# L 2 Вступ до ІТ. Компанії та проєкти. Поняття тестування. Основні стандарти ISTQB

## 1. Вибери предмет з твого оточення (наприклад, чашку, ноутбук, машину, футболку тощо) і спробуй його протестувати. Поясни, чому ти обрав/ла саме таку перевірку.

**Тестування предмету Ножиці**

**Верифікація:**

Дві металеві частини - є

пластикові ручки синього кольору - є

Один круглий отвір і один овальний - є

Болтик посередині - є

Довжина ножиць 15 см - відповідає умовам

**Валідація:**

Гострота ножиць - недостатня, ріжуть не дуже легко, рекомендується заточити

Обробка пластикової частини не дуже акуратна, в наявності гострі зазубрини по стиках, рекомендується додатково обробити пластикові деталі,

Ножиці не люфтують, болтик кріпиться якісно.

**Верифікація** – підтвердження того, що певні вимоги були виконані. Чи виконані умови, які були складені перед початком роботи.

**Валідація** – перевірка того, що продукт відповідає очікуванням і потребам користувачів. Чи працездатний продукт, або кожен його елемент.

## 2. Склади порівняльну таблицю різних видів компаній. Вкажи плюси та мінуси кожної з них (з точки зору працівника).

| № | вид компанії | + | - |
| --- | --- | --- | --- |
| 1 | Product | надійність, стабільність, | однотипна робота, можлива довга бюрократія, мало змін |
| 2 | Startup | мало бюрократії, швидкість узгодження процесів, багато впровадження нових ідей. | не надійність |
| 3 | Outsource | компанія сама шукає проекти, можливість змінювати проекти, команду, різноманітність, можливість гарного росту. | велика залежність від замовника, можуть бути дуже маленькі терміни виконання роботи |
| 4 | Outstaff | різноманітність, можливість гарного росту. Захищеність від компанії, а не замовника. | замовник проекту може бути не компетентним, велика залежність від замовника, можуть бути дуже маленькі терміни виконання роботи |
| 5 | Academy | схоже на продуктову компанію.  Може бути багато розвитку всередині компаній причому не тільки за прямим фахом, | Однотипна робота, оскільки сфера діяльності компанії не змінюється. Майже немає змін проектів. |
| 6 | Recruitment Agency. | Розповість все про вакансію, допоможе отримати найвищу зарплату з можливої дельти. | співпраця суто по співбесіді, фактично, робота в компанії замовника вакансії. |

## 3. Наведи приклади невдалої валідації або верифікації продукту, з якими довелося зіткнутися в житті.

**Невдала валідація:**

Попросила чоловіка купити продукти на суп - він не купив картоплі.

Все є, картоплі - немає. Умови не виконано, супа не буде. Або буде, але не смачний, коротше кажучи, умову не виконано.

Якщо б все купив, що треба, але картопля (або інший скадник) виявилася з середини гнилою, то був би приклад невдалої верифікації

**Невдала верифікація:**

обирала собі новий телефон між двома моделями.

Умови були: не дорогий, в великою кількістью оперативної пам’яті, нормальною камерою.

Обрала, всі ці умови виконано, АЛЕ. Камера виявилася не найкращою, особливо при нічній зйомці. Ще й виступає над корпусом - легко може розбитися, що і відбулося на другий тиждень користування.

Мабуть, тут все ж таки спірне питання, чи виконані умови верифікації, тому, не впевнена, що приклад вдалий. Але ж треба було саме з реального життя. )

Ще один приклад **невдалої валідації:**

Замовила зимові чоботи:

Верифікація:

Розмір та зовнішній вигляд (колір, фасон, товщина підошви) відповідають замовленому та очікованому.

Валідація

Але що до користування **валідації**: чоботи виявилися

1) дуже важкі

2) виявилися надто широкі - гупала нога

3). поганої якості - швидко розклеїлися і стали протікати (і слава Богу, можна викидати з чистим сумлінням!)

користуватися неприємно

## Поясни важливість 2-3 принципів тестування на власний вибір. Наведи приклади з власного досвіду.

Всі принципи буду розглядати через приклад перевірки домашнього завдання дитини

1. **Тестування показує наявність дефектів**

Те що я знайду максимум помилок у домашньому завданні дитини, не означає, що я знайшла їх всі і викладач не знайде помилок

1. **Вичерпне тестування неможливе**

те що я перегляну все домашнє завдання дитини різними способами:

по яким предеметам є ДЗ

чи виконані все предмети зі списку  
чи немає у роботах помилок

і тп

не означає, що точно буде виконано все і викладач не знайде помилок.

1. **Скупчення дефектів**

Якщо я знайшла помилки в першій вправі, наприклад, математики, вірогідність того, що вони будуть далі - дуже велика. І так по кожному предмету.

Якщо дитина не пам’ятає що задано по одному предмету, треба очікувати, що і по іншим може щось забути.

1. **Тестування залежить від контенту**

Математика перевіряється одним чином, література - іншим, а чи виконане ДЗ по всім предметам - зовсім іншим чином. І чи якісно виконується ДЗ - теж перевіряється іншим способом.

# L 3 SDLC та STLC. Методології розробки ПЗ.

## 1. Склади порівняльну таблицю найбільш поширених методологій:

| № | Назва | + | - |
| --- | --- | --- | --- |
| 1 | Waterfall | строга послідовність стадій, чіткі строки, обов’язки та задачі | неможливість адаптування до будь-яких змін, тестування починається тільки з середини проекту |
| 2 | V-model | чіткі етапи,  тестування починається з самого початку, | неможливість змін  довгі терміни виконання роботи |
| 3 | Incremental model | чітка структура етапів, створення кожен раз самостійної версії продукту (це і + і -) можливість змін, замовник може дати відгук на кожну з версій | великі терміни виконання,  через створення кожен раз самостійной версії продукту (це і + і -) - трудоємність функціональна система повинна бути повністю визначена на початку життєвого циклу для виділення ітерацій при постійних змінах структура системи може бути порушена |
| 4 | Spiral model | високий аналіз ризиків;  можливість внесення змін і додавання нової функціональності навіть на відносно пізніх етапах;  раннє створення робочих прототипів | може бути досить дорога у використанні;  управління ризиками вимагає залучення висококласних фахівців;  успіх процесу у великій мірі залежить від стадії аналізу ризиків;  не підходить для невеликих проектів. |
| 5 | Agile | можливість змін швидке прийняття рішень постійним комунікаціям;  мінімізація ризиків;  полегшена робота з документацією. | велика кількість мітингів і обговорень, що може збільшити час розробки продукту;  складно планувати процеси, так як вимоги постійно змінюються;  рідко використовується для реалізації великих проектів |
| 5.1 | Scrum | швидке реагування: Короткі спринти дають змогу швидко адаптуватися до змін і вимог клієнта.  замовник активно залучений до процесу розробки, що сприяє точній реалізації вимог. | деякі люди, які знають продукт, стають незамінними, так як документація не надається в процесі розробки;  неможливо спланувати точну дату завершення, так як все уточнюється за результатами попереднього спринту;  замовники не завжди можуть зрозуміти що саме відбувається і терміни виконання робіт |
| 5.2 | Kanban | наочна система за допомогою дошки учасники бачать, які завдання перебувають у роботі, що покращує координацію.  легко адаптуватися до змін у потребах клієнта або ринку.  Зменшення затримок: Ліміти в процесі знижують затримки і прискорюють виконання завдань | не завжди підходить для складних проектів, де потрібен суворий контроль.  не стійка структура, може бути розбалансованою. |

**2. Напиши розгорнуті відповіді (0,5 - 1 сторінки тексту) на такі два питання:**

## На твою думку, чому з’явився Agile-маніфест?

## Які проблеми він мав вирішити і чи це вдалося?

Agile-маніфест з’явився на противагу попереднім не гнучким системам. Назріла проблема, що початкове завдання потребує неодноразових змін, але ці зміни не передбачені в процесі розробки продукту. Потрібно було створити таку систему, яка може впроважувати зміни, зберігаючи при цьому весь процес роботи.

Тим не менш, досягти згоди щодо однієї сталої нової системи не вдалося, тому були створені маніфест, як вимоги до роботи (взаємодії), і принципи.

Саме маніфест не вирішив проблему цілком, але впровадив новий погляд на організацію процесу розробки ПЗ, але створення принципів Agile дозволило впровадити гнучкі методики розробки ПЗ

На мій погляд, у гнучкості є свої недоліки, іноді затягування процесу і потрапляння в замкнене поле нескінченних змін. Тому в майбутньому будуть виникати все нові й нові методики. Світ не може стояти на місці. )

**3. Ти – засновник/ця стартапу і плануєш випустити на ринок мобільний застосунок для обміну світлинами котиків.**

## Яку методологію ти обереш для процесу розробки і чому? Відповідь текстово обґрунтуй.

Скоріш за все Scrum або Kanban

проект не великий, особливо чіткої структури не потребує Мені здається, дешевше всього буде дотримуватися методики Канбан, щоб не залучати Скрам-майстра та інших вузьких спеціалістів.

# L 4 Робота з вимогами. Аналіз та складання вимог

## 1. Наведи короткі приклади вимог (3-5) до будь-якого предмета з твого оточення, які б відповідали кожному з таких критеріїв оцінки якості:

## Атомарність

основний фон кнопки “Старт” на мікрохвильовій печі повинен бути червоного кольору (загалом, ця вимога відповідає і іншим критеріям)

## Несуперечність

Надпис “старт” на кнопці початку роботи повинен бути такого-то

Надпис “старт” на кнопці початку роботи повинен бути виконан шрифтом Arial

Надпис “старт” на кнопці початку роботи повинен бути такого-то розміру

## Тестованість

Сигнал закінчення нагрівання повинен лунати 6 разів

Перші 5 сигналів повинні бути тривалістю 1 секунда

Останній шостий сигнал повинен тривати 2 секунди

## Відстежуваність.

Кнопка “старт” повинна бути за розміром 10 х 30 мм

## 2. На твою думку, яка з технік тестування вимог гарантує максимально можливу якість фінального результату. Відповідь обґрунтуй (3-5 речень).

Я б не змогла виділити лише одну техніку, яка б гарантовано давала максимальний результат.

1. Рев’ю -це один з гарних варіантів - залучується купа фахівців, які оцінюють вимоги, і це дуже добре з точки зору аналізу. Але при виконанні системи все одно може виявитися щось не помічене або суперелчливе.

. Мінус такого варіанта — витрачається багато часу.

2. Питання

Взагалі, меня дуже подобається ця техніка, тому що справді до вимог оптимально не повинно залишитися жодних запитань перед початком роботи. Але не всі питання можуть бути задані, крім того, у виконавця може бути інше розуміння.

3. Тест-кейси та чек-листи

мені здається, якщо створювати тест-кейси та чек листи, то продукт повинен бути вже написаний відповідно до вимог, може бути вже дуже затратно щось виправляти

4. Дослідження поведінки системи

Необхідно змоделювати процес роботи користувача з системою, створеною за

вимогами, та після цього визначити неоднозначні варіанти поведінки системи.

5. Схеми та майндмепи

в схемах часто можна щось упустити, нібито процеси по за схемою і так зрозумілі, і саме там буде не охоплюватися можливість посилки.

6. Прототипування

в такому разі краще зробити прототип системи, і на ньому подивитися, чи не йде щось не так.

Як на мене, оптимально застосовувати 2 вида технік, одна з яких цн або рев’ю або питання - в залежності від об'єму продукту, а другий - прототипування, або схеми та мейндмейнд, щоб умовно побачити продукт загалом.

## 3. Ти – засновник/ця стартапу, який планує випустити на ринок мобільний застосунок для обміну світлинами котиків. Склади функціональні (5-7 од.) та нефункціональні (5-7 од.) вимоги до застосунку.

**функціональні**

* Можливість реєстрації користувачів
* незареєстрований користувач може користуватися програмою тільки в режимі перегляду
* користувач може ставити реакції на світлини інших користувачів
* користувач може завантажувати до 100 світлин
* пересилати за 1 раз можна до 10 світлин
* зареєстрований користувач може:
  + завантажувати свої світлини
  + редагувати їх
  + пересилати до іншого користувача

**нефункціональні**

* осонвний фон головної сторінки виконан в стилі анімалістичного виду, сірого кольору
* такі-то позначки виконані чорним кольором
* світлини у вікнах попереднього перегляду відображаються розміром таким-то
* для скачування та пересилання іншим користувачам, світлина передається такого-то розміру
* імя користувача відображається таким-то шрифтом, такого-то розміру

# L 5 Тестова документація

## 1. Склади порівняльну таблицю трьох видів тестової документації:

**Чек-ліст**

найбільший, “генеральний “ документ, в якому міститься перелік всіх перевірок, які необхідно зробити, перелік операційних систем, в яких робиться перевірка, результат перевірки, тощо.

Переваги: чек-ліст дозволяє відслідковувати процес перевірки продукту, бачити обсяг роботи, допомагає уникнути пропущень якогось аспекту перевірки

Недоліки: створення чек-ліста потребує часу

не докладний список.

**Тест-кейс**

Це опис конкретного випадку перевірки системи. Він включає в себе повний опис кроків перевірки, очікуваний результат, опис умов та параметрів перевірки.

Переваги: за допомогою тест-кейса легко відстежити, що саме робилося, і які дії призвели до конкретного результату перевірки. Відповідно, легко скласти наступні кроки роботи не тільки виконавцю теста, але й іншим учасникам розробки продукту.

Недоліки: тест-кейс не може описати весть процес роботи - тільки окремий випадок. Не завжди доцільно документувати всі дії перевірки, наприклад, при виконанні перевірок простих процесів, тест-кейс писати довше, ніж саме тестування. Часом, писати тест-кейс просто немає сенсу

**Користувацький сценарій (Use Case)**

перелік дій, за якими користувач взаємодіє с продуктом/системою, щоб досягти якогось конкретного результату. Більш об’ємний за тест-кейс

Переваги: за допомогою юзер кейса можна виявити якісь баги в роботі системи, які не можна виявити у особному тесті. Чудова база для створення тест-кейсів. Юзер-кеси зрозумілі замовнику.

Недоліки: не може охопити весь перелік перевірок.

# L 6 Рівні та види тестування

## 1. Склади порівняльну таблицю функціонального, нефункціонального і пов’язаного зі змінами видів тестування.

**Порівняння має містити такі блоки:**

* **що перевіряється;**
* **коли застосовується;**
* **обмеження;**
* **особливості**

**функціональне тестування:**

що **перевіряється**:  
чи виконує продукт ті функції, які були зазначені у специфікації, та

**коли застосовується**;

під час всього процесу створення продукту, та навіть підтримки продукту після релізу. Починаючи з модульного тестування, закінчуючи регресивним тестуванням, та іншими видами тестування, які перевіряють продукт після виправлення помилок, дописання нових частин

**обмеження, особливості;**

не перевіряють такі властивості як зручність, надійність, безпека, продуктивність, і подібне, що належить до нефункціонального тестування

**нефункціональне тестування:**

**що перевіряється**:

Перевіряє продукт на властивості які не пов’язані з конкретними функціями: стабільності, безпеки, наватнаження, зручності використання, локалізації тощо.

**коли застосовується:**

після створення продукту, або його модуля

**обмеження, особливості;**

не перевіряє конкретні функції продукта, перевіряє продукт загалом, умовно “наскільки надійно він працює”

**тестування, пов’язане зі змінами:**

**що перевіряється**

функції продукту, які змінювались внаслідок виправлення багів, допису продукту

**коли застосовується:**на всіх етапах, коли були внесені зміни в продукт

Димове тестування може застосовуватися не тільки після змін, а в окремих випадках як початкове тестування.  
**обмеження, особливості;**ці види тестування направлені на перевірку виправлення дефектів, можуть застосовуватися велику кількість разів,доки дефекти не буде виправлено

## 2. Поясни, в чому різниця між регресією та ретестингом (5 речень).

Регресія перевіряє продукт після виправлення багів не тільки в місці бага, а і в близьких модулях. Можливо, виправлення багу негативно вплинуло на роботу інших функцій

Ретест - суто повторний тест тієї ж самої проблеми.

## 3. Як ти вважаєш, чи можливе для продукту проведення тільки функціонального тестування, без перевірки нефункціональних вимог?

* **Якщо так – в яких випадках?**
* **Якщо ні – чому?**
* **Обґрунтуй свою відповідь.**

Вважаю, не бажаним застосовувати тільки функціональне тестування, оскільки кожен з видів тестує різні якості продукту. Функціональне - чи виконує продукт якусь очікувану роботу (функції), нефункціональне - чи “добре” працює продукт. Чи не буде відмови при навантаженні, якихось сбоях постачання, налаштуваннях на інших пристроях, чи пристосування до іншої країни, чи захищені дані. Це неможливо перевірити функціональним тестуванням.

можливо, можна обійтись тільки функціональним тестуванням, коли продукт якийсь дуже маленький, але я все одно слабо уявляю-який продукт можна не перевіряти на “надійність” роботи.

## 4. Як ти розумієш необхідність проведення smoke (димового) тестування? Чи завжди воно є доречним?

Це швидка перевірка чи виконує система основні функції. Необхідна для первинного аналізу продукту одразу після випуску продукту, або після змін. Недоречне воно, коли продукт потрібно перевірити докладно.

# L 9 Що таке “Баг”. Робота з баг-трекінговими системами на прикладі Jira

## 2. Наведи власні приклади багів, які можуть мати такі комбінації::

## Severity - Critical/Priority - Low

коли баг блокує, або значно погіршує роботу, але зустрічається вкрай рідко. Наприклад, помилка виникає у дуже старій версії програмного забезпечення,

або, з цією помилкою зустрічається кожний 100000 користувач.

або блокування системи виникає у вкрай рідкому випадку використання, наприклад при вводі якоїсь комбінації клавіш, типу “ююю!???ЮЮЮ”

## Severity - Minor / Priority - Highest

Коли помилка не впливає, або практично не впливає на роботу продукта, але значно погіршує його сприйняття.   
Наприклад, помилка у назві продукту, логотипу, і тп.  
Карикатура замість фону.  
Інші яскраві помилки дизайну.

# L 14 Основи web (HTML, CSS). Кодування символів

## 2.За допомогою Developer Tools визнач кодування символів, використаних на таких вебсторінках:

* [https://beetroot.academy](https://beetroot.academy/) - utf-8
* <https://microseniors76.com/> - iso-8859-1
* <https://www.tennis-warehouse.com/> - utf-8
* <https://www.fidelity.com/> - utf-8

1 Створи єдиний HTML-документ з підключенням в ньому стилів трьома різними способами:

* В середині специфічного тегу (inline)

<!DOCTYPE html>

<html>

<head>

<title>Домашка</title>

</head>

<body>

<h1 style="text-align:center;">Урок 14</h1>

<p style="color:red;">Я не люблю HTML</p>

</body>

</html>

* В розділі “HEAD”

<!DOCTYPE html>

<html>

<head>

<style>

h1 {

color: green;

font-family: verdana;

font-size: 150%;

}

p {

color: purple;

font-family: courier;

font-size: 50%;

}

</style>

<title>Домашка</title>

</head>

<body>

<h1>Урок 14</h1>

<p>Я не люблю HTML</p>

</body>

</html>

У зовнішньому .css файлі

body {

background-color: powderblue;

}

h1 {

color: white;

font-family: verdana;

font-size: 100%;

}

p {

color: purple;

font-size: 250%;

}

# L 15 Тестування інтерфейсу користувача (GUI). Тестування вебзастосунків

## Протестуй UI/UX сайту Оксфорд Медикал. В Google docs створи 5 багів з дотриманням усіх вимог до оформлення багу. Проведи кросбраузерне тестування знайдених раніше багів у трьох різних браузерах.

### ID 1

**Name**

The "Contact the operator" button overrides the "Details" button on the carousel slides

**Testing environment**

Google Chrome, 122.0.6261.69

**Preconditions**

<https://oxford-med.com.ua/> is opened

**Steps**

look to the carousel in head of maine page

**Actual result**

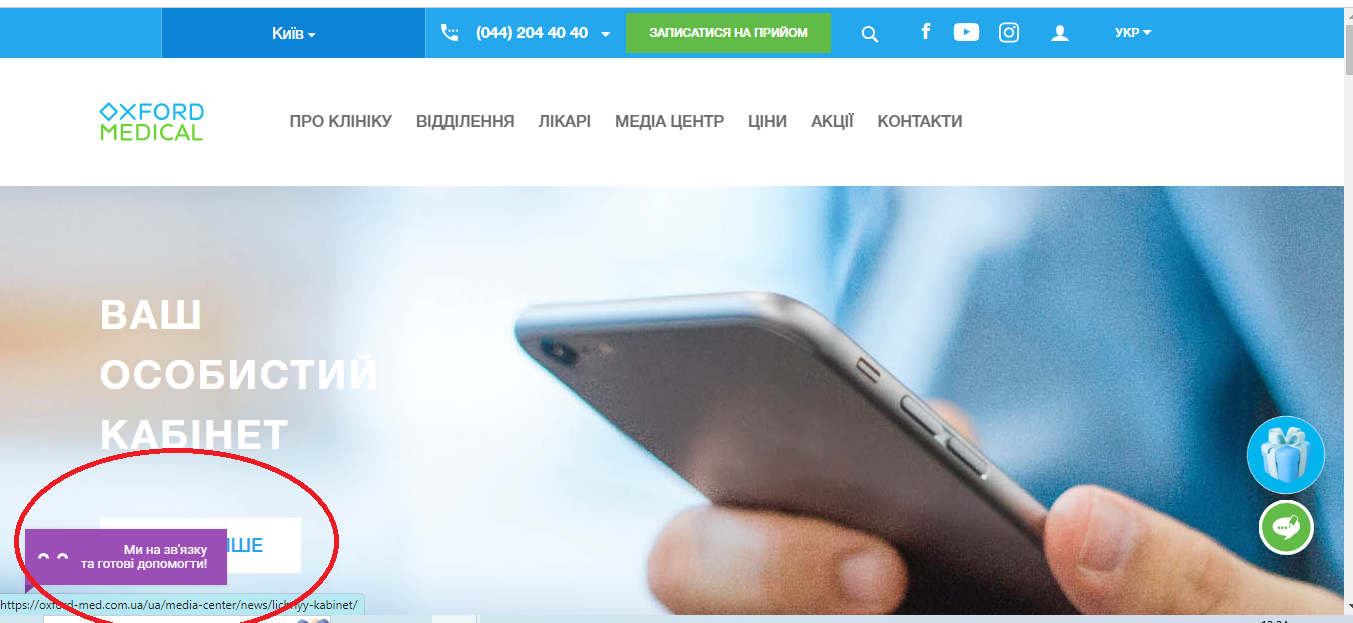
The "Contact the operator" button overrides the "Details" button on the carousel slides

**Expected result**

The "Details" button on the carousel slides is higher than the "Contact the operator" button

**Priority: Medium**

**Severity Minor**



### ID 2

**Name**

Arrows for scrolling carousel slides are missing

**Testing environment**

Google Chrome, 122.0.6261.69

**Preconditions**

<https://oxford-med.com.ua/> is opened

**Steps**

look to the carousel in head of maine page

**Actual result**

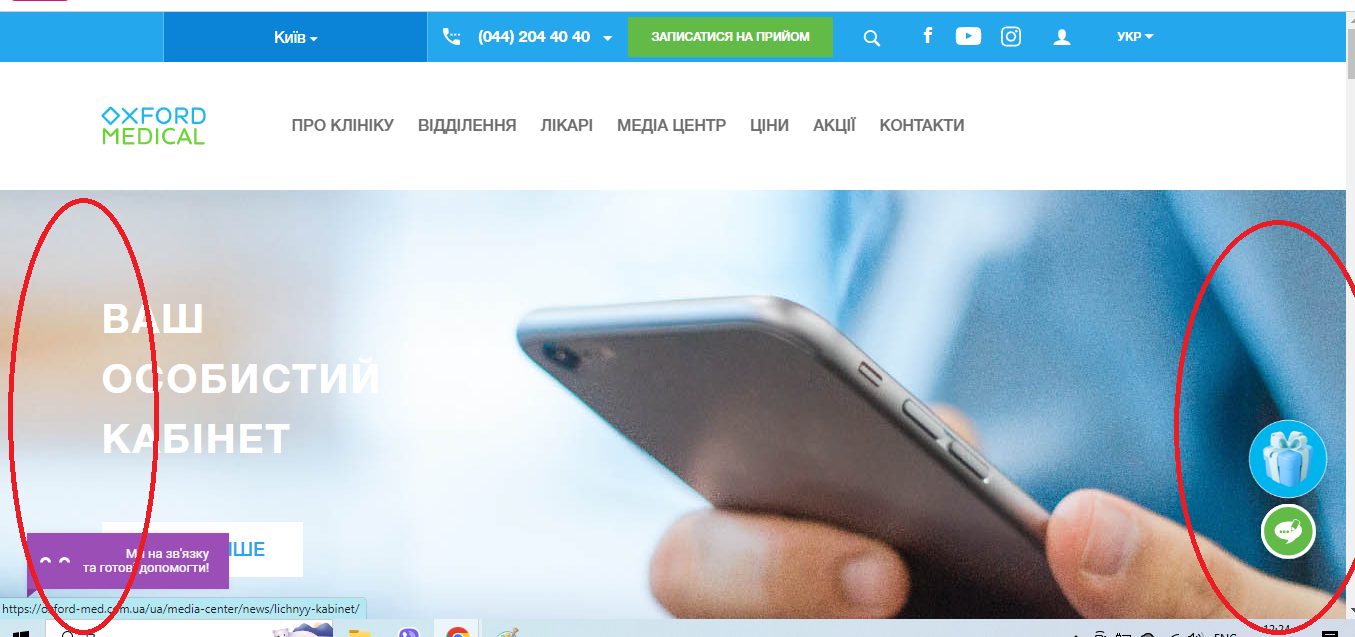
Arrows for scrolling carousel slides are missing. It is not intuitively clear how to independently change the slides on the carousel.

**Expected result**

Carousel scrolling arrows are present on each side of the slides

**Priority: Low**

**Severity Minor**



### ID 3

**Name**

Hint on Pop-Up in Russian

**Testing environment**

Google Chrome, 122.0.6261.69

**Preconditions**

<https://oxford-med.com.ua/> is opened

**Steps**

Click the button with the gift icon on the right.

Do not enter anything in the input fields.

Click the "Subscribe" button

Click on the exclamation mark icon with a hint next to the input field.

**Actual result**

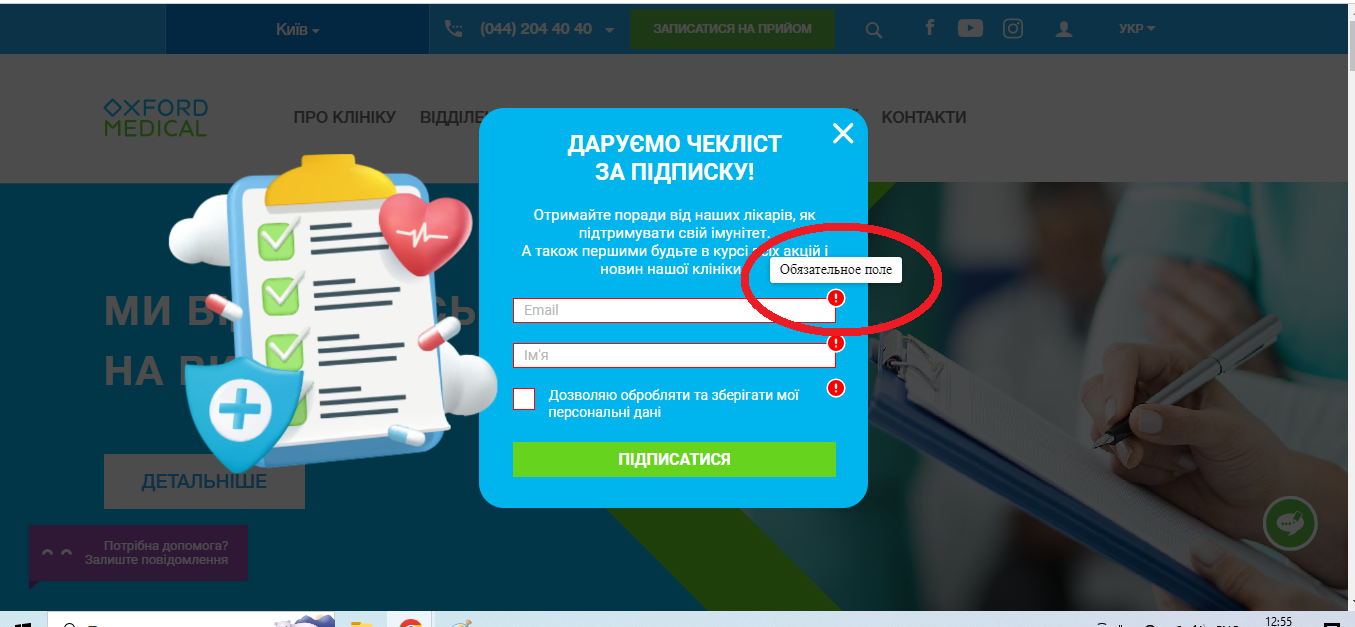
The hint is written in russian language.

**Expected result**

The hint is written in Ukrainian language.

**Priority: Medium**

**Severity Minor**



### ID 4

**Name**

The button for communication with the operator interferes with choosing a city

**Testing environment**

Microsoft Edge 122.0.2365.52

**Preconditions**

<https://oxford-med.com.ua/> is opened

**Steps**

Loock to header

Cklick to Dropdown list with city Kiyv

**Actual result**

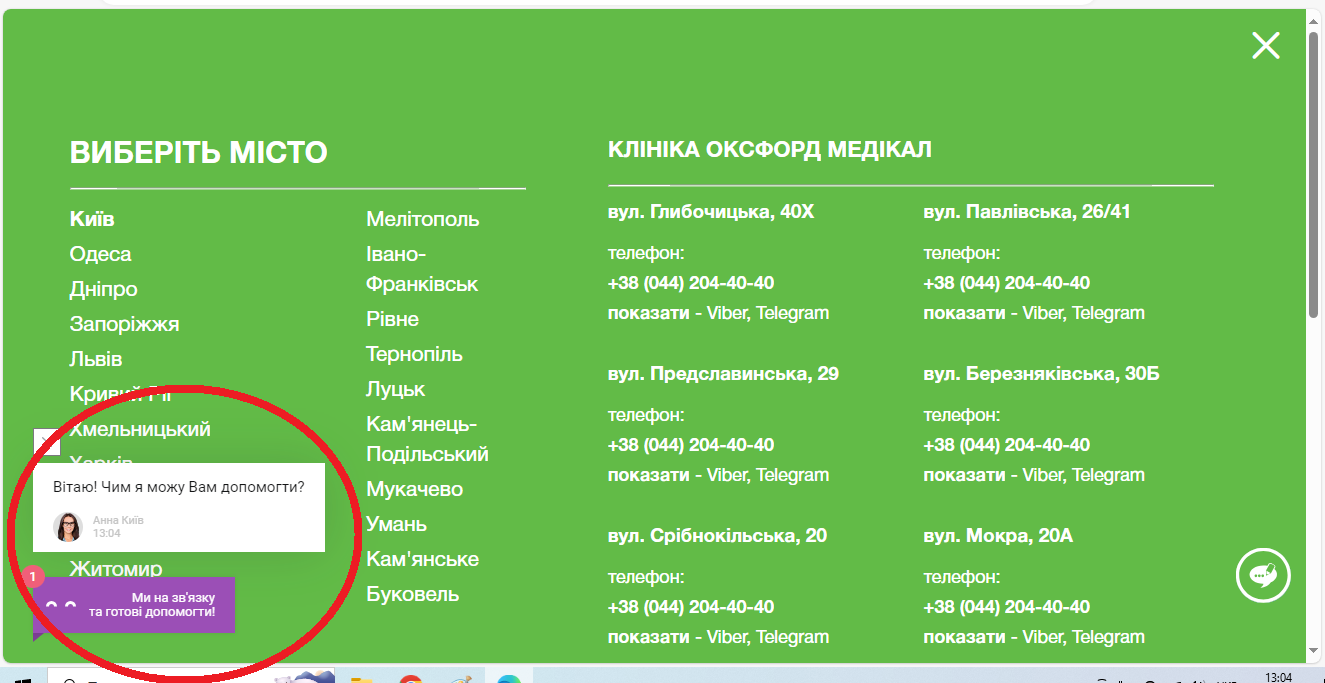
The button for communication with the operator interferes with choosing a city

**Expected result**

The button for communication with the operator is located below the list of cities.

**Priority: Low**

**Severity Minor**

****

### ID 5

**Name**

**Testing environment**

Microsoft Edge 122.0.2365.52

**Preconditions**

<https://oxford-med.com.ua/> is opened

**Steps**

Loock to header

Cklick to Dropdown list with city Kiyv

Choose the Dnipro city

Go to the page of the city of Dnipro

Select the drop-down list again by clicking on the city of Dnipro in the header

**Actual result**

The font of the selected city is little different from other cities.

Swipe on the button "Look on the map".

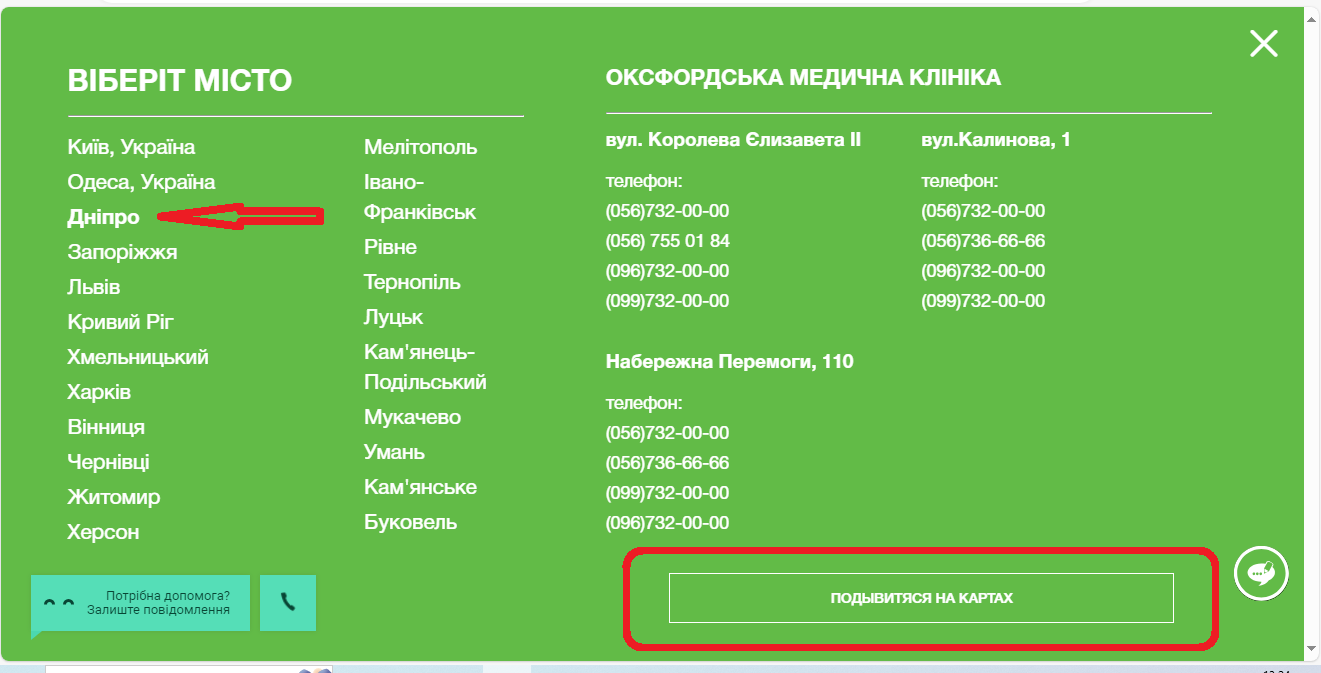
**Expected result**

The font of the selected city is not significantly different from other cities.

The inscription on the button "Look on the map" is written without mistakes.

**Priority: Low**

**Severity Minor**

****

# L 17 Тестування API. JSON. REST.

## 1. В Google docs створи 5 тест-кейсів для функціонального тестування сайту <https://automoto.ua/uk/> (5 найбільш пріоритетних тестів).

### ID 1

**Name**

**Preconditions**

<https://automoto.ua/uk/> is opened

**Steps**

Сlick on the authorization icon in the right corner of the header

In the pop up that opens, enter the phone number in accordance with the requirements “38 0хх ххх хххх”...

**Expected result**

An SMS with a four-digit code for authorization will be sent to the specified number

### ID 2

**Name**

**Preconditions**

<https://automoto.ua/uk/> is opened

Follow the steps from the test case ID1:

Enter a valid number in the authorization field

Receive SMS for authorization

**Steps**

Enter the code from SMS in the input field

Click the "Continue" button

**Expected result**

A "Welcome" pop-up will appear

### ID 3

**Name**

**Preconditions**

<https://automoto.ua/uk/> is opened  
Follow the steps from the test case ID2

a phone number is entered in the authorization window,   
 an SMS with a code is received,   
 the "Next" button is pressed

**Steps**

Click the "Next" button

**Expected result**

A Pop-up with promo carousel will appear

### ID 4

**Name**

**Preconditions**

<https://automoto.ua/uk/> is opened  
Follow the steps from the test case ID3

a phone number is entered in the authorization window,   
 an SMS with a code is received,   
 the "Next" button is pressed  
 the "Next" button from Pop-up with promo carousel is pressed

**Steps**

Click the "Next" button

**Expected result**

The user has switched to his personal account

### ID 5

**Name**

Delete number from the personal account

**Preconditions**

<https://automoto.ua/uk/> is opened

Go to personal account

**Steps**

Clock to the “Settings” tab

Click on the "Delete number" link

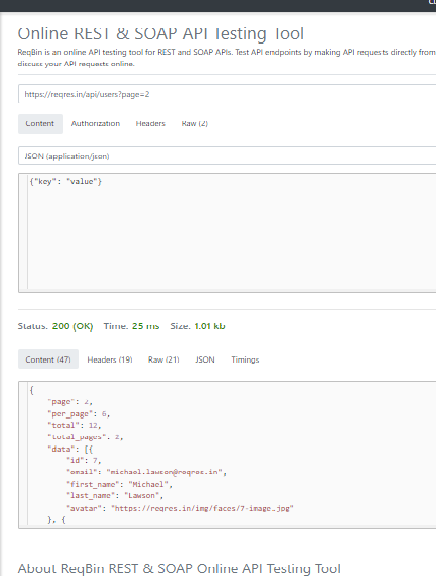
In the pop-up "Are you sure you want to delete the number?" Click the button “delete”

**Expected result**

A warning appeared: "You cannot delete a single number"

## 2 За допомогою https://reqbin.com/ протестуй наступні методи API Reqres:

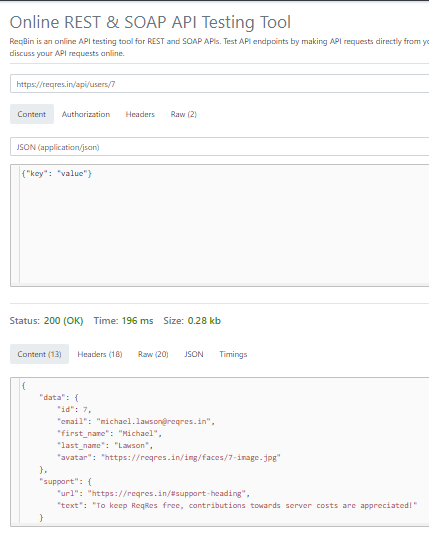
LIST USERS



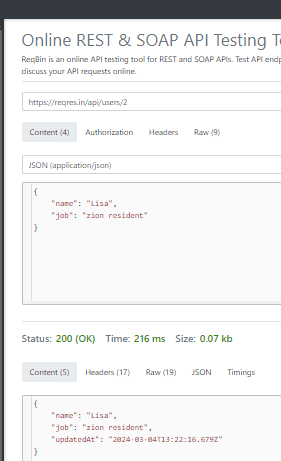
CREATE



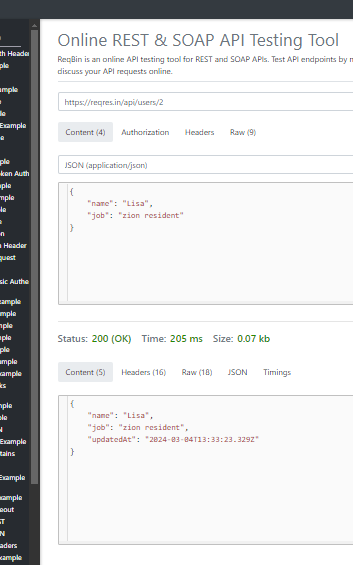
SINGLE USER



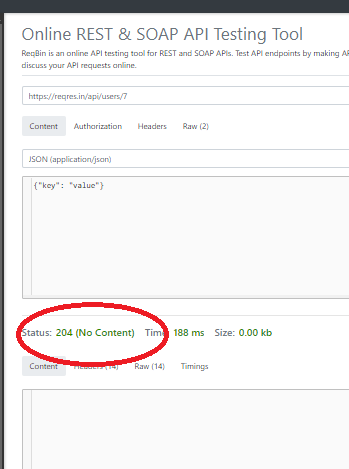
UPDATE (PUT)



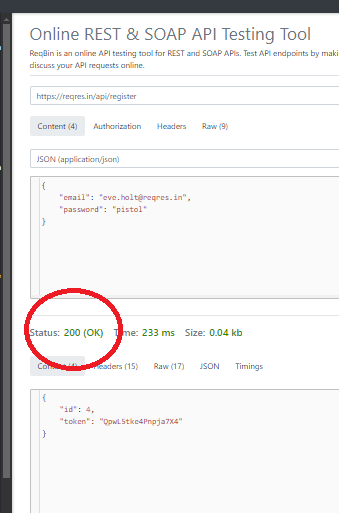
UPDATE (PATCH)



DELETE



REGISTER - SUCCESSFUL



LOGIN - SUCCESSFUL

