

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет комп'ютерних наук та кібернетики
Кафедра інтелектуальних програмних систем

Лабораторна робота №5

з дисципліни «Методи забезпечення якості програмних систем»
на тему: «Юзабіліті-тестування для веб-сайту «Binance» та її API
інтерфейсів»

Виконав:
Студент групи ПЗС-1
Лутай Артем Сергійович

КИЇВ 2025

Вступ

Метою даної лабораторної роботи є проведення юзабіліті-тестування торгової платформи «Binance» для оцінки її зручності (Usability), ефективності та зрозумілості інтерфейсу. Основний акцент робиться на визначенні того, наскільки швидко та безпомилково користувач може виконати критичні дії: знайти торгову пару, розмістити ордер, керувати активами та налаштувати доступ до API.

У ході роботи передбачено виконання таких **основних завдань**:

1. Перевірити інтуїтивність навігації по ринкових даних (Markets).
2. Оцінити складність процесу створення ордера (Buy/Sell) для нових користувачів.
3. Визначити бар'єри при взаємодії з розділом безпеки та генерації API-ключів.
4. Зібрати суб'єктивні оцінки задоволеності (Satisfaction) та зворотний зв'язок (Feedback).

Об'єктом тестування виступає веб-версія платформи Binance (Desktop), фокусуючись на ключових модулях: «Торговий термінал», «Гаманець» та «Управління API» (API Management).

Для проведення дослідження залучено **учасників тестування** (5 респондентів) з різним рівнем досвіду, що дозволяє отримати репрезентативні дані для різних категорій юзерів:

1. Група А (2 особи) — Expert Users: Досвідчені трейдери, які знайомі з інтерфейсами інших криптобірж або торгових терміналів.
2. Група Б (3 особи) — Novice Users: Початківці, які мають базові теоретичні знання про криптовалюту, але не мають практичного досвіду торгівлі на Binance.

Тестування проводиться на персональних комп'ютерах (ноутбуках) під управлінням ОС GNU/Linux та macOS з використанням сучасних браузерів (Firefox / Safari) та стабільним підключенням до мережі Інтернет.

Метрики оцінювання

Для кількісної та якісної оцінки ефективності виконання сценаріїв тестування використовуються наступні метрики:

1. **Успішність виконання (Success Rate):** Відсоток користувачів, які змогли успішно завершити завдання без критичних помилок та сторонньої допомоги.
 - a. *Розрахунок:* $(\text{Кількість успішних спроб} / \text{Загальна кількість спроб}) \times 100\%$.
 - b. *Ціль для Binance:* Очікується рівень $> 90\%$ для базових завдань (вхід, пошук) та $> 80\%$ для складних (створення API, ОСО-ордери).
2. **Час виконання завдання (Time on Task):** Середній час (у секундах), витрачений респондентом на виконання сценарію від моменту початку до досягнення мети.
 - a. *Важливість:* Критична метрика для трейдингу, де швидкість реакції впливає на фінансовий результат.
 - b. *Аналіз:* Порівняння часу досвідчених трейдерів (Benchmark) з часом новачків для оцінки кривої навчання.
3. **Кількість помилок (Error Rate):** Середня кількість помилкових дій користувача під час виконання одного сценарію.
 - a. *Типи помилок:* Кліки на неактивні елементи, вибір невірної мережі для транзакції (напр., ERC20 замість TRC20), помилки валідації форм.
 - b. *Критичність:* Висока для модулів «Гаманець» та «API», де помилка може призвести до втрати коштів.
4. **Суб'єктивна задоволеність (Satisfaction):** Оцінка зручності використання інтерфейсу, яку користувач виставляє після завершення сценарію за шкалою від 1 до 5 (де 1 — «Дуже незручно/Складно», 5 — «Дуже зручно/Інтуїтивно»).
 - a. Додатково фіксуються коментарі респондентів (Verbal Feedback) щодо логіки інтерфейсу та візуального оформлення.

Сценарії тестування

Для проведення юзабіліті-тестування було розроблено 5 ключових сценаріїв, які охоплюють найбільш важливі аспекти взаємодії з платформою Binance. Сценарії побудовані за принципом зростання складності: від базової навігації до специфічних налаштувань безпеки та API.

Сценарій 1. Пошук торгової пари та аналіз графіка

Цей сценарій перевіряє базову навігацію сайту та зручність пошукового інтерфейсу.

- **Мета:** Перевірити, наскільки швидко та інтуїтивно користувач може знайти необхідний фінансовий інструмент серед великої кількості доступних активів.
- **Завдання для користувача:** «Уявіть, що ви хочете проаналізувати курс Біткоїна до долара (USDT). Знайдіть відповідну торгову пару та відкрийте її графік».
- **Оптимальний шлях виконання (Happy Path):**
 1. Зайти на головну сторінку Binance.
 2. У головному меню обрати розділ «Ринки» (Markets) або скористатися пошуковим рядком.
 3. У полі пошуку ввести тікер «BTC».
 4. У списку результатів обрати пару «BTC/USDT».
 5. Клікнути на рядок пари, щоб перейти до торгового терміналу.
- **Критерій успіху:** Користувач успішно відкрив сторінку спотової торгівлі для пари BTC/USDT.

Сценарій 2. Розміщення лімітного ордера (Limit Order)

Сценарій спрямований на оцінку зручності торгової форми (Order Form) — основного інструменту трейдера.

- **Мета:** Оцінити зрозумілість інтерфейсу створення ордерів для користувача, який хоче купити актив не за поточною ціною, а дешевше.
- **Завдання для користувача:** «Ви хочете купити Біткоїн, але чекаєте, поки ціна впаде до 20,000 USDT. Створіть ордер на купівлю 0.001 BTC за цією ціною».
- **Оптимальний шлях виконання:**

1. У торговому терміналі (на сторінці пари BTC/USDT) знайти блок оформлення ордера.
 2. Переконалися, що обрано вкладку «Limit» (а не Market).
 3. У поле «Price» (Ціна) ввести значення 20000.
 4. У поле «Amount» (Кількість) ввести 0.001.
 5. Натиснути кнопку «Buy BTC».
- **Критерій успіху:** Ордер успішно створено, і він відображається у вкладці «Open Orders» (Відкриті ордери) в нижній частині екрана.

Сценарій 3. Генерація адреси депозиту (Wallet Deposit)

Цей сценарій перевіряє логіку роботи з гаманцем та розуміння користувачем важливості вибору мережі блокчейну.

- **Мета:** Перевірити, чи легко користувачу знайти реквізити для поповнення рахунку та чи звертає він увагу на вибір мережі.
- **Завдання для користувача:** «Вам потрібно поповнити свій баланс у стейблкоїні USDT через мережу Tron (TRC20). Знайдіть вашу адресу гаманця для цієї операції».
- **Оптимальний шлях виконання:**
 1. У меню профілю обрати «Гаманець» -> «Фіат і спот» (Wallet -> Fiat and Spot).
 2. У списку монет знайти USDT.
 3. Натиснути кнопку «Депозит» (Deposit) навпроти монети.
 4. У випадяючому списку «Network» (Мережа) обрати «TRX Tron (TRC20)».

Сценарій 4. Створення API-ключа (API Management)

Критичний сценарій для теми алгоритмічної торгівлі. Перевіряє доступність та зрозумілість налаштувань для підключення торгових ботів.

- **Мета:** Оцінити зручність процесу створення ключів доступу для зовнішнього ПЗ та розуміння вимог безпеки.
- **Завдання для користувача:** «Ви хочете підключити торгового бота. Створіть новий API-ключ з назвою "TestBot" і збережіть секретний ключ».
- **Оптимальний шлях виконання:**
 1. Навести курсор на іконку профілю та обрати «Управління API» (API Management).

2. Натиснути жовту кнопку «Створити API» (Create API).
 3. Обрати тип ключа «System generated».
 4. Ввести мітку (Label) — "TestBot".
 5. Пройти перевірку безпеки (ввести коди з email/SMS/2FA).
 6. Скопіювати або зберегти Secret Key, розуміючи, що він буде прихований після виходу зі сторінки.
- **Критерій успіху:** API-ключ успішно створено, користувач усвідомив необхідність негайного збереження Secret Key.

Сценарій 5. Налаштування безпеки (Anti-phishing Code)

Сценарій перевіряє, наскільки легко користувач може знайти специфічні налаштування захисту акаунту.

- **Мета:** Перевірити навігацію в розділі безпеки та здатність користувача активувати додаткові механізми захисту від фішингу.
- **Завдання для користувача:** «Щоб відрізнити справжні листи від Binance від підроблених, встановіть Анти-фішинг код "NEVER.GONNA.GIVE.YOU.UP"».
- **Оптимальний шлях виконання:**
 1. Перейти в меню профілю -> «Безпека» (Security).
 2. Прокролити сторінку до розділу «Розширені налаштування» (Advanced Security).
 3. Знайти пункт «Анти-фішинг код» (Anti-phishing Code) і натиснути «Увімкнути» (Enable).
 4. Натиснути «Створити код», ввести "NEVER.GONNA.GIVE.YOU.UP" та підтвердити дію.

Результати та фідбек користувачів

Нижче наведено детальний протокол тестування для Учасника №1 (Група А — Експерт). Досвідчений трейдер (3+ роки досвіду), активно використовує алгоритмічну торгівлю, знайомий з інтерфейсами Bybit/OKX.

Сценарій 1. Пошук торгової пари та аналіз графіка

- Success Rate (Успішність): 100% (Виконано без допомоги).
- Time on Task (Час): 12 секунд.
- Error Rate (Помилки): 0.
- Satisfaction (Задоволеність): 5 / 5.
- Спостереження: Користувач не використовував меню «Ринки», а відразу скористався глобальним пошуком (Ctrl+K / іконка лупи) або пошуком всередині терміналу, що є типовим патерном для профі.
- Фідбек учасника:
"Пошук працює миттєво, це плюс. Сподобалось, що графік TradingView підтягується автоматично з моїми збереженими індикаторами, якщо я залогінений."

Сценарій 2. Розміщення лімітного ордера (Limit Order)

- Success Rate (Успішність): 100%.
- Time on Task (Час): 20 секунд.
- Error Rate (Помилки): 0.
- Satisfaction (Задоволеність): 4.5 / 5.
- Спостереження: Учасник швидко знайшов форму ордера. Використав слайдер відсотків (25% / 50%) замість ручного введення кількості.
- Фідбек учасника:
"Форма стандартна, звична. Трохи дрібний шрифт у стакані (Order Book), довелося придивлятися до спреду. Але функціонал працює чітко. Добре, що видно комісію в BNB."

Сценарій 3. Генерація адреси депозиту (Wallet)

- Success Rate (Успішність): 100%.

- Time on Task (Час): 30 секунд.
- Error Rate (Помилки): 0.
- Satisfaction (Задоволеність): 5 / 5.
- Спостереження: Учасник уважно перевіряв обрану мережу (Network). Зазначив, що система запам'ятовує останню обрану мережу, що зручно.
- Фідбек учасника:
"Все зрозуміло. Великий плюс за жирне попередження про вибір мережі — новачки часто плутають ERC20 і TRC20. Тут помилитися важко."

Сценарій 4. Створення API-ключа (API Management)

- Success Rate (Успішність): 100%.
- Time on Task (Час): 65 секунд.
- Error Rate (Помилки): 0.
- Satisfaction (Задоволеність): 4 / 5.
- Спостереження: Основний час пішов на проходження перевірки безпеки (Email + Authenticator). Учасник відразу скопіював Secret Key у свій менеджер паролів.
- Фідбек учасника:
"Процес трохи затягнутий через 2FA, але для грошей це виправдано. Єдине зауваження — хотілося б кнопку 'Copy All', щоб скопіювати API Key і Secret Key одним кліком у JSON форматі для конфігів бота. Зараз доводиться копіювати по черзі."

Сценарій 5. Увімкнення анти-фішинг коду (Security)

- Success Rate (Успішність): 100%.
- Time on Task (Час): 45 секунд.
- Error Rate (Помилки): 1 (спочатку зайшов у "Settings" замість "Security").
- Satisfaction (Задоволеність): 4 / 5.
- Спостереження: Користувач знав, що така функція існує, але витратив кілька секунд на пошук потрібної вкладки в великому меню профілю.

- Фідбек учасника:
"Корисна фіча. Меню налаштувань безпеки дуже велике, довелося скролити, щоб знайти цей пункт. Можливо, варто підняти його вище або додати пошук по налаштуваннях."

Загальний висновок по Учаснику №1

Користувач продемонстрував високу ефективність взаємодії з системою. Інтерфейс оцінено як "професійний та надійний".

- **Середня оцінка зручності:** 4.5 / 5.
- **Критичні проблеми:** Відсутні.
- **Рекомендації від учасника:** Додати можливість швидкого копіювання пари ключів (API+Secret) та пошук у налаштуваннях безпеки.