Engineer

Локалізація

пристосування програми до мови й норм конкретного ринку

Що це таке

Локалізація — це робота з пристосування продукту до одного конкретного ринку: до його мови, а разом із нею до норм, яких читачі очікують від дат, чисел, цін, імен, адрес і порядку в списку. W3C визначає її як адаптацію продукту, застосунку чи вмісту документа до мовних, культурних та інших вимог певного цільового ринку, а сам такий ринок називає локаллю. Англійською слово пишуть localization або localisation і скорочують до l10n, бо між першою l та останньою n стоїть 10 літер.

W3C додає, що локалізацію часто вважають лише синонімом перекладу інтерфейсу й документації, хоча насправді це часто значно складніше завдання. Перелік того, що може потребувати адаптації, охоплює формати чисел, дат і часу; валюту; розкладку клавіатури та її використання; сортування; символи, піктограми й кольори; текст і графіку, які в певній культурі можуть хибно витлумачити або сприйняти як образливі; юридичні вимоги; формати адрес, поштових індексів і телефонних номерів та порядок імен; а також вимоги до читабельності кожної писемності. Там, де відрізняється спосіб вести справи, а прикладом W3C наводить бухгалтерію, адаптація може вимагати переосмислити логіку, візуальний дизайн або подання продукту.

Локалізація спирається на інтернаціоналізацію і йде після неї. Інтернаціоналізацію роблять один раз, у коді, щоб жодна особливість окремого ринку не була в ньому зашита. Локалізацію роблять для кожної локалі, у даних, які читає код: це перекладені повідомлення й норми, за якими збережена дата чи сума перетворюється на текст, якого очікує читач. Добре інтернаціоналізований продукт може отримати нову локаль без змін у коді; погано інтернаціоналізований доводиться змінювати заново для кожного ринку, на який він виходить.

Більшість цих норм команді не доводиться записувати самій. Проєкт CLDR Консорціуму Unicode збирає їх для понад 100 мов, а перевіряють їх носії мов і мовознавці. Власний приклад проєкту — список файлів у телефоні: формат дат, чисел, одиниць розміру файлів і алфавітний порядок залежать від мови і зазвичай надходять із CLDR. Проєкт стверджує, що його дані вбудовано в усі сучасні операційні системи й браузери, а також у такі мови програмування, як Java, C#, Swift і JavaScript. Отже, більшість форматів продукт отримує від платформи, якщо код просить їх у платформи, а не задає сам.

Застосуйте поняття до рішення

Застосуйте «Що таке локалізація?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Програма, що природно читається на кожному ринку, де ви продаєте, — з правильними датами, валютами й сортуванням імен.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Створила англо-данську систему i18n — 12,027 повідомлень на локаль — англійською та данською

Створила англо-данську систему i18n Зв'язатися з нами →

Чому це важливо

Перекладений, але не локалізований продукт здається чужим. Усі написи перекладено українською, але дата досі починається з місяця, ціна досі стоїть зі знаком долара, а список міст, відсортований за кодами символів, ставить Ізмаїл перед Вінницею. Читачі помічають кожну таку деталь одразу, і кожна з них підказує, що продукт робили для когось іншого.

Деякі розбіжності змінюють те, що читач розуміє. Рядок 10/11/2026 для американського читача означає 11 жовтня, а для британського 10 листопада; обидві дати коректні, тож нічого не ламається, і один із двох читачів діє не того дня. Рядок 1,234 англійською означає тисячу двісті тридцять чотири, а українською одну цілу двісті тридцять чотири тисячних, тобто різниця в тисячу разів у кількості чи ціні. Помилку в перекладі видно одразу. Помилкова норма виглядає правильною, і її хибно прочитують.

Перелік W3C виходить і за межі екрана. Юридичні вимоги на різних ринках різні, як і способи вести бізнес, тож локаль може вимагати інших умов, інших реквізитів у рахунку чи іншого порядку оформлення покупки. Це рішення для бізнесу, і їм місце в плані локалі ще до того, як перекладуть перше повідомлення.

Витрати рахуються на кожну локаль і ніколи не закінчуються. Кожне нове повідомлення треба перекласти всіма мовами продукту, кожен реліз треба перечитати в кожній локалі перед виходом, і кожен ринок приносить власні норми, які треба перевірити. Команда, що додає мову, додає цю роботу до кожного наступного релізу, тому рішення вийти на ринок водночас є рішенням про процес релізів.

Як це виглядає на практиці

Більшу частину роботи виконує платформа, якщо код передає їй значення й локаль. У JavaScript об'єкти форматування з `Intl` приймають і значення, і локаль та повертають текст:

```
const when = new Date("2026-10-27T12:00:00Z");
const money = { style: "currency", currency: "DKK" };
for (const locale of ["en-US", "en-GB", "da", "nb", "sv", "uk"]) {
  const date = new Intl.DateTimeFormat(locale, { dateStyle: "short" });
  const price = new Intl.NumberFormat(locale, money);
  console.log(locale, date.format(when), price.format(800));
}
```

У Node.js 25, що містить дані CLDR 47, код виводить одну дату й одну ціну в 800 данських крон шістьма способами:

```
en-US 10/27/26 DKK 800.00
en-GB 27/10/2026 DKK 800.00
da 27.10.2026 800,00 kr.
nb 27.10.2026 800,00 DKK
sv 2026-10-27 800,00 Dkr
uk 27.10.26 800,00 DKK
```

Нічого з цього код не обирав. Данський читач бачить kr., шведський Dkr, бо kr у Швеції означає шведську крону, а норвезький та український читачі бачать код валюти. Порядок дня й місяця, роздільники, десяткова кома й місце валюти беруться з даних локалі. У справжньому виводі між сумою та валютою стоїть нерозривний пробіл, U+00A0, який тримає їх на одному рядку.

Сортування працює так само. Шість данських назв місцевостей, відсортованих для трьох локалей:

```
const towns = ["Aabenraa", "Ærø", "Åkirkeby", "Odense", "Østerbro", "Vejle"];
for (const locale of ["en", "da", "sv"]) {
  const order = new Intl.Collator(locale).compare;
  console.log(locale, towns.toSorted(order).join(", "));
}
```

```
en Aabenraa, Ærø, Åkirkeby, Odense, Østerbro, Vejle
da Odense, Vejle, Ærø, Østerbro, Aabenraa, Åkirkeby
sv Aabenraa, Odense, Vejle, Åkirkeby, Ærø, Østerbro
```

Англійська трактує æ, ø і å як різновиди a та o. Данська ставить Æ, Ø і Å після Z і сортує старе написання Aa як Å, тож Aabenraa переміщується з першого місця на передостаннє. Шведська ставить власні Å, Ä і Ö після Z, сортує Æ разом з Ä, а Ø разом з Ö, але Aa вважає двома a, тож і для шведського читача Aabenraa стоїть першим. Кожен порядок правильний для своїх читачів, а данський читач, який шукатиме Aabenraa ближче до кінця списку, відсортованого для англійської, має всі підстави вирішити, що міста там немає.

Сайт на Hugo має ті самі інструменти у своїх шаблонах. Документація Hugo пропускає одну суму через `lang.FormatCurrency`:

```
{{ 512.5032 | lang.FormatCurrency 2 "USD" }}
```

Англійська сторінка показує її як $512.50, французька як 512,50 $US, німецька як 512,50 $, а `lang.FormatNumber` і `lang.FormatPercent` роблять те саме для звичайних чисел і відсотків.

Локалізація входить до однієї з послуг цього сайту. Послуга Інтернаціоналізація та локалізація описує мету як програму, що природно читається на кожному ринку, де продає клієнт, з правильними датами, валютами й сортуванням імен. Серед її результатів є міжмовна перевірка контенту, яка падає, коли переклад губить число, вказане англійською, або коли мова вживає не притаманну їй нотацію, і яка знайшла два данські описи, у яких було втрачено показник з англійського тексту, та шістнадцять українських фрагментів, що цитували на англійський лад.

Де виникають проблеми

Локалізацію бюджетують як переклад. План виходу на новий ринок охоплює файли повідомлень і нічого більше, тож продукт запускають новою мовою з датами, цінами, адресами й порядком сортування старого ринку, а прогалини знаходять клієнти. Плануйте кожну локаль за повним переліком W3C, вирішіть, які пункти стосуються саме цього ринку, і дайте носієві мови переглянути готовий продукт, а не файли повідомлень.

Мову плутають із локаллю. `en` не каже, чи 27 жовтня записується як `10/27/26`, чи як `27/10/2026`, і результат вище показує, що дві англійські локалі розходяться і в порядку, і в записі року. Продукт, який зберігає лише мову, мусить угадувати решту, і норми одного ринку він нав'язує всім іншим. Зберігайте для кожного користувача повний тег, як-от `en-GB` чи `uk-UA`, беріть його з явного налаштування, де воно є, і передавайте в кожен виклик форматування.

Валюту вгадують за локаллю. Специфікація CLDR каже про це прямо: суму ніколи не можна передавати між системами без відомого коду валюти, інакше один користувач прочитає число 3,5 як 3,50 долара, а інший — як 3,50 євро. Вона додає, що код валюти суми не залежить від мови читача і що лише в погано спроєктованих програмах коду немає, тож програмі доводиться вгадувати валюту за локаллю. У Скандинавії така здогадка помиляється непомітно, бо данці, норвежці й шведи однаково скорочують назву своєї крони до kr, тож ціна, збережена як 800, для одного читача означатиме 800 данських крон, а для наступного 800 норвезьких. Зберігайте кожну суму з її кодом ISO 4217 і форматуйте її для читача; український читач тоді побачить данську ціну як 800,00 DKK, що й видно з результату вище.

Списки сортують за кодовими точками. Стандартне сортування в JavaScript, `sort` без функції порівняння, порівнює кодові одиниці UTF-16, тож Є та І опиняються перед рештою української абетки, а Ґ після Я, бо їхні кодові точки лежать поза діапазоном від А до я. Відсортований так список Ялта, Ґанок, Гора, Єнот, Іван, Дуб стає Єнот, Іван, Гора, Дуб, Ялта, Ґанок; з `new Intl.Collator("uk").compare` виходить Гора, Ґанок, Дуб, Єнот, Іван, Ялта, тобто порядок української абетки. Сортуйте текст для показу за локаллю читача, через `Intl.Collator` у коді або через правила сортування в базі даних, а порядок кодових точок лишайте для ключів, яких не бачить жоден читач.

Тести порівнюють із відформатованим текстом, набраним уручну. Тест, який очікує `800,00 ₴` зі звичайним пробілом, падає на результаті форматування, бо там стоїть U+00A0, і той самий нерозривний пробіл розділяє групи тисяч у норвезькій, шведській та українській. До того ж дані локалей переглядають від одного випуску CLDR до наступного, тож рядок, який збігався до оновлення платформи, після нього може вже не збігатися. Будуйте очікуваний текст тим самим засобом форматування і з тією самою локаллю, що й код, або перевіряйте збережене значення й передану локаль, а не готовий рядок.

Макет розраховано на англійську. Кнопка, у яку вміщається Save, має вмістити й Зберегти, удвічі довше слово, а меню, де поміщається Settings, має вмістити й Налаштування. Фіксована ширина обрізає довше слово або переносить його на другий рядок, і часто першим це помічає читач саме на тому ринку. Дайте елементам рости разом із текстом, уникайте фіксованої ширини навколо перекладених слів і переглядайте кожну локаль на найвужчому екрані, який підтримує продукт.

Що почитати далі

- Localization vs. Internationalization — робоче визначення локалізації від W3C, перелік того, що в локалі може потребувати адаптації, і зв’язок локалізації з інтернаціоналізацією.
- Hugo: Multilingual mode — локалізовані дати, валюти, числа й відсотки в шаблонах Hugo, а також таблиці перекладів і URL для кожної мови.
- UTS #35, Part 3: Numbers — формати чисел і валют у CLDR, зокрема правило, що сума завжди супроводжується кодом валюти.
- Unicode CLDR Project — проєкт, що збирає дані локалей для понад 100 мов і постачає їх операційним системам, браузерам і мовам програмування.

Словник

Обирайте за фактичними умовами

Опишіть робочий процес, наявні інтерфейси та користувачів, яким потрібна зміна. Ми можемо визначити перевірюваний результат і спосіб передачі, який дасть вашій команді змогу підтримувати його.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке локалізація?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/localization/