

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ  
Факультет кібербезпеки, комп'ютерної та програмної інженерії  
Кафедра комп'ютерних інформаційних технологій

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

\_\_\_\_\_ А.С.Савченко

« \_\_\_\_\_ » \_\_\_\_\_ 2021 р.

# ДИПЛОМНИЙ ПРОЕКТ

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ “БАКАЛАВР”

**Тема:** «Лендінг-сторінка для закладу готельного бізнесу»

**Виконавець:** студентка УС-412 Перепада Дар'я Андріївна

**Керівник:** д.т.н., професор Зіатдінов Юрій Кашафович

**Нормоконтролер:** ст. викл. Шевченко О.П.

КИЇВ 2021

# НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра комп'ютерних інформаційних технологій

Освітній ступінь: Бакалавр

Галузь знань, спеціальність, спеціалізація: 12 “Інформаційні технології”,  
122 “Комп'ютерні науки”, “Інформаційні управляючі системи та технології”

**ЗАТВЕРДЖУЮ**

Завідувач кафедри

\_\_\_\_\_ А.С. Савченко

« \_\_\_\_ » \_\_\_\_\_ 2021 р.

## **ЗАВДАННЯ**

**на виконання дипломного проекту студента**

Перепада Дар'я Андріївна

1. Тема проекту: «Лендінг-сторінка для закладу готельного бізнесу» затверджена наказом ректора від «22» квітня 2021 р. № 636/ст.
2. Термін виконання роботи: з 10.05.2021 до 14.06.2021.
3. Вихідні дані до роботи: існуючі веб-сайти Landing page, загальні відомості про обрані веб-технології, літературні джерела з досліджуваної проблеми.
4. Зміст пояснювальної записки: аналіз предметної області, переваги та недоліки лендінгу, впровадження покращених методів створення лендінгу, аналіз інструментів створення, розробка структури та проекту.
5. Перелік обов'язкового графічного матеріалу: «Властивості CSS і їх підтримка браузерами», «Порівняння Landing page та звичайного сайту».

## КАЛЕНДАРНИЙ ПЛАН

п/п	Етапи виконання дипломної роботи	Термін виконання етапів	Примітка
1.	Аналіз літератури та джерел за темою дипломного проекту.	10.05.2021 – 12.05.2021	
2.	Розробка та затвердження плану дипломного проекту.	12.05.2021	
3.	Консультації з науковим керівником. Аналітичний огляд і постановка задачі.	13.05.2021	
4.	Розробка розділу 1: Аналіз предметної області	14.05.2021 – 20.05.2021	
5.	Розробка розділу 2: Аналіз обраних інструментів і технологій	20.05.2021 – 23.05.2021	
6.	Розробка Landing page	23.05.2021 – 30.05.2021	
7.	Розробка розділу 3: Реалізація Landing page для закладу готельного бізнесу	31.05.2021 – 02.06.2021	
8.	Висновки та оформлення пояснювальної записки дипломного проекту.	03.06.2021 – 05.06.2021	
9.	Підготовка до захисту та попередній захист дипломного проекту.	06.06.2021 – 11.06.2021	

Студент

( *Перепада Д.А.* )

Керівник дипломної роботи

( *Зіатдінов Ю.К.* )

## РЕФЕРАТ

Пояснювальна записка до дипломної роботи «Лендінг-сторінка для закладу готельного бізнесу» містить: 64 с., 45 рис., 12 літературних джерел, 1 табл., 1 графік.

Мета: створити Landing page з можливістю відправки заповнених даних у контактній формі.

Об'єкт дослідження: інформаційні технології створення веб-сервісу Landing page.

Предмет дослідження: розробка лендінг-сторінки готелю для відправки контактних даних.

Методи дослідження: аналіз методів і програмного забезпечення для створення Landing page.

Отримані результати: реалізовано Landing page, який дозволяє заповнювати контактну форму і відправляти її.

Для розробки Landing page використані HTML, CSS, JavaScript мови та програмна середовища Microsoft Visual Studio.

Результати дипломної роботи планується використовувати для розширення функціоналу, подальшого розвитку та введення в експлуатацію.

WEB-ДОДАТОК, РОЗРОБКА, LANDING PAGE, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ.....	6
ВСТУП.....	7
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....	9
1.1. Поняття Landing page .....	9
1.2. Відмінності лендінгу від веб-сайту.....	10
1.3. Переваги та недоліки посадкового сайту .....	12
1.4. Приклади Landing page.....	14
1.5. Пропозицій щодо удосконалення підходів та методів створення лендінгу ...	17
Висновки до розділу 1 .....	20
РОЗДІЛ 2. АНАЛІЗ ОБРАНИХ ІНСТРУМЕНТІВ І ТЕХНОЛОГІЙ ЛЕНДІНГУ ....	22
2.1. Середовище розробки Visual Studio Code .....	22
2.2. Псевдомова HTML .....	25
2.3. Псевдомова CSS .....	28
2.4. Мова програмування JavaScript.....	30
2.5. Мова програмування PHP .....	33
Висновки до розділу 2 .....	34
РОЗДІЛ 3. СТВОРЕННЯ LANDING PAGE ДЛЯ ЗАКЛАДУ ГОТЕЛЬНОГО БІЗНЕСУ.....	36
3.1. Діаграма станів.....	36
3.2. Контекстна діаграма .....	37
3.3. Структура та основні функції зрозумілого та зручного інтерфейсу .....	37
3.4. Сценарій роботи користувача з системою .....	41
3.5. Реалізація проекту .....	49
Висновки до розділу 3 .....	60
ВИСНОВКИ.....	62
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	64

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

Landing page – односторінковий сайт;

HTML – HyperText Markup Language — мова розмітки гіпертекстових документів;

CSS – Cascading Style Sheets - каскадні таблиці стилів;

PHP – Hypertext Preprocessor - скриптова мова загального призначення;

jQuery — набір функцій JavaScript;

ЦА- цільова аудиторія;

SEO - оптимізація для пошукових систем;

Хедлайн – заголовок;

Футер - структурний елемент, розташований в нижній частині сторінки;

CTAA - Call to action – заклик до дії;

VS Code - Visual Studio Code – редактор вихідного коду.

## ВСТУП

Деякий час назад легка і рентабельна альтернатива у вигляді Landing page виникла у відповідь на громіздкі сайти. Потенціал Landing page швидко розкрили та активно почали використовувати. Особливо привабливою в перший період появи такої інтернет-сторінки на просторах Інтернету стала функція, яка продає. Розробка подібних сайтів почалася у великих масштабах, і пріоритетом було швидше наявність, а не індивідуальність і якість. Неусвідомлений підхід став причиною повторюваних помилок і неробочих рішень. На жаль, не можна сказати, що ця тенденція повністю припинила існування: в мережі маса невдалих прикладів. Але з упевненістю можна стверджувати, що Landing page зайняли своє місце серед сайтів і продовжують свій розвиток.

Чому ж Landing page не втрачають своєї актуальності? В першу чергу — компактність. Сучасну людину оточують тони інформації — стає все складніше орієнтуватися й аналізувати її. З цієї причини багато веб-сайтів трансформуються і спрощуються, намагаючись досягти мінімалізму, щоб відвідувач не розмінювався на несуттєві деталі. По-друге, — маркетинговий інструмент. Landing page, як компетентний реалізатор, володіє умінням переконання без багатослів'я і, як кваліфікований мерчендайзер, допоможе спеціалізувати продукти та послуги й проводити їх оновлення.

Бурхливий розвиток онлайн-продажів в готельній індустрії змушує готельєрів все активніше освоювати інструменти інтернет-маркетингу та користуватися послугами програмістів. Вдала цільова сторінка — найбільш ефективний інструментарій для торгівлі, здатний привернути увагу оточуючих.

Дана дипломна робота присвячена розробці Landing page для закладу готельного бізнесу. Під час виконання роботи було проаналізовано основні проблеми створення посадкових сторінок, виявлено переваги та недоліки застосування даного веб-сервісу, приведено пропозиції щодо удосконалення підходів та методів розробки подібних сайтів. Також було розроблено структуру інтерфейсу Landing page.

Метою написання дипломної роботи є створення Landing page з можливістю відправки заповнених даних у контактній формі.

Об'єктом дослідження є інформаційні технології для створення веб-сайту Landing page.

Предметом дослідження є розробка Landing page для обміну контактними даними клієнтів готелю.

Методами дослідження є аналіз методів і програмного забезпечення для створення Landing page.

Отримані результати — реалізований Landing page, який дозволяє заповнювати контактну форму з даними і розсилати її.

Наукова новизна отриманих результатів: удосконалено структуру посадкової сторінки для готелю.

Для розробки Landing page використовувались: мова розмітки HTML, мова опису зовнішнього виду документа CSS, мова програмування PHP, а також мова програмування JavaScript, зокрема бібліотека jQuery. Програмна середовище - Microsoft Visual Studio.

Сама веб-сторінка складається із головного меню, блоків опису готелю, відео-презентації, зображень готелю та форми заповнення контактних даних, за допомогою якої власник зможе отримувати контакти клієнтів на пошту.



# РОЗДІЛ 1

## АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1. Поняття Landing page

Об'єктом дослідження даної роботи є Landing page для закладу готельного бізнесу. Лендінг (landing page), чи посадкова сторінка — це особливий тип сайтів, оптимізованих для спонукання до дії інтернет-користувача. Традиційний Landing page описує продукт, його переваги і вказує на необхідність в придбанні цього продукту. В ідеальному світі лендінг створюють таким чином, щоб користувач, який нічого не знає про продукт, після відвідування сайту зробив замовлення. Тобто це сайт з однією сторінкою, яка просуває конкретну пропозицію. Для того, щоб скористатися цією послугою, користувач повинен щось зробити: купити, замовити, підписатися, зареєструватися. Зробити це користувач може за допомогою заповнення форми збору контактів.

Які ще можуть бути цілі у Landing page? Наприклад, запис на вебінар, реєстрація клієнта, генерація заявок на послугу або покупку товару, завантаження електронної книги або іншого документу, оформлення підписки на новини і продукти, які пропонуються і т.д. Головною ціллю лендінгу для готелю є бронювання номеру та отримання контактних даних потенційного клієнта.

Виробник має багато інформації про товар, а у користувачів часу, як правило, дуже мало. Користувачі побіжно переглядають сайт для вибору товару або послуги, де розміщена основна інформація: фото, ціна, назва товару.

Зазвичай, всі власники, які вирішили розробити для своєї компанії таку сторінку чекають відразу продажів з лендінгу. Вони хочуть, щоб відвідувач залишав свої дані, вони дзвонили та лише дізнавалися деталі відправки чи бронювань.

Кафедра КІТ				НАУ 21 39 87 000 ПЗ			
Виконала	Перепада Д.А.			АНАЛІЗ ПРЕДМЕТНОЇ ОБ- ЛАСТІ	Літера	Аркуш	Аркушів
Керівник	Зіатдінов Ю.К.					9	11
Консульт.					412 122		
Н-котрол.	Шевченко О.П.						
Зав. каф.	Савченко А.С.						

Звичайно, таке можливо. Але для цього компанії потрібен або дешевий товар, або дуже добре створений e-mail маркетинг, або просування в соціальних мережах.

Насправді, основною загальною метою лендінгу є отримання контакту і виклик інтересу до продукту, який надалі менеджери будуть розвивати і перетворювати в угоду.

Кожен готельєр хоче догодити своїм гостям, зробити їх відпочинок незабутнім, щоб, повернувшись додому, гості із задоволенням розповідали про відпочинок своїм знайомим і обов'язково хотіли повернутися. Незалежно від того, чи є у готелю основний сайт, можна зробити лендінг під окрему послугу і приводити людей з сайту і з реклами. Особливість такої сторінки в тому то, що вона показує більш високу конверсію, ніж середній сайт, тобто від такої ж кількості відвідувачів прийде більше замовлень. Щоб порахувати конверсію, кількість користувачів, які заповнили форму, ділять на кількість користувачів, які зайшли на сторінку. Вище конверсія — ефективніше лендінг. Досягається це завдяки тому, що структура сторінки заточена під вчинення конкретної дії й на ній немає чинників, що відволікають.

## **1.2. Відмінності лендінгу від веб-сайту**

Лендінги досить широко використовуються в маркетингу в силу своєї принципової відмінності від традиційного веб-сайту: якщо головна мета Landing page – конкретна (цільова) дія, то веб-сайти мають багатоцільовий характер. Подібне обмеження дозволяє посадковій сторінці бути неймовірно ефективною, а тому і залишатиметься популярною серед фахівців.

Головна перевага лендінгу – користувач фокусується на одній пропозиції.

Порівняймо (табл.1.2.1) і з'ясуємо, в чому різниця між лендінгом і звичайним сайтом.

Таблиця 1.2.1

<b>Landing page</b>	<b>Звичайний сайт</b>
Більш висока конверсія з відвідувача в контакт	Менша конверсія з відвідувача в контакт
Велика кількість заявок, але менш зацікавлених	Менша кількість заявок, але більше зацікавлених
Інформація доноситься тезисно і основними смислами	Є можливість викласти максимально повну інформацію
Не відволікає від цільової дії	Багато часу знаходяться на сайті, але можуть забути про ціль
Швидке створення, а отже швидкий старт продажів	Повільне створення, а значить затяжний старт продажів
Неможливе масове просування через SEO	Можливе масове просування через SEO
Викликає менше довіри	Викликає більше довіри
Не найкращий варіант для новин компанії	Ідеальний варіант для новин компанії
Високі знання про поведінку людей і інтернет-маркетингу	Базові знання про поведінку людей і інтернет-маркетингу

Хоч і коротко, але ми розібрали основні відмінності лендінгу від багатосторінкового сайту. І щоб підвести якусь рису, ще раз розберемо ситуацію, в яких випадках створення лендінгу буде краще:

- Велика компанія з послугою, яку хочеться виділити окремо (створити окремий сайт з іншим потоком користувачів);
- Компанія тільки стартує, потрібен швидкий запуск і швидке отримання заявок;
- Компанія, яка потребує сайт-візитку компанії з додатковим отриманням контакту;
- Компанія, якій потрібен сайт, що викликає первинний інтерес і "змушує" клієнта залишити свої контактні дані.

Тобто, якщо потрібен швидкий результат — вибираємо лендінг. Таке рішення гарантує максимально швидке отримання заявок або замовлень. Також його можна тестувати та підвищувати конверсію.

Якщо потрібно сформувати довіру до бренду, дати нову інформацію про товари і послуги - можна використовувати стандартний сайт.

### 1.3. Переваги та недоліки посадкового сайту

Наразі дуже складно успішно розвивати бізнес через велику кількість конкурентів. Тому в боротьбі за лідерство необхідно використовувати сучасні та дієві маркетингові інструменти.

Як ми можемо побачити на графіку трендів нижче (рис. 1.3.1), інтерес до Landing page з кожним роком збільшується і спостерігається явний позитивний тренд:

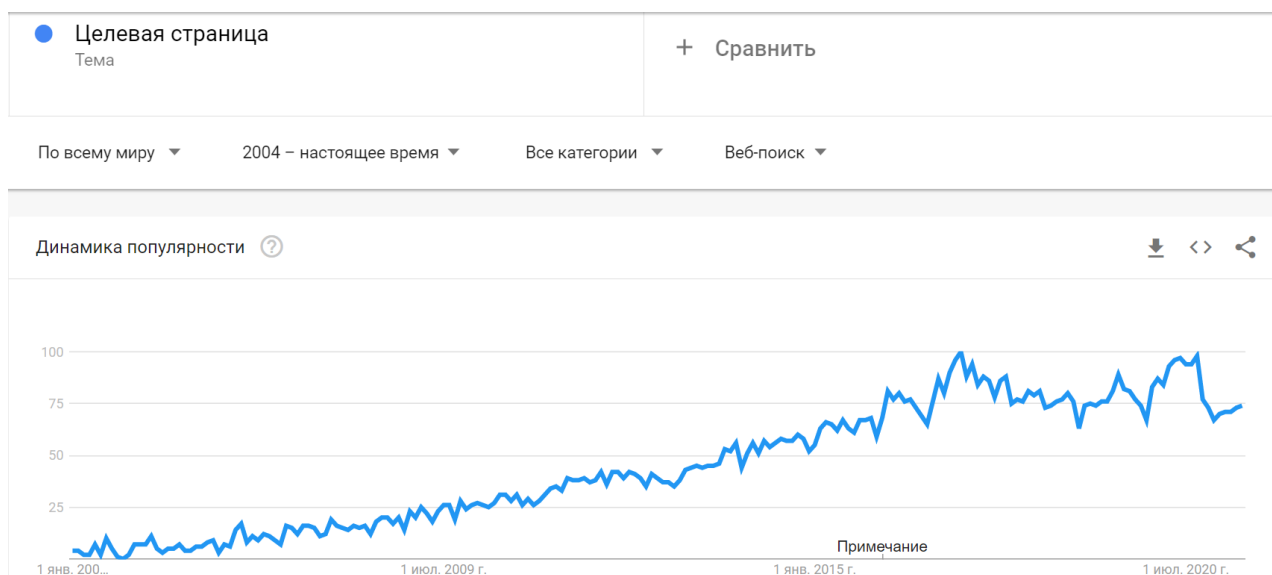


Рис. 1.3.1. Графік трендів

Роздивімося більш детально (рис.1.3.2), чому для просування продукту/товару потрібно використовувати саме лендінг. Зараз визначимо основні переваги створення landing page:

- релевантна запитам користувача сторінка з простим маршрутом «одна мета — одна дія — одна сторінка»;

- заголовки і підзаголовки, які залучають до дії;
- дизайн, що враховує цілі бізнесу та аудиторії;
- простий і зрозумілий текстовий і візуальний контент;
- клікабельна сторінка (активні кнопки з можливістю зробити бронювання, покупку або підписку в один клік);
- інфографіка, що пояснює переваги готелю і принципи роботи;
- залучення певної ЦА, навіть якщо раніше на неї не розраховували;
- отримання контактів відвідувача;
- підвищення конверсії: завдяки лендінгу відсоток цільових дій збільшується в середньому на 15%;
- ефективний заклик до дії в оптимальному місці сторінки, який і стане шляхом до розширення аудиторії.

## Почему именно лендинги?

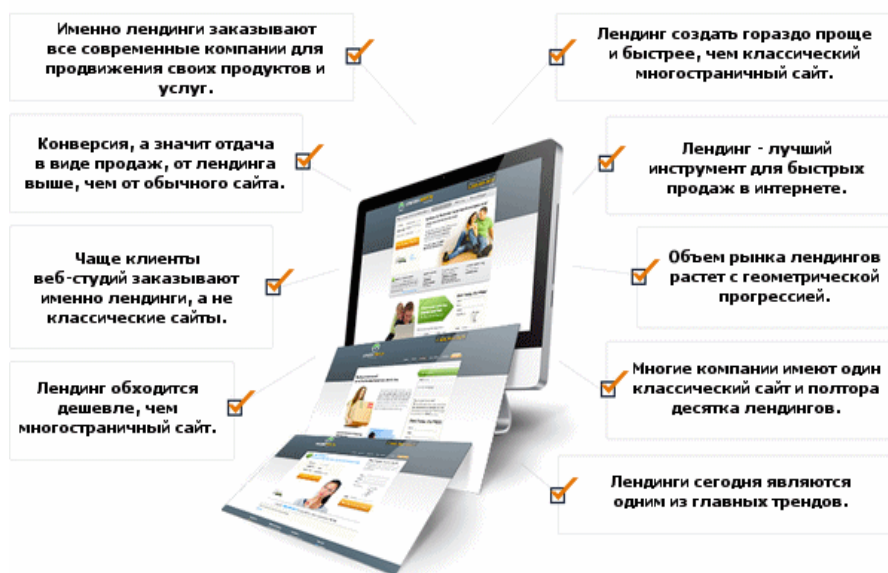


Рис.1.3.2. Переваги лендінгу

Недоліки landing page:

1. обмеження певними рамками: на посадковій сторінці неможливо розписати в дрібних подробицях переваги;
2. погано просувається в SEO і індексується пошуковими системами.

Також було проведено аналіз статистичних даних, які демонструють переваги використання лендінг-сторінок для поліпшення маркетингових показників:

1. Середня конверсія для лендінгу - 9,7%;
2. У формах на лендінгах сама висока конверсія - 23%;
3. Відповідь на страхи ЦА підвищує конверсію на 80%;
4. Персоналізовані СТА конвертують на 202% краще;
5. 48% найкращих лендінгів приваблюють органічний трафік;
6. 65% лендінгів включають бренд у назву;
7. У 30% успішних лендінгів є відео;
8. У 36% найкращих посадкових сторінках є відгуки.

Тобто для підвищень конверсії потрібно використовувати оптимальну структуру сторінки, якісний контент і хороший дизайн.

#### **1.4. Приклади Landing page**

Візьмемо в порівняння інші посадковий сторінки, які мають велику популярність і величезну кількість переглядів в день у своїй статистиці. Зробивши підбірку успішних лендінгів з незвичайним дизайном і контентом, можна розібрати причини успішного лендінгу на їх прикладі.

Університет Атабаски вперше ввів дистанційне навчання в Канаді в 1970-х роках. Сьогодні він використовує лендінг (рис.1.4.1) для активації своїх ініціатив по онлайн-реєстрації і, включаючи цей приклад на малюнку, представляє 14 програм сертифікації. Це розумний вибір, оскільки Landing page дозволяє університету Атабаски зосередити увагу відвідувача на конкретній сфері своїх численних онлайн-програм.

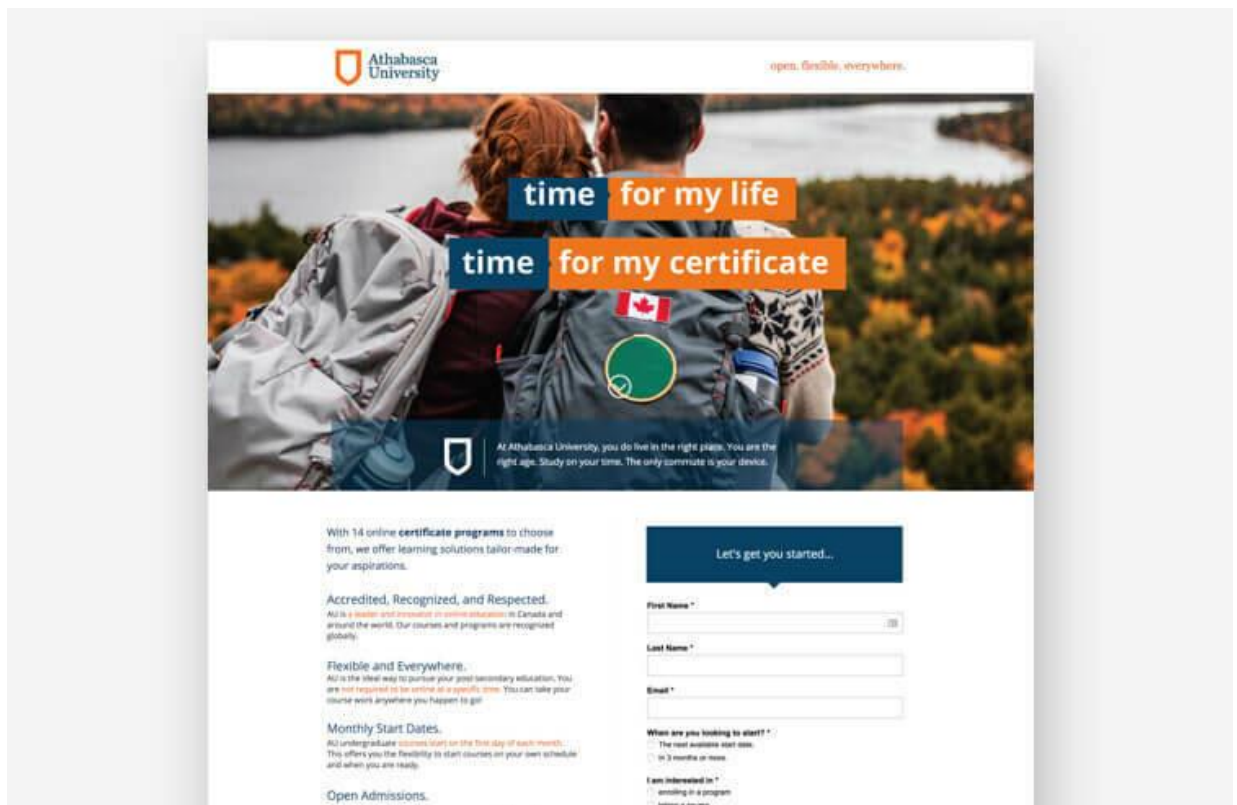


Рис.1.4.1. Лендінг університету Атабаски

Ось сторінка (див. рис.1.4.2) про їжу Variatric Eating (Агентство: Lifestyle Collection), яка показує, чому індивідуальність і стиль так важливі для лендінгу. Можна легко уявити версію сайту цієї компанії, яка виглядає набагато більш клінічною та науковою, але маркетологи з Lifestyle Collective зробили дизайн трохи барвистим і дружнім, що зробило його більш доступним. Підхід працює. Ця сторінка має дивовижний коефіцієнт конверсії, понад 39%.

Переваги:

1. лендінг витриманий в єдиному «ігровому» стилі;
2. дуже запам'ятовується логотип всюди на всіх розворотах;
3. захоплива організація інформації;
4. помітна кнопка соціальних сервісів.



Рис.1.4.2. Лендінг Variatric Eating

Виробники гострого соусу Mule створили красивий Landing page (рис.1.4.3), щоб розповісти про переваги продукту. Динамічна анімація спонукає гортати сторінку до кінця, а частина інформації про продукт упакована в розважальне відео, через що сторінка не перевантажена текстом. В кінці Лендінгу є рецепти і соціальні докази у вигляді відгуків покупців з фотографіями.

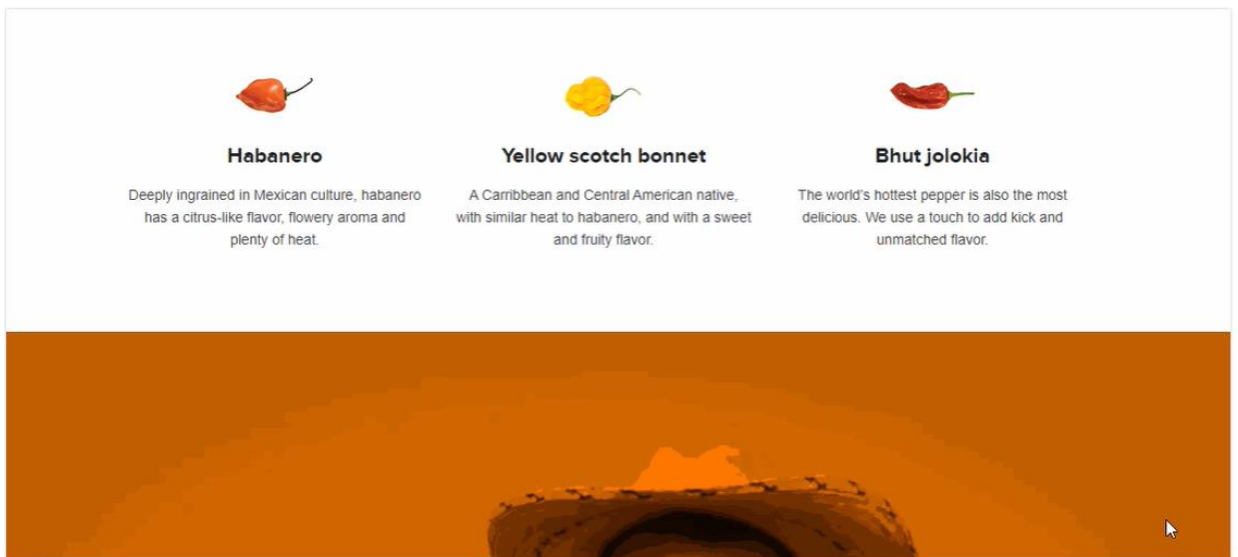


Рис.1.4.3. Лендінг Mule



На сторінці (рис.1.4.4) заводу з виробництва бетону «Центр бетону» - багато рухомих елементів, а фон повторює колір і фактуру бетону. Інформація подана блоками з цінами, акціями, умовами роботи. Яскраві кольори і картинки, що виїжджають з боків, привертають увагу, а невелика кількість тексту спрощує пошук інформації.

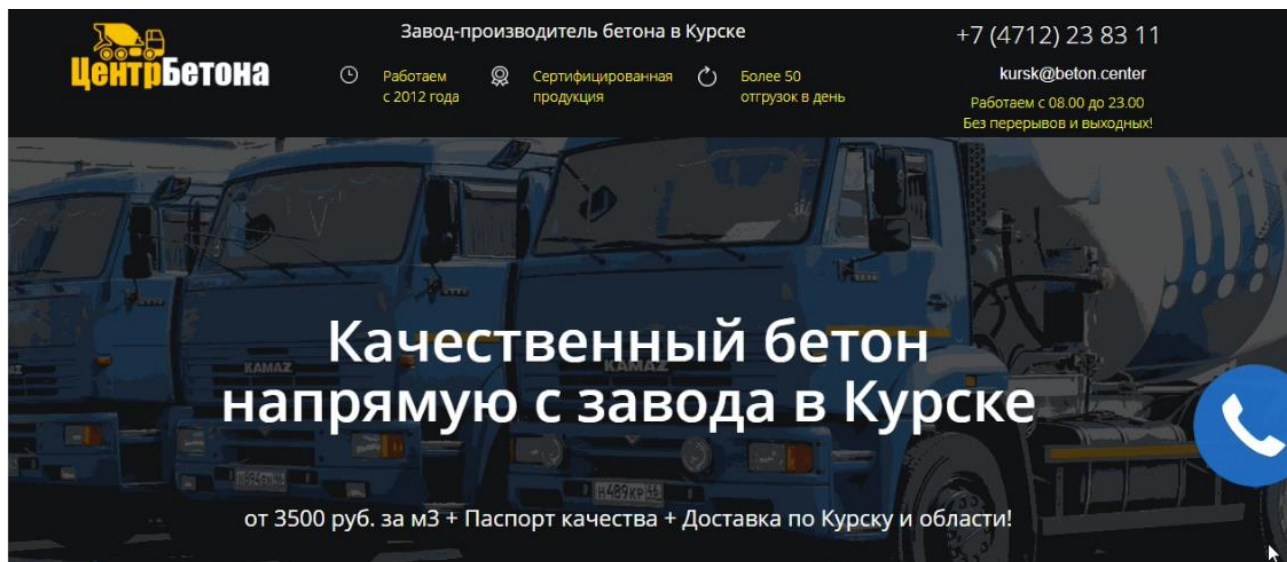


Рис.1.4.4. Лендінг «Центр бетону»

Підбиваючи підсумки, можна виділити наступні плюси хороших лендінгів: лаконічність в тексті і дизайні, стильна анімація, нестокові фотографії, наявність оригінальних малюнків, відгуків, гумору, простих форм реєстрації, чітко позначених переваг та зрозумілих кнопок СТА.

### **1.5. Пропозицій щодо удосконалення підходів та методів створення лендінгу**

Проблема в тому, що посадкові сторінки далеко не завжди працюють так, як хочеться. Припустимо, відвідувачі «приземляються» на вашому сайті, клікнувши по контекстній рекламі. Вони читають вашу пропозицію і залишають лендінг-сторінку, не зробивши конверсійних дій.

Дуже часто посадкові сторінки не приносять очікуваного результату через декілька типових помилок, що допускаються при їх створенні. виправивши їх, можна

значно підвищити ефективність. Нижче представлений опис типових проблем, що перешкоджають повноцінній роботі цільових сторінок.

Проблема 1: неправильний заголовок.

Посадкова сторінка починається з правильного заголовка. Хедлайн — перше, що бачить відвідувач. Тема — це шанс швидко розповісти клієнту про себе, зацікавити його, пояснити свої наміри та мотивувати залишитися на сайті.

Проблема 2: відсутність ціннісної пропозиції.

По суті, ціннісна пропозиція — це обіцянка клієнтові будь-якої вигоди. У наступному прикладі нескладно визначити VP: «Підпишіться на розсилку та отримаєте електронну книгу в подарунок». Ціннісна пропозиція — основний каталізатор конверсії лендінгу. Відвідувачі діють відповідно до конверсійного сценарію тільки в тому випадку, якщо вони розуміють, яку конкретну вигоду отримають.

Причина 3: неправильний текст.

Хороший текст — невід'ємна частина ефективної посадкової сторінки. Природно, ви не можете конвертувати відвідувачів в покупців без слів, оскільки з їх допомогою ви керуєте поведінкою користувачів.

Проблема 4: занадто багато опцій.

Найбільш ефективні посадкові сторінки містять одну конкретну пропозицію. Запам'ятайте формулу: одна цільова сторінка = один продукт і одна пропозиція. Форми для заповнення повинні містити мінімум полів. В ідеалі їх повинно бути не більше двох: для E-mail або номер телефону та імені користувача.

Проблема 5: шаблонний дизайн.

Шаблонний дизайн однаково погано показується на довірі відвідувача до компанії.

Проблема 6: низька швидкість завантаження.

Весь проєкт може наблизитися до марного, якщо він буде повільно завантажуватися. За результатами тестування, якщо посадкова сторінка не завантажується протягом 5 секунд, близько 75% відвідувачів відчувають дискомфорт, підкріплений негативом. Як результат - залишають сторінку, навіть не дочекавшись її завантаження.

Проблема 7: рішення не тестувати сайт на технічні помилки.

Впевненість в тому, що сайт ідеально відображається на комп'ютері, у всіх браузерах або смартфонах і не перевірити це - критична помилка. Помилки у верстці можуть вплинути на дизайн сторінки та неправильно показувати його, помилки в крос-браузерності є причиною криво завантаженого сайту на одному з браузерів, а неробочі кнопки, форми зворотного зв'язку (що є досить частим явищем) просто не дають відвідувачу можливості стати клієнтом.

З огляду на всі найпопулярніші помилки і розібрані головні переваги наявних лендінгів, можна розробити посадкову сторінку з «золотою серединою» для найбільш ефективного досягнення поставленої мети. Проаналізувавши це, я винесла пропозиції щодо удосконалення підходів створення Landing page. І на базі об'єкту дослідження (лендінг для закладу готельного бізнесу) я виділила ключові компоненти ефективної лендінг-сторінки для готелю:

#### 1. Заголовок

Будь-який готельний бізнес повинен починатися з визначення унікальних характеристик бренду. Необхідно добре сформулювати, чим пропонований готель відрізняється від конкурентів. На базі цих висновків оформлюємо заголовок нашої сторінки. Наш готель знаходиться у центрі, тому текст заголовку буде звучати наступним чином: «CENTRAL Kyiv - готель у центрі Києва».

#### 2. Логотип, назва і контакти

Обов'язково розміщуємо на посадковій сторінці логотип готелю, щоб дозволити користувачеві його ідентифікувати. Це підвищить його рівень довіри. Так само з номером телефону для зв'язку.

#### 3. Демонстрація продукту / послуги

Потрібно створити у користувача відчуття тестування продукту. Існує кілька способів для того, щоб досягти цієї мети:

- фотографія – зображення великого розміру для фону хедера, а після розкриття подробиці ілюстраціями відповідної тематики;
- відео — відвідувачі мають більшу довіру до такого контенту.

#### 4. Лід-форма

Поле зворотного зв'язку посеред іншого контенту сайту потенційний клієнт повинен знаходити без труднощів.

## 5. СТАА

Call to Action (СТА), або заклик до дії — один з найважливіших елементів структури лендінг-сторінки. Інфо продукт, що включає такий текст, мотивує користувача зробити конверсійне дію. Стандартним СТА-елементом є графічна кнопка, завдяки якій потенційний замовник переміщається до кінцевого пункту.

## 6. Дизайн лендінгу

Коли розробляється дизайн лендінгу, важливо врахувати відразу кілька факторів:

Комфорт перегляду і читання. Очі не повинні втомлюватися. Важливо дотримуватися приємне поєднання кольорів.

Відповідність продукту або пропонованої послуги. Тут важливі і колірні гамма, і настрій, і окремі фотографії. Також в дизайн повинні входити фото готелю, які б мотивували до бронювання.

## 7. Структура

Найголовнішим фактором при створенні лендінг-сторінки є її структура. Це розроблені і логічно структуровані блоки, які у вигляді своєрідної історії ведуть відвідувача до цільової дії.

Розібравши дані принципи розробки інтерфейсу, прийнято рішення здійснити акцент на простоті та якості інформативності, щоб клієнт, перейшовши на цільову сторінку, міг роздобути чітку і потрібну для нього інформацію про готель. Так само міг дізнатися про те, яким чином здійснити бронювання.

В даній роботі буде розроблена своя структура для ефективного лендінгу.

## **Висновки до розділу 1**

Отже, в даній роботі розглядаються проблеми, з якими стикається розробник ПО при створенні лендінг-сторінки. Після появи односторінкових сайтів їх актуальність тільки зростає. Чому ж лендінги не втрачають своєї актуальності? В першу чергу

— компактність. Наше століття диктує нові умови для сайтів. Сучасну людину оточують тонни інформації — стає все складніше орієнтуватися та аналізувати її. З цієї причини багато веб-сайтів трансформуються і спрощуються, намагаючись досягти мінімалізму, щоб відвідувач не розмінювався на несуттєві деталі. Мінімалістичний сайт пропонує найкомфортніші умови: швидка розробка і редагування, реактивне завантаження сторінки — факти без перебільшень.

Лендінг-сторінка відіграє вирішальну роль у збільшенні впізнаваності компанії, підвищенні продажів, поліпшенні SEO показників і залученні нових клієнтів. Посадкова сторінка — ефективний спосіб розширити клієнтську базу. Це основні причини, чому слід використовувати цільову сторінку.

І на підставі потреб сучасного ринку було проведено аналітичне дослідження деяких статистичних даних, які демонструють переваги використання лендінг-сторінок для поліпшення маркетингових показників:

- середня конверсія для лендінгу - 9,7%;
- у формах на лендінгах сама висока конверсія - 23%;
- відповідь на страхи ЦА підвищує конверсію на 80%;
- персоналізовані СТА конвертують на 202% краще;
- 48% найкращих лендінгів приваблюють органічний трафік;
- 65% лендінгів включають бренд у назву;
- у 30% успішних лендінгів є відео;
- у 36% найкращих посадкових сторінках є відгуки.

Підбивати підсумки, лендінги важливі для того, щоб наголосити на конверсії, підвищити значущість сервісу, підкреслити вигоди для клієнта. Цільові сторінки відносно дешеві в налаштуванні, прості в обслуговуванні, і вони сприяють прямому бронюванню/замовленню/підписці і т.д.

## РОЗДІЛ 2

### АНАЛІЗ ОБРАНИХ ІНСТРУМЕНТІВ І ТЕХНОЛОГІЙ ДЛЯ ЛЕНДІНГУ

Написання хорошого коду за мінімальний час є найважливішою навичкою для кожного розробника програмного забезпечення. Тому що дуже важливо залишатися конкурентоспроможними та продуктивними в цифровий час. Тому під час написання коду важливо користуватися зручними та зрозумілими інструментами та технологіями. В даному розділі я хочу розповісти про обрані мови програмування та середовище розробки, котрими я користувалася під час створення проєкту.

#### 2.1. Середовище розробки Visual Studio Code

Visual Studio Code - це сервіс, який позиціюється як «легкий» редактор коду для кросплатформеної розробки веб- і хмарних додатків. Це надійний і невибагливий до ресурсів інструмент. Навколо нього зібралось потужна спільнота розробників розширень. На мій погляд, це одна з кращих розробок Microsoft.

Редактор допомагає в роботі веб-розробникам і верстальникам всіх рівнів. З одного боку він підходить новачкам, тому що його інтерфейс інтуїтивно простий і зрозумілий. З іншого боку в VS Code вбудовано багато можливостей, які цікаві досвідченим розробникам.

Можливості Visual Studio Code, освоєння яких допомагає прискорити роботу:

1. вбудовані інструменти інтеграції з GitHub, GIT, а також Visual Studio Team Services для швидкого тестування, складання та розгортання різних

Кафедра КІТ				НАУ 21 39 87 000 ПЗ			
Виконала	Перепада Д.А.			Аналіз обраних інструментів і технологій для лендінгу	Літера	Аркуш	Аркушів
Керівник	Зіатдінов Ю.К.					22	24
Консульт.					412 122		
Н-котрол.	Шевченко О.П.						
Зав. каф.	Савченко А.С.						

2. різних типів додатків;
3. зручність роботи з Unity-проектами;
4. робота з Mono і Node.js за допомогою вбудованого відладчика;
5. підтримка TypeScript і JavaScript;
6. публікація створених додатків в Microsoft Azure через сервіс Visual Studio Team Services;
7. підтримка практично всіх мов програмування;
8. написання коду для конкретного завдання з його подальшою інтеграцією в проект (з надбудовою або безпосередньо);
9. велика бібліотека шаблонів, готових фрагментів коду і сніпетів з можливістю додавання своїх елементів;
10. одночасна робота з декількома проектами (в декількох вікнах);
11. інтерфейс можна розділити на дві панелі для порівняння коду;
12. функція налагодження. [7]

Даний редактор підтримує Python, JavaScript, HTML, CSS, TypeScript, C ++, Java, PHP, Go, C #, PHP, SQL, Ruby, Objective-C і багато інших. Деякі з мов програмування, наприклад, JavaScript, CSS і HTML доступні одразу після встановлення додатку. Щоб використовувати інші мови програмування потрібно встановлювати плагіни — незалежно скомпільовані програмні модулі. Їх можна встановити за допомогою інтерфейсу програми Visual Studio Code або платформи VS Code Marketplace.

VS Code відрізняє від конкурентів наявністю вбудованого відладчика. Після невеликих налаштувань можна шукати баги в коді прямо з редактора — наприклад, поставити крапку зупинки та спостерігати за виконанням конкретної ділянки коду. Я активно користувалась даною функціональною можливістю під час розробки лендінг-сторінки. Крім цього у редактора є вбудована консоль, в яку може виводитися результат роботи або повідомлення про помилку, якщо щось пішло не так. Відладчик можна налаштувати під різні мови та різні завдання. В даному редакторі є всі функції, за допомогою яких можна написати, протестувати й успішно захистити проект.

Також хочу виділити зручний інтерфейс (див. рис.2.1.1) програми VS Code. Для мене ця зручність проявляється в одночасному відкритті незалежних вікон і панелей

та візуальному розділенню навпіл програмної області робочого вікна, чим я і користувалася під час розробки. В одній частині екрану я працювала з HTML-кодом, в іншій — з CSS.

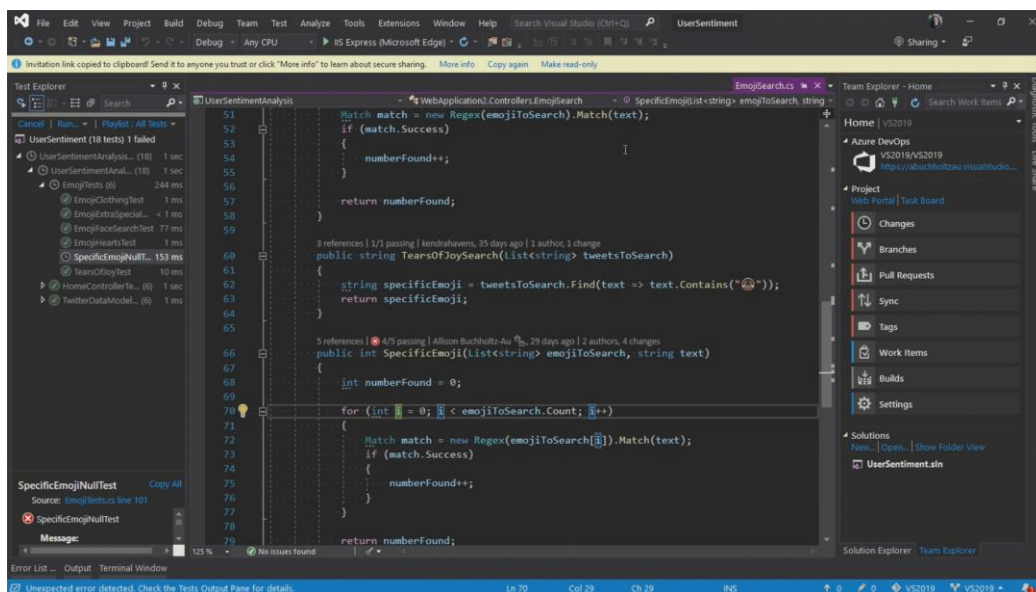


Рис.2.1.1. Приклад інтерфейсу VS Code

Якщо казати про кастомізацію, то як і будь-який інший популярний редактор, Visual Studio Code також забезпечує налаштування. Розмір і гарнітура шрифту, кольорова схема, розмір нових вікон, показ панелей інструментів, висновок підказок і повідомлень - все це і багато іншого може бути налаштоване з урахуванням уподобань користувача. Особливу увагу заслуговує вкладка «Terminal», що дозволяє працювати з консоллю PowerShell, командним рядком CMD або Linux Bash прямо у вікні редактора.

VS Code забезпечений великою кількістю розширень для розробника. Для установки нового пакета потрібно зайти у вкладку "Extensions", ввести назву пакета в рядку пошуку, натиснути кнопку "Install". Приклад встановлювання розширень VS Code зображено на рис. 2.1.2.



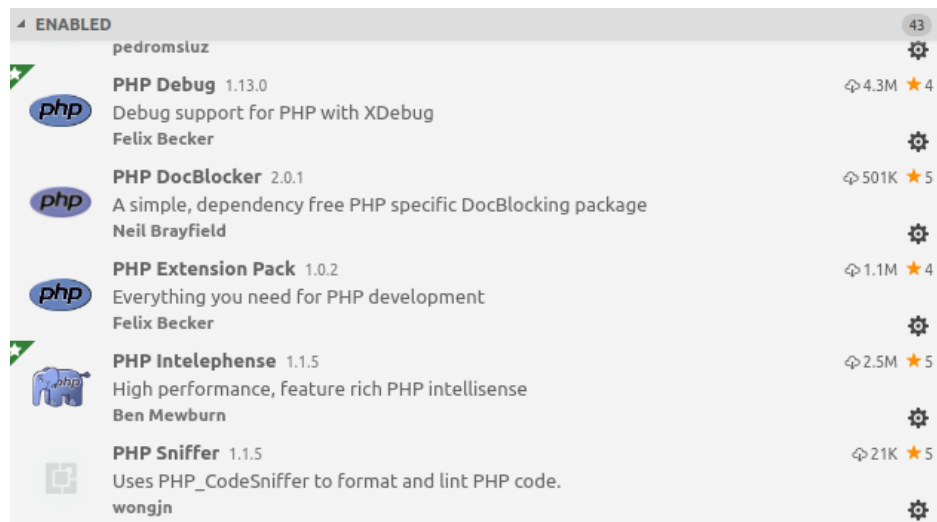


Рис. 2.1.2. Приклад встановлювання розширень VS Code

У підсумку можна виділити декілька основних переваг Visual Studio Code в порівнянні з іншими редакторами коду:

- велика кількість налаштувань (програми в цілому або окремо інтерфейсу);
- бібліотека з доповненнями і зручними готовими рішеннями, яка оновлюється;
- підтримка майже всіх мов, які використовуються для створення програм;
- простота, зручність та гнучкість.

## 2.2. Псевдомова HTML

HTML або мова розмітки гіпертексту - це мова розмітки для документів, призначених для відображення у веб-браузері. У поєднанні з іншими технологіями, такими як CSS та JavaScript, він створює переважну більшість вмісту, який можна побачити на веб-сайтах. HTML використовується для різноманітних речей в Інтернеті, починаючи від створення складних веб-сайтів, що пропонують функції електронної пошти та календаря, до створення простого веб-сайту курсу або резюме.

Стандарт HTML був розроблений W3C або Консорціумом Всесвітньої Павутини в 1997 році. З плином часу в HTML було багато оновлень, і в даний момент його

новою версією є HTML5. HTML5 перш за все є мовою розмітки, але він придбав безліч функцій на відміну від HTML і усунув деякі з суворих обмежень, які були присутні в XHTML. Найбільшою перевагою, яке має HTML5 над своїм нумерований попередником, є те, що у нього є підтримка аудіо і відео високого рівня, які не були частиною специфікації в попередніх HTML. В даній роботі використано мову розмітки гіпертекстових документів HTML5. [8]

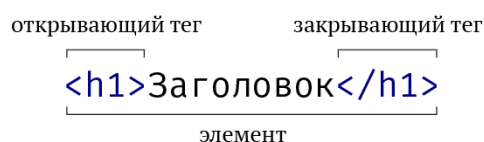
Конструкції HTML, швидше за все, будуть продовжувати використовуватись і в майбутньому, не дивлячись на те, що мова розвивається швидко і не збирається зупинятись. Завдяки тому, що, почавши вивчення HTML, я продовжила розвивати свої уміння і розширювати їх, наразі я можу створювати Web-сторінки, які будь-які користувачі можуть передивлятися в різних браузерах в мережі Інтернет.

Заголовок, параграфи або списки – це основні елементи, які застосовуються в більшості документів. Для того, щоб прочитати HTML-документ, потрібен лише WEB-браузер, який після інтерпретації тегів показує створений документ на екрані.

Для позначення різних елементів в HTML служать теги (tag). Тег HTML складається з:

1. лівої відкриваючої дужки <;
2. імені тега з параметрами або без них;
3. правої закриваючої дужки >.

Теги часто використовуються парами (наприклад, <h2> і </h2>), в яких перший тег задає початок елемента, другий - його закінчення. Завершальний тег має те ж ім'я, що і початковий, але перед ім'ям ставиться коса риска /. Завершальні теги не мають параметрів. Деякі елементи можуть включати атрибути, що застосовуються як параметри в початкових тегах. Нижче наведено приклад написання елемента, дивисись рис. 2.2.1.



```
открывающий тег      закрывающий тег
┌──────────┐ ┌──────────┐
<h1>Заголовок</h1>
└──────────┘
элемент
```

Рис. 2.2.1. Приклад написання елемента HTML

Не всі теги підтримуються всіма браузерами. Якщо який-небудь браузер не підтримує тег, то він його просто ігнорує.

Найпростіша HTML-сторінка складається як мінімум з трьох ключових слів: `<html>`, `<head>` і `<body>`.

Тег `<html>` розташовується в документі відразу після доктайпів і містить всі інші теги, включаючи `<head>` і `<body>`.

Тег `<head>` зберігає важливу службову інформацію, а в тезі `<body>` зберігається зміст сторінки, який відображається у вікні браузера. Всі тексти і картинки додаються всередину `<body>`. У документі теги `head` і `body` можуть бути використані тільки один раз.

Елемент `<title>` містить заголовок документа і ідентифікує його вміст в глобальному контексті. У своєму коді в середину даного елемента я додала елемент `<meta>`, що дає різну метайнформацію (наприклад, дату закінчення терміну документа) і елемент `<link>`, який задає зв'язки з іншими документами.

Поняття абзацу в HTML-документі також аналогічно поняттю абзацу в Microsoft Word. Абзац позначається в документі парними тегами `<P>` і `</ P>`. Втім, застосування закриваючого тега не є обов'язковим. Але специфіка тега `<P>` полягає в тому, що після тексту, який знаходиться в його межах, порожній рядок додається автоматично.

Завдяки HTML5 розробники отримують більше гнучкості, коли створюють дизайн сайтів. Завдяки наступним аспектам робота користувачів змінилась в кращу сторону:

- постійна обробка помилок;
- покращена семантика для елементів;
- розширена підтримка функцій веб-додатків;
- налаштування атрибутів даних.

Треба відзначити, що HTML-документи записуються в ASCII-форматі, і тому для написання таких файлів розробники можуть використовувати будь-який текстовий редактор.

## 2.3. Псевдомова CSS

CSS - це каскадні таблиці стилів. Ця мова розмітки визначає, як HTML-елементи веб-сайту повинні відображатися в інтерфейсі сторінки. CSS був запропонований Хоконом Віум Лі 10 жовтня 1994 року і підтримується через групу людей з W3C, званою робочою групою CSS.

У той час як HTML є основною структурою сайту (landing-сторінці), CSS - це те, що задає всьому сайту стиль. Кольори, цікаві шрифти і фонові зображення - все це заслуга CSS. Ця мова впливає на весь настрій веб-сторінки, що робить його неймовірно потужним інструментом і важливим умінням для веб-розробників. Він також дозволяє веб-сайтам адаптуватися до різних розмірів екрану і типів пристроїв.

Шляхом збереження стилів в окремому файлі, який кешується і при повторному зверненні до нього витягується з кешу браузера, відбувається прискорення завантаження сайту.

Хочу розглянути переваги CSS, які були дуже корисними при написанні коду для landing-сторінки готелю:

- CSS економить час. Написавши CSS один раз, можна використовувати одну і ту ж таблицю на декількох HTML-сторінках. Можна визначити стиль для кожного HTML-елемента і застосувати його до багатьох веб-сторінках.
- Сторінки завантажуються швидше. Написавши одне CSS правило для тега, його можна застосуєте його до всіх входженням цього тега..
- Простота обслуговування. Щоб внести глобальні зміни, я змінювала просто стиль, і всі елементи на всіх веб-сторінках оновлюються автоматично.
- Покращені стилі для HTML. CSS має набагато ширший набір атрибутів, ніж HTML, тому я мала змогу зробити набагато кращий вигляд своєї HTML-сторінки в порівнянні з атрибутами HTML.
- Сумісність декількох пристроїв. Таблиці стилів дозволяють оптимізувати контент для більш ніж одного типу пристроїв.

- Офлайн-перегляд. Веб-додатки можуть зберігати CSS локально за допомогою оффлайн кешу. Використовуючи це, я могла переглядати сайти перебуваючи оффлайн. Кеш також забезпечує швидке завантаження і кращу загальну продуктивність веб-сайту.

- Незалежність від платформи. Скрипт забезпечує незалежність від платформи і підтримує новітні браузери.

Розглянути порівняно невеликий набір головних властивостей CSS3 можна на рисунку 2.3.1.






























СВОЙСТВО	ПОДДЕРЖКА
border-radius	 3+  3+  1+  10.5+  9 beta
text-shadow	 1.1+  2+  3.1+  9.5+
box-shadow	 3+  3+  3.5+  10.5+  9 beta
Multiple background images — несколько фоновых изображений	 1.3+  2+  3.6+  10.5+  9 beta
opacity	 1.2+  1+  1.5+  9+  9 beta
RGBA	 3.2+  3+  3+  10+  9 beta

Рис. 2.3.1. Властивості CSS і їх підтримка браузерами

Розглянемо правила оголошень, які є дуже простими. На початку пишеться селектор, який позначає конкретний елемент на сторінці, після фігурних дужок вказуються властивості і їх значення після двокрапки, а самі властивості відмежовуються один від одного крапкою з комою. Селектори – це найскладніше в CSS оголошенні. Вони показують шлях до тега, в якому міститься потрібна інформація. В якості селектора, зазвичай, використовується певний клас тега, приклад зображено на рис. 2.3.2:

```
1 //HTML:
2 <div class="my-class"></div>
3
4 //CSS:
5 .my-class {
6   background-color: #999;
7 }
```

Рис. 2.3.2. Приклад оголошення CSS-властивостей

Тут селектором виступає клас `my-class` тега `div`, який отримує необхідне оформлення в CSS файлі. В даному випадку фоновий колір - сірий. Відповідно, якщо на сторінці є кілька тегів (не тільки `div`) з класом `my-class`, всі ці елементи отримають однакове оформлення - сірий фон кольору `# 999`.

Використовувати для вказівки на конкретний тег або його вміст можна практично все елементи `html`. Назва тега - досить просто вказати `span`, `div`, `img`, `h2` в місці, де потрібно прописати селектор, і такий селектор буде враховувати всі теги з такою назвою на сторінці.

Отже, CSS - це найсильніший інструмент і один з основних складових практично будь-якого `web-сайту`. Без CSS просто неможливо уявити сучасну діяльність `web-розробників`. Сьогодні CSS - це загальноприйнятий стандарт розробки, який приймається усіма без винятку компаніями-розробниками, що явно показує його значимість і необхідність.

## 2.4. Мова програмування JavaScript

JavaScript є більш складною мовою, ніж HTML або CSS, і він не був випущений в бета-версії до 1995 року. В даний час JavaScript підтримується всіма сучасними веб-браузерами і використовується практично на кожному сайті в Інтернеті для більш потужних і складних функцій.

JavaScript - це логічна мова програмування, яка може використовуватися для зміни вмісту `web-сайту` і змусити його поводитися по-різному у відповідь на дії користувача. Загальне використання JavaScript включає в себе вікна підтвердження, заклики до дії і додавання нових ідентифікаторів до існуючої інформації.

JavaScript популярний не випадково, а завдяки своїм безперечним перевагам:

1. Незамінність для веб-розробки. Підтримка скриптів усіма популярними браузерами; повна інтеграція з версткою сторінок (HTML + CSS) і серверної частиною (backend);

2. Швидкість роботи і продуктивність. JavaScript дозволяє частково обробляти веб-сторінки на комп'ютерах користувача без запитів до сервера. Це економить час і трафік, знижує навантаження на сервер.

3. Потужна інфраструктура (екосистема). Перші 10 років цього не було, але потім кількість готових рішень в відкритому доступі так зросла, що працювати з JavaScript і його фреймворками стало задоволенням;

4. Простота і раціональність застосування. Просту завдання можна вирішити за 5 хвилин, не треба робити зайву роботу. Для складних завдань є варіанти вирішення, можна підібрати кращий, адаптувати.

5. Зручність для користувача інтерфейсів. Заповнення форм, вибір дій, активація кнопок, перевірки введення, реагування на наведення / кліки миші і т.п Це дає приголомшливий рівень юзабіліті.

6. Легкість освоєння. Можливо, це справа звички. Спочатку код може здатися складним, але до його синтаксису, логіки швидко звикаєш. Відображення дій відчутно додає ентузіазму.

Код на мові JavaScript називають скриптом. Його зберігають в окремий файл з розширенням js, а щоб запустити, підключають цей файл на сторінку. У HTML для додавання JavaScript є спеціальний тег (див. рис. 2.4.1).

```
<script src="адрес_файла"></script>
```

Рис. 2.4.1. Приклад додавання JavaScript

Підключають скрипт зазвичай в самому кінці сторінки, перед закриває тегом `</body>`.

Програма на JavaScript - це послідовність інструкцій, тобто вказівок браузеру виконати якісь дії. Інструкції виконуються послідовно, зверху вниз.

Щоб сказати JavaScript, що інструкція закінчена, потрібно поставити крапку з комою або перейти на новий рядок. Новий рядок правильно працює в більшості випадків, а крапка з комою - завжди. Тому краще ставити крапку з комою в кінці кожної інструкції. JavaScript не змінює вихідний файл з розміткою, але, виконуючи інструкції, змінює сторінку прямо в браузері користувача.

Консоль - інструмент розробника, який допомагає тестувати код. Якщо під час виконання скрипта виникне помилка, у консолі з'явиться повідомлення про неї. А ще в консоль можна виводити текстові підказки. Щоб вивести повідомлення в консоль, потрібно використовувати `console.log` (див. 2.4.2).

```
console.log('Привет от JavaScript!');  
// Выведет: Привет от JavaScript!  
  
console.log(document.querySelector('.page'));  
// Выведет в консоль найденный элемент
```

Рис. 2.4.2. Приклад використання `console.log`

Змінна - спосіб зберегти дані, давши їм зрозумілу назву. Змінну можна створити, або оголосити, за допомогою ключового слова `let`. За ним слідує ім'я змінної. Після оголошення в змінну потрібно записати, або привласнити, якесь значення. Приклад оголошення зображено на рисунку 2.4.3.

```
let header = document.querySelector('header');
```

Рис. 2.4.3. Приклад оголошення змінної



Ім'я змінної може бути майже будь-яким, але не повинно починатися з цифри, а з спецсимволів дозволені тільки '\_' і '\$'. Ще для іменування змінних можна використовувати зарезервовані слова. Імена змінних чутливі до регістру: header, Header і HEADER - це різні змінні. Ім'я змінної має описувати те, що в ній зберігається. Коли в коді зустрічається змінна, браузер замість її імені підставляє присвоєне їй значення.

В браузері JavaScript може робити все, що стосується маніпуляцій з вебсторінками, взаємодії з користувачем та веб-сервером. Наприклад, за допомогою JavaScript можна робити наступне у веб-браузері:

1. Додати новий HTML на сторінку, змінити існуючий вміст, змінити стилі;
2. Реагувати на дії користувача (натискання миші, рух вказівника, натискання клавіш);
3. Надсилати запити по мережі на віддалені сервери та завантажувати файли.
4. Отримати та встановити файли cookie, задавати питання відвідувачеві, показувати повідомлення.
5. Запам'ятовувати дані на стороні клієнта («локальне зберігання»).

## **2.5. Мова програмування PHP**

Правильно сказати, що PHP - це мова сценаріїв з відкритим вихідним кодом, яка широко використовується і має загальне призначення. Коротше кажучи, PHP - це мова програмування, що створена для написання web-додатків.

Значення самого позначення PHP наступне: "Hypertext Preprocessor (Препроцесор Гіпертексту)". C, Java і Perl – це мови, які є основою синтаксису PHP. Але великою перевагою перед цими мовами є можливість створення HTML документів, що впроваджують команди PHP.

Особисто мене чіпляє простота вивчення даної мови і її головна перевага - швидке створення динамічно генерування web-сторінок.

Те, що PHP - частково об'єктно-орієнтована мова, дозволяє використовувати код декілька разів. Цей факт бережи сили розробника і економить його час. Давайте

розглянемо фреймворки PHP, до них належать: Symfony, CodeIgniter, Laravel, Joomla, WordPress та ін. Так як для бізнесу дуже важливий час і витрати, то наявність вбудованих функцій, таких як запити GET і POST, скорочує ці показники і тому інвестиції окупаються швидше.

На мою думку, найбільша перевага PHP - це гнучкість і універсальність, так як він сумісний з усіма основними платформами: від Windows, Unix і Linux до MacOS; підтримує більшість серверів (включаючи Apache, Microsoft IIS, Netscape, iPlanet, Caudium, Xitami і Tornado) і більше 20 баз даних (серед яких - MySQL, MongoDB, PostgreSQL та ін.).

Також хочеться відзначити, що змінювати код на PHP зовсім не важко завдяки хорошій читабельності. Як наслідок - менше проблем з підтримкою і оновленнями.

PHP популярний серед хостинг-провайдерів завдяки своїй універсальності, тому що PHP підтримується будь-яким надійним провайдером і надає хостингові послуги для сайтів.

Для утримання клієнтів дуже важливе швидке завантаження. Люди сьогодні в змозі утримувати увагу протягом 6-8 секунд. Тому сайт закривають ще до закінчення завантаження, якщо сайт завантажується довше. Оскільки PHP має хорошу кастомізацію та інтеграції з різноманітними системами, що управляють контентом, то швидкість завантаження буде високою.

Важливий факт, що PHP можна застосовувати для створення будь-яких ІТ-проектів, частіше, у парі з іншими технологіями: починаючи з простих веб-сайтів до CRM-рішень і великих маркетплейсов.

## **Висновки до розділу 2**

В даному розділі було розглянуто технології, які будуть використовуватись при розробці Landing page для готелю. Було описано технології HTML, CSS, JavaScript та мова програмування PHP. Також було розглянуто обране середовище розробки для створення лендінгу, його переваги та недоліки. Обраний редактор коду – Visual Studio

Code я завжди використовую при розробці сайтів, адже він дуже зручний та наповнений потрібним функціоналом для вдалого і швидкого написання коду. Дуже важливо правильно обрати технології та інструменти, щоб у майбутньому не виникало складнощів з розвитком і оновленням проекту.

## РОЗДІЛ 3

### СТВОРЕННЯ LANDING PAGE ДЛЯ ЗАКЛАДУ ГОТЕЛЬНОГО БІЗНЕСУ

#### 3.1. Діаграма станів

Діаграми - невід'ємна складова процесу моделювання будь-яких систем. В даній роботі ми не створюємо баз даних, проте для розуміння роботи системи нашого лендінгу діаграми будуть допоміжними. Також вони можуть бути застосовані для подальшого створення бази даних для даного сайту і його розвитку в майбутньому.

Майже кожна діаграма станів показує всі реальні стани одного примірника конкретного класу і деякі ланцюжки його форсування з одного стану в зовсім інше. Інакше кажучи, моделює всі зміни структури об'єкта як його вплив на зовнішню роботу. Дана діаграма, що описує процес роботи системи в цілому, представлена на рисунку 3.1.1.

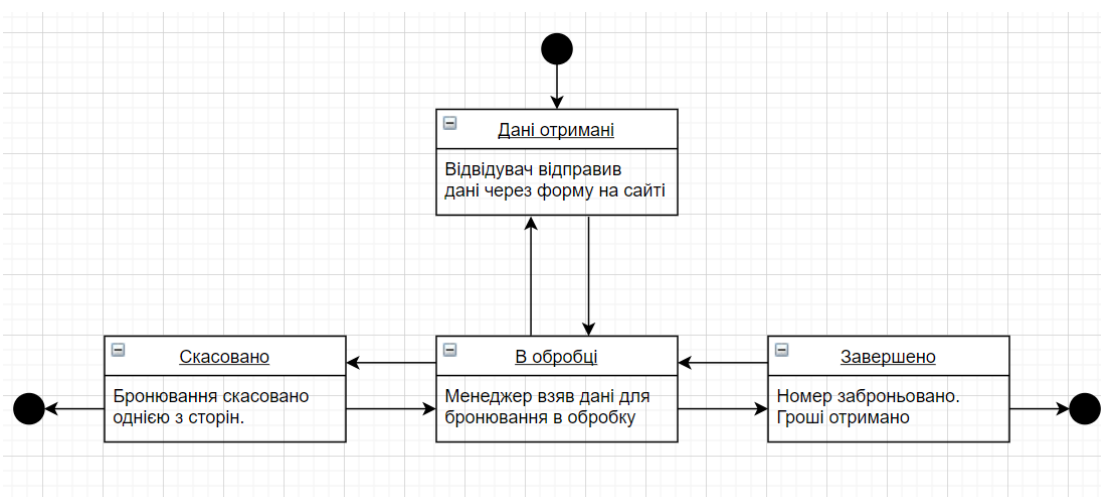


Рис. 3.1.1 Діаграма станів

Кафедра КІТ				НАУ 21 39 87 000 ПЗ			
Виконала	Перепадя Д.А.			Створення лендінг-сторінки для закладу готельного бізнесу	Літера	Аркуш	Аркушів
Керівник	Зіатдінов Ю.К.					36	24
Консульт.					412 122		
Н-котрол.	Шевченко О.П.						
Зав. каф.	Савченко А.С.						

### 3.2. Контекстна діаграма

Контекстна діаграма - це модель, яка показує систему як набір деяких ієрархічних послідовностей і дій, в яких кожен крок перетворює якийсь об'єкт або їх набір. Вищою дією ієрархії є дія контексту - це найвищий рівень, безпосередньо описує систему. Нижні рівні називаються породженими декомпозиції. Вони представляють підпроцеси батьківської дії. Контекстна діаграма описує зовнішні для системи об'єкти, які взаємодіють з нею, але нічого не відображає. Ні внутрішню структуру, ні поведінку системи.

Контекстна діаграма представлена на рисунку 3.2.1.

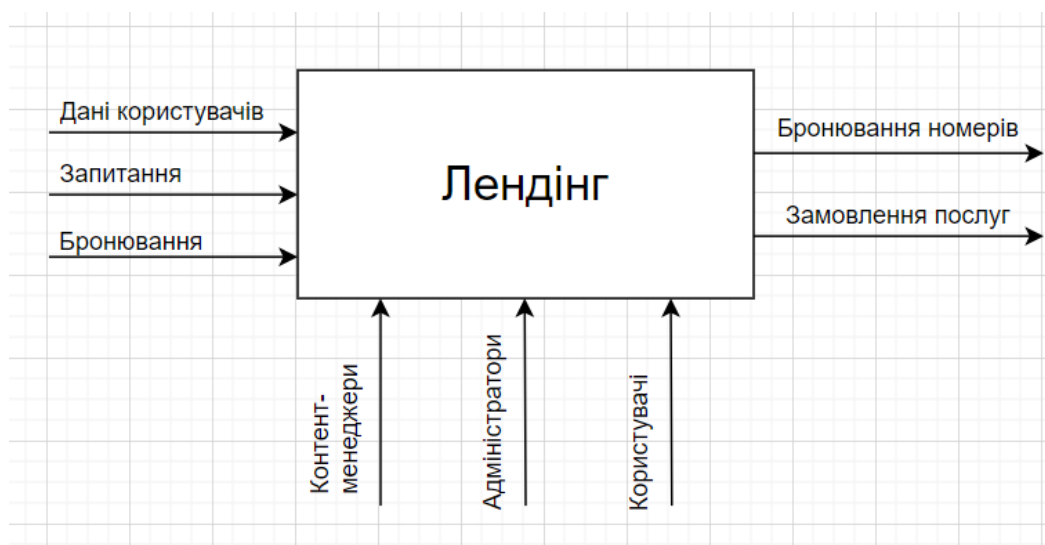


Рис. 3.2.1. Контекстна діаграма

### 3.3. Структура та основні функції зрозумілого та зручного інтерфейсу

В епоху інформаційних систем і технологій швидко почали поширюватися Web - інтерфейси, які дозволяють працювати з різними системами через звичайний браузер.

Web-інтерфейси дуже зручні у використанні через можливість вести спільну роботу з клієнтами, навіть не перебуваючи в одному місці. Невід'ємною частиною будь-якої системи є її інтерфейс. Інтерфейс - це певна грань між працюючими один з

другом об'єктами. Інтерфейс сам задає критерії, процедури і параметри характеристик взаємодії елементів.

Інтерфейс користувача в свою чергу – це елементи та компоненти системи, здатні впливати на роботу клієнта з ПО. В тому числі:

- елементи інформації, що відображається;
- мови користувач-інтерфейс;
- системи і елементи введення різних даних;
- обов'язковий зворотний зв'язок з клієнтом;
- допомога у вирішенні конкретної предметної області;
- інформаційна складова для ознайомлення з продуктом.

Розглядаючи нашу лендінг-сторінку з технічної точки зору, ми не побачимо великої різниці порівняно із іншими сайтами. При його створенні також використовується HTML, CSS, JavaScript і контент. Однак фішка сайту в тому, що цей сайт завжди є односторінковим.

Для посадкової сторінки готелю важливо, щоб вся інформація подавалася логічно, зрозуміло і не перевантажувала увагу користувача. Тому лендінг повинен розроблятися по структурі, яка продумана заздалегідь.

Зазвичай Landing page будують так:

- короткий оффер;
- подробиці і переваги пропозиції;
- варіанти цін із закличком до дії;
- відгуки.

Це базова структура, але під конкретні завдання і проаналізовану цільову аудиторію можна змінювати блоки та їх зміст. Наприклад, якщо продукт ще невідомий - доведеться розповідати про нього докладніше і давати більше варіантів застосування. А якщо відомий - зробити акцент на якість, ціну, сервіс або відгуки.

З точки зору моделі поведінкових факторів, неможливо визначити конкретну послідовність розташування блоків. Тому що це залежить виключно від виду діяльності компанії, що продає послугу або продукт, і їх цільової аудиторії.

Однак жодна структура Landing page не позбавлена деяких блоків, які повинні бути включені в неї. Такі блоки мотивують відвідувача лендінгу стати клієнтом і підводять до цільового дії.

Для сторінки готельного бізнесу була розроблена своя структура написання сайту (див. рис. 3.3.1).



Рис.3.3.1. Структура лендінгу

Почнемо по порядку:

У шапці разом з логотипом готелю розміщуємо гамбургер-меню з посиланнями на розділи сторінки, щоб відвідувачеві було зручніше орієнтуватися в просторі лендінгу. Також додаємо номери телефону для зв'язку. Зазвичай їх розміщують у правому верхньому куті.

Далі розміщуємо блок із назвою готелю, його описом та фото у вигляді слайдів. Опис готелю - коротка інформація про кількість номерів, наявності ресторанів, конференц-залів і так далі. І найголовніше – кнопка бронювання номеру, при натисканні якої користувача переадресовує на форму із заповненням контактних даних для бронювання номеру.

Після опису готелю розмістимо блок, який демонструє переваги цього готелю, наприклад, безкоштовний wifi, парковка, охорона, розташування поруч з метро.

Далі за допомогою HTML, JavaScript та CSS-стилів додаємо відео-презентацію, використовуючи заставку відео та кнопки play/stop.

Продовжуючи розробку структури (див. рис. 3.3.2), я вирішила додати необхідний для успішного лендінгу розділ з відгуками. Відвідувачу завжди цікаво дізнатися інформацію про готель.



Рис. 3.3.2. Структура лендінгу

Форма бронювання номера. Форма не повинна бути переповнена непотрібними полями, при заповненні яких у відвідувача закінчиться терпіння і він закриє сторінку. Тільки основні моменти (запитуємо лише інформацію, необхідну для оформлення броні): поле з ім'ям, номером телефону, електронною поштою, датами заїзду/виїзду та кількістю гостей.



Номери. По-перше, опис кожної категорії номерів - чим детальніше, тим краще. По-друге, наявність якісних фотографій кожного номера з різних ракурсів. Під кожним номером розмістимо його вартість.

Поруч із зазначенням ціни необхідно розміщувати кнопку зв'язку з готелем, після натискання якої користувач може залишити свій номер телефону для зворотнього зв'язку.

Розділ з послугами готелю. Опис додаткових послуг, які надають в готелі. Наприклад, наявність салонів краси, магазинів, надання особистого шофера, трансферу, можливість розміщення з тваринами. Все те, що може зацікавити відвідувача.

У футері розмістимо посилання на соціальні мережі та логотип.

Не забуваємо про адаптивність лендінгу під мобільні пристрої. Якісне відображення сайту на смартфонах і планшетах дозволяє користувачам без зусиль знайти цікаву для них інформацію і забронювати номер.

Таким чином ми перерахували всі елементи структури лендінгу, які варто враховувати при його створенні.

### **3.4. Сценарій роботи користувача з системою**

Як тільки користувач переходить, наприклад, за посиланням, яке він отримав від свого колеги або просто знайшов в інтернеті, перше, що він побачить після завантаження - це «шапка» Лендінгу, яка представлена на рисунку 3.4.1.

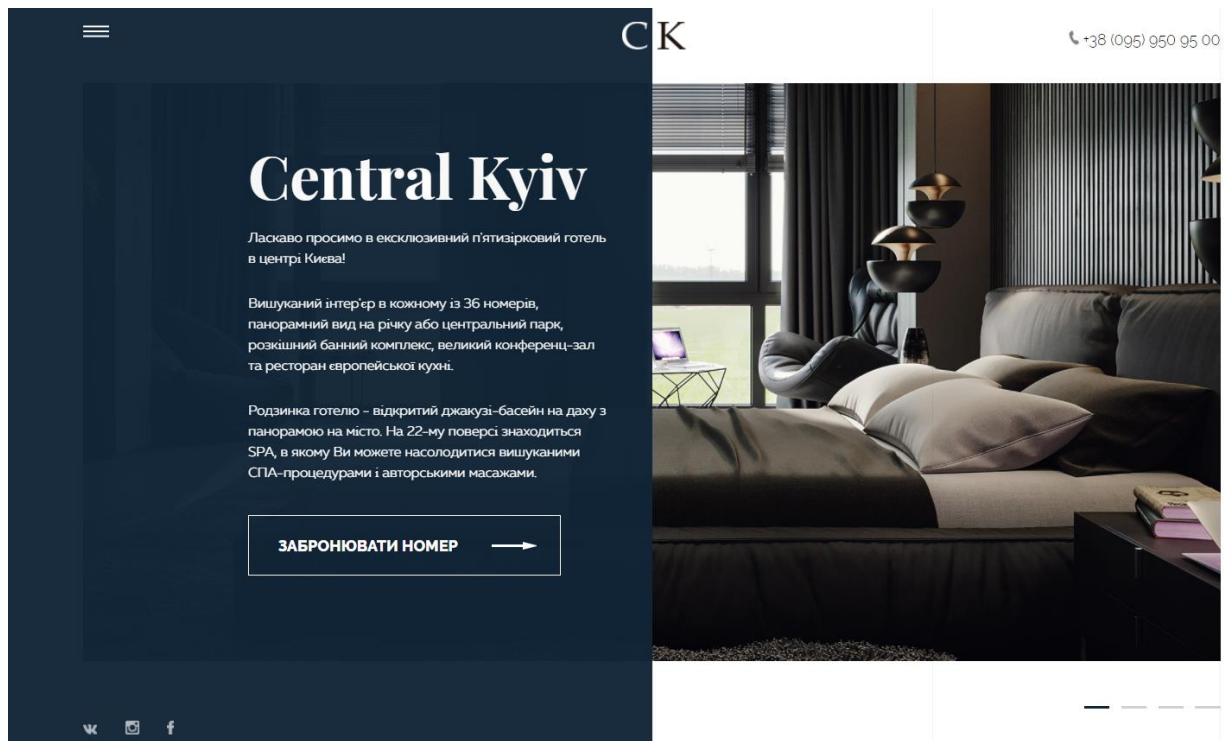


Рис. 3.4.1. «Шапка» лендінгу

У ній міститься практично вся необхідна інформація для того, щоб успішно представити продукт:

1. контактний номер телефону;
2. фірмовий логотип готелю;
3. візуальне представлення готелю з описом, основними перевагами;
4. поле з посиланнями на соціальні мережі;
5. гамбургер-меню для навігації по сторінці.

Візуальне представлення розроблено за допомогою слайдів, реалізація яких зображена на наступному малюнку.

Натискаючи на меню у лівому верхньому куті (див. рис. 3.4.1.), користувач отримує перелік блоків лендінгу, на які він може перейти, натиснувши на необхідну назву.

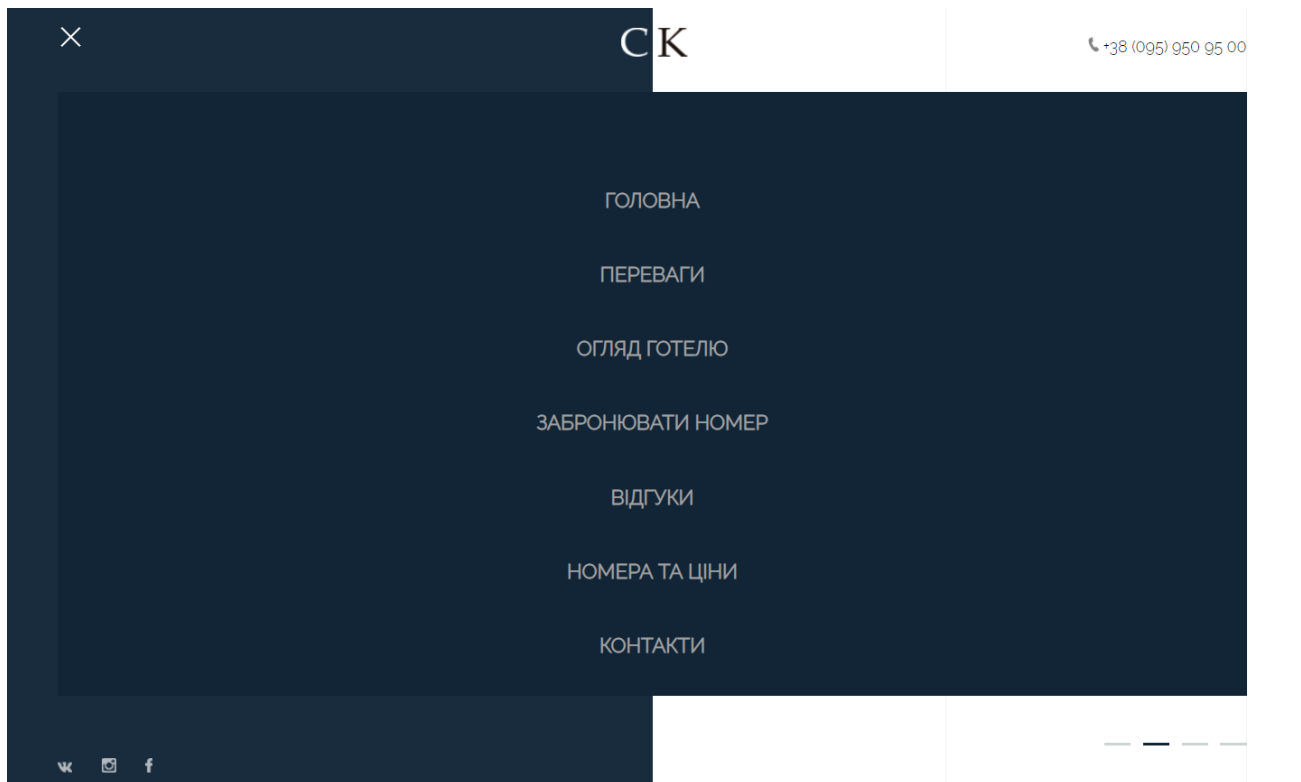


Рис. 3.4.2. Меню лендінгу

Коли користувач проскролить нижче, то опиниться у блоці з описаними перевагами готелю (див. рис. 3.4.3).

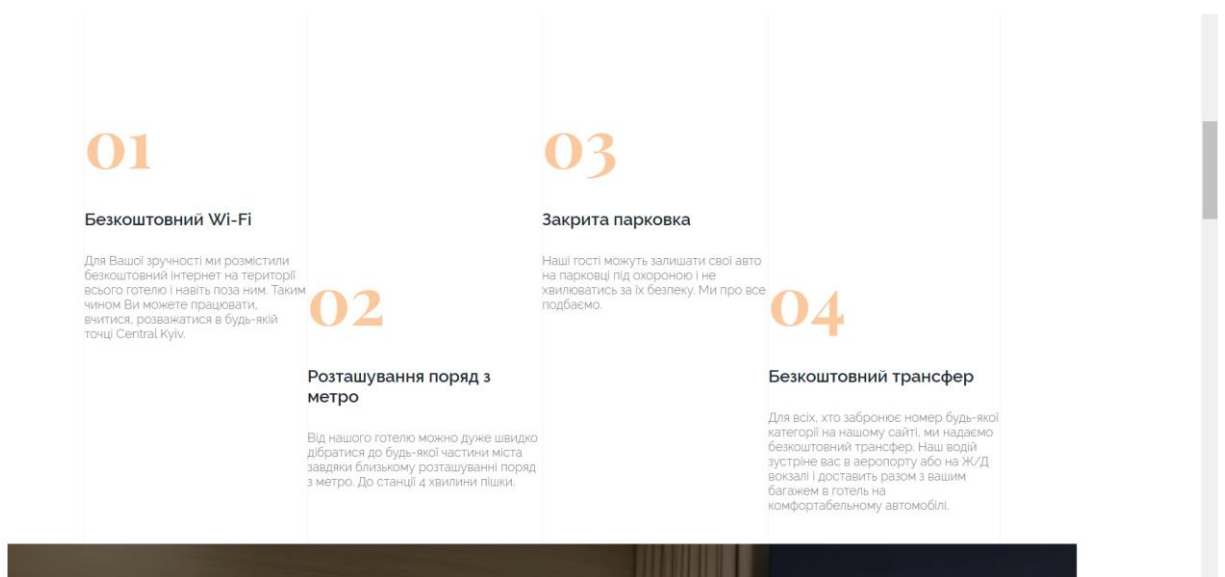


Рис. 3.4.3. Блок переваг готелю

У разі переходу за посиланням «Огляд готелю» користувача перенаправить в розділ, в якому знаходиться відео-презентація. Почати перегляд можна після натискання кнопки «Play».

Реалізація відео-презентації представлена на рис. 3.4.4.

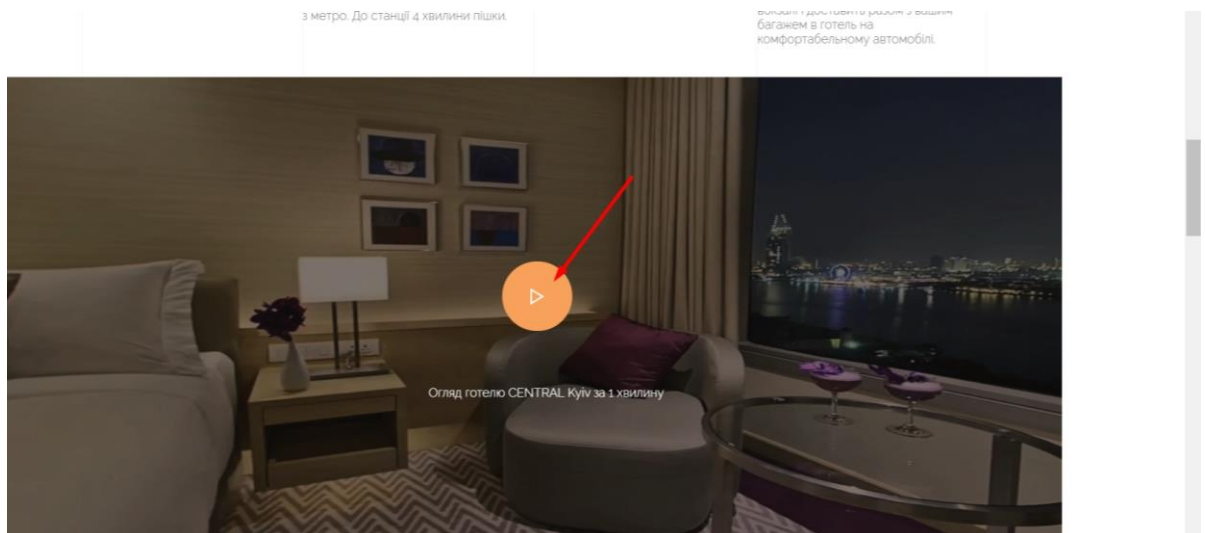


Рис. 3.4.4. Відео-презентація

Також я створила можливість для відвідувача зупинити перегляд відео і відновити знов, а також він може вмикати або вимикати звук відео за допомогою спеціально створених кнопок. Дана можливість представлена нижче (рис. 3.4.5):



Рис. 3.4.5. Кнопки запинки відео та звуку

Переходячи в розділ «Відгуки» (рис. 3.4.6), користувач має можливість ознайомитися із залишеними відгуками гостей, які вже були в даному готелі. Він може дізнатися ім'я гостя, його вік, місце, звідки він приїхав, і текст враження після перебування.



Рис. 3.4.6. Розділ з відгуками

І найголовніший етап – форма контактних даних. Після блоку з відгуками відвідувач опиняється у розділі, де він може залишити контактні дані для бронювання номеру. Майбутній клієнт має залишити своє ім'я, номер телефону та електронну пошту для зв'язку. Натиснувши кнопку «Забронювати», клієнт погоджується з умовами обробки персональних даних та відправляє їх на пошту готелю, де дані будуть отримані менеджерами і надалі оброблені в потрібному порядку. Поряд з самою формою я додала зображення готелю для утримання клієнта та повторного привертання уваги. Результат створення форми представлено на рисунку 3.4.7.

**Хочеш забронювати номер?**

Залиште контактні дані, менеджер зв'яжеться з Вами на протязі 15 хвилин для уточнення деталей.

Имя \_\_\_\_\_

Телефон \_\_\_\_\_

E-mail \_\_\_\_\_

**Забронювати**

\* натискаючи на кнопку, Ви погоджуєтесь з умовами обробки персональних даних.

Рис. 3.4.7. Форма контактних даних

Так виглядає правильне заповнення даних (рис.3.4.8):

**Хочеш забронювати номер?**

Залиште контактні дані, менеджер зв'яжеться з Вами на протязі 15 хвилин для уточнення деталей.

Имя  
Дар'я \_\_\_\_\_

Телефон  
09818181818 \_\_\_\_\_

E-mail  
nau@gmail.com \_\_\_\_\_

**Забронювати**

\* натискаючи на кнопку, Ви погоджуєтесь з умовами обробки персональних даних.

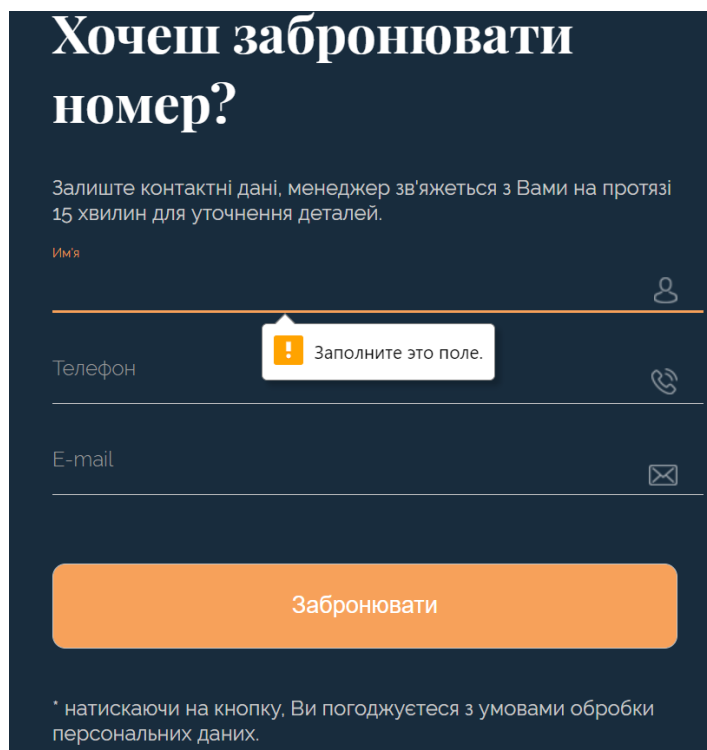
Рис.3.4.8. Успішне заповнення форми

Користувачеві необхідно заповнити поля:

- ім'я;
- номер телефону;

- електронна пошта.

В іншому випадку, якщо якесь поле буде не заповнене, нижче висвітяться повідомлення «Заповніть це поле». Малюнок неправильного заповнення форми представлений на рис. 3.4.9.



The image shows a dark-themed web form titled "Хочеш забронювати номер?". Below the title, there is a sub-header: "Залиште контактні дані, менеджер зв'яжеться з Вами на протязі 15 хвилин для уточнення деталей." The form contains three input fields: "Имя" (Name), "Телефон" (Phone), and "E-mail". The "Телефон" field is highlighted with a white error message box that says "Заполните это поле." (Fill in this field). Below the fields is a large orange button labeled "Забронювати". At the bottom, there is a small asterisk and text: "\* натискаючи на кнопку, Ви погоджуєтеся з умовами обробки персональних даних."

Рис. 3.4.9. Неправильне заповнення форми

У разі успішного заповнення форми і після відправки даних на пошту, користувач отримує повідомлення, що дані успішно відправлено (див. рис. 3.4.10).

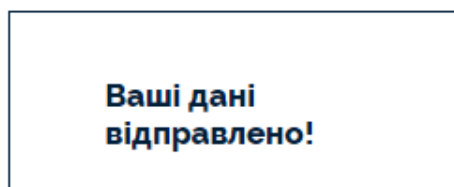


Рис. 3.4.10. Повідомлення про успішну відправку



При переході у розділ «Наші номери» (рис. 3.4.11), користувач може ознайомитися із номерами готелю. А саме: їх назвою, можливою кількістю людей у номері, ціною номеру, його описом та зображеннями.

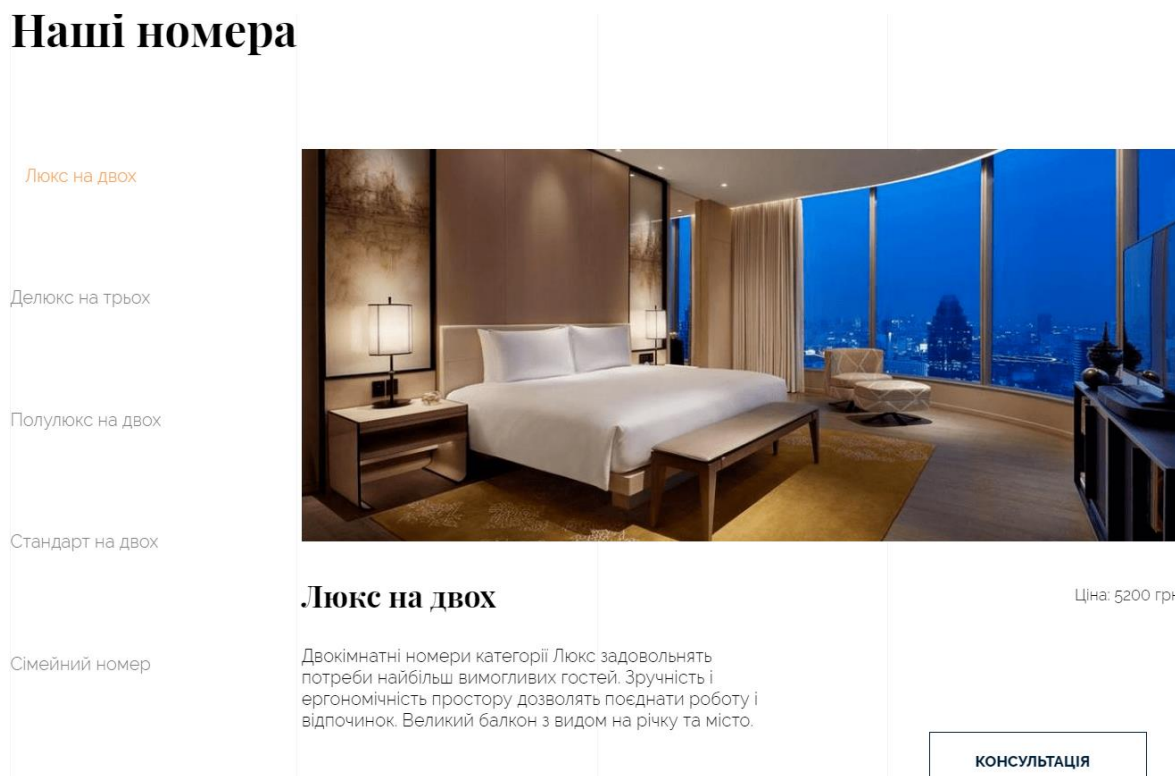


Рис. 3.4.11. Розділ з номерами готелю

Майбутній клієнт має змогу обирати будь-який номер із списку представлених та дізнаватися потрібну інформацію.

У випадку, якщо в нього з'являться питання, він може натиснути на кнопку «Консультація» та подзвонити за вказаним номером (див. рис. 3.4.12).

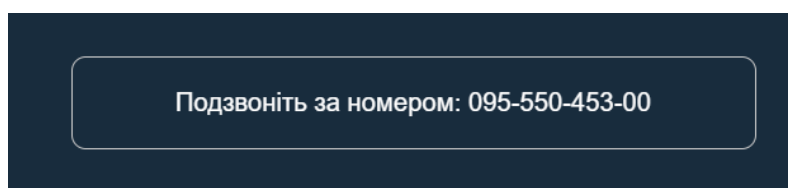


Рис. 3.4.12. Форма з номером телефону

В кінці лендінгу клієнт потрапляє на футер (рис. 3.4.13), в якому розміщено назва та логотип готелю, посилання на соціальні мережі та ім'я розробника сайту.



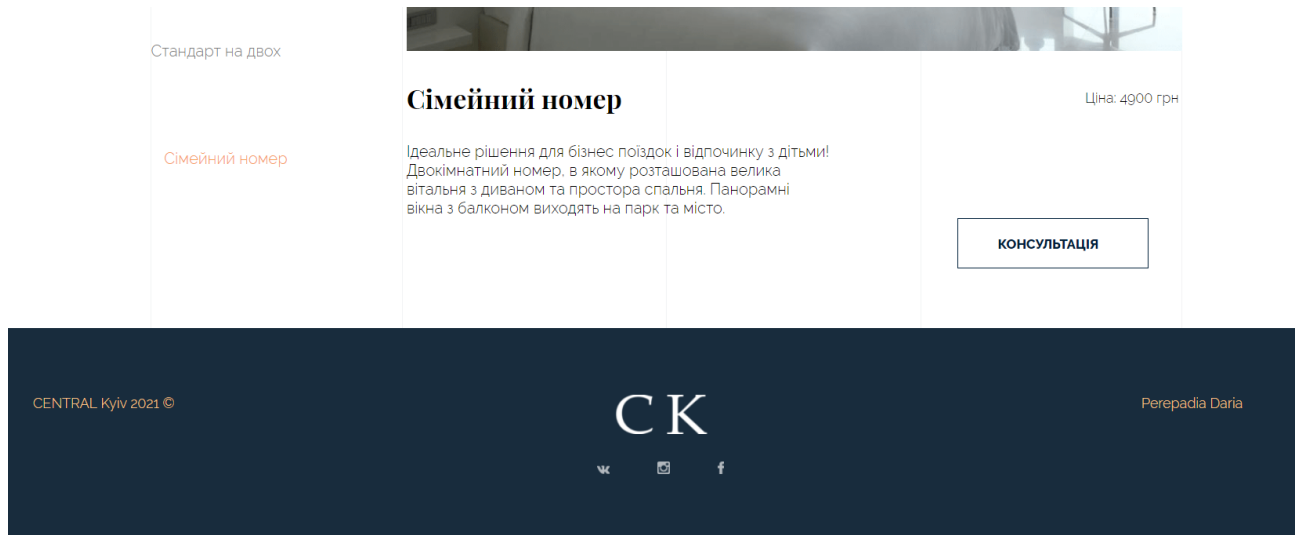


Рис. 3.4.13. Футер лендінгу

### 3.5. Реалізація проекту

Запускаємо редактор Visual Studio Code та додаємо папку (див. рис. 3.5.1):

1. Add workspace folder - відкриваємо меню вибору папки.
2. Створимо нову папку в будь-якому зручному місці і заїдемо в неї.
3. Натиснемо Add.

Після цього зліва з'явиться панель Explorer.

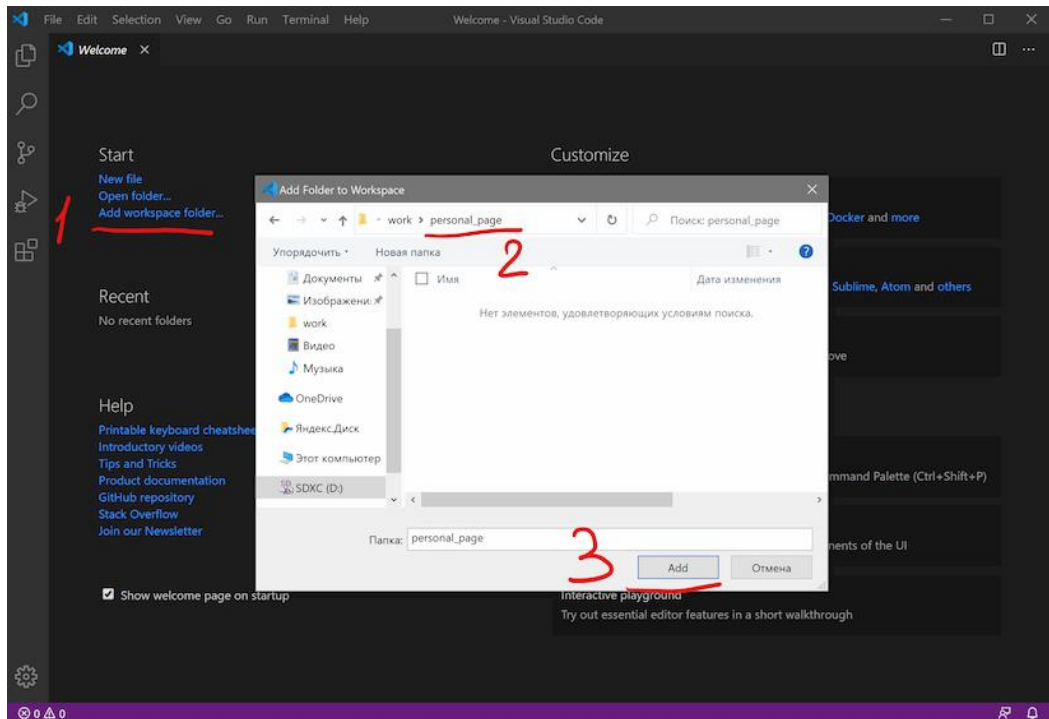


Рис. 3.5.1. Створення папки

Після створення папка порожня. Клацнемо правою кнопкою по заголовку і додамо три файли (рис. 3.5.2), які знадобляться в роботі - `index.html`, `style.css` і `script.js`.

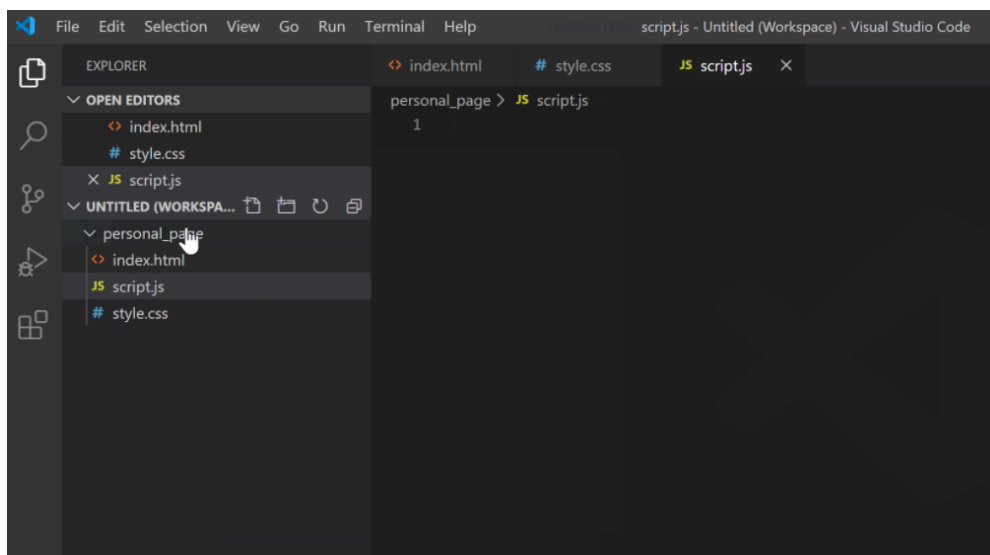


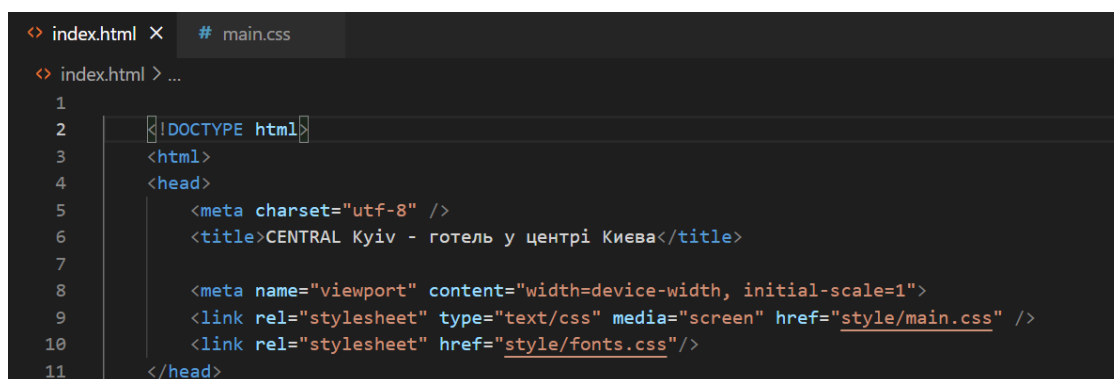
Рис. 3.5.2. Додавання потрібних файлів

Відредагуємо тільки `index.html` (файл з розміткою) і далі розглянемо рядки нашого коду.

Елемент DOCTYPE використовують для вказівки типу поточного документа - DTD (document type definition, опис типу документа), з метою розуміння браузером способу інтерпретування поточної веб-сторінки. Для визначення початку HTML-файла використовують тег html, який має всередині заголовок (head) і тіло документа (body).

Тегом meta можна змінювати кодування сторінки, додавати ключові слова, опис документа і багато іншого. Даний рядок застосовується для того, щоб браузеру було зрозуміло, що він має справу з кодуванням UTF-8 (Unicode transformation format, формат перетворення Юнікод).

На наступному малюнку (рис. 3.5.3.) відображено, як саме я підключала зовнішні файли та додавала заголовок.



```
<> index.html x # main.css
<> index.html > ...
1
2 | <!DOCTYPE html>
3 | <html>
4 |   <head>
5 |     <meta charset="utf-8" />
6 |     <title>CENTRAL Kyiv - готель у центрі Києва</title>
7 |
8 |     <meta name="viewport" content="width=device-width, initial-scale=1">
9 |     <link rel="stylesheet" type="text/css" media="screen" href="style/main.css" />
10 |    <link rel="stylesheet" href="style/fonts.css"/>
11 |   </head>
```

Рис. 3.5.3. Підключення зовнішніх файлів

Далі я оформлювала головну сторінку лендінгу.

Додала гамбургер-меню (рис. 3.5.4) з посиланнями на розділи Landing page, логотип та номер телефону. Контент я розміщую в елементах поділу контенту Div, який не впливає на контент або макет до тих пір, поки не буде стилізований за допомогою CSS.

```

14 <div class="main-screen container">
15   <div class="main-screen-header clearfix">
16     <div class="hamburger">
17       <a class="menu-btn">
18         <span></span>
19       </a>
20     </div>
21     <div class="logo"> <a href="#"> 
22     <div class="phone"> <p class="num">+38 (095) 950 95 00</p></div>
23   </div>
24   <div class="menu">
25     <div class="menu-content">
26       <h3><a href="#">ГОЛОВНА</a></h3>
27       <h3><a href="#feature">ПЕРЕВАГИ</a></h3>
28       <h3><a href="#presentation">ОГЛЯД ГОТЕЛЮ</a></h3>
29       <h3><a href="#stages">ВІДГУКИ</a></h3>
30       <h3><a href="#portfolio">НОМЕРА ТА ЦІНИ</a></h3>
31       <h3><a href="#futer">КОНТАКТИ</a></h3>
32     </div>
33   </div>

```

Рис. 3.5.4. Створення меню

Кнопку меню я стилізувала за допомогою моливостей CSS (рис. 3.5.5). Задала розміри та позицію, вирівнювання, колір, обтікання та інше.

```

<> index.html # main.css 7 X
style > # main.css > .feedback-form-group .bar
54 .hamburger{
55   width: 26px;
56   height: 20px;
57   padding-right: 528px;
58   margin-top:22px;
59   display: flex;
60   justify-content: center;
61   align-items: center;
62   float: left;
63 }
64 .menu-btn {
65   display: block;
66   width: 26px;
67   height: 20px;
68   position: relative;
69   text-decoration: none;
70 }
71 .menu-btn span,
72 .menu-btn span::before,
73 .menu-btn span::after{
74   position: absolute;
75   top:50%; margin-top: -1px;
76   left:50%;margin-left: -13px;
77   width: 26px;
78   height: 2px;
79   background-color: #fff;
80 }

```

Рис. 3.5.5. Стилізування кнопки меню

У створеному файлі hamburger.js додаємо обробник подій (див. рис. 3.5.6). Подія DOMContentLoaded відбувається коли весь HTML буде повністю завантажений і

пройдений парсером, не чекаючи закінчення завантаження таблиць стилів, зображень і фреймів.

```
script > JS hamburger.js > ...
1
2 window.addEventListener('DOMContentLoaded', function() {
3   var menuBtn=document.getElementsByClassName("menu-btn")[0];
4   var menuDisplay=document.getElementsByClassName("menu")[0];
5   function openMenu(event){
6     event.preventDefault();
7     menuBtn.classList.toggle("menu-btn_active");
8     if(menuBtn.classList.contains("menu-btn_active")){
9       menuDisplay.style.display="block";
10    }else{
11      menuDisplay.style.display="none";
12    }
13  }
14  menuBtn.addEventListener("click",openMenu);
15  });
16
```

Рис. 3.5.6. Обробник подій для кнопки меню

Далі розглянемо реалізацію слайдів на головній сторінці лендінгу (рис. 3.5.7). Для відображення на веб-сторінці зображень в графічному форматі GIF, JPEG або PNG використовується тег `img`. Через атрибут `src` задаємо адресу файла.

Зображення розмістила в створеній папці “img”.

```
34 <div class="slider">
35   <div class="slider-item">
36     
37   </div>
38   <div class="slider-item">
39     
40   </div>
41   <div class="slider-item">
42     
43   </div>
44   <div class="slider-item">
45     
46   </div>
47   <div class="slider-dots clearfix">
48     <div class="dot dot-active"></div>
49     <div class="dot"></div>
50     <div class="dot"></div>
51     <div class="dot"></div>
52   </div>
53 </div>
```

Рис. 3.5.7. Реалізація слайдів

За допомогою JavaScript в новому файлі slider.js додаємо функцію, за допомогою якої зображення будуть змінюватися самостійно.

```
window.addEventListener('DOMContentLoaded', function() {
  let slideIndex = 1,
  slides = document.getElementsByClassName('slider-item'),
  dotsWrap = document.querySelector('.slider-dots'),
  dots = document.getElementsByClassName('dot');
  showSlides(slideIndex);
  function showSlides(index){
    if (index > slides.length){
      slideIndex = 1;
    };
    if (index < 1){
      slideIndex = slides.length;
    };
    for (let i = 0; i < slides.length; i++) {
      slides[i].style.display = 'none';
    };
    for (let i = 0; i < dots.length; i++) {
      dots[i].classList.remove('dot-active');
    };
    slides[slideIndex - 1].style.display = 'block';
    dots[slideIndex - 1].classList.add('dot-active');
  };
  function plusSlides(index) {
    showSlides(slideIndex += index);
  };
  function currentSlide(index) {
    showSlides(slideIndex = index);
  };
});
```

Рис. 3.5.8. JavaScript файл для слайдів

На наступному скріншоті (рис. 3.5.9) коду я додала кнопку бронювання номеру, яка є посиланням на форму контактних даних. А також створила посилання на соціальні мережі.

```
<a href="#book"><div class="button-main"> <p>ЗАБРОНЮВАТИ НОМЕР</p> <span class="arrow"> </span></div></a>
<div class="arrow-down"></div>
<div class="social">
  
  
  
</div>
```

Рис. 3.5.9. Посилання на соц.мережі

Далі я створювала блок із перевагами готелю (рис. 3.5.10). Кожен блок із текстом було реалізовано наступним чином у HTML-файлі:

```
<div class="feature container clearfix">
  <a name="feature"></a>
  <div class="content content-1">
    <h2>01</h2>
    <h3>Безкоштовний Wi-Fi</h3>
    <p>Для Вашої зручності ми розмістили безкоштовний інтернет на території всього готелю і навіть</p>
  </div>
```

Рис. 3.5.10. Блок з перевагами

А так його відстилізовано у CSS-документі:

```
.feature{
  margin-top: 0;
  height: 720px;
}
.feature .content{
  height: 270px;
  float:left;
  margin-top: 150px;
}
.feature .content-1{
  width: 24.358%;
}
```

Рис. 3.5.11. Стилізація контенту

Щодо відео-презентації, то її я прикріпила за допомогою тегу <video>, що додає, відтворює і управляє настройками відеоролика на веб-сторінці, та його атрибутів. Також додала кнопки зупинки відео та відключення звуку (див. рис. 3.5.12).

```
<a name="presentation"></a>
<div class="presentation">
  <video id="myvideo" poster="img/video/preview.png" class="presen
    <source src="video/Hotel.webm" type="video/webm">
    <source src="video/Hotel.mp4" type="video/mp4">
    <source src="video/Hotel.ogv" type="video/ogg">
  </video>
  <div class="presentation-button button-play"
  id="button-video-player">

</div>
<p>Огляд готелю CENTRAL Kyiv за 1 хвилину</p>
<div id="pauseplay" class="pause">PAUSE</div>
<span id="volume">
  <svg id="dynamic" version="1.0" id="Layer_1" xmlns="http
  width="16px" height="16px" viewBox="0 0 95.465 95.465">
```

Рис. 3.5.12. Відео-презентація

Давайте більш детально роздивимось реалізацію форми відправки контактних даних на пошту (рис. 3.5.13). Код HTML можна побачити нижче. Обов'язкові поля позначені як required, тому якщо користувач не заповнить їх, то не зможе відправити форму. Система покаже йому текст з помилкою: «Заповніть поля!».

```
<div class="feedback clearfix">
  <div class="feedback-main">
    <form method="get" action="#" name="feedback" class="feedback-main-form">
      <h2>
        <a name="book">Хочеш забронювати номер?</a>
      </h2>
      <p>Залиште контактні дані, менеджер зв'яжеться з Вами
        на протязі 15 хвилин для уточнення деталей.</p>
      <div class="feedback-form-group">
        <input type="text" name="name" required>
        <span class="bar"></span>
        <label>Имя</label>
      </div>
      <div class="feedback-form-group">
        <input type="tel" name="phone" required>
        <span class="bar"></span>
        <label>Телефон</label>
      </div>
      <div class="feedback-form-group">
        <input type="text" name="mail" required>
        <span class="bar"></span>
        <label>E-mail</label>
      </div>
      <button type="submit">
        Забронювати
      </button>
      <p>* натискаючи на кнопку, Ви погоджуєтесь з умовами обробки персональних даних.</p>
    </form>
  </div>
```

Рис. 3.5.13. Реалізація форми відправки даних



Для відправки письма на пошту власника сайту потрібні мови розмітки HTML і мови програмування PHP. На цьому етапі потрібно створити файл form.php, в нього додати html код форми.

Наступний етап - створення файлу, задача якого приймати дані з HTML форми. Це буде файл з ім'ям send.php. У файлі на першому етапі потрібно прийняти дані з пост масиву. Для цього створюємо дві змінні:

```
$name = $_POST['name'];  
$phone = $_POST['phone'];  
$email = $_POST['email'];
```

Рис. 3.5.14. Прийняття даних з POST масиву

Знак \$ розміщується перед назвами змінних в php і в кінці кожного рядка ставиться крапка з комою. В масив \$\_POST передаються дані з форми. При цьому зазначений метод відправки method = "post" у формі html. Таким чином, ми отримуємо дві змінні з форми html. З метою захисту сайту проведемо ці змінні через кілька фільтрів - функцій php.

Розглянемо їх - перша функція перетворює всі символи, які користувач спробує вписати у форму:

```
$name = $_POST['name'];  
$phone = $_POST['phone'];  
$email = $_POST['email'];  
  
$name = htmlspecialchars($name);  
$phone = htmlspecialchars($phone);  
$email = htmlspecialchars($email);
```

Рис. 3.5.15. Перетворення символів

Але в цей час нові змінні в php не створюються, використовуємо вже наявні. Фільтр перетворює символ "<" в '& lt;'. Ті ж самі перетворення він робить з іншими символами, що зустрічаються в html коді.

Друга функція декодує url в тому випадку, коли користувач спробує його додати в форму (рис. 3.5.16.).

```
$name = urldecode($name);  
$phone = urldecode($phone);  
$email = urldecode($email);
```

Рис. 3.5.16. Декодування url

Третя функція - видалення пробілів з початку і кінця рядка, якщо такі є:

```
$name = trim($name);  
$phone = trim($phone);  
$email = trim($email);
```

Рис. 3.5.17. Видалення пробілів

Далі ми використовуємо функцію mail в PHP для відправки даних на пошту. mail ("на яку адресу відправити", "тема листа", "тіло листа", "From: з якого email відправляється лист \r\n");

```
mail("centralkyiv@gmail.com", "Бронювання Central Kyiv",  
"Ім'я:".$name.". E-mail: ".$email.". Телефон:".$phone , "From: client@gmail.com \r\n");
```

Рис. 3.5.18. Функція mail для відправки даних

За допомогою наступної умови ми перевіримо, чи відправилась дані форми на зазначені адреса електронної пошти (рис. 3.5.19).

```
if (mail("centralkyiv@gmail.com", "Бронювання Central Kyiv", "Ім'я:".$name.". E-mail: ".$email.". Телефон:".$phone , "From: client@gmail.com \r\n"))  
{  
    echo "Ваші дані відправлено!";  
} else {  
    echo "При відправленні виникла помилка!";  
}
```

Рис. 3.5.19. Умова перевірки відправки даних

Далі в один файл ми додаємо HTML і PHP код відправки форми в один файл і додаємо умову (рис. 3.5.20), яка перевірятиме наявність змінних в масиві POST (цей масив передається з форми). Таким чином, якщо змінні в масиві не існують, то потрібно показати користувачеві форму. В іншому випадку потрібно прийняти дані з масиву і відправити їх адресату.

```
if(!isset($_POST['name']) and !isset($_POST['email'])){\n    ?> <form action="send.php" method="post">
```

Рис. 3.5.20. Умова для перевірки змінних в масиві

Функцією `isset()` перевіряємо, чи існує змінна в POST масиві. Знак оклику перед цією функцією в умові означає заперечення. Якщо змінної не існує, то потрібно показати нашу форму.

Повернемося до блоків лендінгу. Розділ із зображеннями та інформацією про номер я реалізувала таким чином:

```
<a name="portfolio"></a>\n<div class="portfolio container clearfix">\n  <h2>Наші номери</h2>\n  <ul class="portfolio-menu clearfix">\n    <span class="left">\n      <li class="portfolio-menu-dot portfolio-menu-dot-active">Люкс на двох</li>\n      <li class="portfolio-menu-dot">Делюкс на трьох</li>\n      <li class="portfolio-menu-dot">Полулюкс на двох</li>\n    </span>\n    <span class="right">\n      <li class="portfolio-menu-dot">Стандарт на двох</li>\n      <li class="portfolio-menu-dot">Сімейний номер</li>\n    </span>\n  </ul>\n  <div class="portfolio-slider-item">\n    \n    <div class="portfolio-slider-item-content">\n      <h3>Люкс на двох</h3>\n      <p>Двокімнатні номери категорії Люкс задовольняють потреби найбільш вимогливи  
Зручність і ергономічність простору дозволяють поєднати роботу і відпочинок.  
Великий балкон з видом на річку та місто.\n    </p>
```

Рис. 3.5.21. Реалізація розділу з номерами

І в кінці HTML-файлу я додаю всі використані при розробці JS-файли:

```
<script src="script/scriptimg.js"></script>
<script src="script/hamburger.js"></script>
<script src="script/Jquery.js"></script>

<script src="script/player.js"></script>
<script src="script/modal.js"></script>
<script src="script/slider.js"></script>

</body>
</html>
```

Рис. 3.5.22. Прикріплення JS-файлів

### Висновки до розділу 3

В даному розділі було розв'язано деякі проблеми, через які безліч лендінгів так і не стають успішними. Це наявність логотипу на сторінці, номеру телефону для зв'язку та фото- і відеопрезентацій продукту. Зручна та коротка форма зворотного зв'язку також має велику вагу для успішної посадкової сторінки.

Якщо казати більш детально, то цей розділ демонструє створену структуру сайту, контекстну діаграму та діаграму станів. Також в ньому розглянуто сценарій роботи Landing page для користувачів системи з метою полегшення їхньої взаємодії із розробленим лендінгом. Було створено і реалізовано цікавий дизайн, який буде привертати увагу клієнтів і спонукати залишити свої контактні дані.

В підрозділі зі сценарієм було прикріплено результати розробки даного сайту. Можна побачити, що реалізувати простий лендінг можна, використовуючи якісне програмне забезпечення та правильні й зручні технології. В останньому підрозділі можна знайти детальний опис самої розробки та прикріплені скріншоти коду. Також додано зразки інтерфейсу для спрощення сприйняття інструкції по використанню програми.

Даний Landing page орієнтований на гостей готелю, але цю систему також можна інтегрувати у різні середовища та використовувати для різноманітних потреб. Для прикладу, оформлення заявок на дизайн інтер'єру або запис на вебінар.

## ВИСНОВКИ

Дипломна робота присвячена розробці Landing page для закладу готельного бізнесу. При виконанні дипломного проекту були досліджені методи створення лендінгів та проаналізовані типові проблеми при їх розробці. Після чого, вивчивши актуальну літературу, що включає в себе новітню інформацію з розробки інформаційних систем, було розкрито ключові аспекти реалізації успішної Landing page. При вирішенні поставлених задач були отримані наступні висновки:

1. Лендінг-сторінка відіграє вирішальну роль у збільшенні впізнаваності компанії, підвищення продажів, поліпшення SEO показників і залучення нових клієнтів. Посадкова сторінка - ефективний спосіб розширити клієнтську базу. Це основні причини, чому слід використовувати цільову сторінку.

2. Можна виділити наступні плюси хороших лендінгів: лаконічність в тексті і дизайні, стильна анімація, нестокові фотографії, наявність оригінальних малюнків, відгуків, гумору, простих форм реєстрації, чітко позначених переваг та зрозумілих кнопок СТА.

3. Для успіху компанії потрібно чітко продумати структуру лендінгу, яка залежить від цільової аудиторії. Для готелю була розроблена своя структура з блоками, в яких описано його переваги, відгуки та інформація про номери, а також додано відео- та фотопрезентації.

В ході написання даної роботи була проаналізована актуальність розробки Landing page і розглянуто відмінності від звичайного сайту. Наразі можна сказати, що для формування довіри до бренду, подання нової інформації краще використовувати стандартний багатосторінковий сайт. Але, якщо потрібен швидкий результат, потрібно вибирати саме Landing page. Тому що саме таке рішення може гарантувати замовнику максимально швидке отримання контактних даних або заявок. Також лендінг можна тестувати і таким чином покращувати для підвищення конверсії.

Виходячи з усього вище сказаного, можна зробити наступний висновок: Landing page - це більше, ніж корисний ресурс. Такий односторінковий сайт доносить

до читачів виключно ту інформацію, яка викликає в них інтерес або приносить їм реальну користь. Займає небагато часу для ознайомлення та має цільовий характер.

Також хочу звернути увагу, що було проведено аналіз ефективності використання різних технологій для реалізації функціонального лендінгу. З цією метою було досліджено переваги програмного забезпечення, середі розробки та мов програмування, які застосовувались при розробці сайту. Технології, які було розглянуто, показали, що правильний підбір технологій для розробки лендінгу може не тільки полегшити процес розробки, а й зробити його цікавим.

Як результат дипломної роботи отримано працездатний лендінг, який можна використовувати як систему бронювання номерів готелю, який корисний для обох сторін. Для користувачів (майбутніх гостей) – це легкий шлях для ознайомлення із інформацією готелю та швидке бронювання номеру, оскільки, окрім заповнення своїх контактів, їм не потрібно витрачати багато часу на заповнення форми. Для власника готелю – це швидкі контакти, які потім можна використовувати менеджери для наступних бронювань, реклами або інших цілей. Беззаперечно, цей проект сучасний та актуальний, і буде розвиватись в майбутньому. В планах покращувати готовий функціонал, додавати новий, покращувати зовнішній вид та зручність використання. Надалі можна додати більше посилань на заповнення форми та створення.

## СПИСОК БІБЛЮГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ле В. Landing Page – Как превратить посетителей в продажи/ Ле В., СПб.: Питер, 2015. – Бібліогр.: с. 5-22.
2. Лендинг: основные преимущества и недостатки [Електронний ресурс] – Режим доступу: <http://gusarov-group.by> (дата звернення 14.05.2021) – Назва з екрана.
3. Кишеня П. Формула Landing page/ Кишеня П., Москва, 2015 – Бібліогр.: с. 6.
4. Филиппов, С. А. Основы современного веб-программирования / С. А. Филиппов. – Москва.: НИЯУ МИФИ, 2015. – Бібліогр.: с. 160 с.
5. Visual Studio Code Getting started [Електронний ресурс] – Режим доступу: [https://en.wikipedia.org/wiki/Visual\\_Studio\\_Code](https://en.wikipedia.org/wiki/Visual_Studio_Code) (дата звернення 16.05.2021) – Назва з екрана.
6. Майо Дж. Самоучитель Microsoft Visual Studio 2010 / Джо Майо – СПб.: БХВ-Петербург, 2011. – Бібліогр.: с.450 с.
7. VS Code [Електронний ресурс] – Режим доступу: <https://bizzapps.ru/p/vs-code/> (дата звернення 17.05.2021) – Назва з екрана.
8. HTML5 [Електронний ресурс] – Режим доступу: <http://htmlbook.ru/html5> (дата звернення 17.05.2021) – Назва з екрана.
9. CSS3 [Електронний ресурс] – Режим доступу: <https://metanit.com/web/html5/5.1.php> (дата звернення 18.05.2021) – Назва з екрана.
10. HTML и CSS3 будущее веб-дизайна [Електронний ресурс] – Режим доступу: <http://starwebdesign.com.ua/>(дата звернення 18.05.2021) – Назва з екрана.
11. Бондарев В.М. Основы программирования.js : учеб. пособ. / В.М. Бондарев. – Х. : ФОП Коряк С. Ф., 2015. – Бібліогр.: с. 182 с.
12. What is JavaScript? [Електронний ресурс] – Режим доступу: [https://www.w3schools.com/whatis/whatis\\_js.asp](https://www.w3schools.com/whatis/whatis_js.asp) (дата звернення 21.05.2021) – Назва з екрана.