ANALYSIS AND RESEARCH OF FRAMEWORKS FOR MOBILE APPLICATION DEVELOPMENT

АНАЛІЗ ТА ДОСЛІДЖЕННЯ ФРЕЙМВОРКІВ ДЛЯ СТВОРЕННЯ МОБІЛЬНИХ ДОДАТКІВ

c.t.s., as.prof. Korpan Y.V. / к.т.н., доц. Корпань Я.В.
ORCID: 0000-0002-1455-5977
MSc student Nechyporenko O.V. / магістрант Нечипоренко О.В
MSc student Stohniy V.O. / магістрант Стогній В.О.
c.t.s., as.prof. Nechyporenko O.V. / к.т.н., доц. Нечипоренко О.В.
ORCID: 0000-0002-3954-3796
Cherkassy State Technological University, Cherkassy, Shevchenko blvd, 460, 18006
Черкаський державний технологічний університет, Черкаси, бул. Шевченка 460, 18006

Abstract. The paper presents results of analysis and research of mobile application development for popular platforms (Android, iOS, Windows Phone).

Key words: mobile application development, software framework.

Introduction.

With the development of high technology it became possible to create personal mobile devices and various gadgets, which led to the formation of a separate segment in the market – mobile applications. Mobile application is a special program for mobile device that has certain functionality and performs well-defined functions. This includes the following applications: events, site analogs, games, online stores, promotional offers, business, systems, navigation, multimedia, social networks [1].

The specificity of this IT segment is that the development of applications should be based on the features of mobile devices: interface differences, screen sizes differences, touch controls.

Due to the fact that the market offers a huge variety of mobile devices, they use special platforms for working with applications. Today, the most popular platforms are Android and iOS, as well as BlackBerry and Windows Phone.

Native, cross-platform or hybrid frameworks are used to develop a mobile application. At the initial stage of developing a mobile app, the developer faces the task of selecting a framework. A properly selected framework that satisfies the main requirements of the task will reduce costs and development efforts. Thus framework selection is a topical issue.

The main text

It is known that in many ways the specifics of the application are determined by the subject area, which characterizes the composition and features of the models describing the task, as well as the defining of the right approach to the mobile application development.

The following are the programming languages and application development tools for the most popular mobile operating systems and their main features [2, 3]:

1. iOS applications are developed in Objective-C and Swift. The code written in Swift can work along with the code written in C, C ++ and Objective-C within the same project.

2. Android applications are usually written in Java. They are capable of handling large amounts of data. To develop applications for Android operating system, Android SDK is required, which compatible with all modern computer operating systems such as Windows, macOS and Linux.

3. Application development for Windows Phone is done in C # in the Visual Studio environment.

Analysis shows that the selected operating systems have all the necessary qualities: the ability to solve consuming task, integration with external systems and great usability. But, as the result of considering this issue, we can say that the Android OS is the most acceptable platform for developing applications. This is confirmed by a sufficient number of specialists, the ease of mastering the programming language and the ability to process large amounts of data in the shortest possible time.

Table 1 presents some of the analyzed and investigated frameworks for mobile applications development.

Table 1

Frameworks for mobile applications development

Framework	Key Features	Advantages	Disadvantages
Unity	*Languages:* C#, JavaScript, Boo. *Platforms:* Android, iOS, Windows Phone, Tizen, PS 4, Xbox One, Google Daydream, Gear VR, HTC Vive, Linux, macOS, etc.	The engine gives high-quality results without any complicated configurations. Allows you to make your own shaders and change the way Unity renders the game.	UI and difficulty in use for beginners. Source code not available. Compilers are not optimized for ARM processors on some mobile devices.
Qt	*Languages:* C++ QML. *Platforms:* Android, iOS, WinRT, Windows, Symbian, Linux, QNX	Has a lot of good tools, for example: IDE QT Creator, Qt Designer and code profiling. Has libraries with intuitive API interfaces.	Difficult for beginners.
PhoneGap	*Languages:* JavaScript, HTML5, CSS3, Java, Objective-C, C#. *Platforms:* Android, iOS, Blackberry, Windows Phone, WebOS, Symbian, Bada, Ubuntu.	Has a simple API. Ability to use any existing JavaScript libraries. Supports all mobile platforms.	The interface is visualized using the built-in browser, which creates difficulties in receiving feedback as compared to a native application.
Xamarin	*Languages:* C#, Xaml. *Platforms:* iOS, Android, Windows Phone and Windows 8/RT, Tizen	Ability to use TestCloud to test the application automatically. Applications under different systems will look similar. CustomRenderer's standard controls are easily complemented by arbitrary properties.	Some interface patterns are difficult to implement on monodroid and monotouch. There are problems with the mono, monotouch and monodroid platforms. Android pages cannot be located as part of an existing Activity / Fragment.
Appcelerator Titanium	*Languages:* Python, JavaScript, Ruby,	JavaScript makes it easy to develop	There are delays when starting the

	PHP. *Platforms:* iOS, BlackBerry, Android, Windows, Tizen, Denso	programs without the use of platform languages. Appcelerator allows you to do analytics in real time.	program because of library loading. It is difficult to create complex applications.
Telerik AppBuilder	*Languages:*.Net, JavaScript, HTML5, Java, PHP. *Platforms:* iOS, Android, BlackBerry, Windows, Windows Phone	Telerik provides Visual Studio plugins and Sublime Text for AppBuilder. AppBuilder offers a quick way to import Cordova plugins. Full online IDE.	Few users (weak online community).
Android Studio	*Languages:* Java, Kotlin, XML. *Platforms:* Android.	Designed specifically for Android development and officially supported by Google. Built-in Android emulator.	Apps run slow on the ARM version of the emulator. Android development only.
Xcode	*Languages:* C, C++, Objective-C, Objective-C++, Java, AppleScript, Python, ResEdit (Rez), Ruby, Swift. *Platforms:* macOS, iOS, watchOS.	Designed specifically for iOS development and officially supported by Apple. A large number of tools for easier development. Built-in emulators.	Runs only on macOS. Apple development only. Requires Apple Developer account.

As you can see from the table above, each framework has its own peculiarities, advantages and disadvantages, so the developer must choose an "assistant" based on the needs and tasks of the application. Among the main recommendations for developers the following should be highlighted: if the application uses a lot of resources, or if it requires information processing with high speed, It is advisable to create native application; if the performance is not critical, you can create hybrid or cross-platform applications; if the user only needs to receive the information (if there is a network connection), then the web-application is enough.

Summary and Conclusions.

The paper presents results of analysis and research of mobile application development for popular platforms. Mobile application development features were studied.

The research of native, cross-platform and hybrid frameworks for the mobile applications creation showed their advantages and disadvantages. This made it possible to determine the main properties of the frameworks which promoted their widespread usage.

References:

1. Korpan Y.V. Doslidžennja vykorystannja mobilʹnyx dodatkiv v systemax identyfikaciji ljudyny / Y.V. Korpan, O.V. Nechyporenko, V.O. Stohnij // V Naukova konferencija «Fundamentalʹni ta prykladni doslidžennja u sučasnij nauci». Zbirka naukovyx prac. – Xarkiv, X. : Texnolohičnyj Centr, 2017. – pp. 52

2. Mayorova E.S. Sovremennoe sostoyanie sredstv razrabotki mobilnyih prilozheniy na platformah iOS, Android i Windows Phone / E.S. Mayorova, V.A. Oshurkov, L.S. Tsuprik // Perspektivyi Nauki i Obrazovaniya. 2015. issue 4 (16). pp.83-87.

3. Aksenov K. V. Obzor sovremennyih sredstv dlya razrabotki mobilnyih prilozheniy // Novyie informatsionnyie tehnologii v avtomatizirovannyih sistemah. 2014. issue 17. pp. 508-513.

Аннотация.

В роботі представлено результати аналізу популярних платформ мобільних пристроїв. Приведено результати дослідження нативних, кросплатформних та гібридних фреймворків для створення мобільних додатків, визначені їх переваги та недоліки. Надано рекомендації до вибору певного фреймворку.

Ключові слова: розробка мобільного додатку, фреймворк.

Литература:

1. Корпань Я. В. Дослідження використання мобільних додатків в системах ідентифікації людини / Я. В. Корпань, О. В. Нечипоренко, В. О. Стогній // V Наукова конференція «Фундаментальні та прикладні дослідження у сучасній науці». Збірка наукових праць.– Харків, Х. : Технологічний Центр, 2017. – С. 52.

2. Майорова Е.С. Современное состояние средств разработки мобильных приложений на платформах iOS, Android и Windows Phone / Е.С. Майорова, В.А. Ошурков, Л.С. Цуприк // Перспективы Науки и Образования. 2015. №4 (16). С.83-87.

3. Аксенов К. В. Обзор современных средств для разработки мобильных приложений // Новые информационные технологии в автоматизированных системах. 2014. №17. С.508-513.

Article sent: 22/11/2017 of