**Міністерство освіти і науки України**

**ОДЕСЬКА НАЦІОНАЛЬНА АКАДЕМІЯ ЗВ’ЯЗКУ ім. О.С. ПОПОВА**

**Навчально-науковий інститут інфокомунікацій та програмної інженерії**

**Кафедра комп’ютерних наук**

***Пояснювальна записка***

до дипломної роботи

бакалавра

**НА ТЕМУ:**

Розробка веб-орієнтованої системи автоматизації діяльності ресторану

**Виконав**: студент 4 курсу, групи ПІ-4.2.01

Напрямку підготовки (спеціальності): \_\_\_\_

121. Інженерія програмного забезпечення

Кукол М.С.

**Керівник:** Вороной С.М.

**Рецензент**: ?

**Одеса-2019**

**РЕФЕРАТ**

Текстова частина пояснювальної записки до дипломної роботи: XXX с., XX табл., X додаток.

Об’єкт дослідження – система автоматизації діяльності ресторану.

Метод дослідження – прикладний.

Мета роботи – розробка та програмна реалізація веб-орієнтованої системи автоматизації діяльності ресторану.

У роботі проаналізовано предметну область веб-орієнтованої системи автоматизації діяльності ресторану. Для даної системи були поставлені основні завдання, такі як створити, змінити та видалити замовлення, зберегти ці дані у базі даних та відображати їх коли це буде необхідно, зареєструвати постійних клієнтів, змінювати стоп-лист та/або чек-лист, друкувати чек, додавати знижку. Веб-додаток повинен легко масштабуватися для легкого та швидкого додавання додаткового функціоналу.

Ключові слова: ресторан, пошук страви, замовлення, веб-сервіс, знижка, стіл, стоп-лист, чек-лист, гість, персонал, офіціант, адміністратор.

ЗМІСТ

[ВСТУП 4](#_Toc9380401)

[ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ 5](#_Toc9380402)

[1 ПОСТАНОВКА ЗАДАЧІ ДЛЯ СИСТЕМИ АВТОМАТИЗАЦІЇ ДІЯЛЬНОСТІ РЕСТОРАНУ 6](#_Toc9380403)

[2 ОБҐРУНТУВАННЯ ЗАСОБІВ ТА ІНСТРУМЕНТІВ РОЗРОБКИ ВЕБ-ОРІЄНТОВАНОЇ СИСТЕМИ 7](#_Toc9380404)

[3 ВИМОГИ ДО ВЕБ-ОРІЄНТОВАНОЇ СИСТЕМИ 10](#_Toc9380405)

[ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ 11](#_Toc9380406)

[ВИКОРИСТАНІ ДЖЕРЕЛА 12](#_Toc9380407)

[ДОДАТОК А. СПИСОК РОЗВ’ЯЗУВАНИХ ЗАВДАНЬ 14](#_Toc9380408)

[ДОДАТОК Б. 16](#_Toc9380409)

# ВСТУП

Актуальність даної роботи полягає в тому, що створення системи автоматизації діяльності ресторану дозволить персоналу використовувати свій смартфон чи комп’ютер для створення замовлення клієнтами. Зараз майже усі ресторани перейшли на електронну систему стеження за замовленнями, але вони використовують тільки стаціонарні комп’ютери для реалізації. На даний момент мобільні технології розвиваются швидше за стаціонарні, та набирають високу популярність. У цій же роботі система розрахована для її використання на мобільних пристроях персоналу, а саме на смартфонах та електронних планшетах.

Об'єктом дослідження в даній роботі є система діяльності ресторану.

Предметом дослідження є функціонал, який має реалізувати дана система.

Метою даної роботи є проектування і реалізація інформаційної системи стоврення, зміни замовлень клієнтами ресторану та стеження за цими діями адміністратором чи директором.

Для досягнення даної мети необхідно вирішити такі завдання:

1. виконати аналіз предметної області, скласти завдання, які необхідно вирішити;
2. розробити архітектуру системи;
3. розробити базу даних;
4. реалізувати користувацький інтерфейс

# ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

1. MVC – Model-View-Controller
2. БД – База даних
3. SQL – Structured Query Language
4. HTML – HyperText Markup Language
5. CSS – Cascading Style Sheets

# 1 ПОСТАНОВКА ЗАДАЧІ ДЛЯ СИСТЕМИ АВТОМАТИЗАЦІЇ ДІЯЛЬНОСТІ РЕСТОРАНУ

У результаті дослідження предметної області автоматизації діяльності ресторану були виділені такі користувачі:

* Адміністратор
* Офіціант
* Повар

# 2 ОБҐРУНТУВАННЯ ЗАСОБІВ ТА ІНСТРУМЕНТІВ РОЗРОБКИ ВЕБ-ОРІЄНТОВАНОЇ СИСТЕМИ

Система автоматизації діяльності ресторану розроблена у вигляді веб-додатку. Для роботи з базою даних обрана система управління базами даних PgAdmin третьої версії. Мовою розробки бази даних є PostgreSQL. Для розробки серверної частини додатку використана мова програмування Python та безкоштовний фреймворк для веб-додатків Django, що використовує шаблон проектування MVC. Видима для кінцевого користувача частина додатку розроблена за допомогою мови розмітки HTML та формальної мови опису зовнішнього вигляду веб-сторінки CSS.

PostgreSQL – найдосконаліша в світі мова розробки реляційних баз даних з відкритим вихідним кодом [1]. Версія використана для розробки проекту – 10.0. Вона підтримує більшу частину стандарту SQL і пропонує багато сучасних функцій, таких як:

* Типи даних, функції та оператори
* Індексація та обмеження
* Мова визначення даних
* Розбиття та успадкування
* Продуктивність
* Уявлення та матеріалізовані уявлення
* Безпека даних
* Інтернаціоналізація
* Додаткові модулі
* Користувацькі функції, збережені процедури та тригери
* Резервне копіювання, відновлення, цілісність даних і реплікація

Серед недоліків PostgreSQL, яких немає у його конкурентів, можна виділити такі:

* Для роботи з PostgreSQL на рівні продакшена потрібно вміти професійно його налаштувати. У випадку, якщо його налаштовано інакше – він буде працювати повільно.
* PostgreSQL – не є популярною мовою розробки баз даних, що сприяє повільному розвитку в спільноті.
* Хостинги веб-сайтів, що мають низькі ціни, не полюбляють PostgreSQL через те, що його складно адмініструвати.

Але ці недоліки перекриваються його перевагами:

* Вкладені запити можуть мати свою назву і можуть бути записані окремо у рамках одного запиту до бази даних
* Віконні функції. Віконні функції виконують обчислення для набору кортежів, деяким чином зв’язаних з поточним (або з іншим) кортежом. Їх можна порівняти з агрегатними функціями, але на відміну від других віконні функції не групують схожі кортежі.
* Збережені процедури можна створювати на різних мовах, не тільки на pl/pgsql (мова PostgreSQL) чи SQL. Наприклад: JavaScript, C чи Perl.
* Транзакційність. Вона дозволяє поєднати деяку послідовність дій в одну операцію. Тобто, якщо є проміжна дія у транзакції, що не виконалась, тоді й інші проміжні дії не збережуться до бази даних.
* PostgreSQL має дуже мало відомих на даний момент «багів» на відміну за інші мови.

PgAdmin III – найпопулярніша та багатофункціональна платформа для адміністрування і розробки з відкритим вихідним кодом для PostgreSQL. Вона може бути інстальована на такі операційні системи як Linux, FreeBSD, Solaris, macOS та Windows для управління версіями PostgreSQL від 9.2. PgAdmin має лише ті функції, що має PostgreSQL, але дозволяє їх використовувати зручніше.

Python – проста для вивчення та для читання коду мова програмування високого рівня, яка має відкритий вихідний код. Python – продуктивний та швидкий, використовується переважно для розробки веб-додатків, машинного навчання та для виконання складних і масивних математичних обчислень. Python може бути використаний на будь-якій операційній системі, добре працює разом з іншими мовами програмування. Python – об’єктно-орієнтовна мова програмування, що є великою перевагою на ринці мов програмування на сьогоднішній день. Він має свій продуктивний та зручний менеджер пакетів та модулів Python Package Index, у якому користувачі можуть викладати свої модулі у вільний доступ для інших. Python Package Index значно скорочує час розробки програмних продуктів у десятки або сотні разів [2]. Серед недоліків можна виділити лише декілька: він не призначений для розробки мобільних додатків та 3D-моделювання, Python важко справляється з мультипроцесорними та багатоядерними задачами.

Django – це безкоштовний глобальний веб-фреймворк високого рівня з відкритим вихідним кодом, написаний на мові програмування Python. Він сприяє високій швидкості розробки веб-додатку і чистому та прагматичному дизайну. Django розроблений досвідченими розробниками, він дозволяє сконцентруватися на розробці додатку без необхідності «створювати велосипед». Він розроблюється з 2005 року та підтримується донині, це означає, що він старший за інші фреймворки та більш розвинений. Django також ідеально підходить для розробки, якщо у проекті використовуються бази даних PostgreSQL. Для роботи з базами даних Django має вбудовану ORM-систему, що дозволяє оголошувати таблиці бази даних як класи, її стовпці як атрибути класу, а кортежі як об’єкти класу, що значно полегшує розробку додатку та підвищує швидкість розробки. За необхідністю можна не використовувати ORM-систему, а виконувати «сирі» запити[3]. У ньому є вбудований веб-сервер для запуску на різних пристроях: від локального пристрою до хостингу. Вбудований веб-сервер автоматично виявляє зміни у файлах проекту, та перезапускається для їх відображення. На стадії розробки сервер виводить ймовірні помилки як у терміналі, так і на самій сторінці веб-додатку. На стадії продакшену виведення помилок треба відключати, тому що зловмисник може скористатися вразливістю додатку та використати інформацію у базі даних у гірших цілях.

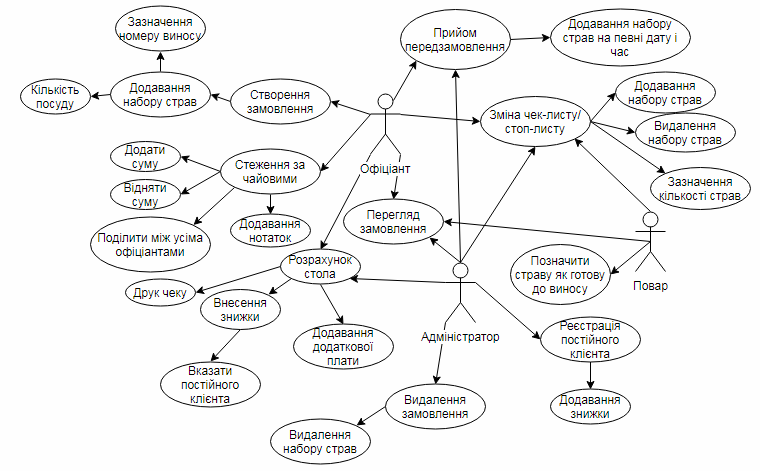
# 3 ВИМОГИ ДО ВЕБ-ОРІЄНТОВАНОЇ СИСТЕМИ

Оптимальна система має виконувати наступні задачі:

* Створення замовлення зробленого клієнтом: додавання набору страв, зазначення номеру виносу та кількість посуду;
* Перегляд замовлення;
* Прийом передзамовлення (додавання набору страв на певні дату і час);
* Зміна даних у чек-листі та стоп-листі: додавання страви, її видалення та зазначення кількості;
* Розрахунок стола: врахування знижки, додаткової плати та друк чеку;
* Додання та видалення знижки для певної страви;
* Видалення набору страв;
* Стеження за чайовими: додавання суми, видалення суми, розподіл між кількістю персон та додавання нотаток;
* Позначення страви як готової до виносу;
* Реєстрація та видалення постійного клієнта, додавання йому знижки.

Повний список головних завдань з описом знаходяться у додатку А.

Всі дійові особи та їх взаємодія з системою представлені на рисунку 3.1.

Рисунок 3.1 – визначення дійових осіб та їх взаємодія з системою

# ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ

В результаті виконання даної роботи проаналізована предметна область веб-орієнтованої системи діяльності ресторану. Для даної веб-системи були поставлені основні завдання, такі як створення, зміна, видалення та перегляд замовлення, зробленного клієнтом, розрахунок окремого клієнта, зміна чек-листу та стоп-листу, реєстрація та видалення постійних клієнтів, додавання приміток для замовлення чи клієнта, розрахунок чайових для персонала, встановлення бронювання для столу, введення додаткової плати за обслуговування та введеня знижки. Тема є актуальною, тому що існуючі системи такого типу не є зручними та досконалими.

Рекомендації. Ґрунтуючись на отриманих результатах, доцільно створити систему у вигляді веб-додатку. Система повинна реалізувати більшість функціоналу, перерахованого в даній роботі і легко масштабуватися. Також рекомендується після створення додатка викласти його на хостінг.

# ВИКОРИСТАНІ ДЖЕРЕЛА

1. PostgreSQL: The world’s most advanced open source database. [Електронний ресурс] – Режим доступу: <https://www.postgresql.org/>
2. PyPI – the Python Package Index ∙ PyPI. [Електронний ресурс] – Режим доступу: <https://pypi.org/>
3. Performing raw SQL queries | Django documentation | Django. [Електронний ресурс] – Режим доступу: <https://docs.djangoproject.com/en/2.2/topics/db/sql/#django.db.models.Manager.raw>

# ДОДАТОК А. СПИСОК РОЗВ’ЯЗУВАНИХ ЗАВДАНЬ

| **Задача** | **Підзадачі** | **Вхідні параметри** | **Вихідні параметри** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# ДОДАТОК Б.