**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ ЮРІЯ КОНДРАТЮКА**

**Навчально-науковий інститут інформаційних технологій та механотроніки**

**Кафедра комп’ютерних та інформаційних технологій і систем**

**РОЗРАХУНКОВО-ГРАФІЧНА РОБОТА**

з навчальної дисципліни:

**«Технологія створення програмних продуктів»**

Варіант №21

|  |
| --- |
| **Виконав:**  *студент групи 301 - ТН*  *Чоха Тарас Олександрович*  **Перевірила:**  *Фурсова Наталія Артемівна* |

**Полтава**

**2020**

Міністерство освіти і науки України

Національний університет «Полтавська політехніка імені Юрія Кондратюка»

Кафедра комп'ютерних та інформаційних технологій і систем

ЗАВДАННЯ

до розрахунково-графічної роботи з дисципліни «Технологія створення програмних продуктів»

Група: 301 - ТН

Студент: Чоха Тарас Олександрович

**Тема:** інформаційна система «Меблевий магазин».

Завдання:

Розроблення специфікації вимог відповідно до теми роботи.

Проектування ІС на основі UML діаграм з описом.

Структура пояснювальної записки:

* Титульна сторінка.
* Завдання.
* Зміст.
* Вступ (1-2)
* Основна частина.
* Висновки.
* Список використаних джерел.
* Додатки (при потребі).

Завдання видала: к.е.н., доц. Фурсова Н.А. Дата здачі РГР: 1.06.2020 р.

**ЗМІСТ**

[**ВСТУП 5**](#_Toc41868976)

[**РОЗДІЛ 1. Розробка специфікації вимог до програмного продукту (Software Requirements Specification, SRS) для інформаційної системи «Магазин меблів». 8**](#_Toc41868977)

[**1.1 Вступ 8**](#_Toc41868978)

[**1.2 Загальний опис 8**](#_Toc41868979)

[**1.3 Вимоги зовнішніх інтерфейсів 12**](#_Toc41868980)

[**1.4 Інші нефункціональні вимоги 13**](#_Toc41868981)

[**Інші вимоги 14**](#_Toc41868982)

[**РОЗДІЛ 2. Розробка діаграм для ІС «Меблевий магазин». 15**](#_Toc41868983)

[**2.1 Діаграма прецедентів 16**](#_Toc41868984)

[**2.2 Діаграма класів 18**](#_Toc41868985)

[**2.3 Діаграма послідовностей 21**](#_Toc41868986)

[**2.4 Діаграма кооперації 23**](#_Toc41868987)

[**2.5 Діаграма станів 24**](#_Toc41868988)

[**2.6 Діаграма діяльності 26**](#_Toc41868989)

[**2.7 Діаграма компонентів 27**](#_Toc41868990)

[**2.8 Діаграма розгортання 28**](#_Toc41868991)

[**ВИСНОВОК 29**](#_Toc41868992)

[**СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 30**](#_Toc41868993)

**ВСТУП**

Створення сучасних електронних обчислювальних машин дозволило автоматизувати обробку даних у багатьох сферах людської діяльності. Без сучасних систем обробки даних важко уявити сьогодні передові виробничі технології, управління економікою на всіх її рівнях, наукові дослідження, освіта, видавнича справа, функціонування засобів масової інформації, проведення великих спортивних змагань. Значно розширило сферу застосування систем обробки даних поява персональних комп'ютерів.

Одним з найбільш поширених класів систем обробки даних є інформаційні системи. Хоча на рівні здорового глузду призначення таких систем зрозуміло кожному, для серйозного обговорення технологій сучасних інформаційних систем необхідно більш чітко визначити, в чому полягають їхні специфічні особливості, чим вони відрізняються від інших систем обробки даних, які функції вони можуть виконувати, якими ресурсами вона володіє.

Будь-який розумний вид діяльності ґрунтується на інформації про властивості стану і поведінки тієї частини реального світу, з якою пов'язана ця діяльність. Для отримання такої інформації в багатьох випадках необхідно регулярно через деякі інтервали часу проводити натурні вимірювання (або спостереження), що дозволяють визначати характеристики стану сутностей реального світу і процесів, що протікають, відповідні моментів часу, коли ці виміри проводяться.

В інших випадках вдається скористатися «матеріалізованою» інформацією, що міститься у різного роду паперових документах, звітах або публікаціях, які також виступають як частина реальності. Необхідну інформацію можна отримати з них шляхом, свого роду, «спостереженням».

Однак деякі натурні вимірювання або спостереження можуть виявитися нездійсненними у відведений для них час у зв'язку з великою трудомісткістю, високою вартістю, недоступністю об'єкта вимірювання (спостереження) і з інших причин.

Значно скоротити обсяг необхідних натурних вимірювань дозволяє комп'ютерне моделювання реальності. Якщо комп'ютерна модель адекватно (щодо інформаційних потреб користувачів) відображає стан і динаміку реальності, то багато необхідні дані можна отримувати за допомогою такої моделі, уникаючи тим самим натурних вимірювань, з істотно меншими витратами часу, а можливо, і при більш низькій вартості. Саме для підтримки таких моделей служить спеціальний клас систем обробки даних – автоматизовані інформаційні системи (АІС).

Автоматизованої інформаційної системою називається комплекс, що включає обчислювальне і комунікаційне обладнання, програмне забезпечення, лінгвістичні засоби та інформаційні ресурси, а також системний персонал і забезпечує підтримку динамічної інформаційної моделі деякої частини реального світу для задоволення інформаційних потреб користувачів.

Частина реального світу, яка моделюється інформаційною системою, називається її предметною областю.

Під динамічною моделлю тут розуміється змінність моделі у часі. Це «жива», діюча модель, в якій відображаються зміни, що відбуваються в предметній області. Така систем; повинна володіти пам'яттю, що дозволяє їй зберігати не тільки відомості про поточний стан предметної області, але і в деяких випадках передісторію.

Оскільки модель предметної області, підтримувана інформаційною системою, матеріалізується у формі організованих необхідним чином інформаційних ресурсів, вона називається інформаційною моделлю.

АІС не завжди функціонує самостійно. Вона може входити як компонента (підсистеми) в більш складну систему, таку, наприклад, як система управління торговою компанією, САПР або система управління виробництвом.

Інформаційні системи вже багато десятків і навіть сотні років існують і використовуються на практиці у формі різного роду картотек та / або колекцій паперових документів. Однак у таких системах відсутня будь-яка автоматизація обробки даних. Вони дозволяють лише реєструвати і підтримувати в систематизованій формі на паперових носіях результати проведених натурних вимірювань.

Побудова АІС є початковим етапом процесу розробки систем управління баз даних (СКБД). СУБД, у свою чергу, є сукупністю, призначеної для обробки на ЕОМ пойменованих даних, яка служить для запитів багатьох користувачів в рамках організації (предметної області).

**РОЗДІЛ 1. Розробка специфікації вимог до програмного продукту (Software Requirements Specification, SRS) для інформаційної системи «Магазин меблів».**

**1.1 Вступ**

***Призначення, мета***

Продукт – інтернет-магазин меблів який може працювати як самостійна система з можливістю інтеграції в неї інших підсистем для покращення надання послуг. Даний інтернет-магазин спеціалізується на представленні товарів та послуг для клієнта окремої сфери – меблів.

**1.2 Загальний опис**

***Перспективи продукту***

В умовах активного розвитку сфери діяльності меблевих магазинів, інтернет магазин цього продукту показує стрімкі темпи розвитку на ринку. Інтернет магазин є самостійним продуктом з можливістю інтегрувати в нього такі сервіси як онлайн-консультація та технічна підтримка. Також наявність таблиці меблів та їх комплектуючих. Інтеграція з магазинами, яка дозволить переглядати вартість обраних елементів, обраховувати загальну ціну всіх складових та робити замовлення просто на сайті.

***Характеристики продукту***

* забезпечення стійкої роботи сервісу;
* підтримка потужної бази даних;
* автоматизація завдань фінансово-господарської діяльності;
* зручний інтерфейс, модульність;
* здатність до оновлення системи;
* наявність служби підтримки;
* інтеграція з зовнішніми програмами;
* велика бібліотека меблів та комплектуючих з детальним описом характеристик товарів;
* ціни з урахуванням доставки;
* зручний пошук в каталозі та обрання товару;
* величезна бібліотека реальної меблів і оздоблювальних
* функція 3D огляду меблів;
* можливість додавати декілька товарів у кошик;
* наявність гарантій та повернення товарів;
* система акцій та відгуків.

***Класи користувачів та їх характеристики***

**Клієнти магазину:**

* обрання товару;
* замовлення (покупка) товару;
* відгуки про якість товару;
* віртуальний огляд меблів;
* розрахунок вартості меблів та матеріалів.

**Адміністрація:**

* робота з клієнтами;
* контроль якості оброблених даних;
* контроль витрат підприємства;
* створення звітності;
* оновлення інформації та дизайну сайту (управління сайтом);
* технічна підтримка покупців;
* дотримання конфіденційності особистої інформації покупців та заходів безпеки платіжних систем.

***Середовище функціонування***

Для користування інтернет сайтом магазину меблів використовують персональні комп’ютери, смартфони або інші ґаджети з доступом до інтернету. Доступ здійснюється за допомогою браузера. Більшість браузерів є кросплатформенними, але деякі функціонують лише на окремих платформах.

***Обмеження проектування і реалізації***

Онлайн-магазин повинен бути не вибагливим до швидкості інтернету та працювати без довгих затримок, захищати платіжну інформацію користувачів, надавати гарантій товару, мати базу даних користувачів, транзакцій та товарів. Має право існувати і можливість зміни мови. Сервіс має взаємодіяти з банківськими установами, податковою службою та іншими установами, що передбачені в роботі магазину меблів. Має бути добре розвинена система програмної допомоги, що дозволяє отримати «підказку» щодо будь-якої функції або дії в програмі. Зручні засоби для забезпечення діалогу користувача з програмним продуктом.

***Документація користувача***

У першу чергу, покупець продукту має вивчити пункти договору та ознайомитися з усіма умовами користування, після цього усі користувачі матимуть право безпечно користуватися продуктом. Процес реєстрації повинен супроводжуватись підказками, а зареєстрований клієнт отримає пакет інструкцій з користування програмним продуктом. На сайті присутнє керівництво по використанню, опис всіх функцій та FAQ з найбільш поширеними питаннями, що часто виникають від користувачів. Під час реєстрації користувач повинен дати згоду на використання персональних даних та погодитись з умовами конфіденційності.

***Припущення та залежності***

Товари які є на сайті не завжди є в наявності. Магазини, з якими співпрацюватиме сервіс, можуть бути не доступні в країні користувача. Ціни повинні відображатися залежно від регіону користувача та у відповідній валюті.

***Характеристики системи***

1. **Реєстрація**для можливості купівлі товарів та залишення відгуків.
   1. Пріоритет – високий.
   2. Обрати створення нового облікового запису, ввести реєстраційні данні (логін, пароль, e-mail адресу, ім'я користувача).

Req-1: перевірка валідності e-mail адреси.

Req-2: перевірка виконання усіх умов створення паролю.

1. **Пошук меблів**за допомогою встановлення різноманітних фільтрів та типу сортування меблів в каталозі*.*
   1. Пріоритет – високий.
   2. Відкрити меню фільтрів, обрати необхідні параметри.

Req-3: розділення меблів по категорії.

1. **Можливість залишити оцінку**придбаноготовару на сторінці товару.
   1. Пріоритет – середній.
   2. Зайти на сторінку товару, залишити відгук в спеціальній для цього секції.

Req-4: перевірка тексту на обмеження.

Req-5: зберігання відгуків в базі даних та відтворення їх у спеціальній секції.

1. **Додавання товару в кошик** для розрахунку декількох товарів одночасно.
   1. Пріоритет – високий.
   2. Зайти на сторінку огляду товарів, натиснути кнопку “До сплати” – товар додається в список кошика, а його ціна додається до суми інших товарів у кошику.

Req-6: зберігання товару в кошику до завершення покупок.

Req-7: підрахунок в кошику всіх цін товарів в одну.

1. **Покупка товару** – вибір способу оплати та доставки обраного товару та підтвердження операції.
   1. Пріоритет – високий.
   2. У кошику натиснути на кнопку “Оформлення замовлення” та обрати спосіб оплати та доставки.

Req-8: Перевірка на заповнення всіх полів.

Req-9: Повідомлення на підтвердження операції (покупки).

Req-10: Обробка платежу.

**1.3 Вимоги зовнішніх інтерфейсів**

***Користувацькі інтерфейси***

* Каталог повинен займати не менше 50% екрану;
* Інтерфейс повинен бути інтуїтивно зрозумілим новим користувачам;
* Всі матеріали та меблі повинні бути посортовані за видами;
* Головні кнопки, такі як зміна режиму перегляду, підрахунок ціни тощо повинні були легко помітними на екрані;
* При виборі матеріалів/меблів повинна відображатись його зображення та написана ціна;
* Також повинен бути розроблений окремий інтерфейс для мобільної версії.

***Апаратні інтерфейси***

Продукт може працювати на пристроях (якщо це версія додатку, яка встановлена на пристрій), що підтримуються операційними системами Windows (старше версії 98), MacOS, iOS(старше версії 6.0), Android(старше версії 4.0), Linux та ін.

***Програмні інтерфейси***

Програмний продукт повинен працювати в онлайн режимі через браузер будь-якого походження або як онлайн-додаток на пристроях, що підтримуються Windows, MacOS, iOS, Android, Linux. До системи підключається база даних яка містить меблі, матеріали, кольори, ціни тощо.

***Комунікаційні інтерфейси***

Програмний продукт магазину меблів працює через додаток на пристрої або через веб-браузер. Використовує електронну пошту (e-mail) для реєстрації та надсилання додаткової інформації про продукт. Використовуються протоколи IPv4 та комунікаційний протокол HTTP. Швидкість передачі даних буде варіюватись та буде залежати від швидкості інтернет з’єднання, зокрема, є припущення, що вона буде досягати 1 Тбіт/с.

**1.4 Інші нефункціональні вимоги**

***Вимоги надійності***

Під час користування продуктом можуть виникнути проблеми з доступом до облікового запису, його можна буде відновити за допомогою зареєстрованої раніше електронної пошти. Для веб версії час очікування на додавання нових об’єктів не повинен перевищувати 3 секунд, навіть при не досить швидкому інтернеті. Продуктивність повинна бути в межах зручної кількості кадрів для нормального перегляду та зміни інтер’єру.

***Вимоги безпеки***

Для аутентифікації повинен використовуватись логін і пароль. Оскільки товар можна оплачувати відразу банківською карткою необхідні заходи безпеки для захисту платіжної інформації користувачів. Безпека даного програмного продукту зумовлена використанням сервісу «GoDaddy», що використовує безпечне шифрування та передбачає надійний захист програмного продукту.

***Атрибути якості програмного продукту***

* придатність – магазин інтуїтивно зрозумілий для всіх категорій людей;
* надійність – повний захист платіжної інформації покупців та надання гарантій з можливістю повернення товарів;
* зручність використання – зручний та простий інтерфейс, відсортовані елементи декору зроблять користування швидким та приємним;
* багатофункціональність – безліч матеріалів та меблів, велика кількість функцій та можливостей зможе втілити всі мрії користувачів та задовольнити їх потреби;
* гнучкість – можливість пристосувати програму до особливостей конкретного підприємства та змін у законодавстві;
* портативність – здатність сайту надійно та швидко працювати, використовуючи будь-який пристрій, навіть при слабкому інтернет з’єднанні;
* стійкість – можливість протидії вірусним атакам та, в окремих випадках, можливість надшвидкого відновлення.

**Інші вимоги**

Бази даних з матеріалами, меблями та іншими елементами декору повинна доповнюватись новими продуктами відповідно до розвитку індустрії дизайну інтер’єрів. Застарілі меблі, які вже не виробляються повинні іноді видалятися, щоб не вводити користувачів в оману.

Ціни в базах повинні регулярно оновлюватись відповідно до цін ринку.

**РОЗДІЛ 2. Розробка діаграм для ІС «Меблевий магазин».**

Для побудови більшості діаграм було використано інструмент для створення блок-схем і діаграм «Draw.io».

**Draw.io** - безкоштовний додаток, призначений для моделювання діаграм і блок-схем та бізнес-процесів. Присутня можливість інтеграції з Google Документами, Dropbox, OneDrive, JIRA, Confluence, Chrome і GitHub. Доповнення для серверів Confluence і JIRA Cloud є платними. Онлайн-версія для синхронізації з Google Диском пропонується в 2 варіантах - з підтримкою Google Apps і без неї. Крім веб-версії існує програма для установки на ПК, яка підтримує ОС Windows, MacOS і Linux.

Додаток буде корисним для менеджерів і керівників компаній, яким потрібен інструмент для побудови схем бізнес-процесів. Сервіс підходить для фахівців, чия діяльність пов'язана зі створенням презентацій та баз даних, побудовою інженерних і мережевих схем, проектуванням програмного забезпечення.

**2.1 Діаграма прецедентів**

Діаграма прецедентів – це UML діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма Прецедентів візуально зображає різноманітні сценарії взаємодії між акторами (користувачами) і прецедентами (випадками використання); описує функціональні аспекти системи. Спочатку створимо таблицю, на якій розмістимо назву актора та його короткий опис (таблиця 1).

*Таблиця 1.*

|  |  |
| --- | --- |
| Актор | Короткий опис |
| Адміністратор | Особи, які здійснюють управління сайтом та інші дії, пов'язані з його використанням. |
| Потенційний покупець | Звичайний користувач, який переглядає інформацію системи. |
| Покупець | Клієнт, який використовує систему для покупок товарів. |

Функції акторів системи:

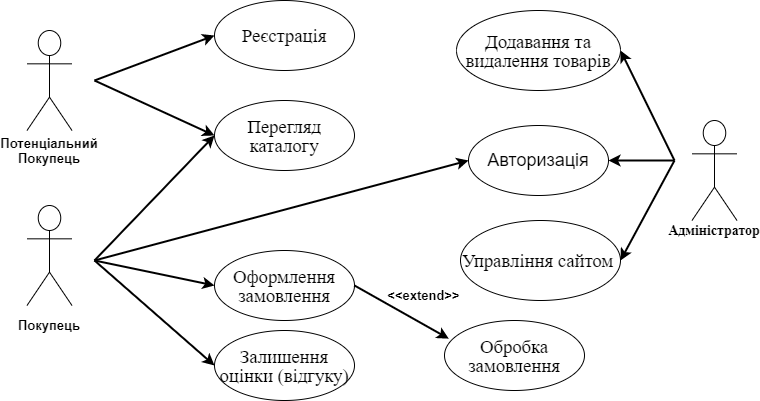
* ***актор «Адміністратор»*** використовує систему для управління сайтом, визначає склад сервісів сайту, його структуру і зовнішній вигляд, оновлює базу даних меблів та інших складових частині інтернет-магазину, обробляє покупки користувачів;
* ***актор* *«Потенційний покупець»*** використовує систему тільки для перегляду інформації або каталогу товарів.
* ***актор «Покупець»*** використовує систему для здійснення покупок товарів та залишення відгуків про придбані товари.

У таблиці 2 найкоротшим чином описано сценарії та їх виконання.

*Таблиця 2.*

|  |  |
| --- | --- |
| **Сценарій** | **Короткий опис** |
| Реєстрація | Виконується потенційним покупцем, дозволяє отримати доступ до особистого кабінету та функцій пов’язаних з ним. |
| Авторизація | Виконується адміністратором та покупцем, дозволяє надати відповідні обмеження використання ресурсу. |
| Перегляд каталогу | Виконується потенційним покупцем та покупцем, дозволяє в подальшому оформити заявку на покупку товару. |
| Оформлення замовлення | Виконується покупцем, дозволяє вносити, змінювати дані про покупку. |
| Залишення оцінки | Виконується покупцем, дає можливість залишити оцінку про придбаний товар на сторінці товару. |
| Управління сайтом | Виконується адміністратором, дозволяє управляти сайтом, визначати склад сервісів сайту, його структуру і зовнішній вигляд. |
| Додавання та видалення товарів | Виконується адміністратором, дозволяє додавати або видаляти товари з бази даних. |

Заключним кроком цього етапу буде побудова, власне, діаграми прецедентів для ІС «Магазин меблів» (рис. 1).

  
*Рис. 1 – Діаграма прецедентів.*

Побудовано діаграму прецедентів у відповідності до завдання.

**2.2 Діаграма класів**

Створимо Діаграму класів і назвемо її «Add New Order». У полі документації запишемо для неї наступний текст: Діаграма класів для сценарію «Додати нове замовлення» прецеденту «Робота із замовленням». Заповнення діаграми розпочинаємо з визначення Класів та їх властивостей.

Даний сценарій складається з:

• замовлення;

• клієнта, який робить замовлення;

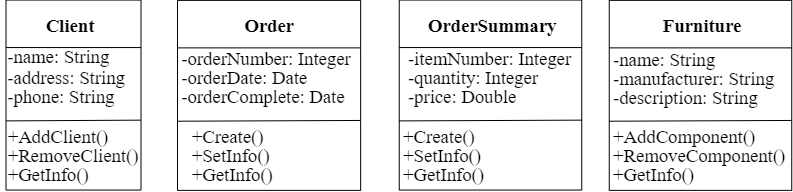
• одиниці меблів (товарів), які входять в замовлення.

Створимо Класи: «Order» (Замовлення), «Client» (Клієнт) і «Furniture» (Одиниця меблів). Оскільки в одне замовлення може входити багато різних меблів, і одиа й та сама одиниця може входити в багато замовлень, то введемо ще один Клас «OrderSummary» (Пункт замовлення). Створимо таблицю, у якій зможемо описати класи, що ми створимо (таблиця 3).

*Таблиця 3.*

|  |  |
| --- | --- |
| *Параметр* | *Значення* |
| *Клас Client* | |
| *Коментар* | *Клас, що є клієнтом сайту* |
| *Атрибути* | name: String – найменування клієнта  address: String – адреса клієнта  phone: String – телефон клієнта  Усі атрибути мають модифікатор доступу – private |
| *Операції* | AddClient() – додавання нового клієнта  RemoveClient() – видалення існуючого клієнта  GetInfo() – отримати інформацію про клієнта  Усі операції мають модифікатор доступу – public |
| *Клас Order* | |
| *Коментар* | *Клас, що є замовленням, яке робить клієнт* |
| *Атрибути* | orderNumber: Integer – номер замовлення  orderDate: Date – дата оформлення замовлення  orderComplete: Date – дата виконання замовлення  Усі атрибути мають модифікатор доступу – private |
| *Операції* | Create() – створення нового замовлення  SetInfo() – занести інформацію про замовлення  GetInfo() – отримати інформацію про замовлення  Усі операції мають модифікатор доступу – public |
| *Клас OrderSummary* | |
| *Коментар* | *Клас, що є пунктом замовлення, яке робить клієнт* |
| *Атрибути* | itemNumber: Integer – номер пункту замовлення  quantity: Integer – кількість комплектуючих замовлення  price: Double – ціна за одиницю  Усі атрибути мають модифікатор доступу – private |
| *Операції* | Create() – створення нового рядка замовлення  SetInfo() – занести інформацію про рядок замовлення  GetInfo() – отримати інформацію про рядок замовлення  Усі операції мають модифікатор доступу – public |
| *Клас Furniture* | |
| *Коментар* | *Клас, що є комплектуючими виробами* |
| *Атрибути* | name: String – найменування  manufacturer: String – виробник  description: String – опис  Усі атрибути мають модифікатор доступу – private |
| *Операції* | AddComponent() – додавання нового комплектуючого виробу  RemoveComponent() – видалення комплектуючого виробу  GetInfo() – отримати інформацію про комплектуючий виріб  Усі операції мають модифікатор доступу – public |

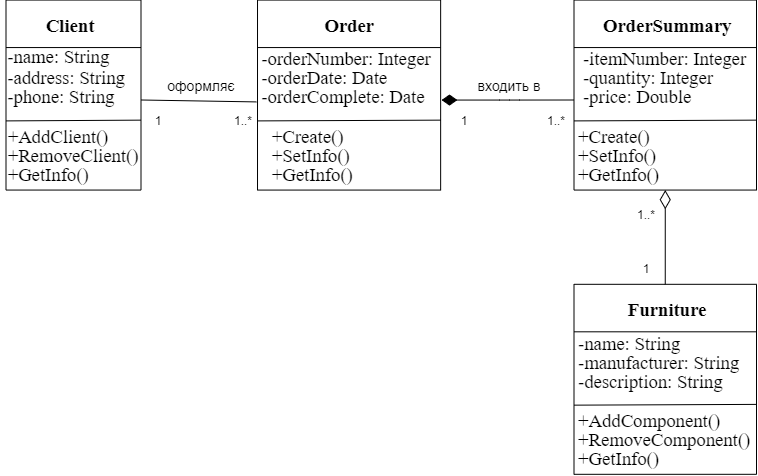
Створені Класи зображені на рис. 2.



*Рис.2 – Класи.*

Встановимо зв'язки між цими класами (рис. 3):

* між Класами “Client” і “Order” – відношення асоціації, оскільки дані Класи просто пов'язані один з одним і ніякі інші типи зв'язків тут застосувати не можна. Один клієнт може зробити декілька замовлень, кожне замовлення надходить лише від одного клієнта, тому кратність зв'язку з боку Класа “Client” – 1, а з боку “Order” – 1. \*;
* між Класами “Order” і “OrderSummary” – відношення композиції, оскільки пункт замовлення є частиною замовлення, і без нього існувати не може. В одне замовлення може входити декілька пунктів замовлень. Пункт замовлення відноситься тільки до одного замовлення, тому кратність зв’язку з боку “Order” – 1, а з боку “OrderSummary”- 1.\*;
* між Класами “OrderSummary” і “Furniture” – відношення агрегації, оскільки комплектуючі вироби є частинами пункту замовлення, але і ті, і інші, являються самостійними класами. Один комплектуючий виріб може входити до багатьох пунктів замовлень. В один пункт замовлення входить тільки один комплектуючий виріб, тому кратність зв'язку з боку “Furniture” – 1, з боку “OrderSummary” – 1. \*.

  
*Рис. 3 – Класи та зв’язки між ними.*

Побудовано діаграму класів у відповідності до поставленого завдання.

**2.3 Діаграма послідовностей**

Оберемо сценарій "Оформлення замовлення". Ініціатором взаємодії виступає *Актор “Client”* (*Клієнт*), тому на діаграмі він буде знаходиться в лівому кутку.

Далі розміщуємо об’єкти:

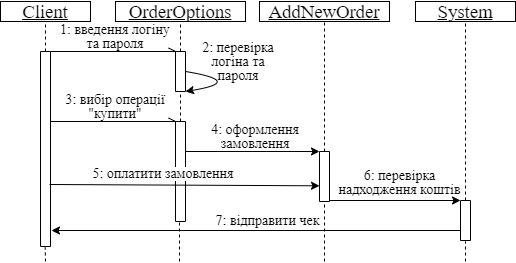
* Клас «OrderOptions» (Параметри роботи із замовленням), що відповідає за вибір можливої дії із замовленням;
* Клас «AddNewOrder» (Додавання нового замовлення), що відповідає за додавання замовлення;
* Клас «System» (Система), що відповідає за перевірку надходження коштів.

Створимо таблицю 4 та внесемо туди дані, що описують об’єкти, що обмінюються повідомленнями.

*Таблиця 4.*

|  |  |  |  |
| --- | --- | --- | --- |
| № Повідомлення | Об'єкт відправник повідомлення | Об'єкт одержувач повідомлення | Назва |
| 1 | Client | OrderOptions | введення логіну та пароля |
| 2 | OrderOptions | OrderOptions | перевірка логіна та пароля |
| 3 | Client | OrderOptions | вибір операції "купити" |
| 4 | OrderOptions | AddNewOrder | оформлення замовлення (введення всіх необхідних даних – спосіб оплати, доставки та інше) |
| 5 | Client | AddNewOrder | оплата замовлення |
| 6 | AddNewOrder | System | перевірка надходження коштів |
| 7 | System | Client | відправка чеку |

Діаграма послідовностей для сценарію "Оформлення замовлення" наведена на рис. 4.



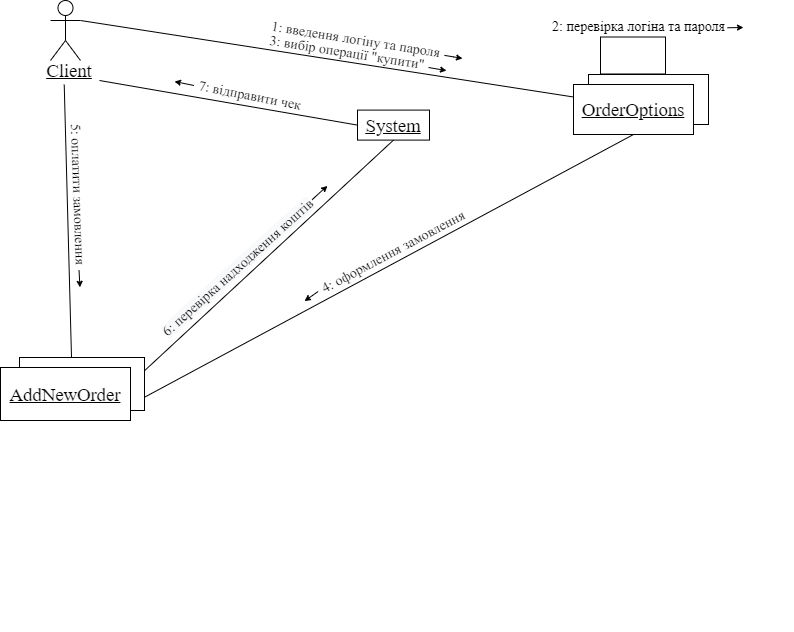
*Рис. 4 – Діаграма послідовностей.*

Побудовано діаграму послідовностей у відповідності до поставленого завдання.

**2.4 Діаграма кооперації**

Діаграма кооперації або діаграма співробітництва (сollaboratіon dіagram) описує статичну структуру об'єктів, що реалізують поведінку підсистеми. Мета кооперації полягає у тому, щоб специфікувати особливості реалізації прецедентів та найважливіших операцій у системі. [4].

Діаграма кооперації для сценарію "Оформлення замовлення" представлена на рис. 5.



*Рис. 5 – Діаграма кооперації.*

Побудовано діаграму кооперації у відповідності до поставленого завдання.

**2.5 Діаграма станів**

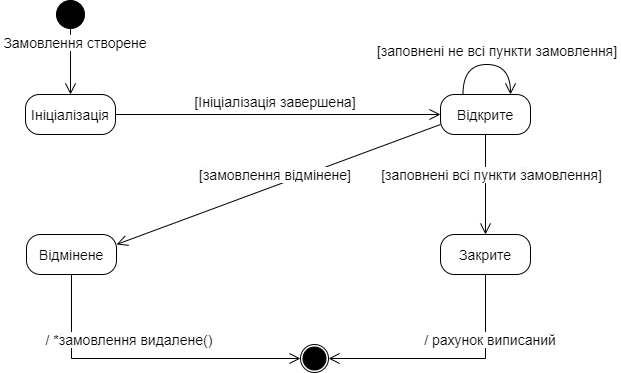
Діаграма станів є графом спеціального виду, який представляє певний автомат. Вершинами графа є можливі стани автомата, зображувані відповідними графічними символами, а дуги позначають його переходи зі стану в стан. Діаграми станів можуть бути вкладені одна в одну для більш детального представлення окремих елементів моделі.

Діаграми станів застосовуються для того, щоб пояснити, яким чином працюють складні об'єкти. Діаграма станів показує, як об'єкт переходить з одного стану в інший. Очевидно, що діаграми станів служать для моделювання динамічних аспектів системи [2].

Складемо діаграму станів для класу “Order” (Рис. 6). Замовлення може перебувати в декількох станах (таблиця 5).

*Таблиця 5.*

|  |  |
| --- | --- |
| Стан | Опис стану |
| Ініціалізація | При створенні замовлення воно переходить у стан ініціалізації, у якому замовлення підготовлюється до обробки |
| Відкритий | При переході від стану ініціалізації замовлення переходить у відкритий стан, у якому до замовлення додаються нові параметри (комплектуючі). |
| Відмінений | Якщо замовлення відмінене, то зі стану Відкрите вона переходить у відмінений стан. При виході із цього стану відбувається скидання всіх параметрів замовлення. |
| Закритий | Якщо заповнені всі необхідні параметри замовлення, після процесу підтвердження воно переходить у закритий стан, у якому відбувається виписка рахунку. Вихід із цього стану відбудеться тільки після того, як рахунок буде виписаний. |



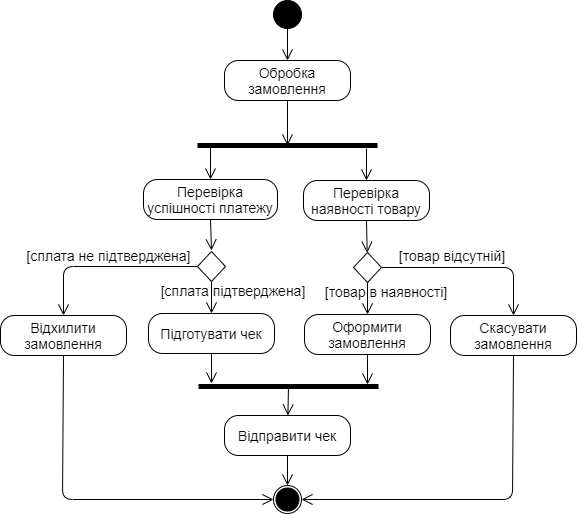
*Рис. 6 – Діаграма станів для класу Order.*

Побудовано діаграму станів у відповідності до поставленого завдання.

**2.6 Діаграма діяльності**

Діаграма діяльності (activity diagram) – в UML, візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій [2].

Побудуємо діаграму діяльності яка зображує систему оформлення замовлення користувачем на покупку товару (рис. 7).



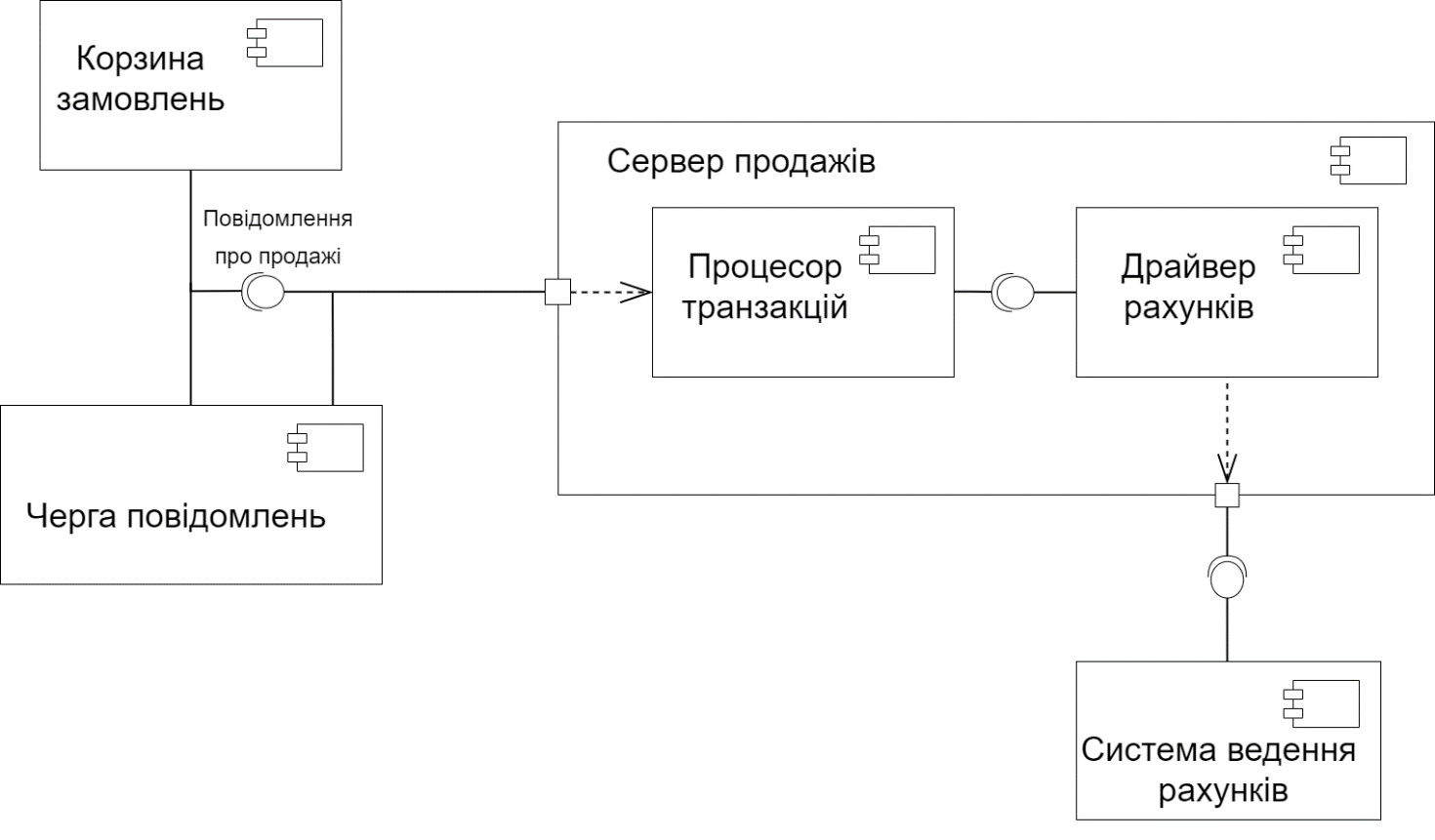
*Рис. 7 – Діаграма діяльності (обробка замовлення).*

Побудовано діаграму діяльності у відповідності до поставленого завдання.

**2.7 Діаграма компонентів**

Діаграма компонент відображає залежності між компонентами програмного забезпечення, включаючи компоненти вихідних кодів, бінарні компоненти, та компоненти, що можуть виконуватись. Модуль програмного забезпечення може бути представлено як компоненту. [5].

Компонент «Корзина замовлень» може взаємодіяти з компонентом «Сервер продажів» за допомогою інтерфейсу «Повідомлення про продажі». Оскільки мережа ненадійна, то компонент «Черга повідомлень» встановлений так, щоб «Корзина замовлень» могла спілкуватися з сервером, коли мережа працює, і спілкуватися з чергою повідомлень, коли мережа відключена. Тоді черга повідомлень зможе спілкуватися з сервером, коли мережа знову стане доступною. В результаті черга повідомлень надає інтерфейс для спілкування з «Корзина замовлень», і вимагає такий же інтерфейс для спілкування з сервером. Сервер розділений на два основні компоненти: «Процесор транзакцій» реалізує інтерфейс повідомлень, а «Драйвер рахунків» спілкується з «Система ведення рахунків». Побудуємо діаграму компонентів (рис. 8).



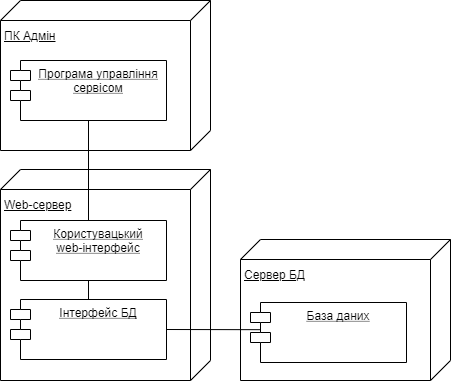
*Рис. 8 – Діаграма компонентів.*

Побудовано діаграму компонентів у відповідності до поставленого завдання.

**2.8 Діаграма розгортання**

Діаграма розгортання (deployment diagram) – діаграма в UML, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, та об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Компоненти, що не мають представлення під час роботи програми на таких діаграмах не відображаються [6].

Побудуємо діаграму розгортання (Рис. 9).



*Рис. 9 – Діаграма розгортання.*

Побудовано діаграму розгортання у відповідності до поставленого завдання. На ній представлено ПК адміністратора, який користується Web-інтерфейсом для доступу до сервера магазину.

**ВИСНОВОК**

У ході виконання розрахунково-графічної роботи з навчальної дисципліни «Технологія створення програмних продуктів» була проаналізована та розроблена специфікація вимог до програмного продукту для інтернет-магазину меблів.

Було розроблено ряд діаграм з детальним описом для інформаційної системи «Магазин меблів» за допомогою додатка «Draw.io», а саме:

* Діаграму прецедентів в якій зображені сценарії виконання та актори з зв'язками між ними, також були створені таблиці опису акторів та сценаріїв виконання;
* Діаграму класів для сценарію «Додати нове замовлення» прецеденту «Робота із замовленням» зобразивши на ній класи та зв’язки між ними, кожному класу були задані атрибути і операції;
* Діаграму послідовностей для сценарію «Оформлення замовлення» з описом об’єктів та повідомлень;
* Діаграму кооперації для сценарію "Оформлення замовлення";
* Діаграму станів для класу “Order” з описом станів;
* Діаграми діяльності, котрі зображують систему входу користувачем в аккаунт та систему оформлення замовлення користувачем на покупку товару;
* Діаграму компонентів, яка зображує систему черги замовлень;
* Діаграму розгортання, котра зображує процес взаємодії адміністратора з системою шляхом використання програми управління сервісом.

**СПИСОК** **ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Елементи UML. [Електронний ресурс]/docs.kde.org. – Режим доступу: <https://docs.kde.org/stable5/uk/kdesdk/umbrello/uml-elements.html>.
2. Навчальний посібник «Теорія систем і системний аналіз», С.М. Павлов, [Томськ](https://ua-referat.com/%D0%A2%D0%BE%D0%BC%D1%81%D1%8C%D0%BA): [Томський](https://ua-referat.com/%D0%A2%D0%BE%D0%BC%D1%81%D1%8C%D0%BA) міжвузівський центр дистанційної освіти, 2003, 134 с.
3. Навчальний [методичний посібник](https://ua-referat.com/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4%D0%B8%D1%87%D0%BD%D0%B8%D0%B9_%D0%BF%D0%BE%D1%81%D1%96%D0%B1%D0%BD%D0%B8%D0%BA) «Теорія систем і системний аналіз», [Томськ](https://ua-referat.com/%D0%A2%D0%BE%D0%BC%D1%81%D1%8C%D0%BA): Томський міжвузівський центр дистанційної освіти, 2003, 34 с.
4. Навчальний посібник «Моделі та проектування баз даних», В.Д. Сибільов, Томськ, 2002.
5. Інтернет-університет інформаційних технологій, курс «Проектування інформаційних систем», http://www.intuit.ru/department/se/devis/1/.  
   «Бізнес-процеси, основні [стандарти](https://ua-referat.com/%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82) їх описи», С.М. Ковальов, журнал «Довідник економіста» № 11'2006.
6. «Особливості автоматизації конструкторського і технологічного проектування в меблевому виробництві», Павло Бунаков, журнал «САПР і графіка» № 7'2007.
7. «Автоматизація підприємства. З чого почати? », Ілля Коломін, журнал« Фабрика меблів », № № 1-4 '2006.