In our previous post we began discussing the main new features of iOS 12. Let’s have a look at other business benefits the new release can bring.

Authentication Services

+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk

A whole new relevant framework has been added in iOS 12. Here a new native mechanism for the *Single Sign-On* principle has been implemented. If you're providing a password management service, you're now able to interoperate with the P*asswords AutoFill* function and input your records into it. The user is thus given a convenient and secure method of creating and storing passwords, without ever having to type them. Besides, it's now possible to autofill SMS auth codes into native app inputs, which your users are gonna love!

- By means of Auth Services, your website and a native app may securely share cookies and auth tokens.

CarPlay for Navigation Apps

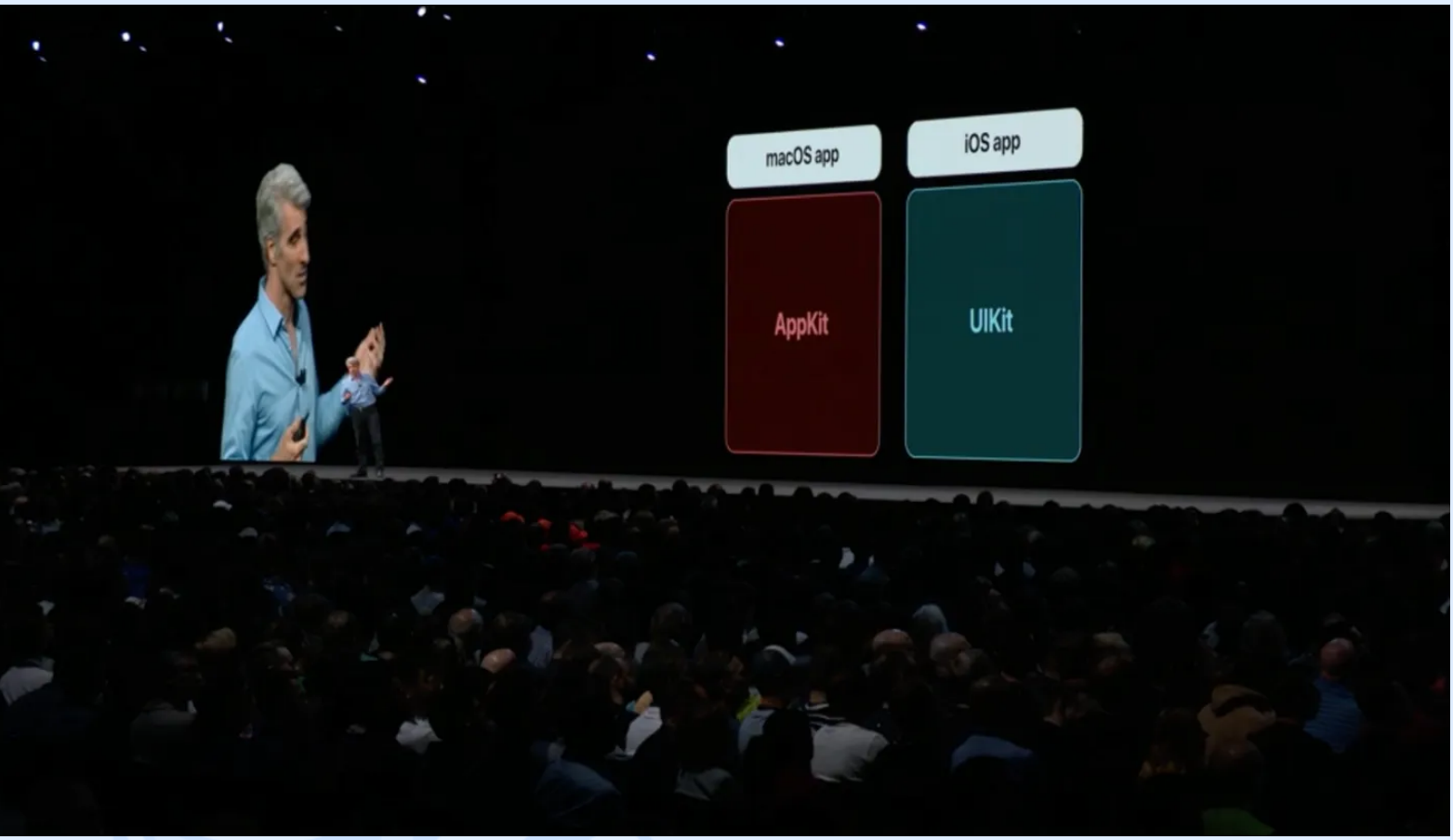
+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk

If you’re running a navigation service or app, this is really big news for you, since it is now possible to integrate with *CarPlay* from any navigation app. For others, it offers nothing yet, but I consider *CarPlay* a potentially huge media content delivery channel just as Apple Watch was in its time for on-the-go tasks.

Network Framework

+44 20 7183 5820

info@magora.co.uk
sales@magora.co.uk



And again a whole new framework. This one is purely under the bonnet, and might not sound like something big and interesting. Nonetheless, it changes a lot for particular use cases. If you're developing a high-performance real-time networking app, like a text chat or a media broadcasting application – or maybe a multiplayer game – you really have to examine this new framework.

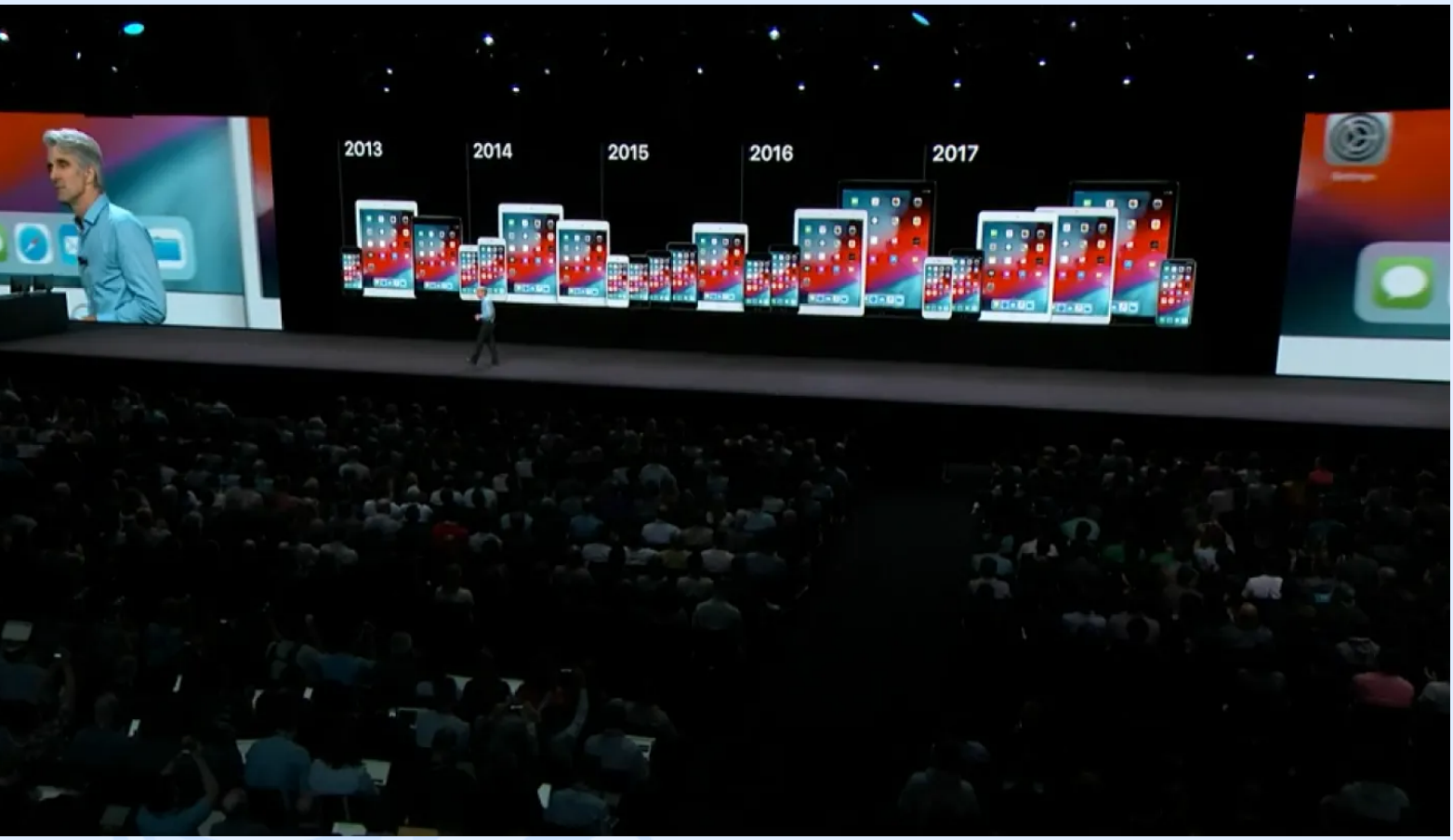
- Low-level network programming may become available for a much broader range of developers and definitely makes it possible to produce a high-quality low-level networking code with much less effort.

- What does this mean from a business point of view?

- The new development environment lets you cut about two weeks off the hard coding of initial technical things. With the Network framework programmers can start to implement the actual features of an app without the lengthy prearrangement and the overall cost of a complicated real-time game or app will be beneficially lower.

Natural Language

+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk



Using natural language as a mechanism of communicating with the user is becoming a new trend. Some time before, it took years and millions of dollars to create such a system, one that could understand what a real human meant, and this system was something exclusive. Thanks to Apple, we're now able to extract some knowledge from a natural language text for free inside any mobile app. You are given an opportunity to isolate separate words, distinguish parts of speech, and even extract the names of people, places and organisations. If you have a suitable idea for utilising such technology, don't wait – go and use it!

OpenGL ES deprecated

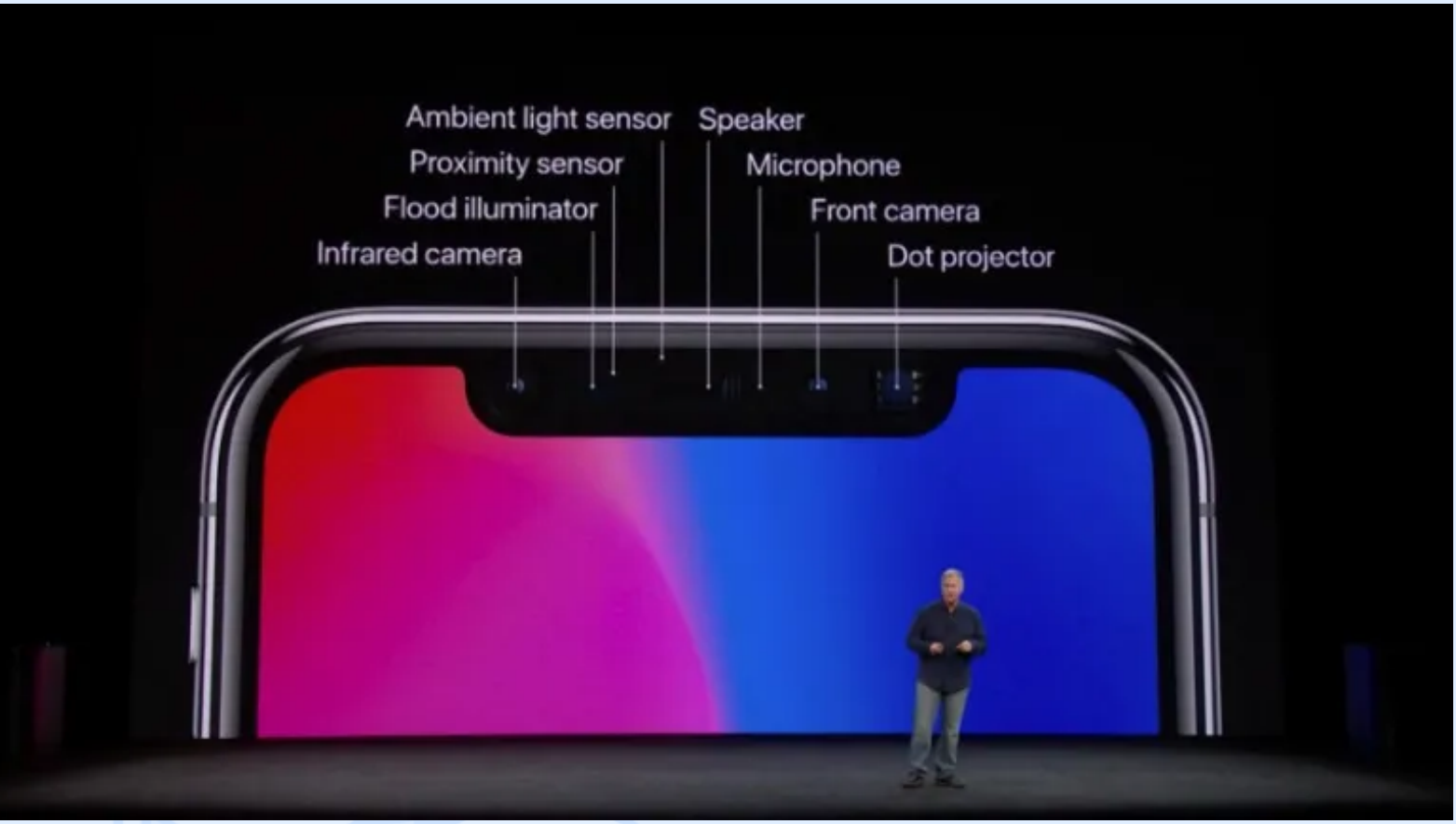
+44 20 7183 5820

info@magora.co.uk
sales@magora.co.uk

This news sounds like something purely technical again, but in fact, affects a lot of projects. The thing is that Open GL|ES has been a de facto standard of games development for years, going back to the original iPhone. OpenGL is not unique to the iPhone, however, and actually arrived from the PC. That's why most cross-platform games have been based on it. Not only games, in fact, but any high-performance graphics — scientific charts, beautiful calendars, heavily customised animations… So if you have some of that, maybe it's time to move forward and port the graphics into the Metal platform. This complication is the price of increased iOS graphics performance and hardware usage efficiency. Universal platforms, on the other hand, are never that efficient.

Other

+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk



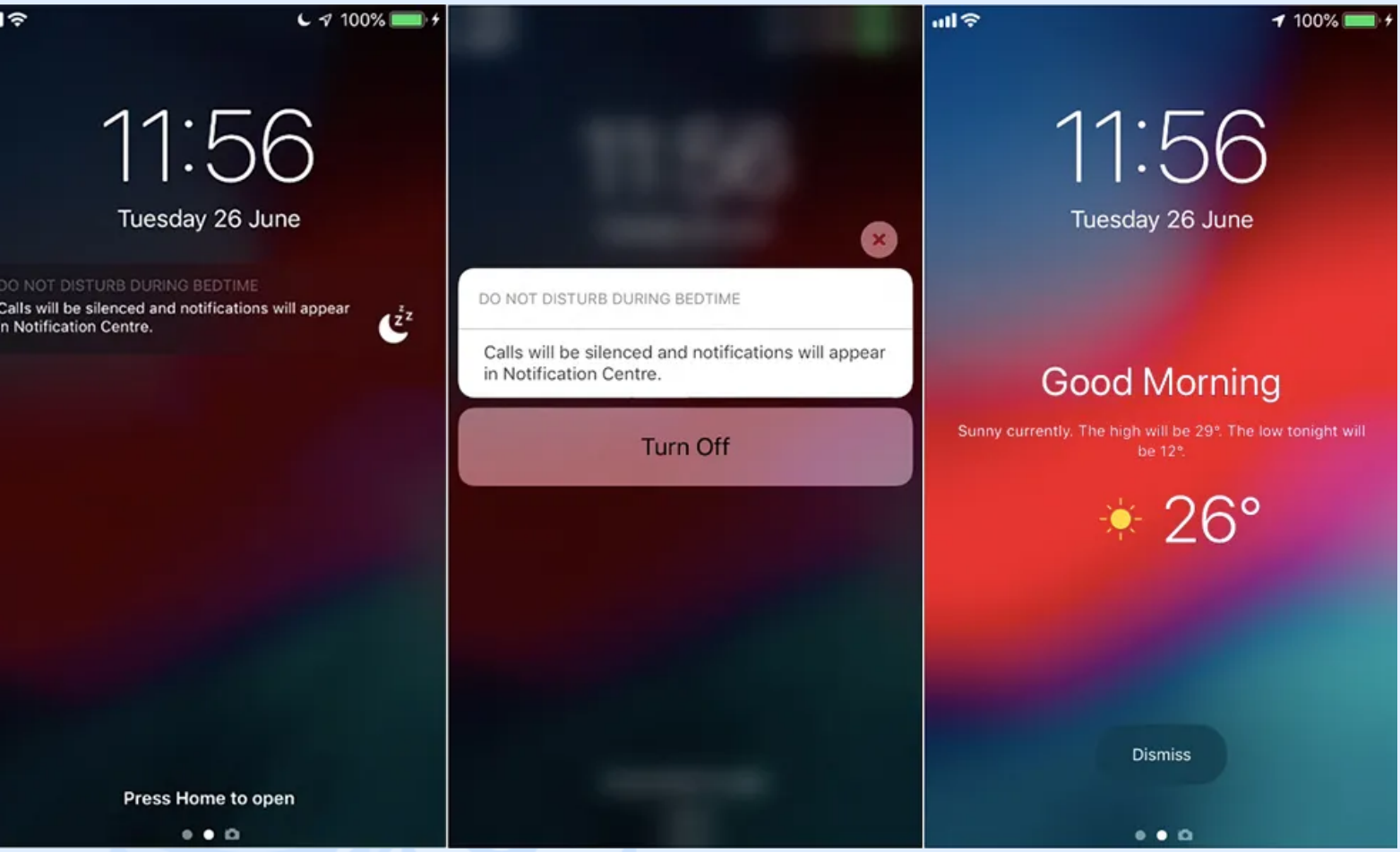
Apple makes available to app developers real-time access to the iPhone X TrueDepth camera system. It is now possible to implement visual effects and user interactions based on a 3D scene from the camera.

CreateML is a new framework for MacOS, which makes it possible to create and teach custom models for later usage with the CoreML. Machine Learning is not a simple technology for development, but things are moving forward and many more applications are incorporating it, making apps simpler to use. Having powerful instruments like CreateML also makes them simpler to develop.

What’s already here

+44 20 7183 5820

info@magora.co.uk
sales@magora.co.uk



Each year, as a new Beta version is made available, every app owner really should begin testing his app on the new platform. You shouldn't fear many of the new bugs appearing in the first beta, as some of them may be caused by issues with the system itself. If you encounter the same for several betas in a row, however, maybe this is a signal to delve deeper and look for problems in your code. The app should work perfectly while a new OS version is made publicly available.

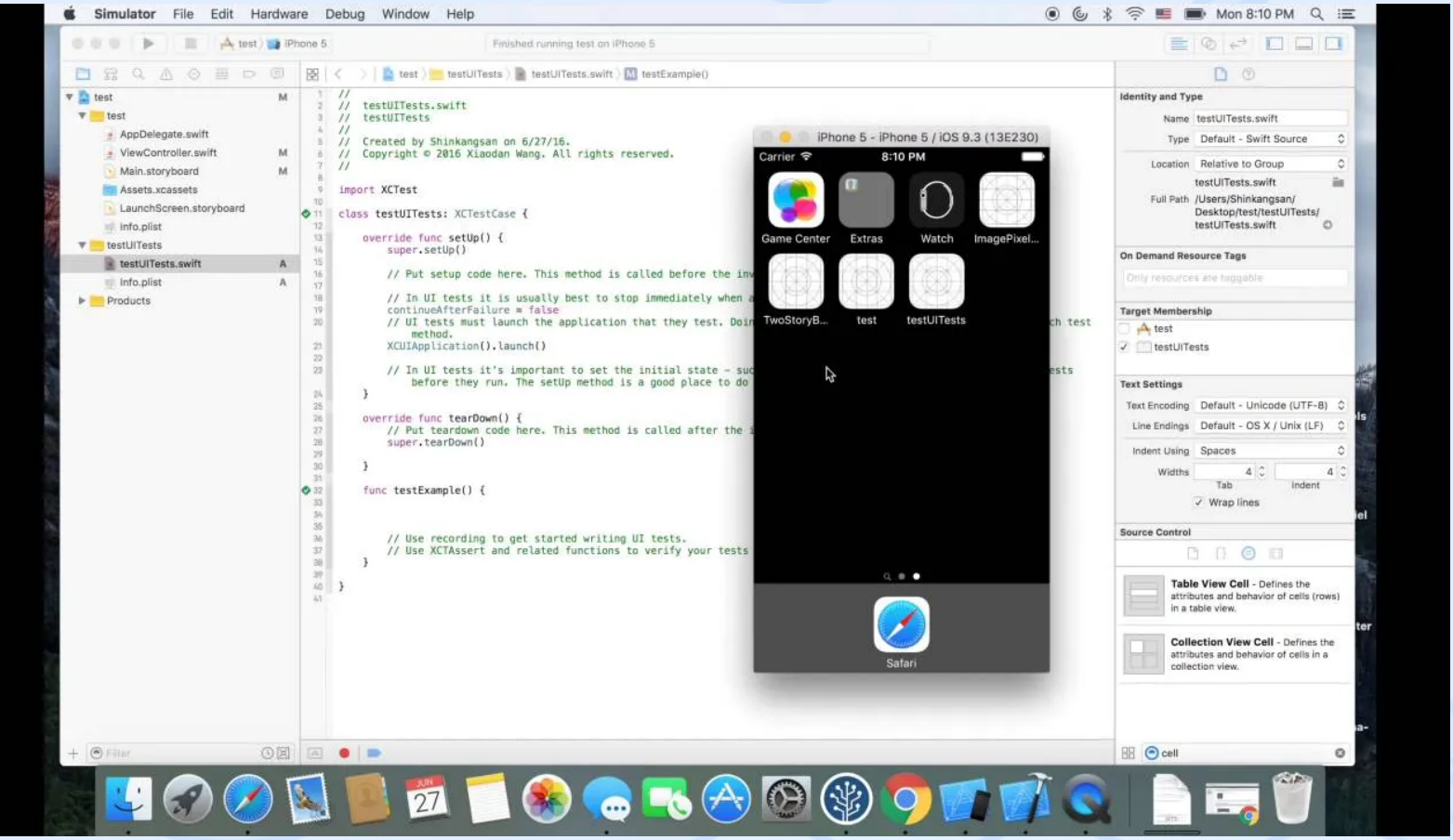
We've gone through lots of new features and technologies. Let's remember what's already here and what was added years ago and is yet to be widely adopted.

- After some time with the OS version available on the market, you should check the adoption rate for your own application. In most cases we've found it reasonable to support 2 to 3 of the latest iOS v3ersions. This gives around 95% of users coverage, while leaving development comfortable enough not to waste the owner's money. Supporting more than 3 major OS versions becomes too expensive for both development and testing and makes it hard to focus on new OS features, as you have to make some kind of segmented differentiation between users with modern and legacy versions.
- Each year you choose the "just right" moment and drop some old version, say iOS 9. Look at that, now *all* of your users have the features introduced in iOS 10. Now you can easily integrate the Speech framework, making Siri voice recognition available inside your app, or move onto a modern iOS 10-specific mechanism of interactive animations — these work smoothly and take

less effort to deal with compared to the older one. You've just dropped iOS 10 — let's use iOS 11's Drag and Drop functionality for your iPad users.

Our experience shows that not many app owners think of this. Don't repeat others' faults – go through not only latest iOS changes, but also through all the possibilities now available to you and that may be available in each iOS version.

Swift 5 and Xcode 10 (Developer corner)



User-facing features aside, development instruments evolve with the times as well. There are many innovations in this field coming in late 2018. And I, as one of the Magora development team's members, wish to highlight the most intriguing upcoming language and IDE features.

- The Xcode build system was a real disaster in the times of several early Swift versions. It crashed every now and then; syntax highlighting almost never functioned; it was possible to catch a compilation error three times before you could finally run the project — without changing anything, pure black magic! Supporting a large project was quite a painful task. Add to this a huge compilation period every time you change anything in the app. In normal conditions only changed

+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk

code should be recompiled each time you run the application during the development and debugging process. With Swift, this is becoming true only now, five years after it was released and became popular. It would be fair to say that Swift became an industrial standard even before growing from an enthusiast-level toy into a robust tool. Well, it's also fair to say it's not Swift's problem – it's all about community choice (Swift over Objective-C). Surely, things are getting better now, but they're still not ideal. It's been announced that Xcode 10 will use a new build system that should solve most of the listed issues.

Our department switched to Swift when it was at v3 for commercial code. Before that, it looked as though developers were taking money from customers and putting them on tech-support.

Developers had to use hacks, as there were no coding standards, design patterns or best practices yet created for the new language. And the language itself changed a lot for the first three years. If anyone implemented a simple application using Swift 1 and the libraries which were relevant that time, several years later rewriting this app from scratch would be cheaper than supporting the old one. Now the language has become pretty stable and it takes less effort while migrating to a new version.

There have been a lot of rumours surrounding Swift's ABI stability (binary compatibility) since its initial release. Most of the system and commercial 3rd party solutions are delivered as binary libs. With Swift we had to use the open-source model, or else suffer with supporting each individual compiler version, even a minor one. So, developing a proprietary library with Swift was never a wise decision.

- Swift 5 should finally bring ABI stability for the standard libraries (Swift Dynamic Library). It offers a notable reduction in app bundle size — now just one copy of Swift itself will be stored in the system, instead of everything being copied into each application. Later, when the whole language is ABI stable, it will be the beginning of a new era when commercial b2b programmatic solutions can be developed in Swift.

There's also a bunch of cool new language features and syntax sugar. To list a few, these include language support for regular expressions; first-class concurrency support with async/await pattern and an actors model. All of these (if they are in fact included in it) could make Swift 5 one of the most advanced and modern languages in terms of language-level features. It gives us the freedom to express quite complex things in a few lines of code. Supporting really complex systems thus become a lot easier and doesn't require a developer to be Einstein. And that's how we moved forward from a simple command line utility that cost you a fortune several decades ago to the AR experience, deep ML and all the other things we have in a cell phone.

Conclusion

If you have a long-running product, just looking into hype technologies may not be enough. To stop your users from escaping to competitors, your app needs to become really comfortable and accustomed to the user. To achieve this, integrate with the iOS ecosystem as much as possible to make your app look like an essential part of your user's phone, part of their everyday routine. No one ponders removing Safari, right? Try to become the same in your segment. And here we can help - contact us to discuss your app development project.

+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk

magora

magora

+44 20 7183 5820

info@magora.co.uk
sales@magora.co.uk