**Технічні умови**

**до розробки електронної системи продажу піци**

**“Pizza$builder”**

**Київ 2021**

Зміст

1. Загальні відомості

1.1 Найменування замовника та набувача

1.2 Термін надання послуг

1.3 Порядок оформлення та пред’явлення Замовнику результатів

1. Призначення та цілі
2. Характеристики об’єкту та предмет автоматизації
3. Вимоги до електронної системи продажу піци “Pizza$builder”

4.1 Вимоги до Pizza$builder в цілому

4.2 Функціональні вимоги до Pizza$builder

4.2.1 Модуль “Вхід”

4.2.2 Модуль “Пошук шаблонів піц”

4.2.3 Модуль “Формування замовлення”

4.2.4 Модуль “Конструктор піци”

4.2.5 Модуль “Інформація про піцу”

4.2.6 Модуль “Корзина”

4.2.7 Модуль “Підтвердження замовлення”

4.3 Загальна архітектура рішення

4.4 Вимоги до інформаційного забезпечення

4.5 Вимоги до лінгвістичного забезпечення

4.6 Вимоги до прикладного програмного забезпечення

## 

## Терміни та визначення:

“Pizza$builder” - назва сервісу;

БД - База Даних;

# **1. Загальні відомості про зміст робіт**

### **1.1 Найменування замовника та користувача**

У рамках курсу дисципліни Java EE, потрібно написати веб-застосунок, використовуючи технології Java Enterprise Edition. Вирішено написати веб-застосунок “Pizza$builder” орієнтований на продаж піци з можливістю її конструювання.

Замовником сервісу є викладач курсу, Олег Курп’як.

Постачальником сервісу визначено тріо студентів: Першута Павло, Накитняк Вадим, Шутяк Таїсія.

### **1.2 Термін виконання проекту**

Встановлюється кінцем весняного навчального триместру (прибл. 30.04.2021).

### **1.3 Порядок оформлення та представлення Замовнику результатів**

Постачальник після завершення розробки сервісу передає Замовнику результати розробки, а саме:

- програмне забезпечення (в одному екземплярі, архів на пошту Замовника);

- посилання на реалізацію проекту на платформі GitLab.

# **2. Призначення та цілі**

Мета проєкту - розробка веб-застосунку, що надасть клієнтам можливість купувати італійську національну страву, піцу. Система дозволить підбирати піцу, базуючись на смакових вподобаннях клієнта, або клієнт самостійно зможе конструювати власну піцу з наявних на складі продуктів. Після отримання замовлення, клієнт зможе залишити відгук, оцінюючи смак піци та якість наданих послуг.

Проєкт призначений для:

- надання можливості користувачу знаходити піцу, що відповідає його вподобанням;

- надання можливості користувачу додати піцу у корзину і зробити замовлення;

- надання можливості користувачу створити піцу власноруч за власними вподобанням з наявних на складі продуктів.

# **3. Характеристика об’єкту**

У рамках створення сервісу проводиться надання користувачам послуг електронного сервісу по продажу піци та можливість конструювати піцу власноруч. Пропонуються процеси замовлення товарів на певний день та час.

# **4. Вимоги до веб-застосунку**

### **4.1 Вимоги до застосунку в цілому**

Для користувачів застосунок має представляти собою веб-сторінку, на якій користувач може отримати інформацію про наявні шаблони піц та можливість перейти у конструктор, де можна створити власний шаблон та зберегти його до вподобань, а також замовити її.

Застосунок повинен включати наступні компоненти:

- клієнтська частина (веб-сторінка);

- серверна частина.

В “Pizza$builder” мають бути наявні наступні функціональні можливості:

- авторизація користувачів;

- замовлення піци;

- конструювання власної піци;

- можливість скасувати замовлення;

- пошук піци за певними критеріями, наприклад: начинка, тісто і тд.

- поділ можливостей користувачів за ролями

Створений веб-додаток “Pizza$builder” повинен відповідати наступним вимогам:

- відповідати функціоналу інтерфейсу сучасним вимогам (інтерфейс повинен мати звичний та зручний для користувача вигляд і набір команд);

- адаптивний дизайн.

Доступ до застосунку зовнішніми користувачами повинен надаватись з автоматизованих робочих місць на базі персональних комп’ютерів, ноутбуків або КПК, підключених до мережі Інтернет. Для роботи з “Pizza$builder” зовнішнім користувачам необхідно мати встановлений та налаштований один із сучасних веб-браузерів.

Користувачем застосунку є будь-яка авторизована особа.

**Ролі користувачів:**

* Незареєстрований користувач;
* Зареєстрований користувач;
* Менеджер/адмін;
* супер адмін;

**Можливості:**

Незареєстрований користувач:

* Переглядати зміст сайту;
* Перегляд списку існуючих шаблонів піц;
* Перегляд детального шаблону піци;
* Перейти на сторінку логіну/реєстрації.

Зареєстрований користувач:

* Переглядати зміст сайту;
* Додавати товари в корзину;
* Створити власний шаблон піци з власною назвою;
* Додавати товари в корзину і робити замовлення;
* Додавати товари в “улюблене”/”побажання”;
* Залишати відгуки;
* Оцінювати існуючі шаблони.

Менеджер/адмін:

* Додавати нові шаблони;
* Затвердження доданих користувачами шаблонів;
* Видаляти шаблони та інгредієнти;
* Додавати інгредієнти до складу;
* Змінювати статус замовлення(зареєстроване/підтверджене/скасоване);
* Блокувати користувачів.

Супер адмін:

* Має весь функціонал менеджера;
* Додавати нових менеджерів;
* Видаляти менеджерів.

### **4.2 Функціональні вимоги до застосунку**

Застосунок розробляється по модульному принципу, тобто система складається з окремих модулів, кожен з яких реалізує певний набір функцій, притаманних виключно йому.

В “Pizza$builder” реалізуються наступні модулі :

- Модуль “Вхід”

- Модуль “Пошук шаблонів піц”

- Модуль “Формування замовлення”

- Модуль “Конструктор піци”

- Модуль “Інформація про піцу”

- Модуль “Корзина”

- Модуль “Підтвердження замовлення”

#### **4.2.1 Модуль “Вхід”**

Цей модуль призначений для початкового входу користувачі з вже встановленими правами.

#### **4.2.2 Модуль “Пошук шаблонів піц”**

Цей модуль призначений для фільтрації товарів за параметрами користувача. У модулі наявна можливість заповнення таких полів для пошуку піци:

- наповнення бортику;

- веганська/вегетеріанська/класична;

- пошук за інгредієнтами;

- сортування за рейтингом;

У цьому модулі наявні параметри, як обов`язкові, так і не обов'язкові для заповнення.

#### **4.2.3 Модуль “Конструктор піци”**

Цей модуль призначений для формування створення власної піци користувачем та надсилання її на обробку адміністраторам сервісу. У модулі потрібно заповнити такі поля для створення заявки на замовлення піци:

- Основний склад(м’ясо, овочі, сир).

- Соус(інгредієнти).

- Додаткові складники(ананаси, перец чилі/халапеньо + ще щось).

#### **4.2.4 Модуль “Інформація про піцу”**

Цей модуль призначений для короткого та змістовного ознайомлення користувача з товарами.

У модуля містяться наступні пункти:

- наповнення бортику(сир/майонез);

- веганська/вегетаріанська/класична;

- розмір small, medium, large, ultra;

#### **4.2.5 Модуль “Підтвердження заявки”**

Цей модуль призначено для підтвердження заявки, сформованої Модулем “Створення піци”

### **4.3 Загальна архітектура рішення**

“Pizza$builder” буде побудована за класичною трирівневою архітектурою: backend – middleware – client-side.

ІС повинна включати такі компоненти на відповідних рівнях:

1) На рівні БД

- сервер БД, який містить інформацію шаблонів піц та користувачів.

2) На рівні middleware

- сервер прикладного програмного забезпечення, який реалізовує основну бізнес-логіку “Pizza$Builder”.

3) На клієнтському рівні

- веб-сервер, який забезпечує доступ користувачів до особистих кабінетів.

### **4.4 Вимоги до інформаційного забезпечення**

Дані “Pizza$builder” повинні знаходитись у структурованому вигляді. Структура даних повинна задавати строгі правила зберігання інформації для точної ідентифікації її в “Pizza$builder”.

### **4.5 Вимоги до лінгвістичного забезпечення**

Інтерфейс користувачів ІС повинен бути виконаний українською мовою (можлива реалізація локалізації на інші мови).

### **4.6 Вимоги до прикладного програмного забезпечення**

Прикладне програмне забезпечення повинно базуватись на використанні вільно розповсюджуваного програмного забезпечення, яке не потребує придбання ліцензій, у разі некомерційного використання:

- Мова програмування Java та JavaScript;

- Spring Boot - фреймворк для Java;

- система керування БД - PostgreSQL;

- React front-end з material-ui бібліотекою компонентів;

- HTML + SASS;

- Система керування репозиторієм програмного забезпечення GitLab.