

INNOVATIVE TOOLS AND PLATFORMS FOR MOBILE DEVELOPMENT

Urinbaeva Azada Bakhadirovna

Kokan State Pedagogical Institute named after Mukimi

Intern-Teacher of the Department of Informatics

azadabax@gmail.com

ABSTRACT

Mobile application development technologies were studied, advantages and disadvantages of these technologies were identified.

Keywords: mobile application, programming languages, Android studio, Flutter, Visual Studio Code.

INTRODUCTION

Mobile development is one of the fastest growing areas in technology. In recent years, mobile devices such as smartphones and tablets have become important in our daily life. As a result, the demand for mobile applications and services has increased significantly.

There are many technologies and programming languages used in mobile development. In this article, we will look at the most popular of them, as well as compare their advantages and disadvantages.

Native Development is a technology for creating mobile applications that use operating systems and programming languages designed for specific devices. For example, iOS uses Swift or ObjectiveC, while Android uses Java or Kotlin. Advantages of this technology: high performance and fast program operation, possibility to use all functions of the device, safe programming can be included.

Cross-platform development is a mobile application development technology that allows you to create applications for multiple platforms at the same time. Tools like React Native, Flutter, Xamarin are used for this. The advantages of this technology include: reducing development time and costs, the ability to use the same code for several platforms, and ensuring the versatility of the application.

Hybrid development is a mobile application creation technology that combines elements of native and cross-platform development. Tools such as Apache Cordova and PhoneGap are used for this. Advantages of this technology:

can include reduced development time and costs, the ability to use HTML, CSS, and JavaScript for development, cross-platform work.

Creating mobile apps can be a very simple process: you open an integrated development environment, do a quick build, easy testing, and you submit the app to the App Store - all this takes half a day. Or, conversely, you do a careful initial design, perform usability testing on multiple highly complex devices, maintain a full beta version life cycle, and then distribute in a variety of ways.

Today, the following environments and tools are often used for mobile application development:

Android Studio: The main official IDE for Android development, built on IntelliJ IDEA from JetBrains. Android Studio offers a wide range of tools integrated with Google, making it the preferred choice for Android developers.

Xcode: The main IDE for iOS, macOS, watchOS and tvOS developed by Apple. Xcode includes all the tools you need to build, test, and deploy apps to the Apple ecosystem.

The tools included in this IDE have been chosen to make it easy for a developer to go through the full cycle of creating an app, primarily for Apple platforms. It's similar to a text editor, but with more functionality. The code editor recognizes the language's syntax and highlights its structures for greater readability. It can signal syntax errors or provide a brief description of a particular command. It counts lines, has flexible code navigation and many other features.

Visual Studio Code: A lightweight and extensible environment from Microsoft, popular among developers due to its many extensions. With its help, you can develop cross-platform applications using various frameworks such as Flutter and React Native.

Visual Studio Code (VS Code) is a cross-platform code editor of Microsoft Corporation, developed on the basis of the Electron framework. With its help, you can develop cross-platform desktop applications using web technologies.

VS Code's great strength is that it's not tied to a specific programming language, so you can use it to build websites, mobile apps, work with databases, and test services. A huge library of plugins allows you to extend the functionality of the editor, and if you can't find a suitable plugin, you can always write your own.

React Native: A framework from Facebook for building cross-platform mobile apps using JavaScript and React. This allows you to use the same code for Android and iOS, which significantly speeds up the development process.

Flutter: Google's framework for building cross-platform applications using the Dart language. Flutter enables the development of high-performance applications with a beautiful user interface.

Flutter stands out among cross-platform development platforms due to the following advantages.

Performance is close to platform-dependent applications. Flutter uses the Dart programming language and compiles to machine code. Devices understand this code, which ensures fast operation and high performance.

Fast, consistent and customizable display. Flutter does not rely on platform-specific display tools, but uses Google's open-source Skia graphics library to render the user interface. This provides users with consistent visual elements regardless of the platform used to access the app.

Developer-friendly tools Google designed Flutter with ease of use in mind. With features like hot reload, developers can see what changes to the code will look like without losing state. Other tools, such as the Widget Inspector, make it easy to visualize and troubleshoot UI layouts.

Xamarin: Microsoft's tool for cross-platform development in C#. Allows you to write code that works on Android, iOS and Windows systems.

Xamarin is an open source platform for building modern, efficient apps for iOS, Android and Windows with .NET. The Xamarin platform is an abstraction layer that provides control over

the interaction between shared code and the underlying platform code. Xamarin runs in a managed environment that implements features such as memory allocation and garbage collection.

With Xamarin, an average of 90% of application code can be used across platforms unchanged. With this pattern, a developer can write all business logic in the same language (or use existing application code), but still get the performance, appearance, and behavior specific to each respective platform.

Xamarin apps can be written on a PC or Mac and compiled into native app packages, such as apk files for Android or .ipa files for iOS.

In conclusion, the choice of technology and programming language depends on the goals and requirements of the mobile application. Native development involves creating separate versions of the program for each platform, which can be complex and expensive, but provide high performance. Cross-platform development allows you to build apps for multiple platforms at the same time, which reduces time and cost, but may result in a loss of productivity. Hybrid development combines elements of native and cross-platform development, which reduces development time and costs, but also limits the use of device functionality.

It should also be taken into account that the choice of technology and programming language can affect the further expansion of the program, its security and reliability. Therefore, before choosing a technology, it is necessary to study all the advantages and disadvantages, as well as analyze the application requirements.

REFERENCES

1. Mamadjanova Svetlana Valentinovna, Urinbaeva Azada Baxadirovna. "The Role Of Mobile Applications In The Education System". Onomázein, no. 62 (2023): December, Dec. 2023, pp. 2088-93,
2. Bakhadirovna, Urinbaeva Azada, Erkinjonova Shakhрина Oybekjonovna, and Yolchiyeva Oygul Khan Shermuhamadovna. "THE ROLE OF MOBILE TECHNOLOGIES IN THE DEVELOPMENT OF INFORMATION COMMUNICATION TECHNOLOGIES IN EDUCATION." Galaxy International Interdisciplinary Research Journal 11.12 (2023): 1192-1194.
3. Urinbaeva Azada Baxadirovna, & Abdumannonov Olimjon Xurramjon o'g'li. (2024). RIVOJLANIB BORAYOTGAN DUNYODA "MASOFAVIY TA'LIM" O'RNI VA AHAMIYATI . Ta'lim Innovatsiyasi Va Integratsiyasi, 21(2), 40–42. Retrieved from <http://web-journal.ru/index.php/ilmiy/article/view/5052>
4. Urinbaeva Azada Baxadirovna, & Azimov Azizbek Alisher o'g'li. (2024). 7-SINFLARDA KO'PAYTMA VA YEG'INDINING ASSOTSIATIVLIK XOSSASINI GEOMETRIK SHAKLLAR ORQALI TUSHUNTIRISH . Ta'lim Innovatsiyasi Va Integratsiyasi, 21(2), 43–45. Retrieved from <http://web-journal.ru/index.php/ilmiy/article/view/5053>
5. Urinbayeva Azada Baxadirovna, & Abdullayeva Niluzar Valijon qizi. (2024). O'NLI KASRLARNI HISOBLASHNI O'RGANISHDA KHAN ACADEMY PLATFORMASIDAN FOYDALANISH . Ta'lim Innovatsiyasi Va Integratsiyasi, 21(2), 46–48. Retrieved from <http://web-journal.ru/index.php/ilmiy/article/view/5054>

6. Urinbaeva Azada Baxadirovna, Ne'matov Zubayr Yo'lchiboy o'g'li. "Veb Xavfsizligi Zaifliklari Va Qarshi Choralarning Chuqur Tahlili: Umumiy Hujumlar Va Samarali Mudofaalarni o'rganish". Science Promotion, vol. 9, no. 1, May 2024, pp. 363-71, <https://sciencepromotion.uz/index.php/sp/article/view/1736>.
7. <https://blog.skillfactory.ru/glossary/xcode/>
8. <https://skillbox.ru/media/code/visual-studio-code-ustanovka-nastroyka-rusifikatsiya-i-spisok-goryachikh-klavish/>
9. <https://aws.amazon.com/ru/what-is/flutter/>
10. <https://learn.microsoft.com/ru-ru/previous-versions/xamarin/get-started/what-is-xamarin>