

МІНІСТЕРСТВО ОСВІТИ І НАУКРИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра ІСМ

КУРСОВА РОБОТА

З дисципліни «Проектування інформаційних систем»

НА ТЕМУ:

«Інформаційна система керування автодилерською мережею»

Виконав студент групи СА-33

Шугета О.В

Оцінка	Балів	Дата

Номер залікової книжки

21169329

Керівник проєкту

Висоцька Вікторія Анатоліївна

Львів 2023

Завдання та календарний план виконання

<i>Завдання до виконання</i>	<i>Дата</i>
Анотація	02.10
Вступ	03.10
Розділ 1. Аналіз літературних джерел та предметної галузі	06.10
1.1. Основні засади дослідження	06.10
1.2. Аналіз відомих засобів вирішення проблеми	06.10
1.3. Функціональність програмної системи	07.10
Висновок до першого розділу	07.10
Розділ 2. Вибір методології проектування та планування	09.10
2.1. Опис системних вимог згідно з методологією RUP	10.10
2.2. Деталізація функцій та аналіз категорій користувачів	11.10
2.2.1. <i>Діаграма варіантів використання</i>	12.10
2.2.2. <i>Діаграма класів</i>	13.10
2.2.3. <i>Діаграми кооперації</i>	14.10
2.2.4. <i>Діаграма послідовності</i>	15.10
2.2.5. <i>Діаграма діяльності</i>	16.10
2.2.6. <i>Діаграма компонентів</i>	17.10
Висновок до другого розділу	17.10
Розділ 3. Вибір засобів реалізації та проектування системи	23.10
3.1. Вибір програмного рішення	23.10
3.2. Аналіз контрольного прикладу	27.10
Висновок до третього розділу	27.10
Висновки	01.11
Список використаних джерел	02.11
Додатки	03.11

Завдання прийнято до виконання _____

(підпис студента)

Керівник проекту

Висоцька Вікторія Анатоліївна

Зміст

Завдання та календарний план виконання	2
Анотація	4
Вступ	5
Розділ 1. Аналіз літературних джерел та предметної галузі.	7
1.1. Основні засади дослідження.....	7
1.2. Аналіз відомих засобів вирішення проблеми.	9
1.3. Функціональність програмної системи.	11
1.4. Висновок до першого розділу.....	15
Розділ 2. Проектування системи.	16
2.1. Опис системних вимог згідно з методологією RUP	16
2.2. Деталізація функцій та аналіз категорій користувачів.	19
2.2.1. Діаграма варіантів використання (use case diagram)	19
2.2.2. Діаграма класів (class diagram)	22
2.2.3. Діаграми кооперації (collaboration diagram)	29
2.2.4. Діаграма послідовності (sequence diagram)	32
2.2.5. Діаграма діяльності (activity diagram)	34
2.2.6. Діаграма компонентів (deployment diagram)	35
2.3. Висновок до другого розділу.....	37
Розділ 3. Вибір засобів реалізації та конструювання системи.	37
3.1. Вибір програмного рішення.....	37
3.2. Аналіз контрольного прикладу.....	39
3.3. Висновок до третього розділу.	45
Висновки	46
Список використаних джерел	47
Додаток А	49

Анотація

Ця курсова робота має за мету вивчення інформаційних систем для автодилерських мереж, які займаються продажем автомобілів та управлінням автосалонами.

Було проведено аналіз ринку та потреб користувачів на основі статистики, що вказує на зростання кількості компаній, які торгують автомобілями.

У цьому дослідженні були визначені ключові проблеми існуючих автосалонів, висвітлено шляхи їх вирішення, та розроблено систему, яка дозволить ефективно функціонувати інформаційній системі.

Завершальною метою є реалізація автоматизованої інформаційної системи з продажу автомобілів у вигляді веб-сайту.

Вступ

Актуальність теми.

Тема для курсової роботи є дуже актуальною у наш час, так як дуже багато людей мають хоча б один автомобіль. Тому такий процес, як покупка автомобіля є дуже важливим аспектом.

Мета.

Метою цієї курсової роботи є створення інформаційної системи, яка допоможе у керуванні автодилерською мережею. На виході ми отримаємо сайт, який допоможе надавати послуги у продажі автомобілів.

Задачі дослідження

- a) Аналіз та дослідження ринку послуг продажу автомобілей.
- b) Аналіз проблем у конкурентів.
- c) Відомі засоби, як можна вирішити дані проблеми.
- d) Опис функціональності програмної системи.
- e) Проектування системи згідно методології RUP.
- f) Дослідження та розробка програмного рішення.
- g) Створення та підключення бази даних для успішного функціонування системи.
- h) Тестування системи.

Об'єкт дослідження

Об'єктом дослідження є процеси керування автодилерською мережею.

Предмет дослідження

Предметом дослідження є процеси продажу та купівлі автомобілів. Система

Практичне значення.

У результаті дослідження, має бути створена система для керування автодилерською мережею, яка допоможе користувачам купляти автомобілі.

Розділ 1. Аналіз літературних джерел та предметної галузі.

1.1. Основні засади дослідження.

Інформаційна система управління автодилерською мережею – певний комплекс програмних рішень, які допомагають автоматизувати та оптимізувати різні процеси у сфері продажу автомобілів. Такими процесами можуть бути, наприклад: реєстрація клієнта, реєстрація замовлення, заповнення замовлення, передача замовлень на автомобільні заводи і так далі...

На сьогоднішній день існує дуже багато різних автосалонів, які займаються перепродажем автомобілів від фабрик. Наприклад:

1. Авто-Граф – авто-дилер у Львові.
2. Lion Motors – авто-дилер у Львові.

Суттєвим недоліком цих компаній є те, що багато хто з них не має автоматизованої системи для продажів, через це доводиться наймати окремий персонал, який буде це все робити вручну.

В результаті чого при роботі з даними компаніями є наступні проблеми:

- Неактуальність інформації: щоб дізнатись, які у автосалоні є автомобілі, потрібно сходити у даний автосалон фізично. Це може бути незручно для деяких груп користувачів.
- Час замовлення: через те, що у салоні немає автоматизованої системи, час замовлення буде явно більший, ніж при автоматизованій системі.
- Робочий час: через те, що у людей робочі години від 9 до 18, замовити можна лише у цей час автомобіль. А з сайтом це можна зробити у будь який час.

Створення сайту дозволить вирішити ці недоліки та оптимізувати процеси, а також залучити більшу кількість потенційних клієнтів, які не можуть зробити замовлення через часові обмеження.

1.2. Аналіз відомих засобів вирішення проблеми.

Розглянемо наших потенційних конкурентів. Проаналізуємо їх слабкі та сильні сторони:

1) Авто-Граф:

Детальний опис:

Авто-граф є компанією, що продає автомобілі на ринку вже більше 16-ти років. Компанія співпрацює такими компаніями як Fiat, Opel, Citroen, Peugeot, Land-Rover, Jaguar, Alfa-Romeo, Kawasaki, Bosch Service.

Переваги:

- Компанія надає послуги вже більше 16-ти років.
- Компанія співпрацює з компаніями-гігантами.

Недоліки:

- У компанії немає сайту.
- Компанія має багато співробітників, які потрібні для обслуговування клієнтів та функціонування системи.
- Через відсутність сайту – можливість зробити замовлення є лише у робочий час.
- Щоб подивитись на моделі авто потрібно бути фізично присутнім.
- Послуги: компанія дає відносно високі ціни за свої послуги.

2) Lion Motors

Детальний опис:

Lion Motors є компанією, що надає послуги на ринку вже більше 10-ти років. Компанія також співпрацює з компаніями-гігантами, що постачають їй автомобілі. Має сформоване коло клієнтів та добре себе зарекомендувала на ринку.

Переваги:

- Термін: компанія на ринку більше 10-ти років.
- Співпраця: компанія співпрацює з великими автовиробниками.
- Клієнти: через довгий час роботи у компанії добре сформована база клієнтів.

Недоліки:

- Відсутність автоматизованої системи.
- Обмеження робочого дня.
- Неможливість керувати даними.
- Погана конкурентноспроможність, через те, що компанії можуть працювати 24\7
- Відсутність гнучкості.

Підсумовуючи, можна зрозуміти, що уся проблема є тільки у тому, що потрібно зробити автоматизацію певних процесів, що суттєво дозволить нам підвищити продажі та надасть змогу конкурування з цими компаніями.

1.3. Функціональність програмної системи.

Для відображення функціональності системи зроблено таблицю функцій та надано опис.

Таблиця 1.1

№	Назва	Опис
1	Вхід на головне меню	Користувач повинен мати підключення до інтернету та не мати обмежень для доступу до сайту.
2	Вибір марки авто	Користувач матиме змогу обрати марку автомобіля, яку він бажає придбати.
3	Перегляд доступних моделей.	Користувач матиме змогу переглянути усі моделі певної марки, які йому цікаві, а також обрати комплектацію авто та ціну.
4	Кнопка: «Замовити»	Користувач буде мати змогу перейти з сторінки моделей авто на сторінку замовлення, де він зможе зробити замовлення.
5	Full Name	Користувач матиме змогу ввести своє ПІБ.
6	Email	Користувач матиме змогу ввести свій Email.
7	Phone Number	Користувач матиме змогу ввести свій номер телефону.
8	Select Brand	Користувач матиме змогу обрати бажану марку авто.
9	Select Model	Користувач матиме змогу обрати модель авто відповідно до марки.

10	Car Configuration	Користувач матиме змогу обрати комплектацію автомобіля.
11	Interior Configuration	Користувач матиме змогу обрати комплектацію салону.
12	Car Color	Користувач матиме змогу обрати колір автомобіля.
13	Delivery	Користувач матиме змогу вказати адресу доставки автомобіля.
14	Перевірка заповнення	Кожне поле буде перевірено на заповнення, щоб користувач ввів правильно дані та не забув про певне поле.
14	SubmitOrder()	Дані будуть відправлятись на сервер та в залежності від заповнення буде створюватись новий клас.
15	Index()	Сервер буде брати клас Order та додавати його у базу даних.
16	Додавання у БД	У сервері буде перевірено чи було додано дані у базу даних. В разі помилки користувача буде повернено на форму-замовлення.
17	Кнопка «Place order»	Саме вона буде активувати процес передання даних з сайту у БД. А також, вона буде відправляти користувача на сторінку-підтвердження/відхилення замовлення.
18	«Повернутись а головну»	Після успішного замовлення користувач буде направлений на сторінку-повідомлення, де буде кнопка, яка буде повертати користувача на головну сторінку.

Функції системи

Таблиця 1.2

№	Статус	Пріоритет	Складність	Ризик	Стабільність	Версія	Призначення
1	Включена	Критична	Низька	Середній	Висока	1.0	Вхід на головне меню
2	Включена	Критична	Висока	Високий	Висока	1.0	Вибір марки авто
3	Включена	Критична	Висока	Високий	Середня	1.0	Перегляд доступних моделей.
4	Включена	Важлива	Середня	Низький	Висока	1.0	Кнопка: «Замовити»
5	Включена	Критична	Середня	Низький	Низька	1.0	Full Name
6	Включена	Важлива	Середня	Низький	Низька	1.0	Email
7	Включена	Критична	Середня	Низький	Низька	1.0	Phone Number
8	Включена	Критична	Висока	Середній	Середня	1.0	Select Brand
9	Включена	Критична	Висока	Середній	Середня	1.0	Select Model
10	Включена	Критична	Середня	Низький	Середня	1.0	Car Configuration

№	Статус	Пріоритет	Складність	Ризик	Стабільність	Версія	Призначення
11	Включена	Критична	Середня	Низький	Середня	1.0	Interior Configuration
12	Включена	Критична	Середня	Низький	Низька	1.0	Car Color
13	Включена	Критична	Середня	Низький	Низька	1.0	Delivery
14	Включена	Критична	Середня	Високий	Низька	1.0	Перевірка заповнення
15	Включена	Критична	Висока	Високий	Середня	1.0	SubmitOrder()
16	Включена	Критична	Висока	Високий	Середня	1.0	Index()
17	Включена	Критична	Висока	Високий	Висока	1.0	Додавання у БД
18	Включена	Критична	Середня	Середній	Низька	1.0	Кнопка «Place order»
19	Включена	Корисний	Низька	Низький	Низька	1.0	Кнопка «Повернутись а головну»

1.4. Висновок до першого розділу.

В підсумку цього розділу було показано попит на дилерські мережі. Виявлено великий попит на дану структуру. Проведення аналізу конкурентів показало їх слабкі та сильні сторони та надало нам можливість зрозуміти, що нам потрібно робити для спромоги конкуренції з ними.

Здобуті в ході досліджень висновки показують, що є важливим вдосконалення та впровадження нових функцій у інформаційну систему. Поглиблені консультації з командою розробників та бізнес-аналітиками стали ключовим етапом у формуванні стратегії розвитку нашої компанії.

На основі цього аналізу і думок експертів були сформульовані конкретні завдання та описано функції, необхідні для подолання слабких сторін конкурентів та забезпечення конкурентних переваг. Метою нашої подальшої роботи є впровадження цих рішень для створення інформаційної системи, яка буде не лише технічно вдосконаленою, але й зручною та ефективною для кінцевих користувачів.

Актуальність цього напряму дослідження підкреслюється реальним попитом на покращення та інновації в галузі дилерських мереж, що робить наші дослідження надзвичайно важливими для подальшого розвитку та успішного функціонування на ринку

Через це, головною метою є виконання всіх завдань для створення інформаційної системи, яка буде зручна та проста у користуванні.

Розділ 2. Проектування системи.

2.1. Опис системних вимог згідно з методологією RUP

1) Зацікавлені особи та їх вимоги:

- a) Клієнт – особа, яка робить та оплачує замовлення.
- b) Менеджер – особа, що займається замовленнями, консультацією та іншими справами.
- c) Постачальник – група осіб, яка зацікавлена у продажі своїх продуктів.
- d) Доставка – група осіб, яка зацікавлена у наданні саме своїх послуг для доставки авто.
- e) Автосалони – зацікавлені у продажі автомобіля.

2) Користувач ППС тобто основний актор цього прецеденту

Клієнт, який буде вводити дані в проєктовану систему, для того, щоб отримати можливість замовити автомобіль, який йому подобається та підходить.

3) Основний успішний сценарій:

1. Клієнт заходить на сайт
2. Клієнт обирає марку авто
3. Клієнт бачить перед собою певні моделі даної марки авто.
4. Клієнт бачить все візуально та переходить до замовлення нажимаючи кнопку «Замовити»
5. Клієнт заповнює свої дані, дані про авто, та адресу доставки на потрібний йому автосалон у спеціальній формі.
6. Клієнт, заповнивши усі дані тисне кнопку «Place order» та передає дані на сервер.
7. Сервер перевіряє дані та створює клас “Order”.
8. Сервер відкриває доступ до БД та зберігає туди дані повертаючи клієнту код 200 та переадресовуючи його на сторінку-повідомлення про успішний запис.
9. Межеджер отримує сповіщення про нове замовлення та отримує інформацію про нього.
10. Менеджер формує список замовлень та передає їх на фабрику.
11. Фабрика виготовляє авто та передає його на службу доставки.
12. Служба доставки передає авто на потрібний автосалон.
13. Менеджер повідомляє клієнта про готове замовлення.
14. Автосалон продає автомобіль клієнту.

4) Розширення основного сценарію або альтернативні потоки:

1. У клієнта немає підключення до інтернету: Йому покаже Error 404 та перекине на головну сторінку.
2. Дані не всі заповнені: користувача проінформує про те, що поле не заповнене.
3. Код 500: У такому випадку БД не зберегла дані, тому користувача буде переадресовано на сторінку з замовлення та повідомлено про помилку.
4. Помилки не у автоматизованій системі: у такому випадку користувачу потрібно буде звернутись на гарячу лінію за номером: +3808715424234. Або Email: Autoclub_hotline@gmail.com

5) Спеціальні СВ:

Потрібно зробити багатомовність системи для користувачів інших країн.

6) Необхідна технологія та додатковий пристрій

ПС має бути розроблена як Web-орієнтована система.

2.2. Деталізація функцій та аналіз категорій користувачів.

2.2.1. Діаграма варіантів використання (use case diagram)

Щоб побачити загальну картину щодо функціоналу усієї системи було побудовано use case diagram:

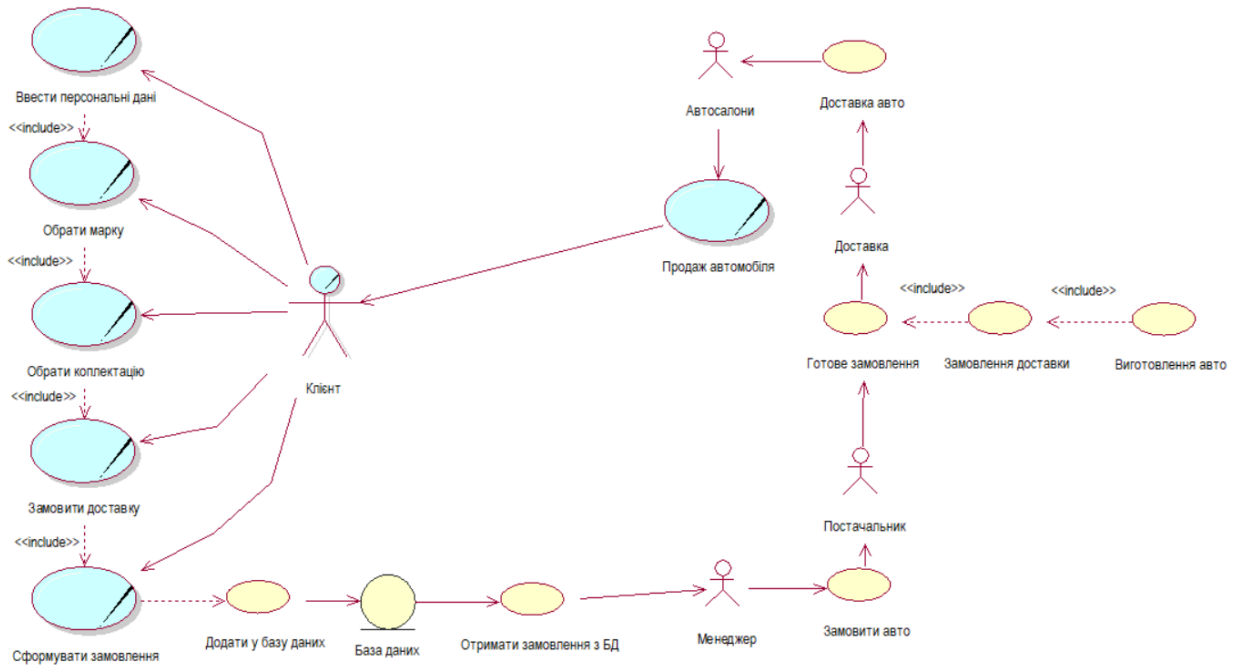


Рис.2.1 Діаграма варіантів використання

Актори:

1. Клієнт – Бізнес-актор. Особа, яка робить замовлення та оплачує його.
Навколо нього працює уся система.
2. Менеджер – особа, яка отримує замовлення, уточнює дані, передає замовлення на завод та контролює процес до продажі авто.
3. Постачальник – людина, яка працює на фабриці та отримує замовлення. Його роль отримати замовлення – виготовити його та замовити доставку в салон.
4. Доставка – ряд людей чи компаній, які отримують готове авто та доставляють його прямо у салони нашої компанії.
5. Автосалони – ряд людей, які є очільниками автосалонів. Їх мета отримати авто та продати його.
6. База даних – загальна база даних для збереження інформації про замовлення та клієнта.

Сценарій роботи:

«Клієнт» буде взаємодіяти з сайтом, вписуючи інформацію про себе, обираючи марку авто, модель, комплектацію салону та мотору. Буде вказувати, у який салон йому потрібно доставити авто для купівлі.

Саме так буде формуватись замовлення. Коли уже усі поля заповнені – формується замовлення та додається у базу даних, де воно буде зберігатись.

Після додавання замовлення у БД – у менеджера буде спрацьовувати сповіщення про нові замовлення, з якими він буде працювати. Його задачею буде пов'язати замовлення з етапами розробки, доставки та продажі.

Наступним кроком менеджер замовлятиме автомобіль у фабрики, яка буде виготовляти даний автомобіль та передавати його на служби доставки, які у свою чергу доставлятимуть авто у салони.

Майже фінальним кроком буде доставка автомобіля з точки А у точку Б, для продажу або ж зберігання у салоні.

І фінальним етапом стає продаж автомобіля автосалоном клієнту. Під час якого будуть задіяні менеджери, клієнт, автосалон.

2.2.2. Діаграма класів (class diagram)

Побудова діаграми класів дозволяє нам представити систему у ООП-парадигмі, що в свою чергу дозволить краще розуміти ситему та вимоги для побудови ПІС:

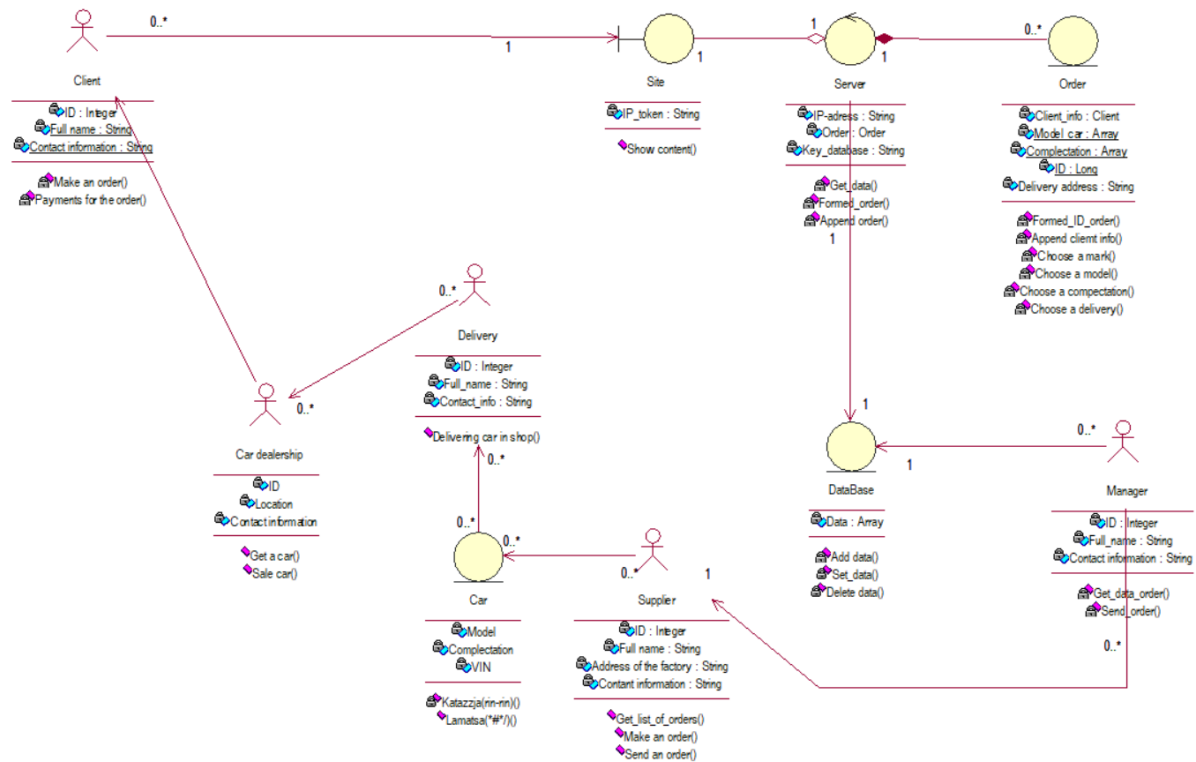


Рис.2.2 Діаграма класів

На даній діаграмі (рис.2.2) зображені основні класи (їх атрибути та методи) та взаємозв'язки між ними, для відображення роботи системи.

1. Клас-актор: Client

Даний клас відповідає та узагальнює усіх клієнтів, які будуть користуватись даною інформаційною системою.

a. Атрибути:

- i. *ID: ID клієнта*
- ii. *Full_name: ПІБ клієнта*
- iii. *Contact_information: Контактна інформація*

b. Операції:

- i. *Make an order(): Зробити замовлення*
- ii. *Payments for the order(): Оплатити замовлення*

2. Клас-інтерфейс: Site

Даний клас відповідає за весь сайт, його візуальну частину та наповнення.

a. Атрибути:

- i. *IP_token: IP-токен для запуску сайту*

b. Операції:

- i. *Show content(): Показує наповнення сайту.*

3. Клас-процес: Server

«Голова» даної ІС. Запускає core для сайту, опрацьовує усю інформацію, яка у нього поступає, формує замовлення та класи, для додавання усієї інформації у БД.

a. Атрибути:

- i. *IP-adress: Адреса даного сервера*
- ii. *Order: інформація про замовлення*

iii. *Key_database*: Доступ до БД

b. Операції:

i. *Get_data()*: Конструктор для даних

ii. *Formed_order()*: Формує замовлення

iii. *Append_order()*: Додає замовлення.

4. Клас-сутність: Order

Основне, навколо чого працює ця ІС. Клас, який отримує інформацію про замовлення та формується, як повноцінний клас для подальшої обробки БД.

а. Атрибути:

- i. *ID: ID замовлення*
- ii. *Client-info: Інформація про клієнта*
- iii. *Model_car: Модель авто*
- iv. *Complectation: Комплектація авто*
- v. *Delivery adress: адреса доставки*

б. Операції:

- i. *Formed_ID_order(): формує ID замовлення*
- ii. *Choose a mark(): Обирання марки*
- iii. *Choose a model(): Обирання моделі*
- iv. *Choose a complectation(): Обирання комплектації*
- v. *Choose a delivery(): Обирання адреси доставки*

5. Клас-сутність: DataBase

«Пам'ять» даної ІС. Зберігатиме у себе усю дану їй інформацію про клієнтів та замовлення.

а. Атрибути:

- i. *Data: Дані замовлення*

б. Операції:

- i. *Add_data(): Додавання даних*
- ii. *Set_data(): Зміна даних*
- iii. *Delete_data(): Видалення даних.*

6. Клас-актор: Manager

Даний клас узагальнює усіх менеджерів, які у нас працюють.

a. Атрибути:

- i. *ID*: ID менеджера
- ii. *Full_name*: ПІБ менеджера
- iii. *Contact_information*: контактна інформація

b. Операції:

- i. *Get_data_order()*: Отримання замовлення
- ii. *Send_order()*: Надсилання замовлення

7. Клас-актор: Supplier

«Руки нашої системи». Саме фабрики виробляють усі наші замовлення та тим самим дозволяють нам заробляти гроші.

a. Атрибути:

- i. *ID: ID контактного лиця*
- ii. *Full name: ПІБ контактного лиця*
- iii. *Address of the factory: адреса заводу*
- iv. *Contact information: Контактна інформація*

b. Операції:

- i. *Get_list_of_orders(): Отримання списку замовлень.*
- ii. *Make an order(): Виготовлення замовлення.*
- iii. *Send an order(): Надсилання замовлення.*
- iv. *Choose a mark(): Обирання марки*

8. Клас-сутність: Car

Те, заради чого ми усі тут зібрались. Основна потреба клієнта та потенційний наш прибуток.

a. Атрибути:

- i. *Mark: Марка авто*
- ii. *Model: Модель авто*
- iii. *Complectation: комплектація*
- iv. *VIN: унікальний ключ-значення кожного авто*

b. Операції:

- i. *Katazzja(rin-rin)(): кататись на автівці*
- ii. *Lamatsa(): Отримувати поломку автомобіля.*

9. Клас-актор: Delivery

«Ноги компанії». Адже саме ці люди відповідають за доставку та перевезення машин з точки А у точку В.

а. Атрибути:

- i. *ID*: Унікальне значення
- ii. *Full_name*: ПІБ контактнього лиця.
- iii. *Contact_info*: Контактна інформація

б. Операції:

- i. *Delivering car in shop()*: доставка автомобіля у магазин

10. Клас-актор: Car dealership

Основні точки, куди буде доставлено замовлення та продано його клієнту.

а. Атрибути:

- i. *ID*: Унікальне значення
- ii. *Location*: Розташування
- iii. *Contact_information*: Контактна інформація

б. Операції:

- i. *Get a car()*: Отримати авто
- ii. *Sale car()*: Продаж авто

2.2.3. Діаграми кооперації (collaboration diagram)

Побудова collaboration diagram описує логіку перебігу різних подій у даній компанії (рис. 2.3):

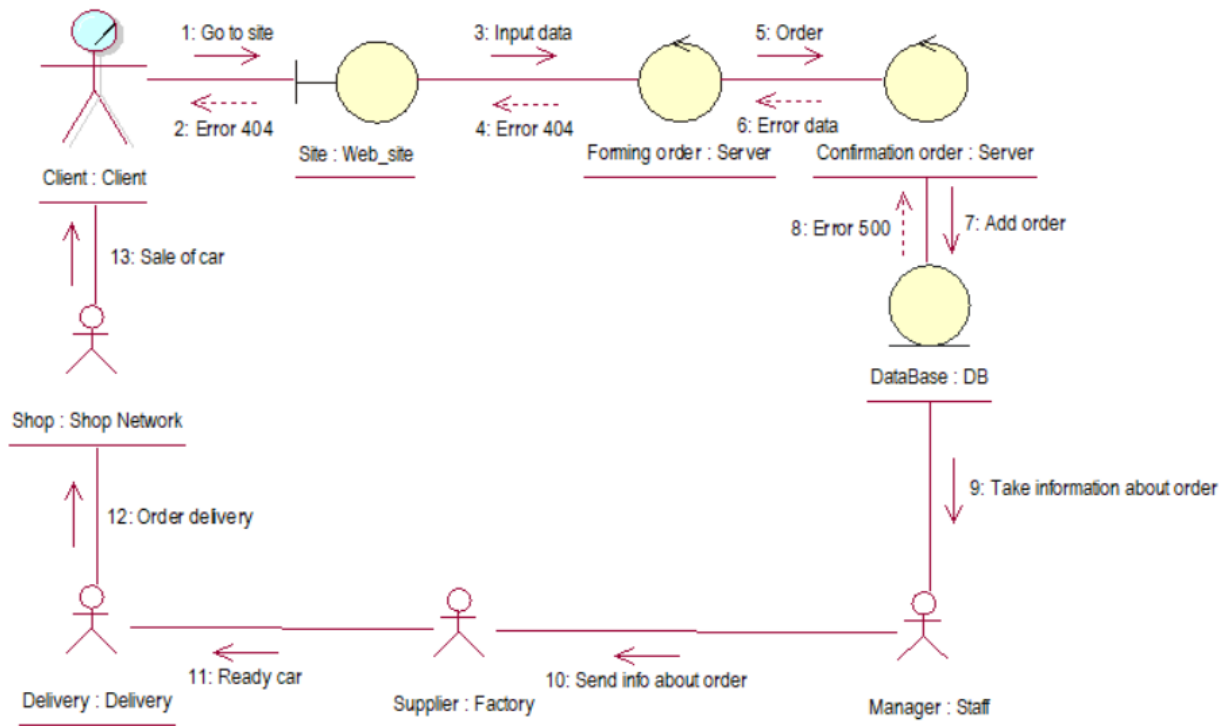


Рис.2.3 Діаграма кооперації

Перелік класів діаграми:

- 1) Назва – Client; Клас Business Actor (Client)
- 2) Назва – Web_site; Клас boundary (Site)
- 3) Назва – Forming order; Клас control (Server)
- 4) Назва – Confirmation order; Клас control (Server)
- 5) Назва – DataBase; Клас entity (DB)
- 6) Назва – Manager; Клас Actor (Staff)
- 7) Назва – Supplier; Клас Actor (Factory)
- 8) Назва – Delivery; Клас Actor (Delivery)
- 9) Назва – Shop; Клас Actor (Shop Network)

Дії та їхня послідовність:

- 1) Вхід на сайт(перевірка підключення)
- 2) Error 404 – У разі проблем з підключенням користувачу повернеться Not found.
- 3) Input data – Введення користувачом даних для створення замовлення.
- 4) Error 404 – У разі проблем з підключенням користувачу повернеться Not found.
- 5) Order – сформоване замовлення перевіряється на усі поля та додається у БД.
- 6) Error data – Неправильно введені дані. Замовлення не збережено. Користувач отримає повідомлення.
- 7) Add order – Додавання замовлення у БД
- 8) Error 500 – помилка підключення до БД. Користувача буде проінформовано.
- 9) Take information about order – Отримання інформації про замовлення менеджером.
- 10) Send info about order – Надсилання інформації про замовлення постачальнику.
- 11) Reay car – Готова машина переходить від постачальника до доставки.
- 12) Order delivery – Доставка автівки у салон.
- 13) Sale of car – Продаж машини клієнту.

2.2.4. Діаграма послідовності (sequence diagram)

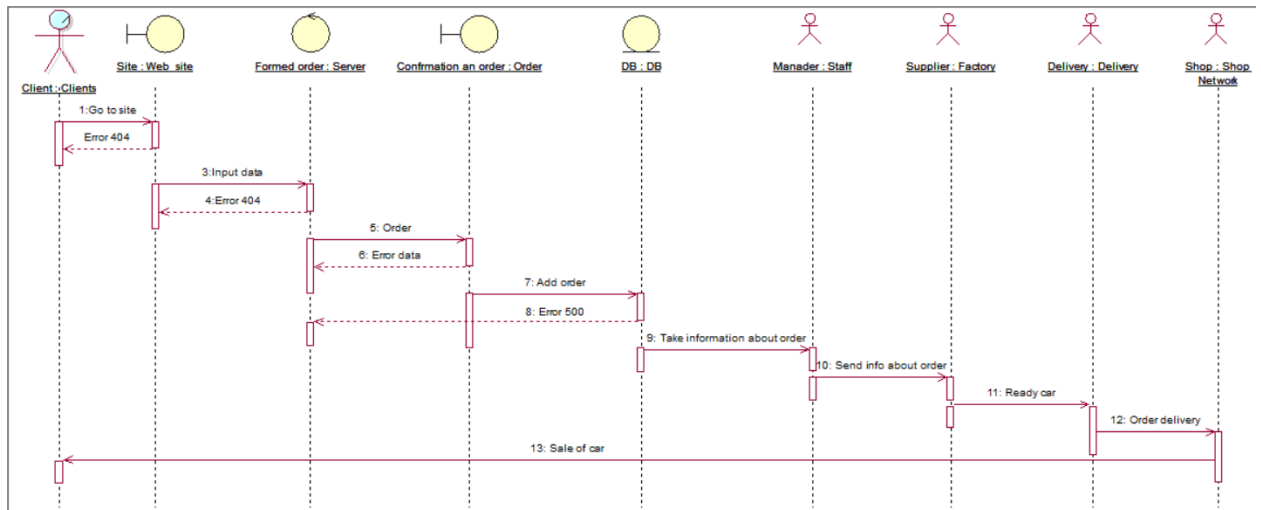


Рис. 2.4. Діаграма послідовності

На даній діаграмі показані основні об'єкти та зв'язки між ними:

Насамперед клієнт заходить на сайт. Після чого сайт або завантажується, якщо є підключення до інтернету або ж підключення немає і клієнт отримує Error 404.

Розглянемо хороший варіант, коли сайт прогрузився. Наступним іде обирання авто, моделі, комплектації(візуальна складова) та введення це у формі замовлення. Якщо сайт прогрузив форму(знову ж таки тут залежить все тільки від підключення) – буде доступно ввід даних про замовлення. Інакше – Error 404. Після цього дані, які були введені ідуть на підтвердження, де вони генерують клас і якщо щось йде не так, то клас не створюється – клієнт отримує повідомлення про помилку – робить замовлення знову.

Якщо ж все йде добре, замовлення додається у БД або ж знову спрацьовує try/except і користувач отримує повідомлення про те, що замовлення не було збережено, тому йому варто зробити повторне замовлення. Якщо ж все пройшло успішно, замовлення зберігається у БД.

Наступним етапом йде взаємодія між менеджером та замовленням. Щойно замовлення потрапить у БД – менеджер отримує сповіщення про нові замовлення, які потім пересилає на компанію-виробника, яку вказав клієнт та дає точні вказівки роботи. Виробник, у свою чергу «будує авто» та повідомляє доставку, що авто готове і його потрібно перевезти у салон за адресою, яку вказав менеджер. Після чого менеджер повідомляє клієнта про готове замовлення та клієнт купує або ж не купує дане авто. Якщо не купує – авто залишається у салоні.

2.2.5. Діаграма діяльності (activity diagram)

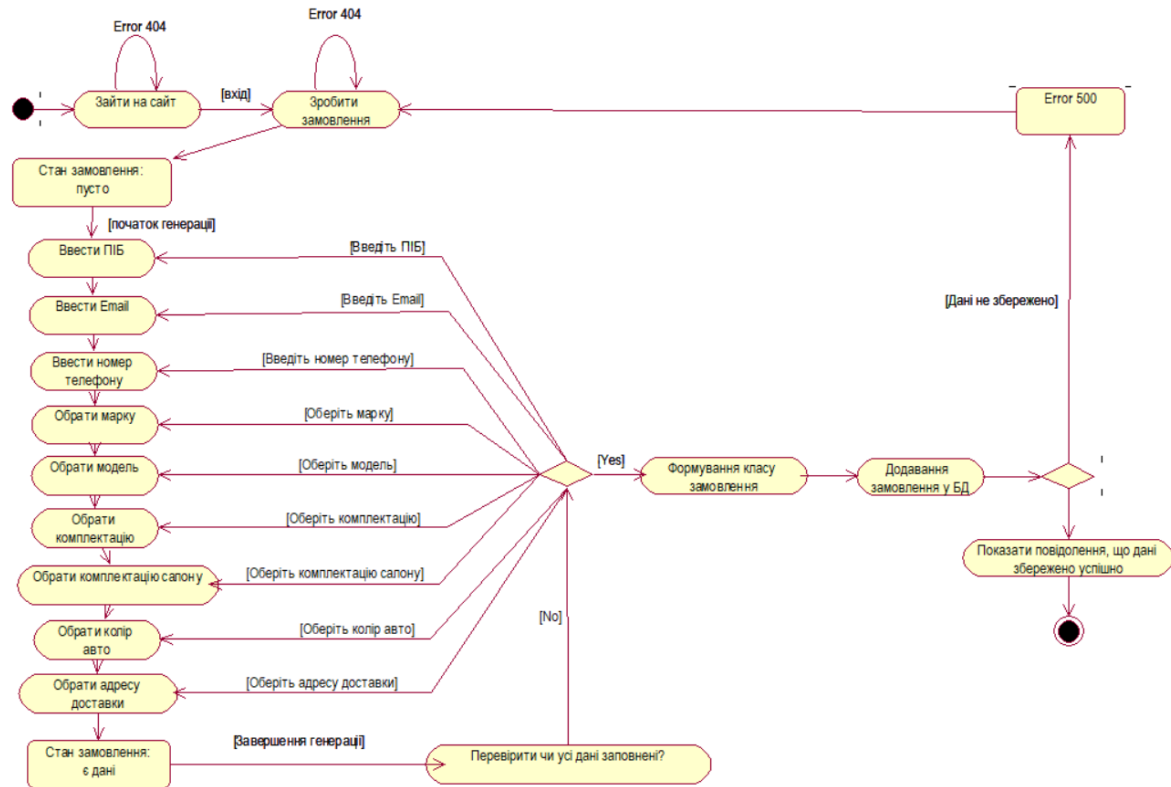


Рис.2.5 Діаграма діяльності для моделювання роботи системи

Спершу, користувач має зайти на сайт, якщо у нього немає підключення до інтернету, йому покаже Error 404 та перенаправить на цю ж сторінку. Далі користувач входить до головної форми замовлення, і тут схожий принцип з помилкою.

Після чого, якщо у користувача є з'єднання, то у нього починається процес формування замовлення, де йому потрібно вказати усі дані, які від нього вимагають. Без заповнених полів він не зомже зробити замовлення та йому буде вказано на помилку у заповненні.

Якщо ж все пройшло добре, та дані всі правильно заповнені іде процес формування класу Order та додавання його у баз даних.

Далі іде перевірка з'єднання з базою даних – якщо з'єднання не получилось встановити буде помилка 500 та перенаправлення на форму-замовлення.

Якщо ж все проходить добре, то замовлення буде успішно сформовано та записано. Користувач отримає відповідне повідомлення та процес завершиться.

2.2.6. Діаграма компонентів (deployment diagram)

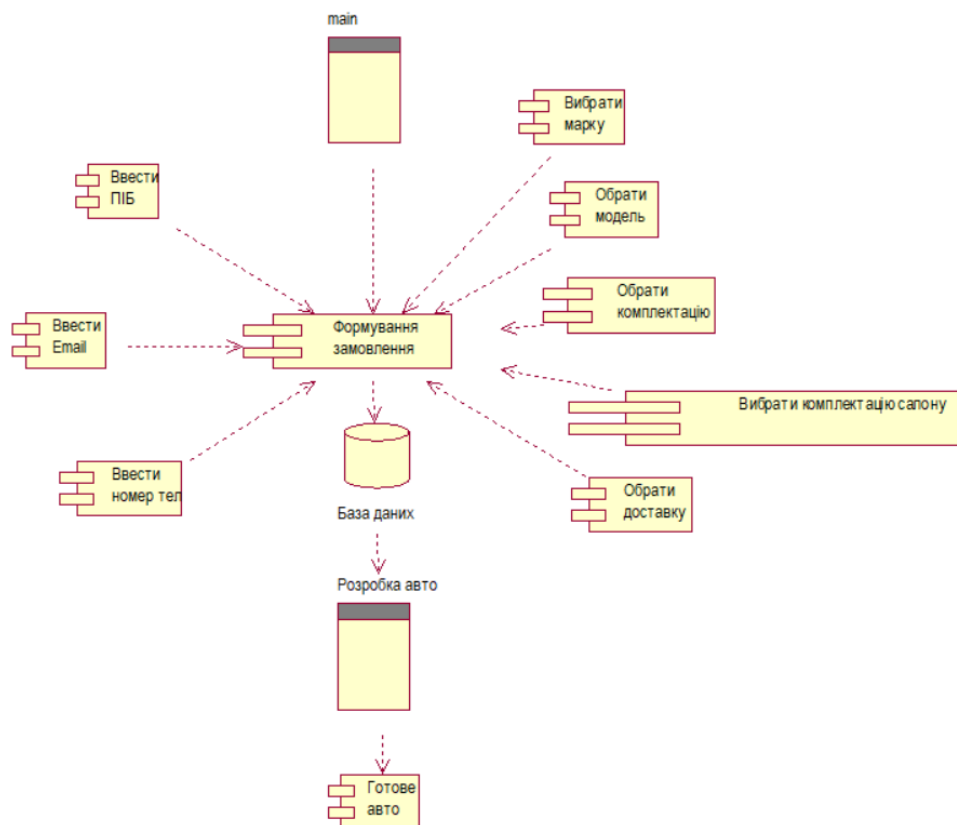


Рис.2.7 Діаграма компонентів проектованої системи

Опис:

Найважливішою функцією на даній діаграмі є main, яка запускає всі процеси(знаходиться вона у частині сервера). Дана функція сполуає з собою наступну дуже важливий компонент: формування замовлення. Завдяки ньому ми і формуємо наше замовлення з менших компонентів, які входять у його склад.

Настпним кроком ці дані переходять до бази данпх. Звідки вони потрапляють на наступну важливу функцію(Розробка авто). У цій функції ми описуємо, як будуть виробляти авто.

На виході даної діаграми ми отримуємо повністю готове авто, яке варто продати потенційному клієнту.

2.3. Висновок до другого розділу.

У даному розділі було проведено детальний аналіз та проектування ключових аспектів системи за допомогою різноманітних діаграм. Кожна з використаних діаграм слугувала своєю унікальною метою, сприяючи кращому розумінню вимог системи та її архітектурних рішень.

Діаграма варіантів використання визначила основні сценарії взаємодії користувачів з системою, надаючи узагальнене уявлення про функціональність.

Діаграма класів надала детальний огляд основних класів системи та їх взаємодії, що сприяє покращенню розуміння структури програмного коду.

Діаграма кооперацій вказала на взаємодію об'єктів та їх ролей, що є важливим для розуміння взаємодії компонентів системи в конкретних сценаріях.

Діаграми послідовності та діаграма діяльності відображають хідвиконання операцій та взаємодію об'єктів у часі, що дозволяє легше виявити важливі етапи та взаємодію між компонентами системи.

Діаграма компонентів визначила ключові компоненти системи та їх залежності, сприяючи розумінню структури великомасштабних архітектурних вирішень.

Загальною метою цього розділу було не лише проектування окремих елементів системи, але і визначення їх взаємодії та інтеграції для забезпечення ефективного та надійного функціонування системи.

Розділ 3. Вибір засобів реалізації та конструювання системи.

3.1. Вибір програмного рішення.

Для реалізації даної роботи було обрано наступний стек технологій:

- Візуальна частина: Html, Css, JS
- Серверна частина: Python(Flask)
- База даних: SQLite

Ось декілька причин для цього вибору:

1) Простота та Легкість Використання:

HTML, CSS і JavaScript є потужними та простими засобами для створення інтуїтивно зрозумілого та привабливого інтерфейсу. Flask, який є простим фреймворком для Python, пропонує простий та зрозумілий спосіб розробки серверної частини.

2) Швидкість Розробки:

Flask володіє мінімалістичною архітектурою, що полегшує розробку та швидкість написання коду. JavaScript дозволяє реалізовувати інтерактивність та динамічність веб-сторінок без перезавантаження.

3) Доповнення:

Flask може бути легко розширений за допомогою різних розширень та бібліотек. Цей стек технологій гнучкий і дозволяє масштабувати додаток в майбутньому, додавати нові функції та вдосконалювати його.

4) Універсальність та Сумісність:

Використання HTML, CSS, Python і JavaScript дозволяє створювати веб-застосунки, які можуть працювати на різних платформах та браузерах. Даний стек технологій підтримує велику кількість пристроїв і операційних систем.

5) Велика Спільнота та Підтримка:

Python і JavaScript є популярними мовами програмування з великою спільнотою розробників. Це означає наявність великої кількості бібліотек, фреймворків та ресурсів для вирішення різних задач.

3.2. Аналіз контрольного прикладу.

Для початку, користувач попадає на головну сторінку, де потрібно буде обрати марку автомобіля:

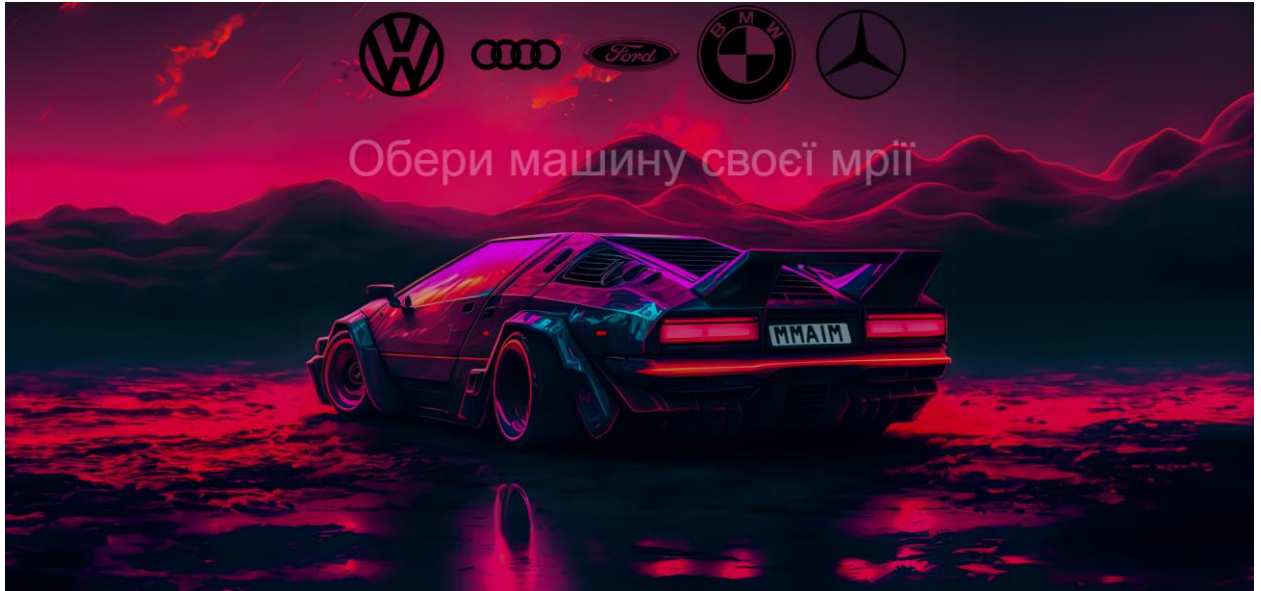



Рис. 3.1. Реєстрація

Після того, як марка автомобіля була успішно обрана(у нас Audi), користувача перекине на сторінку з моделями:

Ласкаво просимо до нашого автосалону!

Автомобілі марки audi



Комплектація мотору:

- Light: 380 к.с і 2200 к.м
- Medium: 500 к.с і 2000 к.м
- Full: 600 к.с і 2200 к.м

Комплектація салону:

- Light: Шкіряний салон
- Medium: Шкіряний салон з масажем
- Full: Шкіряний салон з масажем та вентиляцією

Палітра кольорів: Red, Blue, Black, White, Silver


Ціна: \$30,000

Комплектація мотору:

- Light: 380 к.с і 2200 к.м
- Medium: 500 к.с і 2000 к.м
- Full: 600 к.с і 2200 к.м


Комплектація салону:

- Light: Шкіряний салон
- Medium: Шкіряний салон з масажем
- Full: Шкіряний салон з масажем та вентиляцією



Палітра кольорів: Red, Blue, Black, White, Silver

Ціна: \$40,000



Комплектація мотору:

- Light: 300 к.с і 1800 к.м
- Medium: 400 к.с і 1600 к.м
- Full: 500 к.с і 1800 к.м

Комплектація салону:

- Light: Спортивні сидіння
- Medium: Спортивні сидіння з підігрівом
- Full: Спортивні сидіння з підігрівом та пам'яттю

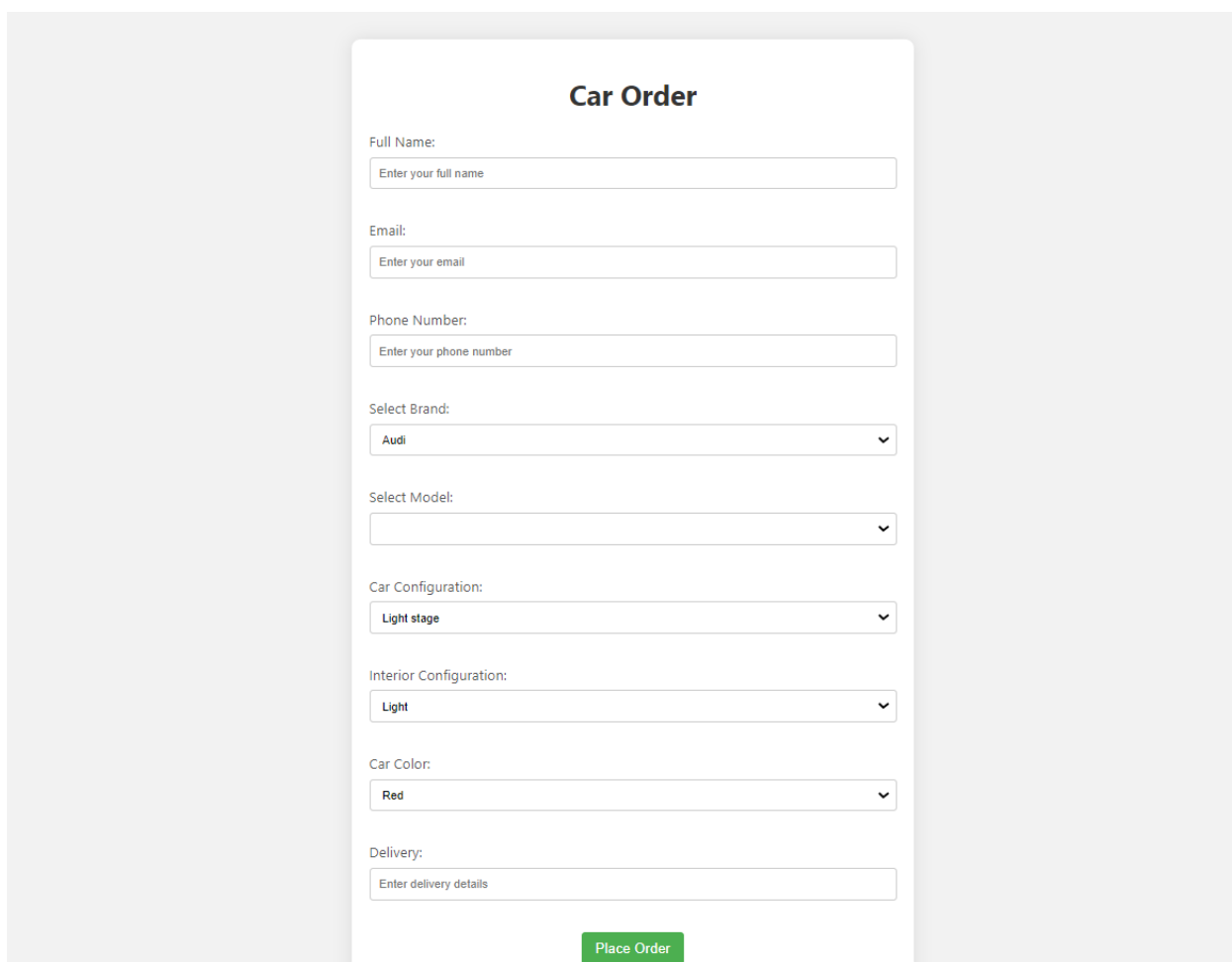
Палітра кольорів: Red, Blue, Black, White, Silver

Ціна: \$35,000

Замовити

Рис. 3.2-3. Список моделей та їх комплектацій.

Тут ми можемо переглянути усі наші доступні моделі, усі їх варіації та складові. Після чого потрібно натиснути на кнопку «замовити», щоб перейти до форми-замовлення:

A screenshot of a web form titled "Car Order". The form is centered on a light gray background. It contains several input fields and dropdown menus. The fields are: "Full Name:" with a text input field containing the placeholder "Enter your full name"; "Email:" with a text input field containing the placeholder "Enter your email"; "Phone Number:" with a text input field containing the placeholder "Enter your phone number"; "Select Brand:" with a dropdown menu showing "Audi"; "Select Model:" with a dropdown menu showing a downward arrow; "Car Configuration:" with a dropdown menu showing "Light stage"; "Interior Configuration:" with a dropdown menu showing "Light"; "Car Color:" with a dropdown menu showing "Red"; and "Delivery:" with a text input field containing the placeholder "Enter delivery details". At the bottom right of the form is a green button labeled "Place Order".

Car Order

Full Name:
Enter your full name

Email:
Enter your email

Phone Number:
Enter your phone number

Select Brand:
Audi

Select Model:
▼

Car Configuration:
Light stage

Interior Configuration:
Light

Car Color:
Red

Delivery:
Enter delivery details

Place Order

Рис. 3.4. Форма для замовлення

На цьому етапі, користувач повинен внести усі дані у форму, щоб перейти на наступний крок. При не внесенні дані хоча б одного поля(Рис. 3.5) форма повідомить користувачу, що поле було не вписано.

Car Order

Full Name:

Губка Боб Миколайович

Email:

Enter your email

Phone Number:

+380244412355

Select Brand:

Audi

Select Model:

TT

Car Configuration:

Medium stage

Interior Configuration:

Full

Car Color:

White

Delivery:

Дно стріт, 31. USU

Place Order

Рис. 3.5. Попередження

Car Order

Full Name:
Губка Боб Миколайович

Email:
Genius112@gmail.com

Phone Number:
+380244412355

Select Brand:
Audi

Select Model:
TT

Car Configuration:
Medium stage

Interior Configuration:
Full

Car Color:
White

Delivery:
Дню стріт, 31. USU

Place Order

Рис. 3.6. Внесені дані.

Ми внесли тестові дані для цього сайту у нашу форму, єдине, що тепер нам залишилось – натиснути кнопчку «Place order». І наші дані миттєво буде збережено у Базі Даних.

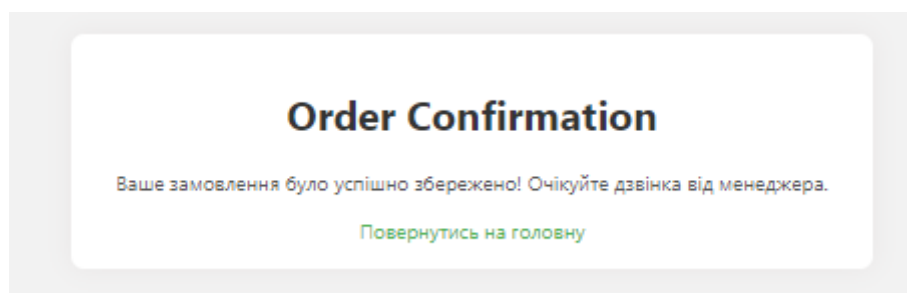


Рис. 3.7. Повідомлення про успішне збереження для клієнта

```
127.0.0.1 - - [14/Dec/2023 07:56:53] "POST /templates/order.html HTTP/1.1" 200 -
```

Рис. 3.7. Повідомлення про успішне збереження для серверної частини

Після успішного збереження даних, у нас доступна кнопка повернення на головну(Рис 3.1). При цьому наші дані буде збережено у базі даних, як це показано на скріншоті(Рис. 3.8)

7	7 Губка Боб Миколайович Genius112@gmail.com	+380244412355	audi	TT	medium	full	white	Дно стріт, 31. USU
---	---	---------------	------	----	--------	------	-------	--------------------

Рис. 3.8. Повідомлення про успішне збереження даних у БД.

На цьому автоматизація процесів завершується, далі працює з базою даних менеджер та наступні ланки механізмів, як це було показано на діаграмах.

3.3. Висновок до третього розділу.

У даному розділі було здійснено побудову автоматизованої частини, а саме сайту, що надає послуги користувачам, які бажають придбати автівку у автосалоні.

Описано основні функції та роботу сайту.

Наведено контрольний приклад, що підтверджує те, що сайт повністю працездатний, та вся система функціонує згідно поставлених завдань.

Написання сайту відбувалось за допомогою стеку технологій: html, css, JS, python(Flask), SQLite. Які були зручними для використання та простими у моментах виправлення помилок.

Розроблені елементи повністю відповідають усім вимогам, які були поставлені. Дизайн сайту є інтелектуально зрозумілим для користувача, а при написанні коду дотримувались основних вимог оптимізації коду, тому для програмістів та системних адмінів він є зрозумілим та простим під час виправлення помилок.

Висновки

Під час побудови даної інформаційної системи було вивчено нову предметну область. Проаналізовано конкурентів, визначено їх слабкі та сильні сторони, було прийнято рішення щодо впровадження технологій, які допомогли нам б конкурувати з даними компаніями, та показати себе краще. Також було вивчено наших головних партнерів, які допоможуть нам працювати.

Результатом даного дослідження, було створено сайт, який автоматизує певні процеси у компанії, що допоможе розвантажити нам трафік користувачів та суттєво скоротити витрати на різні інші послуги.

Після проєктування та обирання стеку технологій була створена програмна реалізація, яка і є нашим результатом даного дослідження.

Список використаних джерел

Сайти

1. w3schools.com. HTML Introduction. URL: <https://www.w3schools.com/html/> (дата звернення: 23.10.2023).
2. w3schools.com. CSS Tutorial.
URL: <https://www.w3schools.com/css/default.asp> (дата звернення: 23.10.2023).
3. OpenAI. ChatGPT: Language Models for Conversational Agents. URL: <https://chat.openai.com/> (дата звернення: 10.10.2023)
4. W3Schools. JavaScript Tutorial. URL: <https://www.w3schools.com/js/default.asp> (дата звернення: 25.10.2023).
5. W3Schools. Python Tutorial. URL: <https://www.w3schools.com/python/default.asp> (дата звернення: 23.10.2023).
6. Flask Documentation. Flask 3.0.x Documentation. URL: <https://flask.palletsprojects.com/en/3.0.x/> (дата звернення: 25.10.2023).
7. SQLAlchemy. URL: <https://www.sqlalchemy.org/> (дата звернення: 26.10.2023).
8. SQLite Tutorial. URL: <https://www.sqlitetutorial.net/> (дата звернення: 27.10.2023).

Книги, статті:

1. Литвин В.В. Проектування інформаційних систем /Н. Б. Шаховська, В. В. Литвин – Львів: “Магнолія-2006”. – 380 с.
2. Пономаренко В.С. Проектування інформаційних систем : посібник / В.С. Пономаренко, О.І. Пушкар, І.В. Журавльова, С.В. Мінухін - К. : Академія, 2002. - 488 с.
3. Проектування інформаційних систем: навчальний посібник / В.С. Авраменко, А.С. Авраменко. – Черкаси: Черкаський національний університет ім. Б. Хмельницького, 2017. – 434 с.: іл.
4. Методичні вказівки для виконання лабораторних робіт з курсу "Проектування інформаційних систем" / Укладач: Басюк Т. М., к.т.н., доцент каф. ІСМ. - НУЛП, 2021. - 105 с.

Додаток А.

Посилання на репозиторій з кодом до сайту:

<https://github.com/Sashik01/Kursova.git>