

Кинутий кошик: практичний аудит checkout

Як знайти бар'єр перед оформленням, описати проблему та визначити наступну дію

У ЦЬОМУ ГАЙДІ

Кинутий кошик не є однією проблемою. Покупець може зупинитися ще в кошику, на виборі доставки, під час заповнення форми, на оплаті або вже після створення замовлення. Робота починається з поділу цих станів, ручного проходження сценарію та фіксації кожної точки, де покупець не може зрозуміло продовжити дію.

- 01 1. Визначте, який саме сценарій не завершився**
- 02 2. Пройдіть маршрут покупця та зафіксуйте зміни стану**
- 03 3. Ведіть журнал перевірки та звіряйте його з даними**
- 04 4. Виправляйте бар'єри в правильному порядку**
- 05 5. Відокремте локальне виправлення від потреби перебудови**
- 06 6. Запускайте повернення до кошика лише після перевірки**
- Q Часті питання**

01 · 1. ВИЗНАЧТЕ, ЯКИЙ САМЕ СЦЕНАРІЙ НЕ ЗАВЕРШИВСЯ

Перш ніж шукати причину, відокремте незавершене оформлення від інших станів замовлення. Це прибирає хибні висновки: не доведеться змінювати форму, якщо проблема виникає під час оплати, або запускати нагадування для замовлень, які вже створилися й були скасовані.

Для робочої перевірки вважайте незавершеним сценарієм товар у кошику або почате оформлення без підтвердженого замовлення в бізнес-процесі. Це визначення потрібне для аудиту шляху, а не для пояснення мотиву кожного покупця.

Не об'єднуйте в одному висновку товар, залишений у кошику, незавершений checkout, непідтверджений платіж і скасоване замовлення. Це різні стани з різними власниками проблеми: інтерфейсом товару та кошика, логікою доставки, платіжною інтеграцією або внутрішньою обробкою замовлень.

01 Складіть перелік станів

Окремо запишіть: товар залишився в кошику; checkout відкрито без підтвердження; покупець перейшов до оплати без її підтвердження; замовлення створено й потім скасовано.

02 Призначте перевірку для кожного стану

Для кошика перевіряйте товар, ціну, підсумок і перехід до оформлення. Для оплати — повернення з платіжної сторінки, повідомлення для покупця та статус замовлення.

03 Відокремте скасування замовлень

Не включайте скасовані замовлення до незавершеного checkout. Вони потребують окремого розбору причин скасування та процесу обробки.

04 Зафіксуйте робоче визначення для команди

Домовтеся, які саме стани команда називає кинутим кошиком у внутрішніх звітах. Це дасть маркетингу, менеджерам і розробникам одну точку відліку.

ДЕ ВТРАЧАЮТЬ

Найчастіша втрата часу — рахувати скасовані замовлення разом із незавершеним checkout і шукати проблему в кнопці оплати, хоча замовлення вже було створене.

ЧЕКЛІСТ РОЗДІЛУ

- ☐ Розділіть кошик, незавершене оформлення, непідтверджений платіж і скасоване замовлення.

Без цього один звіт змішує різні причини та веде до неправильних змін.

- ☐ Визначте, який стан є предметом поточної перевірки.

Без чіткої межі команда не зможе перевірити, чи зміна справді вирішила проблему.

- ☐ Призначте відповідального за перевірку кожного стану.

Без розподілу ролей проблема між checkout, оплатою й адмінчастиною залишатиметься нічийною.

02 · 2. ПРОЙДІТЬ МАРШРУТ ПОКУПЦЯ ТА ЗАФІКСУЙТЕ ЗМІНИ СТАНУ

Ручна перевірка показує не припущення, а конкретний момент, де сценарій ламається або стає незрозумілим. Її проходять від картки товару до підтвердження замовлення, не пропускаючи звичні для команди кроки.

Перевіряйте checkout як послідовність контрольних точок. На кожній із них покупець має розуміти поточний стан, наступну дію та умови, які впливають на суму, доставку, оплату або можливість завершити замовлення.

Під час тесту записуйте не загальне враження, а зміну стану: що було на екрані, яку дію виконали, що система показала після неї. Такий запис дозволяє повторити проблему й передати її без переказів та здогадок.

01 Перевірте картку товару

Переконайтеся, що видно наявність товару, вибраний варіант і ціну. Зіставте ціну в картці з ціною після додавання в кошик.

02 Протестуйте кошик

Додайте товар, змініть кількість, видаліть його та додайте знову. Перевірте підсумкову суму, умови доставки, можливість повернутися до покупок і збереження вибору.

03 Перейдіть до оформлення

Переконайтеся, що товари й обрані параметри не зникли. Перевірте, чи очевидний наступний крок і чи доступні ключові умови до переходу на оплату.

04 Пройдіть усі доступні варіанти доставки

Оберіть кожен доступний спосіб і з'ясуйте, чи пояснює інтерфейс причину недоступності варіанта. Умова має бути зрозуміла до того, як покупець витратив час на наступні кроки.

05 Перевірте форму та оплату

Лишіть обов'язкове поле порожнім, перевірте текст і розташування помилки, а також збереження введених даних. Потім пройдіть кожен доступний спосіб оплати до повернення на сайт.

06 Зіставте підтвердження з фактом створення замовлення

Після тестового замовлення перевірте повідомлення для покупця та появу замовлення в адмінчастині. Повернення з оплати саме по собі не підтверджує створення замовлення.

ДЕ ВТРАЧАЮТЬ

Не тестуйте лише кнопку оплати. Втрата вибраного товару, зміна суми, недоступна доставка або очищення форми зупиняють покупця раніше.

ЧЕКЛІСТ РОЗДІЛУ

- ☐ **Пройдіть шлях від товару до підтвердження без пропуску кроків.**
Без повного маршруту легко не помітити, де зникає стан кошика або змінюються умови.
- ☐ **Перевірте кожен доступний спосіб доставки та оплати.**
Без цього нестабільний або недоступний окремий варіант залишиться прихованим.
- ☐ **Зіставте сторінку успіху зі статусом замовлення в адмінчастині.**
Без цього можна помилково прийняти повернення з оплати за успішне замовлення.

Завдання аудиту — перетворити спостереження на відтворюваний опис проблеми. Для цього потрібен один журнал перевірки, де ручний маршрут можна зіставити з подіями, які вже збирає аналітика.

Не створюйте у звіті події, яких система не збирає. Спочатку з'ясуйте, чи видно додавання товару, відкриття кошика, початок оформлення, вибір доставки й оплати, помилку форми, повернення з платіжної сторінки та створення замовлення.

Назви подій можуть бути іншими, але логіка має зберігатися: дія покупця, зміна стану та результат. Аналітика не замінює ручну перевірку, проте допомагає зрозуміти, чи зафіксований момент зупинки в реальному маршруті.

01 Створіть журнал проходу

Для кожного кроку записуйте: час перевірки, пристрій або середовище, дію, умову, очікуваний результат, фактичний результат і посилання на скриншот за потреби.

02 Позначте контрольні події

Використайте внутрішні назви на кшталт: товар відкрито, товар додано в кошик, кошик змінено, оформлення відкрито, доставку обрано, оплату обрано, форма показала помилку, оплату підтверджено, замовлення створено.

03 Зіставте події з ручним маршрутом

Перевірте, чи з'являється в даних кожен важливий перехід. Якщо події немає, позначте це як обмеження спостереження, а не як доказ відсутності проблеми.

04 Опишіть відтворювану проблему

Формулюйте її у чотирьох частинах: дія, умова, очікуваний результат, фактичний результат. Наприклад, не «оформлення незручне», а точний опис моменту зупинки.

05 Розподіліть ролі

Власник магазину підтверджує бізнес-умови, менеджер перевіряє появу й статус замовлення, розробник відтворює та виправляє технічний збій, відповідальний за аналітику підтверджує наявність подій.

ДЕ ВТРАЧАЮТЬ

Фраза «покупці кидають кошик» не є задачею для розробки. Без дії, умови та фактичного результату команда почне змінювати checkout навмання.

ЧЕКЛІСТ РОЗДІЛУ

☐ Заповніть журнал для всього маршруту, а не лише для точки помилки.

Без контексту неможливо зрозуміти, яка попередня дія створила проблему.

☐ **Перевірте, які події реально доступні в аналітиці.**

Без цього звіт може спиратися на дані, яких система не збирає.

☐ **Передавайте в роботу тільки відтворювані описи.**

Без відтворюваності неможливо ані оцінити виправлення, ані перевірити результат.

Після перевірки не потрібно змінювати все, що виглядає підозріло. Спочатку прибирають те, що відтворюється та заважає завершити дію; потім прояснюють умови; лише після цього оцінюють тексти, макет або кампанії повернення.

Порядок пріоритетів простий: відтворювана помилка, неясна умова, зайва дія. Помилка блокує сценарій напрямку. Неясна умова змушує покупця зупинитися, бо він не може оцінити доставку, оплату або підсумкову суму. Зайва дія створює тертя там, де її можна прибрати без втрати потрібної бізнес-логіки.

Критерієм приймання є не саме внесення зміни, а повторний успішний прохід сценарію. Якщо виправляли платіж, недостатньо побачити оновлене повідомлення: потрібно пройти шлях до повернення на сайт, перевірити зрозумілий стан для покупця та появу замовлення в адмінчастині.

01 Усуньте відтворювані помилки

Передайте розробнику точний запис із журналу. Після виправлення повторіть той самий сценарій за тих самих умов.

02 Покажіть умови до критичного рішення

Якщо доставка має обмеження або спосіб недоступний, поясніть це поруч із вибором. Не переносить ключову умову в окреме повідомлення, довідку або футер.

03 Виправте поведінку форми

Помилка має бути біля поля, де вона виникла, зрозуміло пояснювати наступну дію та не стирати коректно введені дані без пояснення.

04 Приберіть непотрібні дії

Перегляньте кожне поле й крок: для чого він потрібен бізнес-процесу. Не додавайте поля «про всяк випадок».

05 Прийміть результат за сценарієм

Зафіксуйте критерії: товар і параметри зберігаються, сума зрозуміла, доставка пояснена, форма зберігає дані, оплата повертає в ясний стан, замовлення відображається в адмінчастині.

ДЕ ВТРАЧАЮТЬ

Знижка, редизайн або промокод не виправляють збій оплати, втрату даних чи незрозумілу доставку. Вони лише маскують проблему й витрачають бюджет.

ЧЕКЛІСТ РОЗДІЛУ

- ☐ Розставте проблеми за порядком: помилка, умова, зайва дія.
Без пріоритету команда може витратити час на тексти, коли сценарій технічно не працює.
- ☐ Сформулюйте критерії приймання до початку змін.
Без них неможливо довести, що виправлення вирішило проблему, а не змінило її вигляд.
- ☐ Повторно пройдіть повний маршрут після виправлення.
Без повторної перевірки локальна зміна може зламати інший етап оформлення.

Не кожна проблема checkout вимагає нового сайту або зміни платформи. Перебудову обговорюють тоді, коли поточна логіка не дає виконати обов'язковий сценарій для покупця чи працівника, а не тому, що інтерфейс потребує одного уточнення.

Ознака системного обмеження — ручні обходи, які стали частиною щоденної роботи: працівники переносять дані між системами, вручну виправляють статуси або обходять правила доставки. Описуйте не назву платформи, а конкретну дію людини, момент її виникнення та результат, якого вона намагається досягти.

Перед оцінкою реалізації сформулюйте вимоги до кошика, правил товарів, доставки, оплати, повернення після оплати, статусів замовлення, адмінчастини та зовнішніх інтеграцій. Для кожної вимоги позначте, чи вона обов'язкова для роботи магазину, чи лише бажана.

01 Відокремте інтерфейсну проблему від системного обмеження

Перевірте, чи можна усунути бар'єр зміною тексту, поля, повідомлення або переходу. Якщо необхідна логіка недоступна в поточній системі, зафіксуйте це окремо.

02 Опишіть ручні обходи

Запишіть, хто саме виконує дію, після якого кроку, які дані переносить або виправляє та який результат має отримати.

03 Складіть карту обов'язкових вимог

Розподіліть вимоги за блоками: кошик, товарні правила, доставка, оплата, статуси, адмінчастина, інтеграції.

04 Позначте пріоритет кожної вимоги

Відокремте обов'язкове від бажаного. Це захищає бюджет від рішень, які виглядають масштабно, але не вирішують критичний сценарій.

05 Оцініть варіанти після формулювання вимог

Лише після цього визначайте, чи достатньо локальної зміни, чи потрібна глибша перебудова checkout.

ДЕ ВТРАЧАЮТЬ

Заміна всієї платформи без описаної вимоги часто переносить старий процес в нову систему разом із його проблемами.

ЧЕКЛІСТ РОЗДІЛУ

- ☐ Опишіть усі ручні обходи в обробці замовлень.
Без цього системні обмеження виглядатимуть як поодинокі помилки працівників.
- ☐ Сформууйте вимоги без назв платформ і готових рішень.
Без нейтральних вимог вибір технології визначить процес замість потреб магазину.
- ☐ Позначте обов'язкові та бажані зміни.
Без пріоритетів неможливо коректно оцінити обсяг робіт і бюджет.

Нагадування може повернути покупця до вже зібраного кошика, але не усуває бар'єр у доставці, формі чи оплаті. Його розглядають тільки тоді, коли ручна перевірка не виявила проблеми, що зупиняє сценарій.

У повідомленні дайте зрозумілий шлях назад до кошика та повторіть умови, які покупець уже бачив: склад замовлення, доставку й оплату. Не змінюйте правила між checkout і нагадуванням та не приховуйте обмеження за посиланням або дрібним текстом.

Не використовуйте вигаданий дефіцит, зміну ціни чи інші твердження без реальної підстави. Перед будь-яким контактом також перевірте правову підставу для використання даних у конкретному ринку та способі зв'язку, а також готовність команди обробити відповідь покупця.

01 Підтвердьте відсутність технічного бар'єра

До запуску сценарію повернення пройдіть checkout руками та переконайтеся, що доставка, форма, оплата і підтвердження працюють зрозуміло.

02 Перевірте підставу для контакту

З'ясуйте, чи має магазин право використати контактні дані для такого повідомлення в конкретному ринку та каналі зв'язку.

03 Поверніть покупця в збережений сценарій

Посилання має вести до кошика, де склад замовлення та зрозумілі умови не суперечать тому, що покупець бачив раніше.

04 Повторіть важливі умови

Нагадайте про склад замовлення, доставку й оплату без приховування обмежень та без тиску.

05 Переконайтеся, що відповідь є кому обробити

Не обіцяйте умови або допомогу, які магазин не може підтримати після звернення покупця.

ДЕ ВТРАЧАЮТЬ

Кампанія повернення, що веде в нестабільний checkout, повторно приводить покупця до тієї самої проблеми та підриває довіру.

ЧЕКЛІСТ РОЗДІЛУ

☐ Запускайте нагадування лише після ручного проходження checkout.

Без цього повідомлення може маскувати проблему, а не повертати покупця.

- ☐ **Перевірте законність використання контактних даних для обраного каналу.**
Без цього магазин ризикує використати чужий сценарій без належної підстави.
- ☐ **Переконайтеся, що умови в повідомленні збігаються з умовами в кошику.**
Без узгодженості покупець бачить суперечність і втрачає довіру.

Що вважати кинутим кошиком?

Для перевірки це товар у кошику або почате оформлення без підтвердженого замовлення в бізнес-процесі. Важливо окремо позначати, на якому саме етапі зупинився сценарій.

Чи є скасоване замовлення кинутим кошиком?

Ні. Замовлення вже створилося, тому потрібно окремо перевіряти причину скасування та подальший процес його обробки.

З чого почати аудит checkout?

Пройдіть шлях покупця руками: картка товару, кошик, доставка, форма, оплата та підтвердження. На кожному кроці записуйте дію, очікуваний і фактичний результат.

Коли передавати проблему розробнику?

Коли її можна описати та повторити: за якої дії й умови вона виникає, що мало статися і що сталося фактично. Загальне формулювання про незручність checkout не дає задачі для виправлення.

Чи варто одразу давати знижку тим, хто залишив кошик?

Ні. Спершу виключіть технічну помилку, неясну умову, втрату введених даних або недоступну оплату. Знижка не виправляє бар'єр, який не дає завершити замовлення.

ЩО ДАЛІ

Обговорити з Apricode



apri-code.com/kontakty/