НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«Київський політехнічний інститут імені Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра програмного забезпечення комп’ютерних систем

**ТЕХНІЧНЕ ЗАВДАННЯ НА КУРСОВУ РОБОТУ**

***з дисципліни "Основи веб-програмування"***

***на тему:***

***“Онлайн-ресторан”***

Виконав: Симонюк Володимир Павлович

Група: КП83X

Затверджено

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1 семестр 2019/2020

Зміст

[**1. Вступ**](#_12r4nkjm57dj) **3**

[1.1. Найменування програми](#_dtk7203vfyyp) 3

[1.2. Призначення і галузь застосування](#_kfp74rqc1ndd) 3

[**2. Вимоги до програми**](#_48yike4eis0f) **4**

[2.1. Вимоги до функціональних характеристик](#_y117h0nlsn45) 4

[2.2. Вимоги до проекту](#_fynj9gq4057c) 6

[2.3. Вимоги до надійності](#_lm4knadnkex) 6

[2.3.1. Відмови через некоректні дії користувачів системи](#_4ec0v583vai6) 6

[**3. Умови експлуатації**](#_z7rd2mvrhnne) **7**

[3.1. Кліматичні умови експлуатації](#_8ru3y4l0f6pl) 7

[3.2. Вимоги до складу і параметрів технічних засобів](#_kzrp8z4in8mi) 7

[**4. Вимоги до програмної документації**](#_zj8ddb9vtnt) **7**

[4.1. Попередній склад програмної документації](#_t20y1sh0nh0i) 7

[**5. Техніко-економічні показники**](#_yagqop1zsu6s) **7**

[5.1. Економічні переваги розробки](#_gt5yhm1armsa) 7

[**6. Стадії і етапи розробки**](#_dkayaqec3yd6) **8**

[6.1. Стадії розробки](#_s2y4epsrd137) 8

[6.2. Етапи розробки](#_z4aghr8xz7f1) 8

[**7. Порядок контролю і приймання**](#_ryi73m74vtqz) **8**

[7.1. Види випробувань](#_cex10csyop7h) 8

[7.2. Загальні вимоги до приймання роботи](#_xg6l4p1jtw53) 9

# 1. Вступ

## 1.1. Найменування програми

Найменування програми: “Онлайн-ресторан”.

## 1.2. Призначення і галузь застосування

Програма призначена для створення, керування вмістом бази даних, що містить дані:

1. Користувачів системи
2. Відвідувачі ресторану
3. Офіціанти
4. Зали та столики
5. Меню

Програма надає веб-інтерфейс для зручного керування записами в базі даних:

1. Редагування адміністратором зал, столиків, та меню
2. Редагування меню, привязаного до столика, офіціантом який за ним закріплений, або відвідувачем, який його зайняв
3. Перегляд кухарем всіх замовлень

# 

# 2. Вимоги до програми

Перед входом у систему користувач проходить аутентифікацію.

Система повинна забезпечувати одночасну роботу з одним сервером користувачів з різними правами доступу.

## 2.1. Вимоги до функціональних характеристик

Програма має задовольняти такі вимоги:

1. Наявний веб-інтерфейс користувача. Графічні елементи використані відповідно до їхнього основного призначення.
   1. Наявно декілька сторінок або станів односторінкового сайту і веб-форм.
   2. Дотримується єдиний стиль для всіх веб-сторінок. Всі елементи стилізовано. Кнопки створення, редагування та видалення виділені спеціальними стилями (кольорами чи іншим способом).
   3. Домашня сторінка містить логотип, опис і найважливіші посилання. На сайті наявні гіперпосилання між сторінками\розділами\сутностями.
   4. Наявна навігація по сайту у вигляді меню чи навігаційної панелі.
   5. Наявна сторінка з детальним описом сайту та контактами розробника
2. Можливість створення, редагування та видалення даних.
   1. На формі створення обов’язкові поля мають значення за замовчуванням.
   2. Наявна форма редагування даних має мати вже заповнені поля із попередніми значеннями. Є можливість відмінити оновлення даних.
   3. Можна видаляти дані (з підтвердженням цієї дії).
3. Використання графічних списків та таблиць для відображення колекцій даних.
   1. Дані в таблиці не є лише відображенням однієї таблиці (колекції) БД, а містять склеєні чи змінені дані.
   2. Дані у списках та таблицях пагінуються.
4. Можливість пошуку сутностей.
   1. Сайт дозволяє шукати сутності по одному чи декількох полях.
   2. Пошук повинен відбуватися по підрядку з ігноруванням регістру символів.
   3. Показувати по яких полях і що шукається, загальну кількість знайдених результатів.
   4. При невдалому пошуку виводити повідомлення про відсутність шуканих елементів.
5. Програма взаємодіє з базою даних. Запити до бази винесено у спеціальний окремий модуль (вид сховища).
   1. База даних містить мінімум 4 таблиці (колекції).   
      Реляційні таблиці в БД приведені до 1-ї та 2-ї нормальних форм.
   2. Між таблицями (колекціями) бази даних є one-to-many та many-to-many зв'язки
6. Програма має можливість завантаження файлів з файлової системи користувача, їх збереження у сховище даних.
   1. Завантажені користувачем зображення можна переглядати на сторінках сайту.
   2. Файли зберігаються у базі даних або на спеціальному віддаленому файловому хостингу.
7. Наявна реєстрація користувачів у системі. Паролі користувачів не зберігаються у відкритому виді, а хешуються обраним алгоритмом хешування рядків.
   1. Наявна форма аутентифікації. Пароль користувача при введені у форму прихований (використано спеціальний елемент). Сторінка цієї форми має сповіщення про некоректність введених користувачем даних.
   2. Програма авторизує доступу до даних по користувачах. У базі є дані, що належать конкретному користувачу (напр, авторство сутностей).
   3. Є мінімум 2 ролі користувачів. Обов’язкова роль адміністратора, що може керувати іншими користувачами.
   4. Користувач може вийти з сайту (logout). Наявна персональна сторінка користувача.
8. У сервера є можливість доступу до даних через API.
   1. Частина API шляхів (endpoints) надають доступ лише після авторизації.
   2. На сайті є веб-сторінка із коротким описом API.
9. Всі вхідні дані від користувача та з файлів перевіряються, помилки обробляються, користувачу відображається інформація про помилки.
10. Використання нотифікацій за допомогою ботів, веб-сокетів, webrtc, використання мультимедійних бібліотек.
    1. Можна підключити телеграм бот для отримання персоналізованих даних від бота та автоматичних нотифікацій підписаних користувачів при певних діях на сайті.
11. SPA сайт із використанням Front-end фреймворка Vue.
12. Використання JWT для SPA та Social Login за допомогою OAuth2

## 2.2. Вимоги до проекту

Проект ведеться у приватному git репозиторії.

1. Всі сторінки (після рендеру) відповідають HTML5.
2. Весь CSS винесено у зовнішні файли.
3. Код веб-сервера структуровано, розділено на модулі. Використовуються роутери для розділення коду обробників
4. Запити до бази даних винесені у окремий модуль.
5. Сервер використовує асинхронні виклики функцій за допомогою Promise або async\await
6. Весь браузерний JS підключається із зовнішніх файлів.

## 2.3. Вимоги до надійності

### 2.3.1. Відмови через некоректні дії користувачів системи

Відмови програми внаслідок некоректних дій користувача при взаємодії з програмою через графічний інтерфейс неприпустимі.

# 3. Умови експлуатації

## 3.1. Кліматичні умови експлуатації

Кліматичні умови експлуатації, при яких повинні забезпечуватися задані характеристики, повинні задовольняти вимогам, що пред'являються до технічних засобів в частині умов їх експлуатації.

## 3.2. Вимоги до складу і параметрів технічних засобів

До складу технічних засобів повинні входити:

1. Серверний комп’ютер, що включає в себе:
   1. Оперативну пам’ять об’ємом 1 Гігабайт, не менше.
   2. Операційну систему Ubuntu Linux версії 16.04 або вище.
2. Персональний комп'ютер із підтримкою веб-браузерів:
   1. Google Chrome версії 76 або вище
   2. Mozilla Firefox версії 68 або вище
3. Мобільний телефон із підтримкою веб-браузерів:
   1. Google Chrome for Android версії 76 або вище

# 4. Вимоги до програмної документації

## 4.1. Попередній склад програмної документації

Склад програмної документації повинен містити:

1. Технічне завдання
2. Звіт виконання

# 5. Техніко-економічні показники

## 5.1. Економічні переваги розробки

Орієнтовна економічна ефективність не розраховуються. Аналогія не проводиться зважаючи на унікальності пропонованих вимог до розробки.

# 6. Стадії і етапи розробки

## 6.1. Стадії розробки

Розробка повинна бути проведена в три стадії:

1. Розробка технічного завдання
2. Робоче проектування
3. Захист курсової роботи

## 6.2. Етапи розробки

На стадії розробки технічного завдання повинен бути виконаний етап розробки і затвердження даного технічного завдання.

На стадії робочого проектування повинні бути виконані перечислені етапи робіт:

1. Розробка програми
2. Розробка програмної документації (звіту)
3. Тестування програми.

На стадії захисту курсової роботи повинні бути виконані етапи підготовки результатів виконання і захист результатів з демонстрацією їх Викладачу.

# 7. Порядок контролю і приймання

## 7.1. Види випробувань

Приймально-здавальні випробування повинні проводитись на об’єкті Студента або Викладача в обумовлені терміни.

Приймально-здавальні випробування включають збірку розробленої програми із проекту репозиторія Студента та демонстрації всієї розробленої функціональності.

## 7.2. Загальні вимоги до приймання роботи

У випадку виникнення труднощів в реалізації функцій програми, ТЗ може бути відкориговане. Пропозиції щодо зміни і доповнення ТЗ можуть вносити як Студент так і Викладач.