

Завдання 1

1.1

Назва тест-кейсу: Перевірка функціоналу пошуку за критеріями: марка, модель, рік випуску та ціна.

Опис тест-кейсу:

Тест-кейс для перевірки функціоналу пошуку на веб-сайті <https://automoto.ua/uk/>.

Передумови:

1. Веб-сайт <https://automoto.ua/uk/> відкритий.
2. Сторінка пошуку по фільтрах завантажена.

Кроки:

1. Клікнути на поле "Марка".
2. Вибрати зі списку "Audi".
3. Клікнути на поле "Модель".
4. Вибрати зі списку "A4".
5. Ввести в поле "Ціна від \$ " значення "23000".
6. Ввести в поле "Ціна до \$ " значення "23000".
7. Клікнути на поле "Рік від".
8. Вибрати зі списку "2019".
9. Клікнути на поле "Рік до".
10. Вибрати зі списку "2019".
11. Клікнути на кнопку "Показати".

Очікувані результати:

1. Випадає список з марками авто.
2. В полі "Марка" відображається "Audi".
3. Випадає список з моделями авто.
4. В полі "Модель" відображається "A4".
5. Відображається значення "23000".
6. Відображається значення "23000".
7. Випадає список з роками.
8. В полі "Рік від" відображається "2019".
9. Випадає список з роками.
10. В полі "Рік до" відображається "2019".

11. Відображаються автомобілі, які відповідають заданим критеріям пошуку. Результат пошуку містить марку "Audi", модель "A4", рік випуску "2019" та ціну "23000 \$".

1.2

Назва тест-кейсу: Перевірка фільтру пошуку за типом кузова

Опис тест-кейсу:

Перевірка функціоналу пошуку за типом кузова на веб-сайті <https://automoto.ua/uk/>.

Передумови:

1. Веб-сайт <https://automoto.ua/uk/> відкритий.
2. Сторінка пошуку по фільтрах завантажена.

Кроки:

1. Обрати з випадаючого списку "Тип транспорту" - "Легкові"
2. Вибрати зі списку "Тип кузова" - "Седан".
3. Клікнути на кнопку "Показати".

Очікувані результати:

1. "Легкові" відображаються в полі пошуку.
2. Обрані автомобілі, які мають тип кузова "Седан".
3. Відображаються автомобілі, які відповідають заданим критеріям пошуку. Результат пошуку містить автомобілі з типом кузова "Седан"

1.3

Назва тест-кейсу: Перевірка вибору двох типів коробок передач одночасно

Опис тест-кейсу:

Перевірка функціоналу вибору двох типів коробок передач одночасно на веб-сайті <https://automoto.ua/uk/>.

Передумови:

1. Веб-сайт <https://automoto.ua/uk/> відкритий.
2. Сторінка пошуку (по фільтрах) завантажена.

Кроки:

1. Вибрати в розділі "Технічні характеристики" підрозділу "Коробка передач" - "Ручна / Механіка"

2. Вибрати в розділі “Технічні характеристики” підрозділу “Коробка передач” - “Автомат”
3. Клікнути на кнопку "Показати".

Очікувані результати:

1. Відображається позначка в списку “Ручна / Механіка” коробки передач.
2. Відображається позначка в списку “Автомат” коробки передач.
3. Відображаються автомобілі, які відповідають заданим критеріям пошуку. Результат пошуку містить автомобілі з “Ручна / Механіка” та “Автомат” коробками передач.

1.4

Назва тест-кейсу: Перевірка функціоналу пошуку за кольором кузова

Опис тест-кейсу:

Перевірка функціоналу пошуку за кольором кузова на веб-сайті <https://automoto.ua/uk/>.

Передумови:

1. Веб-сайт <https://automoto.ua/uk/> відкритий.
2. Сторінка пошуку по фільтрах завантажена.

Кроки:

1. Клікнути на зелений колір в розділі “Технічні характеристики” підрозділу “Колір”.
2. Клікнути на кнопку "Показати".

Очікувані результати:

1. Відображається галочка на зеленому кольорі
2. Відображаються автомобілі, які відповідають заданим критеріям пошуку. Результат пошуку містить автомобілі зеленого кольору.

1.5

Назва тест-кейсу: Перевірка функціоналу фільтрації за об'ємом двигуна

Опис тест-кейсу:

Перевірка функціоналу пошуку за об'ємом двигуна на веб-сайті <https://automoto.ua/uk/>.

Передумови:

1. Веб-сайт <https://automoto.ua/uk/> відкритий.

2. Сторінка пошуку по фільтрах завантажена.

Кроки:

1. Ввести в поле "Об'єм двигуна від" - "1" значення.
2. Ввести в поле "Об'єм двигуна до" - "2" значення.
3. Клікнути на кнопку "Показати".

Очікувані результати:

1. Відображається значення "1" в полі "Об'єм двигуна від".
2. Відображається значення "2" в полі "Об'єм двигуна до".
3. Відображаються автомобілі, які відповідають заданим критеріям пошуку. Результат пошуку містить автомобілі з об'ємом двигуна від 1л до 2л.

Завдання 2

2.1

Протестувавши наступні методи API **Reqres**:

- LIST USERS
- SINGLE USER
- CREATE
- UPDATE (PUT)
- UPDATE (PATCH)
- DELETE
- REGISTER - SUCCESSFUL
- LOGIN - SUCCESSFUL

За допомогою <https://reqbin.com/> я отримала наступні відповіді:

Status: 200OK

Запит виконано. Значення результату "успіх" залежить від методу HTTP:

* GET: ресурс було отримано та передано в тілі повідомлення.

*PUT або POST: ресурс, що описує результат дії, передається в тілі повідомлення.

Status: 201 (Created)

Запит виконано успішно і в результаті створено новий ресурс.

Зазвичай це відповідь, що надсилається після запитів POST.

Status: 204 (No Content)

Запит виконано успішно, але відповідь не містить сутності.

Завдання 2.2 (із багами)

1) REGISTER - SUCCESSFUL

Status:

400 (Bad Request) {

```
"error": "Note: Only defined users succeed registration"
}
```

Ця помилка означає, що спроба реєстрації користувача не вдалася, оскільки введені дані не відповідають визначеним в системі. Ймовірно, це означає, що ми намагаємось зареєструватися з неправильними або невірними даними, такими як неправильна електронна адреса, невірний пароль або ім'я користувача, якого немає в системі.

2) LOGIN - SUCCESSFUL

Умова нашого запиту:

```
{
  "email": "eve.holt@reqres.in",
  "password": "cityslicka"
}
```

Введені інші дані в поле "password":

```
{
  "email": "eve.holt@reqres.in",
  "password": "kapibara"
}
```

Проте наш Status: **200 (OK)**....

В цьому випадку - Статус 200 (OK) означає, що запит на входження в систему було успішно оброблено, тобто наш запит на вхід було успішно автентифіковано.

Інакше кажучи, якщо ми можемо успішно увійти в систему з різними паролями, то це може бути ознакою проблеми з безпекою або недостатньої перевірки при входженні.

Це може означати, що система не перевіряє паролі достатньо суворо, щоб забезпечити безпеку користувача.

Я рекомендую звернутися до розробників системи або служби підтримки користувачів, щоб вирішити цю проблему та зробити систему більш безпечною ;)

3) CREATE

Це наша умова - Response 201

```
{
  "name": "morpheus",
  "job": "leader",
  "id": "423",
  "createdAt": "2023-04-06T06:55:38.269Z"
}
```

А це вже наша відповідь - Status: 201 (Created)

```
{
  "name": "morpheus",
  "job": "leader",
  "id": "393",
  "createdAt": "2023-04-06T06:54:13.549Z"
}
```

Виходить, що по умові у нас мав створитись користувач під "id": "423", а створився під "id": "393".

Наш Status: 201 (Created) - означає, що запит на створення було успішно оброблено, тобто запит на створення користувача в цьому випадку було успішно виконано.

Те, що id користувача створився не тим, як очікувалося, може бути наслідком внутрішньої роботи системи, зокрема можливо зміненням налаштувань бази даних або випадкової помилки в коді.

Я рекомендую звернутися до розробників системи або служби підтримки користувачів, щоб вирішити цю проблему та перевірити, чому id користувача створився не тим, як очікувалося :)

4) UPDATE (PUT)

У нас задана умова Request -

```
{
```

```
    "name": "morpheus",  
    "job": "zion resident"  
  }.
```

та наша відповідь, яка повинна повернутись - Response 200

```
{  
  "name": "morpheus",  
  "job": "zion resident",  
  "updatedAt": "2023-04-06T07:38:20.382Z"  
}.
```

проте я змінила умову в Content на

```
{  
  "name": "morpheus",  
  "job": "resident"  
}.
```

і мені прийшла відповідь Status: 200 (OK).

Якщо відповідь зі статусом 200 (OK) при методі UPDATE (PUT) приходить навіть після зміни умови в запиті, це означає, що сервер не перевіряє всі поля в запиті на відповідність вимогам, а лише перевіряє обов'язкові поля або поля з обмеженими варіантами значень.

Така "слабка валідація" може дозволяти змінювати поля запиту, які насправді повинні бути обов'язковими, або змінювати поля на значення, які не є допустимими. Це може бути наслідком недостатньої перевірки даних на стороні сервера.

Тому, якщо нам потрібно запобігти таким ситуаціям, нам необхідно забезпечити належну валідацію даних на стороні сервера, яка повинна перевіряти всі обов'язкові поля та поля з обмеженими варіантами значень на відповідність вимогам. Таким чином, ми зможемо забезпечити правильну обробку даних і уникнути непередбачуваних результатів.

Завдання 3

Виконуючи завдання 3, я відправила по одному валідному запиту для кожного методу в [GO REST](#). Я авторизувалась на сайті та отримала API Token.

Я порівнювала і його деталі, введені у <https://reqbin.com/> і відображені в Developer Tools.

Порівнюючи деталі введені і відображені в Developer Tools, я дійшла висновку, що вони не збігаються.

Де виникала помилка в запиті, допустимо 422 то в Developer Tools це висвічувалось як 200 - успіх.

Також я змінювала дані для створення нового користувача і операція відбулася успішно зі Status: 201 (Created).

Можливо, різниця в запитах пов'язана з тим, що різні інструменти можуть використовувати різні налаштування за замовчуванням при відправленні запитів.

Наприклад, різні інструменти можуть автоматично додавати різні заголовки до запиту, або використовувати різні параметри посилання для виконання запиту. Також можуть бути різні параметри захисту від CSRF атак, які додаються автоматично в деяких інструментах.

Крім того, різниця може бути пов'язана з тим, що запити відправляються з різних місць, тобто з різних мереж, і це може впливати на те, як запити обробляються сервером.

У будь-якому випадку, якщо ми бачимо різницю в запитах, які ми надсилаємо через різні інструменти, нам потрібно детальніше дослідити різні параметри запиту, заголовки та параметри посилання, які використовуються в кожному інструменті, щоб визначити, що саме впливає на відмінність в запитах.