

ІНТЕРФЕЙС ЯК ОБ'ЄКТ АВТОРСЬКОГО ПРАВА

INTERFACE AS A COPYRIGHT OBJECT

Цуркану С.І., к.ю.н., доцент,
юрист

Стаття присвячена визначенню інтерфейсу як об'єкту авторського права з виокремленням проблем, що виникають на цьому шляху. Автор зазначає, що комп'ютерна програма може бути поділена на самостійні об'єкти права інтелектуальної власності. Це такі об'єкти: код програми, алгоритм побудови програми, графічний інтерфейс користувача. Розглядається оригінальність інтерфейсу через призму концепцій "look and feel" та "fair use". Згідно з концепцією "look and feel" були виділені дві важливі позиції з точки зору охорони авторським правом комп'ютерних програм: охороні підлягають не лише літеральні складники комп'ютерної програми (так звані код програми (бекенд)), а й нелітеральні компоненти (структура, послідовність та організація (алгоритм)) програми; охороноздатними вважаються ті об'єкти авторського права, які за загальним сприйняттям розумної людини не є аналогічними вже існуючим творам та мають свою оригінальність. У подальшому надання авторської охорони через критерій загального сприйняття (концепція "look and feel") було змінено на критерій порушення структури оригінальності (справа "Altai та Apple Computer, Inc. v. Microsoft Corporation"). Такий підхід був зумовлений можливістю генерування однакового зображення (інтерфейсу) двома різними комп'ютерними програмами. Зрештою, американська судова практика свідчить про те, що інтерфейс не підлягає авторській охороні на підставі доктрини "fair use" з точки зору необхідності стандартизації комп'ютерних технологій. Судовий прецедент у справі "Apple Computer, Inc. v. Microsoft Corporation" започаткував охорону графічного інтерфейсу користувача шляхом реєстрації патентів. Автор дійшов висновку, що інтерфейс може охоронятися як нормами авторського права, так і нормами патентного права. З метою мінімізації ризиків невизнання інтерфейсу об'єктом авторського права з огляду на відсутність такої ознаки, як оригінальність твору, та для інтенсифікації процесів стандартизації інтерфейсів рекомендовано реєструвати патенти на інтерфейс. У подальших дослідженнях будуть розглянуті особливості інтерфейсу як об'єкта патентного права.

Ключові слова: Україна, США, авторське право, комп'ютерна програма, інтерфейс, патент.

The article is devoted to the definition of the interface as an object of copyright, highlighting the problems that arise in this way. The author notes that a computer program can be divided into independent objects of intellectual property rights, in particular, the program code; program construction algorithm; graphical user interface.

The originality of the interface is considered through the prism of the concepts of "look and feel" and "fair use". According to the concept of "look and feel", two important positions were identified in terms of copyright protection of computer programs: not only the literal components of a computer program (so-called program code), but also non-literary components (structure, sequence and organization (algorithm)) of the program, protected objects are those copyrighted objects that, according to the general perception of a reasonable person, are not similar to existing works, and have their own originality. Subsequently, the provision of copyright protection through the criterion of general perception (the concept of "look and feel") was changed to the criterion of violation of the structure of originality ("Altai and Apple Computer, Inc. v. Microsoft Corporation"). This approach was due to the possibility of generating the same image (interface) by two different computer programs. Finally, US case law shows that the interface is not subject to copyright protection based on the doctrine of "fair use" in terms of the need to standardize computer technology. The case law of "Apple Computer, Inc. v. Microsoft Corporation" started protecting the graphical user interface by registering patents. The author concludes that the interface can be protected by both copyright and patent law. In order to minimize the risks of non-recognition of the interface as an object of copyright, given the lack of such a feature as the originality of the work and the intensification of the processes of standardization of interfaces, it is recommended to register patents for the interface. Further research will consider the features of the interface as an object of patent law.

Key words: Ukraine, USA, copyright, computer program, interface, patent.

Вступ. В Україні активно розвивається такий сегмент внутрішнього ринку, як ІТ-бізнес. Розробники, працюючи на національні та закордонні ІТ-компанії, створюють ІТ-продукт, що неминуче призводить до актуалізації проблеми охорони програмного забезпечення. На міжнародному рівні з прийняттям Бернської конвенції про охорону літературних та художніх творів увага нормотворців була зосереджена на виявленні правової природи та обсягу охорони лише комп'ютерних програм як об'єкту права інтелектуальної власності. Інтенсифікація світових процесів розвитку програмного забезпечення зумовила появу нових об'єктів права інтелектуальної власності в результаті програмування, які на сьогодні не мають чітко розробленого механізму правового регулювання їх охорони.

Виклад основного матеріалу дослідження. Не вдаючись до аналізу поняття «прилад» (розглядаємо його зміст у широкому розумінні), значимого, що функціонування комп'ютера або іншого девайсу здійснюється за допомогою двох таких складових частин: апаратного забезпечення (системний блок, процесор, блок живлення, диск, плата, екран тощо); програмного забезпечення (операційна система, транслятори, редактори, графічний інтерфейс користувача) [1, с. 204]. Взаємодія між даними компонентами забезпечує керування та роботу комп'ютера.

Умовно програмне забезпечення можна поділити на системне та прикладне. Системне програмне забезпечення створює функціональне оточення для роботи інших

прикладних додатків. Зокрема, жоден фоторедактор не може відредагувати фото без операційної системи.

І системне, і прикладне програмне забезпечення з точки зору правової регламентації є комп'ютерною програмою.

Згідно зі ст. 1 Закону України «Про авторське право і суміжні права» під комп'ютерною програмою розуміється набір інструкцій у вигляді слів, цифр, кодів, схем, символів чи у будь-якому іншому вигляді, виражених у формі, придатній для читання комп'ютером, які приводять його у дію для досягнення певної мети або результату (це поняття охоплює операційну систему і прикладну програму, виражені у вихідному або об'єктному коді). Згідно зі ст. 18 цього Закону комп'ютерні програми охороняються як літературні твори. Така охорона поширюється на комп'ютерні програми незалежно від способу чи форми їх вираження. Аналогічний підхід закріплений і в міжнародних документах. Так, відповідно до статті 4 Бернської конвенції про охорону літературних та художніх творів, статті 10 Угоди про торговельні аспекти прав інтелектуальної власності комп'ютерні програми у вихідному або об'єктному коді захищаються як літературні твори. Законодавче визначення комп'ютерної програми не дає змоги усвідомити її технологічні особливості і, як наслідок, обсяг необхідної правової охорони.

У правовій доктрині були висловлені думки щодо необхідності виділення окремих складників комп'ютерної

програми. Посилаючись на статтю 1 Типових положень з захисту комп'ютерного програмного забезпечення та проект Угоди з захисту комп'ютерного програмного забезпечення, Л.Є. Стрельцов визначив комп'ютерну програму як набір інструкцій, який являє собою алгоритм або сукупність алгоритмів [2, с. 168]. Всебічне визначення змісту комп'ютерної програми надає С.Р. Водорезова, яка пропонує виділяти такі її складові частини, як алгоритм (набір структурних блоків, до яких не входять загальновідомі математичні формули, методи, розрахунки, які реалізують функціональність програмного забезпечення), інтерфейс (дизайн, зовнішній вигляд програми), код програми (варіант виконання, текст програми) [3, с. 74].

На нашу думку, компоненти комп'ютерної програми повинні виокремлюватися інженерами (розробниками), які займаються програмуванням і розуміють процеси, що відбуваються. Фахівці у галузі права мусять проаналізувати виокремлені компоненти та надати їм належний режим охорони.

В цілому процес функціонування програми умовно можна поділити на дві частини – бекенд та фронтенд. Бекенд (від англ. “back-end” – «зворотна сторона») – програмний код, що відповідає за роботу із сервером (базою даних), а також з даними для подальшого їх включення у базу даних. Фронтенд (від англ. “front-end” – «лицьова сторона») – публічна частина програми, з якою безпосередньо контактує користувач, та функціонал, який зазвичай обігрується на стороні користувача (у браузері) [4, с. 17].

Бекенд формується програмістом шляхом написання вихідного коду у вигляді символів, цифр, слів тощо, який за допомогою компілятора мови програмування генерується в об'єктний код (має вигляд різного поєднання нулів та одиниць), що зчитується сервером (комп'ютером) [5, с. 337]. Вихідний код комп'ютерної програми може містити частину коду, що знаходиться у відкритому доступі (тобто код заздалегідь був написаний іншими програмістами і використовується під час написання нової програми). Ідеться про ту частину коду, яка вперше написана розробником для створення програми. На даному етапі розгляду технічної частини актуалізується питання правомірності використання коду, що знаходиться в Open Source, яке буде досліджене нижче.

Будь-яка програма, як і прилад у цілому, створюється з метою задоволення потреб та забезпечення продуктивності праці користувача. Найчастіше користувачі не є спеціалістами в галузі інформатики. Користувач автоматизованої обчислювальної системи має право очікувати не тільки точних результатів обробки, а й зручності у використанні системи. Очам людини необхідно, щоб образи мали визначений розмір, рівень яскравості, контрастності і розміщувались на зручній відстані. Деякі кольори сприймаються краще з інші [6, с. 5]. Зручною та привабливою для сприйняття комп'ютерну програму роблять розробники фронтенду, що інакше іменується як графічний інтерфейс користувача.

Отже, в цілому комп'ютерна програма охороняється правовими засобами авторського права. Водночас багатокомпонентна побудова комп'ютерної програми зумовлює необхідність виокремлення таких самостійних об'єктів права інтелектуальної власності, як код програми, алгоритм побудови програми, графічний інтерфейс користувача. Виділені об'єкти потребують дослідження їх правової природи та встановлення особливостей правового регулювання.

Персональний комп'ютер, що має монітор з графічним інтерфейсом та системою «вікон» (персональний комп'ютер марки “Alto”), розробив американський науковець Боб Тейлор в лабораторії дослідницького центру «Ксерокс» в Пало-Альто. У ПК “Alto” використовувалась мова програмування Smalltalk, за допомогою мишки із

меню можна було обрати відповідну операцію [7]. Проте компанія «Ксерокс» не стала початківцем у випуску персональних комп'ютерів у промислових масштабах.

Досі залишається загадкою, яким чином Стівен Джобс отримав технологію з розробки графічного інтерфейсу, навколо якої витає безліч міфів та фактів. За словами Стівена Леві, в 1979 році «Ксерокс» вклав гроші у компанію “Apple”, усвідомивши відсутність власних можливостей для широкомасштабної розробки доступних у фінансовому плані для користувачів персональних комп'ютерів. На той момент інженери “Apple” безрезультатно займалися розробкою графічного інтерфейсу. Коли «Ксерокс» дозволив розробникам «яблука» побачити свої дослідження у цій сфері, Стів Джобс вигукнув: «Сидите на золотій міні. Не розумію, як ви не змогли отримати з цього зиск!» [8, р. 90].

Графічний інтерфейс користувача (GUI) – це інтерфейс типу «людина – комп'ютер» (тобто спосіб взаємодії людей з комп'ютерами), який використовує вікна, піктограми та меню і з яким можна взаємодіяти (маніпулювати) мишею і часто в обмежених межах клавіатурою [9]. За допомогою прямих маніпуляцій графічними елементами, які складають графічний інтерфейс, таких як рух курсором, клікання ним по іконці, що знаходиться на екрані, набір букв на клавіатурі та їх поява у вікні програми “Word”, використання джойстиків та ін., користувач задає необхідну задачу комп'ютерній програмі, яка кінцевий результат виконання відображає в інтерфейсі користувача.

Інженери, які розробляють графічний інтерфейс користувача, зіштовхуються з проблемами забезпечення зручності, зрозумілості та зовнішньої привабливості інтерфейсу [10, р. 104]. Наприклад, якщо взяти програму “Word”, то набір функціональних задач розміщений зверху, а не посередині вікна з полем набору, за рахунок чого досягається зручність. Для користувача цієї програми є зрозумілими функції «Файл», «Головна», «Вставка» та ін. Вони є стандартними для цієї програми. Зрештою, кожна функціональна панель має своє зовнішнє оздоблення. Якщо взяти варіативність шрифтів, то кожен запропонований шрифт демонструється у його назві.

Значені загальні вимоги до розробки інтерфейсу створюють підґрунтя для інтуїтивного використання програми, оскільки розробники в процесі програмування повинні дотримуватися затверджених стандартів. Такий підхід є виправданим, оскільки, отримавши нову модель приладу, користувач не буде змушений вивчати нові назви старих функцій.

Показовою в цьому контексті є робота зі створення додатків iOS (iOS є операційною системою “Apple”). Зокрема, “Apple” випустив навіть довідник з iOS-дизайну, в якому прописані стандарти інтерфейсу додатків на мобільних телефонах компанії “Apple”. Зокрема, прикладом затверджених стандартів є чотири типи іконок: іконка програми, іконка для магазину додатків “AppStore”, іконка для “Spotlight”, іконка для налаштувань. Системним шрифтом є San Francisco Font. Цей шрифт став основним ще від iOS 9. Від iOS 7 яскраві кольори стали обличчям iOS (помаранчевий, червоний, жовтий, синій, блакитний, зелений). На iOS-пристроях і в “AppStore” ми бачимо заокруглені кути іконок (кути з радіусом). Ні в якому разі розробнику не потрібно самому створювати ці заокруглення і зберігати їх у фінальному файлі. “Apple” обріже іконку з необхідним радіусом самостійно.

Через таку кількість та детальність прописаних стандартів виникає питання щодо охороноздатності засобами авторського права інтерфейсу з точки зору наявності такої ознаки твору, як творчий характер, адже однакові настанови зі створення твору не можуть мати результатом оригінальність. Наприклад, якщо певній групі людей поставити задачу намалювати квітку із кіл, то отримані малюнки будуть приблизно однаковими.

В. Веінке зазначив, що робота може називатися твором у тому разі, коли вона є результатом творчої, духовної роботи та містить відбиток індивідуальності автора [11, с. 41].

Незалежно від того, як автори називають підходи до визначення творчості (зовнішній чи внутрішній [11, с. 43]; суб'єктивний чи об'єктивний [12, с. 41]), суть їх має таке вираження: творчість визначається або через самостійну діяльність автора, або через результат такої діяльності, що характеризується оригінальністю щодо наявних творів.

Крім того, «Український тлумачний словник» слово «оригінальний» пояснює так: «1) який не є копією; справжній, автентичний. Створений самостійно, без наслідування зразків; 2) який привертає до себе увагу своєю незвичайністю, своєрідністю. Не схожий на інших, самобутній» [13].

Визначення оригінальності через привернення уваги до незвичайності твору та через його загальне сприйняття цілком відповідає доктрині “look and feel”, яка існувала до 1992 року як відправна точка визначення оригінальності в авторському праві в США. Зазначена доктрина була вперше сформульована у справі “Whelan v. Jaslow” (1986 р.), в якій були виділені дві важливі позиції з точки зору охорони авторським правом комп'ютерних програм: охороні підлягають не лише літеральні складові частини комп'ютерної програми (так звані код програми (бекенд)), а й нелітеральні компоненти (структура, послідовність та організація (алгоритм)) програми; охороноздатними вважаються ті об'єкти авторського права, які за загальним сприйняттям розумної людини не є аналогічними вже існуючим творам та мають свою оригінальність [14].

Згодом надання авторської охорони через критерій загального сприйняття (концепція “look and feel”) було змінено на надання такої охорони за критерієм порушення структури оригінальності (справи “Altai та Apple Computer, Inc. v. Microsoft Corporation”). Такий підхід був зумовлений можливістю генерування однакового зображення (інтерфейсу) двома різними комп'ютерними програмами. Розглянемо більш детально загальновідоме судове рішення за справою “Apple Computer, Inc. v. Microsoft Corporation”.

Компанія “Apple” надала ліцензію на використання розробленого її інженерами інтерфейсу (спливаюче вікно та керування програмою за допомогою миші) комп'ютерів марок “Liza” та “Macintosh” корпорації “Microsoft”, на підставі якої була випущена програма “Windows 1”. Подальші версії “Windows 2.03” та “Windows 3” викликали претензії з боку “Apple”, оскільки “Microsoft”, на їх думку, перевищив межі наданих ліцензією прав [15]. Під час розгляду справи суд виходив не із концепції “look and feel”, а здійснив порівняння окремих елементів. Як наслідок, із заявлених компанією “Apple” скопійованих 189 елементів судом було відхилено 179 претензій. 10 претензій, що залишилися, на думку Каліфорнійського суду, не підлягають охороні на підставі доктрини “fair use” з точки зору необхідності стандартизації комп'ютерних технологій [16, с. 236]. Таким чином, судовий прецедент у справі “Apple Computer, Inc. v. Microsoft Corporation”

започаткував охорону графічного інтерфейсу користувача шляхом реєстрації патентів.

Цікавою в контексті використання доктрини “fair use” є 9-річна судова справа “Oracle проти Google”, яка стала показовою в аспекті визначення оригінальності твору в умовах стандартизації комп'ютерних програм. Скорочено ця справа отримала назву «справа API».

API визначає функціональність, яку надає програма, і її компоненти взаємодіють один з одним за допомогою API. Усі операційні системи використовують API, за допомогою яких програмісти можуть створювати додатки для цієї операційної системи [17].

У 1996 році компанія “Sun Microsystem” під керівництвом Еріка Шмідта створила відому мову програмування Java. У 2007 році компанія “Google” офіційно випустила операційну систему Android, яка є аналогом Java і частково на ній побудована (було використано 37 API, які зареєстровані в семи патентах). У 2010 році компанія “Oracle” набула права на компанію “Sun Microsystem” та подала позов про порушення авторських прав на комп'ютерну програму.

У судах нижчих інстанцій компанія “Google” аргументувала свою позицію фактом функціональності використаної частини твору, яка є елементарною з точки зору програмування та запозиченою виключно з метою забезпечення функціональності та інтенсифікації розвитку комп'ютерних технологій. Попри дані аргументи, суд підтримав позивача, мотивуючи це тим, що розробники компанії “Google” могли створити цю частину коду [18].

У 2015 році юристи компанії “Google” подали апеляцію до Верховного Суду США, який передав справу на розгляд в Окружний суд Каліфорнії. Саме цей суд використав доктрину “fair use” у справі “Apple проти Microsoft”. Не відступаючи від наміченої тенденції, Каліфорнійський суд на підставі “fair use” задовольнив апеляцію компанії “Google”.

Відповідно до параграфа 107 Закону США «Про авторське право» факторами, що визначають добросовісність використання, необхідно вважати такі: 1) мету і характер використання (зокрема, комерційна або освітня мета); 2) правову природу самого твору; 3) кількість та значимість використання старого твору по відношенню до нового у цілому; 4) ефективність та цінність нового твору для подальшого розвитку.

Зважаючи на кількість використаного коду (3% від загальної кількості написаного) та його відкритість (код з open source), Каліфорнійський суд констатував добросовісність використання компанією “Google” коду мови програмування Java, чим спричинив подання чергової апеляції з боку компанії “Oracle”.

Висновки. Таким чином, інтерфейс може охоронятися як нормами авторського права, так і нормами патентного права. З метою мінімізації ризиків невизнання інтерфейсу об'єктом авторського права з огляду на відсутність такої ознаки, як оригінальність твору, та для інтенсифікації процесів стандартизації інтерфейсів рекомендовано реєструвати патенти на інтерфейс. У наступних дослідженнях будуть розглянуті особливості інтерфейсу як об'єкта патентного права.

ЛІТЕРАТУРА

1. Білоус-Сергеева С.О. Проблеми споживання програмного забезпечення в умовах глобалізації. *Теоретичні і практичні аспекти економіки та інтелектуальної власності*. 2016. № 14. С. 203–208.
2. Стрельцов Л.Є. Поняття та ознаки комп'ютерної програми у міжнародному праві. *Науковий вісник Міжнародного гуманітарного університету*. 2014. № 11. Вип. 2. С. 168–171.
3. Водорезова С.Р. Особливості правової охорони комп'ютерної програми як об'єкта інформаційних відносин. *Право та інновації*. 2014. № 3 (7). С. 69–76.
4. Фленов М.Е. Linux глазами хакера : 5-е издание, доп. и перераб. Санкт-Петербург, 2019. 416 с.
5. Шеховцов В.А. Операційні системи. Київ : Видавнична група ВВВ, 2005. 576 с.
6. Майданюк В.П., Петух А.М. Інтерфейс «користувач-комп'ютер» : навчальний посібник. Вінниця : ВДТУ, 1999. 66 с.
7. Рылев Ю. 6000 изобретений XX и XXI веков, изменившие мир. *Litres*. 5 сент. 2017 г. 4828 с. URL: https://books.google.com.ua/books?id=XR5IAAAQBAJ&hl=ru&source=gbs_navlinks_s.

8. Grace Lees-Maffei *Iconic Designs: 50 Stories about 50 Things*. *Bloomsbury Publishing*. 2020 г. 240 с. URL: https://books.google.com.ua/books?id=qsPuBQAAQBAJ&dq=kseroks+alto+steve+jobs&hl=ru&source=gbs_navlinks_s.
9. Зеров К.О. Правова охорона графічного інтерфейсу користувача комп'ютерної програми. *IT-право: проблеми і перспективи розвитку в Україні* : матеріали Другої міжнародної щорічної конференції. URL: <http://aphd.ua/publication-372/>.
10. Patent law for computer Scientists. *Steps to protect computer-implemented protection* / D. Closa, A. Gardiner, F. Giemsa, J. Machek. Springer-Verlag, Heidelberg, Germany, 2010. 191 p. URL: <https://books.google.com.ua/books?id=Aw0rTPiu9uIC&printsec=frontcover&hl=ru#v=onepage&q&f=false>.
11. Петренко І. Правова характеристика літературного твору як об'єкта авторського права. *Теорія і практика інтелектуальної власності*. 2014. № 6. С. 40–47.
12. Семенюта Б.Е. Графический пользовательский интерфейс программы для ЭВМ: проблемы правового регулирования. *Отрасли права* : аналитический портал. URL: <http://xn----7sbbaj7auwnffhk.xn--p1ai/article/19180>
13. Український тлумачний словник. URL: https://ukrainian_explanatory.academic.ru/.
14. Stanley Lai. Copyright Protection of Computer Software in the United Kingdom. *Hart Publishing*. 2000. 250 p. URL: https://books.google.com.ua/books?id=ZpR-CAFuYtgC&dq=doctrine+look+and+feel+copyright&hl=ru&source=gbs_navlinks_s.
15. Apple Computer, Inc. v. Microsoft Corp., 35 F.3d 1435, 1439 (9th Cir. 1994). URL: <https://casetext.com/case/apple-computer-inc-v-microsoft-corp>.
16. Жарова А.К. Защита интеллектуальной собственности : учебник / под общей редакцией А.А. Стрельцова. Москва : Издательство «Юрайт», 2019. 341 с. URL: <https://books.google.com.ua/books?id=1ZR9DwAAQBAJ&pg=PA236&lpg=PA236&dq=1994+%D1%81%D1%83%D0%B4%D0%BE%D0%B2%D0%B5+%D1%80%D1%96%D1%88%D0%B5%D0%BD%D0%BD%D1%8F+apple+v+microsoft&source=bl&ots=VxTBiW9ht5&sig=ACfU3U2JWLW6DaasOsS7RYPn2ciHIIIE1Tw&hl=ru&sa=X&ved=2ahUKEwjtlL2nkeDnAhUkyKYKHZc9CQ8Q6AEwBXoEC AoQAQ#v=onepage&q=1994%20%D1%81%D1%83%D0%B4%D0%BE%D0%B2%D0%B5%20%D1%80%D1%96%D1%88%D0%B5%D0%BD%D0%BD%D1%8F%20apple%20v%20microsoft&f=false>.
17. Митягин К. Языковые разногласия: как Oracle и Google судятся из-за Java. URL: <https://roem.ru/10-10-2016/231632/oracle-vs-google/>.
18. Томаров І., Ромащенко Д. ТОП-5 судових спорів у сфері технологій. *Юридична газета online*. URL: <https://jur-gazeta.com/publications/practice/inshe/top5-sudovih-sporiv-u-sferi-tehnologiy.html>.