

ІНСТРУМЕНТИ ТА ЕТАПИ ЛОКАЛІЗАЦІЇ ІНТЕРФЕЙСІВ ДЛЯ ПІДВИЩЕННЯ ДОСТУПНОСТІ ТА ЗРУЧНОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

У статті досліджуються інструменти, які допомагають створювати локалізацію інтерфейсів системи, також, висвітлюється вплив цих інструментів на ефективне впровадження локалізації до системи. Таким чином, розкривається питання щодо доступності та зручності користування системою.

Ключові слова: локалізація, інструменти для локалізації, інтерфейс системи.

The article examines the tools that help create localized system interfaces and highlights the impact of these tools on the practical implementation of localization in the system. Thus, the issue of accessibility and ease of use of the system is revealed.

Key words: localization, localization tools, system interface.

Доступність і зручність інтерфейсу системи (англ. «usability») є важливими критеріями для масового її використання. Локалізація програмного забезпечення – це важлива частина дизайну, що безпосередньо впливає на користувацький досвід (UX), який може бути як позитивним, так і негативним. Саме тому слід розглянути інструменти та стратегії, які підвищують доступність та зручність системи через локалізацію інтерфейсу. Локалізація – це не лише переклад слів, а й адаптація програмного забезпечення, щоб взаємодіяти з користувачами з усіх куточків світу. Налаштовуючи інтерфейси, вміст і функції відповідно до унікальних уподобань і культурних відмінностей різноманітних аудиторій, локалізація перетворює програмне забезпечення на універсальний інструмент, який пристосований до мов різних груп користувачів.

Мета статті – розглянути особливості локалізації інтерфейсу за допомогою інструментів та стратегій.

Спочатку, слід розкрити поняття локалізації, насамперед, це адаптація програмного забезпечення (вебсайтів, додатків) до мовних та культурних особливостей певної аудиторії. Інакше кажучи, це коли замість іноземного тексту ми бачимо інтерфейс рідною мовою, час та інші вимірювання в потрібному форматі, кольори та символи відповідно до культурних особливостей. Це дозволяє розширити аудиторію конкретного програмного забезпечення, залучити більше користувачів.

Одним з важливих елементів локалізації є інтерфейс, саме тому, слід звертати увагу на такі аспекти при створенні інтерфейсу та подальшій його локалізації: адаптувати елементи дизайну до місцевих уподобань, застосовувати відповідні культурі кольори та зображення, і звичайно,

використати інтуїтивно зрозумілу навігацію для користувачів усіма мовами [3].

Спочатку слід перейти до самих етапів локалізації інтерфейсів. Серед них ми виділили наступні:

- Вибір мов для локалізації. Нами було обрано мови, які, ми вважаємо, є найбільш розповсюдженими серед майбутніх користувачів, а саме: українська, англійська, німецька, іспанська.
- Співпраця з фахівцями у сфері локалізації. Насамперед, це фахівці, які будуть перекладати сайт, редагувати та тестувати.
- Розробка локалізації. Створення словників, внесення тексту перекладу у код програми, налаштування локалізації сайту.
- Тестування локалізаційного інтерфейсу. Одним з важливих елементів є тестування продукту, виявлення та виправлення помилок, щоб забезпечити якість локалізованого інтерфейсу.
- Підтримка та оновлення. Оскільки, сайт може змінюватися або розширюватися то важливим елементом є підтримка оновлень локалізації інтерфейсу.

Відповідно до вищенаведених етапів існує низка інструментів, які допомагають у локалізації інтерфейсу. Вони насамперед допомагають перекладати текст і керувати термінологією, а також перевіряти наявність помилок, контролювати версії продукту та обмінюватися файлами.

До прикладу, фахівці з перекладу можуть користуватися такими інструментами як, комп'ютерний переклад (CAT) – цей інструмент CAT допомагає у процесі локалізації, використовуючи збережені дані для інформування про те, як перекладається текст. Наступним прикладом можуть бути інструменти для тестування локалізації, такі як: Bugtracking, Global App Testing для функціонального тестування (інженерами), Global App Testing (Phrase) для команд, продуктів і проектів (раніше відома як Memsource) – цей інструмент пропонує інтелектуальну автоматизацію та функції штучного інтелекту, які разом із інструментами розробника, API та інструментами інтеграції спрощують масштабування проєкту [4].

Що ж стосується безпосередньо інструментів для розробки самої локалізації, то можна розглянути наступні фреймворки та бібліотеки: React, Angular і Vue.

React або «React JS» – це застосунок, що являє собою бібліотеку для значного спрощення побудови й маніпулювання DOM елементами. Засновником є компанія Facebook. Спочатку React розробляли й використовували лише для внутрішніх потреб, потім її зробили публічною, надавши до неї доступ для програмістів. До плюсів React можна віднести: зрозумілий синтаксис, функціональний підхід до написання компонентів, велика кількість різноманітної інформації про помилки і основні проблеми та багато інших.

Angular, засновником якого є компанії Google, є повноцінним фреймворком, який керується принципами MVVM побудови застосунків, і

має такі можливості, як: CLI для генерування структури файлів різних Angular-конструкцій, має різноманітні бібліотеки, також засоби лінтування, і звичайно Karma + Jasmine для Unit і Endto End тестування.

Vue можна назвати сумішню Angular і React. З однієї сторони –це бібліотека для роботи з DOM, з іншої – це фреймворк, який має свій офіційний Vue CLI й певна екосистема бібліотек навколо нього. Він має можливість писати додатки, як з білд бібліотеками, так і без них, також має високу швидкість збірки й роботи застосунку в цілому [1].

Ще одним важливим інструментом для локалізації інтерфейсу є i18next – у свою чергу, i18next – це не просто бібліотека, яка виконує переклади тексту, i18next – це фреймворк, що дозволяє розгорнути повноцінне рішення для локалізації застосунків. І саме i18next можна використовувати із будь-яким фреймворком, такими як: React, Angular, Vue тощо. Отже, i18next є платформою для локалізації інтерфейсу [2].

Підсумовуючи вище сказане, ми робимо наступні висновки, безпосередньо локалізація програмного забезпечення впливає на доступність використання системи різними користувачами, оскільки важливим є той фактор, щоб локалізація несла за собою не лише переклад тексту, а саме адаптацію його до певного регіону чи мови. Що стосується етапів локалізації, то нами були виділені наступні: вибір мов, переклад та адаптація, розробка, тестування, підтримка. Загалом, було розглянуто низку інструментів доступних для локалізації інтерфейсу, але найважливішими для розробника являються саме фреймворки та бібліотеки, такі як: React, Angular, Vue, i18next. Вони допомагають розробнику реалізувати локалізацію інтерфейсу та забезпечити його доступність, зручність, продуктивність та підвищити зрозумілість для користувача.

ЛІТЕРАТУРА:

1. Олександр Гуцуляк Порівнюємо React, Angular і Vue — найпопулярніші бібліотеки й фреймворки у 2022 році, 14.09.2022. URL:<https://dou.ua/forums/topic/39933/> (дата звернення: 05.09.2024)
2. Локалізація React (Mobx) застосунку за допомогою i18next, 07.09.2018 URL:<https://devzone.org.ua/post/lokalizatsiia-react-mobx-zastosunku-za-dopomohoiu-i18next>(дата звернення: 05.09.2024)
3. The importance of localization in software development: 20 March 2024. URL:<https://moldstud.com/articles/p-the-importance-of-localization-in-software-development> (дата звернення: 05.09.2024)
4. Global App Testing The best localization tools for your software URL:<https://www.globalapptesting.com/localization-testing/localization-tools> (дата звернення: 05.09.2024)

Науковий керівник докторка педагогічних наук, професорка Валько Н.В.