IOS app development using Python

What types of mobile products are good for Python?

Python mobile development was not popular previously. But things have changed with the appearance of different Python GUI frameworks. And currently, this programming language is widely used in mobile apps development.

Python is rather versatile. It can be used for building various apps: starting with web-browsers and ending with simple games.

One more powerful advantage is being cross-platform. So, it's possible to develop both Android and iOS apps in Python.

IOS application development

iOS application development is the process of making mobile applications for Apple hardware, including iPhone, iPad and iPod Touch. The software is written in the Swift programming language or Objective-C and then deployed to the App Store for users to download.

This document has been produced with the support of the EUROPEAN COMMISSION under the ERASMUS+ Programme, KA2 – Capacity Building in the Field of Higher Education: 598092-EPP-1-2018-1-BG-EPPKA2-CBHE-SP. It reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

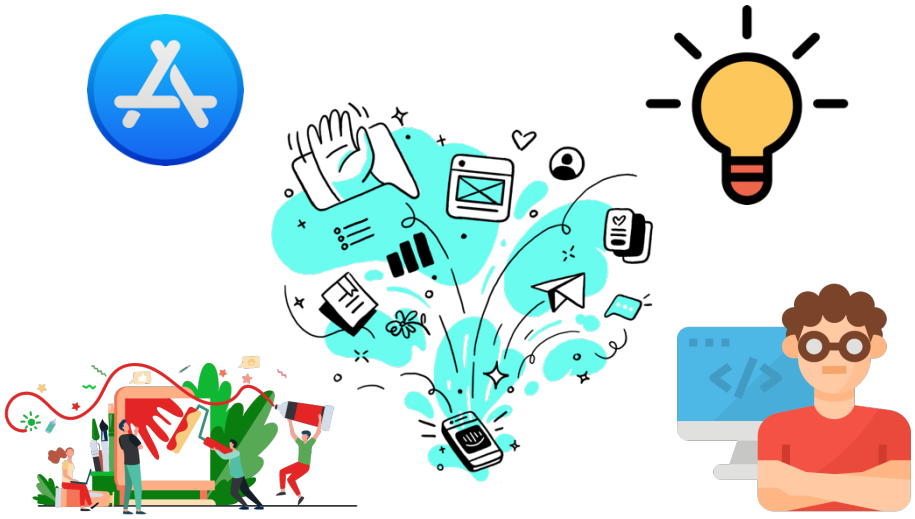
If you're a mobile app developer, you may have had reservations about iOS development. For example, each developer needs a Mac computer—and Macs are generally more expensive than their Windows-based counterparts.

Mobile App Development Using Python

Python is a rather universal programming language that is appropriate for solving a wide range of tasks. And everybody knows that it is used in such spheres as web development, data science, and different processes automation.

The increase in the number of smartphone users and the high demand of mobile apps have led to the use of Python for mobile app development. In fact, this programming language has grown to become a top choice of the mobile app development market. Python is known for its simplicity, clean syntax, and smooth operation.

Python is one of the finest and most easily readable programming languages when compared to various programming languages in existence. Most programmers prefer working with this programming language, which can be used to develop apps for Macs, Windows, Amiga, OS/2, and Linux.



This document has been produced with the support of the EUROPEAN COMMISSION under the ERASMUS+ Programme, KA2 – Capacity Building in the Field of Higher Education: 598092-EPP-1-2018-1-BG-EPPKA2-CBHE-SP. It reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

If you are going to build an app by yourself, the main resource you need is time. How long will it take to develop anything in Python? Of course, it depends on the difficulty of the idea and the level of your programming skills.

If you are going to manage the project but not to be engaged in development then you need to know how much time different stages take:

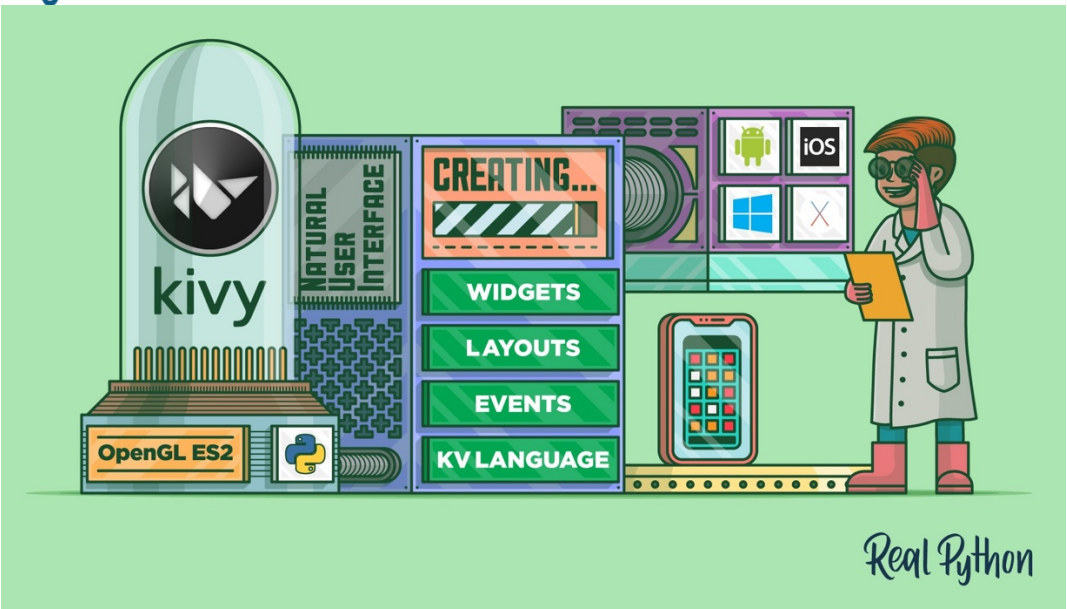
1. Creating a brief for the developers — 7-14 days.
2. Looking for the appropriate programmers and discussing the idea with them — approximately a month.
3. Design development — 1,5-3 months.
4. Development of the app — 3 months.
5. Release on App store — up to 14 days.

Python libraries

Python also has libraries and frameworks that let you create one codebase for various platforms (i.e. Android, iOS). This is called cross-platform development.

When it comes to Python mobile apps, cross-platform development is a big advantage. Traditionally, to create an app for both iOS and Android, the developers had to work simultaneously on two completely different app versions based on different programming languages. Today, Python frameworks such as Kivy and BeeWare enable developers to easily transform one piece of code into versions working with iOS, Android, or any other platform of your choosing, without having to create separate apps one by one.

This document has been produced with the support of the EUROPEAN COMMISSION under the ERASMUS+ Programme, KA2 – Capacity Building in the Field of Higher Education: 598092-EPP-1-2018-1-BG-EPPKA2-CBHE-SP. It reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



Kivy is a cross-platform Python framework created to assist in rapid app development. It supports various user interfaces, including multi-touch screens and various platforms, including iOS, Android, and Windows.

Kivy has its own custom UI toolkit, which will look consistent and work exactly the same between Android, iOS, Linux, and Raspberry Pi, but it won't use any native features of any of those platforms. This can be a downside or an upside, depending on what kind of app you're planning to develop. On the one hand, users tend to favor native look in most apps, but UI design that stands out can be a powerful design choice that lets users work in your app on various platforms seamlessly.

Kivy's most important features:

- One code for all platforms
- Robust graphics engine built over OpenGL ES 2
- Published on MIT license: it's free to use for business
- Custom UI toolkit
- One codebase for all platforms
- Published on BSD license and free to use for business
- Apps with native look and feel
- Huge community of invested developers
- Not just one tool, but a collection of projects
- Fast, simple, and precise

This document has been produced with the support of the EUROPEAN COMMISSION under the ERASMUS+ Programme, KA2 – Capacity Building in the Field of Higher Education: 598092-EPP-1-2018-1-BG-EPPKA2-CBHE-SP. It reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

BeeWare

The IDEs of Python

Another popular, slightly newer framework that lets you create one Python code and release it to multiple platforms (including iOS, Android, and Windows) is **BeeWare.** The big difference here being BeeWare supports a fully native user experience for each platform. End users don't care what language was used to build the app they're using, but they may care about native appearance and behavior.
With BeeWare, all the fonts and buttons in your app are the ones the user is most used to. Even the feel and behavior of your Python mobile app is distinctly Android or iOS. BeeWare is BSD licensed and available for all to use and modify. The team of developers behind BeeWare development actively and successfully encourages the growth of a diverse and welcoming community.

How BeeWare stands out?

- One codebase for all platforms
- Published on BSD license and free to use for business
- Apps with native look and feel
- Huge community of invested developers
- Not just one tool, but a collection of projects
-

This document has been produced with the support of the EUROPEAN COMMISSION under the ERASMUS+ Programme, KA2 – Capacity Building in the Field of Higher Education: 598092-EPP-1-2018-1-BG-EPPKA2-CBHE-SP. It reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Firstly, Kivy has a custom UI, BeeWare supports native UI. None of these approaches is unequivocally better, it always depends on the type of app you want to make. Kivy is an excellent choice if you expect users to operate your app on different devices and you need its look and controls to be consistent. On the other hand, BeeWare works from a codebase to produce different code versions for different platforms. This in exchange allows it to fully support native UI, so your app will use things like platform-specific buttons and behave in the way that is expected on the platform.
Secondly, BeeWare is still in its development phase, while Kivy is much more established. But that doesn't necessarily mean that BeeWare has worse support – there's a thriving community of developers working actively to improve it.
Finally, the third biggest difference between the two frameworks lies in their complexity. Kivy is fast, straightforward, and precise, allowing developers to create great simple apps for various platforms from one code. BeeWare, being the complex toolset that it is, may be better suited for more sophisticated projects.

Should you create your mobile app in Python?
Although we believe that Python, as of 2021, is a perfectly capable language for mobile development, there are ways in which it is somewhat lacking for mobile development. Python is not native to neither iOS or Android, so the deployment process can be slow and difficult. This may also lead to some inconsistencies between different app versions.
Fortunately, those cons can be easily mitigated by working with experienced Python and mobile application experts. If you're looking for mobile development experts to build your app with Python, our specialists are ready to help you.

This document has been produced with the support of the EUROPEAN COMMISSION under the ERASMUS+ Programme, KA2 – Capacity Building in the Field of Higher Education: 598092-EPP-1-2018-1-BG-EPPKA2-CBHE-SP. It reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.