

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

РОЗРОБЛЕННЯ СЕРВІСУ «FarmInTime»

Кваліфікаційна робота

на здобуття ступеня вищої освіти “бакалавр”

Виконала: студентка 4 курсу

Спеціальності: 121 Інженерія програмного
забезпечення

Освітньо-професійної програми:

Інженерія програмного забезпечення

Пчелінцева Тетяна Олександрівна

Керівник: доктор пед.н., професор

Вінник М.О.

Співкерівник: доктор педагогічних наук,
професор Співаковський О.В.

Рецензент: Senior Software

Engineer (EPAM)

Кожевніков Д.І.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	3
ВСТУП	4
РОЗДІЛ 1. ТЕОРЕТИЧНИЙ ОГЛЯД.....	6
1.1 АНАЛІЗ ПРОБЛЕМ ЦІЛЬОВОЇ АУДИТОРІЇ	6
1.2 ЗАСОБИ РОЗРОБКИ ДОДАТКУ	7
1.2.1 ТЕХНОЛОГІЇ FRONT-END ПРОГРАМУВАННЯ	7
1.2.2 ТЕХНОЛОГІЇ BACK-END ПРОГРАМУВАННЯ	12
РОЗДІЛ 2. ПРОЕКТУВАННЯ СЕРВІСУ	15
2.1 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	15
2.3 КОНЦЕПТУАЛЬНА МОДЕЛЬ UML	17
2.3 МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБПЕЧЕННЯ	20
2.3.1 CLASS DIAGRAM.....	20
2.3.2 ДІАГРАМА ПРЕЦЕДЕНТІВ	22
2.3.3 ДІАГРАМА ПОСЛІДОВНОСТІ.....	23
РОЗДІЛ 3. РОЗРОБКА СЕРВІСУ	24
3.1 ДИЗАЙН – МАКЕТ	24
3.2 FRONT-END ПРОГРАМУВАННЯ	28
3.3 BACK-END ПРОГРАМУВАННЯ	29
ВИСНОВОК	30
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	32

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

UML – Unified Modeling Language

ПЗ – Програмне забезпечення

ТЗ – Технічне завдання

БД - База даних

ОС - Операційна система

API - Application Programming Interface

IDE - Інтегроване середовище розробки

JSON - JavaScript Object Notation

HTTP - HyperText Transfer Protocol

SQL - Structured Query Language

URL - Uniform Resource Locator

SDK - Software Development Kit

JDK - Java Development Kit

HTML - HyperText Markup Language

CSS - Cascading Style Sheets

ISO - International Organization for Standardization

СУБД - Система Управления Базами Данных

ВСТУП

Сільське господарство – один із провідних секторів української економіки. У сільській місцевості проживає близько третини населення країни, і від ступеня розвитку сільськогосподарського виробництва багато в чому залежить життєвий рівень не лише тих, хто працює тут, а й тих, хто тією чи іншою мірою пов'язаний із цією сферою. Однією з гострих проблем агропромислового виробництва є невисока оперативність та ефективність прийнятих управлінських рішень через недостатнє використання нових інформаційних технологій. Сільське господарство - ідеальне середовище для впровадження інформаційних технологій. У зв'язку з цим для ефективного та сталого зростання продуктивності сільського господарства необхідно застосовувати передові інформаційні технології, що дозволяють виявити недоліки та залучити інвестиції.

Сьогодні складно уявити комерційну діяльність без власної веб-сторінки, майже у будь-якої організації на зараз є приватний сайт. Застосування онлайн сервісів, соціальних мереж є невід'ємною частиною існування бізнесу, що створює сприятливі умови для реклами на велику аудиторію та дає зацікавлення додаткових покупців. Однак, не звертаючи уваги на сучасну важливість та невід'ємність інтернет-простору, не кожна комерційна сфера діяльності, а саме комунікація між незалежними фермерами та споживачами була реалізована та долучена до рядів онлайн сервісів.

У попередніх експериментальних роботах були проаналізовані потреби людей, які вимагають реалізації та купівлі товарів аграрного господарства не в промисловій, кількості, також був проведено аналіз та порівняння наявних реалізацій трейдерських сервісів. Дане дослідження продемонструвало, що у сфері сільськогосподарської індустрії немає платформ, що мають необхідний функціонал і не можуть бути

використані на території України як сервіс-посередник для фермерів та клієнтів, що у свою чергу засвідчило важливість вибраних цілей.

Отже, як об'єкт аналізу, було вирішено розпочати проектування та розробку платформи-посередника між фермерами та споживачами - сервісу «FarmInTime».

Актуальність розробки сервісу визначена відсутністю сервісів з відповідним функціоналом, що може займати роль посередника у масовій комунікації фермерів та охочих придбати продукти аграрної промисловості. Створення «FarmInTime» спрямовано на вирішення проблеми з реалізацією та пошуку продукції.

Метою кваліфікаційної роботи є проектування та розробка сервісу «FarmInTime».

Об'єкт дослідження – технології створення Web-додатку.

Предметом дослідження є розробка сервісу «FarmInTime».

Завдання дослідження:

- Проектування «FarmInTime».
- Створення дизайн – макету
- Front-end програмування
- Back-end програмування
- Тестування

РОЗДІЛ 1. ТЕОРЕТИЧНИЙ ОГЛЯД

У цьому розділі дипломної роботи описаний процес розробки сайту: Перший крок – проведення досліджень, які визначають, як виглядатимуть наступні кроки. Завданням на даний момент - чітко розуміння цілей майбутнього веб-сайту та цільової аудиторії. Завдяки ретельно описаному та детальному плану можна уникнути витрат додаткових ресурсів на вирішення непередбачених проблем, таких як зміна дизайну або додавання функціоналу, які спочатку не планувалася. Другий крок – вибір технологій за допомогою яких буде реалізовано сервіс.

1.1 АНАЛІЗ ПРОБЛЕМ ЦІЛЬОВОЇ АУДИТОРІЇ

Створення сервісу «FarmInTime» є актуальним зараз, оскільки існує проблема швидкої реалізації товару, значних збитків та співробітництва без участі третьої сторони.

Майже вся реалізації товару від невеликих господарств відбувається саме на стихійних ринках яких багато як в містах, та поза їх межами, на станціях, зупинках та на узбіччі автомагістралей але стихійна торгівля заборонена Законом України «Про благоустрій населених пунктів» «Торгівля в містах з рук на вулицях, площах, у дворах, під'їздах, скверах та в інших невстановлених місцях - тягне за собою накладення штрафу від одного до семи неоподатковуваних мінімумів доходів громадян з конфіскацією предметів торгівлі або без такої. Торгівля з рук у невстановлених місцях промисловими товарами - тягне за собою накладення штрафу від шести до п'ятнадцяти неоподатковуваних мінімумів доходів громадян з конфіскацією предметів торгівлі [16] .

Розробка сервісу «FarmInTime» спрямована на покращення торгівельної справи серед незалежних господарств та споживачів.

1.2 ЗАСОБИ РОЗРОБКИ ДОДАТКУ

Перш ніж буде описано технології та інструменти, які були використані для створення сервісу. Буде наведено кілька термінів, які допоможуть краще зрозуміти взаємодію веб технологій.

Front-end: Це частина клієнта, тобто все що знаходиться на стороні клієнта. До front-end відноситься такі веб технології як - CSS, HTML, JavaScript - за допомогою яких створюється інтерфейс користувача.

Back-end: Це серверна частина, на якій знаходиться ядро сайту, це ядро забезпечує основний функціонал та підтримку front-end. У back-end використовується одна з серверних мов програмування, такі як Java, Python, Ruby, Perl та інші. Серверні мови програмування розраховані працювати з базами даних, такі як - MySQL, MongoDB тощо.

Далі будуть наведені веб-технології, на основі яких реалізовано сервіс «FarmInTime».

1.2.1 ТЕХНОЛОГІЇ FRONT-END ПРОГРАМУВАННЯ

HTML

HTML – це мова розмітки гіпертекстових документів. HTML описує структуру веб-сторінки за допомогою тегів (HTML елементів), ці елементи визначають розташування тексту, малюнка, таблиць та багато іншого.

HTML (або XHTML) є основним інструментом створення веб-сторінок. Мова HTML інтерпретується браузерами та відображається у вигляді документа, у зручній для людини формі. HTML є додатком («приватним випадком») SGML (стандартної узагальненої мови розмітки) і відповідає міжнародному стандарту ISO 8879.

HTML створювався як мова обміну між людьми, які не мають навичок у програмуванні науковою, технічною документацією. HTML успішно справлявся із проблемою складності SGML шляхом визначення невеликого набору структурних елементів та дескрипторів. HTML надає змогу, без великих зусиль, зробити нескладний, гарно оформлений документ. До HTML внесена підтримка гіпертексту. Мультимедійні можливості було додано пізніше.

Спочатку мова HTML була задумана і створена як засіб структурування та форматування документів без їх прив'язки до засобів відтворення (відображення). В ідеалі, текст з розміткою HTML повинен був без стилістичних та структурних спотворень відтворюватися на обладнанні з різним технічним оснащенням (кольоровий екран сучасного комп'ютера, монохромний екран органайзера, обмежений за розмірами екран мобільного телефону або пристрою та програми голосового відтворення текстів).

Однак сучасне застосування HTML дуже далеке від його початкового завдання. З часом основна ідея платформонезалежності мови HTML була віддана в своєрідну жертву сучасним потребам у мультимедійному та графічному оформленні. Текстові документи, що містять розмітку на мові HTML (такі документи традиційно мають розширення .html або .htm), обробляються спеціальними програмами, які відображають документ у форматovanому вигляді.

Такі програми, які називаються "браузерами" або "інтернет-браузерами", зазвичай надають користувачеві зручний інтерфейс для запити web-сторінок, їх перегляду (і виведення на інші зовнішні пристрої) і, при необхідності, відправлення введених користувачем даних на сервер. Найбільш популярними сьогодні браузерами є Internet Explorer, Mozilla Firefox, Apple Safari, Google Chrome і Opera.

CSS

CSS – це правила, що описують стиль (вид) HTML документа. За допомогою CSS, веб-сторінка набуває зорово зрозумілого інтерфейсу. CSS описує правила для HTML тегів, тексту, зображень, змінюючи тим самим - розташування елементів відносно один одного, розмір, колір, шрифт та інші візуальні зміни.

Основною метою розробки CSS було розділення опису логічної структури web-сторінки (яке виробляється за допомогою HTML або інших мов розмітки) від опису зовнішнього вигляду цієї web-сторінки (яке тепер виробляється за допомогою формальної мови CSS). Такий поділ може збільшити доступність документа, надати більшу гнучкість та можливість управління його поданням, а також зменшити складність та повторюваність у структурному вмісті. Крім того, CSS дозволяє представляти той самий документ у різних стилях або методах виводу, таких як екранне представлення, друковане представлення, читання голосом[15].

Каскадні таблиці стилів або CSS (від англійської Cascading Style Sheets) – це досить пізнє нововведення. Якщо сам HTML з'явився 1989 року, то таблиці стилів - лише 1997. Мало того, таблиці стилів не вважаються частиною HTML. Пов'язано це з тим, що WWW була створена вченими як для обміну текстовими документами, а HTML був мовою, за допомогою якої створювали ці документи. Для вчених головним був вміст документа, а не його оформлення. Тому перші версії HTML не включали навіть засобів для розміщення на Web-сторінках графічних зображень, не те що для складного форматування тексту.

Каскадні таблиці стилів або просто таблиці стилів (CSS - Cascading Style Sheets) - це набір правил, що описують форматування

різних фрагментів HTML-коду та зберігаються окремо від нього. Одне таке правило, що відображає форматування одного фрагмента або однотипної групи фрагментів коду, називається стилем. Таблиці стилів описуються особливою мовою CSS і зберігаються в спеціальних файлах з розширенням CSS, хоча можуть бути впроваджені в саму Web-сторінку.

Усього стандарт CSS визначає три способи завдання стилю для елемента сторінки. Перерахуємо їх та дамо короткий опис кожному способу.

1. Зовнішня (або прив'язана) таблиця стилів. Стили зберігаються в окремому файлі з розширенням CSS і прив'язуються до Web-сторінці за допомогою особливого тегу <LINK>.

2. Внутрішня (або впроваджена) таблиця стилів. Таблиця стилів у цьому випадку має той самий формат, що й зовнішня, але розташовується в секції заголовка тієї ж Web-сторінки і вміщується всередину спеціального тега <STYLE>.

3. Внутрішні (також вбудовані чи впроваджені) стилі. Визначення стилю поміщається у потрібний тег, навіщо використовується спеціальний атрибут STYLE.

Правило каскадності: якщо якийсь стиль буде визначено одночасно два чи три рази, то до визначення стилю, зробленого у зовнішній таблиці, додає визначення, зроблене у внутрішній таблиці. А якщо визначення зачіпає той самий атрибут, береться визначення, зроблене у внутрішній таблиці.

JavaScript

JavaScript – це мова програмування на стороні клієнта. Він розроблений для того щоб виконувати будь-які програмні дії в браузері, надаючи тим самим інтерактивність на веб-сторінці. JavaScript дозволяє маніпулювати HTML елементами, CSS стилями веб-сторінки та багато іншого. Для обміну даними між Front-end та Back-end, використовується протокол HTTP. Це протокол для обміну інформацією мережі інтернет.

JavaScript одна з трьох основних мов програмування для веб-розробників:

HTML: дозволяє додати вміст на веб-сторінку.

CSS: визначає стилі та зовнішній вигляд веб-сторінки.

JavaScript: покращує поведінку веб-сторінки.

Синтаксис мови JavaScript багато в чому нагадує синтаксис Сі і Java, семантично ж мова набагато ближче до Self, Smalltalk або навіть Лиспу.

Структурно JavaScript можна як об'єднання трьох чітко помітних друг від друга елементів: ядро (ECMAScript); об'єктна модель браузера (Browser Object Model або BOM(de)); об'єктна модель документа (Document Object Model чи DOM). Якщо розглядати JavaScript в оточенні від браузера, то об'єктна модель браузера і об'єктна модель документа можуть не підтримуватися. Об'єктну модель документа іноді розглядають як окрему від JavaScript сутність, що відповідає визначенням DOM як незалежного від мови інтерфейсу документа. На противагу цьому ряд авторів знаходять BOM та DOM тісно взаємопов'язаними. [14]

1.2.2 ТЕХНОЛОГІЇ BACK-END ПРОГРАМУВАННЯ

Java

Java - одна з найпопулярніших мов програмування. Вона універсальна та використовується вже понад 20 років.

Універсальність забезпечується віртуальною машиною Java (Java Virtual Machine, JVM). Багато мов під час компіляції програма переводиться в код, який може працювати по-різному на різних пристроях або платформах. У Java цієї проблеми немає. JVM грає роль проміжного рівня - з програми Java вона робить код, який може виконуватися на будь-якому комп'ютері незалежно від того, де код був скомпільований.

Java має велику спільноту, і в інтернеті можна знайти відповіді практично на будь-які питання про мову.

Технології Java можуть використовуватися для багатьох завдань, серед яких:

- розробка мобільних програм під Android;
- розробка веб-сайтів;
- розробка API для роботи з базами даних;
- цифрове оброблення зображень;
- створення настільних додатків;
- програмування мережевих завдань.

Моїм вибором стала Java тому що це найпоширеніша мова програмування у світі та навколо неї утворилося найбільше співтовариство програмістів. Також Java – це не тільки мова програмування, але й величезна кількість інструментів та технологій, що охоплюють усі етапи створення та супроводу ПЗ.

MySQL

SQL (Structured Query Language) - це мова програмування, за допомогою якої можна керувати інформацією: додавати, модифікувати, видаляти та отримувати дані. Запити до бази даних формуються мовою SQL.

MySQL працює за принципом клієнт-сервер. Комп'ютер користувача (клієнт) надсилає запит. Сервер баз даних його обробляє та надає відповідь. Ось тому часто можна почути поняття MySQL-сервер. Це сервер, де зберігається база даних.

SQL використовується у MySQL. Багато РСУБД (реляційні системи управління базами даних) використовують цю мову для роботи з даними. Наприклад:

- Microsoft SQL Server
- PostgreSQL
- Oracle Database
- MariaDB
- SQLite

SQL використовується у запитах при зверненні до бази даних.

Ця СУБД працює як на Linux, MacOS і Windows, так і інших менш відомих операційних системах. Тому MySQL дуже популярна серед розробників сайтів та веб-додатків. Її використовують у своїй роботі такі великі компанії, як Tesla, Netflix, Cisco, PayPal та інші.

До основних переваг MySQL також можна віднести такі:

- повністю безкоштовна СУБД

- необмежений розрахований на багато користувачів режим
- безліч плагінів, що полегшують роботу з даною СУБД
- підтримує різні типи таблиць (MyISAM, InnoDB, HEAP, MERGE)
- дозволяє додавати до 50 мільйонів рядків у таблиці

РОЗДІЛ 2. ПРОЕКТУВАННЯ СЕРВІСУ

2.1 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Бізнес вимоги

Ідея проекту – створення онлайн сервісу «FarmInTime», який виконує роль посередника між сільськогосподарською сферою та споживачами. Інтернет-платформа представляє собою онлайн дошку оголошень в категоріях пропозиція та попит. Після вдалого запуску проекту, в перспективі під'єднання монетизації, розміщення реклами, просування оголошень, створення форми підписки з різними рівнями тарифами та можливостями. До основних послуг належать створення публікація та пошук оголошень.

Пчелінцева Т. Проектування сервісу «FarmInTime»(Херсон,2021): «

Вимоги користувачів

НЕАВТОРИЗОВАНИЙ КОРИСТУВАЧ

- наявність опції входу до облікового запису
- наявність опції реєстрації
- наявність опції перегляду оголошень інших користувачів
- наявність опції пошуку за словом
- наявність опції зворотнього зв'язку з онлайн консультантом
- наявність опції фільтрації пошуку

ЗАРЕЄСТРОВАННИЙ КОРИСТУВАЧ

Високий пріоритет

- опції опублікування оголошення
- опції перегляду оголошень інших користувачів

- опції перегляду контактної інформації оголошення
- отримання переліку потенційних угод з опублікованими оголошеннями
- опції видалення облікового запису
- опції зміни облікового запису

Середній пріоритет

- опції прикріплення фото, відео до оголошення
- опції редагування оголошення
- опції редагування інформації у профілі
- опції видалення оголошення
- опції пошуку за словом
- опції фільтрації пошуку

Низький пріоритет

- отримання повідомлення про порушення правил розміщення оголошень

АДМІНІСТРАТОР

- надсилання попередження про порушення правил користувачам
- видалення оголошення
- блокування користувача
- видалення користувача
- отримання повідомлень від користувачів»

Нефункціональні вимоги

Візуал сервіса виконаний у біло-зелених відтінках, фіксована панель фільтрів оголошень.

Однаково швидка робота на різноманітних пристроях (ПК, телефони, смартфони, планшетні комп'ютери).

Робота на Chrome, Safari, Opera, Firefox.

Проект має бути написаний мовами Java, JavaScript, MySQL.

Використовувати бази даних MySQL версії 8.0 та вище.

Працювати на веб-сервері Apache.

Веб-сайт повинен мати частину користувача.

У разі адміністративної частини повинна бути передбачена авторизація для входу в панель управління сайтом для його зміни.

Не зберігати паролі у БД у відкритому вигляді.

2.3 КОНЦЕПТУАЛЬНА МОДЕЛЬ UML

UML (скор. від англ. Unified Modeling Language – уніфікована мова моделювання) – мова графічного опису для об'єктного моделювання в галузі розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, який використовує графічні позначення створення абстрактної моделі системи, званої UML моделлю. UML був створений для визначення, візуалізації, проектування та документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей як коду, що інтерпретується, можлива кодогенерація. Використання UML не обмежується моделюванням програмного забезпечення. Його також використовують для моделювання бізнес-процесів, системного проектування та відображення організаційних структур[17].

UML дозволяє розробникам програмного забезпечення досягти угоди в графічних позначеннях для представлення загальних понять (таких як клас, компонент, узагальнення (generalization), об'єднання (aggregation)

та поведінка) та більше сконцентруватися на проектуванні та архітектурі.

Модель представляється у вигляді сутностей та відносин між ними, що показуються на діаграмах.

Сутності – це абстракції, які є основними елементами моделі, зв'язують їх між собою, а діаграми групують набори сутностей, що представляють інтерес[18].

Діаграма – це графічне уявлення набору елементів, найчастіше зображеного як зв'язного графа вершин (сутностей) і шляхів (зв'язків). Мова UML включає 13 видів діаграм, серед яких на першому місці в списку - діаграма класів [18].

Діаграми класів показують набір класів, інтерфейсів, і навіть їх зв'язку. Діаграми цього виду найчастіше використовуються для моделювання об'єктно-орієнтованих систем. Вони призначені для статичного уявлення системи. Більшість елементів UML мають унікальну та пряму графічну нотацію, яка дає візуальне представлення найважливіших аспектів елемента [18].

Сутності

Діаграми класів оперують трьома видами сутностей UML:

- Структурні.
- Поведінкові.
- Ануючі.

Структурні сутності – це «іменники» в моделі UML. В основному, статичні частини моделі, що представляють концептуальні, або фізичні елементи. Основним видом структурної сутності у діаграмах класів є клас[18].

Поведінкові сутності – динамічні частини моделей UML. Це «дієслова» моделей, що представляють поведінку моделі у часі та просторі. Основною з них є взаємодія – поведінка, яка полягає в обміні повідомленнями між наборами об'єктів чи ролей у певному контексті задля досягнення певної мети[18].

Поведінкові сутності

Анотуючі сутності – це пояснюючі частини UML-моделей, іншими словами коментарі, які можна застосувати для опису, виділення та пояснення будь-якого елемента моделі. Головна з анотує сутностей - примітка. Це символ, який служить для опису обмежень і коментарів, які стосуються елемента чи набору елементів. Графічно представлений прямокутником із загнутим кутом; всередині міститься текстовий або графічний коментар[18].

Діаграми використання (англ. use-case diagram) описують функціональність ІС, яка буде помітна користувачам системи. "Кожна функціональність" зображується у вигляді "прецедентів використання" (use case) або просто прецедентів. Прецедент - це типова взаємодія користувача із системою, яка при цьому: описує видиму користувачем функцію, може представляти різні рівні деталізації, забезпечує досягнення конкретної мети, що є важливою для користувача[19].

Діаграми послідовностей (англ. sequence diagram) використовуються для уточнення діаграм прецедентів більш детального опису логіки сценаріїв використання. Вони зазвичай містять об'єкти, які взаємодіють у рамках сценарію, повідомлення, якими вони обмінюються, і результати, пов'язані з повідомленнями, що повертаються. Втім, результати, що часто повертаються, позначають лише в тому випадку, якщо це не очевидно з контексту[20].

2.3 МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБПЕЧЕННЯ

2.3.1 CLASS DIAGRAM

Діаграма класів сервіса «FarmInTime»

Класи:

Наступні класи було сформульовано у праці Пчелінцева Т. Проектування сервісу «FarmInTime»(Херсон,2021): «Guest Class – керує операціями незареєстрованих користувачів, User Class – керує операціями зареєстрованих користувачів, Administrator Class – клас, що відповідає за обробку надісланих повідомлень, та видалення оголошень з порушеннями, WarningMessage Class – клас, що відповідає за формування попереджень користувачам, Proposal Class – клас, що відповідає за всі операції з оголошенням, PotentialDeal Class – генерує потенційні угоди та відповідає за їх обробку, TypeOfProposal Class – керує всіма операціями з типами оголошень, Category Class – керує всіма операціями категорій»

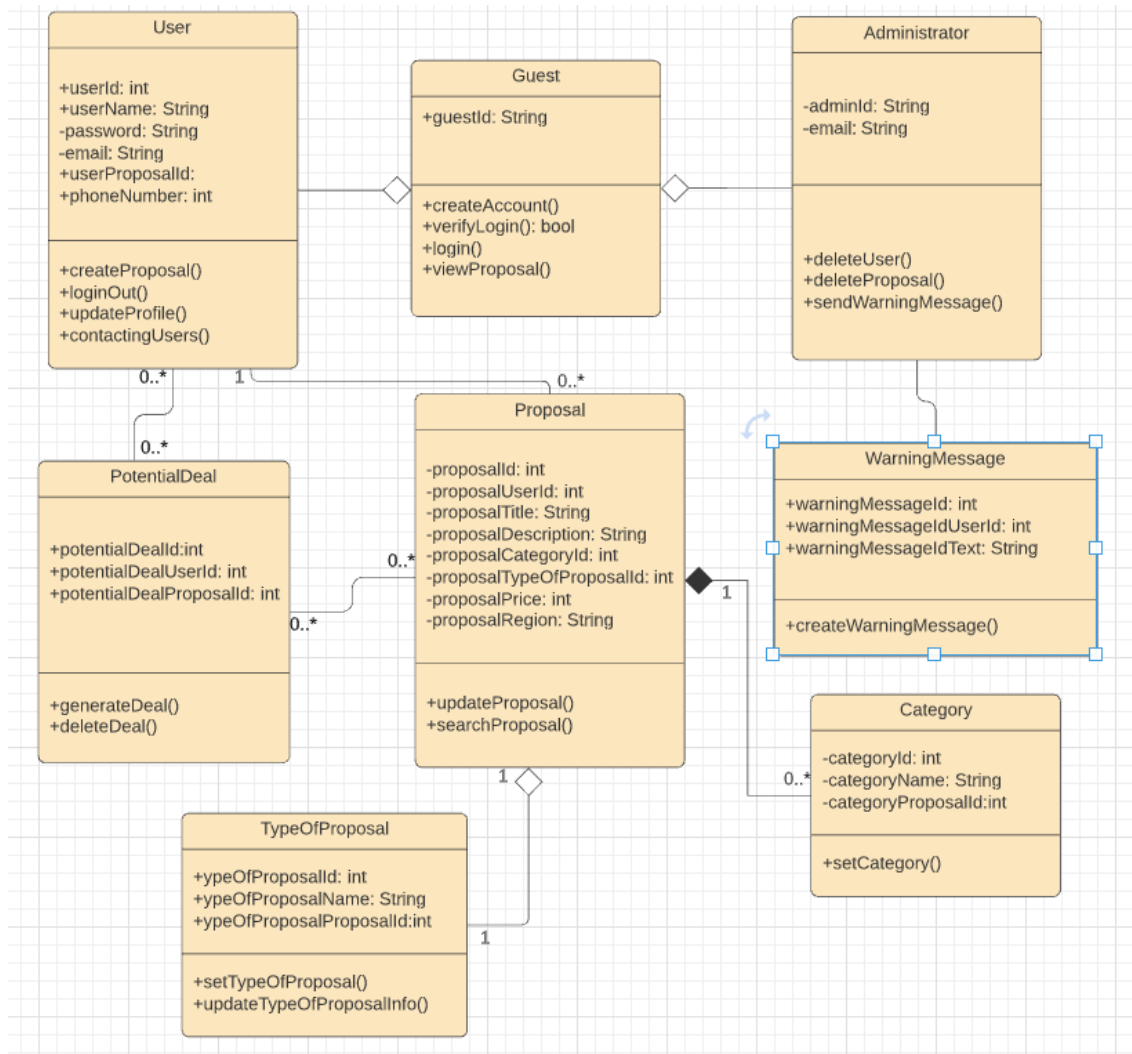


Рисунок 2.0 – Class Diagram

2.3.2 ДІАГРАМА ПРЕЦЕДЕНТІВ

На рисунку 2.2 зображено USE Case діаграму – вона відображає зв'язок елементів між собою, та їх можлива взаємодія у системи «FarmInTime»

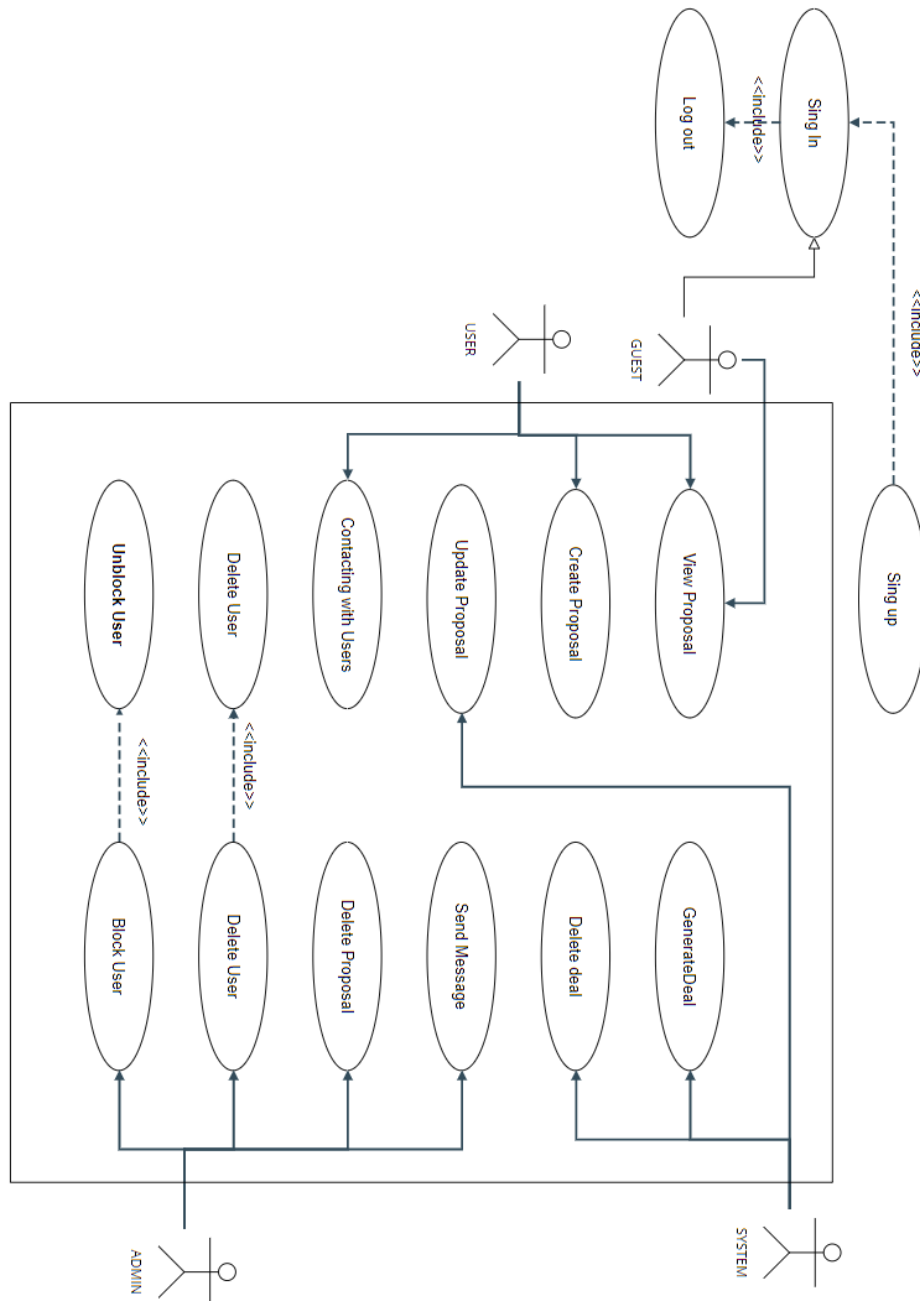
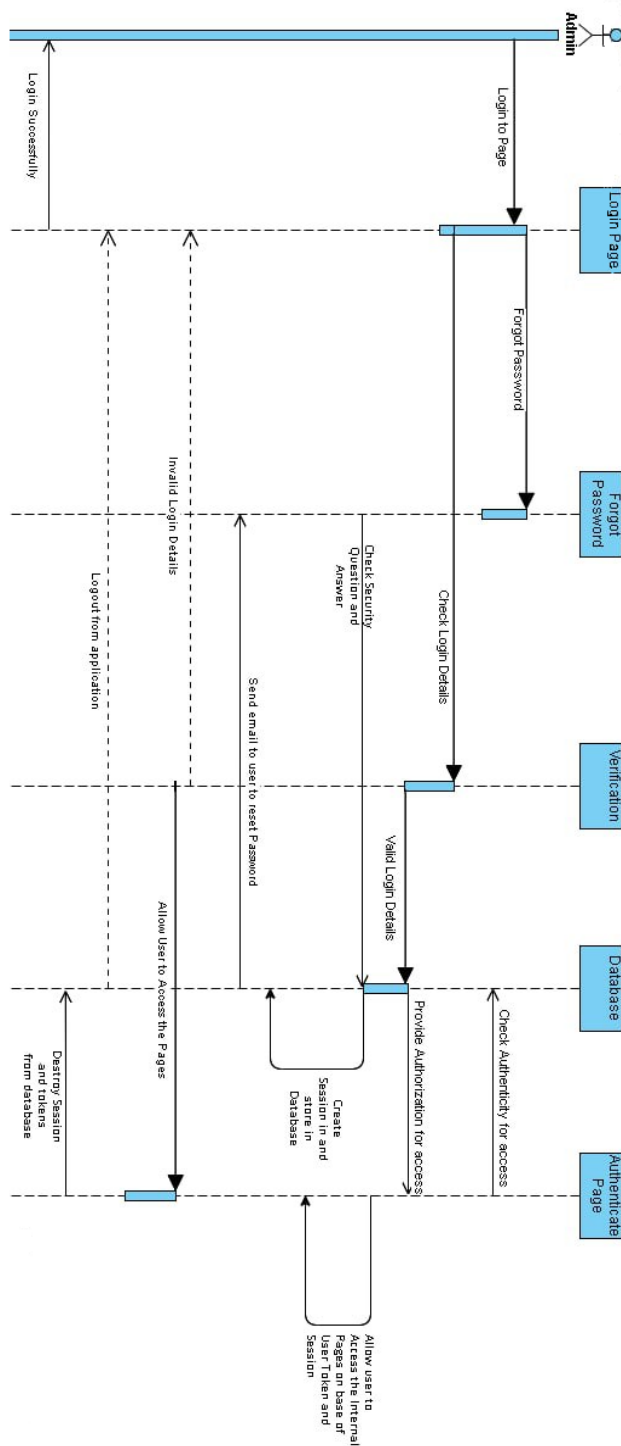


Рисунок 2.2 – USE Case

2.3.3 ДІАГРАМА ПОСЛІДОВНОСТІ

На рисунку 2.3 зображена діаграма послідовності авторизації до сервісу, а саме процес входу адміністратора в обліковий запит, використовуючи свої облікові дані. Після входу користувач може керувати усіма операціями[9].



Рисунку 2.3 – Діаграма послідовності

РОЗДІЛ 3. РОЗРОБКА СЕРВІСУ

3.1 ДИЗАЙН – МАКЕТ

На сьогоднішній день існує безліч інструментів для продуктивної роботи дизайнерів. Лідерами у створенні візуальної частини сайту є перевірений часом Photoshop, більш вузько направлений Sketch та відносно новий сервіс Figma. Останній інструмент швидко стає лідером веб-дизайнерів.

Figma — векторний онлайн-сервіс розробки інтерфейсів та прототипування з можливістю організації спільної роботи, що розробляється однойменною компанією. Працює у двох форматах: у браузері та як клієнтський додаток на десктопі. Зберігає онлайн-версії файлів, з якими працював користувач[11].

Що таке Figma з погляду функціоналу? Це зручний графічний редактор, у якому можна створювати:

- прототипи web-сайтів та додатків;
- окремі елементи інтерфейсу: іконки, кнопки, форми та багато іншого;
- Векторні зображення та ілюстрації тощо.

При цьому потрібно розуміти, навіщо Figma розроблялася. А саме для веб-дизайна. Тут немає підтримки СМҮК, неможливо переключити сітку із сантиметрів на міліметри. Також, на даний момент з макетами із Photoshop Figma працювати не дозволяє.

- Десктопна версія Figma доступна на MacOS та Windows.
- Web версія - у ній доступні всі інструменти та можливості. Підтримуються такі версії: Safari , Chrome, Firefox , Edge

Базові інструменти програми

Figma — це платформа, з необхідним набором інструментів, що дозволяє створити повноцінний дизайн-макет сайту.

Основні види інструментів:

- Фрейми або артборди - є готові варіанти, також можна створювати свої, індивідуально для кожної мети.
- Модульна сітка — інструмент для впорядкування елементів дизайну у кадрах.
- Векторні форми — щоб відобразити різні елементи інтерфейсу.
- Криві — створення кривих та простих векторних форм. Можна додатково завантажити їх із Sketch або Adobe Illustrator.
- Зображення — щоб швидко додати графічні елементи до макету.
- Ефекти та маски – у тому числі з ефектами накладання, градієнтами та можливістю зручно працювати з кількома шарами.
- Текст — із підтримкою Google Fonts та інструментами підключення додаткових за допомогою Font Installers (для десктопної версії).

І це лише мала частина інструментарію. Більше того, Figma підтримує плагіни, що здатні значно розширити базовий функціонал.

Наприклад:

- Grid for Tilda Publishing – інструмент для генерації сітки для Tilda Publishing.

- Unsplash – за допомогою нього лише знадобиться декілька кліків для імпортувати зображення з однойменного сервісу.
- Iconify — відкриває доступ до 40 тисяч векторних іконок для будь-яких ваших проектів.
- Figma — забезпечує можливість додавати анімації прототипам, що звільняє від потреби використання додаткових програмних засобів.
- Vlobs — плагін для роботи з векторними зображеннями.

Хмарне сховище Figma забезпечує зручне зберігання проектів, що дозволяє уникнути використання сторонніх сервісів типу Google Диска або Dropbox. Відповідно, інші редактори також отримують доступ до спільного сховища. Це дуже зручно під час спільного редагування.

Прототипування

До появи Figma, для створення повноцінного дизайн-макету сайту приходилось використовувати декілька програмних засобів одночасно, а саме такі як Invision або Marvel, зараз увесь необхідних функціонал прототипування вже є у Фігмі.

Багатоцільність - можливе перемикання між проектами та внесення змін до них без погіршення роботи Figma, забезпечує найвищу продуктивність роботи на сервісі.

У Figma реалізовано режим презентації, за допомогою посилання на проект можливо переглянути виконану роботу, досконала система роботи із символами, велика бібліотека компонентів із можливістю доповнювати власними, функціональна панель для програмістів, інтеграція із Zerplin та ще багато корисного [12].

Для створення дизайн-макету сервісу було обрано Figma

На рисунках 3.1.1 – 3.1.3 зображено дизайн-макети сервіса.

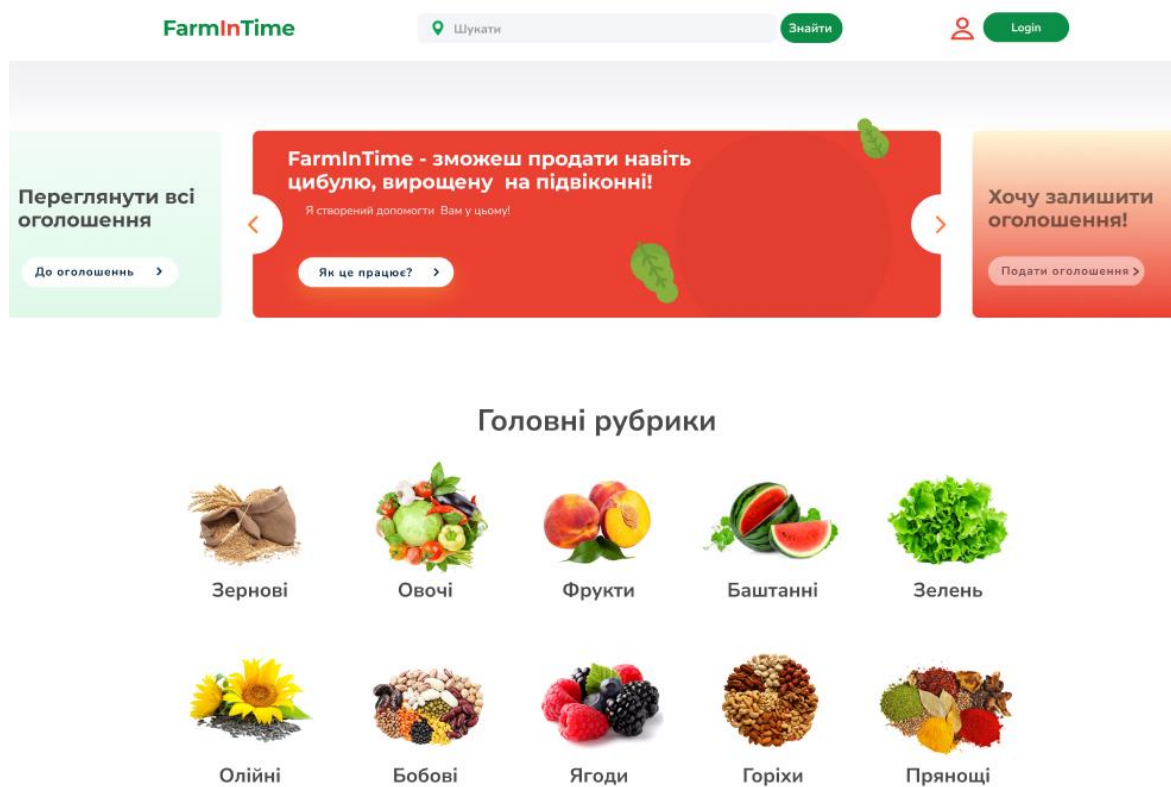
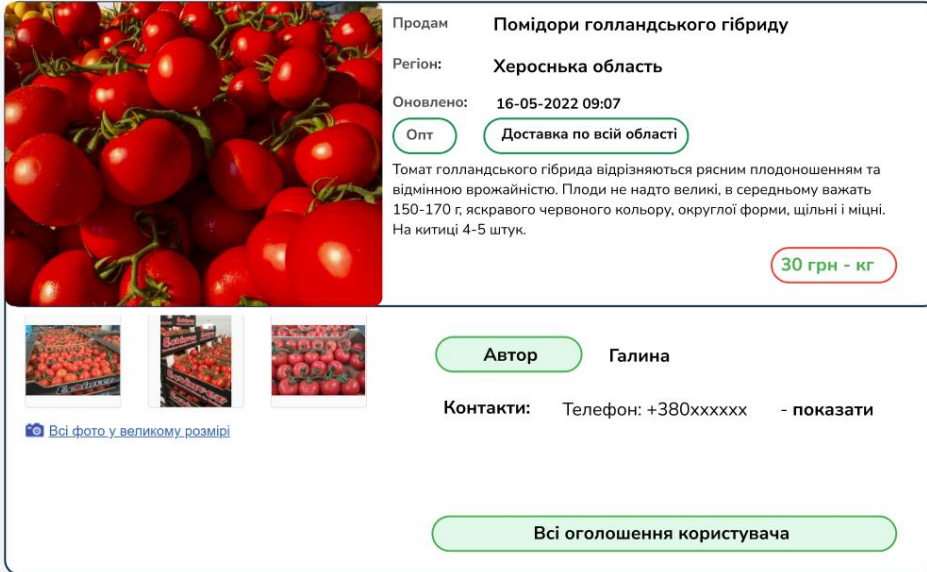


Рисунок 3.1.1 –головна сторінка сервісу

Рисунок 3.1.2 –головна сторінка всіх оголошень



Продам **Помідори голландського гібриду**

Регіон: **Херсонська область**

Оновлено: **16-05-2022 09:07**

[Опт](#) [Доставка по всій області](#)

Томат голландського гібрида відрізняються рясним плодоношенням та відмінною врожайністю. Плоди не надто великі, в середньому важать 150-170 г, яскравого червоного кольору, округлої форми, щільні і міцні. На китиці 4-5 штук.

30 грн - кг

[Всі фото у великому розмірі](#)

Автор Галина

Контакти: Телефон: +380xxxxxx - [показати](#)

[Всі оголошення користувача](#)

Рисунок 3.1.3 – форма оголошення

3.2 FRONT-END ПРОГРАМУВАННЯ

Header– це верхня частина сторінки. У цій частині розміщено логотип сервісу, поле для пошуку оголошень, поле «Подати оголошення», «Увійти» та «Зареєструватися»

Основна система навігації головної сторінки- звідси можна дістатися всіх розділів. Зайшовши в потрібний розділ, ви побачите безпосередньо пропозиції користувачів та координати зв'язку з ними. Далі все просто: шукаєте товар, що вас цікавить, і зв'яжетеся з власником оголошення

Допоміжна система навігації - вона включає елементи, які не відіграють великої ролі у процесі комунікації між собою. До них належать розділи «Про нас» та «Контакти».

На сервісі передбачені фільтри пошуку за типом пропозиції (пропозиція - попит), датою додавання оголошення, регіону України, категорії продукції, групи товарів

3.3 BACK-END ПРОГРАМУВАННЯ

Реалізація серверної частини була виконана з використанням мови Java. Створення бази даних відбувалося з використанням інструменту візуалізації MySQL Workbench.

Для з'єднання з базою даних використовується Java Database Connectivity (JDBC) - це інтерфейс прикладного програмування (API) для мови програмування Java, який визначає, як клієнт може отримати доступ до бази даних.

ВИСНОВОК

Дослідження існуючих потреб людей, які мають проблеми з реалізацією товарів власного аграрного господарства а також їх потенційних клієнтів, разом з підсумками минулих досліджень продемонстрували, що створення сервісу «FarmInTime» може значно покращити ситуацію з прямою співпрацею без посередників, заощадже час, збільше кількість успішних угод, підвище прибутковість та задоволеність потреб обох сторін.

Внаслідок виконання дипломного проекту було створено сервісу «FarmInTime».

Цільовою аудиторією цього додатка є приватні особи, бажаючі продати або купити сільськогосподарську продукцію та продукти фермерства. Основне завдання даної роботи – створення зручної програми, за допомогою якої користувачі можуть переглядати та додавати оголошення.

Головним напрямком розвитку сервісу на далі є розширення функціоналу а саме генерація потенційних угод, розрахунок логістики, впровадження опції чату для користувачів в середині сервісу.

Під час виконання кваліфікаційної роботи було досягнена мета створення сервісу «FarmInTime», що включає в себе наступні успішно виконані задачі

- Аналіз проблем реалізації сільськогосподарської продукції та виділення основних потреб цільової аудиторії.
- Проаналізовано інструменти розробки front-end, back-end частин сервісу

- Розробка функціональних та не функціональних вимог, та бізнес логіки сервісу,
- Створення UML діаграм а саме прецедентів, послідовності, класів
- Створення дизайн-макету сервісу
- Проектування бази даних
- Реалізація front-end та back-end частини

Результати виконаної роботи будуть використані у подальшій розробці та запуску сервісу «FarmInTime»

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. URL: <https://zakon.rada.gov.ua/laws/show/80731-10#Text> (дата останнього звернення: 10.11.2021).
2. Про забезпечення функціонування української мови як державної: Закон України від 25 квітня 2019 Стаття 27 URL : https://zakon-rada.gov.ua.translate.google.com/laws/show/2704-19?x_tr_sl=ru&x_tr_tl=uk&x_tr_hl=uk&x_tr_pto=nui#Text (дата останнього звернення: 10.11.2021).
3. Ещё раз про семь основных методологий разработки [он-лайн ресурс]. URL: <https://habr.com/ru/company/edison/blog/269789/> (дата останнього звернення: 04.11.2021).
4. Использование диаграммы вариантов использования UML при проектировании программного обеспечения [он-лайн ресурс]. URL: <https://habr.com/ru/post/566218/> (дата останнього звернення: 06.11.2021).
5. Теория и практика UML. Диаграмма последовательности http://it-gost.ru/articles/view_articles/94 (дата останнього звернення: 08.11.2021).
6. Фаулер М. UML. Основы, 3-е издание. – Пер. с англ. – СПб: Символ-Плюс, 2004. – 192 с., ил. ISBN 5-93286-060-X
7. Лозинин, А.И., Шубинский, И.Б. Определение требований к программному обеспечению. [Электронный ресурс]: URL: <http://www.ibtrans.ru/Requirements.pdf> (дата обращения: 16.10.2013).
8. Вигерс, К.И. Разработка требований к программному обеспечению. - М.: ИТД «Русская редакция», 2004.
9. Пчелінцева Т. Проектування сервісу «FarmInTime» Pchelintseva_fknfm_2021_check (Херсон, 2021)

10. [он-лайн ресурс]. URL: <https://uristy.ua/laws/show/80731-10>
11. Figma [он-лайн ресурс]. URL: <https://uk.wikipedia.org/wiki/Figma>
(дата останнього звернення: 06.04.2022).
12. Що таке Figma: функції, інструменти та переваги [он-лайн ресурс].
URL: <https://wezom.academy/ua/chto-takoe-figma-funktsii-instrumenty-ipreimuschestva>
13. Розробка з боку Front end – що це таке і чим відрізняється від Back end? [он-лайн ресурс]. URL: <https://dan-it.com.ua/blog/razrabotka-so-storony-front-end-chto-jeto-takoe-i-chem-otlichaetsja-ot-back-end/>
14. Научные статьи [он-лайн ресурс]. URL:
<https://bank.nauchniestati.ru/primery/kurov%C3%A0-rabota-na-temu-razrabotka-obuchayushhego-sajta-po-izucheniyu-elementov-shkolnoj-matematiki-imwp/>
15. CSS [он-лайн ресурс]. URL: <https://calvarybaptistshsv.org/wiki/CSS>
16. Закон України Стаття 160. Торгівля з рук у невстановлених місцях [он-лайн ресурс]. URL: <https://nadoest.com/povnovajennya-vikonavchih-komitativ-silesekih-rad-ta-administr>
17. Unified Modeling Language [он-лайн ресурс]. URL:
https://uk.wikipedia.org/wiki/Unified_Modeling_Language
18. UML-диаграммы классов [он-лайн ресурс]. URL: <https://prog-cpp.ru/uml-classes/>
19. Диаграмма вариантов использования [он-лайн ресурс]. URL:
<https://sites.google.com/site/metodsybd/blok-4-sredstva-proektirovania/4-1-diagramma-variantov-ispolzovania>
20. Диаграмма последовательности (Sequence diagram) [он-лайн ресурс]. URL: https://flexberry.github.io/ru/fd_sequence-diagram.html