для юристів час – гроші. Американська асоціація юристів повідомила, що виставляють рахунки в середньому адвокати лише за 1.9 години з кожних 8 відпрацьованих годин. Вивчаючи відсутні 6 годин, вони виявили, що 48% цього часу витрачається на адміністративні завдання, а 33% використовується на спроби отримати новий бізнес [5]. Ось чому найуспішніші адвокати використовують технології автоматизації маркетингу. Вони також виявили, що чат-боти та проста у використанні CRM програмна система допомагають юристам і співробітникам повернути багато годин втраченого часу, надавати кращі послуги та належним чином відслідковувати нових потенційних клієнтів. Дослідження підтверджують, що той, хто першим відгукнеться на потенційного клієнта, має набагато більше шансів отримати цього клієнта. Компанії, які відповідають на запит про продаж протягом 1 години, мають майже в 7 разів більшу імовірність отримати запит, порівняно з тими, хто намагався зв'язатися з клієнтом навіть через годину. Системи CRM значно спрощують керування мережами джерел перенаправлення та наявними клієнтами, зберігаючи важливі дані та нагадуючи правильній особі про те, щоб слідкувати за потенційними клієнтами. Кожен, хто знайомий з роботою типової юридичної фірми, може сказати, що це набір функцій на вагу золота. Оскільки більшість

юристів отримують свій дохід через погодинну оплату, кожна хвилина, витрачена на роботу не з клієнтами, коштує їм грошей. Автоматизація скорочує це, дозволяючи юристам доручати повторювані завдання машині, як-от відстеження оплачуваних годин або планування зустрічей, таким чином звільняючи свій час на інші завдання.

Якщо говорити про сферу виробництва, CRM допомагає прогнозувати попит, що може мати вирішальне значення для компаній, що займаються високими технологіями та електронікою. Інтегрований модуль CRM може порівнювати дані, зібрати в різних відділах компанії-виробника, з історичною інформацією, пов'язаною із замовленнями, для розробки досить точних прогнозів щодо попиту на продукцію. Маючи цю інформацію, компанії можуть швидше поставляти продукцію на ринок і випереджати конкурентів. Дефектна продукція є основною причиною незадоволеності клієнтів для компаній-виробників. Для успіху на ринку недостатньо розробляти продукти найвищого класу, якщо ланцюг постачання неефективний, то уся важка робота буде витрачена даремно. Програмне забезпечення CRM, створене відповідно до специфіки виробничої компанії, може надати чітку картину того, як працює ланцюг розподілу. Це може допомогти виявити потенційні слабкі місця та визначити, як швидко вивести продукт на ринок.

Останніми роками у світовій індустрії фітнесу спостерігається бум, оскільки все більше і більше людей починають розуміти вплив свого способу життя на їхнє здоров'я та благополуччя. Загалом, це сектор, який зараз коштує близько 87 мільярдів доларів щорічно у всьому світі і який виявився благодатним ґрунтом для CRM та технологій автоматизації. Оскільки більшість підприємств у цьому просторі орієнтовані на обслуговування клієнтів з мінімальною кількістю персоналу, переваги є очевидними та глибокими. CRM рішення виявилось блогом для зусиль з утримання клієнтів, які є ключем до прибутковості у галузі з історично низьким рівнем залучення [6]. Автоматизація також дає змогу розвантажити серверні завдання, щоб звільнити персонал для безпосередньої роботи з клієнтами, що є першочерговим фактором запобігання

скачування членства та забезпечення високого рівня задоволеності клієнтів без збільшення витрат на робочу силу.

Сектор освіти також ефективно використовує інструменти CRM для покращення стандартів навчання та розвитку кращого спілкування між батьками та викладачами. Використовуючи CRM в освіті, заклади можуть легко керувати всією мережею батьків, учителів і учнів. Це також важливо для легкого керування процесом вступу, тримаючи батьків в курсі різних етапів реєстрації. Тобто CRM рішення діє як центральна база з даними, що містить інформацію про кожного студента певної організації. Ця інформація автоматично оновлюється в міру проходження учнем класів. Це корисно для вищих навчальних закладів для відстеження студентів, зацікавлених у навчальному курсі, який вони пропонують. За допомогою таких програмних систем університети, коледжі можуть зв'язуватися з майбутніми студентами, вирішувати їхні запити, викладачі матимуть можливість легко завантажувати тести та завдання, вирішувати студентські організаційні проблеми та навіть призначати дистанційно батьківські збори. Використовуючи програмне забезпечення CRM в навчальних закладах, можна своєчасно надсилати нагадування про оплату за допомогою різних засобів, таких як SMS чи пошта. Незалежно від того, який спосіб спілкування використовував студент, навчальна установа може відповісти йому поштою, текстовим повідомленням або телефоном, не виходячи з інтерфейсу CRM системи.

Отож, підсумовуючи, можна сказати, що CRM рішення зараз популярніші, ніж будь-коли. Основною причиною їхнього застосування в усіх галузях є те, що ці системи допомагають розвивати повноцінні, надійні відносини з клієнтами та завжди надавати їм послуги найкращої якості [7].

1.2 Постановка задачі

Після проведеного аналізу предметної області можна сміло стверджувати, що інформації достатньо, а сумнівів а актуальності на сьогоднішній день даної теми немає. Для того, щоб створити список варіантів використання і, відповідно,

перелік сутностей можна застосувати відомий метод мозкового штурму, який полягає у інтенсивному роздумі і абстрактному викладенні ідей, що спадають на думку за певний виділений час для цього. В результаті такого методу ми можемо отримати перелік імовірних варіантів використання, які можна вважати як ті, що несуть певну цінність для програмної системи і являються частиною її реалізації, оскільки немає сенсу у тому, аби у повному обсязі і у повній мірі реалізовувати усі ідеї із такого списку. Для нас буде краще обрати декілька основних і вміло побудувати взаємозв'язок між ними, щоб в кінцевому результаті побачити логічно структуровану систему.

Основною метою, яку я поставила перед собою, виконуючи дану магістерську роботу, є розробити веб систему, яка буде призначена для управління та контролю взаємодії з потенційними клієнтами. Дана система буде виконувати роль CRM системи і включатиме можливість:

- реєстрації нових користувачів системи;
- зміни паролю, особистої інформації та прикріплення фото для користувачів системи;
- додавання нових потенційних клієнтів;
- редагування потенційних клієнтів;
- видалення потенційних клієнтів;
- створення додаткові динамічні поля з атрибутами різного типу для кожного потенційного клієнта;
- редагування та видалення динамічних полів;
- переглядання графіку, який показуватиме залежність тих чи інших показників.

Значною перевагою даної системи має бути максимальна зручність використання, тому необхідно враховувати той факт, що користувацький інтерфейс має бути зрозумілим і простим. Адаптивність веб системи до будь-якого типу гаджета має бути забезпечена у частковій або ж у повній мірі, оскільки увага буде сконцентрована саме на серверній частині. Не варто забувати і про підключення бази даних, де і зберігатиметься уся інформація,

оскільки саме база даних дасть змогу пришвидшити процес опрацювання запитів та роботу з величезною кількістю даних.

Якщо розглядати набір сутностей, які мають бути присутніми у даній системі, то варто виділити простого користувача, який буде мати доступ до основної функціональності системи, оскільки це, скоріш за все, будуть працівники компанії, які безпосередньо відповідатимуть за роботу, пов'язану із налагодженням взаємовідносин з клієнтами, а також будуть здійснювати контроль за процесом комунікації з клієнтами. Такі користувачі не матимуть доступ лише до тієї частини програми, де буде висвітлена інформація про усіх користувачів системи. Інформація про користувачів буде доступною лише адміністратору системи.

Однією із незвичайних частин програмної системи буде динамічне поле, оскільки, у даній системі буде виділено немалу частку функціоналу для цього задуму. Ідея полягає у тому, щоб реалізувати можливість створювати набір динамічних полів для кожного потенційного клієнта в залежності від інформації, яку ми б хотіли про нього зберегти та зареєструвати у системі. Отож, якщо, допустимо у клієнта є історія замовлень, які він здійснює і відповідно вподобання, то ми можемо створити відповідне поле для такої інформації і система буде пам'ятати, що конкретно за таким клієнтом зберігається інформація про специфічні вподобання і працівники компанії будуть мати можливість звернути на це особливу увагу і прийняти правильне рішення щодо типу комунікації з клієнтом або ж висновок щодо пропозицій для нього. Або ж, до прикладу, інша ситуація, у якій можна спостерігати що у клієнта є велика кількість номерів, які можна класифікувати за типом: домашній номер, номер компанії та інші. Для такого випадку є можливість створити відповідне поле для різного виду номерів і зберігати у повній мірі контакти клієнта та за необхідності звертатись, використовуючи потрібний контакт. Ще однією сутністю є компанія, до якої належить той чи інший клієнт.

Таким чином, проаналізувавши вимоги до програмної системи, можна вважати, що набір функцій складено, мета даної роботи зрозуміла та логічно обгрунтована, а сама ідея є актуальною.

1.3 Обґрунтування вибору середовища розробки програмної системи та мови програмування

Така мова програмування загального призначення, як Java, здатна підтримувати широкий спектр програм. Не кажучи вже про те, що над Java активно працюють і щодня вдосконалюють. Крім того, існує багато бібліотек з відкритим вихідним кодом і пропрієтарних бібліотек для підтримки Java у цьому відношенні. Розробка веб-додатків стала неминучою потребою для підприємств усіх типів і розмірів, незалежно від їх ціннісних пропозицій. Тим не менш, існує кілька способів розробки веб-додатків, найпопулярнішим з яких є використання фреймворків. Фреймворки для веб розробки набули популярності за останні кілька років, оскільки вони забезпечують простий підхід до розробки користувацьких веб програм. Крім того, розробники отримують дедалі чутливіший та інтуїтивно зрозуміліший інтерфейс програмної системи, який полегшує програмування підтримуваними мовами. Перш за все, фреймворки забезпечують спрощене представлення підтримуваних мов, що дозволяє розробникам швидко писати код та й загалом виконувати роботу швидше.

Якщо ми подивимося на кілька різних веб-додатків, то побачимо, що є багато дубльованих функцій. Багато з них мають автентифікацію користувачів, сторінки, які відображаються, вони підключаються до баз даних, у них є профілі, які ви можете переглядати, усі вони мають стилізовані канали інформації тощо. Можна написати всю цю логіку самостійно щоразу, коли потрібно створити нову програмну систему; однак є також багато доступних інструментів, які можуть зробити це замість нас. Крім того, у більшості випадків, коли ми створюємо програми, ми хочемо відформатувати наш код певним чином, щоб допомогти нам стати ефективнішими та допомогти іншим розробникам зрозуміти наш код.

Веб-фреймворки допомагають нам створити структуру в наших програмах і надають нам додаткові функції, які ми можемо додати до них без зайвої праці. Фреймворки дають нам початок, щоб ми могли зосередитися на функціях, а не на деталях конфігурації. Код читається набагато більше разів, ніж пишеться, і дуже важливо, коли ми працюємо з іншими розробниками, щоб вони знали, де

пишеться код для певних завдань, щоб вони могли використовувати його у власному коді. Фреймворки також мають конвенції щодо написання та структури коду, які стандартизують те, як розробники пишуть свій код. Більшість веб-сайтів мають дуже схожу, якщо не ідентичну, структуру. Мета фреймворків полягає в тому, щоб забезпечити загальну структуру, щоб розробникам не доводилося переробляти її з нуля, і вони могли повторно використовувати наданий код. Таким чином фреймворки дозволяють нам скоротити велику частину роботи та заощадити багато часу. На відміну від багатьох інших інструментів розробки, фреймворки забезпечують попередньо визначену кодову базу та вказівки для безперервної веб-розробки. Водночас це прискорює процес розробки та скорочує час виходу на ринок за рахунок скорочення життєвого циклу розробки програми. Розробка веб-додатків з використанням фреймворків зосереджується в першу чергу на покращенні якості та читабельності коду для зручності розробників. Окрім найефективнішого способу кодування та розробки веб-додатків, фреймворки полегшують налагодження та обслуговування програмного забезпечення. Кілька веб-фреймворків надають вбудовану підтримку для налагодження та тестування якості, що дозволяє розробникам виконувати модульні тести одночасно. Тим не менш, для тестування програмного забезпечення високого рівня потрібні інженери з контролю якості, які володіють навичками використання таких інструментів, як Selenium, JMeter і Katalon Studio. Веб-фреймворки зосереджені головним чином на покращенні читабельності коду та можливості повторного використання, забезпечуючи тим самим найшвидший і найефективніший спосіб розробки самопідтримуваних веб-додатків. Крім того, розробники отримують адаптивне середовище кодування та розширені функції, як-от гаряче перезавантаження, живе перезавантаження, які зрештою призводять до швидшої веб-розробки. Завдяки багатьом таким вбудованим функціям розробники можуть швидко виконувати складні завдання, пов'язані з кодом, на виконання яких за інших умов знадобилися б години. Подібним чином веб фреймворки зосереджуються на зменшенні складності розробки шляхом автоматизації таких критичних завдань, як кешування, відображення URL-адрес, керування сесіями

тощо. Окрім цього, фреймворки пропонують швидке створення прототипів, просте налагодження та зміни коду в режимі реального часу для реалізації спеціальних функцій.

Збільшення випадків кібератак викликало серйозні занепокоєння щодо безпеки веб-розробки. Однак використання фреймворків для розробки веб-додатків дає змогу захищати веб-додаток від поширених атак, таких як DDoS, впровадження SQL, і запобігає підробці даних. Це завдяки вбудованим функціям безпеки, які пропонують більшість веб фреймворків.

Spring можна назвати екосистемою продуктів, призначення якої - це допомогти розробникам Java. Це, по суті, величезний контейнер, що містить у собі такі компоненти як Spring Framework, Spring Data JPA, Spring Security, Spring JDBC, Spring Boot та інші.

Spring Boot - це інструмент із відкритим вихідним кодом, який полегшує розробникам створення автономних цифрових продуктів і готових до виробництва Spring додатків, включаючи програми Java і веб-сервіси [8]. Він використовує мікрофреймворк, що робить його найбільш корисним для створення мікросервісів для веб-сайтів і мобільних додатків. Спільнота Java прийняла Spring як свою основну основу для розробки додатків. Незважаючи на це, протягом останніх кількох років використання Spring стало значно складнішим через додавання нових функцій. Запуск нового проекту зазвичай передбачає проходження кількох етапів і витрати значного часу на це. Програма Spring Boot була розроблена, щоб допомогти людям не починати з нуля та заощадити час. Даний інструмент має велике число переваг, а саме:

- допомагає скоротити час, витрачений на розробку, а також підвищити загальну ефективність команди розробників;
- у комплект входять вбудовані HTTP-сервери, такі як Jetty і Tomcat, для тестування веб-додатків;
- всі компоненти готової до роботи програми Spring можуть автоматично налаштовуватися;

- надає конфігурацію за замовчуванням як для модульних тестів, так і для інтеграційних тестів, що полегшує розробку та тестування програм на основі платформи Java;
- допомагає усунути трудомістке написання шаблонного коду, анотацій і масштабних налаштувань XML, що значно економить час;
- легко інтегрувати Spring Boot з рештою екосистеми Spring;
- пропонує широкий вибір плагінів, які розробники програмного забезпечення можуть використовувати для легкої та ефективної співпраці з вбудованими базами даних і базами даних у пам'яті;
- можна легко та зручно підключатися до служб бази даних і черги, таких як Oracle, MySQL, MongoDB і ActiveMQ;
- пропонує адміністративну допомогу, дозволяючи керувати програмою за допомогою віддаленого доступу;
- забезпечує швидкий доступ до інтерфейсу командного рядка, який прискорює тестування та розробку програм Spring Boot.

Ніibernate — це рішення об'єктно-реляційного відображення з відкритим кодом на Java. Він легкий і усуває всі недоліки, з якими ми стикаємося під час роботи з JDBC. Ніibernate — це фреймворк у Java, який постачається з рівнем абстракції та обробляє реалізацію внутрішньо. Реалізації включають такі завдання, як написання запиту для операцій CRUD або встановлення з'єднання з базами даних. Тобто це, в основному, програмне забезпечення, яке забезпечує абстракцію для кількох технологій, таких як JDBC, сервлет тощо. Ніibernate розробляє логіку збереження, яка зберігає та обробляє дані для більш тривалого використання. Він легкий і є інструментом ORM, і, що найважливіше, має відкритий код, що дає йому перевагу над іншими фреймворками. Інструмент ORM спрощує створення даних, маніпуляції та доступ до них. Він внутрішньо використовує Java API для взаємодії з базами даних.

Перевагою легкої конструкції є значно менший пакет для встановлення. Ефективність підвищується, якщо не використовувати контейнер для виконання. Незважаючи на те, що Ніibernate може працювати з кількома технологіями

одночасно, це не означає, що Hibernate не може працювати окремо. Ми також можемо працювати в сплячому режимі самостійно, тобто без жодних технологій.

Hibernate має особливу природу, де йому не потрібно реалізовувати інтерфейси Hibernate API або розширюватися від класів Hibernate API, оскільки класи розробки додатків у режимі hibernate слабо пов'язані.

Найуспішніші веб-сайти – це ті, які виділяються, привабливі та запам'ятовуються тим, як вони зображують бренд. Аби приголомшити своїх користувачів, щоб вони купували у компанії, передавали послуги компанії друзям або ставали лояльними клієнтами. Коли сторінка зливається з натовпом, то компанія ризикує бути забутою. Володіючи знаннями щодо створення власного веб-сайту за допомогою HTML і CSS, можна виділитися з натовпу за допомогою автентичного, створеного вручну представлення бізнесу. HTML або мова розмітки гіпертексту — це мова, яка в основному використовується для створення базової основи веб-сторінок [9]. Разом із Javascript і CSS, HTML є однією з піонерських інструментів, який дозволяє розробникам створювати інтуїтивно зрозумілу веб-сторінку. CSS, або каскадні таблиці стилів, є ще одним відомим інструментом веб-розробки, який використовується для створення стильних елементів веб-сайту. Якщо HTML — це кістки тіла, то CSS — це шкіра, яка його покриває. Він використовується для кольору фону, стилю, макета, рамок, затінення – усіх важливих елементів дизайну, завдяки яким веб-сторінка виглядає вишуканою. CSS дає змогу відрізнити презентацію від вмісту, змінюючи дизайн і відображення елементів HTML. Основна причина, чому HTML і CSS не вважаються мовами програмування, полягає в тому, що вони лише визначають структуру та стиль веб-сторінки, яку створюють.

Дані — це все, а керування базою даних — найважливіша й критична річ у розробці програмного забезпечення. Дрібні помилки викликають велике лихо. Liquibase — це незалежна від бази даних бібліотека з відкритим кодом для відстеження, керування та застосування змін схеми бази даних. Liquibase в основному підтримує файли XML, YAML, JSON і SQL. Цей інструмент дозволяє легше відстежувати зміни в базі даних, працює більш плавно під час гнучкої розробки програмного забезпечення.

Для того, щоб полегшити роботу із стилями я обрала Bootstrap. Bootstrap - це величезна колекція зручних багаторазових фрагментів коду, написаних на HTML, CSS і JavaScript. Це також фреймворк розробки інтерфейсу, який дозволяє розробникам і дизайнерам швидко створювати повністю адаптивні веб-сайти. По суті, Bootstrap позбавляє нас від написання великої кількості коду CSS, даючи більше часу для розробки веб-сторінок.

Для того, аби представляти дані на клієнтській стороні, я використовуватиму Thymeleaf. Thymeleaf — це сучасний серверний движок шаблонів Java як для веб, так і для автономних середовищ. Він ґрунтується на концепції шаблонів: файлів шаблонів, які можна відкривати безпосередньо в браузерах і які все одно правильно відображаються як веб-сторінки. Spring Boot автоматично налаштовує Thymeleaf, коли знаходить залежність у файлі Maven POM. Метою Thymeleaf є надання стильного та добре сформованого способу створення шаблонів. Він заснований на тегах і атрибутах XML. Ці XML-теги визначають виконання попередньо визначеної логіки в DOM (об'єктній моделі документа) замість того, щоб явно писати цю логіку як код усередині шаблону. Це заміна JSP. Архітектура Thymeleaf дозволяє швидко обробляти шаблони, що залежить від кешування проаналізованих файлів. Він використовує найменшу можливу кількість операцій введення-виведення під час виконання.

Щодо середовища розробки, то я зупинилась на IntelliJ Idea, оскільки це середовище використовую уже кілька років і воно для мене є дуже зручним. Простими словами, IDE — це зручна частина програмного забезпечення, яка функціонує як текстовий редактор, налагоджувач і компілятор в одному [10]. Деякі IDE також дозволяють встановлювати в них плагіни, щоб надати ще більше можливостей. В IntelliJ є така можливість і я часто встановлюю плагіни для роботи із фреймворками, оскільки це полегшує процес написання коду. Вибір IDE зазвичай залежить від типу проекту, над яким працюють, а також різноманітних вимог середовища, таких як платформа, мова програмування, система контролю версій тощо. І, звичайно, потрібно також враховувати бюджет і особисті переваги. IntelliJ Idea стало для мене незамінним середовищем, оскільки я працюю з мовою програмування Java.

2 РОЗРОБКА МОДЕЛІ ТА ПРОЕКТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ

2.1 Вибір процесу розробки

Коли досліджено предметну область, виділено основні функції програмної системи, охарактеризовану бажану поведінку програмного забезпечення та прийнято рішення щодо конкретного середовища та технологій, варто подумати над тим, яку модель процесу розробки обрати.

Методологія розробки програмного забезпечення відноситься до структури, яка використовується для структурування, планування та контролю процесу розробки програмної системи [11]. Протягом багатьох років розвивалася велика різноманітність таких моделей процесу розробки програмного забезпечення, кожна з яких має свої власні сильні та слабкі сторони.

Методологія розробки програмного забезпечення — це спосіб керування проектом розробки програмного забезпечення. Метод розробки програмного забезпечення зазвичай стосується таких питань, як вибір функцій для включення в поточну версію, коли буде випущено програмне забезпечення, хто над чим працює та яке тестування проводиться.

На практиці кожна організація реалізує управління проектом розробки програмного забезпечення різними способами, які часто дещо відрізняються від одного проекту до іншого. Більшість сучасних процесів розробки можна нечітко описати як гнучкі. Інші методології включають водоспад, прототипування, ітераційну та поступову розробку, спіральну розробку, швидку розробку додатків та екстремальне програмування.

Традиційні методології розробки програмного забезпечення, такі як «водоспад», які мають окремі фази, іноді називаються методологіями життєвого циклу розробки програмного забезпечення (SDLC), хоча цей термін також можна використовувати в більш загальному вигляді для позначення будь-якої методології. Підхід «життєвого циклу» з окремими фазами відрізняється від гнучких підходів розробки програмного забезпечення, які визначають процес ітерації, але де проектування, конструювання та розгортання різних частин

можуть відбуватися одночасно. Одна методологія розробки програмного забезпечення не обов'язково підходить для використання в усіх проектах. Кожна з доступних методологій розробки програмного забезпечення найкраще підходить для конкретних типів проектів, виходячи з різних технічних, організаційних, проектних і командних міркувань. Кожна з основних методологій розробки програмного забезпечення розглядається в контексті бізнесу, програм, організації та технічного середовища.

Метод розробки програмного забезпечення Waterfall — це послідовний підхід до розробки, за якого розробка розглядається як постійний рух вниз (як водоспад) через кілька фаз, як правило:

- Аналіз вимог, результатом якого є специфікація вимог до програмного забезпечення;
- дизайн програмного забезпечення;
- реалізація;
- тестування;
- інтеграція, якщо є кілька підсистем;
- розгортання (або встановлення);
- технічне обслуговування.

Модель водоспаду — це традиційний інженерний підхід, застосований до розробки програмного забезпечення. Суворий каскадний підхід перешкоджає повторному перегляду та перегляду будь-якої попередньої фази після її завершення. Ця «негнучкість» моделі чистого водоспаду стала джерелом критики з боку прихильників інших більш «гнучких» моделей.

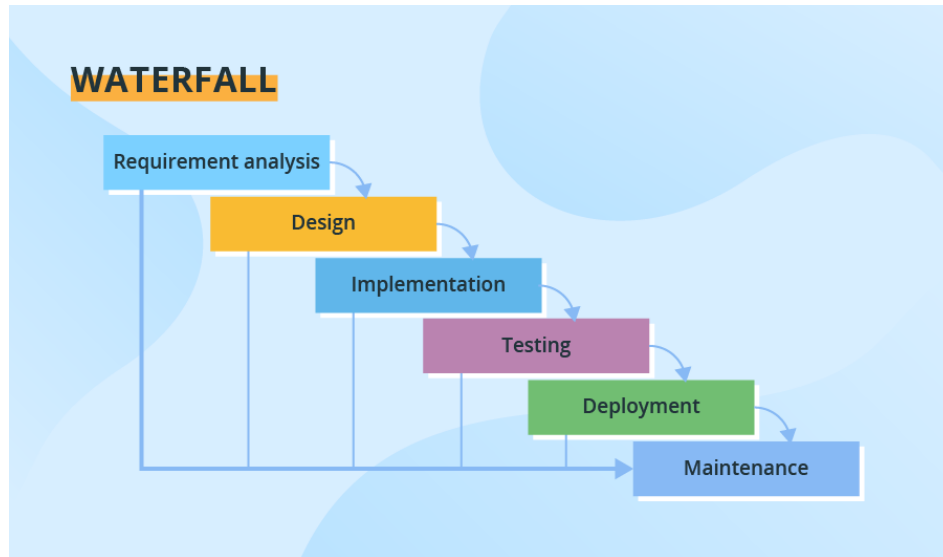


Рисунок 2.1.1 – Водоспадна модель розробки

Метод розробки програмного забезпечення Waterfall часто звинувачують у кількох великомасштабних урядових проєктах, які з часом перевищують бюджет і іноді не відповідають вимогам. За винятком випадків, коли це вимагається контрактом, модель водоспаду була значною мірою замінена більш гнучкими та універсальними методологіями, розробленими спеціально для розробки програмного забезпечення.

Ситуації, коли водоспадний метод розробки програмного забезпечення найбільш прийнятний:

- Проєкт спрямований на розробку пакетної системи на основі мейнфрейму або орієнтованої на транзакції.
- Проєкт великий, дорогий і складний.
- Проєкт має чіткі цілі та рішення.
- Немає тиску на негайне впровадження.
- Вимоги до проєкту можна викласти однозначно і вичерпно.
- Вимоги до проєкту є стабільними або незмінними протягом життєвого циклу розробки системи.
- Спільнота користувачів добре обізнана в бізнесі та застосуванні.
- Члени команди можуть бути недосвідченими.
- Склад команди нестабільний і очікується коливання.
- Керівник проєкту може не мати повного досвіду.

- Ресурси потрібно берегти.
- Існують суворі вимоги щодо офіційних схвалень на визначених етапах.

Прототипування програмного забезпечення — це діяльність зі створення прототипів програмних додатків, тобто неповних версій програмного забезпечення, що розробляється.

1.Процес створення прототипу програмного забезпечення включає наступні етапи:

2.Визначення основних вимог, включаючи бажану вхідну та вихідну інформацію. Такі деталі, як безпека, зазвичай можна ігнорувати.

3.Розробка початкового прототипу, який включає лише інтерфейс користувача.

4.Перегляд прототипу. Клієнти, включно з кінцевими користувачами, перевіряють прототип і надають відгук про можливі доповнення або зміни.

5.Перегляд і вдосконалення прототипу. За допомогою зворотного зв'язку можна покращити як специфікації, так і прототип. Можливо, знадобляться переговори щодо того, що входить до сфери дії контракту/продукту. У разі внесення змін може знадобитися повторення кроків №3 і №4.

Ситуації, коли метод розробки прототипного програмного забезпечення найбільш доречний:

- Проект спрямований на розробку онлайн-системи, що вимагає широкого діалогу користувача, або на менш чітко визначену експертну систему підтримки прийняття рішень.
- Проект є великим із багатьма користувачами, взаємозв'язками та функціями, де необхідно зменшити ризик проекту, пов'язаний із визначенням вимог.
- Цілі проекту не зовсім зрозумілі.
- Існує тиск щодо негайного виконання чогось.
- Функціональні вимоги можуть змінюватися часто і значно.
- Користувач не повністю обізнаний.
- Члени команди мають досвід (особливо, якщо прототип не є одноразовим).
- Склад команди стабільний.

- Керівник проекту досвідчений.
- Немає необхідності повністю мінімізувати споживання ресурсів.
- Жодних суворих вимог щодо схваленень на визначених етапах не існує.
- Аналітики/користувачі оцінюють наявні бізнес-проблеми ще до того, як розпочнуть проект.
- Інноваційні, гнучкі конструкції, які враховуватимуть майбутні зміни, не є критичними.

Основна ідея інкрементального методу розробки програмного забезпечення полягає в тому, щоб розробити систему шляхом повторюваних циклів (ітераційний) і меншими порціями за один раз (інкрементний), що дозволяє розробникам програмного забезпечення скористатися тим, що було отримано під час розробки попередніх частин або версій системи.

Навчання походить як від розробки, так і від використання системи, де можливі ключові кроки в процесі починаються з простого впровадження підмножини вимог до програмного забезпечення та ітеративно вдосконалюють версії, що розвиваються, доки не буде реалізовано повну систему. На кожній ітерації вносяться зміни в дизайн і додаються нові функціональні можливості. За допомогою інкрементального методу розробки програмного забезпечення продукт проектується, впроваджується та тестується поступово (щоразу додається трохи більше), доки продукт не буде завершено. Це передбачає як розробку, так і підтримку. Виріб вважається готовим, якщо він задовольняє всі його вимоги. Ця модель поєднує в собі елементи моделі водоспаду з ітеративною філософією прототипування.

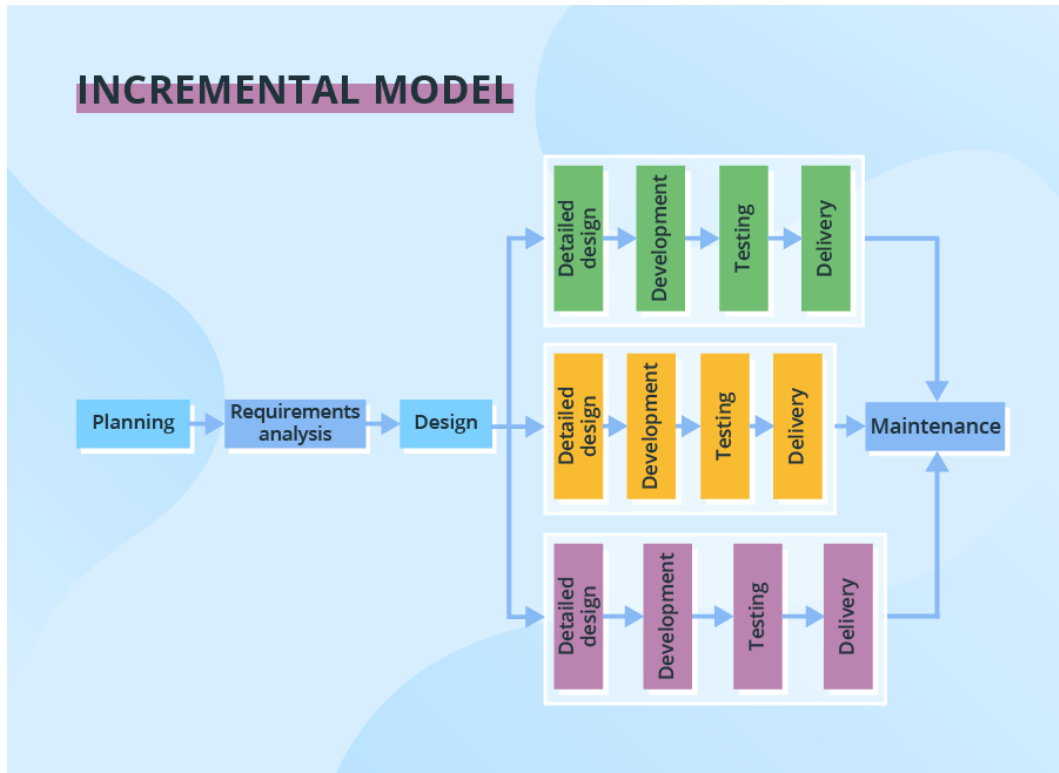


Рисунок 2.1.2 – Інкрементна модель розробки

Ситуації, коли метод інкрементної розробки програмного забезпечення найбільш прийнятний:

- Великі проекти, де вимоги недостатньо зрозумілі або змінюються через зовнішні зміни, зміну очікувань, зміни бюджету або швидку зміну технологій;
- веб-інформаційні системи (WIS) та системи, керовані подіями;
- передові програми.

Спіральна модель – модель розробки програмного забезпечення, орієнтована більше на управління ризиками, а не на щось інше. Ця модель розробки програмного забезпечення є однією з найважливіших методологій у бізнесі. По суті, це можна розглядати як спіраль над головою, де ви дивитеся на явища зверху в центрі коробки, яка розділена на чотири менші коробки. Кожна петля означає фазу, радіус — вартість, а швидкість прогресування ілюструється кутовими розмірами. Поділ фаз тут можна розбити на наступні чотири фази:

- Визначення цілей та визначення альтернативних рішень;
- виявлення та вирішення ризиків;

- розробка наступної версії продукту;
- огляд і план наступного етапу.

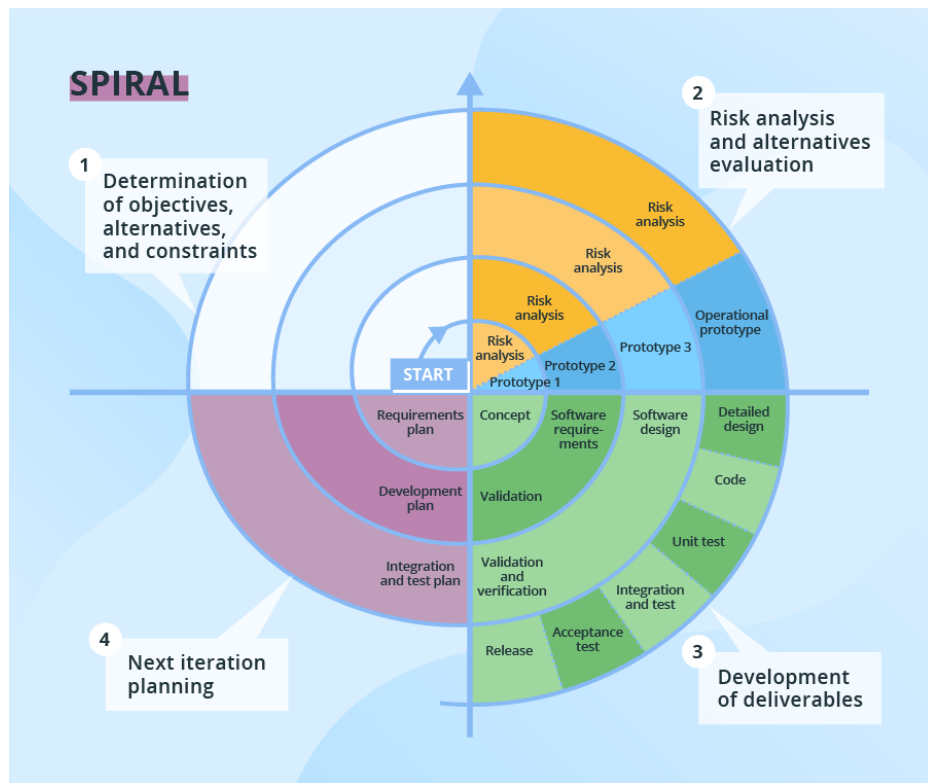


Рисунок 2.1.3 – Спіральна модель

Agile – найбільш широко використовувана модель розробки програмного забезпечення в галузі завдяки тому, що це неймовірно динамічний і гнучкий процес управління проектами. Незалежно від того, скільки людей працює над проектом, легко вести співпрацю просто через те, що легко адаптуватися до змін, які викликають ринок або клієнти. На відміну від багатьох інших моделей, гнучка методологія фактично народилася з іншої моделі. Вона мала бути більш універсальною і менш схильною до жорсткості, таким чином дозволяючи програмістам реагувати на мінливі вимоги за потреби. Серед найбільш важливих аспектів цієї конкретної моделі є Agile Manifesto, який для початку містить чотири фундаментальні цінності. По суті, це основа, яка робить гнучку методологію такою, якою вона є. Крім того, є також 12 ключових принципів, які служать якорями, що тримають проекти на правильному шляху, а саме:

- Задоволеність клієнтів завдяки ранній і безперервній доставці програмного забезпечення
- Пристосування до мінливих вимог протягом усього процесу розробки;
- Часта доставка робочого програмного забезпечення;
- Співпраця між бізнес-стейкхолдерами та розробниками протягом усього проекту;
- Підтримка, довіра та мотивація залучених людей;
- Забезпечити особисту взаємодію;
- Працююче програмне забезпечення є основним показником прогресу;
- Гнучкі процеси для підтримки постійного темпу розвитку;
- Увага до технічних деталей і дизайну підвищує маневреність;
- Простота;
- Самоорганізовані команди заохочують чудову архітектуру, вимоги та дизайн;
- Регулярні роздуми про те, як стати ефективнішим.

2.2 Розробка моделі предметної області

Візуалізація є ключовим способом забезпечення зрозумілості системи. Візуалізація абстрактних ідей дає людям змогу використати уяву, необхідну, щоб «побачити» систему. Саме моделювання дасть нам змогу продемонструвати поведінку та бажану структуру програмного забезпечення. Після опису цілей програмної системи ми отримаємо певний шаблон програмної системи, що дасть змогу у майбутньому не лише розширити її, але й мінімізувати ризики, пов'язані із переробкою чи частковими змінами або ж реалізацією.

Оскільки предметною областю даної роботи є CRM система, то можна виділити такі основні цілі даної програмної системи:

- 1) Залучення нових клієнтів використовуючи різні канали зв'язку.
- 2) Організована клієнтська база.
- 3) Аналіз залежності одних показників від інших та їхнє графічне представлення.
- 4) Можливість використання системи будь-яким видом бізнесу.

- 5) Автоматизація документообігу.
- 6) Авторизація в залежності від наданих прав доступу.
- 7) Можливість змінити пароль та інформацію профілю, включаючи медіа дані.
- 8) Можливість відновити акаунт, якщо забули пароль.

2.3 Розробка бізнес моделі

За допомогою моделі можна побачити майбутню програмну систему разом з усіма її компонентами та їх взаємодією наочно, тому дуже важливо спроектувати бізнес модель до початку реалізації програми, тобто до початку написання програмного коду. Не менш важливим є і аналіз вимог до програмного забезпечення, оскільки він дозволяє отримати результат у вигляді діаграми варіантів використання, що зазвичай виконується на мові UML [12]. Така діаграма покаже взаємодію системи і кінцевого користувача. Кожен варіант використання представляє собою окрему функцію програмної системи, яка буде доступною та зрозумілою для користувача. Як правило на діаграмі варіантів використання можна побачити сутності та варіанти використання (use cases). Сутностями можуть бути користувачі з різними правами доступу до тих чи інших функцій системи. Простіше кажучи, користувачі з різними правами доступу мають дозвіл користуватись певними функціями системи та бачити певний контент в залежності від прав доступу, наданих їм системою. Зазвичай існує один користувач, який виконує роль адміністратора системи, який має доступ до всіх функцій програми, а також може маніпулювати даними зареєстрованих користувачів у системі та інший користувач, який має роль звичайного користувача і обмежений доступ до програми. Тобто звичайному користувачеві, зазвичай, заборонений доступ до сторінки, де відображається інформація про усіх користувачів чи такі дані, які мають бути доступними лише адміністратору.

Було спроектовано наступну діаграму варіантів використання:

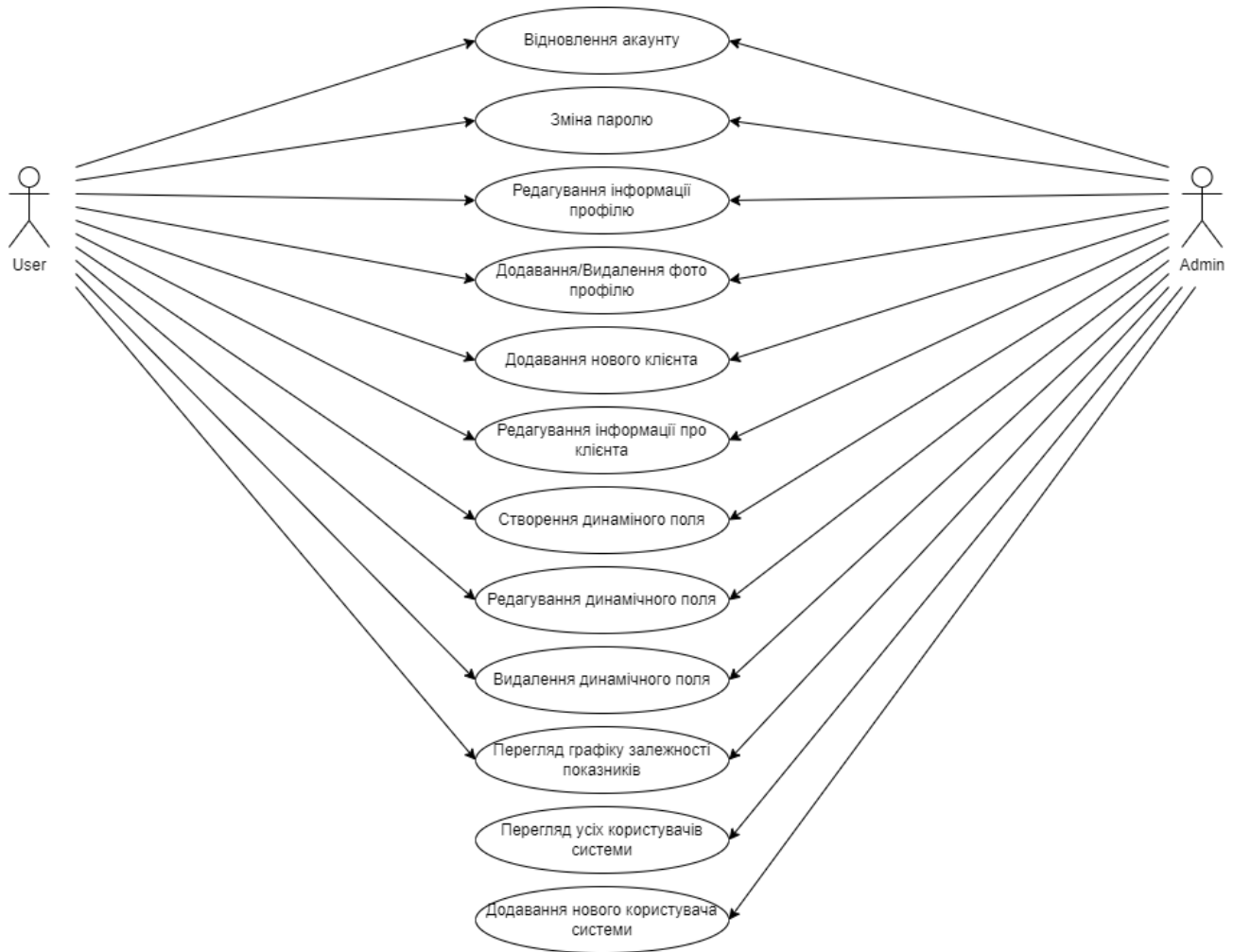


Рис. 2.3.1 – Діаграма варіантів використання

Як ми бачимо на рисунку 2.3.1 у програмній системі існуватимуть дві ролі: адміністратор і звичайний користувач. Користувачу доступні такі дії:

- відновлення акаунту;
- зміна паролю;
- редагування інформації профілю;
- додавання та видалення фото профілю;
- додавання нового клієнта;
- редагування інформації про клієнта;
- створення динамічного поля;
- редагування динамічного поля;
- видалення динамічного поля;
- перегляд графіку залежності показників.

Адміністратору доступні такі дії:

- відновлення акаунту;
- зміна паролю;
- редагування інформації профілю;
- додавання та видалення фото профілю;
- додавання нового клієнта;
- редагування інформації про клієнта;
- створення динамічного поля;
- редагування динамічного поля;
- видалення динамічного поля;
- перегляд графіку залежності показників;
- перегляд усіх користувачів системи;
- додавання нового користувача системи.

Адміністратор має доступ до усіх функцій системи, тобто може контролювати виконання тих чи інших робіт працівниками, виправляти результати, а також лише йому доступна функція додавання нового користувача системи. Ця функція дає можливість адміністратору вручну зареєструвати нового користувача програмної системи. Також адміністратор має змогу бачити усіх користувачів поточної компанії, тобто компанії, до якої вони належать. Користувачу недоступні функція перегляду усіх користувачів системи чи додавання нового, звичайний користувач може лише додавати потенційних клієнтів або ж нові динамічні поля.

2.4 Проектування архітектури

Після проведення аналізу предметної області, було зрозуміло, які сутності існуватимуть в програмній системі та атрибути цих сутностей.

На рисунку 2.4.1 представлено діаграму бази даних.

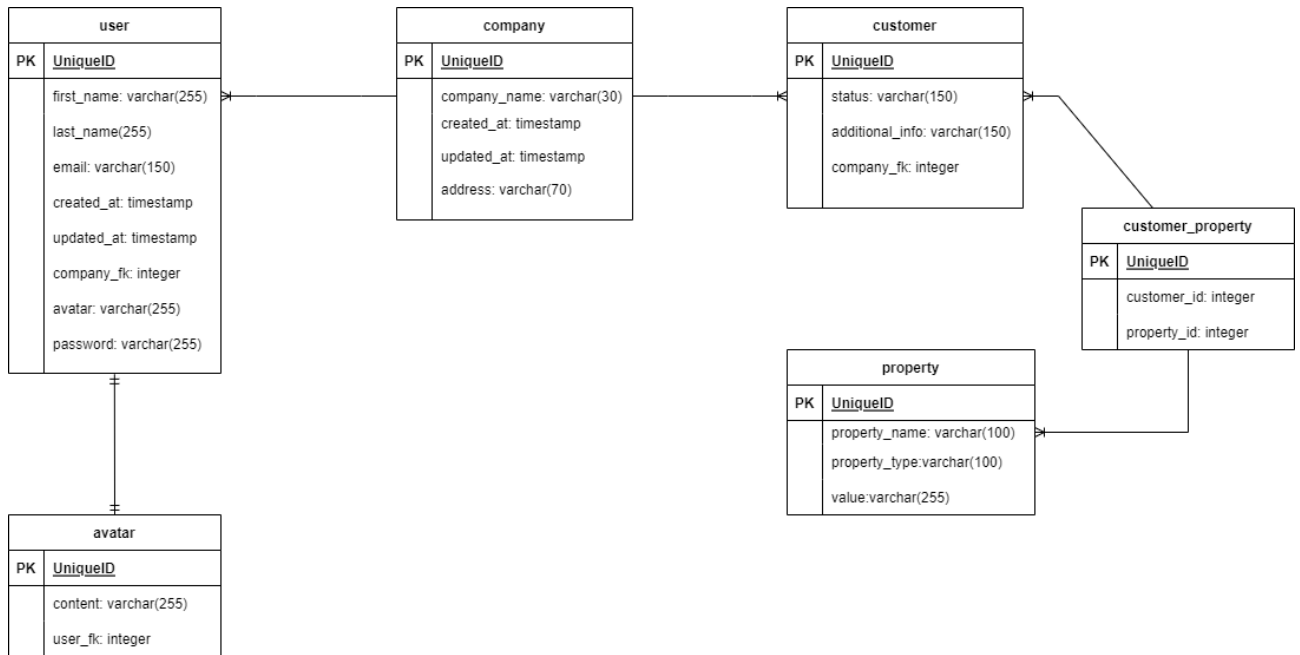


Рисунок 2.4.1 – Діаграма бази даних

Як видно з діаграми бази даних, існують такі таблиці:

- user – таблиця для збереження інформації про користувачів системи, яка відноситься до таблиці company як «багато до одного»;
- avatar – таблиця для збереження фото профілю користувача, яка має зв'язок «один до одного» з таблицею user;
- company – таблиця для збереження інформації про компанію;
- customer – таблиця для збереження інформації про клієнта, що відноситься до таблиці company як «багато до одного»;
- property – таблиця для збереження даних про динамічні поля, що відноситься до таблиці customer «як багато до багатьох»;
- customer_property – таблиця для реалізації зв'язку «багато до багатьох» для таблиць customer і property.

Отож, як показано на діаграмі бази даних, один користувач може мати одне фото профілю, належати одній компанії. Клієнт може належати одній компанії в даному випадку та може мати безліч динамічних полів, ті самі динамічні поля можуть належати іншим клієнтам.

3 КОНСТРУЮВАННЯ СИСТЕМИ

3.1 Реалізація ключових класів

У ході виконання роботи було розроблено основні класи для базової логіки програмної системи, які представлені на рисунку. Загалом ієрархія класів розпочинається із батьківських класів, які наслідуються дітьми, тобто похідними класами. Ієрархія будується зверху вниз, тобто батьківський клас завжди буде знаходитись вище.

На рисунку 3.1.1 зображено uml діаграму класів.

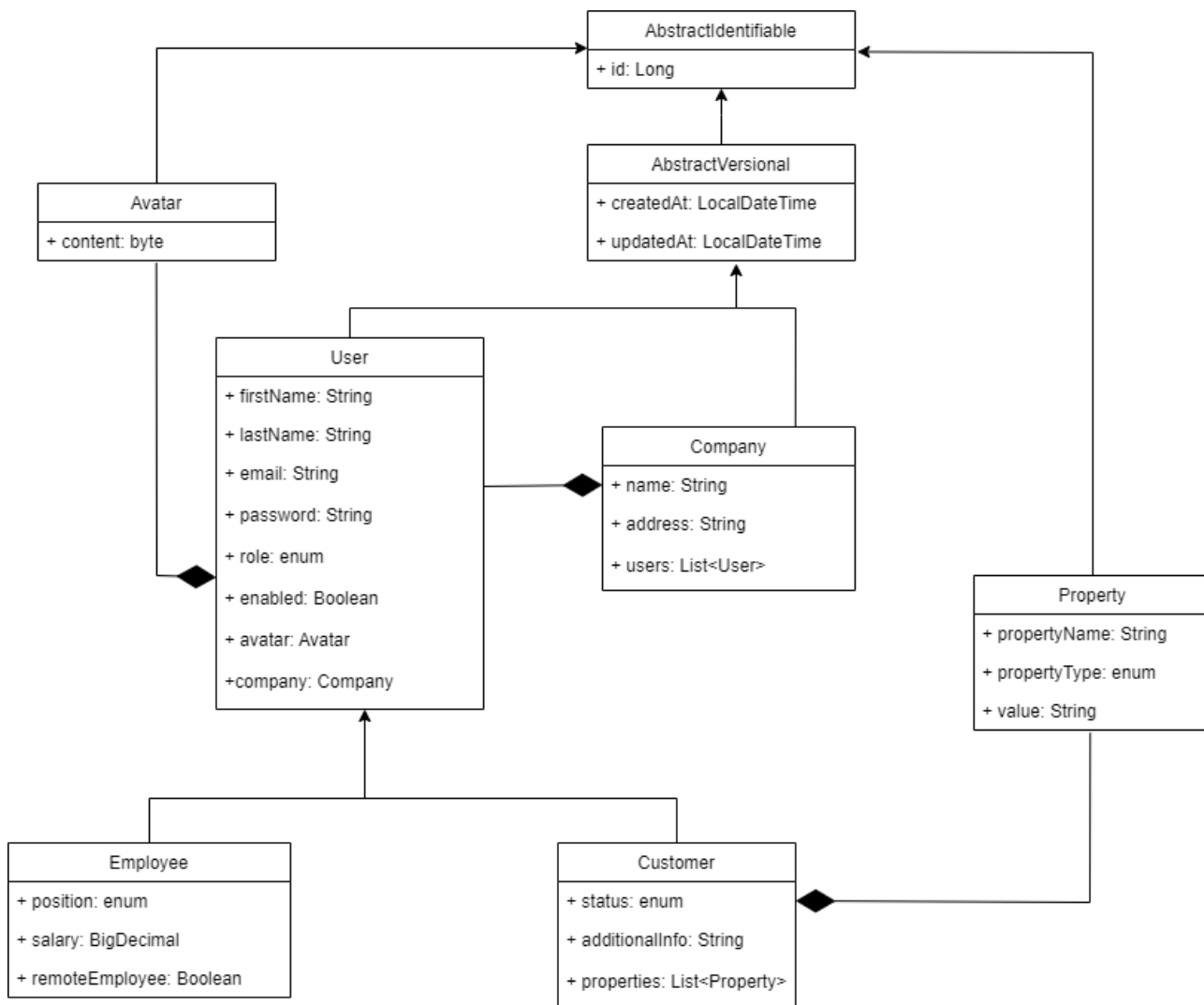


Рисунок 3.1.1 – Uml діаграма класів

Клас `AbstractIdentifiable`, який є базовим абстрактним класом, що містить поле `id` для можливості ідентифікації сутності.

Клас `AbstractVersional`, який є абстрактним класом, що містить поля `createdAt` і `updatedAt` для позначення часу, коли об'єкт було створено і коли було редаговано.

Клас `User`, який містить такі поля як:

- `firstName`, що призначене для збереження імені користувача;
- `lastName`, що призначене для збереження прізвища користувача;
- `email`, що призначене для збереження електронної адреси користувача;
- `password`, призначене для збереження паролю;
- `role`, призначене для збереження ролі користувача;
- `enabled`, призначене для позначення активності користувача;
- `avatar`, призначене для збереження фото профілю користувача;
- `company`, призначене для збереження інформації про компанію, до якої належить користувач.

Клас `Customer`, який містить поля:

- `status`, що зберігає інформацію про статус клієнта;
- `additionalInfo`, призначене для збереження додаткової інформації про клієнта;
- `properties`, що зберігає перелік полів, що доступні даному клієнту.

Клас `Employee`, що містить поля:

- `position`, що призначене для збереження інформації про посаду працівника;
- `salary` для збереження інформації про зарплату працівника;
- `remoteEmployee`, що зберігатиме інформацію про те, чи віддалено працює людина.

Клас `Property` відповідає за динамічні поля, а тому містить такі поля:

- `propertyName`, призначене для збереження назви динамічного поля;
- `propertyType`, що зберігатиме інформацію про тип динамічного поля;
- `value`, що зберігатиме саме значення динамічного поля.

Клас `Avatar`, який відповідатиме за фото профілю користувача і містить поле `id` для ідентифікації та поле `content` для зберігання фото.

Клас Company відповідатиме за збереження інформації про компанію, до якої належать користувачі програмної системи і містить поле name для збереження назви компанії та поле address для збереження адреси розташування компанії, а також перелік користувачів, що належать до певної компанії, які зберігатимуться у колекції.

3.2 Розробка GUI

Інтернет-бізнесу слід розуміти важливість дизайну інтерфейсу користувача (UI) і зручності використання веб-сайту. Чудово сконструйований інтерфейс користувача не рідко виконується сучасними досвідченими розробниками та дизайнерами, оскільки це величезне та творче поле. Було доведено, що дизайн інтерфейсу користувача є основним інструментом, який підвищує залучення користувачів. Залучення користувачів і досвід користувача (UX) є основними факторами продажу. Те, як цільова аудиторія відчувається під час навігації веб-сайтом, визначає успіх або падіння бізнесу.

Дизайн інтерфейсу користувача включає всі візуальні аспекти веб-сайтів, такі як кольорні схеми, шрифти, зображення, анімація тощо. Ці елементи відіграють важливу роль у тому, щоб користувачі почувалися комфортно. Вони також допомагають їм легко переходити з однієї сторінки на іншу. Крім того, вони дбають про те, щоб весь вміст був зрозумілим для відвідувачів. Це допомагає підвищити рівень конверсії, забезпечуючи відсутність плутанини серед клієнтів. Розробка інтерфейсу користувача спрямована на те, щоб передбачити шлях користувача на вашому веб-сайті та переконатися, що веб-сайт є легкодоступним і послідовним. Інтерфейс користувача поєднує три ключові концепції: дизайн взаємодії, візуальний дизайн і архітектура веб-сайту.

Люди хочуть мати можливість легко розуміти речі, тому більшість фізичних продуктів можна зрозуміти, не копаючись у великих посібниках. Зробити дизайн інтерфейсу користувача зрозумілим без посібника – це різниця між повним успіхом і потенційно невтішним розчаруванням. По суті, дизайн інтерфейсу користувача є обов'язковим, тому що він може означати різницю між

ситуацією «привабити» або «відштовхнути» цільову аудиторію. Сучасний та зрозумілий інтерфейс створює менше клопоту, збільшує залучення користувачів, покращує функціональність і створює міцний зв'язок між клієнтами та сайтом.

Мною було виділено кілька основних сторінок, які, на мою думку, повинні бути присутніми у CRM системі, а саме: сторінка для відображення переліку клієнтів та користувачів, сторінка для демонстрації графіку залежності, сторінка для виконання дії з динамічними полями, а також сторінка профілю користувача. Зважаючи на чітко окреслені функції системи у попередніх розділах, можна вважати, що даний функціонал буде у повній мірі відображено.

На рисунку 3.2.1 представлено сторінку, на якій відображено графік, що показує кількість клієнтів компанії за певну дату.

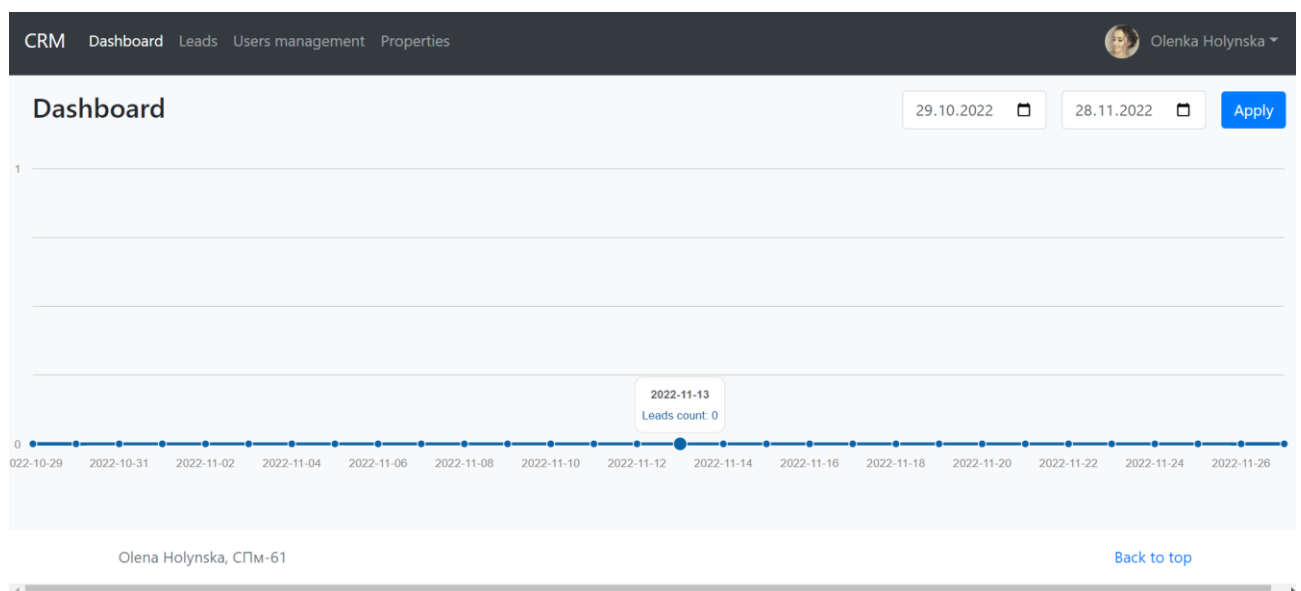


Рисунок 3.2.1 – Сторінка із графіком

На рисунку 3.2.2 можна побачити сторінку для відображення переліку усіх клієнтів компанії. Це достатньо зручно бачити перелік інформації, що по факту зберігається у базі даних, у формі такої таблиці, що дозволяє не лише переглядати інформацію про клієнтів, але виконувати ряд операції, пов'язаних із редагуванням, додаванням нових користувачів тощо.

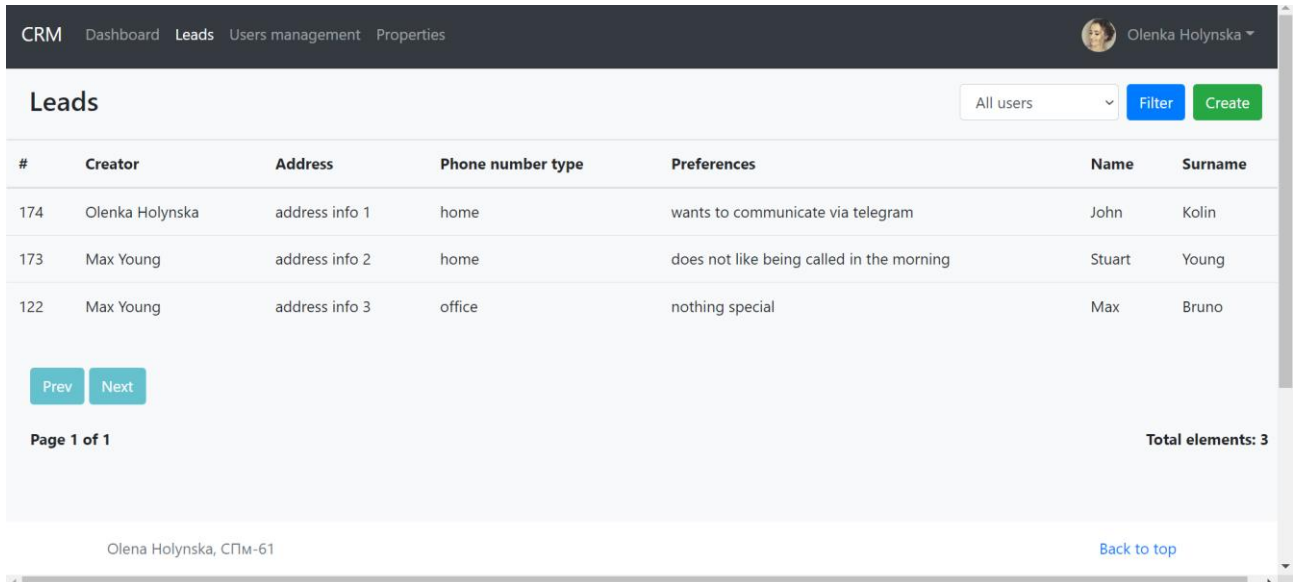


Рисунок 3.2.2 – Сторінка з інформацією про клієнтів

На рисунку 3.2.3 зображено сторінку для маніпуляції динамічними полями. Такі динамічні поля відобразатимуться для кожного клієнта аби додати якусь нестандартну потрібну інформацію. Це корисно у тих випадках, коли ми володіємо цінною інформацією про клієнта, його вподобання, але не знаходимо потрібного інтерфейсу щоб занотувати це та зберегти.

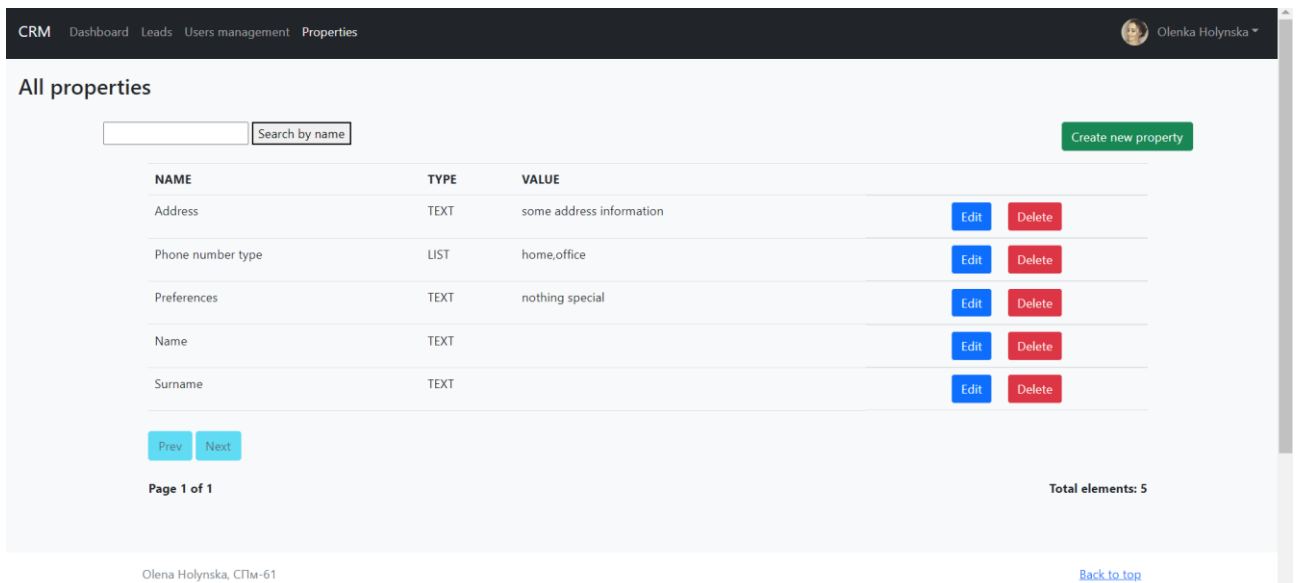


Рисунок 3.2.3 – Сторінка для динамічних полів

На рисунку 3.2.4 ми бачимо функціонал, що доступний лише адміністратору системи, стосовно менеджменту працівників компанії, тобто тих

користувачів, які зареєстровані у системі та належать до компанії і виконують роботу, що націлена на управління відносинами із потенційними та наявними клієнтами.

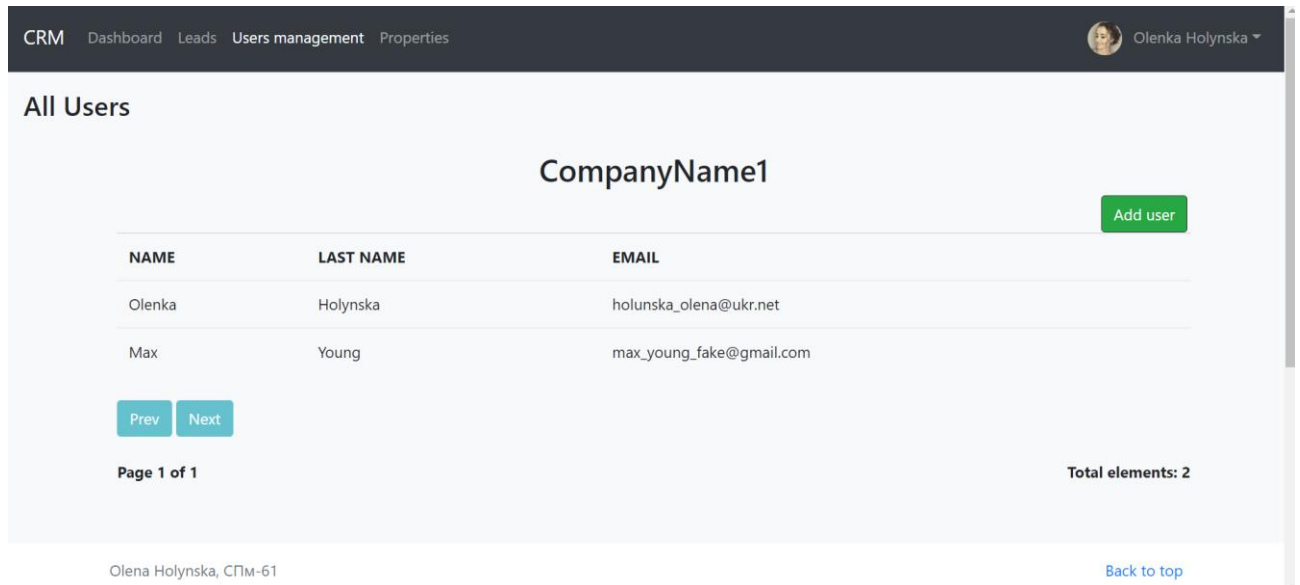


Рисунок 3.2.4 – Сторінка для менеджменту працівників компанії

На рисунку 3.2.5 представлено сторінку профілю користувача, де можна редагувати інформацію профілю, додавати та видаляти фото профілю, а також змінювати пароль до профілю. Тобто це сторінка, яка доступна кожному користувачеві, що зареєстрований у програмній системі.

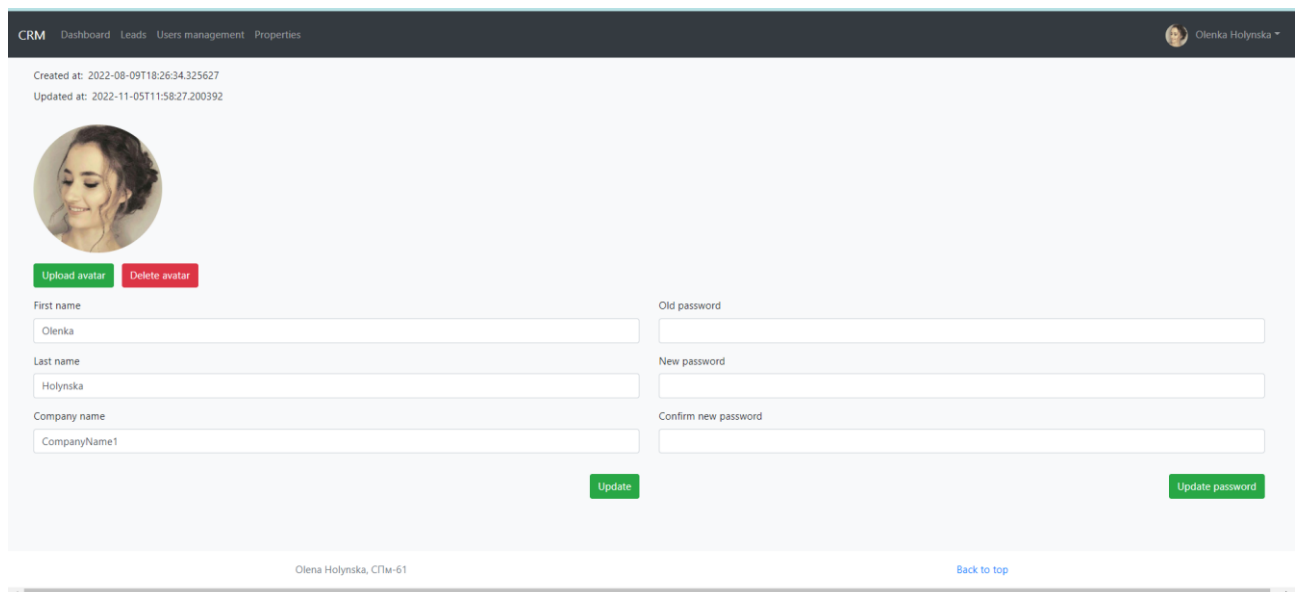


Рисунок 3.2.5 – Сторінка профілю користувача

Отже, графічний інтерфейс користувача реалізовано успішно, вдало розміщено дані у таблицях, додано можливості виконувати операції додавання, видалення, редагування інформації.

3.3 Тестування та оцінка якості ПЗ

У цьому сучасному світі контроль якості та тестування є одними з найбільших обмежень, коли справа доходить до своєчасної доставки таких продуктів, як програмне забезпечення. Хоча клієнти зазвичай розуміють, що жоден продукт ніколи не є повністю вільним від помилок, коли він запускається вперше, помилка може мати руйнівний вплив на репутацію та продажі програмного забезпечення. Таким чином, на сьогоднішній день спостерігається великий потяг до автоматизованих тестів. А в нову епоху автоматизованого тестування ручне тестування вважають безглуздом і марною тратою робочої сили. Проте до цих пір у більшості компаній ручне тестування вважається однією з важливих частин процедури забезпечення якості. Різні підприємства та компанії не досягають бажаних результатів лише за допомогою автоматизації тестування. Ручне тестування є ключовим методом для запуску тестування програмного забезпечення вручну, порівнюючи очікування кількох програм із певними результатами для аналізу програмних помилок.

Ручне тестування — це процес тестування програмного забезпечення, у якому тестові випадки виконуються вручну без використання будь-яких автоматизованих інструментів. Усі тестові випадки виконуються тестером вручну відповідно до точки зору кінцевого користувача. Це гарантує, чи працює програма, як зазначено в документі вимоги, чи ні. Тестові випадки плануються та впроваджуються для завершення майже 100 відсотків програми. Звіти про тестові випадки також створюються вручну. Ручне тестування є одним із найважливіших процесів тестування, оскільки воно може виявити як видимі, так і приховані дефекти програмного забезпечення. Різниця між очікуваним результатом і результатом, заданим програмним забезпеченням, визначається як

дефект. Для ручного тестування використовуються різні методи. Кожна методика використовується відповідно до своїх критеріїв тестування.

Тестування розробленої програмної системи можна розпочати із головної сторінки. Додамо у базу даних кілька клієнтів із різною датою, аби перевірити чи працює функція для відображення залежності кількості клієнтів від часу, а також чи коректно це відображатиметься на UI частині. На рисунку 3.3.1 представлено результат.

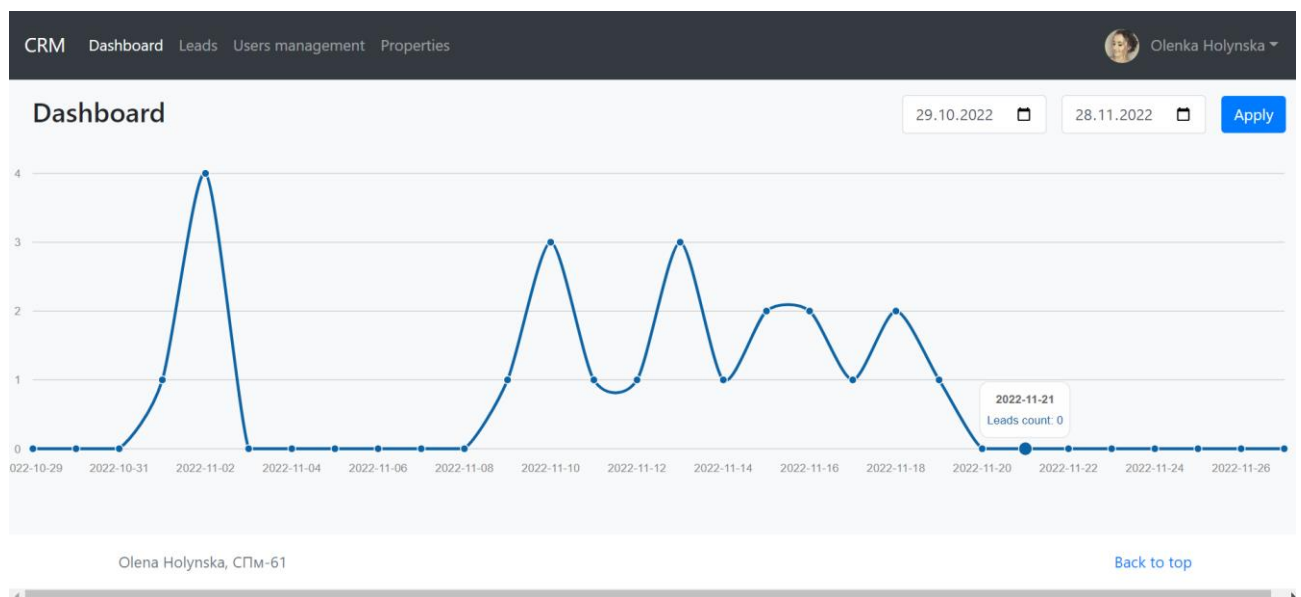


Рисунок 3.3.1 – Результат тестування сторінки для відображення графіку

Сторінка для відображення користувачів системи повинна бути протестована на можливість додавання нового користувача. Якщо натиснути на кнопку «Add user», то можна побачити сторінку для внесення інформації, що представлено на рисунку 3.3.2.

CRM Dashboard Leads Users management Properties Olenka Holynska

Add new user

Enter email:

Enter password:

Enter name:

Enter last name:

[Cancel](#) [Add](#)

Olenka Holynska, CRM-61 [Back to top](#)

Рисунок 3.3.2 – Сторінка для додавання нового користувача

На сторінці, що відповідає за відображення динамічних полів, розміщені також кнопки для здійснення операцій додавання, видалення та редагування динамічних полів. Якщо натиснути кнопку «Create new property» то ми побачимо завантажену сторінку для внесення інформації.

CRM Dashboard Leads Users management Properties Olenka Holynska

Create a new property

Write property name:

Choose property type:

[Cancel](#) [Submit](#)

Olenka Holynska, CRM-61 [Back to top](#)

Рисунок 3.3.3 – Сторінка для створення нового динамічного поля

На сторінці для створення нового динамічного поля є такі поля для вводу як назва поля та тип поля. Тип поля може бути звичайною стрічкою на кілька

символів, текстом, листом, який включає у собі значення і мітку та датою як показано на рисунку 3.3.4.

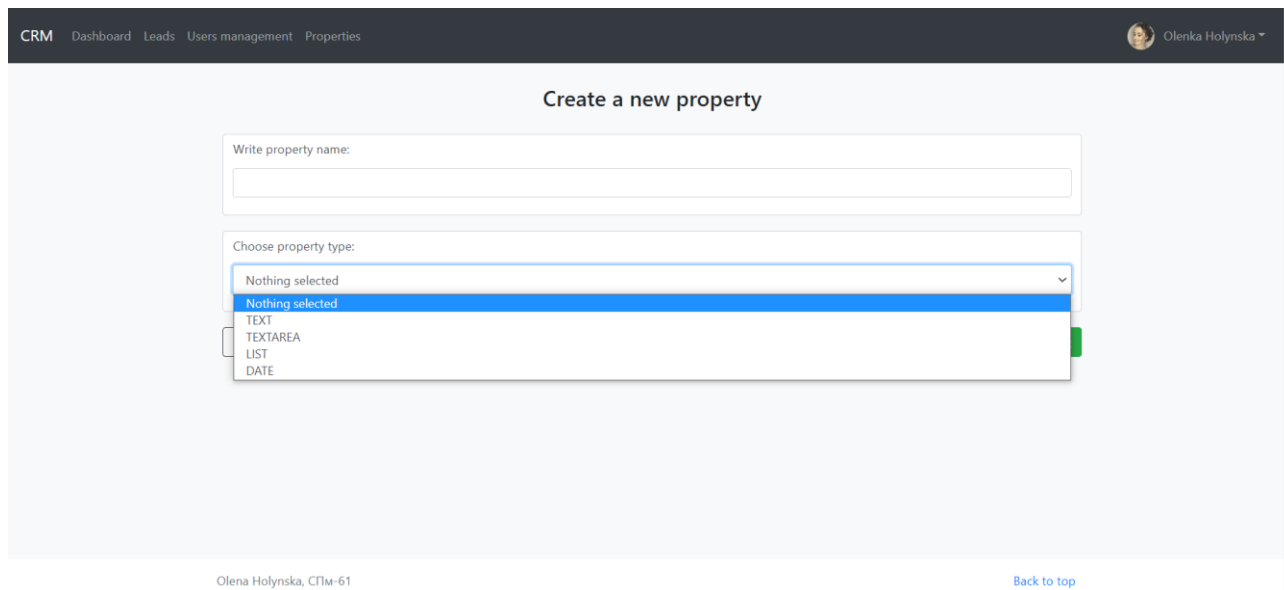


Рисунок 3.3.4 – Типи динамічного поля

Хотілося б звернути увагу на тип динамічного поля лист. Даний тип є масивом, що зберігає пару значення і мітки, таких значень може бути кілька. За допомогою добре продуманого інтерфейсу для типу листа у нас є можливість додавати пару та видаляти залежно від потреб, що показано на рисунку 3.3.5.

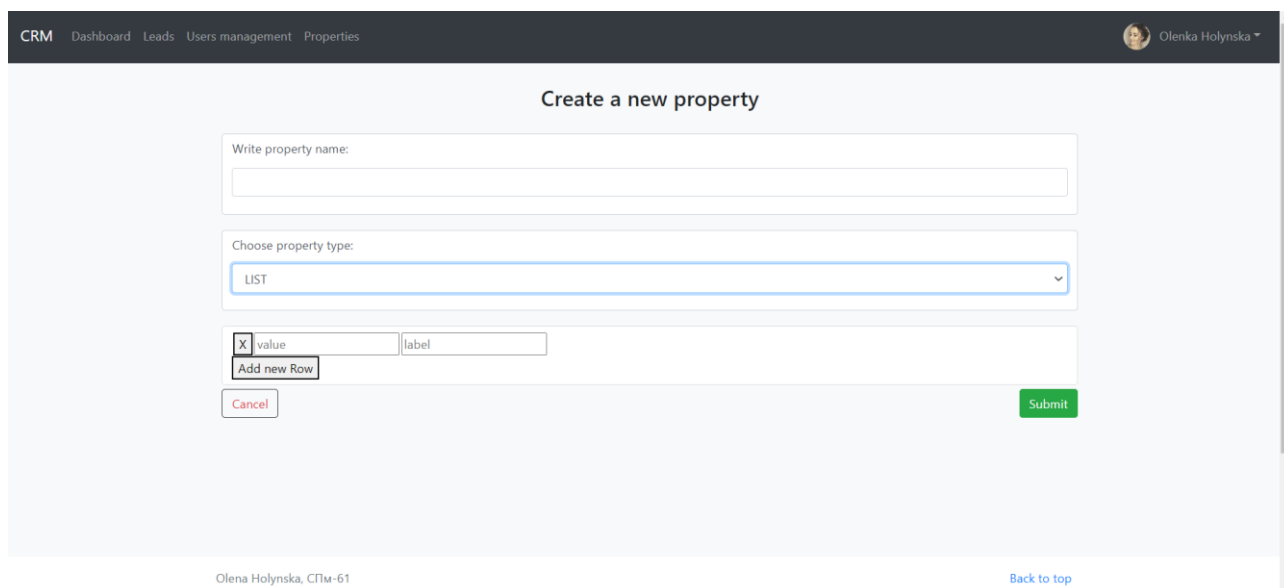


Рисунок 3.3.5 – Тип лист для динамічного поля

Якщо натиснути кнопку «Edit», то нам завантажиться сторінка для редагування динамічного поля, де можна вказувати актуальну інформацію, як показано на рисунку 3.3.6.

The screenshot shows a web interface for editing a property. At the top, there is a navigation bar with 'CRM Dashboard Leads Users management Properties' and a user profile 'Olenka Holynska'. The main heading is 'Edit property'. Below it, there are three input fields: 'Write property name:' containing 'Address', 'Choose property type:' containing 'TEXT', and 'address information'. At the bottom of the form, there are 'Cancel' and 'Save' buttons. The footer contains 'Olena Holynska, СПМ-61' and a 'Back to top' link.

Рисунок 3.3.6 – Сторінка для редагування динамічного поля

Якщо усі поля валідні, тобто інформація, що вказана у формі, відповідає усім вимогам, то операція додавання, редагування чи видалення проходить успішно і ми бачимо повідомлення протягом кількох секунд про те, що все пройшло добре, яка показано на рисунку. Але якщо вказати неправильний тип даних або вийти за ліміт допустимого значення, то ми побачимо повідомлення про помилку, а сама операція не виконається і ми повернемося до попередньої сторінки.

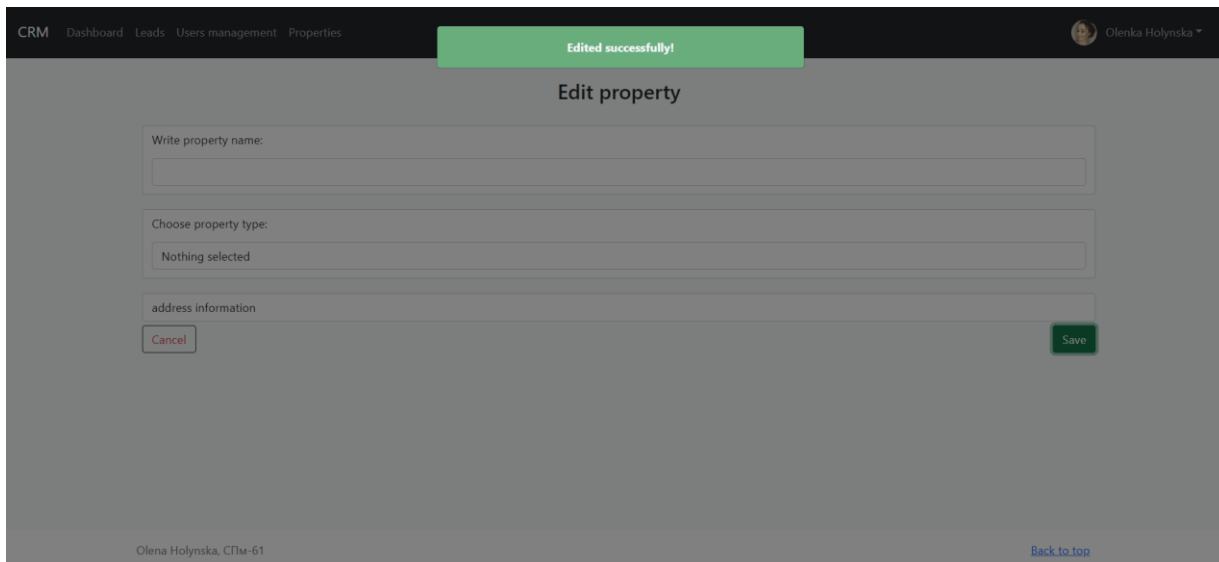


Рисунок 3.3.7 – Повідомлення про успішність виконання операції редагування

Якщо потрібно знайти динамічне поле за назвою, то можна скористатись пошуком за назвою. Результат пошуку динамічного поля за назвою представлено на рисунку 3.3.8.

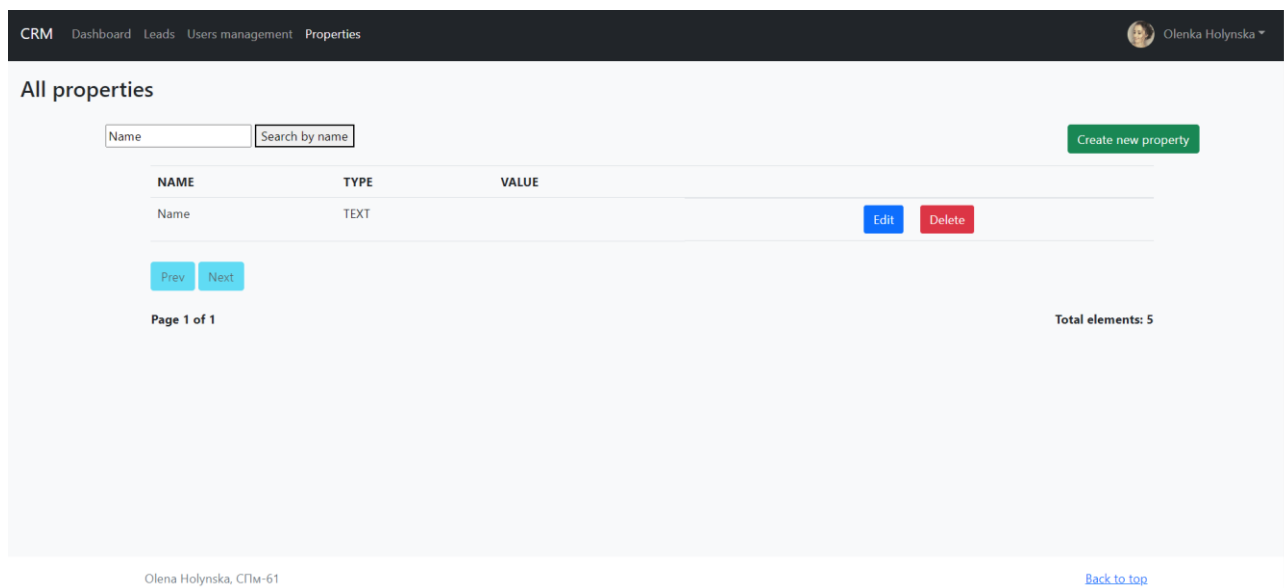


Рисунок 3.3.8 – Результат пошуку динамічного поля за назвою

Для того, щоб додати нового клієнта необхідно натиснути кнопку «Create» і заповнити необхідні поля у формі, що представлена на рисунку 3.3.9.

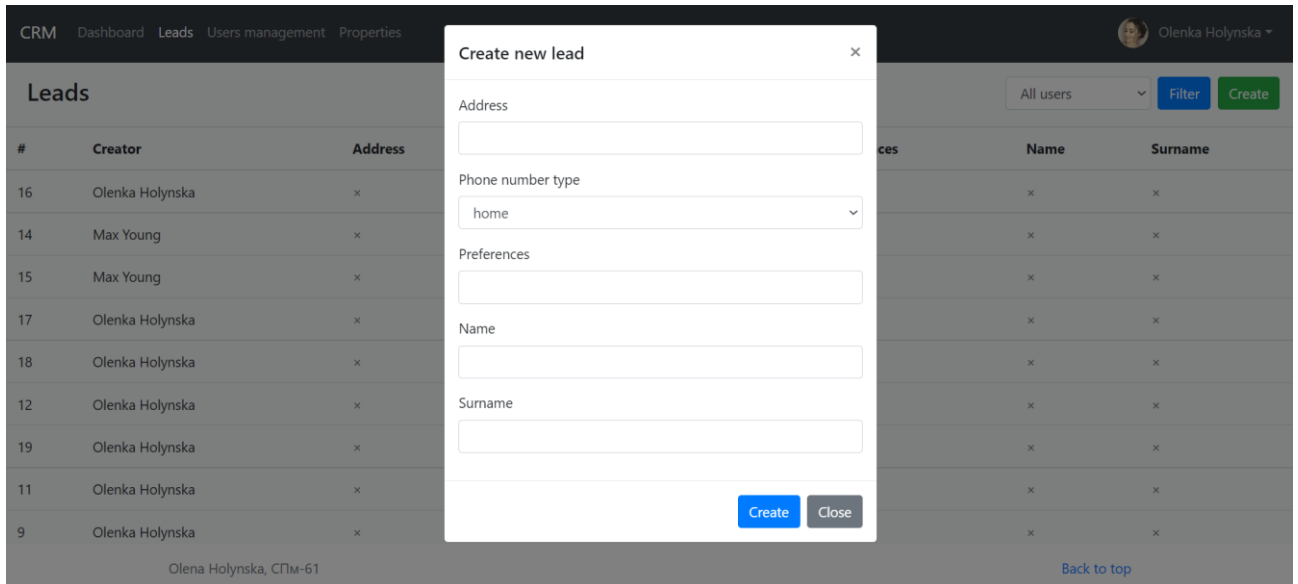


Рисунок 3.3.9 – Форма для створення нового клієнта

У програмній системі є також можливість редагувати інформацію про клієнта. Для цього необхідно натиснути на записі у таблиці, після чого з'явиться форма для редагування інформації про клієнта, як показано на рисунку 3.3.10.

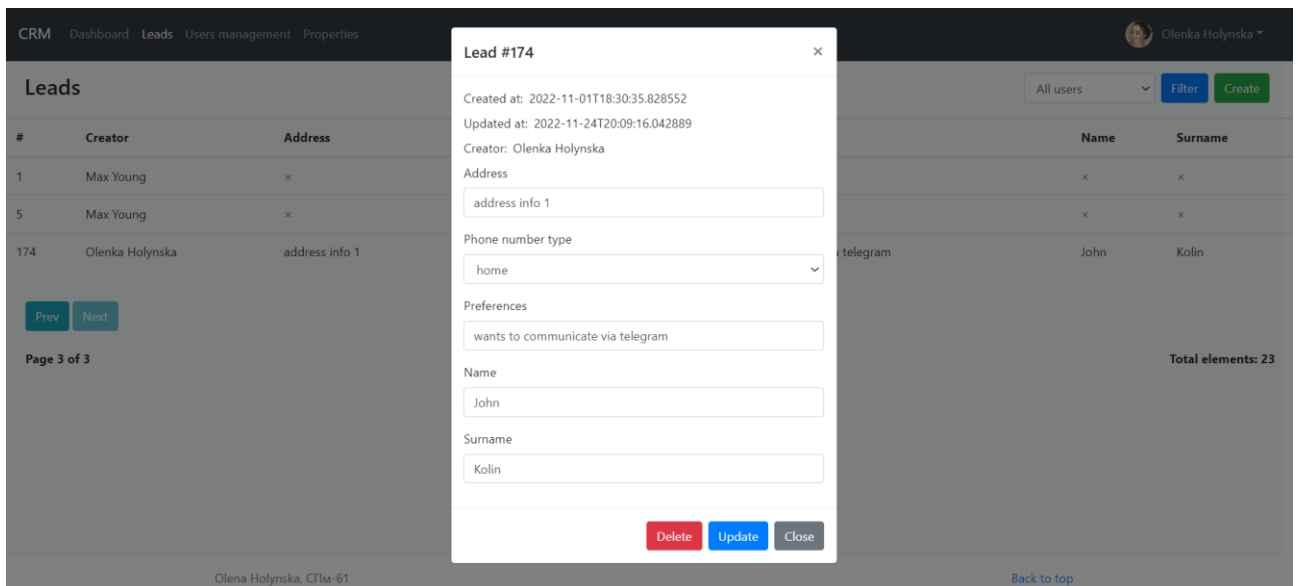


Рисунок 3.3.10 – Редагування інформації про клієнта

Для того, щоб відобразити інформацію про клієнтів за особою, яка створила ці записи, можна скористатись фільтром, що розміщений у правому верхньому куті.

The screenshot shows a CRM interface with a 'Leads' table. The table has columns for '#', 'Creator', 'Address', 'Phone number type', 'Preferences', 'Name', and 'Surname'. All data rows show 'Olenka Holynska' as the creator. Above the table, there is a search bar with 'Olenka Holynska' entered, and 'Filter' and 'Create' buttons. Below the table, there are 'Prev' and 'Next' buttons, 'Page 1 of 2', and 'Total elements: 13'. The footer contains 'Olena Holynska, CRM-61' and a 'Back to top' link.

#	Creator	Address	Phone number type	Preferences	Name	Surname
16	Olenka Holynska	x	x	x	x	x
17	Olenka Holynska	x	x	x	x	x
12	Olenka Holynska	x	x	x	x	x
18	Olenka Holynska	x	x	x	x	x
19	Olenka Holynska	x	x	x	x	x
11	Olenka Holynska	x	x	x	x	x
9	Olenka Holynska	x	x	x	x	x
8	Olenka Holynska	x	x	x	x	x
10	Olenka Holynska	x	x	x	x	x
7	Olenka Holynska	x	x	x	x	x

Рисунок 3.3.11 – Фільтрація клієнтів у таблиці за особою, що додала їх

Якщо користувач забув пароль, то можна скористатись логін сторінкою, щоб його відновити, натиснувши на «forgot password». Нижче на рисунку 3.3.12 показано яким чином виглядає сторінка для відновлення паролю.

Forgot Password

We will send a reset password link to you.

[Login](#)

Рисунок 3.3.12 – Сторінка для відновлення паролю

Після того, як користувач введе свій логін, тобто адресу електронної пошти, за якою реєструвався у системі до цього, можна побачити повідомлення про успішне відновлення акаунту.

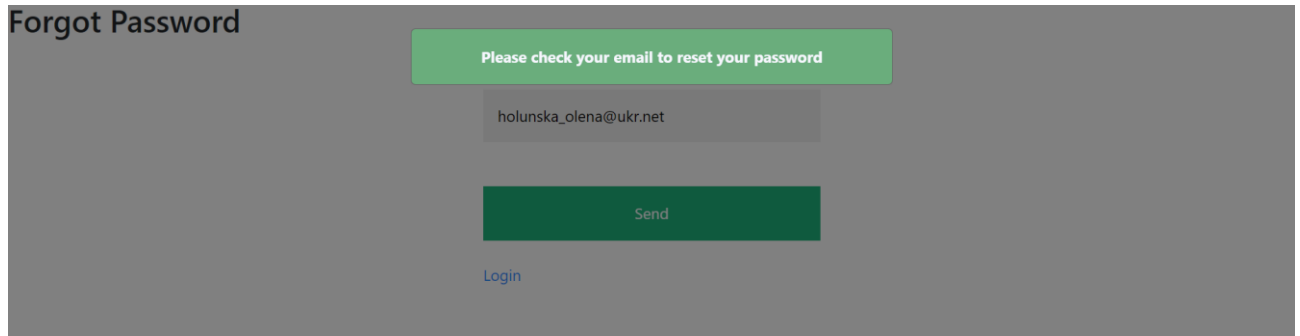


Рисунок 3.3.13 – Повідомлення про успішну зміну паролю

Для того, аби зареєструватись у програмній системі необхідно на сторінці для логіну натиснути «create account», після чого ми побачимо завантажену сторінку для реєстрації як показано на рисунку 3.3.14.

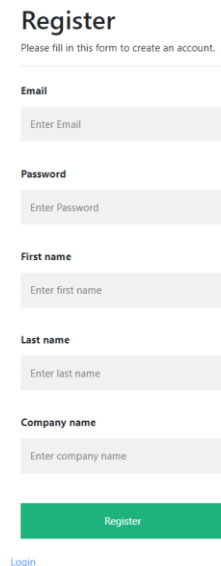


Рисунок 3.3.14 – Сторінка для реєстрації нових користувачів

Змінити пароль можна і у профілі користувача, але для цього необхідно спершу успішно увійти у профіль і вже тоді користувач матиме можливість змінити пароль, скориставшись формою. Після реєстрації можна побачити повідомлення про успішність реєстрації.

You have signed up successfully!

Please check your email to verify your account.

[Click here to Login](#)

Рисунок 3.3.15 – Повідомлення про успішну реєстрацію користувача у системі

Отже, програму успішно протестовано, результати представлено у вигляді рисунків. Згідно представлених рисунків можна зробити висновок про досить успішну реалізацію користувацького інтерфейсу, а також про різноманітність передбачених випадків, що стосуються здійснення операцій, повідомлень для користувачів про статус операцій та зручність використання функцій системи.

4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці

При розробці CRM системи для контролю та управління взаємовідносинами із потенційними клієнтами інженер-програміст використовує персональні електронно-обчислювальні машини. Тому виконання правил з охорони праці є важливим фактором для запоруки безпеки та здоров'я.

На перший погляд, робота за комп'ютером здається безпечною, але саме легковажність до неї може призвести до певних проблем у здоров'ї людини. Професія програміста та інших фахівців ІТ-технологій пов'язана з колосальним розумовим напруженням. Розробники – настільки захоплені люди, що навіть відволікаючись від роботи над проектом, продовжують думати про роботу. Нерідко відпочинком вони вважають паралельну заміну основної діяльності, наприклад, читання профільної літератури, верстку сайтів, вивчення нових мов програмування.

Працюючи за персональний комп'ютером ступінь розумової напруги значно зростає. Навантаження також спричиняє емоційне вигорання, є високим показником напруженості зору і навантаженням на м'язи рук при використанні клавіатури. У певній мірі, підвищення ефективності і результативності праці робітника залежить від факторів втомлюваності чи бадьорості. Згідно з Державними санітарними правилами і нормами роботи з візуальними дисплейними терміналами електронно обчислювальних машин ДСанПН 3.3.2.007-98 відповідність розташування та організація робочого місця з ВДТ повинна відповідати ергономічними вимогами, враховуючи особливість трудової діяльності.

При CRM системи для контролю та управління взаємовідносинами із потенційними клієнтами повинні враховуватись існуючі санітарні нормативи освітлення, вимоги до параметрів мікроклімату (температура, відносна вологість), ступеня і сили вібрації, звукового шуму і вогнестійкості приміщення. Конкретні показники зазначених санітарних норм є у інструкції “Державні

санітарні правила і норми роботи з візуальними дисплейними терміналами (ВДТ) електронно-обчислювальних машин” ДСанПіН 3.3.2.007-98.

Для об’єктивного оцінювання умов праці робочого місця здійснюється його атестація, а також «Гігієнічна класифікація праці». Дані показників загрозливих, шкідливих чинників залежать від самого виробничого середовища та гостроти напруження трудового процесу [13].

Розробка комерційного програмного продукту відбувається в офісному приміщенні, в якому знаходяться комп’ютеризовані робочі місця та експлуатується комп’ютерна техніка інженерами-програмістами. Під час планування приміщення слід дотримуватись державних нормативно-правових актів та законів. Згідно ДБН В.2.5-28:2018 приміщення, в якому розробляється система оцінки вартості житлового будівництва, повинно мати природне і штучне освітлення. Природне освітлення має здійснюватись через світлові прорізи, орієнтовані переважно на північ чи північний схід, і забезпечувати коефіцієнт природної освітленості (КПО) не нижче, ніж 1,5%. Також у приміщеннях глибиною 6 м та більше доцільно застосовувати на вікнах спеціальні світловідбивні екрани та жалюзі, що перерозподіляють світловий потік в глибину приміщення.

Під час роботи за комп’ютером інженер-програміст піддається впливу небезпечних та шкідливих здоров’ю чинників. Психофізіологічні чинники: напруження уваги, пам’яті та зору; значна кількість інформації, яку необхідно обробити за одиницю часу; в деяких випадках монотонність завдань. Фізичні чинники: збільшений рівень інфрачервоного, ультрафіолетового та рентгенівського випромінювання; відсутність рівномірного розподілу яскравості на робочому місці. Для зменшення впливу негативних чинників, під час розробки програмної системи, було правильно розміщено робочі місця та комп’ютерне обладнання. Це питання регулюється широким спектром нормативно-правових актів. Законом України “Про охорону праці” передбачено обов’язки роботодавця, які описують необхідність забезпечення безпечних та комфортних умов праці.

За таких умов зростає роль та значення охорони праці як системи правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, що спрямовані на збереження здоров'я і працездатності людини в процесі праці [14].

Отже, правильна організація робочого простору є важливим та доцільним елементом, для забезпечення якого необхідно уникати незручності в розташуванні засобів праці.

4.2 Планування заходів цивільного захисту на об'єкті у випадку надзвичайних ситуацій

Цивільний захист України - є державною системою органів управління, сил і засобів, що створюється для організації і забезпечення захисту населення від наслідків надзвичайних ситуацій техногенного, екологічного, природного та воєнного характеру.

Об'єкт господарської діяльності - це підприємства (державні і приватні), установи і організації, навчальні заклади та інші. На всіх об'єктах Цивільний захист організовується з метою завчасної підготовки їх до захисту від наслідків надзвичайних ситуацій, зниження втрат, створення умов для підвищення стійкості роботи об'єктів та своєчасного проведення рятувальних та інших невідкладних робіт (РІНР). Відповідальність за організацію та стан Цивільного захисту, за постійну готовність сил і засобів до проведення РІНР несе начальник цивільного захисту (НЦЗ) об'єкта - керівник підприємства, установи та організації.

Цивільний захист об'єкта - система організаційних, інженерно-технічних, санітарно-гігієнічних, протиепідеміологічних та інших заходів, що здійснюються керівництвом об'єкта господарської діяльності з метою запобігання та ліквідації надзвичайних ситуацій, які загрожують життю та здоров'ю людей у мирний та воєнний час.

Основні завдання цивільного захисту

- запобігання виникненню надзвичайних ситуацій техногенного та природного походження, запровадження заходів щодо зменшення збитків та втрат у разі аварій, катастроф, вибухів, пожеж та стихійного лиха;
- оперативне оповіщення працівників про виникнення або загрозу виникнення надзвичайної ситуації, своєчасне достовірне інформування про обстановку, яка складається, та заходи, що вживаються для запобігання надзвичайним ситуаціям та подолання їх наслідків;
- організація захисту населення і територій від надзвичайних ситуацій, надання невідкладної психологічної, медичної та іншої допомоги потерпілим;
- проведення невідкладних робіт із ліквідації наслідків надзвичайних ситуацій та організація життєзабезпечення постраждалого населення;
- забезпечення постійної готовності сил і засобів цивільного захисту до запобігання надзвичайним ситуаціям та ліквідації їх наслідків;
- навчання населення способам захисту в разі виникнення надзвичайних, несприятливих побутових або нестандартних ситуацій та організація тренувань;
- створення, збереження і раціональне використання матеріальних ресурсів, необхідних для запобігання надзвичайним ситуаціям;
- забезпечення відповідно до законодавства своїх працівників засобами колективного та індивідуального захисту;
- організація та здійснення під час виникнення надзвичайних ситуацій евакуаційних заходів щодо працівників та майна суб'єкта господарювання;
- створення об'єктових формувань цивільного захисту відповідно до Кодексу Цивільного Захисту, інших законодавчих актів, необхідної для їх функціонування матеріально-технічної бази і забезпечення готовності таких формувань до дій за призначенням;
- проведення оцінки ризиків виникнення надзвичайних ситуацій на об'єктах суб'єкта господарювання, здійснення заходів щодо неперевищення прийнятних рівнів таких ризиків;
- проведення об'єктових тренувань і навчань з питань цивільного захисту.

Як свідчить аналіз надзвичайних ситуацій за останні 5–8 років, значна кількість різноманітних надзвичайних ситуацій виникає на об'єктовому рівні. До нього належать і невеликі (малі) підприємства, установи, організації, заклади з чисельністю працівників 50 осіб і менше у сфері виробництва, логістики, торгівлі, освіти та науки, медицини, розважальної індустрії тощо [15].

Від ефективності розроблення та впровадження в життя заходів із запобігання та ліквідації надзвичайної ситуації в разі її виникнення залежатиме життя та здоров'я персоналу та відвідувачів цих підприємств і розміри заподіяної шкоди. Відповідно до Кодексу цивільного захисту України, підготовка персоналу на підприємствах незалежно від форм власності до дій у надзвичайних ситуаціях здійснюється за спеціально розробленою схемою заходів захисту населення та територій.

Для великих і малих підприємств система заходів захисту від надзвичайних ситуацій включає:

- планування та здійснення необхідних заходів для захисту своїх працівників, об'єктів господарювання;
- розроблення планів локалізації та ліквідації аварій з подальшим погодженням з Державною службою України з надзвичайних ситуацій;
- підтримання у готовності до застосування сил і засобів із запобігання виникненню та ліквідації наслідків надзвичайних ситуацій;
- створення та підтримання матеріальних резервів для попередження та ліквідації надзвичайних ситуацій;
- забезпечення своєчасного оповіщення своїх працівників про загрозу виникнення або при виникненні надзвичайної ситуації.

Наведені вище заходи мають загальний характер, вони не повністю враховують специфіку діяльності конкретного підприємства, чисельність працівників тощо.

Основною особливістю дій малих підприємств при загрозі або виникненні надзвичайних ситуацій є в першу чергу захист персоналу та відвідувачів. Виходячи з цього, ст. 130 Кодексу цивільного захисту України передбачає, що на підприємствах з чисельністю персоналу 50 осіб і менше розробляються та

затверджуються інструкції щодо дій при загрозі або виникненні надзвичайних ситуацій.

Крім Інструкції, на малому підприємстві розробляється План евакуації при пожежі або загрозі вибуху. Особливо це важливо для тих об'єктів, на території яких може знаходитись значна кількість відвідувачів.

Деякі конкретні заходи, не відображені в нормативних документах підприємства, потребують внесення до посадових інструкцій працівників. Крім того, на малому підприємстві необхідно розробляти й доводити до всіх працівників Порядок цілодобового оповіщення керівництва та працівників у випадку загрози або виникнення надзвичайної ситуації.

Всі працівники підприємства повинні бути навчені діям, чітко знати свої обов'язки та неухильно їх виконувати. Це також стосується адміністрації малого підприємства, яка в екстремальній обстановці не може приймати помилкові рішення або віддавати необґрунтовані розпорядження.

Уникнути цього дозволить якісно розроблена Інструкція щодо дій персоналу малого підприємства при загрозі або виникненні надзвичайних ситуацій.

Цивільний захист здійснюється за такими основними принципами:

1. гарантування та забезпечення державою конституційних прав громадян на захист життя, здоров'я та власності;
2. комплексного підходу до вирішення завдань цивільного захисту;
3. пріоритетності завдань, спрямованих на рятування життя та збереження здоров'я громадян;
4. максимально можливого, економічно обґрунтованого зменшення ризику виникнення надзвичайних ситуацій;
5. гласності, прозорості, вільного отримання та поширення публічної інформації про стан цивільного захисту, крім обмежень, встановлених законом;
6. добровільності - у разі залучення громадян до здійснення заходів цивільного захисту, пов'язаних з ризиком для їхнього життя і здоров'я;

7. відповідальності посадових осіб органів державної влади та органів місцевого самоврядування за дотримання вимог законодавства з питань цивільного захисту;

Планування заходів цивільного захисту передбачає управління надзвичайними ситуаціями. Для забезпечення безпеки людини в НС управління об'єктами повинно включати здійснення 3-х стратегій: запобігання причин виникнення; запобігання самих НС; пом'якшення, максимальне ослаблення наслідків НС. Стратегія запобігання причин виникнення НС передбачає недопущення таких дій чи процесів, які несуть загрозу населенню. Дана стратегія здійснюється або відмовою від будівництва небезпечних об'єктів, або знищенням чи перепрофілюванням виробництв – джерел підвищеної небезпеки. Друга стратегія — запобігання самих НС – передбачає недопущення виходу небезпечного процесу з-під контролю шляхом використання надійних аварійних систем, сигналізації, автоматики й інших заходів з підвищення надійності і стійкості роботи підприємств, а також шляхом заходів превентивної евакуації тощо. Третя стратегія — пом'якшення, максимальне ослаблення наслідків НС – передбачає орієнтацію на ослаблення, локалізацію наслідків НС. Ця стратегія має пріоритет у керуванні стихійними лихами і ситуаціями «комбінованого» типу. У практиці управління найбільший ефект дає спільне використання всіх трьох стратегій, особливо при промислових аваріях. У НС, викликаних стихійними лихами, пріоритет надається другій і третій стратегіям. Для реалізації кожної зі стратегій управління необхідно розробляти і приймати комплекс превентивних та оперативних заходів.

Отже, планування заходів цивільного захисту на об'єкті у випадку надзвичайних ситуацій безпосередньо є одним з найважливіших процесів для коректної роботи об'єкта. Дотримання визначених правил дозволить здійснити захист працівників об'єкту та майна від надзвичайних ситуацій шляхом запобігання таким ситуаціям, ліквідації їх наслідків і надання допомоги постраждалим.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи магістра було здійснено аналіз предметної області, визначено основну мету та завдання, а також вимоги до розробки CRM системи для контролю та управління взаємовідносинами із потенційними клієнтами.

Немає сумніву, що клієнти є джерелом корпоративних прибутків, і те, як ефективно керувати клієнтами, є ключем до успіху компанії. У сучасному середовищі жорсткої конкуренції пристосування до потреб клієнтів, надання клієнтам права вибору продуктів і надання їм можливості отримати те, що вони дійсно хочуть, є ключем до конкурентного успіху будь-якої компанії.

Система управління взаємовідносинами з клієнтами є важливим елементом основної програми підприємства. У простому розумінні CRM – це центр обробки даних клієнтів, який може обробляти різні деталі всіх клієнтів компанії в CRM, включаючи відстеження та керування кожним потенційним клієнтом, який взаємодіяв із корпоративним брендом, веб-сайтом або продуктом, а також може аналізувати його сліди та переваги. Зворотній зв'язок ринку протягом багатьох років довів, що це дуже допомагає розширювати бізнес підприємств, тому сучасні підприємства інвестували багато ресурсів у розвиток клієнтів, щоб швидше та ефективніше розширювати свою територію.

Виконуючи список поставлених задач, в результаті даної роботи розроблено CRM систему для зберігання інформації про клієнтів компанії, обробки та аналізу даних задля покращення процесу комунікації та налагодження співпраці, а також для збільшення прибутку підприємства.

Під час виконання були проаналізовані різні варіанти реляційних баз даних і обрано PostgreSQL як основну, оскільки вона задовільняє усі програмні вимоги. Для реалізації програмної системи обрано мову програмування Java та Spring Framework для побудови веб-застосунку. Було також використано додаткові інструменти та бібліотеки для побудови злагодженої програмної структури, що готова до змін і зручного користувацького інтерфейсу, без якого було б важко виконувати ряд задач, що закріплені за працівниками компанії. Спроектвана

діаграма варіантів використання дала змогу сфокусуватись на найголовніших функціях програмної системи, оскільки це графічне представлення, що ми отримали в результаті проектування діаграми показало чітку картину того, що мало б бути у реалізації, а також дозволило спрогнозувати, які ж ролі та права доступу отримають користувачі системи. На діаграмі класів було зображено ієрархію класів, на основі якої було реалізовано CRM систему з усіма її атрибутами та цілями.

Одним із мінімальних очікувань від системи управління взаємовідносинами з клієнтами є можливість отримати окрему інформацію, що пов'язана з клієнтами, функціонувати як єдине ціле, а не бути ізольованою. Мною було виділено кілька основних сторінок, що складають основу CRM системи, а саме: сторінка для відображення переліку клієнтів та користувачів, сторінка для демонстрації графіку залежності, сторінка для виконання дії з динамічними полями, а також сторінка профілю користувача.

Результатом кваліфікаційної роботи магістра є готова веб-орієнтована CRM система для контролю та управління взаємовідносинами з потенційними клієнтами. Таку систему можна вдосконалювати у майбутньому, додаючи можливість виконання операцій щодо продажу товарів та послуг компанії, а також функціонал, пов'язаний із аналізом та обробкою статистичних даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. CRM (customer relationship management) – [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.techtarget.com/searchcustomerexperience/definition/CRM-customer-relationship-management>
2. What Is The Role Of A CRM? – [Електронний ресурс] – Режим доступу до ресурсу: <https://pipelinecrm.com/blog/what-is-the-role-of-a-crm>
3. Importance of Customer Relationship Management (CRM) – [Електронний ресурс] – Режим доступу до ресурсу:
<https://managementstudyguide.com/importance-of-crm.htm>
4. Customer Relationship Management – [Електронний ресурс] – Режим доступу до ресурсу: <https://sci-hub.se/https://link.springer.com/book/10.1007/978-3-662-55381-7>
5. Benefits of CRM Software for Different Industries ? – [Електронний ресурс] – Режим доступу до ресурсу: <https://yourstory.com/mystory/benefits-of-crm-software-for-different-industries>
6. How 4 different industries use CRM and automation technology – [Електронний ресурс] – Режим доступу до ресурсу: <https://keap.com/business-success-blog/sales/crm/how-4-different-industries-use-crm-and-automation-technology>
7. The Customer is The King. How CRM Systems Help Companies From Different Industries to Reach Their Potential – [Електронний ресурс] – Режим доступу до ресурсу: <https://xbsoftware.com/blog/use-of-crm-in-different-spheres/>
8. Spring Framework – [Електронний ресурс] – Режим доступу до ресурсу:
<http://spring-projects.ru/projects/spring-framework/.0>
9. MSDN – Основи HTML – [Електронний ресурс] – Режим доступу до ресурсу:
<https://msdn.microsoft.com/ru-ru/library/wx4hctt4.aspx>.
10. IntelliJ IDEA [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.jetbrains.com/idea/?host=intellijidea.net>.
11. Брауде Е. Технология разработки программного обеспечения. СПб. : Изд-во "Питер" .2004. 655 с.

12. Ларман К. Применение UML и шаблонов проектирования М.: Вильямс, 2012. 496 с.
13. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями [Електронний ресурс] – Режим доступу до ресурсу: <https://zakon.rada.gov.ua/laws/show/z0508-18#n14>.
14. Русаловський А. В. Правові та організаційні питання охорони праці: Навч. посіб. – 4-те вид. – К.: Університет «Україна», 2009. – 295 с.
15. Желібо Є., Заверуха Н., Зацарний В. Безпека життєдіяльності. К.: 2001. 483 с
16. Методичні вказівки до виконання атестаційної роботи магістра за спеціальністю 121 – Інженерія програмного забезпечення (Освітньо-професійна програма – «Програмне забезпечення систем», Освітньо-наукова програма – «Інженерія програмного забезпечення») для студентів усіх форм навчання / Упор.: М.Р. Петрик, Д.М. Михалик, О.Ю. Петрик, Г.Б. Цуприк - Тернопіль: ТНТУ, 2020-51с.

ДОДАТКИ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ
Кафедра програмної інженерії

ТЕХНІЧНЕ ЗАВДАННЯ
на виконання кваліфікаційної роботи магістра
«Розробка CRM системи для контролю та управління взаємовідносинами із
потенційними клієнтами»

Керівник роботи:
ст. викладач, PhD Мудрик І.Я.
“ ___ ” _____ 2022р.

Виконавець:
студентка групи СПм-61
Голинська Олена Олегівна
“ ___ ” _____ 2022р.

ЗМІСТ

1. ПІДСТАВИ ДО РОЗРОБКИ
2. ПРИЗНАЧЕННЯ РОЗРОБКИ
3. ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ
 - 3.1 Функціональні вимоги
 - 3.2 Технічні вимоги
 - 3.3 Програмні вимоги
4. СТАДІЇ НАПИСАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ
5. СУПРОВІДНА ДОКУМЕНТАЦІЯ
6. ПОРЯДОК ЗДАЧІ КВАЛІФІКАЦІЙНОЇ РОБОТИ МАГІСТРА
7. ВІДМІТКИ ПРО ВИКОНАННЯ ЕТАПІВ В РОБОТІ

1 ПІДСТАВИ ДО РОЗРОБКИ

Розробка проводиться у відповідності до графіку навчального плану підготовки магістрів за спеціальністю «Інженерія програмного забезпечення» 2021 - 2022 р.

Тема кваліфікаційної роботи магістра: «Розробка CRM системи для контролю та управління взаємовідносинами із потенційними клієнтами».

Термін виконання: до __.__._____р.

2 ПРИЗНАЧЕННЯ РОЗРОБКИ

Для розробки CRM системи для контролю та управління взаємовідносинами із потенційними клієнтами було використано мову програмування Java, фреймворк Spring, HTML/CSS, Javascript, JQuery та Bootstrap. В якості бази даних для даної системи було обрано PostgreSQL.

Зменшення числа клієнтів, а також велика кількість негативних відгуків погано впливає на імідж компанії. Тому, що клієнти, які незадоволені обслуговуванням, зазвичай схильні більш розширено розповідати про свій досвід ніж задоволені. Тому для компаній будь-якого виду діяльності важливо залучати клієнтів та правильно взаємодіяти з ними. Для вирішення подібного роду проблем призначені CRM системи, які можуть збирати та обробляти величезний обсяг інформації щодо взаємовідносин з клієнтами. Такі системи надають можливість автоматизованого управління, а також є чудовим інструментом для аналізу даних, необхідних для побудови ефективної взаємодії та отримання прибутку.

Зовсім недоцільно вважати, що CRM системи призначені лише для великих підприємств, оскільки це не так. Не важливо у якій галузі працює компанія і якого розміру, автоматизована система стане незамінною частиною на шляху до розвитку та росту компанії.

3 ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ

3.1 Функціональні вимоги

Система повинна надавати користувачу такі можливості:

- Залучення нових клієнтів використовуючи різні канали зв'язку.
- Організована клієнтська база.
- Аналіз залежності одних показників від інших та їхнє графічне представлення.
- Можливість використання системи будь-яким видом бізнесу.
- Автоматизація обліку.
- Авторизація в залежності від наданих прав доступу.
- Можливість змінити пароль та інформацію профілю, включаючи медіа дані.
- Можливість відновити акаунт, якщо забули пароль.

3.2 Технічні вимоги

Вимоги до серверної частини: ОС Linux, Windows не менше ніж 4Гб ОЗП;

Вимоги до клієнтської частини: Наявність браузера, пристрої вводу і виводу інформації;

Додаткові вимоги: наявне підключення до мережі Інтернет, автоматичне резервування, забезпечення одночасної роботи до 1000 клієнтів.

3.3 Програмні вимоги

Використання СУБД: PostgreSQL.

Розробка клієнтської частини: мова програмування JavaScript, бібліотека Bootstrap, HTML/CSS, бібліотека JQuery.

Розробка серверної частини: Java, фреймворк Spring.

Середовище розробки: IntelliJIdea.

4 СТАДІЇ НАПИСАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Написання кваліфікаційної роботи проводиться в наступному порядку:

- вибір та затвердження теми
- аналіз предметної області
- постановка задач розробки
- проектування діаграми варіантів використання
- вибір СКБД та опис її фізичної моделі
- опис програмної реалізації
- опис схем використання програмного забезпечення
- тестування програмного забезпечення
- написання розділу «Охорона праці»
- написання розділу «Безпека в надзвичайних ситуаціях»
- оформлення записки кваліфікаційної роботи магістра
- попередній захист
- нормоконтроль
- захист кваліфікаційної роботи магістра

Результати виконання кожного етапу кваліфікаційної роботи магістра погоджуються з керівником роботи.

5 СУПРОВІДНА ДОКУМЕНТАЦІЯ

Для кваліфікаційної роботи магістра повинні бути розроблені наступні документи:

- записка кваліфікаційної роботи;
- презентація;
- рецензія на кваліфікаційну роботу магістра;
- відгук керівника на кваліфікаційну роботу магістра;
- авторська довідка;
- протокол аналізу звіту подібності керівником роботи;
- диск з кваліфікаційною роботою магістра.

Записка кваліфікаційної роботи магістра оформляється згідно діючих вимог до нормоконтролю.

6 ПОРЯДОК ЗДАЧІ КВАЛІФІКАЦІЙНОЇ РОБОТИ МАГІСТРА

Розроблена системи повинна відповідати вимогами, що складаються з перерахованих у 3 розділі цього документу характеристик.

Для здачі проекту необхідно підготувати весь перелік документів зазначений у розділі 5 цього документу.

Приймання проекту проводиться спеціально створеною комісією в термін зазначені в розділі 1 цього документу.

7 ВІДМІТКИ ПРО ВИКОНАННЯ ЕТАПІВ В РОБОТІ

Назва етапу	Відмітка *
Вибір та затвердження теми	
Аналіз предметної області	
Постановка задач розробки	
Проектування діаграми варіантів використання	
Вибір СКБД та опис її фізичної моделі	
Опис програмної реалізації	
Опис схем використання програмного забезпечення	
Тестування програмного забезпечення	
Написання розділу «Охорона праці»	
Написання розділу «Безпека в надзвичайних ситуаціях»	
Оформлення записки кваліфікаційної роботи магістра	
Попередній захист	
Нормоконтроль	
Захист кваліфікаційної роботи магістра	

* відмітки про виконання етапу ставляться керівником проекту

Додаток Б – Публікація у науковому виданні

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ**

МАТЕРІАЛИ

Х НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

**«ІНФОРМАЦІЙНІ МОДЕЛІ,
СИСТЕМИ ТА ТЕХНОЛОГІЇ»**



7–8 грудня 2022 року

**ТЕРНОПІЛЬ
2022**

УДК 004.45, 004.9 О. Голинська, І. Мудрик

(Тернопільський національний технічний університет імені Івана Пулюя,
Україна)

РОЛЬ CRM-СИСТЕМИ У СУЧАСНИХ БІЗНЕС-ПРОЦЕСАХ

О. Holyns'ka, I. Mudryk

ROLE OF CRM SYSTEM IN MODERN BUSINESS PROCESSES

У сучасну інформаційну епоху, де все взаємопов'язано, усі традиційні речі зазнають інновацій, викликаних технологічним прогресом. Завдяки передовій мережевій технології режим роботи комерційного ринку також дещо змінився за останні десять років, і підприємства також почали використовувати різні системи програмного забезпечення для підтримки власної діяльності та розвитку. Немає сумніву, що клієнти є джерелом корпоративних прибутків, і те, як ефективно керувати клієнтами, є ключем до успіху компанії. У сучасному середовищі жорсткої конкуренції пристосування до потреб клієнтів, надання клієнтам права вибору продуктів і надання їм можливості отримати те, що вони дійсно хочуть, є ключем до конкурентного успіху будь-якої компанії.

Система управління взаємовідносинами з клієнтами є важливим елементом основної програми підприємства. У простому розумінні CRM – це центр обробки даних клієнтів, який може обробляти різні деталі всіх клієнтів компанії в CRM, включаючи відстеження та керування кожним потенційним клієнтом, який взаємодіє із корпоративним брендом, веб-сайтом або продуктом, а також може аналізувати його сліди та переваги. Зворотній зв'язок ринку протягом багатьох років довів, що це дуже допомагає розширювати бізнес підприємств, тому сучасні підприємства інвестували багато ресурсів у розвиток клієнтів, щоб швидше та ефективніше розширювати свою територію.

Роль CRM полягає в тому, щоб допомогти підприємствам скрутити відповідні відділи в єдину мотузку. Коли підприємство вступає в стадію швидкого зростання, деякі клієнти можуть бути маргіналізованими. Як і інші корпоративні прикладні системи, CRM-система також дуже допомагає автоматизувати роботу підприємств, прискорити процес виконання різноманітних завдань.

Таким чином, важливість CRM-систем у сучасному бізнес-середовищі добре відома як розробникам, так і потенційним та реальним клієнтам. Така система збирає, обробляє та керує всіма даними клієнтів, які є основою кожного бізнесу. CRM аналізує активних і потенційних клієнтів, і результати аналізу дозволяють підприємствам налагоджувати міцні ділові відносини з клієнтами, тим самим збільшуючи швидкість транзакцій. Що ще важливіше, CRM може допомогти компаніям знайти нових клієнтів і розвивати новий бізнес.

Література

1. Glova B., Mudryk I. Application of Deep Learning in Neuromarketing Studies of the Effects of Unconscious Reactions on Consumer Behavior. 2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP): Conference, Lviv, 21–25 August 2020. P. 337–340.
2. Kumar, V., & Reinartz, W. (2018). Customer Relationship Management. Springer Texts in Business and Economics. Doi:10.1007/978-3-662-55381-7

А. Блавіцький, С. Мацюк, С. Криськова ОЦІНКА РОЗВИТКУ БЕЗПЕКИ ОПЛАТИ ПЛАТІЖНИМИ КАРТКАМИ A. Blavitskyi, S. Matsiuk, S. Kryskova ASSESSMENT OF THE SECURITY DEVELOPMENT OF PAYMENT CARDS	17
А. Буковська ПАРАЛЕЛЬНЕ ТА РОЗПОДІЛЕНЕ ГЕНЕРУВАННЯ POWERSET З ВИКОРИСТАННЯМ ПЛАТФОРМИ ОБРОБКИ ВЕЛИКИХ ДАНИХ A. Bukovska PARALLEL AND DISTRIBUTED POWERSSET GENERATION USING A BIG DATA PLATFORM	18
В. Василенко, Н. Стадник ВИКОРИСТАННЯ СТАКУ ELK ДЛЯ ДОСЛІДЖЕННЯ ПОДІЙ V. Vasylenko, N. Stadnyk USING ELK STACK TO RESEARCH OF EVENTS	20
В. Василенко, Н. Стадник ЛОГУВАННЯ – ЩО ЦЕ І В ЧОМУ ЙОГО КОРИСТЬ V. Vasylenko, N. Stadnyk LOGGING – WHAT IS IT AND WHAT IS ITS BENEFIT	21
Р. Волошин АУДИТ БЕЗПЕКИ AMAZON SELLING PATRNER API R. Voloshyn AMAZON SELLING PATRNER API CYBERSECURITY AUDIT	22
І. Воробець ПОРІВНЯННЯ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ I. Vorobets COMPARISON OF TIME SERIES FORECASTING METHODS	23
М. Гаврилов ПОВТОРНА ІДЕНТИФІКАЦІЯ ЛЮДЕЙ ЗА ФОТО ТА ВІДЕО ЗАСОБАМИ COMPUTER VISION M. Havrylov RE-IDENTIFICATION OF PEOPLE FROM PHOTOS AND VIDEOS BY MEANS OF COMPUTER VISION	24
О. Голінська, Я. Мудрик РОЛЬ CRM-СИСТЕМИ У СУЧАСНИХ БІЗНЕС-ПРОЦЕСАХ O. Holyns'ka, Lecturer, ROLE OF CRM SYSTEM IN MODERN BUSINESS PROCESSES	25
В. Грицюк, М. Стадник КЛАСТЕРИЗАЦІЯ СПАМ-ДОМЕНІВ МЕТОДАМИ МАШИННОГО НАВЧАННЯ V. Hrytsiuk, M. Stadnyk SPAM DOMAINS CLUSTERIZATION BY USING MACHINE LEARNING METHODS	26
Н. Зарічний, Є. Тиш АВТОМАТИЗАЦІЯ ТЕСТУВАННЯ МОБІЛЬНИХ ДОДАТКІВ ЗА ТЕХНОЛОГІЄЮ AGILE N. Zarichnyi, Ye. Tysh, Ph.D. AUTOMATION OF MOBILE APPLICATION TESTING USING AGILE TECHNOLOGY	27
О. Кравчук ВИЗНАЧЕННЯ ПОГОДНИХ УМОВ У TELEGRAM O. Kravchuk DETERMINATION OF WEATHER CONDITIONS IN TELEGRAM	28

Додаток Г – Диск із кваліфікаційною роботою магістра