

**ЗАКЛАД ВИЩОЇ ОСВІТИ
УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА**

**Факультет суспільних і прикладних наук
Кафедра інформаційних технологій**

на правах рукопису

Мотало Тарас Ігорович

УДК 004.415

**Розробка web-застосунку для онлайн замовлення їжі та
напитків**

Спеціальність 121 — «Інженерія програмного забезпечення»

Кваліфікаційна робота на здобуття освітнього ступеня
бакалавра

Науковий
керівник д.т.н.,
професор
Мельничук Степан
Іванович

Івано-Франківськ — 2021

Зм.	Лист	№ докум.	Підпис	Дата	КР.ІПЗ. – 10.00.000 ПЗ			
Розроб.		Мотало Т.І.			WEB-застосування для замовлення онлайн їжі та питтівків Пояснювальна записка	Літ.	Арк.	Аркушів
Перевір.		Мельничук С.І.					6	63
Реценз.		Кузь М.В.				ЗВО «УКД» ІПЗс-2017		
Н. Контр.		Зорін В.О.						
Затверд.		Пашкевич О.П.						

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	9
ВСТУП.....	10
РОЗДІЛ 1. ПЕРЕДУМОВИ, АНАЛІТИКА ТА ПРИЗНАЧЕННЯ, ОПИС НАЯВНИХ ВЕБ-САЙТІВ АНАЛОГІВ.....	12
1.1 Передумови.....	12
1.2 Призначення та аналітика	15
1.3 Опис наявних веб-сайтів аналогів.....	17
Висновки до розділу 1	22
РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ФУНКЦІОНАЛУ	23
2.1 Функціонал	23
2.2 Проектування бази даних.....	23
Висновки до розділу 2	28
РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ ІНТЕРНЕТ РЕСУРСУ	29
3.1 Огляд структури проекту	29
3.2 Розробка функціоналу сайту.....	31
3.3 Розробка інтерфейсу сайту.....	36
Висновки до розділу 3	44
РОЗДІЛ 4. ОХОРОНА ПРАЦІ	45
4.1 Умови праці на робочому місці з ПК.....	45
4.2 Мікроклімат та ступінь іонізації повітря у приміщені.....	47
4.3 Вимоги до безпечної експлуатації комп'ютерної техніки.....	55
Висновки до розділу 4	56
ВИСНОВКИ.....	57

					КР. ІПЗ – 10.00.00.000 ІПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

					КР. ПЗ – 10.00.00.000 ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

- CSS – це спеціальна мова стилю сторінок, що використовується для опису їхнього зовнішнього вигляду.
- HTML – це спеціальна мова стилю сторінок, що використовується для опису їхнього зовнішнього вигляду.
- SQLite – полегшена реляційна система керування базами даних.
- Cov 19 – це інфекційна хвороба, яку спричиняє новий штам коронавірусу.
- СУБД – Система управління базами даних.
- UML – уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування.
- БД – база даних.
- MVT – це шаблон дизайну програмного забезпечення. Сукупність трьох важливих компонентів Model View та Template.
- API – інтерфейс програмування застосунків.
- REST – підхід до архітектури мережевих протоколів, які надають доступ до інформаційних ресурсів.
- UI – це принципи та відповідний процес розробки інтерфейсу користувача для машин та програмного забезпечення.
- GIT – розподілена система керування версіями файлів та спільної роботи.
- Use case – це опис поведінки системи, як вона відповідає на зовнішні запити.

					КР. ІПЗ – 10.00.00.000 ІЗ	Арк.
						9
Зм.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Актуальність теми. Весь світ захопила епідемія коронавірусу. Влада кожної країни вводить карантин і позбавляє людей можливості відвідувати публічні місця, в тому числі – кафе і ресторани, але всім хочеться інколи відпочити та поїсти щось приготоване не вдома. Вихід з цієї ситуації – онлайн замовлення їжі та напоїв.

Такий вихід з даної ситуації це не тільки безпека для свого здоров'я, але і зручність для клієнтів, можливість не з банкрутувати для закладів харчування та можливість розширити бізнес.

Інтернет – магазини вже давно не новина. Один із перших таких магазинів Amazon – зараз один із найпопулярніших сайтів в США, в Україна один із перших онлайн-магазинів Rozetka – набула великого успіху, людям зручно замовляти з дому чи будь якого місця на планеті не встаючи з крісла. Це дає можливість людям замовляти їм потрібний товар, не маючи потреби тратити фінанси та час для відвідування закладів.

Успіх інтернет – магазинів дозволяє бути впевненим, що сервіси онлайн-замовлень лиш набирають актуальності, і можна вважати що такі сервіси будуть успішні для бізнесу.

Лиш недавно люди дізнались про такі онлайн-сервіси замовлення як Glovo та Raketa, це безумовно успішні стартапи, які в першу чергу доставляють їжу, а в умовах пандемії це ідеальний помічник в буденному житті. Навіть якщо би ми представили що пандемія не відбулась, ці програми були такі ж актуальні, людям необхідно харчуватись в незалежності від того що коїться в світі. Ви можете бути зайняті на роботі, чи у вас море хатніх справ і у вас немає часу готувати їжу чи йти в кафе, тоді онлайн замовлення та доставка їжі ідеальне вирішення проблеми.

Мета роботи. Розробити web-застосунок для замовлення їжі та напоїв, з можливістю обслуговування клієнтів на адмініструванням сайту.

					КР. ПЗ – 10.00.00.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		10

Об'єкт роботи. Процес замовлення їжі та напоїв.

Предмет роботи. Web-застосунок для замовлення їжі та напоїв, з можливістю обслуговування клієнтів на адмініструванням сайту.

Завдання роботи. Відповідно до вибраної теми є наступні задачі:

- Розробити зручний та швидкий інтерфейс для користувача та адміністратора.
- Розробити дієвий та доступний функціонал.
- Розробити сучасний та приємний дизайн.
- Розробити функціонал реєстрації для користувача та адміністратора.
- Розробити функціонал замовлення продукту для користувача.
- Розробити функціонал додавання продукту для адміністраторів.
- Проведення тестування продукту.

Методи роботи. Для вирішення поставленої задачі були використані відомості про сайти онлайн-замовлень, офіційна документація веб-фреймворка Django, мови програмування Python, мови гіпертекстової розмітки HTML та каскадної таблиці стилів CSS. Даний проект дозволяє додавання в подальшому нового функціоналу.

Результати роботи. Результатом дипломної роботи є web-застосунок для замовлення їжі та напоїв, який дає можливість замовляти продукт, реєструвати користувачів, адміністраторів та вносити в базу даних інформацію про продукт, користувачів та замовлення

Структура роботи. Розділи – 4. Загальний обсяг основної частини – 45. Список використаних джерел містить – 15 позиції.

					КР. ІПЗ – 10.00.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		11

РОЗДІЛ 1. ПЕРЕДУМОВИ, АНАЛІТИКА ТА ПРИЗНАЧЕННЯ, ОПИС НАЯВНИХ ВЕБ-САЙТІВ АНАЛОГІВ

1.1 Передумови

Від коли людство придумало перший комп'ютер, програмісти та інженери не припиняються створювати речі, які полегшують повсякденне життя людини. З кожним роком з'являється все більше інноваційних та цікавіших сайтів та стартапів які приносять користь людству.

Спершу це були не зрозумілі для простої людини мови програмування типу Fortran та Asembler, а зараз Python, Pascal та JavaScript починають вчити в старших класах школи. Тепер люди бачать сайти, з інтуїтивним інтерфейсом, в якому розбереться навіть дитина. Безумовно інформаційні технології приносять тільки користь людству. Вони дали нам інформаційні сайти по типу «блогів», медичних, технічних чи історичних. Зрозуміло, дуже багато залежить, насамперед, від вас самих і команди фахівців, яка допомагає вам на різних етапах роботи. Сайти почали замінити нам бібліотеки, кінотеатри та навіть магазини. Немає речей які не можна замовити в інтернеті, це економить час, а час завжди був в людини в найважливіших пріоритетах. Так виникли перші сайти онлайн-замовлень, такі як Amazon (рис. 1.1) в США, і Rozetka (рис. 1.2) в Україні. Доставка товарів відіграла один з найважливіших факторів успіху сайтів онлайн-замовлень. Amazon один із найуспішніших інтернет-проектів в США, це дало задуматись всім іншим програмістам з усього світу створювати аналоги в себе в країні.

Свіжі продукти, які раніше переважно продавалися в офлайн, отримали величезне зростання онлайн-замовлень. Поставки овочів збільшилися на 600% в порівнянні з минулим роком в період Китайського Нового року. Китайський онлайн-ритейлер JD.com повідомив, що його онлайн-продажі зросли на 215% в

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

порівнянні з аналогічним періодом минулого року до 15 000 тонн протягом 10-денного періоду з кінця січня до початку лютого.

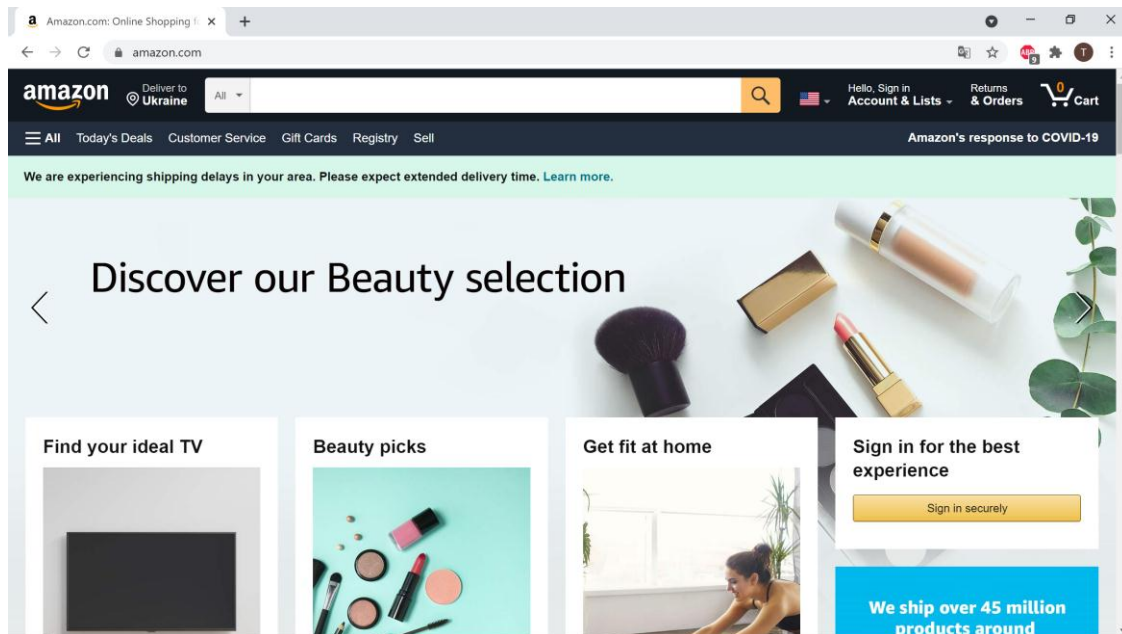


Рисунок 1.1 – Сайт Amazon

Програмісти та бізнесмени почали створювати інтернет-магазини на різну тематику. Це онлайн-аптеки, магазини із автомобільними запчастинами, електричними приборами, онлайн-бібліотеки, книжкові магазини, магазини з одягом і багато іншої тематики. Люди отримали змогу економити час та не ходити в по багатьох магазин в своєму місті. Інколи потрібного товару немає ніде в твоєму місті і виникає потреба їхати в інше місто, а то і в іншу країну. Завдяки сайтам, де можна замовляти онлайн, така потреба перестала існувати. Люди мають доступ до каталогів з різними товарами, з різних країн та міст. Так і в нас з'явилась Rozetka, яка з кожним роком збільшує кількість категорій з товарами, від смартфонів до алкогольних напоїв та спортивного спорядження. Людям просто потрібно зайти на сайт, знайти потрібний товар і замовити. Служби доставки доставлять його за декілька днів. Також не так давно з'явилась платформа prom.ua, яка вміщає багато сайтів для онлайн замовлення.

					КР.ІПЗ – 10.00.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13

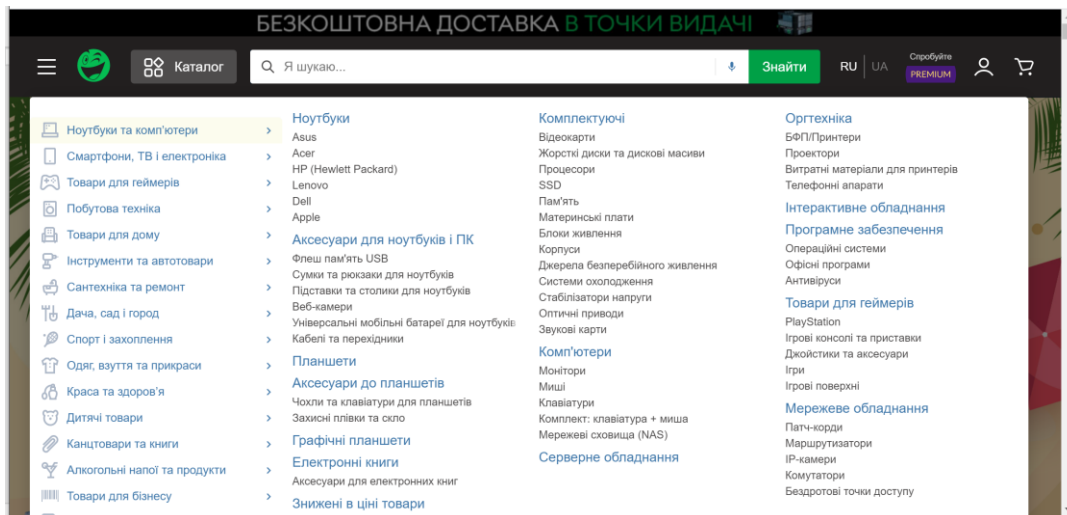


Рисунок 1.2 – Сайт Rozetka

Тепер будь який товар можна замовити з будь-якої точки планети. Для прикладу можна взяти компанію Alibaba (рис. 1.3), яка керує сайтом AliExpress. Завдяки такому інтернет-магазину людина за пару хвилин замовляє абсолютно любий товар з Китаю прямо до себе в поштову скриньку. Їхнім сервісом користуються десятки мільйонів і популярність тільки росте.

Найбільшу цінність сайтів онлайн-замовлень світ напевне побачив в 2020, коли почалась пандемія вірусу Cov-19. Весь світ буквально опинився на карантині. Влада всіх країн почала вводити заборони на відвідування публічних місць, в тому числі торгових центрів, ресторанів та інших привичних людям для відвідування місць. Люди лишались закритими вдома, без змоги відвідати для них привичні ресторани і інші для них привичні публічні місця проведення вільного часу і не тільки.

Можна прожити без торгових центрів, кінотеатрів, зоопарків чи інших розважальних центрів, але без їжі та продуктів ніяк. Звички покупців змінюються, а хвороба продовжує поширюватися по континентах.

Під час спалаху коронавірусу цілі міста пішли на карантин, сервіси доставки готової їжі додому стали свого роду рятівним колом як для ресторанів, так і для їх клієнтів.

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		14

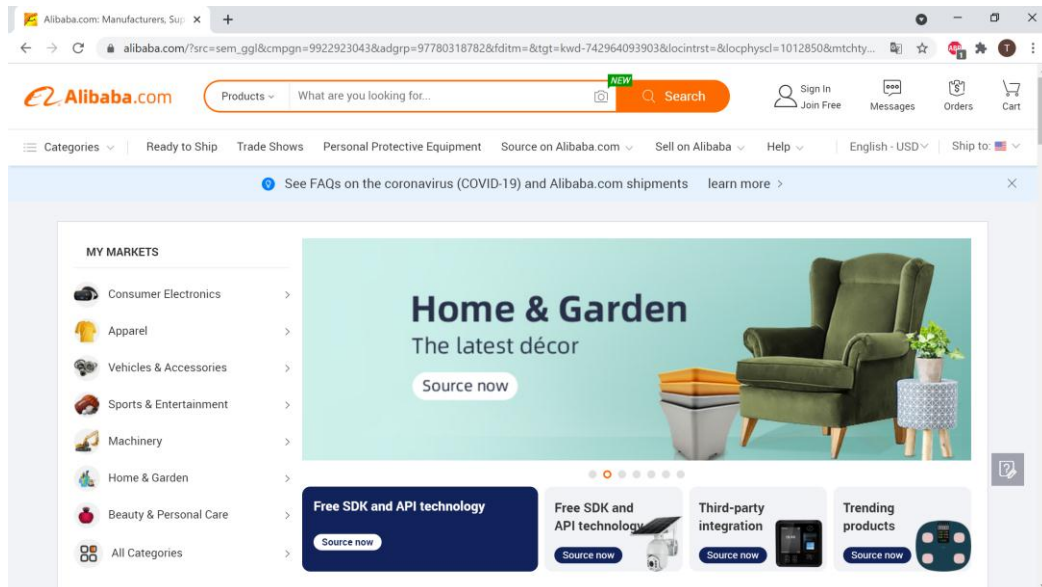


Рисунок 1.3 – Сайт Alibaba

В цей самий час набирають популярності сервіси доставки такі як Glovo та Raketa. Вроді би сервіси доставки і до цього були, але ці почали вирізнятись. На мою думку причина, в простоті та доступності, тільки подумайте, запустив додаток, пройшов швидко реєстрацію і можна замовляти їжу прямо з телефону за пару хвилин. Ці сервіси доставки також дали можливість багатьом людям під час карантину мати додатковий заробіток. Простота відбувалась не тільки для клієнтів, а також для працівників. Люди мали змогу мати додаткову роботу і полегшувати життя людям, які не мали змоги самі ходити за продуктами.

1.2 Призначення та аналітика

Складається так що саме такі сервіси онлайн-замовлень та доставки приходять на допомогу в тяжкі часи пандемії. Люди не можуть відвідувати магазини та ресторани, через заборону влади та ризику хвороби, і через це масово замовляють все додому. В багатьох країнах вводили комендантську годину, тоді навіть при «самовиносі» забрати замовлення вечором буде вкрай важко. Залишається тільки онлайн-замовлення та доставка.

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

Людині зручно навіть перебуваючи не в зоні карантину користуватись сервісами онлайн-замовлення. Будучи на роботі в офісі, чи в іншому місті у відрядженні, навіть вдома інколи будучи зайнятий домашніми справами це самий зручніший спосіб замінити похід в кафе чи інший заклад харчування чи або навіть приготування самому, ну ж погодьтесь, не завжди є час приготувати пообідати. Інколи просто легше витратити пару хвилин на замовлення і трішки почекати на доставку і можна спокійно обідати. Онлайн-замовлення їжі та напоїв завжди допоможуть під час святкування будь яких подій, чи то до вас просто завітали друзі, чи на офісі після робочого часу ви влаштуєте «тімбілдинг» з піцою і настільними іграми.

Не тільки зі сторони клієнта можна бачити вигоду зі сторони таких сервісів, а й зі сторони бізнесу. На довгі місяці закривають заклади харчування, і дозволяють працювати тільки на «самовиніс». Онлайн замовлення та доставка буквально дозволили виживати закладам харчування. Вони дають змогу замовляти людям їжу не відвідуючи заклад, і заклади продовжують працювати у звичному для них режимі.

Клієнти не мають ризикувати своїм здоров'ям та порушувати норми карантину, відвідуючи заклади харчування, але без клієнтів кафе та ресторани не зможуть працювати. Адміністрація буде змушена звільняти та скорочувати своїх робітників, щоб якимось втримати бізнес та не закритись через втрату фінансового прибутку. Деякі заклади харчування приймали рішення робити літні майданчики, збільшувати дистанцію між столами та обмежувати кількість відвідувачів. Такі методи не завжди працюють, наприклад зимою на літніх майданчиках клієнтів обслуговувати не вийде, а збільшення дистанції між столами не відмінить заборони на повне відвідування публічних місць та закладів харчування, лишається тільки працювати на замовлення онлайн та доставку.

Замовлення онлайн дозволяють дотримуватись всіх норм карантину в будь яку пору року. Адміністрація може сама налаштувати меню закладу, переглядати базу даних користувачів і вносити в неї корективи. Клієнти при

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		16

реєстрації залишають свої контактні дані, такі як номер телефону та пошту, завдяки цьому адміністратори можуть виходити на зворотній зв'язок, для уточнення питань та вирішення деяких нюансів, які можуть виникнути при замовленні товару. На сайті присутні контактні дані закладу харчування, через які клієнти можуть зв'язуватись з рестораном, якщо виникнуть якісь питання або скарги. Такий функціонал дозволяє наладжувати хорошу комунікацію

В умовах карантину ідеальним рішенням як для клієнтів та бізнесменів, які утримують заклади харчування являються сайти для замовлення онлайн та доставки додому. Вони дозволяються клієнтам напряму замовляти будь який товар, який адміністратор закладу занесе в меню на сайті, а закладам харчування продовжити успішно обслуговувати клієнтів.

1.3 Опис наявних веб-сайтів аналогів

Переважає більшість закладів харчування мають свої сайти для замовлення онлайн їхніх продуктів та послуг. Деякі просто несуть інформаційний характер для вибору продукції, деякі повноцінні з базою даних клієнтів, аналітики, популярності продуктів в різних містах. Багато популярних таких сайтів викладає рекламу, і заробляє на цьому. Якщо говорити про серйозні мережі закладів харчування, які мають свій сайт, вони можуть збирати інформацію за клієнтів та історію замовлень, щоб в подальшому використовувати для маркетингу свого закладу та рекомендацій продукту, за для збільшення продажів власних продуктів.

Сервіси доставки онлайн також мають свої сайти та додатки, найвідоміші в Україні напевне всім відомі Raketa та Glovo. Вони надали перевагу додаткам для смартфона. Користувачу потрібно просто зайти на App Store або Play Market, скачати додаток та пройти швидко реєстрацію. Вони мають велику та не просту базу даних. В ній зберігаються дані про користувачів, заклади харчування, меню кожного закладу харчування, дані про кур'єрів, магазини, аптеки та інших закладів, які можна реєструвати в даних сервісах. Для

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

підтримки цих додатків витрачаються мільйони, і ще більший вони приносять прибуток власникам.

Давайте розглянемо декілька сайтів для онлайн-замовлення їжі в місті Івано-Франківськ. Першим мабуть буде самий відомий в місті сайт від закладу Pronto Pizza (рис. 1.4). В основному цей заклад готує піцу та має власного кур'єра. Зайшовши на головну сторінку ми бачимо на всю ширину браузера банери про акції, це безумно плюс, тому що всі клієнти люблять акції, і краще знати про них до того як зробив замовлення. Акції притягують клієнтів та вирізняють заклад між іншими закладами.

На сайті Pronto Pizza можна робити замовлення пряму з сайту, додаючи товар в корзину, без обов'язкової потреби їм телефонувати. Після створення замовлення з вами зв'язується адміністратор для підтвердження замовлення. Якщо немає змоги зробити замовлення онлайн, на сайті є контактні дані, по яких можна на пряму зв'язатись з менеджером.

В нижній частині сайту можна побачити карту міста, на ній позначаються області, в межах яких проводиться доставка замовлення. Замовлення доставляється на протязі години.

До плюсів можна додати що на сайті присутня сторінка, на якій можна самому вибирати з чого буде складатись піца, так званий «конструктор». Також якщо потрібно на пряму зв'язатись з менеджером є кнопка зв'язку. Після натискання на неї відкривається форма, в яку потрібно вписати свій мобільний номер телефону та очікувати дзвінка менеджера. Орієнтовно з дзвінок менеджера відбудеться за одну хвилину.

З мінус можна винести, що відсутність реєстрації являється не продуктивним рішенням.

Відсутність бази даних з клієнтами не йде на користь закладу, та не дає можливості проводити рекламу орієнтуючись із зібраних з бази даними інформації про клієнтів та їх замовлення. Також менеджери самі мають при дзвінку збирати інформацію про клієнтів, замовлення та адресу доставки.

					КР.ІПЗ – 10.00.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		18

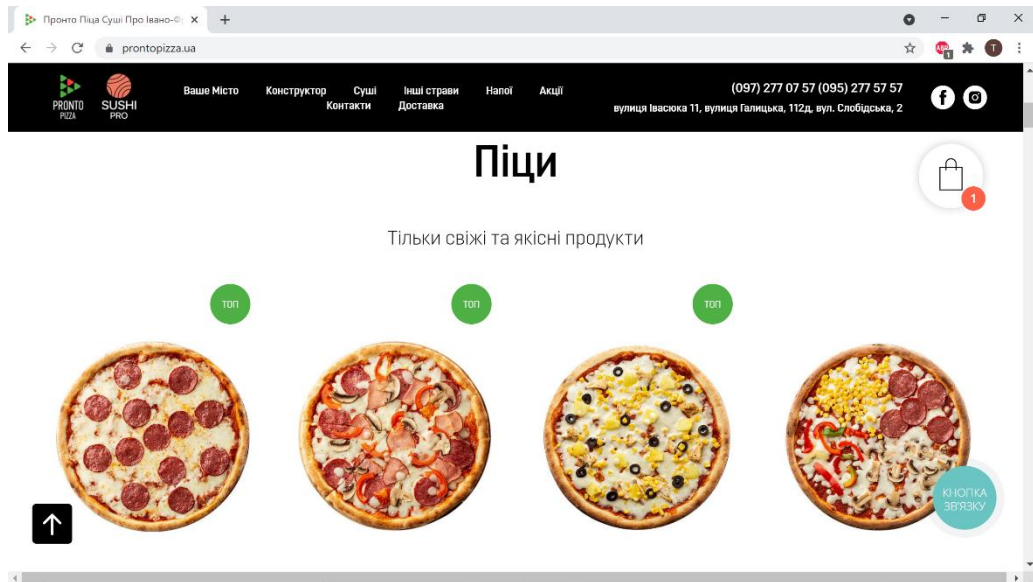


Рисунок 1.4 – Сайт Pronto Pizza

Наступний сайт для онлайн замовлення – від закладу «Piratos» (рис. 1.5). Це простий інформаційний сайт, в якому є меню, з опис кожного товару, галерея, інформаційна сторінка закладу та сторінка з контактною інформацією. Перше що бачить клієнт коли заходить на сайт – це контактну інформацію закладу та адрес. Прогортавши в низ ми бачим розділ «спеціальні пропозиції» і також акції. В горі знаходиться меню сайту. Там присутні розділи, описані вище.

З плюсів можна вивести гарний дизайн, також зручний та швидкий інтерфейс.

З мінусів – відсутність можливості замовлення товару прямо із сайту. На сайті відсутня корзина або інший функціонал для прямого замовлення. Для того, щоб зробити замовлення потрібно зв'язуватись з менеджером по мобільному телефону. Також відсутня реєстрація клієнтів. Така ж сама проблематика присутня на попередньому сайті. Сайт написаний без бази даних, що дозволяє економити на утримуванні сайту. Менеджери не опрацьовують дані, які зберігаються при реєстрації клієнтів. Це не дозволяє використовувати історію замовлень, для проведення реклами в тому регіоні, де працює заклад.

Такі засоби як реєстрація клієнтів та історія їх замовлень є обов'язковими.

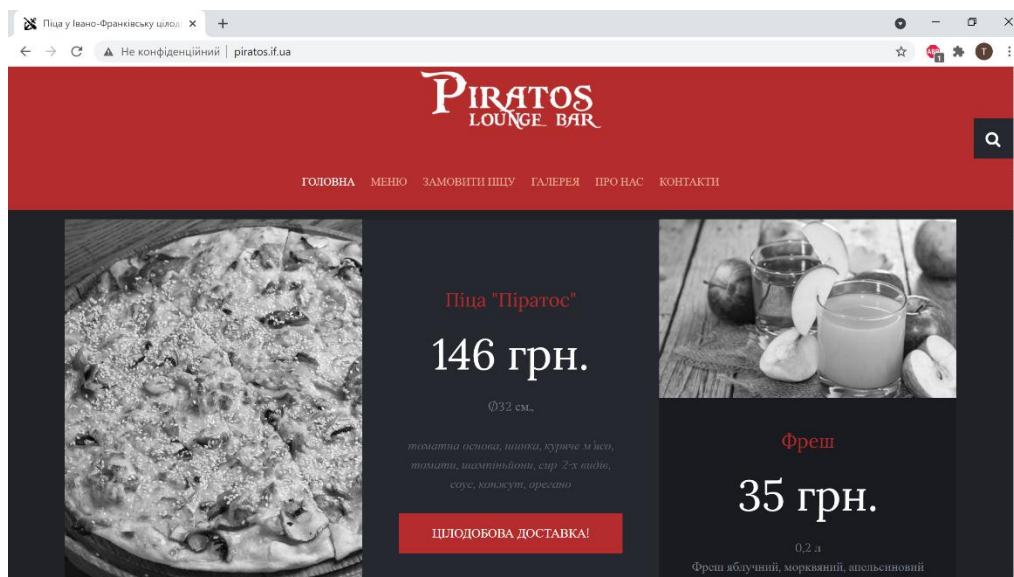


Рисунок 1.5 – Сайт ресторану «Піратос»

Давайте розглянемо мобільні додатки для замовлення і доставки їжі та інших продуктів онлайн. Як я описав в горі, в Україні є два самих популярних додатка, це Glovo та Raketa, вони надзвичайно схожі функціоналом та дизайном. Ці додатки були придумані в основному виключно для доставки товару, їжі та напоїв. Розглянемо їх функціонал на прикладі додатку «Glovo».

Відкривши додаток Glovo (рис. 1.6) ми бачим наступні розділи: «Магазини», «Їжа», «Здоров'я та краса», «Експрес кур'єр», «Супермаркети» та «Що завгодно». В розділі «Магазини» ми бачимо список зареєстрованих в додатку магазини, які знаходяться у вашому місті, ви можете замовити напряму в даних магазинах товар на доставку до вашого дому. Розділ «Супермаркети», «Квіти» працює так же само як і розділ «Магазини». В розділі «Краса та здоров'я» ви можете знайти аптеки і замовити в них товар на доставку. В розділі «Їжа» присутні заклади харчування. В кожному закладі є своє меню та категорії їжі та напоїв. В розділі «Що завгодно» ми самі вказуємо що нам потрібно придбати та доставити, після чого кур'єр шукає заданий товар то

						КР.ІПЗ – 10.00.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата			20

доставляє його. Замовлення товару відбувається через додавання його в корзину. Під час вибору товару можна замовити додатки до замовленої їжі. Далі ми вказуємо адресу куди потрібно доставити, повідомляємо чи немає алергії на вибраний товар, підтверджуємо адресу доставки, вибираємо спосіб оплати та підтверджуємо замовлення.

З плюсів можна вивести зручну форму реєстрації та можливість відстежити на якому етапі доставки ваше замовлення. Є етап пошуку кур'єра, підтвердження закладу, готування страви, передавання кур'єру та сама доставка до адреси, яку вказав клієнт. Також плюсом для кур'єра являється можливість залишити чайові, при оплаті банківською карткою. А з мінусів – це те що не вся територія міста входить в межі доставки і інколи приходиться довго чекати на доставку замовлення, якщо адреса доставки не в центрі міста.



Рисунок 1.6 – Додаток «Glovo»

					КР.ІПЗ – 10.00.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		21

Висновки до розділу 1

В процесі роботи над розділом було розглянуто передумови актуальності web – застосунку, його призначення для користувачів і підприємців та були розглянуті сайти аналоги, їх плюси та мінуси.

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
						22
Зм.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ФУНКЦІОНАЛУ

2.1 Функціонал

Навка — це продуманий сайт для замовлення їжі та напоїв онлайн. Його функціонал дозволяє вести базу даних з користувачами, продуктами та замовленнями. Інформаційні дані також будуть занесені в базу даних.

Адмініструванням сайту буде займатись менеджер закладу харчування. В нього буде можливість керувати інформацією про товари внесеною в базу даних та підтвердженням інформації про замовлення.

Було створено Use case проекту (рис. 2.1).

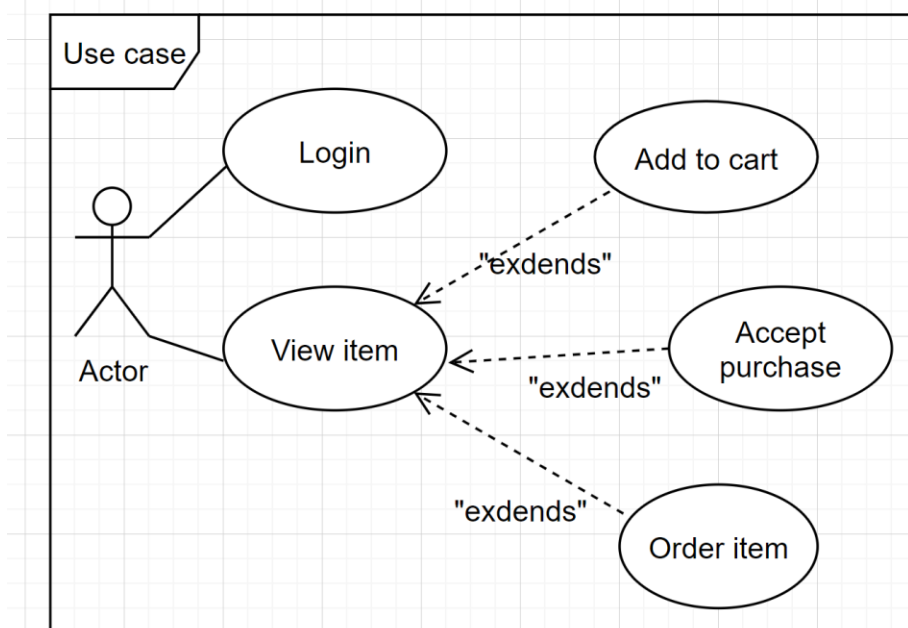


Рисунок 2.1 – Use case

2.2 Проектування бази даних

Для створення бази даних я використав СУБД – SQLite. Для коректної роботи сайту було створено наступні таблиці (рис. 2.2):

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		23

- Contacts.
- Order.
- OrderProduct.
- Product.
- Kind.
- Promotions.
- Users.

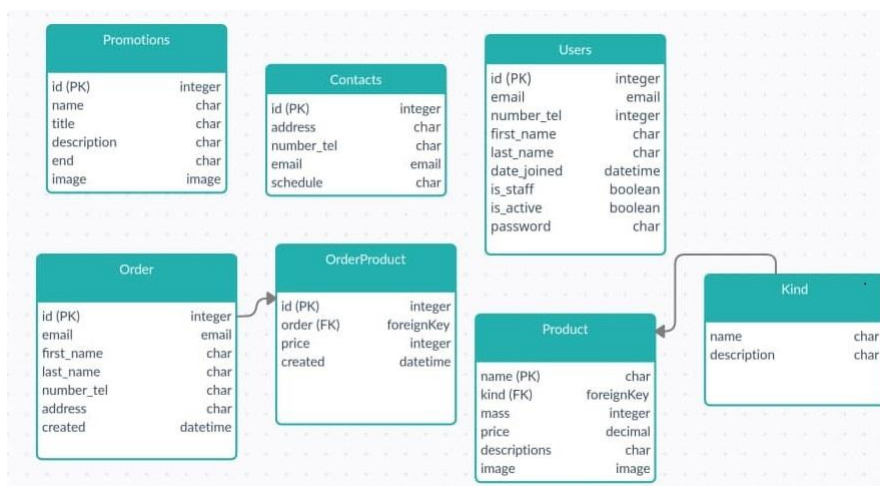


Рисунок 2.2 – UML діаграма

Contacts – таблиця з контактною інформацією закладу (рис. 2.3).

Имя	Тип	НП	ПК	АИ	У	По умолчанию	Проверить	Внешний ключ
id	integer	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
address	varchar(50)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
number_tel	integer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
email	varchar(254)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
schedule	text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			

Рисунок 2.3 – Таблиця Contacts

- Address – адреса.
- Number_tel – номер телефону.
- Email – електронна пошта.
- Shulde – графік роботи.

Order – таблиця з інформацією про замовника (рис. 2.4).

Имя	Тип	НП	ПК	АИ	У	По умолчанию	Проверить	Внешний ключ
id	integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
email	varchar(254)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
first_name	varchar(50)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
last_name	varchar(50)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
address	varchar(250)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
created	datetime		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
number_tel	varchar(12)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			

Рисунок 2.4 – Таблица Order

- Email – адреса електронної пошти.
- First_name – імя замовника.
- Last_name – прізвище замовника.
- Address – адреса доставки.
- Created – дата створення замовлення.
- Number_tel – номер телефону.

OrderProduct – таблиця з інформацією про замовлені продукти (рис.2.5).

Имя	Тип	НП	ПК	АИ	У	По умолчанию	Проверить	Внешний ключ
id	integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
product	varchar(200)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
price	real		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
created	datetime		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
order_id	bigint		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			"order_order"("id") DEFERRABLE INITIALLY DEFERRED

Рисунок 2.5 – Таблица OrderProduct

- Product – назва продукту.
- Price – ціна продукту.
- Created – дата створення продукту.
- Order_id – вторинний ключ на таблицю order.

Product – таблиця з інформацією про продукти, які є в меню (рис. 2.6).

Имя	Тип	НП	ПК	АИ	У	По умолчанию	Проверить	Внешний ключ
id	integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
name	varchar(50)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
mass	integer		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
price	decimal		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
description	varchar(500)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
image	varchar(100)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
kind_id	bigint		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			"product_kind"("id")

Рисунок 2.6 – Таблица Product

- Name – назва продукту.
- Mass – маса в грамах.
- Price – ціна продукту.
- Description – опис продукту.
- Image – фото продукту.
- Kind_id – вторинний ключ.

Kind – таблиця з каталогами продукту (рис. 2.7).

Имя	Тип	НП	ПК	АИ	У	По умолчанию	Проверить	Внешний ключ
id	integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
name	varchar(50)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
description	varchar(500)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			

Рисунок 2.7 – Таблица Kind

- Name – назва каталогу товарів.
- Description – опис каталогу.

Promotions – таблиця з інформацією про акції (рис. 2.8).

Имя	Тип	НП	ПК	АИ	У	По умолчанию	Проверить	Внешний ключ
id	integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
name	varchar(100)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
title	varchar(200)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
description	varchar(500)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
end	varchar(100)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
image	varchar(100)		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			

Рисунок 2.8 – Таблица Promotions

- Name – назва акції.
- Title – заголовок акції.
- Description – опис акції.

- End – тривалість акції.
- Image – фото акції.

Users – таблиця з інформацією про зареєстрованих користувачів та адміністраторів. В цій таблиці присутній функціонал для надання прав доступу для менеджерів (рис. 2.9).

Имя	Тип	НП	ПК	АИ	У	По умолчанию	Проверить	Внешний ключ
id	integer	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
password	varchar(128)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
last_login	datetime	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
is_superuser	bool	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
email	varchar(254)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>			
first_name	varchar(50)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
last_name	varchar(50)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
date_joined	datetime	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
is_staff	bool	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
is_active	bool	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
number_tel	varchar(12)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			

Рисунок 2.9 – Таблица users

- Password – пароль, для входу в систему.
- Last_login – час успішного входу в систему.
- Is_superuser – поле з логічним значенням, чи є користувач супер-адміністратором.
- Email – адреса електронної пошти користувача.
- First_name – ім'я користувача.
- Last_name – прізвище користувача.
- Date_joins – дата реєстрації.
- Is_staff – поле з логічним значенням, чи є користувач адміністратором.
- Is_active – поле з логічним значенням, чи користувач активний.
- Number_tel – мобільний номер користувача.

Створення самої бази даних відбувається в шаблоні models з веб-фреймворка Django. В даній базі даних відсутні зв'язки один до одного, або багато до багатьох. Дані, які зберігаються в БД мають визначену логічну структуру, тобто описуються деякою моделлю представлення даних (моделлю даних), яка підтримується СУБД. Було використано реляційну модель бази даних.

Висновки до розділу 2

В розділі було описано функціонал та поведінку системи, як вона відповідає на зовнішні запити. Було також описано базу даних та її модель.

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
						28
Зм.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ ІНТЕРНЕТ РЕСУРСУ

3.1 Огляд структури проекту

Для розробки проекту було використано MVT паттерн. Коли веб-сайт отримує запит, то url-dispatcher оприділяє до якого ресурсу передати запит, ресурс – це View, який отримує запит, під час обробки запиту звертається до бази даних (Model), отримує або вносить і зберігає в неї дані. Результат обробки повертається назад у View та передається в templates (рис. 3.1).

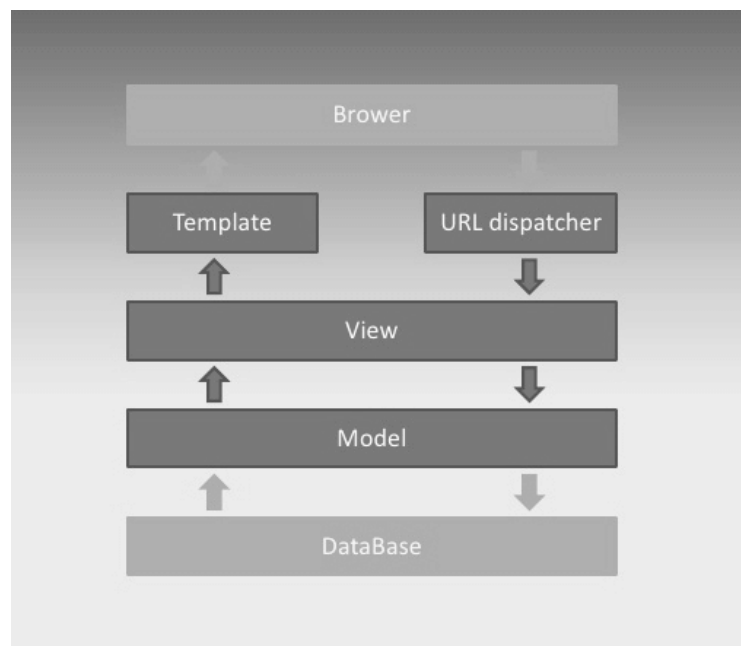


Рисунок 3.1 – MVC модель

Проект умовно поділено на дві частини, серверна(бек-енд), та клієнтська(фронт-енд). Оскільки фронтенд – це зовнішній вигляд, а бекенд – це власне вся механіка сайту.

Серверна частина відповідає за:

- опис моделі даних для БД

- шляхи(роути) для API, з якими може працювати проект
- контролери для цих роутів, що задають функціонал для окремих REST-запросів
- запускає сервер
- налаштовує зв'язок та взаємодію з БД
- налаштовує необхідні middleware
- Клієнтська частина відповідає за:
- надання зручного UI для взаємодії з бек-ендом

Програма складається із застосунків(підпрограм) та головного каталогу (рис. 3.2), в якому зберігаються основні налаштування проекту. Також є файл бази даних, файл із залежностями та файл для системи управління версіями Git.

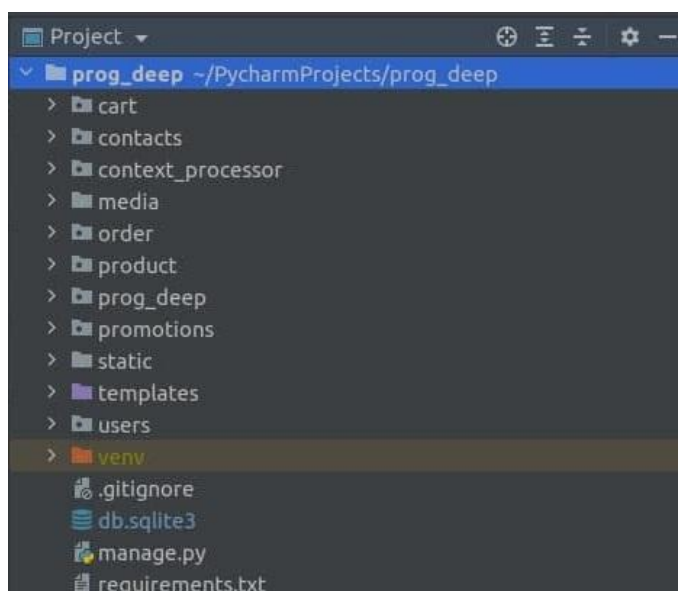


Рисунок 3.2 – Структура проекту

- Prog_deep – папка самого проекту, в якому зберігаються всі дані та архітектура.
- Cart – застосунок, в якому описується функціонал корзини.
- Contacts – застосунок, який відповідає за контактні дані закладу харчування.

- Context_processor – застосунок, в якому написана функція для контекстного процесору програми.
- Media – каталог, в якому зберігаються всі медіа-файли програми та її статичні файли.
- Order – застосунок, в якому описується функціонал для замовлення продукту.
- Product – застосунок, в якому описується функціонал для товару.
- Prog_deep – головний каталог програми, в ньому описуються налаштування проекту та основний диспетчер.
- Promotions – застосунок, в якому описується функціонал для акцій закладу харчування.
- Static – каталог, в якому зберігаються статичні файли(css та media).
- Templates – каталог в якому зберігаються html-сторінки.
- Users – застосунок, в якому описується функціонал для реєстрації користувачів.
- gitignore – файл, в якому вказується які файли та каталоги будуть ігноруватись системою контролю версіями Git.
- Venv – каталог з віртуальною машиною.
- Db.sqlite3 – файл з базою даних.
- Manage.py – інтерфейс для django-admin.
- Requirements.txt – файл, в якому зберігаються назви всіх необхідних залежностей проекту.

В застосунках, де використовується база даних, створюються підкаталогиз файлами міграції. Міграції використовуються в Django для переносу змін у моделях (додавання полів, видалення моделей тощо) на структурі баз даних.

3.2 Розробка функціоналу сайту

В проекті розроблено 6 застосунків:

					КР.ІІЗ – 10.00.000 ІЗ	Арк.
						31
Зм.	Арк.	№ докум.	Підпис	Дата		

- Cart.
- Contacts.
- Product.
- Promotions.
- Order.
- Users.

Застосунок складається з наступних файлів:

- `__init__` – файл, який відображає Django, що поточний каталог є пакетом Python.
- `admin` – реєструє моделі бази даних на сторінці адміністрування.
- `apps` – використовується для конфігурації застосунку.
- `models` – зберігає в собі опис моделей бази даних.
- `forms` – описує форми, які використовують для занесення інформації в базу даних.
- `tests` – використовується для написання тестів.
- `urls` – диспетчер url-адресів.
- `views` – в даному файлі описуються функції, які отримують запит від користувача, обробляють його та повертають назад.

Розглянемо код файлів кожного застосунку. Почнімо із застосунку users.

Файл `models`:

```
class User(AbstractBaseUser, PermissionsMixin):
    email = models.EmailField(verbose_name='email address', unique=True)
    number_tel = models.CharField(max_length=12, verbose_name='Мобільний номер')
    first_name = models.CharField(max_length=50, blank=True, verbose_name='Імя')
    last_name = models.CharField(max_length=50, blank=True,
    verbose_name='Прізвище')
    date_joined = models.DateTimeField(auto_now_add=True)
    is_staff = models.BooleanField(default=False)
    is_active = models.BooleanField(default=True)
```

Далі було розроблено методи для форматowanego виведення інформації на сторінку адміністрування сайту.

```
def get_full_name(self):
    return self.first_name + " "
    + self.last_name

def get_short_name(self):
```

					КР.ІПЗ – 10.00.000 ІПЗ	Арк.
						32
Зм.	Арк.	№ докум.	Підпис	Дата		

```

return self.first_name
def __str__(self):
return self.email

```

Файл forms:

```
class CustomUserCreationForm(forms.ModelForm):
```

Форми для пароля користувача:

```

password1 = forms.CharField(help_text='Введіть пароль', label='',
widget=forms.PasswordInput, error_messages={'required': ''})
password2 = forms.CharField(help_text='Введіть пароль ще раз', label='',
widget=forms.PasswordInput, error_messages={'required': ''})
email = forms.EmailField(help_text='Вкажіть вашу електронну пошту', label='',

```

Форми для контактних даних користувача:

```

first_name = forms.CharField(help_text='Вкажіть ваше імя', label='',
error_messages={'required': ''})
last_name = forms.CharField(help_text='Вкажіть ваше прізвище', label='',
error_messages={'required': ''})
number_tel = forms.CharField(help_text='Вкажіть ваш мобільний номер', label='',
error_messages={'required': ''})

```

Методи для шифрування та зберігання інформації в формах для паролів:

```

def clean_password2(self):
password1 = self.cleaned_data.get("password1")
password2 = self.cleaned_data.get("password2")
if password1 and password2 and password1 != password2:
raise forms.ValidationError("Паролі не збігаються!")
return password2

def save(self, commit=True):
user = super(CustomUserCreationForm, self).save(commit=False)
user.set_password(self.cleaned_data["password1"])
if commit:
user.save()
return user

```

Файл urls:

```

app_name='users'
urlpatterns = [ path('register/', RegisterFormView.as_view(), name="register"),
path('login/', views.LoginView.as_view(), name='login'),
path('logout/', views.LogoutView.as_view(), name='logout'), ]

```

Файл views:

```

class RegisterFormView(FormView):
form_class = CustomUserCreationForm
success_url = "/main/"
template_name = "registration/register.html"

```

									Арк.
									33
Зм.	Арк.	№ докум.	Підпис	Дата					

```

def form_valid(self, form):
    form.save()
    return super(RegisterFormView, self).form_valid(form)
def form_invalid(self, form):
    return super(RegisterFormView, self).form_invalid(form)

```

Застосунок order. Файл models.

Клас для таблиця order:

```

class Order(models.Model):
    email = models.EmailField(verbose_name='Електронна пошта')
    first_name = models.CharField(max_length=50, verbose_name='Імя')
    last_name = models.CharField(max_length=50, verbose_name='Прізвище')
    number_tel = models.CharField(max_length=12, blank=True, verbose_name='Номер

```

Клас для таблиці OrderProduct:

```

class OrderProduct(models.Model):
    order = models.ForeignKey(Order, on_delete=models.CASCADE,
verbose_name='Замовив(ла)')
    product = models.CharField(max_length=200, verbose_name='Товар')
    price = models.FloatField(default=0, verbose_name='Ціна')
    created = models.DateTimeField(auto_now_add=True, blank=True,
verbose_name='Дата створення')
    class Meta:
        verbose_name = 'Замовлений товар'
        verbose_name_plural = 'Замовлені товари'

```

Файл forms:

```

class OrderForm(forms.ModelForm):
    email = forms.EmailField(help_text='Вкажіть вашу електронну пошту',
label='Електронна пошта',
error_messages={'required': ''})
    first_name = forms.CharField(help_text='Вкажіть ваше імя', label='Імя',
error_messages={'required': ''})
    last_name = forms.CharField(help_text='Вкажіть ваше прізвище', label='Прізвище',
error_messages={'required': ''})

```

Файл views. Метод для створення замовлення:

```

def order_create(request):
    if not request.user.is_authenticated:
        return redirect('users:login')
    cart = Cart(request)
    if request.method == 'POST':
        form = OrderForm(request.POST)
        order = form.save()
        mass = []

```

Метод, який виконується після успішного створення замовлення:

```

def done(request):
    if not request.user.is_authenticated:
        return redirect('users:login')

```

						Арк.
						34
Зм.	Арк.	№ докум.	Підпис	Дата		

```
return render(request, 'order_done.html')
```

Файл urls:

```
app_name='
order'
urlpatterns = [
    path('order_create/', views.order_create, name='order_create'),
    path('order_done/', views.done, name='order_done')
```

Застосунок product. Файл models. Клас для таблиці Kind:

```
class Kind(models.Model):
    name = models.CharField(max_length=50, verbose_name='Назва')
    description = models.CharField(max_length=500, verbose_name='Опис')
    def __str__(self):
        return self.name
```

Клас для таблиці Product:

```
class Product(models.Model):
    kind = models.ForeignKey(Kind, on_delete=models.CASCADE,
        verbose_name='Продукт', related_name='RelProduct')
    name = models.CharField(max_length=50, verbose_name='Назва')
    mass = models.IntegerField(verbose_name='Маса в грамах')
    price = models.DecimalField(max_digits=10, decimal_places=2,
        verbose_name='Ціна')
    description = models.CharField(max_length=500, verbose_name='Опис')
    image = models.ImageField()
```

Файл views. Метод для виведення інформації на головну сторінку:

```
def main(request):
    if not request.user.is_authenticated:
        return redirect('users:login')
    return render(request, 'index.html')
def menu(request):
    cart_form = CartForm
    if not request.user.is_authenticated:
        return redirect('users:login')
    return render(request, 'menu.html', {'product_list': Kind.objects.all(), 'cart_form':
cart_form})
```

Файл urls:

```
app_name = 'product'
urlpatterns = [
    path('main/', views.main, name='main'),
    path('menu/', views.menu, name='menu'),
]
```

Для замовлення продуктів створена корзина, яка працює через сесії.

```
class Cart(object):
```

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
						35
Зм.	Арк.	№ докум.	Підпис	Дата		

```

def __init__(self, request):
    self.session = request.session
    cart = self.session.get(settings.CART_SESSION_ID)
    if not cart:
    cart = self.session[settings.CART_SESSION_ID] = {}
    self.cart = cart

```

Метод для додавання товару в корзину:

```

def add(self, product, quantity=1, update_quantity=False):
    product_id = str(product.id)
    if product_id not in self.cart:
        self.cart[product_id] = {'id': str(product_id),
                                  'name': str(product.name),
                                  'quantity': 0,
                                  'price': str(product.price)}
    if update_quantity:
        self.cart[product_id]['quantity'] = quantity
    else:
        self.cart[product_id]['quantity'] += quantity
    self.save()

```

Метод для зберігання товару в корзині:

```

def save(self):
    self.session[settings.CART_SESSION_ID] = self.cart
    self.session.modified = True

def remove(self, product):
    product_id = str(product.id)
    if product_id in self.cart:
        del self.cart[product_id]
    self.save()

```

Функції додавання товару в корзину.

```

@require_POST
def cart_add(request, product_id):
    if not request.user.is_authenticated:
        return redirect('users:login')
    cart = Cart(request)
    product = get_object_or_404(Product, id=product_id)
    form = CartForm(request.POST)
    if form.is_valid():
        cd = form.cleaned_data
        cart.add(product=product,
                 quantity=cd['quantity'],
                 update_quantity=cd['update'])
    return redirect('cart:cart_detail')

```

3.3 Розробка інтерфейсу сайту

					КР.ІІЗ – 10.00.000 ІІЗ	Арк.
						36
Зм.	Арк.	№ докум.	Підпис	Дата		

У проєкті важливу ролі посідає дизайн, UI частина. Саме через неї користувач взаємодіє з системою, проходить реєстрацію та замовляє потрібний товар. Чим красивіший, приємніший і простіший дизайн, тим більше користувачів захоче ним користуватись і збільшиться кількість клієнтів в закладі харчування.

Часто шаблони мають однотипну базову структуру, блоки повторюються, але вміст різний. Це дозволяє розробити єдиний стиль сайту, де веб-сторінки мають одні і ті ж структурні блоки та елементи – меню, хедер, футер і так далі.

Тоді ми можемо виділити всі шаблони окремо. Але якщо потрібно буде змінити якийсь блок, наприклад, додати в хедер ще один пункт, або в меню кнопку, то потрібно буде міняти всі шаблони, а їх може бути багато і це буде далеко не продуктивно. В такому випадку ми можемо використовувати один базовий шаблон, в якому оприділяються всі основні блоки.

В даному проєкті я розробив базовий шаблон, і назвав його base.html.

Код файлу base.html:

```
{% load static %}
<meta charset="UTF-8">
<link rel="stylesheet" href="{% static 'css/main.css' %}">
{% block title %}<title>Навка</title>{% endblock %}
```

За допомогою елементів `{% block %}` `{% endblock %}` оприділяються блоки шаблонів. Між тегами вказується контент.

Наприклад:

```
{% block title %} title {% endblock title %}
```

По замовчуванню значення буде використано значення яке вписане в блоці базового шаблону.

Далі базовий шаблон наслідується в інших шаблонах, наприклад:

```
{% extends "base.html" %}
{% block title %}Меню{% endblock title %}
{% block header %}Головна{% endblock header %}
```

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37

Також використовується тег `{% load static %}`, він вказує що ми будемо використовувати статичні файли(css та media) з каталогу із статичними файлами.

Розглянемо наступні шаблони, які наслідують базовий шаблон `base.html`.

Шаблон `cart.html` (рис. 3.3), він створений для відображення сторінки корзини:

```
{% load static %}
{% block title %}
    <title>Корзина</title>
{% endblock %}
{% block background %}
    <div class="banner" id="p"></div>
{% endblock %}
{% block content %}
    <h1>Ваша корзина</h1>

    <ul>
        {% for item in cart %}
            <li>Назва: {{ item.name }}</li>
            <li>Кількість: {{ item.quantity }}</li>
            <li>Ціна: {{ item.price }}</li>
            <a href="{% url "cart:cart_remove" item.id %}">
                <a href="{% url "product:menu" %}" class="glow-button">Продовжити
покупки</a>
                <a href="{% url "order:order_create" %}" class="glow-button">Підтвердити
замовлення</a>
            </p>
        {% endblock %}
```

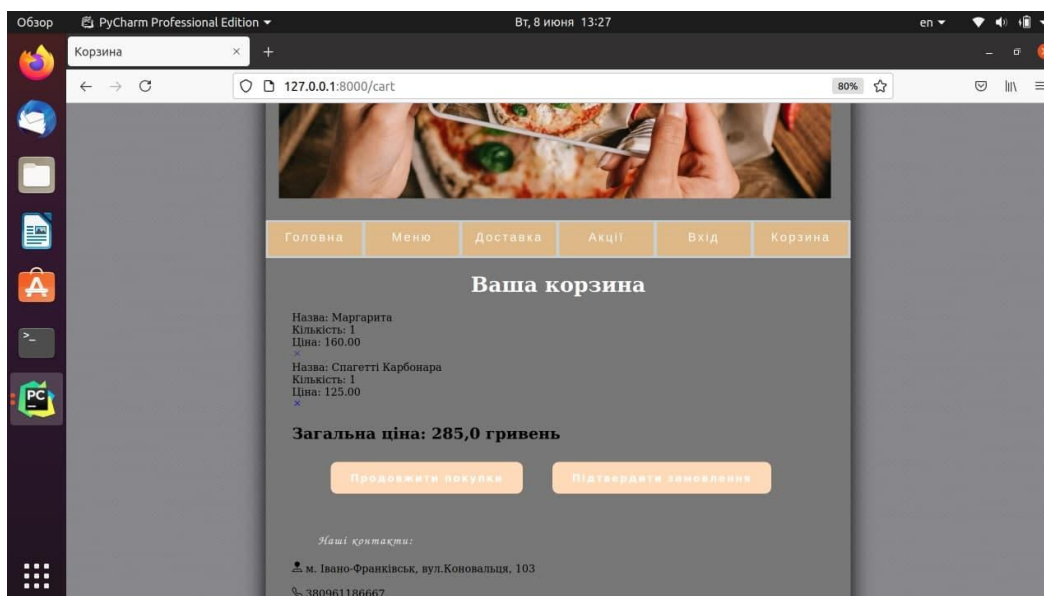


Рисунок 3.3 – Сторінка корзини

Шаблон delivery.html (рис. 3.4), в ньому відображається сторінка з інформацією про доставку:

```
{% extends 'base.html' %}
{% load static %}
{% block title %}
    <title>Доставка</title>
{% endblock %}

{% block background %}
    <div class="banner" id="p"></div>
{% endblock %}

{% block content %}
```

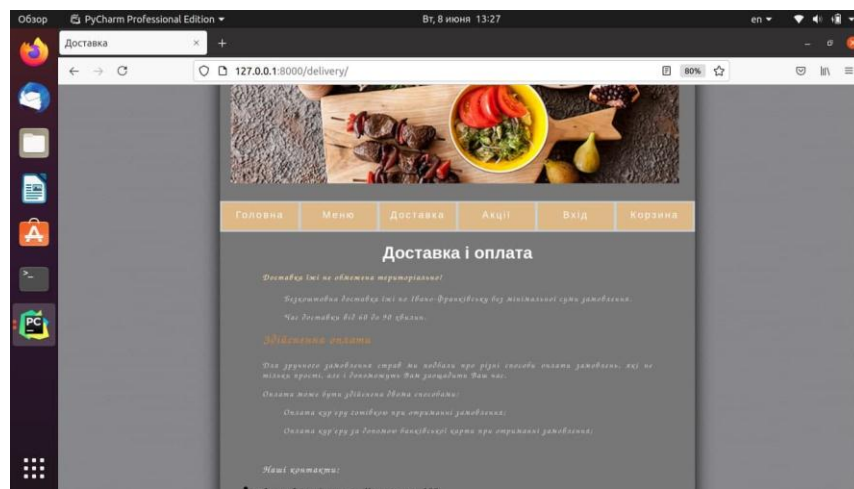


Рисунок 3.4 – Сторінка доставки

Шаблон index.html (рис. 3.5), в ньому відображається головна сторінка сайту:

```
{% extends 'base.html' %}
{% load static %}
{% block title %}
    <title> Головна </title>
{% endblock %}
{% block background %}
    <div class="banner" id="p"></div>
{% endblock %}
{% block content %}
{% endblock %}
```

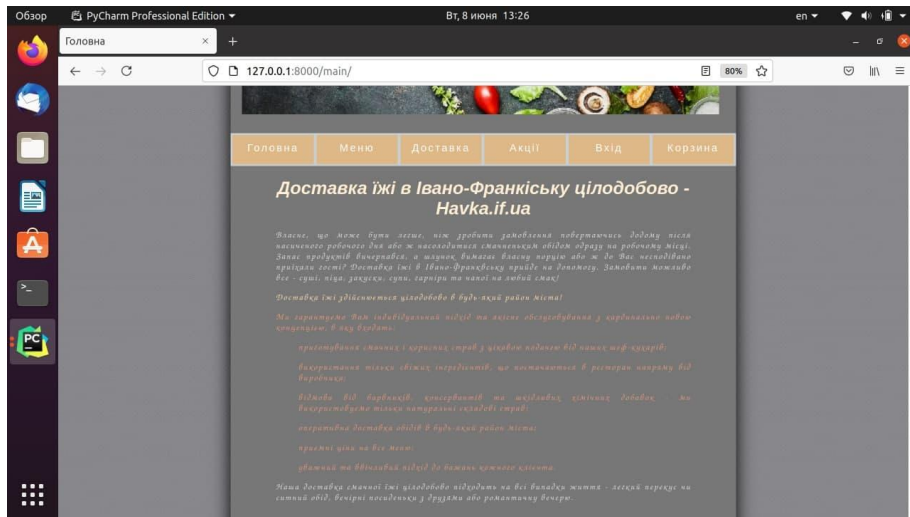


Рисунок 3.5 – Головна сторінка сайту

Шаблон menu.html (рис. 3.6) відображає сторінку, на якій розміщено МЕНЮ з їжею та напоями:

```
{% extends 'base.html' %}
{% load static %}
{% block title %}
    <title>Меню</title>
{% endblock %}
{% block background %}
    <div class="banner" id="p"></div>
{% endblock %}
{% block content %}
```

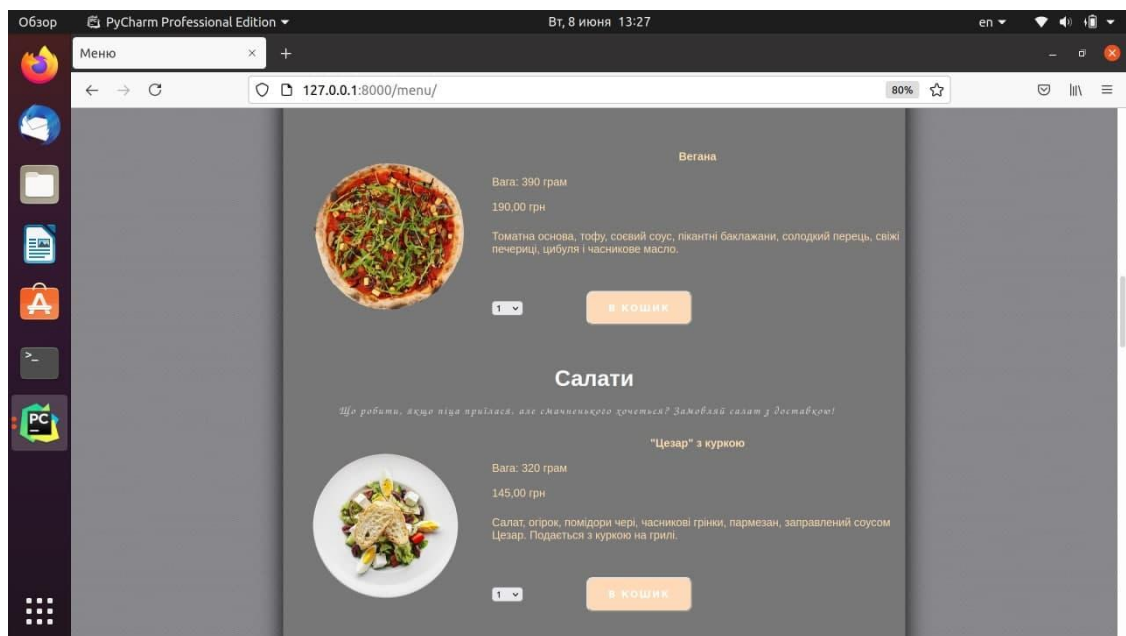


Рисунок 3.6 – Меню

Шаблон order.html (рис. 3.7) використовується для створення замовлення:

```

{% extends "base.html" %}
{% load static %}
<title>Замовлення</title>
{% endblock %}
{% block background %}
<div class="banner" id="p"></div>
{% endblock %}
{% block content %}

```

Виведення інформації на сторінку відбувається через теги.

```

<ul>
{% for item in cart %}
<li><h4>Назва товару: {{ item.name }}</h4></li>
<li><h4>Кількість: {{ item.quantity }}</h4></li>
{% endfor %}

```

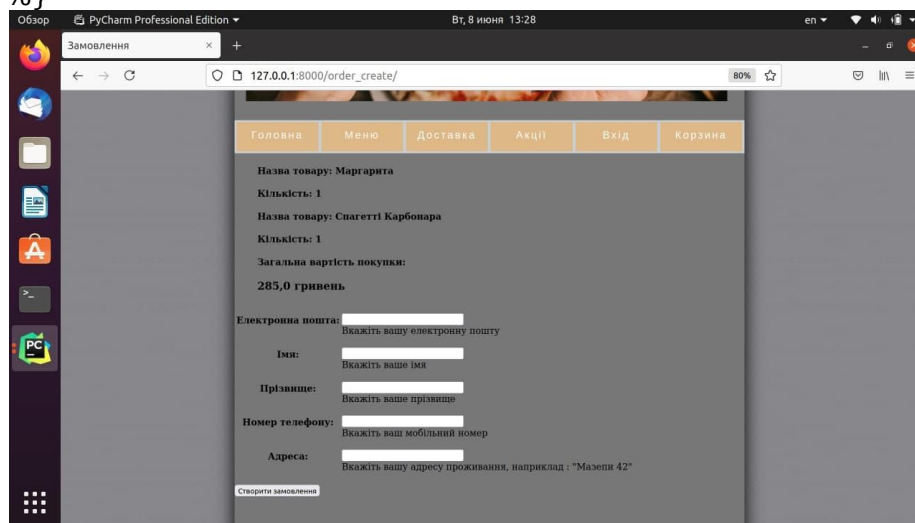


Рисунок 3.7 – Сторінка створення замовлення

Шаблон promotions.html(рис. 3.8) відображає сторінку із акціями закладу:

```

{% extends 'base.html' %}
{% load static %}
{% block title %}
<title>Акції</title>
{% endblock %}
{% block background %}
<div class="banner" id="p"></div>
{% endblock %}
{% block content %}

```

По замовчуванню значення буде використано значення яке вписане в блоці базового шаблону.

```

{% load static %}
{% block title %}
<title>Замовлення</title>
{% endblock %}

```

```
{% block background %}
```

Для кожної сторінки головне фото заднього фону міняється

```
<div class="banner" id="p"></div>
{% endblock %}
{% block content %}
{% endblock %}
```

Для виведення інформації на сторінку було використано цикл, за допомогою тегів Django.

```
{% for prom in prom_list %}
  <tr><td rowspan="4"></td>
```

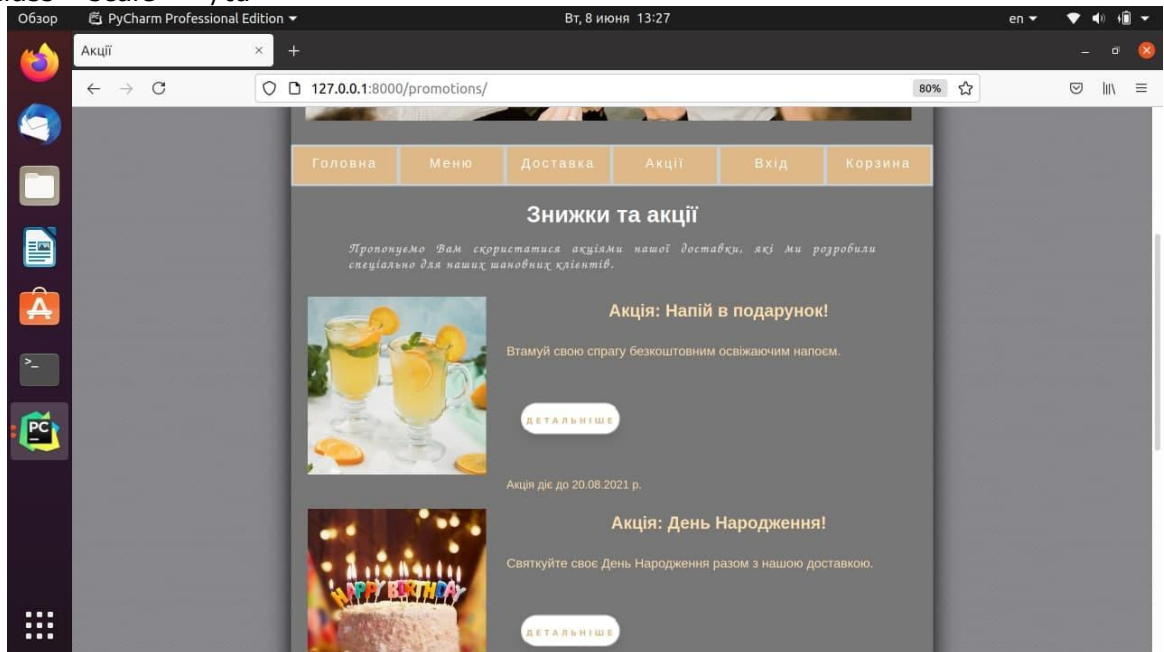


Рисунок 3.8 – Сторінка із акціями закладу

Шаблон login.html (рис. 3.9) відображає сторінку для автентифікації користувачів на сайт, якщо користувач не пройде автентифікацію, жодна сторінка йому не відкриється і буде перенаправлено на сторінку автентифікації або реєстрації.

```
{% extends 'base.html' %}
{% block title %}
  <title>Вхід</title>
{% endblock %}
{% block background %}
  <div class="banner" id="p"></div>
{% endblock %}
```

```
{% block content %}
```

На сторінці автентифікації є переходи на сторінку реєстрації та кнопку яка виходить із сесії, після чого потрібно пройти автентифікацію повторно.

```
{% extends 'base.html' %}
{% load static %}
{% block title %}
    <title>Register</title>
{% endblock %}
<form action="." method="POST">
    {% csrf_token %}
    <table>{{ form }}</table>
    <br>
    <p><input name="guruweba_example_submit" type="submit"
class="sliding-button" value="Зареєструватись"/></p>
```

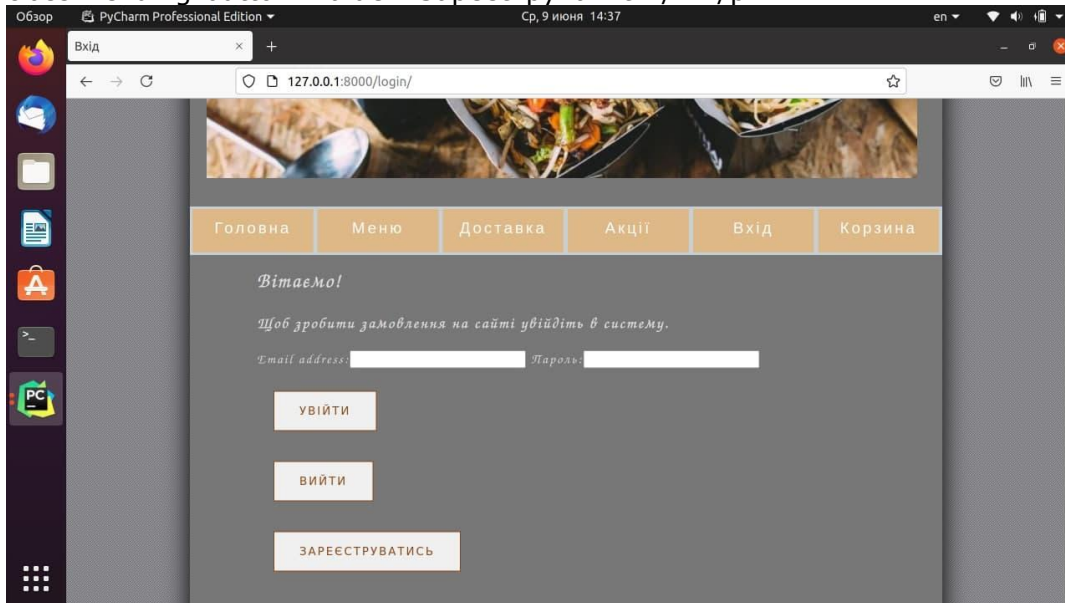


Рисунок 3.9 – Сторінка автентифікації

Шаблон register.html(рис. 3.10) призначений для сторінки реєстрації:

```
{% extends 'base.html' %}
{% load static %}
{% block title %}
    <title>Вхід</title>
{% block background %}
    <div class="banner" id="p"></div>
{% endblock %}
{% block content %}
```

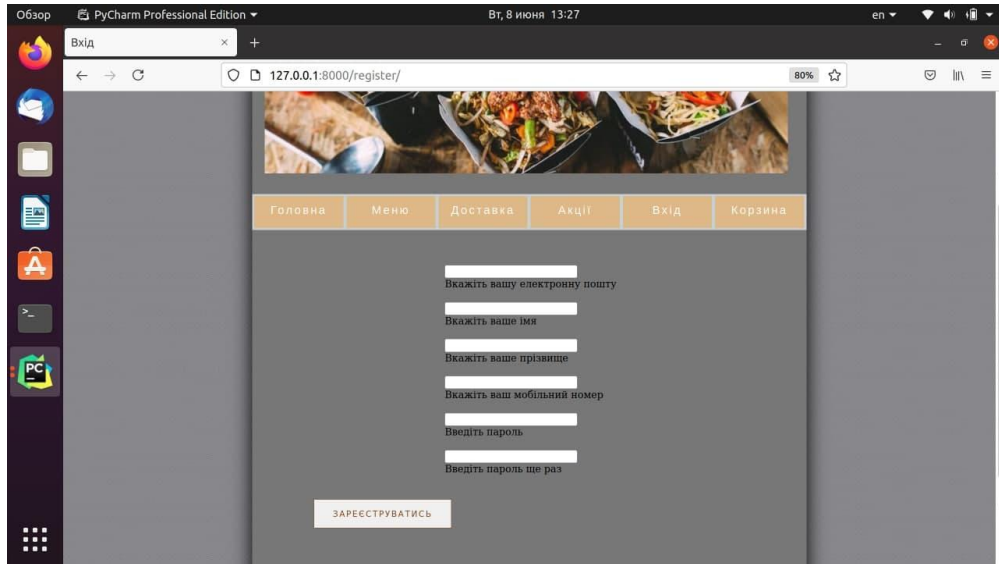


Рисунок 3.10 – Сторінка реєстрації

Висновки до розділу 3

В розділі було оглянуто структуру проекту, яка застосовується для сайту, розробку функціоналу на мові Python, та інтерфейсу web – застосунку. Було показано який результат видає web – застосунок.

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1 Умови праці на робочому місці з ПК

Навчальний процес з використанням комп'ютера, відбувається в середовищі, яке в певній мірі впливає на функціональний стан студентів, які перебувають у комп'ютерній аудиторії. Найважливішими несприятливими факторами середовища при роботі за комп'ютером є:

фізичні параметри мікроклімату, освітлення, електромагнітні випромінювання різних частотних діапазонів; виробничий шум та вібрація; іонізація повітря; статична електрика;

психофізіологічні – перенапруження зорового аналізатора; недостатня рухома діяльність; нервово-емоційне напруження.

Аналіз умов праці виконується для приміщення аудиторії №419 яка знаходиться, на 4 поверсі п'ятиповерхової будівлі.

Відповідно до НПАОП 0.00-1.28-10 є неприпустимим розташування приміщень, призначених для роботи з ВДТ у підвалах та цокольних поверхах. Також забороняється розташування вибухонебезпечних приміщень категорії А і Б (ОНТП 24-86) та виробництв з мокрими технологічними процесами поряд з приміщенням, де розташовуються ЕОМ (ПЕОМ), а також над такими приміщеннями, або під ними. Окрім того, виробничі приміщення для роботи з ВДТ не повинні межувати з приміщеннями, у яких рівень шуму та вібрації перевищує допустимі значення.

У процесі роботи з комп'ютером необхідно дотримувати правильний режим праці і відпочинку. У протилежному випадку в працюючого за ПК відзначаються значна напруга зорового апарата з появою скарг на незадоволеність роботою, головні болі, дратівливість, порушення сну, втома і хворобливі відчуття в очах, у попереку, в області шиї і руках.

					КР.ІПЗ – 10.00.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		45

Розташування робочого місця особи, що працює з ПК, в даному випадку студента, повинне відповідати вимогам ДСТУ 22.269-79 «Робоче місце оператора. Взаємне розташування елементів робочого місця».

Облаштування робочого місці повинно забезпечувати:

- правильне розміщення робочого місця у виробничому приміщенні;
- належні умови освітлення приміщення і робочого місця, відсутність відблисків;
- належні ергономічні характеристики основних елементів робочого місця;
- характер та особливості трудової діяльності.

Для зменшення втоми, місця користувачів ЕОМ мають бути зручними.

Конструкція робочого місця користувача ПК, (при роботі сидячи) має забезпечувати підтримання оптимальної робочої пози з такими ергономічними характеристиками:

- ступні ніг – на підлозі або на підставці для ніг;
- стегна – в горизонтальній площині;
передпліччя – вертикально;
- лікті – під кутом 70-90 до вертикальної площини;
- зап'ястя зігнуті під кутом не більше 20 відносно горизонтальної площини;
- нахил голови – 15-20 відносно вертикальної площини.

Робоче місце користувача ПК, обладнується робочим столом, стільцем і підставкою для ніг. Висота робочого стола має бути в межах від 0,65 до 0,8 м, а ширина повинна забезпечувати можливість виконання операцій в зоні досяжності моторного поля.

Клавіатуру слід розташовувати на поверхні столу на відстані 200 мм від краю, звернутого до працюючого. У конструкції клавіатури має передбачатися опорний пристрій (виготовлений із матеріалу з високим коефіцієнтом тертя, що перешкоджає його переміщенню), який дає змогу змінювати кут нахилу поверхні клавіатури у межах 5...15.

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		46

4.2 Мікроклімат та ступінь іонізації повітря у приміщенні

Як фактор виробничого середовища, мікроклімат впливає на теплообмін організму людини з цим середовищем. Необхідною умовою життєдіяльності людини є терморегуляція, тобто здатність організму регулювати віддачу тепла в оточуюче середовище. Цей процес визначається параметрами мікроклімату.

Метеорологічні умови визначаються такими параметрами:

- температурою повітря в приміщенні, С;
- відносною вологістю повітря, %;
- рухливістю повітря, м/с;
- тепловим випромінюванням, Вт/м³.

Принцип нормування мікроклімату – створення оптимальних умов для теплообміну тіла людини з навколишнім середовищем.

Параметри мікроклімату, вміст шкідливих речовин на робочих місцях, оснащених моніторами, відповідають вимогам ДСН 3.3.6.042-99 «Державні санітарні норми мікроклімату виробничих приміщень», ДСТУ 12.1.005-88 «ССБТ. Повітря робочої зони. Загальні санітарно-гігієнічні вимоги до повітря робочої зони».

Обчислювальна техніка є джерелом істотних тепловиділень, що може привести до підвищення температури і зниження відносної вологості в приміщенні.

В приміщеннях, де встановлені комп'ютери, повинні дотримуватися оптимальні параметри мікроклімату, які визначають комфортні умови. Ці параметри залежать від періоду року, категорії робіт за важкістю, і від теплових характеристик виробничого приміщення (табл. 4.1).

Робота за комп'ютером характеризується малими фізичним навантаженнями, цей вид діяльності належить до категорії легких робіт-Ia за критерієм загальних енерговитрат організму (ДСН 3.3.6.042-99).

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		47

Таблиця 4.1

Параметри мікроклімату для приміщень

Період року	Категорія робіт	Параметр мікроклімату	Величина
Холодний	Роботи легкі I а	Температура повітря в приміщенні	22÷24°C
		Відносна вологість	40÷60%
		Швидкість руху повітря	до 0,1м/с
Теплий		Температура повітря в приміщенні	23÷25°C
		Відносна вологість	40÷60%
		Швидкість руху повітря	0,1,0,2м/с

Під час роботи комп'ютерної техніки в повітряному середовищі відбувається суттєва трансформація іонного складу, істотно знижується концентрація легких, середніх та важких негативно зарядних частинок. Така зміна балансу іонного складу призводить до негативного впливу на здоров'я працюючих.

Рівні іонізації повітря приміщень при роботі на персональних комп'ютерах визначені в (таблиці 4.2) (відповідно до НПАОП 0.03-3.06-80).

Таблиця 4.2

Рівні іонізації повітря в приміщенні з ПК

Рівні	Кількість іонів в 1 см ³ повітря	
	n+	n-
Мінімально необхідні	400	600
Оптимальні	1500-3000	3000-5000
Максимально допустимі	50000	50000

Для підтримки оптимальних значень мікроклімату та підтримання нормальної концентрації позитивних та негативних іонів в приміщенні

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
						48
Зм.	Арк.	№ докум.	Підпис	Дата		

аудиторії пропонується удосконалити системи опалення, природної вентиляції та встановити кондиціонер.

Дисплеї на основі ЕПТ є потенційним джерелом випромінювання кількох діапазонів електромагнітного спектра: рентгенівського, оптичного, радіочастотного. Кожний вид випромінювання відрізняється своїми особливими характеристиками впливу на організм людини.

Рентгенівське випромінювання. Дослідження показують, що відео термінал не несе небезпеки для користувача ПЕОМ, оскільки інтенсивність такого випромінювання нижча за гранично допустимі норми (ГДН). Відповідно до «Норм радіаційної безпеки України» (НРБУ-97) гранично допустима потужність експозиційної дози рентгенівського випромінювання на відстані 5 см від поверхні екрана відео терміналу становить $7,74 \cdot 10^{-12}$ Кл/кг, що відповідає еквівалентній дозі 0,1 мбер/год. (100 мкР/год.).

Оптичне випромінювання. Оптичні види випромінювання виникають завдяки взаємодії електронів з шаром люмінофору, нанесеного на екран ВДТ. Область оптичного випромінювання включає ультрафіолетове (УФ), світлове та інфрачервоне (ІЧ) випромінювання.

УФ-випромінювання впливає на шкіру та очі людини. Такий вплив на шкірі проявляється досить швидко, а для очей характерним є період прихованої дії. Рівень УФ - випромінювання, який був виявлений, досить низький і становить 1 середньому $0,001 \text{ Вт/м}^2$.

Світлове випромінювання впливає в основному на око і призводить до втоми очей, запалення райдужної оболонки. Однак ці симптоми швидко минають і не викликають патологічних змін.

ІЧ-випромінювання - довжина хвиль обмежена від 0,76мм до 1мм. Для більшості біологічних матеріалів випромінювання цього діапазону вважаються непрозорими. Інтенсивність інфрачервоних випромінювана нижча за значення, передбачені ДсанПіН 3.3.2. - 007-98.

Електромагнітні випромінювання (ЕМВ) радіочастотного діапазону. Джерелом ЕМВ є відео термінал. Тому, обираючи робоче місце для комп'ютера,

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		49

необхідно пам'ятати, що його задня і бокові стінки можуть бути джерелом значно більшого ЕМВ, ніж екран.

З метою профілактики несприятливого впливу електромагнітного випромінювання від ВДТ на користувача необхідно:

- встановити на робочому місці відео термінал, що відповідає сучасним вимогам стосовно захисту від випромінювані;
- не переобтяжувати приміщення значною кількістю робочих місць з ВДТ;
- не концентрувати на робочому місці великої кількості радіо електронних пристроїв. Гранично допустимі рівні випромінювань на робочих місцях з ПК наведено в (таблиці 4.3).

Таблиця 4.3

Гранично допустимі рівні випромінювань

Вид випромінювання	Діапазон хвиль (частот)	Гранично допустимий рівень
Іонізуюче випромінювання		
М'яке рентгенівське випромінювання (на відстані 0,05 м від екрана та корпусу ВДТ)	0,01 – 1 нм	100мкР/год
Оптичні випромінювання		
Ультрафіолетове випромінювання	315 – 400 нм	10 Вт/м ²
Видиме випромінювання (яскравість)	400 – 700 нм	1000 кд/м ²
Інфрачервоне випромінювання	700 нм – 1мм	100 Вт/м ²
Електромагнітні випромінювання (поля радіочастотного та низькочастотного діапазонів)		

Продовження таблиці 4.3

Напруженість електромагнітного поля на відстані 0,5 м навколо монітору за електричною складовою	2 кГц – 400 кГц 5 Гц – 2 кГц	2,5 В/м 25 В/м
Щільність магнітного потоку	2 кГц – 400 кГц 5 Гц – 2 кГц	25 нТл 250 нТл
Електростатичні поля		
Поверхневий електростатичний потенціал	–	500 В

Відомо, що шум несприятливо діє на слуховий аналізатор та інші органи та системи організму людини. Визначальне значення щодо такої дії має інтенсивність шуму, його частотний склад, тривалість щоденного впливу, індивідуальні особливості людини, а також специфіка виробничої діяльності. Ті види діяльності, у яких поєднується напружена розумова робота та інтенсивне використання комп'ютера (редагування тексту, верстка оригіналу, "запуск" та відлагодження програм тощо) характеризується відчутним впливом навіть незначних рівнів шуму. Цей вплив виражається у зниженні розумової працездатності, швидкій втомлюваності, послабленні уваги, появі головного болю та ін.

Рівні звукового тиску в октавних смугах частот, рівні звуку та еквівалентні рівні звуку на робочих місцях, обладнаних ВДТ і ПК визначені ДСанПіН 3.3.2-007-98 не повинні перевищувати 65 дБА. Для приміщення аудиторії, рівень шуму не перевищує 60 дБА.

Для забезпечення нормованих рівнів шуму у виробничих приміщеннях та на робочих місцях застосовуються шумопоглинальні засоби, вибір яких обґрунтовується спеціальними інженерно-акустичними розрахунками. Рівні вібрації під час виконання робіт з ЕОМ у виробничих приміщеннях не повинні перевищувати допустимих значень - 70 дБ.

Для зниження вібрації обладнання, пристрої, пристосування необхідно встановлювати на спеціальні амортизуючі прокладки, передбачені документами.

Близько 90% всієї інформації, що отримується людиною, приходиться на органи зору. Організація освітленості робочих місць грає велику роль у житті людини. Недостатнє та нераціональне освітлення веде до стомлення очей, розладу центральної нервової системи, зниженню розумової та фізичної працездатності, а у ряді випадків може бути причиною травматизму (близько 5% травм приходиться на частку нераціонального та недостатнього освітлення).

Щоб уникнути перевтоми, а також для профілактики професійних захворювань та виробничого травматизму потрібно дотримуватись наступних вимог:

- створювати на робочій поверхні освітленість, що відповідає характеру зорової роботи і не є нижчою за встановлені норми;
- забезпечити достатню рівномірність та постійність рівня освітленості у виробничих приміщеннях, щоб уникнути частоті преадаптації органів зору;
- не створювати засліплюваної дії як від самих джерел освітлення, так і від інших предметів, що знаходяться в полі зору;
- не створювати на робочій поверхні різких та глибоких тіней (особливо рухомих);
- повинен бути достатній для розрізнення деталей контраст поверхонь, що освітлюються;
- не створювати небезпечних та шкідливих виробничих чинників (шум, теплові випромінювання, небезпека ураження струмом, пожежо та вибухонебезпека світильників);
- повинно бути надійним і простим в експлуатації, економічним та естетичним .

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		52

Приміщення аудиторії №419 має природну та штучну системи освітлення, які повинні відповідати вимогам ДБН В.2.5-28-2996 «Інженерне обладнання будинків та споруд. Природне та штучне освітлення».

Штучне освітлення в приміщенні з комп'ютеризованим робочим місцем здійснюється системою загального освітлення.

Як джерела світла в разі штучного освітлення застосовуються світильники серії ЛПО353. Світильники укомплектовані високочастотними пускорегулювальними апаратами (ВЧ ПРА). Яскравість світильників загального освітлення в зоні кутів випромінювання від 50 до 90 град. з вертикаллю в повздовжній та поперечній площинах становить не більше 200 кд/м, захисний кут світильників – не менше ніж 40 град. Рівень освітленості у аудиторії, $E = 300 \text{лк}$.

Електробезпека – це система організаційних та технічних заходів і засобів, що забезпечують захист людей від шкідливого та небезпечного впливу електричного струму, електричної дуги, електромагнітного поля і статичної електрики.

Приміщення аудиторії № 419 відноситься до приміщень з підвищеною безпекою.

ЕОМ, периферійні пристрої ЕОМ та устаткування для обслуговування, ремонту та налагодження ЕОМ, електропроводи та кабелі за виконанням та ступенем захисту мають відповідати класу зони за ПВЕ, мати апаратуру захисту від струму короткого замикання та інших аварійних режимів.

Використання нульового робочого провідника як нульового захисного провідника забороняється.

Нульовий захисний провід прокладається від склейки групового розподільчого щита до розеток живлення.

Не допускається підключення на щиті до одного контактного затискача нульового робочого та нульового захисного провідників.

Площа перерізу нульового робочого та нульового захисного провідника в груповій три провідній мережі повинна бути не менше площі перерізу фазового

					КР.ІПЗ – 10.00.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		53

провідника. Усі провідники повинні відповідати номінальним параметрам мережі та навантаження, умовам навколишнього середовища, умовам розподілу провідників, температурному режиму та типам апаратури захисту, вимогам ПВЕ.

З метою підвищення рівня електробезпеки в приміщенні аудиторії № 419 неприпустимим є:

- експлуатація кабелів та проводів з пошкодженою або такою, що втратила захисні властивості за час експлуатації, ізоляцією;
- залишення під напругою неізольованих кабелів та проводів;
- застосування саморобних подовжувачів, які не відповідають вимогам ПВЕ до переносних електропроводок;
- користування пошкодженими розетками, розгалужувальними та з'єднувальними коробками, вимикачами та іншими електровиробами;
- застосування для опалення приміщення нестандартного (саморобного) електронагрівального обладнання.

ВДТ є джерелом електростатичних зарядів. Тривале перебування в електростатичному полі, створеному цими зарядами, негативно впливає на здоров'я працюючих: бронхо-легеневі захворювання, порушення серцево-судинної та нервової систем, ураження шкіри тощо .

Напруженість електростатичного поля на робочих місцях, в тому числі й з ВДТ, не повинна перевищувати 20 кВ/м відповідно до ГОСТ 12.1.045-84 "ССБТ. Електростатичні поля. Допустимі рівні на робочих місцях і вимоги до проведення контролю", поверхнево електростатичній Потенціал відео термінала не повинні перевищувати 500В.

Для запобігання створенню значної напруженості поля та захисту від статичної електрики необхідно:

- встановити нейтралізатори статичної електрики;
- підтримувати в приміщенні з ВДТ відносну вологість повітря не нижче 45-50% (чим сухіше повітря тим більше електростатичних зарядів); можна для цього використати навіть побутові зволожувачі;

					КР.ІПЗ – 10.00.000 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		54

- застелити підлогу в приміщеннях з ВДТ антистатичним лінолеумом і проводити щоденне вологе прибирання;
- складати всі полімерні покриття (чохли) ВДТ у найбільш віддаленому від користувачів місці розміщення;
- протирати екран та робоче місце спеціальною антистатичною серветкою або зволоженою тканиною.

4.3 Вимоги до безпечної експлуатації комп'ютерної техніки

Пожежна безпека при роботі за комп'ютером повинна забезпечуватись у відповідності з вимогами Закону України «Про пожежну безпеку» та НПАОП 0.01-1.01-95 «Правил пожежної безпеки в Україні» та інших нормативних документів.

Відповідно до ОНТП 24-86 «Визначення категорій приміщень і будинків по вибух пожежної і пожежної небезпеки» приміщення аудиторії № 419 відноситься до категорії Д.

Найбільш імовірною причиною виникнення пожеж в приміщенні є порушення вимог при експлуатації комп'ютерної техніки та займання електропроводки внаслідок коротких замикань.

Для того щоб уникнути виникнення пожежі, потрібно дотримуватися наступних заходів:

- дотримання правил пожежної безпеки при роботі з комп'ютером,
- електрообладнанням та освітлювальними приладами;
- періодичний контроль цілісності і надійності електроізоляції;
- наявність інструкцій з пожежної безпеки;
- навчання, атестація і переатестація персоналу з пожежної безпеки;
- наявність системи захисту від атмосферної електрики;
- періодичне зняття зарядів статичної електрики;
- заборона куріння в приміщенні.

					КР.ІПЗ – 10.00.000 ПЗ	Арк.
						55
Зм.	Арк.	№ докум.	Підпис	Дата		

- застосування будівельних конструкцій із ступенем вогнестійкості не нижче II, а також використання важко горючих або негорючих матеріалів в інтер'єрі виробничого приміщення;
- наявність схеми евакуації;
- наявність пристроїв автоматичного вимкнення ПЕОМ та іншого електроустаткування на випадок пожежі;
- наявність первинних засобів пожежогасіння (вогнегасник ВВК-5 - 2шт).

Висновки до розділу 4

Працюючи за комп'ютером, рекомендовано дотримуватися правил тривалості роботи, правильної постави, розміру шрифтів та зображень, вимог до приміщення тощо. Якщо притримуватись всіх необхідних норм, то робота за ПК стане не тільки зручною, а ще й ефективнішою та не задасть шкоди користувачеві.

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
						56
Зм.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи було розроблено web-застосунок за допомогою мови програмування Python та вебфреймворку Django з продажу їжі та напоїв з повним функціоналом як для клієнтів, так і для адміністраторів сайту.

Програмний продукт не вимогливий до системних ресурсів комп'ютера та швидкості інтернету. Застосунок в подальшому може вдосконалюватись та оновлювати свій функціонал.

					КР.ІПЗ – 10.00.000 ІЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Python *Wikipedia*: вебсайт. URL: uk.wikipedia.org/wiki/Python
2. Django *Wikipedia*: веб-сайт. URL: uk.wikipedia.org/wiki/Django
3. SQLite *Wikipedia*: веб-сайт. URL: uk.wikipedia.org/wiki/SQLite
4. HTML *Wikipedia*: веб-сайт. URL: uk.wikipedia.org/wiki/HTML
5. CSS *Wikipedia*: веб-сайт. URL: uk.wikipedia.org/wiki/CSS
6. Створення проекту *Metanit*: веб-сайт. URL: metanit.com/python/django/4.php
7. Розширення шаблонів *Metanit*: веб-сайт. URL: metanit.com/python/django/2.6.php
8. Rozetka *Rozetka*: веб-сайт. URL: rozetka.com.ua/ua/pages/beachtime_2021/
9. UML діаграми *Evergreens*: веб-сайт. URL: evergreens.com.ua/ua/articles/uml-diagrams.html
10. Use case *Dou*: веб-сайт. URL: dou.ua/lenta/articles/use-cases/
11. MVT архітектура *Wikipedia*: вебсайт. URL: en.wikipedia.org/wiki/MVT
11. Django documentation *Djangoproject*: вебсайт. URL: docs.djangoproject.com/en/3.2/
12. Python documentation *Python*: вебсайт. URL: python.org/doc/
13. SQLite documentation *Sqlite*: вебсайт. URL: sqlite.org/docs.html
14. CSS documentation *Snippets*: вебсайт. URL: css-tricks.com/snippets/css/a-guide-to-flexbox/
15. HTML documentation *w3schools*: вебсайт. URL: w3schools.com/tags/tag_doctype.asp

					КР.ІПЗ – 10.00.000 ІПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		58



					КР.ІПЗ – 10.00.000 ІЗ	Арк.
						59
Зм.	Арк.	№ докум.	Підпис	Дата		