Multiplatform Programming

Intelligent Infrastructure Design for the Internet of Things

Antonio Navarro

Index

- References
- Introduction
- What is a platform?
- A model-based approach
- Cross-platform development tools
- Apache Cordova
- Conclusions

References

- (Abernethy, 2011) Abernethy, M. *Just what is Node*.js? https://www.ibm.com/developerworks/ssa/opensource/library/os-nodejs/index.html
- (Baldwin, 2017) Baldwin, Ch. *Cross Platform Mobile Development Tools: Ending the iOS vs. Android Debate.* http://thinkapps.com/blog/development/develop-for-ios-v-android-cross-platform-tools/
- (Clutch, 2017) *Top Cross Platform App Development Tools.* https://clutch.co/app-development/cross-platform
- (Cortes&Navarro, 2017) Cortes, H., Navarro, A. Enterprise WAE: a lightweight UML extension for the characterization of the presentation tier of enterprise applications with MDD-based mockup generation. *IJSEKE,* 27(8)

References

- (Furlan, 2017) Furlan, A. *Cross Platform Mobile App Development Guide 2017.* http://www.businessofapps.com/guide/cross-platform-mobile-app-development/
- (Looper, 2015) Looper, J. *What is a WebView?* https://developer.telerik.com/featured/what-is-a-webview/
- (OMG, 2003) MDA Guide version 1.0.1 http://www.omg.org/news/meetings/workshops/UML_2003_Manual/00-2_MDA_Guide_v1.0.1.pdf
- (Wiki, 2018) Wikipedia. Computing Platform. https://en.wikipedia.org/wiki/Computing_platform

Introduction

- In this topic we will analyze the concept of *platform* and we will see some examples of development in Apache Cordova.
- The objectives of the topic are:
  - Familiarization with the platform concept
  - To know the basic concepts of model-driven architecture.
  - See how Apache Cordova is able to do cross-platform development.
- The practice of this topic will provide an Apache Cordova code to be modified.

What is a platform?

- The first definition of platform that comes to mind is the one provided by the MDA-Guide (OMG, 2003):
  A platform is a set of subsystems and technologies that provide a coherent set of functionality through defined interfaces and patterns of use, which any application supported by the platform can use without concern for the details of how the functionality provided by the platform is implemented

What is a platform?

- The guide itself gives examples of platform types:
  - Generic
    - Object-oriented
    - By batches
    - Data flows
  - Technology specific:
    - CORBA
    - J2EE
  - Vendor specific
    - CORBA: Micro Focus Orbix
    - J2EE: IBM WepSphere, Oracle WebLogic
    - Microsoft .NET

What is a platform?

- This concept is a bit lofty, and we will return to it later.
- Perhaps the Wikipedia concept is closer to what is currently handled (Wiki, 2018):
  A computing platform is the environment in which a piece of software runs.
  It can be the hardware, or the operating system, or even a web browser or other underlying software, as long as the program code runs on it.
  A platform has different levels of abstraction: (i) computer architecture; (ii) operating system; (iii) execution libraries.

What is a platform?

- With such a generic definition, Wikipedia considers almost anything a platform:
  - Hardware, as in the case of embedded systems
  - Browser
  - a scripting language application (such as a spreadsheet)
  - Software frameworks, such as J2EE
  - *Cloud computing* and *Platform as a Service* environments
  - Virtual machines, such as Java
  - Virtualized versions of complete systems

What is a platform?

- In fact, Wikipedia itself considers that a platform can be a combination of hardware, operating system and software framework.
- So, what should we understand by platform?
- Whatever interests us in the context:
  - For example, in web development we can talk about J2EE or .NET.
  - In mobile development, we can talk about Android, iOS or Windows Phone.

A model-based approach

- We will dedicate a couple of slides to the *Model-Driven Architecture* (MDA) approach of the *Object-Management Group* (OMG, 2003).
- One of the main objectives of MDA is to make the specification of a system independent of the implementation details.
- MDA provides a framework that allows:
  - Give a specification of a system independent of implementation details.
  - Specify implementation platforms
  - Choose one of these platforms
  - Transform the system specification into an implementation platform.

A model-based approach

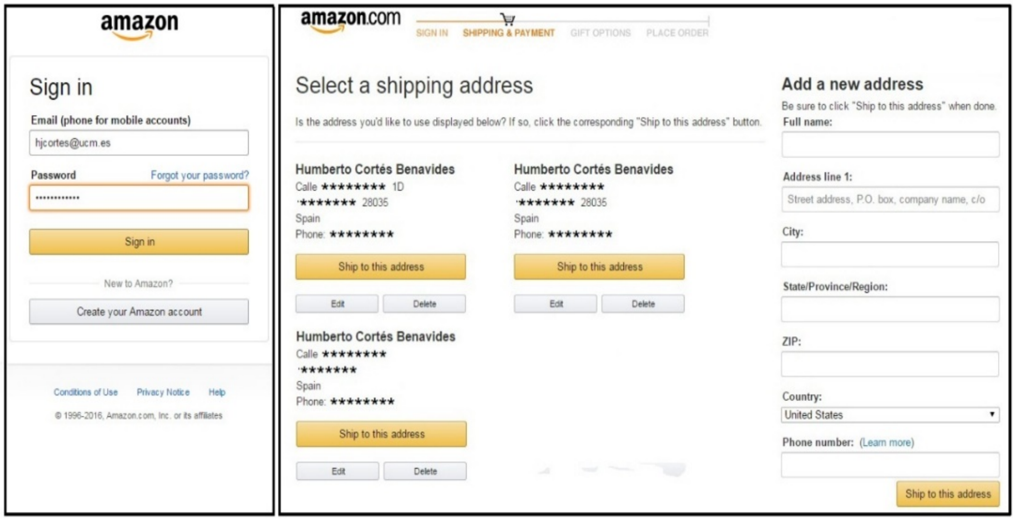
- Independence is achieved using different types of models
- These models allow to provide different system specifications from different points of view
- These models are:
  - CIM (*Computation Independent Model*): system characterization from the domain point of view (domain/business model)
  - PIM (*Platform Independent Model*): design view omitting platform-specific details
  - PSM (*Platform Specific Model*): design view considering specific details of a platform.

A model-based approach

- A fundamental component of the MDA approach is the use of *transformations*, especially in the transition from PIM to PSM.
- To define transformations between models, transformation rules are defined between models of models, i.e. between *metamodels.*
- For example, we have *E-WAE* (Cortés&Navarro, 2017).
    - PIM: E-WAE, notation for characterizing presentation layer enterprise applications
    - PSMs: WAE4JSF, WAE4.NET: notations for characterizing presentation layer JSF and ASP .NET MVC enterprise applications
    - Transformations: PIM -> PSMs -> code

A model-based approach

- Example authentication and address on Amazon:



Sign in

Email (phone for mobile accounts)

hjcortes@ucm.es

Password Forgot your password?

·············

New to Amazon?

Conditions of Use Privacy Notice Help

© 1996-2016, Amazon.com, Inc. or its affiliates

Select a shipping address

Is the address you'd like to use displayed below? If so, click the corresponding "Ship to this address" button.

Humberto Cortés Benavides
Calle ★★★★★★★★ 1D
'★★★★★★★ 28035
Spain
Phone: ★★★★★★★★

Humberto Cortés Benavides
Calle ★★★★★★★★
'★★★★★★★ 28035
Spain
Phone: ★★★★★★★★

Humberto Cortés Benavides
Calle ★★★★★★★★
'★★★★★★★
Spain
Phone: ★★★★★★★★

Add a new address

Be sure to click "Ship to this address" when done.

Full name:

Address line 1: Street address, P.O. box, company name, c/o

City:

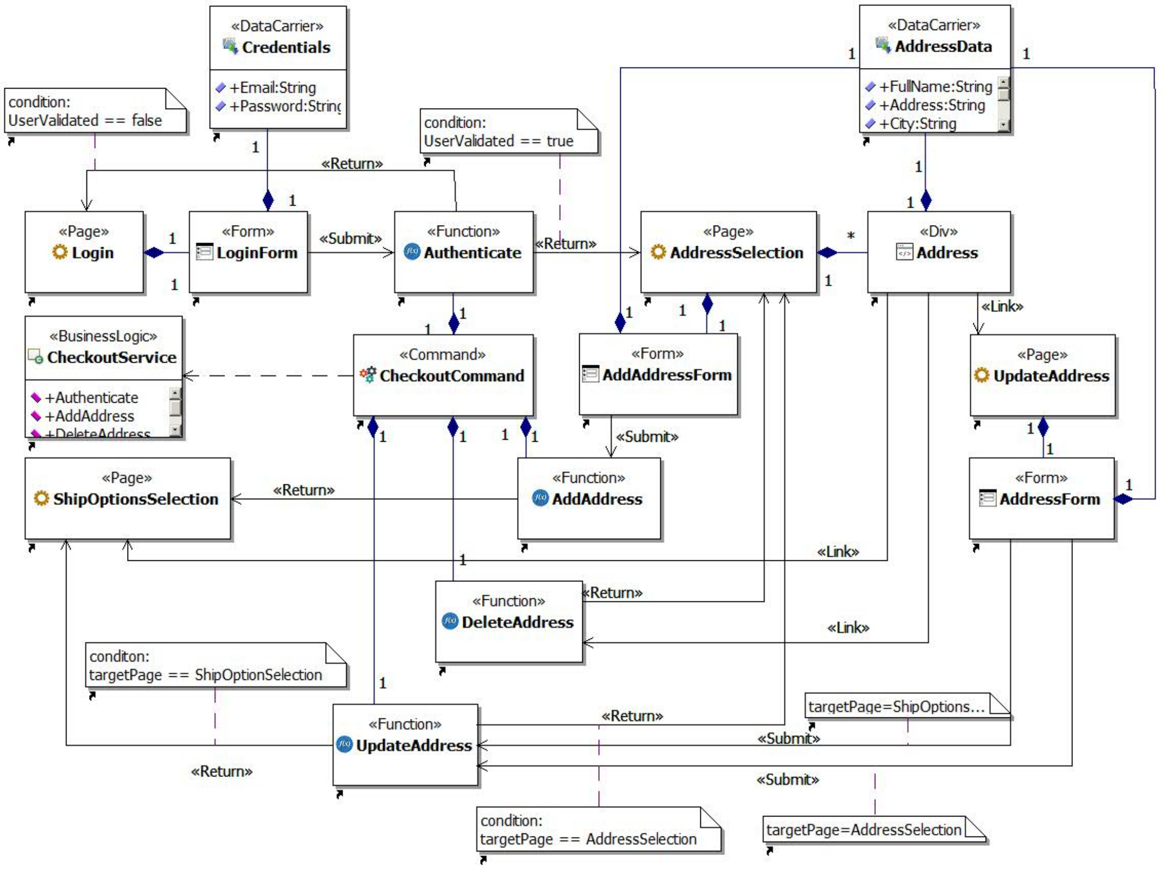
State/Province/Region:

ZIP:

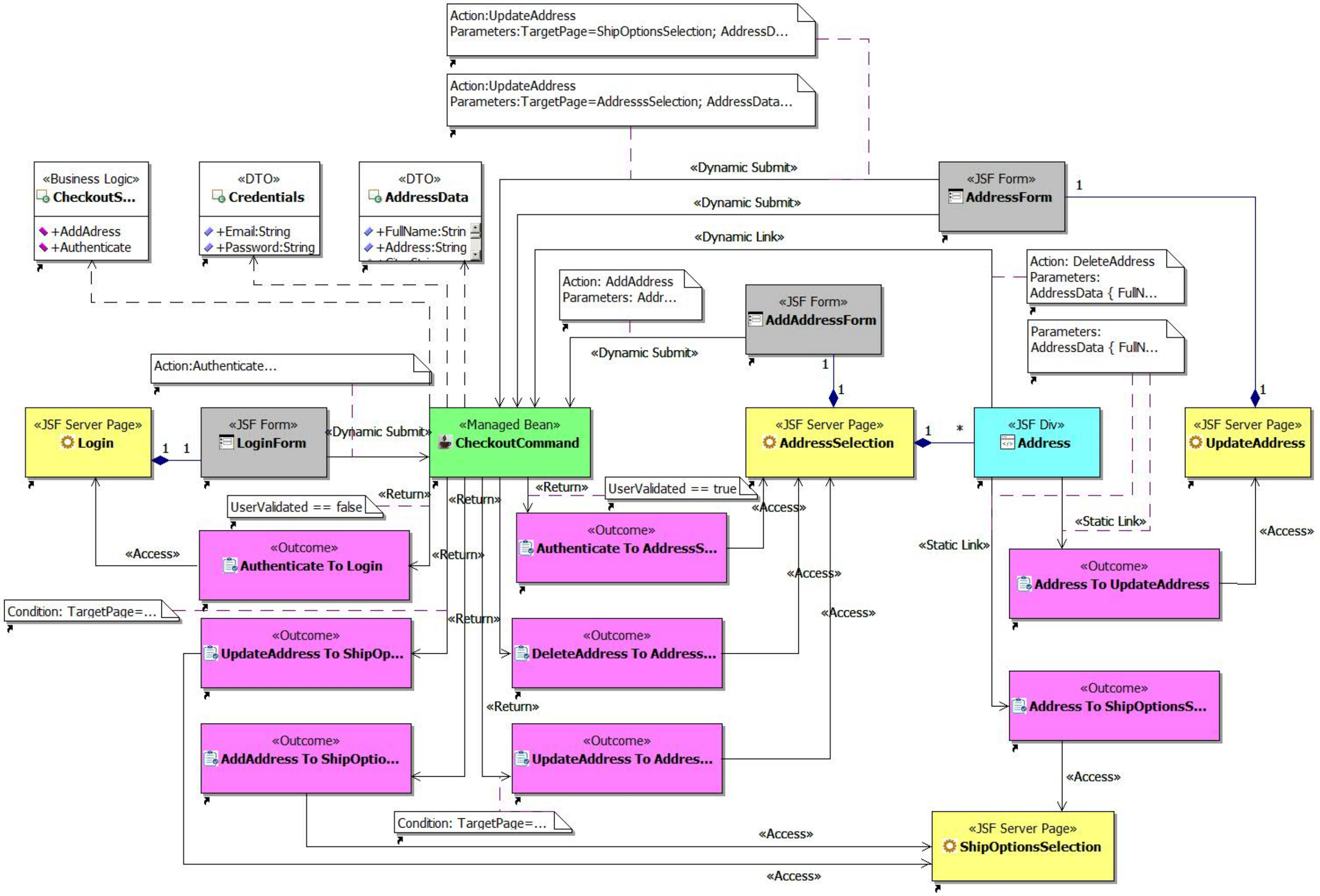
Country: United States

Phone number: (Learn more)

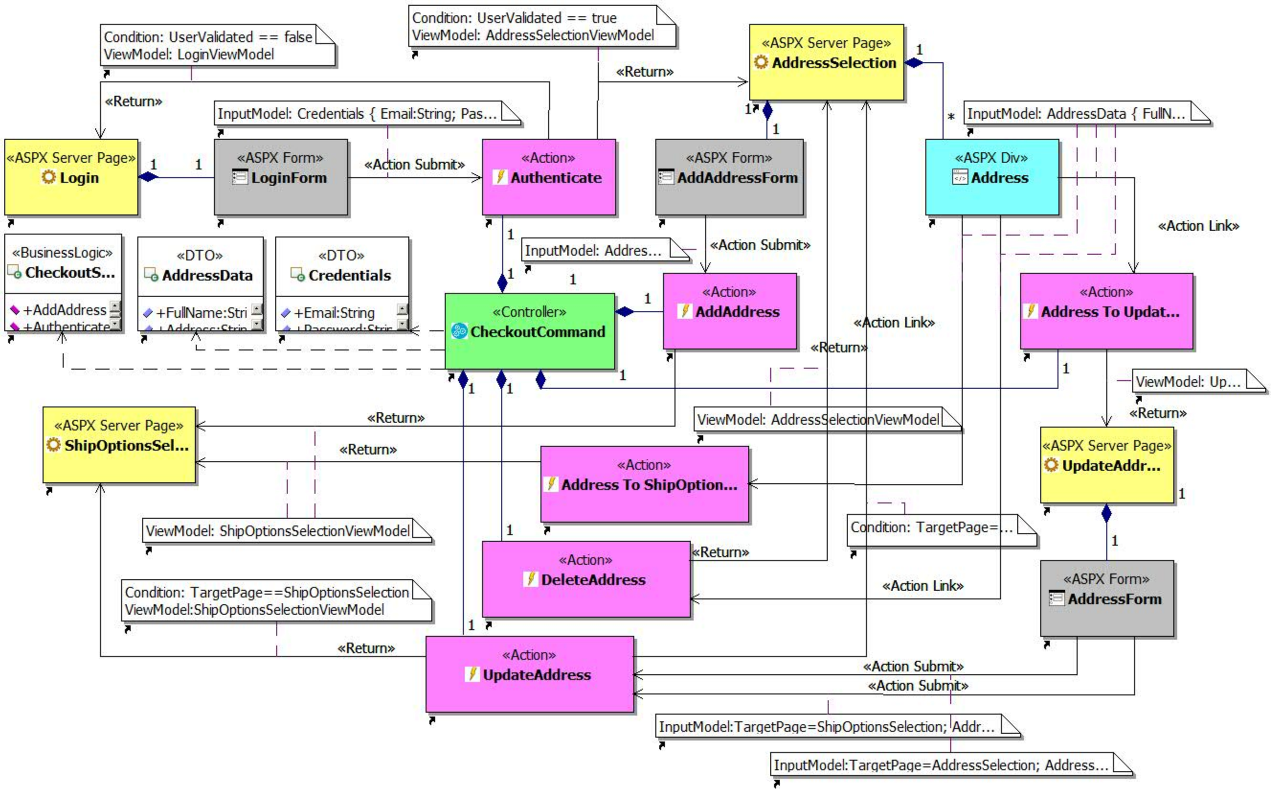
Authentication and shipping address selection in Amazon



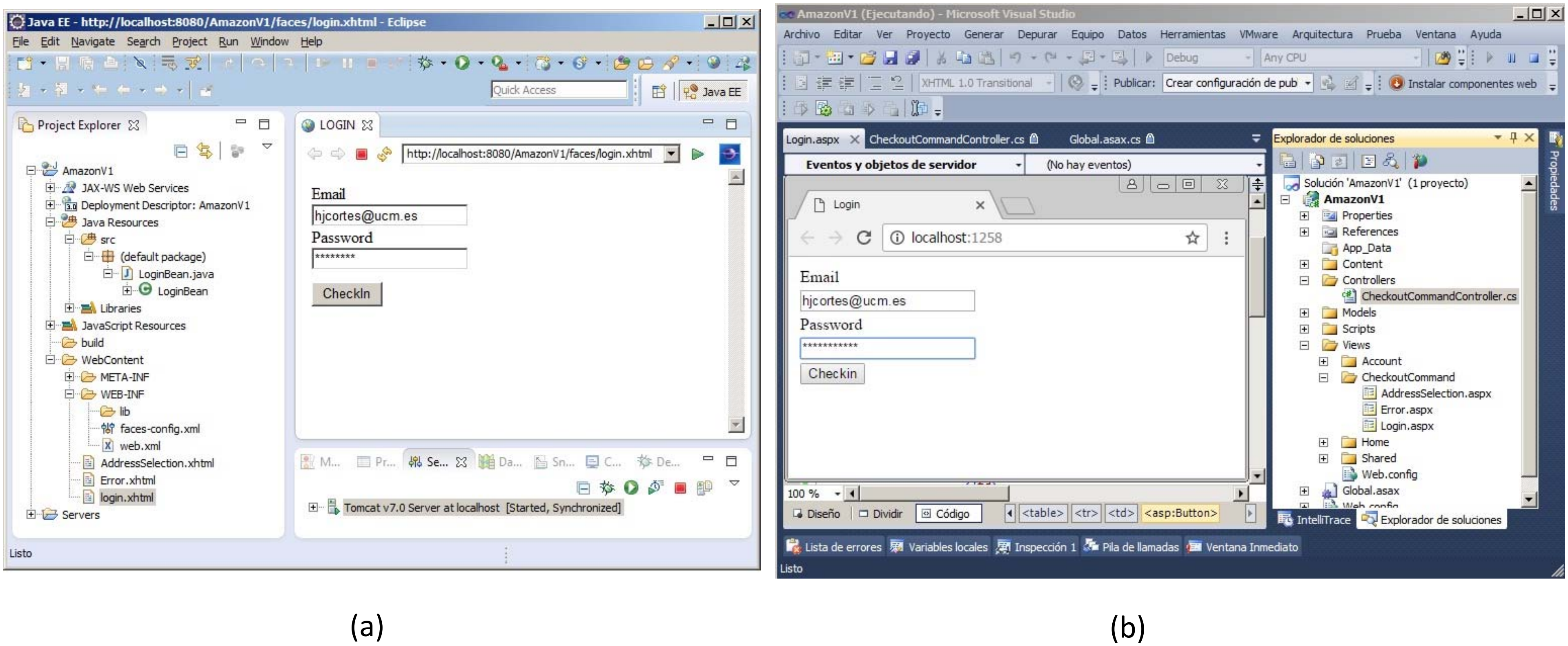
E-WAE model for authentication and address selection on Amazon



WAE4JSF model for Amazon authentication and address selection generated from E-WAE



WAE4.NET model for Amazon authentication and address selection generated from E-WAE



(a)

Java EE - http://localhost:8080/AmazonV1/faces/login.xhtml - Eclipse

LOGIN — http://localhost:8080/AmazonV1/faces/login.xhtml

Email: hjcortes@ucm.es

Password: ********

CheckIn

Tomcat v7.0 Server at localhost [Started, Synchronized]

(b)

AmazonV1 (Ejecutando) - Microsoft Visual Studio

Login — localhost:1258

Email: hjcortes@ucm.es

Password: ***********

Checkin

Mockups and code generated in Eclipse (a) and in MS Visual Studio (b) from WAE4x models

Cross-platform development tools

- MDAs approaches are elegant, well defined... and quite complex.
- MDA approaches are based on defining an abstract model and particularizing it into models for different platforms, which are finally translated into code.
- Another lower level option is to use a language that can be executed on different platforms, and in this sense, interpreted languages are a good option for multiplatform programming.
- Let's see Java.... Or not

Cross-platform development tools

- Java is an excellent programming language, whether for enterprise, web or mobile applications:

- However, we are going to see here something more focused on what is currently known as *cross-platform development.*
- This development is particularly relevant in the development of applications for cell phones and video games.

Cross-platform development tools

- Some cross-platform development tools are (Baldwin, 2017; Clutch 2017):
  - Cocos2d: 2D game development for C++, JavaScript, C#, Objective-C, Python
  - Corona: multiplatform 2D game development
  - Unity 3D: game development in UnityScript, C# or boo and export to various platforms: iOS, Android, Windows, Web, Playstation Xbox, Wii, Linux
  - 5App: Security-based HTML5 and JavaScript development translated to Android and iOS
  - Adobe PhoneGap: HTML5, CSS and JavaScript translated to mobile platforms (based on Apache Cordova)
  - Alpha Anywhere: Development based on C#, JavaScript, Xbasic, VB.NET-supported languages (VB, Cobra, C++) and translation to mobile platforms

Cross-platform development tools

  - AngularJS (Google): HTML5, CSS and JavaScript translated to mobile platforms
  - Appcelerator Titanium: Reuses JavaScript code common to mov platforms.
  - ionic: HTML5, CSS and JavaScript translated to mobile platforms
  - Qt: subscription-based service for C++ translated to mov platforms.
  - Sencha: HTML5-based and translations to mobile platforms
  - Xamarin (MS): Ruby or C# code translated to mobile platforms

- (Furlan, 2107), gives another view, with a short summary of programming languages

Cross-platform development tools

- We will focus on Apache Cordova
- However, I ask myself a question: how long will the market support so many platforms, so many programming languages and so many development applications?

Apache Cordova

- The underlying idea behind Apache Cordova is quite simple:
  - Build a mobile application based on the use of HTML5 and CSS for the presentation layer and JavaScript for the *business layer.*
  - Apache Cordova provides generic libraries for handling mobile-specific functions (accelerometer, camera, geolocation, etc.). These libraries are called *Core Plugins*
  - HTML and CSS run in a web browser window without the address bar on mobile: the *WebView* (Looper, 2015).
  - JavaScript code that can be interpreted by a browser, is maintained
  - Calls to *core plugins* are translated into platform-specific library calls.

Apache Cordova

Platform-independent application

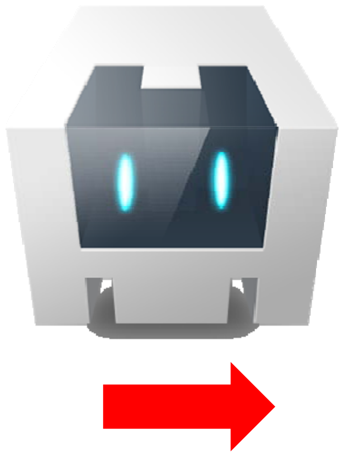
HTML5

CSS

JavaScript

+

Calling Apache Cordova Core Plugins



Apache Cordova Engine

+

Platform Selection (iOS)

Platform-specific application (iOS)

WebView

JavaScript

+

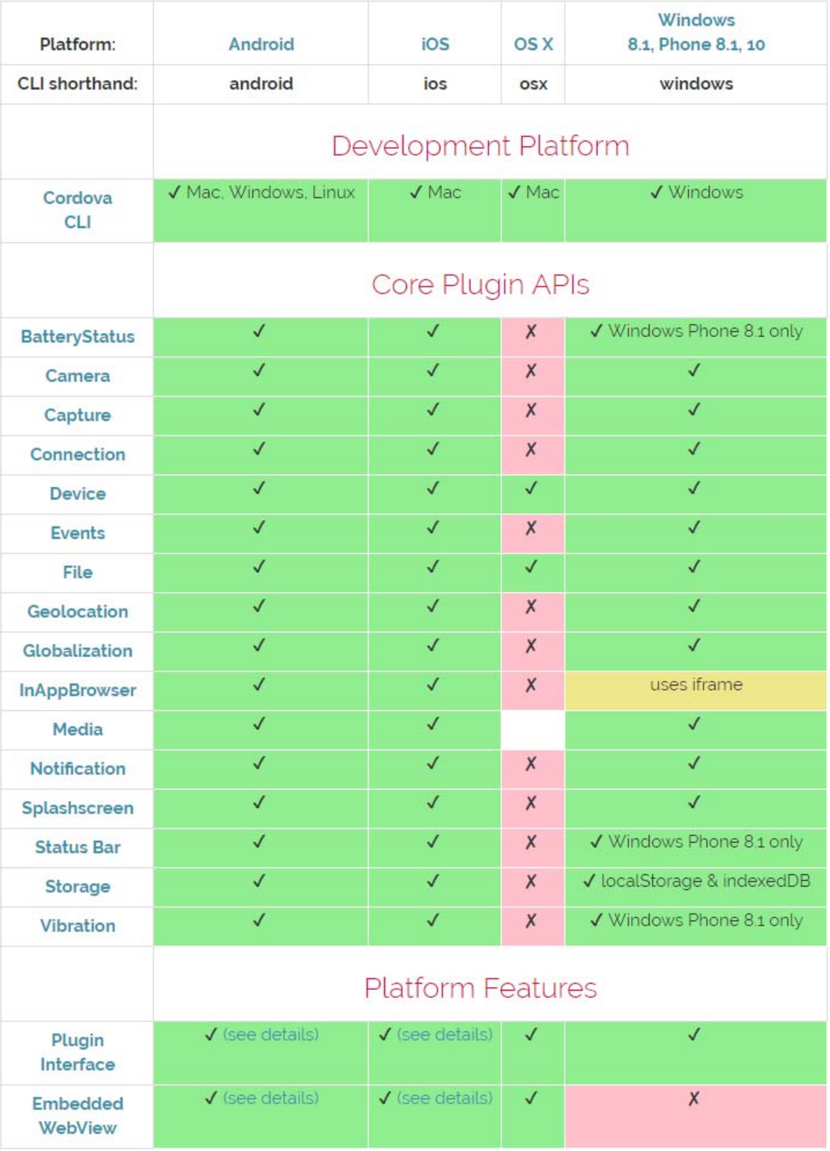
Calls to iOS libraries

Apache Cordova

- In a way, it is a similar concept to MDA, but the transformations are defined between programming languages, not between design metamodels.
- It is simpler, but we do not have a design model.

Apache Cordova

- The platforms supported by Apache Cordova are:
  - Android
  - ~~BlackBerry 10~~
  - iOS
  - OS X (very limited)
  - ~~Ubuntu~~
  - Windows Phone 8.1
  - Windows 8.1, 10
- Apache Cordova can work for both cross-platform and platform-specific development.

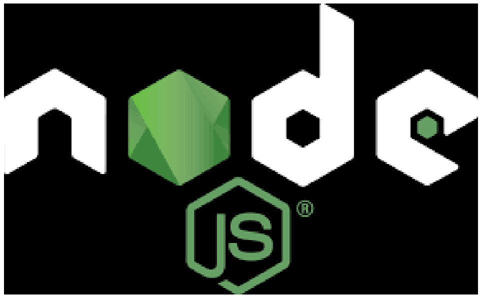


Platform:	Android	iOS	OS X	Windows 8.1, Phone 8.1, 10
CLI shorthand:	android	ios	osx	windows
	Development Platform			
Cordova CLI	✓ Mac, Windows, Linux	✓ Mac	✓ Mac	✓ Windows
	Core Plugin APIs			
BatteryStatus	✓	✓	✗	✓ Windows Phone 8.1 only
Camera	✓	✓	✗	✓
Capture	✓	✓	✗	✓
Connection	✓	✓	✗	✓
Device	✓	✓	✓	✓
Events	✓	✓	✗	✓
File	✓	✓	✓	✓
Geolocation	✓	✓	✗	✓
Globalization	✓	✓	✗	✓
InAppBrowser	✓	✓	✗	uses iframe
Media	✓	✓		✓
Notification	✓	✓	✗	✓
Splashscreen	✓	✓	✗	✓
Status Bar	✓	✓	✗	✓ Windows Phone 8.1 only
Storage	✓	✓	✗	✓ localStorage & indexedDB
Vibration	✓	✓	✗	✓ Windows Phone 8.1 only
	Platform Features			
Plugin Interface	✓ (see details)	✓ (see details)	✓	✓
Embedded WebView	✓ (see details)	✓ (see details)	✓	✗

Apache Cordova platform support

Apache Cordova

- Apache Cordova runs on *node.js*
  - node.js is a server-side JavaScript interpreter that uses V8 JavaScript as its underlying engine (Abernethy, 2011).
  - V8 JavaScript is the engine that Google uses with its Chrome browser.
  - Although node.js is a server-side interpreter, it can be used locally.
  - It is not intended to be deployed as easily as Apache Web Server or Tomcat.



Apache Cordova

- Building an application with Cordova is very simple: https://cordova.apache.org/docs/en/latest/guide/cli/index.html
  - You need to have it installed:
    - Android SDK
    - Android Studio, with an Android Virtual Device (AVD) created with **no spaces in the name**
  - An application is created from the command line: `$ cordova create hello`
  - The desired platforms are added: `$ cordova platform add android`
  - Plugins are added: `$ cordova plugin add cordova-plugin-camera`
  - HTML, CSS and JavaScript files are edited externally.
  - The application is built (with or without platform): `$ cordova build android`
  - The application is tested: `$ cordova emulate android`

Apache Cordova

- Example:

```
< html>
................................
  < body>
    < div>
      <h1>Say hello</h1>.
      < input type="button" id="sayHello" value="greet"/>
    </div>
    <div>
      <h1>Photo</h1>
      <input type="button" id="takePhoto" value="photo" />
      < div id="thePhoto">
    </div>
   <script type="text/javascript" src="cordova.js"></script>
   < script type="text/javascript" src="js/index.js"></script>
  </body>
</html>
```

Apache Cordova

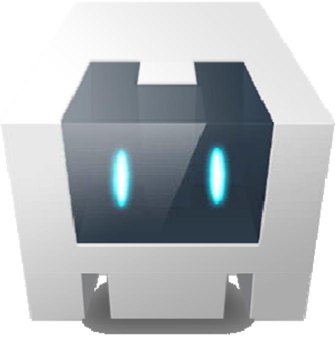
```
document.addEventListener("deviceready", onDeviceReady, false);
function onDeviceReady() {

   document.getElementById('sayHello'). onclick = function () {
                                                  alert("hello"); }
   document.getElementById('takePhoto'). onclick = function () {
navigator.camera.getPicture( function(imageURI){
                         var lastPhoto=document.getElementById("thePhoto");
lastPhoto.innerHTML="< img src='" + imageURI + "' style='width=60%;'/>" } )
}    }
```

Apache Cordova

```
body {
      color:darkblue;
      background-color:DeepSkyBlue; }
```

Apache Cordova

- Demo



Conclusions

- The concept of platform in computer science is quite broad.
- If you want to have cross-platform design and programming, the MDA/MDD (Model-Driven Develpoment) approach makes the most sense.
- MDA/MDD can be very demanding, which is why simpler solutions are appearing.
- Multiple platforms and tools: until when?
- Apache Cordova enables cross-platform programming for mobile devices based on HTML5, CSS, JavaScript, and HTML5.

Multiplatform Programming

Intelligent Infrastructure Design for the Internet of Things

Antonio Navarro