

Interview with Cocomore, agency for marketing and IT services

Posted on [September 30, 2015](#) by [Dominique Hazael-Massieux](#) — [Leave a comment](#)

The HTML5Apps continues its series of interviews of European SMEs. Today, we go to Germany, where the [Cocomore](#) headquarters are located. The discussion was held with Alejandro Leiva, based in the Cocomore Spanish office in Seville, is the technical lead of mobile and emerging technologies, and with Christian Winter, Head of Frontend Development in Frankfurt, specialized in responsive Web design and the optimization of interactive Web applications for mobile devices.

Hello! Would you please describe your company, its mission/vision and its customer audience?

Cocomore is an international agency for Marketing and IT services with 140 employees at our head office in Frankfurt, and in Geneva and Seville. We are developing integrated communication and IT solutions with best practices in crossmedia, branding, CRM, marketing automation and e-commerce.

Our mission and our promise are measurable results in these particular areas, achieved using creativity and technology. Our main focus is on projects where it comes not only to appeal to customers, but also to create a long-term customer relationship. Therefore, data and IT is important but above all, you need content which is appealing, entertaining and useful for communication with customers.

That is why Cocomore has not only designers and creative directors, but also editors. On the basis of our four values, which are innovation, tradition, progress and responsibility, our agency is trusted by leading clients including: the European Union, Lilly, Nestlé, Procter & Gamble, Merz, RTL and Sanofi. Our customer relationships usually grow from year to year and are long-term based.

Can you tell us more about cross-platform mobile applications you have worked on?

Cocomore has extensive experience in the implementation of native and cross-platform apps.

Recently, we have accomplished a project regarding a cross-platform mobile application for Germany's largest drugstore chain: dm. We developed a game app, named "Mission Morgen" to support communication around sustainability topics. After the functional concept phase was completed, we looked at which technological approach would be best to implement the app; in particular, we evaluated which of a pure native or hybrid app would fit the project. For reasons of cost, time, supported devices and the required features, we decided to go with an hybrid app.

Our technology selection has proven to be the right one: the combination of ngCordova, AngularJS and Ionic allowed us to focus on the important tasks as we did not need to spend much time on the development of basic functions. Especially ionic provides an optimal basis. With the use of AngularJS, we were able to divide the tasks clearly and to scale optimally the frontend team. In addition, thanks to this technological choice, we have retained over the entire duration of the project a good code base and we were able to integrate new modules quickly and with high quality in the existing app. The backend development was realized on top of Drupal. The resulting free application is available for smart phones with the operating system iOS or Android.

As another example, Cocomore developed a pure native mobile application for Pampers. We are currently evaluating relaunching it as a hybrid app. Our goal is to provide updates with new features or extensions several times a year. Regularly, we provide information with the dynamic newsfeed on the homepage of the app about current actions on the Web site and on Facebook (competitions, new craft instructions, new stories to download). The Web site and the mobile app are optimally matched: there is no duplication of content, the app features complement the Web site features.

Initially, when we first developed the app, we wanted to target as many mobile devices as possible, so we designed two native apps, one for Android and one for iOS. We have now decided to relaunch as a hybrid app since this means a smaller development cost and the opportunity to operate on more mobile devices.

Cocomore has realized many other mobile projects, e.g. an iPad app for our client Fresenius. This app teaches health care professionals in an entertaining way about the benefits of the product OsvaRen. It is a very visual app with embedded videos and an in-app game.

We also developed a guided product tour for the medical device company Spine Art. It shows to surgeons the product range of implants, how-to-videos and animated 3D visualizations of their products.

Other projects included a career app for the HR department of a large pharmaceutical client, a recipe database targeted at diabetes patients to easily monitor calory intake, an ecommerce app allowing consumers to shop diapers when waiting for the bus or tram. And many more.

What are your views on native vs Web apps?

A native app needs to be developed in specific programming language which is determined by the operating system such as Objective C, Swift, Java, C ++, C #, XAML. For that you need a programmer who has knowledge of all development methods (IDE's development tools, emulators, debuggers, SDK's, etc.) in addition to the programming language or more software developers for each varying operating system.

Web apps are very different in this regard: they can be programmed by an experienced Web developer for several different smartphone operating systems without having unique knowledge about the device-specific development methods. For us, with our strong existing engineering skills in Web development, this is a huge benefit.

Moreover, native applications are designed for an operating system such as iOS, Windows Phone or Android, which means they run only on these devices. Here, the different operating system versions have to be considered, so that the correct functionality is ensured. When a new version of the operating system is introduced, the native app must be updated.

This does not affect Web apps. Web applications are specially programmed HTML5 sites that recognize the mobile device and optimize the content for representing, they run on all Web-enabled devices.

The big drawback of Web apps is their lack of access to phone specific functionalities like camera, microphone. Since Web apps run in the browser of the device access to phone hardware is very limited.

Cocomore deliberately focuses on cross-platform mobile applications (also called hybrid apps), which help bridge the gap between native and Web apps. This kind of application consists of a Web app, i.e. a backend and frontend running on a server. Thereby, the app can also be accessed by desktop browser. It is then displayed as a normal Web site.

The second component is an app container which can be uploaded to the various app stores. This container allows the access to the smartphone hardware and it draws its content from the Web app. Thereby it combines the advantages of both types of apps. In fact those hybrid apps are built using Web technologies and with some native code, so they can be converted via framework in a native app container very easily. But due to their architecture leveraging the many advantages of the Web applications, such as the cross browser compatibility and the cost benefit, the future could be these hybrid apps. The most important factor being here the availability of all the native features.

Ultimately, which considerations drive your choices between native or Web?

Depending on the individual requirements, each of these applications can be a better choice. If you want to create an app that makes very advanced use of the hardware and has high performance demands (execution speed, processing speed, output speed / graphics), a native app will be the better choice. The same applies if a high level of integration in the operating system such as look and feel of the UI elements or interaction with other applications is required. Also if the app should be executable without an internet connection, the native app is a better fit.

A Web app is particularly advantageous regarding total cost of ownership and implementation time. It bears comparably low development costs and short development periods. This technology

is directly accessible via a link, so it has the ability to run on many mobile devices and operating systems and runs on each smartphone without installation.

In practice, the approach we currently recommend to our customers is to have a mobile Web portal for general information sharing, and a cross-platform hybrid app for more interactive / engaging content.

What is missing to do the interactive content from within the browser, rather than going through an hybrid app?

There are various considerations that makes a browser-based approach less attractive:

- Our customers don't necessarily accept that a browser-based approach can give good enough results in terms of performance and UI,
- They tend to see also a "real" app as a better marketing/communication channel,
- The app tend to be for shorter-term campaign, with often a gamification aspect, and it's not obvious to end-users that you would obtain a game or game-like experience from within the browser,
- On the Web, our quality expectations require good results on a much wider set of platforms that we would target with our hybrid apps (i.e. mostly iOS and Android), and thus the more interactive you make the Web app, the more likely you'll encounter bugs in older or less advanced mobile platforms.

How do you deal with adapting the user interfaces of the apps across devices and platforms?

For adapting the app across devices of various sizes and shapes, we've found that using responsive design via the Ionic framework lets us deal quite well with the huge diversity of devices on the market, from smartphones to tablets. There are some issues when dealing with older Web views (esp. on Android), but it remains overall quite manageable.

With regard to the specific native look and feel of each platform, we usually recommend to use a single UI paradigm across platforms, without trying to customize it. Once you start customizing the UI, and taking into account e.g. the inconsistent availability of a hardware back-button between iOS and Android, it makes maintaining the whole app workflow much more difficult.

As an SME, what are your needs in terms of Web standards? Which Web standard(s) are you waiting for specifically?

The arrival of the Push API is very exciting for us! Clearly this will make the offering of browser-based content a lot more attractive to many of our customers, even for the simple mobile portal. We can't wait to see it adopted in more browsers, and we already know we'll use it in our projects in the upcoming few months.

An area where we've found we had to rely on specific Cordova plugins in our hybrid developments is for media playback; for instance, we've found that playing an audio file in loop

using the basic audio capabilities of a Web view wouldn't work well on mobile, and that's a pretty fundamental issue e.g. in a game.

We've also had issues with using Web sockets in some older Web views, but this had more to do with implementation bugs than issues in the standard from what we've determined.

We've struggled for a while with storing data — `localStorage` proved unreliable as the stored data could be wiped out by the OS; but we've now switched to Indexed Database which seems to solve that issue well enough for us.

Any other business related to mobile, Web and standards?

We are an agency for marketing and IT. We help our clients with state-of-the-art technology serving a marketing purpose. This can be CRM system implementation, Web and app development, e-commerce platform development, data management, social media and the like. In most of those projects, and in today's world, systems are no longer independent. A very big issue is the creation of interfaces in order to connect systems creating a technological ecosystem. Standards around communication interfaces between systems are a very important topic. Being in a position to shape and apply standards is one of our motivations of being a W3C member.

Thank you!