**ТЕСТОВИЙ ПЛАН ДЛЯ ВЕБ-ПЛАТФОРМИ OLX**

**1. Вступ**

**1.1 Призначення документу**

Цей документ є тестовим планом для веб-платформи OLX. Він описує стратегію, об'єкти, підходи, ресурси, розклад та інші аспекти тестування, які будуть використовуватися під час процесу тестування платформи. Документ призначений для команди розробників, тестувальників, проектних менеджерів та інших зацікавлених сторін.

**1.2 Область застосування**

Даний тестовий план охоплює процес тестування веб-платформи OLX, включаючи функціональне, нефункціональне, інтеграційне, UI/UX та регресійне тестування. План включає в себе як ручне, так і автоматизоване тестування.

**1.3 Посилання на суміжні документи**

* Специфікація програмних вимог (SRS) для веб-платформи OLX
* Технічна документація веб-платформи OLX
* IEEE 829 Стандарт для документації з тестування програмного забезпечення
* Керівництво з UI/UX для платформи OLX

**2. Об'єкти тестування**

**2.1 Функціональні модулі**

1. **Модуль реєстрації та авторизації**
   * Реєстрація нових користувачів
   * Авторизація користувачів
   * Відновлення паролю
   * Верифікація електронної пошти/телефону
   * Авторизація через соціальні мережі
   * Керування профілем користувача
2. **Модуль управління оголошеннями**
   * Створення нових оголошень
   * Редагування оголошень
   * Видалення оголошень
   * Завантаження медіа-файлів (фото, відео)
   * Управління статусом оголошення
   * Просування оголошень
3. **Модуль пошуку та фільтрації**
   * Пошук за ключовими словами
   * Фільтрація за категоріями
   * Фільтрація за місцезнаходженням
   * Фільтрація за ціною та іншими параметрами
   * Сортування результатів
   * Збереження пошукових запитів
4. **Модуль комунікації**
   * Обмін повідомленнями між користувачами
   * Відправка сповіщень
   * Система відгуків та рейтингів
5. **Модуль транзакцій**
   * Оплата преміум-послуг
   * Система безпечних угод
   * Генерація електронних чеків
6. **Адміністративний модуль**
   * Модерація оголошень
   * Управління користувачами
   * Аналітика та звітність

**2.2 Інтерфейси користувача**

1. **Веб-інтерфейс**
   * Адаптивний дизайн для різних розмірів екранів
   * Підтримка різних браузерів
2. **Мобільні додатки**
   * Android-додаток
   * iOS-додаток

**2.3 Інтеграції з зовнішніми системами**

1. **Платіжні системи**
   * LiqPay
   * PayPal
   * Apple Pay
   * Google Pay
2. **Картографічні сервіси**
   * Google Maps
   * Mapbox
3. **Сервіси комунікації**
   * SendGrid (email)
   * Twilio (SMS)
   * Firebase Cloud Messaging (push-сповіщення)
4. **Сервіси аналітики та моніторингу**
   * Google Analytics
   * New Relic
   * Sentry

**3. Характеристики, що підлягають тестуванню**

**3.1 Функціональні характеристики**

1. **Функціональність реєстрації та авторизації**
   * Коректність процесу реєстрації та авторизації
   * Захист від несанкціонованого доступу
   * Функціональність відновлення паролю
   * Коректність верифікації користувачів
2. **Функціональність управління оголошеннями**
   * Коректність процесу створення, редагування та видалення оголошень
   * Правильність відображення оголошень на різних пристроях
   * Коректна робота з медіа-файлами
   * Правильність функціонування системи просування оголошень
3. **Функціональність пошуку та фільтрації**
   * Точність пошуку за ключовими словами
   * Коректність роботи фільтрів
   * Правильність сортування результатів
   * Коректність роботи збережених пошукових запитів
4. **Функціональність комунікації**
   * Коректність системи обміну повідомленнями
   * Правильність доставки сповіщень
   * Точність розрахунку рейтингів на основі відгуків
5. **Функціональність транзакцій**
   * Правильність проведення платежів
   * Коректність роботи системи безпечних угод
   * Точність генерації електронних чеків
6. **Функціональність адміністративного модуля**
   * Коректність процесу модерації оголошень
   * Правильність управління користувачами
   * Точність аналітичних даних та звітів

**3.2 Нефункціональні характеристики**

1. **Продуктивність**
   * Час відгуку системи
   * Пропускна здатність
   * Час завантаження сторінок
2. **Безпека**
   * Захист від SQL-ін'єкцій, XSS, CSRF
   * Безпека передачі даних
   * Безпека зберігання паролів та чутливої інформації
3. **Надійність**
   * Відмовостійкість
   * Відновлюваність після збоїв
   * Стабільність при довготривалій роботі
4. **Зручність використання (UI/UX)**
   * Інтуїтивність інтерфейсу
   * Доступність для користувачів з обмеженими можливостями
   * Коректність локалізації
5. **Сумісність**
   * Сумісність з різними браузерами
   * Сумісність з різними версіями операційних систем
   * Коректність роботи на різних пристроях
6. **Масштабованість**
   * Здатність системи обробляти збільшення навантаження
   * Ефективність роботи з великими об'ємами даних

**4. Характеристики, що не підлягають тестуванню**

1. **Характеристики сторонніх систем**
   * Внутрішні алгоритми платіжних систем
   * Внутрішні алгоритми картографічних сервісів
   * Внутрішні алгоритми сервісів комунікації
2. **Характеристики інфраструктури**
   * Конфігурація серверів та мережевого обладнання
   * Налаштування системи моніторингу
   * Налаштування системи резервного копіювання
3. **Характеристики безпеки на рівні мережі та ОС**
   * Налаштування мережевих екранів
   * Налаштування антивірусних систем
   * Налаштування систем виявлення вторгнень

**5. Підходи до тестування**

**5.1 Типи тестування**

1. **Функціональне тестування**
   * Тестування на рівні окремих модулів (Unit Testing)
   * Інтеграційне тестування
   * Системне тестування
   * Тестування взаємодії (End-to-End Testing)
2. **Нефункціональне тестування**
   * Тестування продуктивності
   * Тестування безпеки
   * Тестування зручності використання (Usability Testing)
   * Тестування доступності (Accessibility Testing)
   * Тестування локалізації
3. **Регресійне тестування**
   * Після виправлення помилок
   * Після додавання нових функцій
   * Перед випуском нової версії
4. **Тестування сумісності**
   * Тестування на різних браузерах
   * Тестування на різних пристроях
   * Тестування на різних операційних системах
5. **Тестування безпеки**
   * Тестування на проникнення
   * Тестування на вразливості
   * Тестування автентифікації та авторизації

**5.2 Методи тестування**

1. **Ручне тестування**
   * Для перевірки UI/UX
   * Для перевірки складних сценаріїв, які важко автоматизувати
   * Для дослідницького тестування
2. **Автоматизоване тестування**
   * Unit-тести для перевірки бізнес-логіки
   * API-тести для перевірки взаємодії між компонентами
   * UI-тести для перевірки інтерфейсу користувача
   * Навантажувальні тести для перевірки продуктивності
3. **Дослідницьке тестування**
   * Для виявлення неочевидних проблем
   * Для перевірки граничних випадків

**6. Критерії входу та виходу**

**6.1 Критерії входу**

1. **Для початку тестування**
   * Завершена розробка функціоналу згідно з SRS
   * Успішно пройдені unit-тести
   * Проведено первинний code review
   * Середовище тестування налаштоване та доступне
   * Тестова документація (чек-листи, тест-кейси) підготовлена
2. **Для початку регресійного тестування**
   * Виправлені всі критичні та високопріоритетні помилки
   * Інтеграція нового функціоналу завершена
   * Автоматизовані тести оновлені відповідно до нового функціоналу
3. **Для початку тестування продуктивності**
   * Завершене функціональне тестування
   * Налаштоване середовище, яке відповідає продакшн-середовищу
   * Визначені сценарії та метрики для тестування продуктивності

**6.2 Критерії виходу**

1. **Для завершення тестування**
   * Виконані всі заплановані тест-кейси
   * Всі виявлені критичні та високопріоритетні помилки виправлені та перевірені
   * Всі середньопріоритетні помилки проаналізовані та їх виправлення заплановане
   * Всі автоматизовані тести успішно пройдені
   * Тестування продуктивності показало відповідність системи вимогам
2. **Для випуску версії**
   * Завершене повне регресійне тестування
   * Всі критичні, високопріоритетні та середньопріоритетні помилки виправлені
   * Низькопріоритетні помилки проаналізовані та внесені до беклогу
   * Тестування безпеки не виявило критичних вразливостей
   * Тестування продуктивності показало стабільну роботу системи під навантаженням

**7. Призупинення та відновлення тестування**

**7.1 Критерії призупинення тестування**

1. **Критичні помилки**
   * Виявлення критичних помилок, які блокують подальше тестування
   * Нестабільність середовища тестування
2. **Технічні причини**
   * Недоступність середовища тестування
   * Недоступність тестових даних
   * Недоступність зовнішніх систем, необхідних для тестування
3. **Організаційні причини**
   * Зміни у вимогах, які потребують оновлення тестової документації
   * Зміни у складі команди тестування

**7.2 Критерії відновлення тестування**

1. **Усунення критичних помилок**
   * Виправлення критичних помилок, які блокували тестування
   * Стабілізація середовища тестування
2. **Усунення технічних причин**
   * Відновлення доступу до середовища тестування
   * Відновлення доступу до тестових даних
   * Відновлення доступу до зовнішніх систем
3. **Усунення організаційних причин**
   * Оновлення тестової документації відповідно до змін у вимогах
   * Введення нових членів команди в курс справи

**8. Тестові артефакти**

**8.1 Тестова документація**

1. **Тестовий план**
   * Даний документ, який визначає стратегію тестування
2. **Чек-листи**
   * Перелік перевірок для кожного функціонального модуля
   * Перелік перевірок для нефункціональних характеристик
3. **Тест-кейси**
   * Детальний опис тестових сценаріїв
   * Передумови, кроки, очікувані результати
4. **Звіти про помилки**
   * Опис виявлених помилок
   * Кроки для відтворення
   * Пріоритет та серйозність помилок
5. **Звіти про тестування**
   * Результати виконання тестових наборів
   * Метрики тестування (покриття, кількість виявлених помилок, тощо)

**8.2 Тестові дані**

1. **Набори тестових даних**
   * Тестові облікові записи користувачів
   * Тестові оголошення
   * Тестові транзакції
2. **Скрипти для генерації тестових даних**
   * Скрипти для створення тестових користувачів
   * Скрипти для створення тестових оголошень

**8.3 Автоматизовані тести**

1. **Unit-тести**
   * Тести для перевірки окремих компонентів системи
2. **API-тести**
   * Тести для перевірки взаємодії між компонентами через API
3. **UI-тести**
   * Тести для перевірки інтерфейсу користувача
   * Тести для перевірки адаптивності та відображення на різних пристроях
4. **Навантажувальні тести**
   * Тести для перевірки продуктивності системи під навантаженням

**9. Середовище тестування**

**9.1 Апаратне забезпечення**

1. **Сервери для тестування**
   * Сервери для розгортання тестового середовища
   * Сервери для проведення навантажувальних тестів
2. **Робочі станції тестувальників**
   * Комп'ютери для проведення ручного тестування
   * Комп'ютери для розробки та запуску автоматизованих тестів
3. **Мобільні пристрої**
   * Смартфони та планшети різних виробників з різними версіями ОС
   * Пристрої з різними розмірами екрану та роздільною здатністю

**9.2 Програмне забезпечення**

1. **Операційні системи**
   * Windows 10/11
   * macOS Monterey/Ventura
   * Linux (Ubuntu, Debian)
   * Android 10+
   * iOS 15+
2. **Браузери**
   * Chrome (останні 3 версії)
   * Firefox (останні 3 версії)
   * Safari (останні 3 версії)
   * Edge (останні 3 версії)
   * Internet Explorer 11 (для базової функціональності)
3. **Інструменти для тестування**
   * Selenium WebDriver / Playwright для автоматизації UI-тестів
   * Postman / REST Assured для тестування API
   * JMeter / Gatling для навантажувального тестування
   * Jest / Mocha для unit-тестування
   * OWASP ZAP для тестування безпеки

**9.3 Мережеве оточення**

1. **Конфігурації мережі**
   * Локальна мережа (LAN)
   * Віртуальна приватна мережа (VPN)
   * 3G/4G/5G з'єднання
   * Wi-Fi з'єднання різної якості
2. **Симуляція мережевих умов**
   * Низька пропускна здатність
   * Висока затримка
   * Втрата пакетів

**10. Ролі та відповідальності**

**10.1 Команда тестування**

1. **Тест-менеджер**
   * Планування процесу тестування
   * Розподіл завдань між тестувальниками
   * Контроль виконання плану тестування
   * Звітування про стан тестування
2. **QA інженери**
   * Розробка тестових сценаріїв
   * Проведення ручного тестування
   * Звітування про виявлені помилки
3. **Автоматизатори тестування**
   * Розробка автоматизованих тестів
   * Підтримка та оновлення автоматизованих тестів
   * Проведення автоматизованого тестування
4. **Спеціалісти з тестування продуктивності**
   * Розробка сценаріїв для тестування продуктивності
   * Проведення навантажувального тестування
   * Аналіз результатів та оптимізація продуктивності
5. **Спеціалісти з тестування безпеки**
   * Проведення тестування на проникнення
   * Аналіз безпеки коду
   * Виявлення та оцінка вразливостей

**10.2 Інші ролі**

1. **Розробники**
   * Виправлення виявлених помилок
   * Участь у code review
   * Написання unit-тестів
2. **Проектні менеджери**
   * Координація роботи команди
   * Визначення пріоритетів
   * Управління ризиками
3. **Продуктові менеджери**
   * Уточнення вимог
   * Оцінка якості продукту з точки зору бізнесу
   * Прийняття рішень щодо релізу
4. **UX-дизайнери**
   * Участь у тестуванні зручності використання
   * Аналіз UI/UX-проблем
   * Розробка рекомендацій з покращення UX

**11. Розклад тестування**

**11.1 Етапи тестування**

1. **Планування тестування**
   * Аналіз вимог та документації
   * Розробка тестового плану
   * Підготовка тестового середовища
2. **Розробка тестових сценаріїв**
   * Розробка чек-листів
   * Розробка тест-кейсів
   * Розробка автоматизованих тестів
3. **Виконання тестування**
   * Функціональне тестування
   * Нефункціональне тестування
   * Регресійне тестування
4. **Звітування та аналіз**
   * Аналіз результатів тестування
   * Підготовка звітів
   * Оцінка якості продукту

**11.2 Оцінка часових витрат**

| **Етап тестування** | **Тривалість** | **Залежності** |
| --- | --- | --- |
| Планування тестування | 2 тижні | Завершення розробки вимог |
| Розробка тестових сценаріїв | 3 тижні | Завершення планування тестування |
| Функціональне тестування | 4 тижні | Завершення розробки функціоналу |
| Тестування безпеки | 2 тижні | Завершення функціонального тестування |
| Тестування продуктивності | 2 тижні | Завершення функціонального тестування |
| Регресійне тестування | 2 тижні | Виправлення виявлених помилок |
| Звітування та аналіз | 1 тиждень | Завершення всіх етапів тестування |

**11.3 Віхи проекту**

| **Віха** | **Дата** | **Опис** |
| --- | --- | --- |
| Затвердження тестового плану | 01.10.2025 | Тестовий план узгоджений та затверджений |
| Завершення розробки тестових сценаріїв | 22.10.2025 | Розроблені всі чек-листи та тест-кейси |
| Завершення функціонального тестування | 19.11.2025 | Завершено тестування всіх функціональних модулів |
| Завершення нефункціонального тестування | 03.12.2025 | Завершено тестування продуктивності, безпеки та UI/UX |
| Завершення регресійного тестування | 17.12.2025 | Завершено перевірку виправлення помилок та регресійне тестування |
| Готовність до випуску | 24.12.2025 | Продукт готовий до випуску, підготовлені всі звіти |

**12. Ризики та пом'якшення**

**12.1 Ризики тестування**

1. **Технічні ризики**
   * Недоступність середовища тестування
   * Проблеми з інтеграцією з зовнішніми системами
   * Нестабільність автоматизованих тестів
2. **Ризики планування**
   * Недостатність часу для повного тестування
   * Зміни у вимогах, які потребують перепланування тестування
   * Залежності від інших команд або зовнішніх факторів
3. **Ресурсні ризики**
   * Недостатність кваліфікованих тестувальників
   * Обмеженість технічних ресурсів
   * Відсутність специфічних знань або інструментів

**12.2 Пом'якшення ризиків**

1. **Пом'якшення технічних ризиків**
   * Раннє налаштування середовища тестування
   * Використання mock-об'єктів для симуляції зовнішніх систем
   * Регулярне оновлення та підтримка автоматизованих тестів
2. **Пом'якшення ризиків планування**
   * Додавання буферного часу у розклад тестування
   * Пріоритезація тестування критичних функцій
   * Гнучкість у плануванні та можливість коригування плану
3. **Пом'якшення ресурсних ризиків**
   * Раннє планування потреб у ресурсах
   * Навчання та розвиток команди тестування
   * Використання зовнішніх експертів для специфічних завдань

**13. Метрики та звітність**

**13.1 Метрики тестування**

1. **Метрики покриття**
   * Покриття вимог
   * Покриття коду
   * Покриття функціоналу
2. **Метрики дефектів**
   * Кількість виявлених дефектів за пріоритетом
   * Кількість виправлених дефектів
   * Швидкість виявлення та виправлення дефектів
3. **Метрики прогресу**
   * Відсоток виконаних тест-кейсів
   * Відсоток успішних тест-кейсів
   * Витрачений час на тестування
4. **Метрики продуктивності**
   * Час відгуку системи
   * Пропускна здатність
   * Споживання ресурсів

**13.2 Звітність**

1. **Щоденні звіти**
   * Прогрес виконання тест-кейсів
   * Виявлені критичні помилки
   * Блокери та ризики
2. **Щотижневі звіти**
   * Загальний прогрес тестування
   * Статистика виявлених та виправлених помилок
   * Оновлення щодо ризиків та їх пом'якшення
3. **Звіти про тестування релізу**
   * Загальний статус тестування
   * Відкриті та вирішені проблеми
   * Рекомендації щодо релізу

**14. Інструменти для тестування**

**14.1 Інструменти для управління тестуванням**

1. **JIRA / TestRail**
   * Управління тестовими сценаріями
   * Відстеження виконання тест-кейсів
   * Звітування про прогрес тестування
2. **Confluence**
   * Зберігання тестової документації
   * Спільна робота над тестовими артефактами
   * Обмін знаннями та інформацією

**14.2 Інструменти для автоматизації тестування**

1. **Selenium WebDriver / Playwright**
   * Автоматизація UI-тестів для веб-додатку
   * Тестування на різних браузерах та пристроях
   * Інтеграція з системами безперервної інтеграції
2. **Appium**
   * Автоматизація тестування мобільних додатків
   * Тестування на реальних пристроях та емуляторах
   * Крос-платформне тестування
3. **Postman / REST Assured**
   * Тестування API
   * Автоматизація API-тестів
   * Валідація відповідей API

**14.3 Інструменти для тестування продуктивності**

1. **JMeter / Gatling**
   * Навантажувальне тестування
   * Стрес-тестування
   * Тестування витривалості
2. **New Relic / Datadog**
   * Моніторинг продуктивності
   * Збір та аналіз метрик
   * Виявлення вузьких місць

**14.4 Інструменти для тестування безпеки**

1. **OWASP ZAP**
   * Сканування вразливостей
   * Перевірка на відомі атаки
   * Аналіз безпеки веб-додатків
2. **Burp Suite**
   * Перехоплення та аналіз HTTP-трафіку
   * Тестування на проникнення
   * Виявлення вразливостей у веб-додатках

**15. Процес виявлення та управління дефектами**

**15.1 Класифікація дефектів**

1. **За пріоритетом**
   * P1 (Критичний): Блокує роботу системи або критичного функціоналу
   * P2 (Високий): Серйозно впливає на роботу системи, але є обхідні шляхи
   * P3 (Середній): Впливає на роботу окремих функцій, але не блокує основний функціонал
   * P4 (Низький): Мінімальний вплив на роботу системи, косметичні проблеми
2. **За типом**
   * Функціональні дефекти
   * Дефекти UI/UX
   * Дефекти продуктивності
   * Дефекти безпеки
   * Дефекти сумісності
   * Дефекти локалізації
   * Дефекти документації
3. **За статусом**
   * Новий
   * Призначений
   * В роботі
   * Виправлений
   * Перевірений
   * Закритий
   * Відхилений

**15.2 Процес звітування про дефекти**

1. **Виявлення дефекту**
   * Тестувальник виявляє проблему під час тестування
   * Збирає всі необхідні дані для відтворення
2. **Створення звіту про дефект**
   * Заповнення всіх необхідних полів у системі відстеження дефектів
   * Опис кроків для відтворення
   * Очікуваний та фактичний результат
   * Додавання скріншотів, логів та інших артефактів
3. **Тріаж дефектів**
   * Оцінка пріоритету та серйозності
   * Перевірка на дублювання
   * Призначення відповідальних за виправлення
4. **Відстеження та закриття**
   * Моніторинг статусу виправлення
   * Перевірка виправлення
   * Закриття дефекту після успішної перевірки

**15.3 Шаблон звіту про дефект**

1. **Заголовок**: Короткий опис проблеми
2. **ID**: Унікальний ідентифікатор
3. **Пріоритет**: P1/P2/P3/P4
4. **Серйозність**: Критична/Висока/Середня/Низька
5. **Статус**: Новий/Призначений/В роботі/Виправлений/Перевірений/Закритий/Відхилений
6. **Версія**: Версія ПЗ, в якій виявлено дефект
7. **Модуль**: Функціональний модуль, в якому виявлено дефект
8. **Середовище**: Операційна система, браузер, пристрій
9. **Передумови**: Умови, необхідні для відтворення дефекту
10. **Кроки відтворення**: Покрокова інструкція з відтворення дефекту
11. **Очікуваний результат**: Опис правильної поведінки системи
12. **Фактичний результат**: Опис поведінки системи з дефектом
13. **Додаткова інформація**: Скріншоти, логи, відео
14. **Дата виявлення**: Дата та час виявлення дефекту
15. **Звітував**: Ім'я тестувальника, який виявив дефект
16. **Призначений**: Ім'я розробника, відповідального за виправлення

**16. Тестування юзабіліті**

**16.1 Цілі тестування юзабіліті**

1. **Оцінка загального користувацького досвіду**
   * Визначення зручності використання інтерфейсу
   * Оцінка інтуїтивності навігації
   * Виявлення проблем з доступністю
2. **Оцінка ефективності виконання завдань**
   * Вимірювання часу на виконання типових завдань
   * Виявлення точок фрустрації користувачів
   * Аналіз шляхів користувачів через інтерфейс
3. **Збір відгуків користувачів**
   * Отримання суб'єктивних оцінок користувачів
   * Збір пропозицій щодо покращення інтерфейсу
   * Визначення пріоритетів для подальшого вдосконалення

**16.2 Методи тестування юзабіліті**

1. **Модеровані сесії**
   * Безпосереднє спостереження за користувачами під час виконання завдань
   * Інтерв'ю з користувачами до та після тестування
   * Аналіз поведінки користувачів та їх коментарів
2. **Немодеровані сесії**
   * Віддалене тестування з записом екрану
   * Збір аналітичних даних про поведінку користувачів
   * Опитування користувачів після завершення сесії
3. **A/B тестування**
   * Порівняння різних варіантів інтерфейсу
   * Вимірювання ефективності різних дизайнерських рішень
   * Визначення оптимального варіанту на основі метрик

**16.3 Сценарії для тестування юзабіліті**

1. **Реєстрація та авторизація**
   * Створення нового облікового запису
   * Авторизація з використанням електронної пошти та паролю
   * Авторизація через соціальні мережі
   * Відновлення забутого паролю
2. **Управління оголошеннями**
   * Створення нового оголошення
   * Завантаження фотографій та відео
   * Редагування існуючого оголошення
   * Видалення оголошення
   * Просування оголошення
3. **Пошук та фільтрація**
   * Пошук товарів за ключовими словами
   * Використання фільтрів для уточнення результатів
   * Сортування результатів пошуку
   * Збереження пошукових запитів
4. **Комунікація**
   * Написання повідомлення продавцю
   * Відповідь на отримане повідомлення
   * Перегляд історії повідомлень
   * Залишення відгуку про користувача
5. **Транзакції**
   * Оплата преміум-послуг
   * Використання системи безпечних угод
   * Перегляд історії транзакцій

**16.4 Метрики для оцінки юзабіліті**

1. **Об'єктивні метрики**
   * Час виконання завдання
   * Кількість помилок при виконанні завдання
   * Рівень успішного завершення завдання
   * Кількість кліків для досягнення мети
2. **Суб'єктивні метрики**
   * Рівень задоволеності користувача (на основі опитувань)
   * System Usability Scale (SUS)
   * Net Promoter Score (NPS)
   * Оцінка зрозумілості інтерфейсу

**17. Автоматизація тестування**

**17.1 Стратегія автоматизації**

1. **Рівні автоматизації**
   * Unit-тестування: автоматизація 90% функціоналу
   * API-тестування: автоматизація 80% функціоналу
   * UI-тестування: автоматизація 60% критичних шляхів користувачів
2. **Пріоритизація автоматизації**
   * Критичні бізнес-процеси
   * Часто використовуваний функціонал
   * Стабільний функціонал, який рідко змінюється
   * Функціонал з високим ризиком регресії
3. **Підходи до автоматизації**
   * Page Object Model для UI-тестування
   * Data-driven testing для тестування з різними наборами даних
   * Keyword-driven testing для спрощення підтримки тестів

**17.2 Технічні рішення для автоматизації**

1. **Фреймворки та бібліотеки**
   * Selenium WebDriver / Playwright для UI-тестування
   * Appium для тестування мобільних додатків
   * REST Assured / Axios для API-тестування
   * JUnit / Jest для unit-тестування
2. **Інструменти для CI/CD**
   * Jenkins / GitHub Actions для безперервної інтеграції
   * Docker для створення ізольованих середовищ
   * Allure для звітності
3. **Інструменти для управління тестовими даними**
   * Faker для генерації тестових даних
   * Бази даних для зберігання тестових даних
   * API для створення та видалення тестових даних

**17.3 План впровадження автоматизації**

1. **Фаза 1: Підготовка**
   * Налаштування середовища для автоматизації
   * Вибір та налаштування інструментів
   * Навчання команди
2. **Фаза 2: Пілотна автоматизація**
   * Автоматизація базових тест-кейсів
   * Валідація підходу та інструментів
   * Збір зворотного зв'язку та внесення коректив
3. **Фаза 3: Масштабування**
   * Автоматизація більшої кількості тест-кейсів
   * Інтеграція з CI/CD
   * Налаштування регулярного запуску автотестів
4. **Фаза 4: Підтримка та оптимізація**
   * Регулярне оновлення та підтримка автотестів
   * Оптимізація часу виконання
   * Розширення покриття автотестами

**18. Додатки**

**18.1 Глосарій**

| **Термін** | **Визначення** |
| --- | --- |
| SRS | Software Requirements Specification - Специфікація вимог до програмного забезпечення |
| UI | User Interface - Інтерфейс користувача |
| UX | User Experience - Досвід користувача |
| API | Application Programming Interface - Програмний інтерфейс додатку |
| CI/CD | Continuous Integration/Continuous Delivery - Безперервна інтеграція та доставка |
| QA | Quality Assurance - Забезпечення якості |
| SUS | System Usability Scale - Шкала зручності використання системи |
| NPS | Net Promoter Score - Індекс споживчої лояльності |
| Mock | Об'єкт, що імітує поведінку реального об'єкту для тестування |
| Regression Testing | Тестування, яке проводиться після внесення змін, для перевірки, що існуючий функціонал продовжує працювати коректно |
| Smoke Testing | Базове тестування, яке проводиться для перевірки основного функціоналу |
| Sanity Testing | Тестування, яке проводиться для перевірки логічності та стабільності системи |

**18.2 Шаблони тестових артефактів**

1. **Шаблон чек-листа**
   * Розділ: Назва функціонального модуля або області тестування
   * Підрозділ: Деталізація функціональності
   * Перевірки: Список перевірок з можливістю відмітки статусу (Пройдено/Не пройдено/Пропущено)
   * Коментарі: Місце для додаткової інформації
2. **Шаблон тест-кейсу**
   * ID: Унікальний ідентифікатор
   * Назва: Коротка назва, що відображає суть тест-кейсу
   * Передумови: Умови, необхідні для виконання тест-кейсу
   * Кроки: Покрокова інструкція з виконання тест-кейсу
   * Очікуваний результат: Опис правильної поведінки системи
   * Фактичний результат: Поле для заповнення під час виконання
   * Статус: Пройдено/Не пройдено/Заблоковано
   * Примітки: Додаткова інформація
3. **Шаблон звіту про тестування**
   * Дата та час тестування
   * Версія продукту
   * Середовище тестування
   * Виконані тест-кейси (кількість та перелік)
   * Результати тестування (% успішних, % невдалих)
   * Виявлені дефекти (кількість за пріоритетами)
   * Рекомендації
   * Загальний висновок

**18.3 Приклади тест-кейсів**

1. **TC-001: Реєстрація нового користувача з валідними даними**
   * **Передумови:** Відкрита сторінка реєстрації, користувач з вказаною електронною поштою не зареєстрований в системі
   * **Кроки:**
     1. Ввести валідну електронну пошту
     2. Ввести валідний телефон
     3. Ввести пароль (мінімум 8 символів, включаючи літери та цифри)
     4. Підтвердити пароль
     5. Натиснути кнопку "Зареєструватися"
     6. Ввести код підтвердження, отриманий на електронну пошту
   * **Очікуваний результат:** Користувач успішно зареєстрований, автоматично авторизований в системі та перенаправлений на головну сторінку з персоналізованим контентом
2. **TC-002: Створення нового оголошення з усіма обов'язковими полями**
   * **Передумови:** Користувач авторизований в системі, відкрита сторінка створення оголошення
   * **Кроки:**
     1. Вибрати категорію "Електроніка"
     2. Вибрати підкатегорію "Мобільні телефони"
     3. Ввести заголовок "iPhone 13 Pro Max 256GB"
     4. Ввести опис "Новий, не розпакований iPhone 13 Pro Max 256GB, колір - графітовий"
     5. Ввести ціну "35000"
     6. Завантажити 5 фотографій товару
     7. Вибрати місцезнаходження "Київ"
     8. Натиснути кнопку "Опублікувати"
   * **Очікуваний результат:** Оголошення успішно створено, користувач перенаправлений на сторінку перегляду створеного оголошення

**18.4 Контрольні списки для тестування**

1. **Контрольний список для тестування реєстрації та авторизації**
   * Реєстрація з валідними даними
   * Реєстрація з невалідною електронною поштою
   * Реєстрація з невалідним телефоном
   * Реєстрація з коротким паролем
   * Реєстрація з неспівпадаючими паролями
   * Авторизація з валідними даними
   * Авторизація з невалідними даними
   * Відновлення паролю
   * Авторизація через Facebook
   * Авторизація через Google
   * Вихід з системи
2. **Контрольний список для тестування створення оголошень**
   * Створення оголошення з усіма обов'язковими полями
   * Створення оголошення без заголовку
   * Створення оголошення без опису
   * Створення оголошення без ціни
   * Створення оголошення без фотографій
   * Створення оголошення без вибору місцезнаходження
   * Завантаження фотографій різних форматів
   * Завантаження фотографій розміром більше 10 MB
   * Створення оголошення з максимальною кількістю фотографій (20)
   * Створення оголошення з відео
   * Створення оголошення з активацією преміум-послуг

**19. Підписи та затвердження**

| **Роль** | **ПІБ** | **Дата** |
| --- | --- | --- |
| Тест-менеджер |  |  |
| Керівник проекту |  |  |
| Представник команди розробки |  |  |