



Universidad Católica “Nuestra Señora de la Asunción”  
Facultad de Ciencias y Tecnología  
Ingeniería Informática



# Un enfoque MDD para el desarrollo de aplicaciones móviles nativas enfocadas en la capa de datos

**Tutor:** Ing. Magalí González

**Co-tutor:** Ph.D. Luca Cernuzzi

**Profesor Colaborador:** M.Sc. Nathalie Aquino

El presente Proyecto Final de Carrera, ha sido desarrollado con el apoyo financiero del Consejo Nacional de Ciencia y Tecnología (CONACYT, Paraguay) en el marco del proyecto denominado “Mejorando el proceso de desarrollo de software: propuesta basada en MDD” (14-INV-056).

## Outline

- Aplicaciones móviles y algunas de sus problemáticas
- El desarrollo de aplicaciones móviles
- Objetivos
- MoWebA y su extensión para móviles
- Experiencia de validación
- Conclusiones y Trabajos Futuros

## Outline

- **Aplicaciones móviles y algunas de sus problemáticas**
- El desarrollo de aplicaciones móviles
- Objetivos
- MoWebA y su extensión para móviles
- Experiencia de validación
- Conclusiones y Trabajos Futuros

# Teléfonos móviles

- Antecedentes



## Select Digital Usage Metrics Worldwide, 2015 & 2019 % of population

### Mobile phone users



### Internet users



### Mobile phone internet users



### Smartphone users



### Tablet users



■ 2015

■ 2019

Source: eMarketer, July 2015

194038

www.eMarketer.com



# Aplicaciones móviles

## - Características

- Manejan sensores y hardware específico del dispositivo móvil



- Trabaja con recursos limitados, como la memoria del dispositivo, la batería y la conectividad, entre otros



- La fragmentación
- La persistencia y la naturaleza transitoria

La fragmentación de la persistencia

# Problemáticas

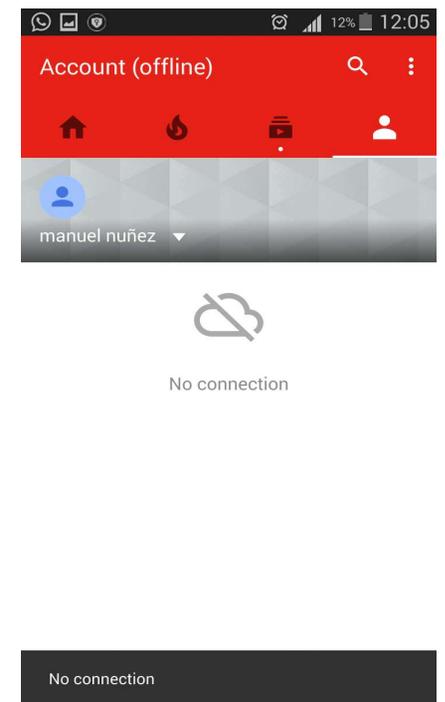
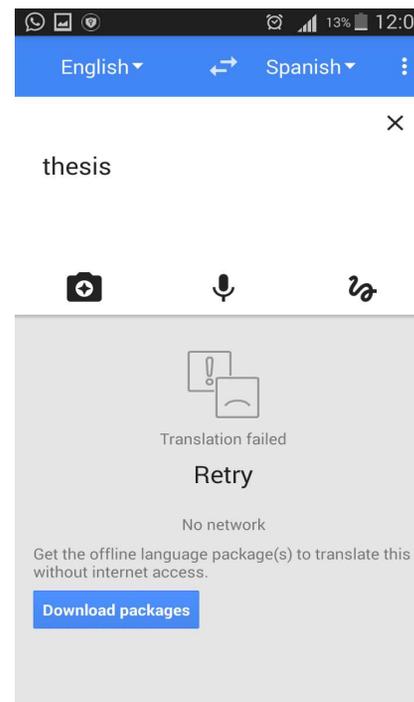
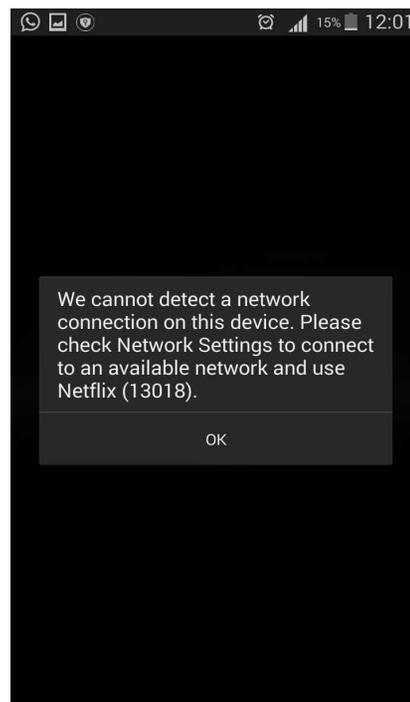
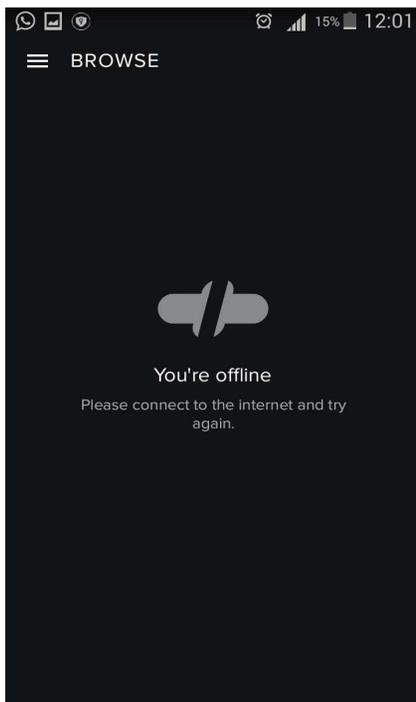
- Fragmentación



# Problemáticas

- Persistencia y la naturaleza transitoria

## ■ Modo “sin conexión”



## Problemáticas

### - Persistencia y la naturaleza transitoria

- Opciones de almacenamiento más populares



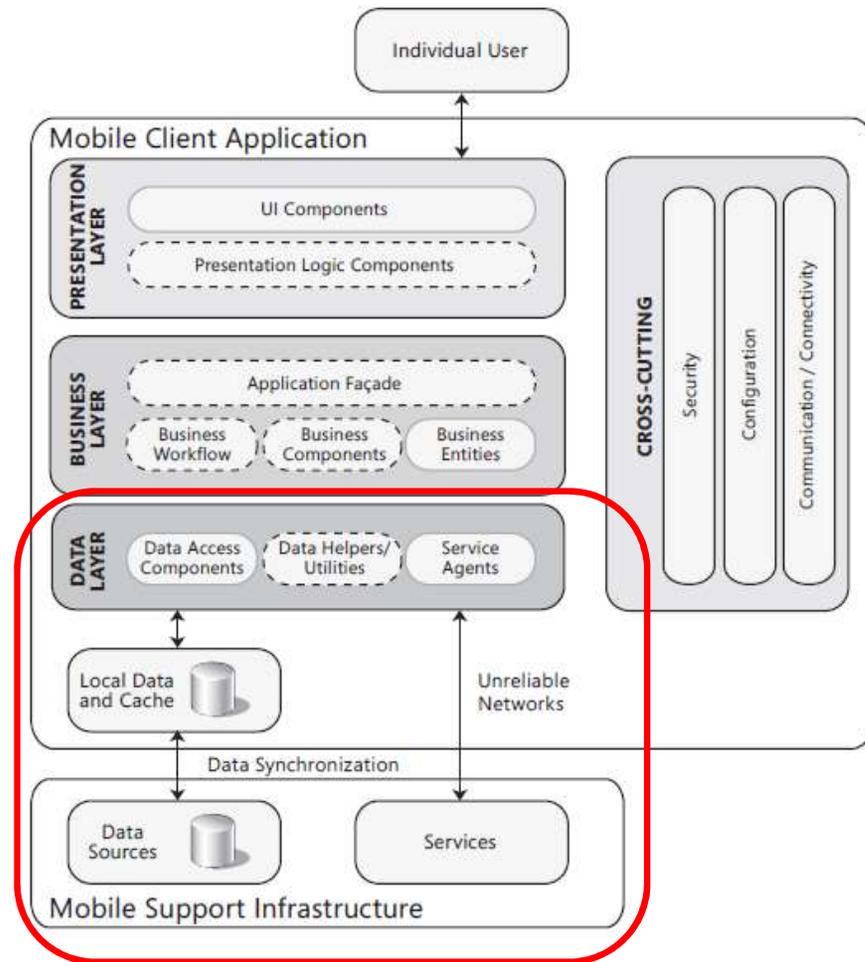
- Mecanismos de persistencia



# Problemáticas

- Persistencia y la naturaleza transitoria

- Capa de datos  
(*Microsoft Architecture Application Guide*)



## Problemáticas

- La fragmentación de la persistencia

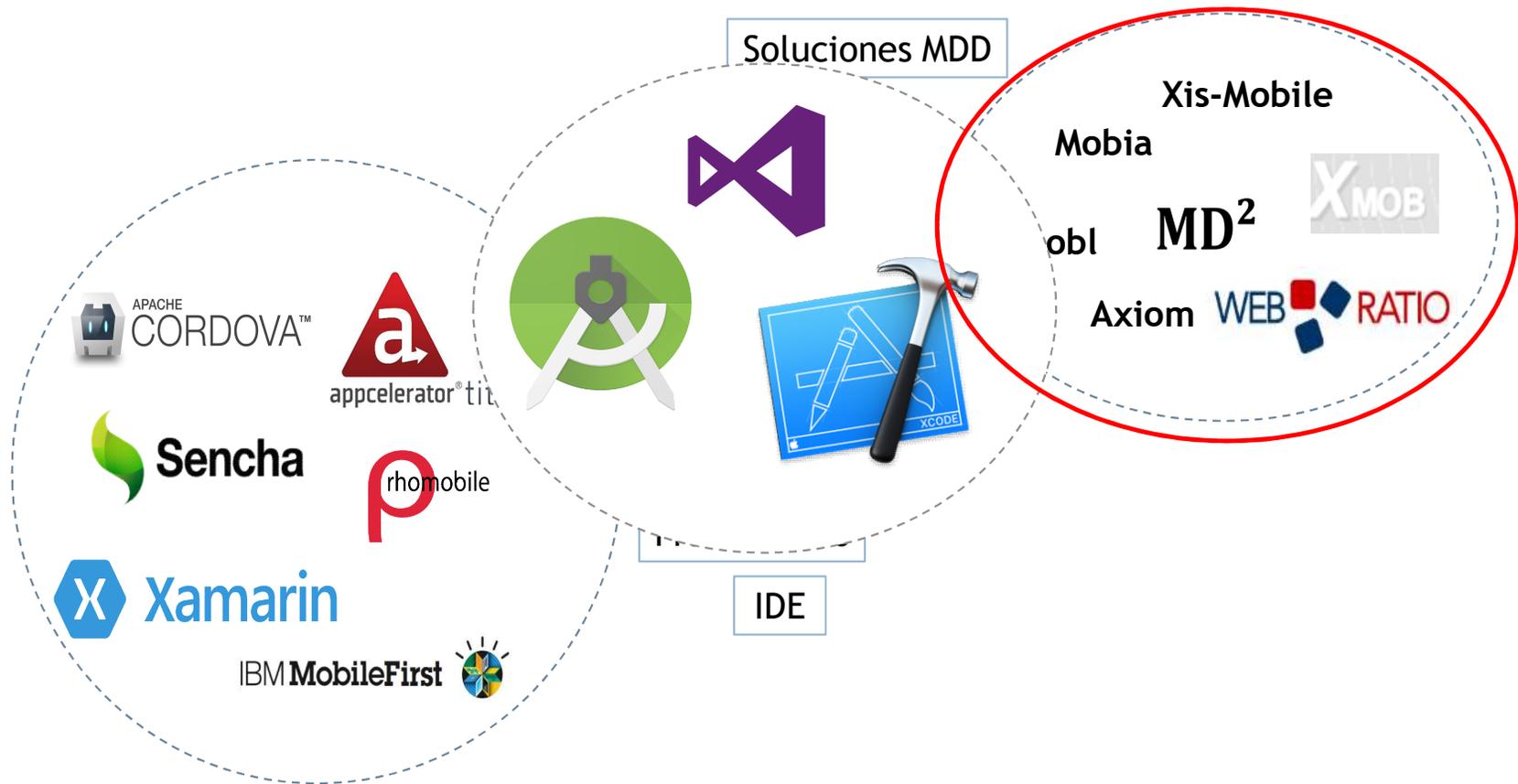
Cómo afecta la fragmentación a la persistencia?

Plataformas	Par (clave, valor)	Manejo de archivos locales	Soporte para almacenamiento externo	Base de datos
Android	SharedPreferences, Bundles	Java Files Stream	Sí	Content Provider, SQLite
iOS	NSUserDefaults, Property lists	NSFileManager	Sí	Core Data, SQLite
Windows Phone	Isolated Storage Settings	Isolated Storage File System	Sí	Isolated Storage, SQLite

## Outline

- Aplicaciones móviles y algunas de sus problemáticas
- El desarrollo de aplicaciones móviles
- Objetivos
- MoWebA y su extensión para móviles
- Experiencia de validación
- Conclusiones y Trabajos Futuros

# Alternativas para el desarrollo de aplicaciones móviles



# Desarrollo Móvil Dirigido por Modelos

## - Propuestas MDD para el desarrollo de aplicaciones móviles

Propuestas basadas en MDD para el desarrollo de aplicaciones móviles															
Propuesta	Aspectos considerados	En qué se basa para el modelado	Lenguaje de modelado	Tipo de lenguaje de modelado	Utiliza estándar OMG	Plataformas posibles de destino	Utiliza MVC	Generación de código nativo	Generación completa de la aplicación	Tipo de aplicación generada	Presenta evaluación de su propuesta	Sus evaluaciones dieron resultado positivos	Tiene en cuenta la persistencia en el modelado	Soporte modalidad "sin conexión"	Permite conexión con bases de datos
P1	Estructural y lógica de negocios	Perfil UML	UML	Gráfico	UML	Android y Windows Phone <sup>8</sup>	No	Sí	Sí	Soporta el desarrollo de todo tipo de aplicaciones	Sí	Sí	Sí <sup>9</sup>	No especificado	No especificado
P2 y P3	GUI	DSL	UML y Technology Neutral DSL	Textual	MDA	Android <sup>8</sup>	No	Sí	No	No especificado	Sí	Sí	No aplica	No aplica	No aplica
P4	GUI y lógica de negocios	EMF	EMF	Gráfico	No	Android y iOS	Sí	Sí	Sí	Aplicaciones de negocio dirigidas por datos	Sí	Sí	No especificado	Sí	No especificado
P5	GUI y lógica de negocios	IFML	IFML para móvil	Gráfico	IFML	Android y iOS <sup>8</sup>	No	No, native wrapper	Sí	Aplicaciones móviles basadas en tecnologías web <sup>10</sup>	Sí	Sí	No aplica	No aplica	No especificado
P6	Estructural, lógica de negocios y GUI	IFML	IFML para móvil (interfaz) y Flujo Visual (backend)	Gráfico	IFML	Android y iOS	No	No, native wrapper	Sí	Aplicaciones móviles basadas en tecnologías web <sup>10</sup>	No	No aplica	No especificado	Sí	Sí
P7	No especificado	λ calculus	MobDSL	Textual	No	Android y iOS	No especificado	No, interpreta el código a través de la máquina virtual	Sí	Aplicaciones dirigidas por datos	Sí	Sí	No especificado	No especificado	Sí <sup>11</sup>
P8	Estructural, lógica de negocios y GUI	Perfil UML llamado Xis	Xis-Mobile DSL	Gráfico	No	Android, iOS y Windows Phone <sup>8</sup>	No	Sí	Sí	Aplicaciones dirigidas por datos	Sí	Sí	Sí <sup>12</sup>	No especificado	No especificado
P9	Datos, estructural, lógica de negocios, y GUI	DSL	RAPPT DSL	Gráfico/Textual <sup>13</sup>	No	Android	Sí	Sí	Sí	Aplicaciones dirigidas por datos	Sí	Sí	No especificado	No especificado	Sí
P10	Estructural, lógica de negocios y GUI	Perfil UML	UML	Gráfico	MDA	Android y Windows Phone.	Sí	Sí	Sí	Aplicaciones sencillas, que no requieran gráficos avanzados	Sí	Sí	No especificado	No especificado	No especificado
P11	GUI	MIM	MIM DSL	Gráfico	No	No especificado	No	No especificado	No <sup>14</sup>	Aplicaciones dirigidas por datos	Sí	Sí	No aplica	No especificado	Sí
P12	Requerimientos y lógica de negocios, y GUI	DSL	AXIOM DSL	Textual	MDA	Android y iOS <sup>8</sup>	Sí	Sí	Sí	Soporta el desarrollo de todo tipo de aplicaciones	Sí	Sí	No especificado	No especificado	No especificado
P13	Datos, GUI y lógica de negocios	DSL	Xmob DSL(PIM); perfil UML(PSM)	Textual	MDA	Android, iOS y .NET	Sí	Sí	Sí	Aplicaciones dirigidas por datos	No	No aplica	Sí <sup>15</sup>	No especificado	Sí
P14 y P15	Datos, GUI y lógica de negocios	DSL	MD2 DSL	Textual	No	Android y iOS	Sí	Sí	Sí <sup>16</sup>	Aplicaciones dirigidas por datos	Sí	Sí	Sí <sup>17</sup>	No especificado	No especificado
P16	GUI	UML	UML	Gráfico	MDA	Android y Blackberry	No	No especificado	No <sup>14</sup>	No especificado	Sí	Sí	No aplica	No aplica	No aplica
P17	Datos, GUI, lógica de negocios y recursos de hardware	Perfil UML	UML	Gráfico	UML	Windows Phone 7	Sí	No especificado	No especificado	No aplica	Sí	Sí	Sí <sup>18</sup>	No especificado	No especificado
P18	Datos, GUI, lógica de negocios y recursos de hardware	Perfil UML	UML	Gráfico	UML	Android	Sí	No especificado	No especificado	No aplica	Sí	Sí	Sí <sup>19</sup>	No especificado	No especificado

Cuadro comparativo de las propuestas MDD resultantes del estudio de la literatura

## Desarrollo Móvil Dirigido por Modelos

- Análisis de la persistencia en las propuestas MDD para el desarrollo de aplicaciones móviles

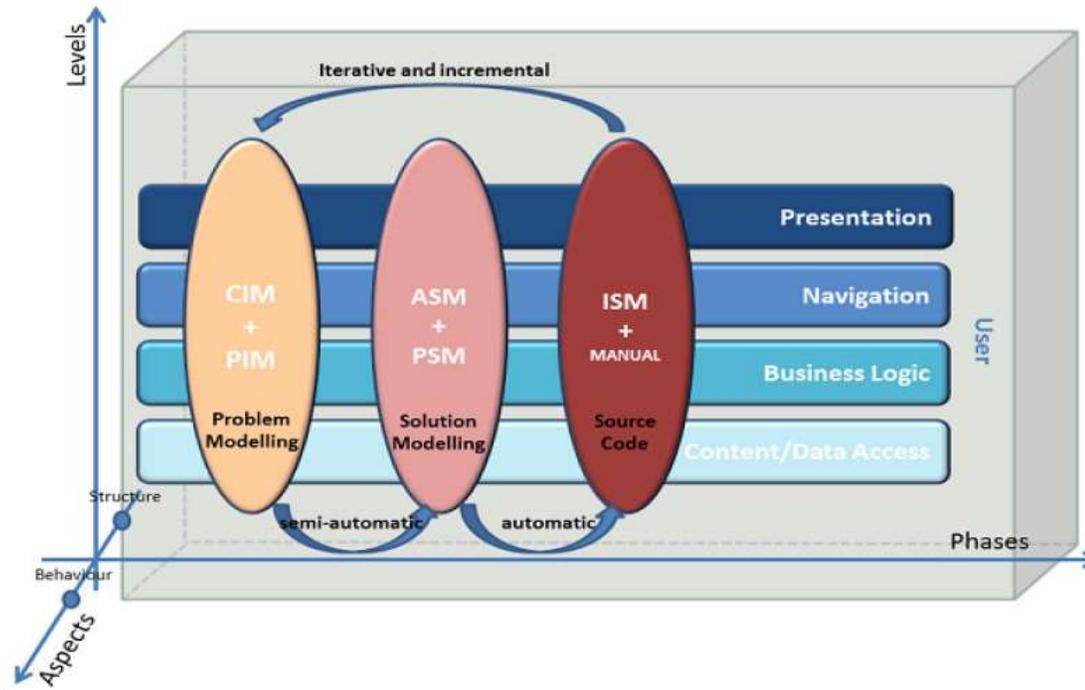
### Destacamos:

- Ninguna de las propuestas adopta todos los aspectos considerados de la persistencia.
- Poca adopción de MDA

### Se necesita:

- Cubrir conceptualmente la persistencia en el desarrollo de aplicaciones móviles.
- Elementos específicos para definir las distintas opciones y mecanismos de persistencia en el modelado.
- Permitir el almacenamiento de datos incluso sin conexión de red.
- Considerar la conexión con fuentes remotas.

# Model Oriented Web Approach (MoWebA)

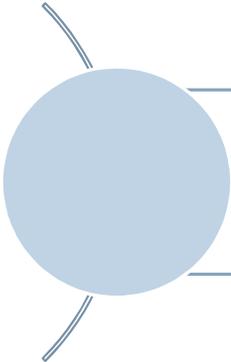


## Outline

- Aplicaciones móviles y algunas de sus problemáticas
- El desarrollo de aplicaciones móviles
- **Objetivos**
- MoWebA y su extensión para móviles
- Experiencia de validación
- Conclusiones y Trabajos Futuros

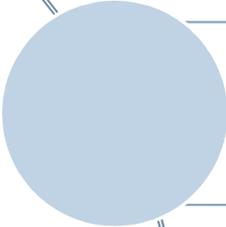
## Objetivo General

---

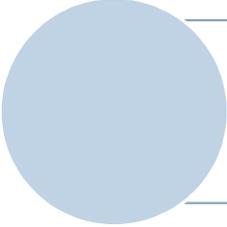


Definir un enfoque MDD, mediante la extensión de MoWebA, para el desarrollo de aplicaciones móviles nativas enfocadas en la capa de datos

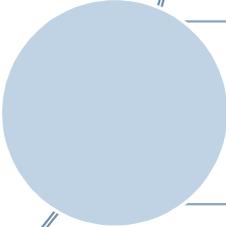
## Objetivos Específicos



Analizar las propuestas MDD para el desarrollo de aplicaciones móviles, observando los aspectos de persistencia utilizados.



Extender MoWebA para el desarrollo de aplicaciones móviles, considerando la capa de datos mediante metamodelos y perfiles ASM, y el establecimiento de reglas de transformación para tal propósito.



Realizar un análisis de la propuesta mediante la valoración de resultados aplicados a un caso de prueba.

## Outline

- Aplicaciones móviles y algunas de sus problemáticas
- El desarrollo de aplicaciones móviles
- Objetivos
- **MoWebA y su extensión para móviles**
- Experiencia de validación
- Conclusiones y Trabajos Futuros

## MoWebA Mobile

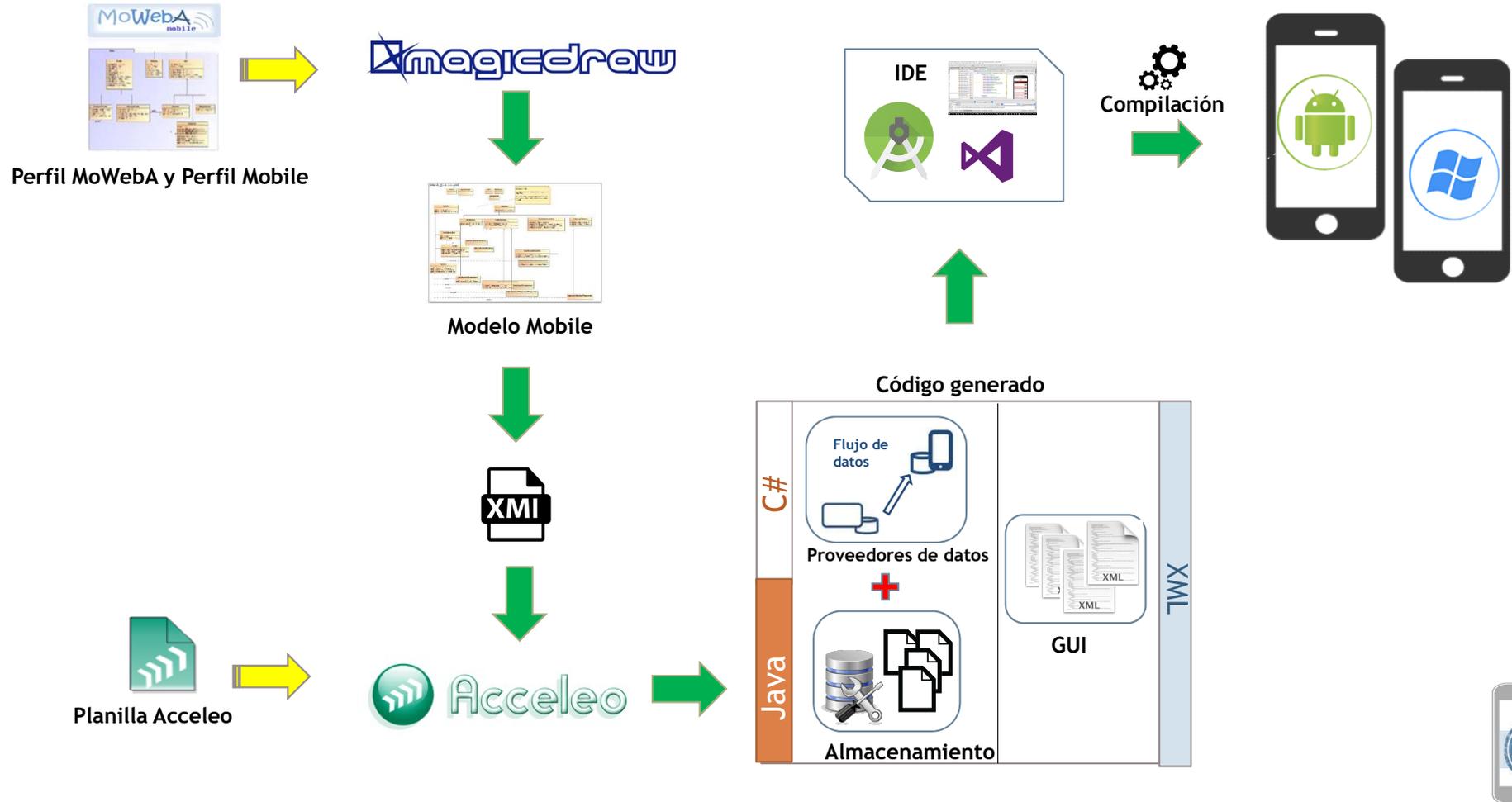
- Una extensión de MoWebA para el desarrollo de aplicaciones móviles

- ASM móvil, parte de la extensión del PIM de MoWebA
- Desarrollo de aplicaciones móviles nativas enfocadas en la capa de datos:
  - Persistencia local de datos: permite identificar qué datos van a almacenarse en el teléfono, y qué tipo de almacenamiento se utilizará.
  - Proveedores de datos: permite diseñar los proveedores de datos externos, internos e interoperabilidad con otras aplicaciones.



# MoWebA Mobile

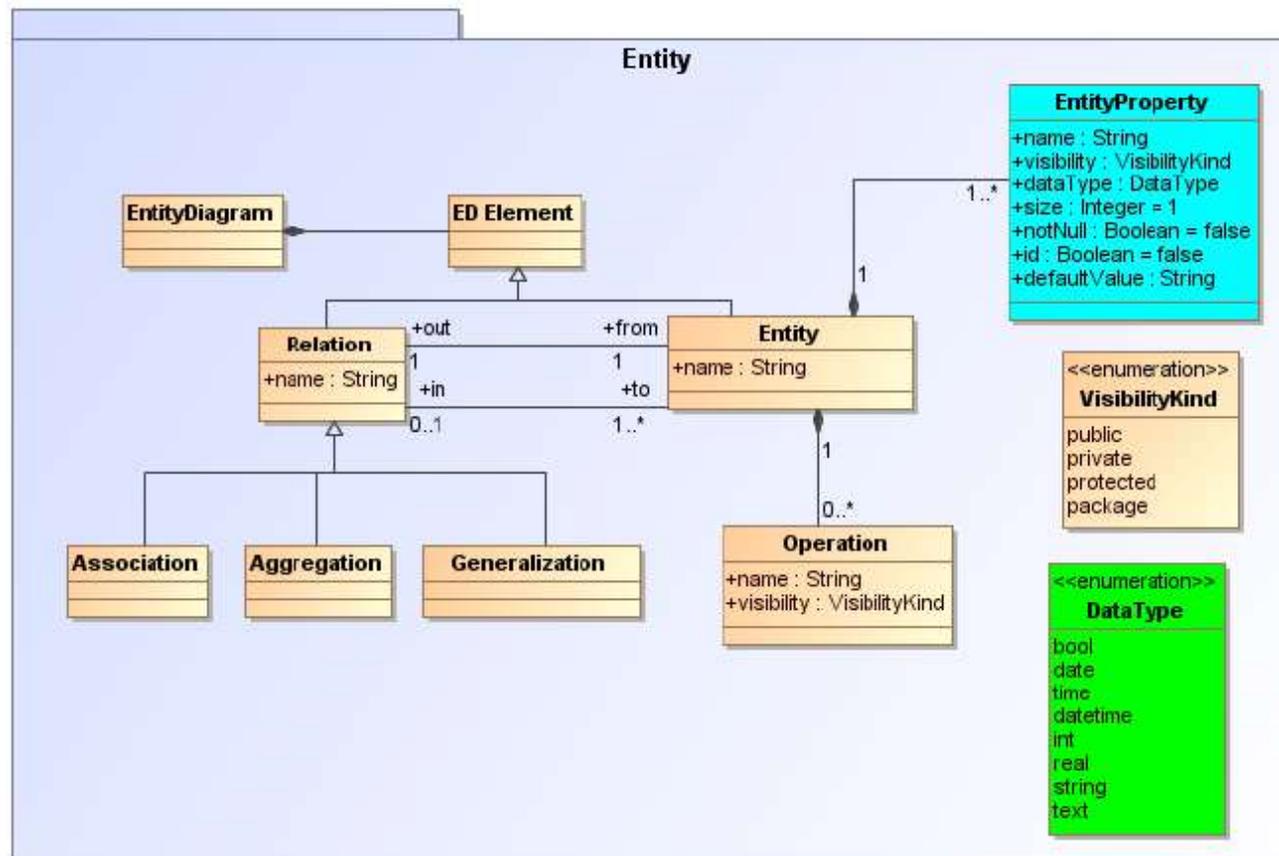
- Proceso de desarrollo





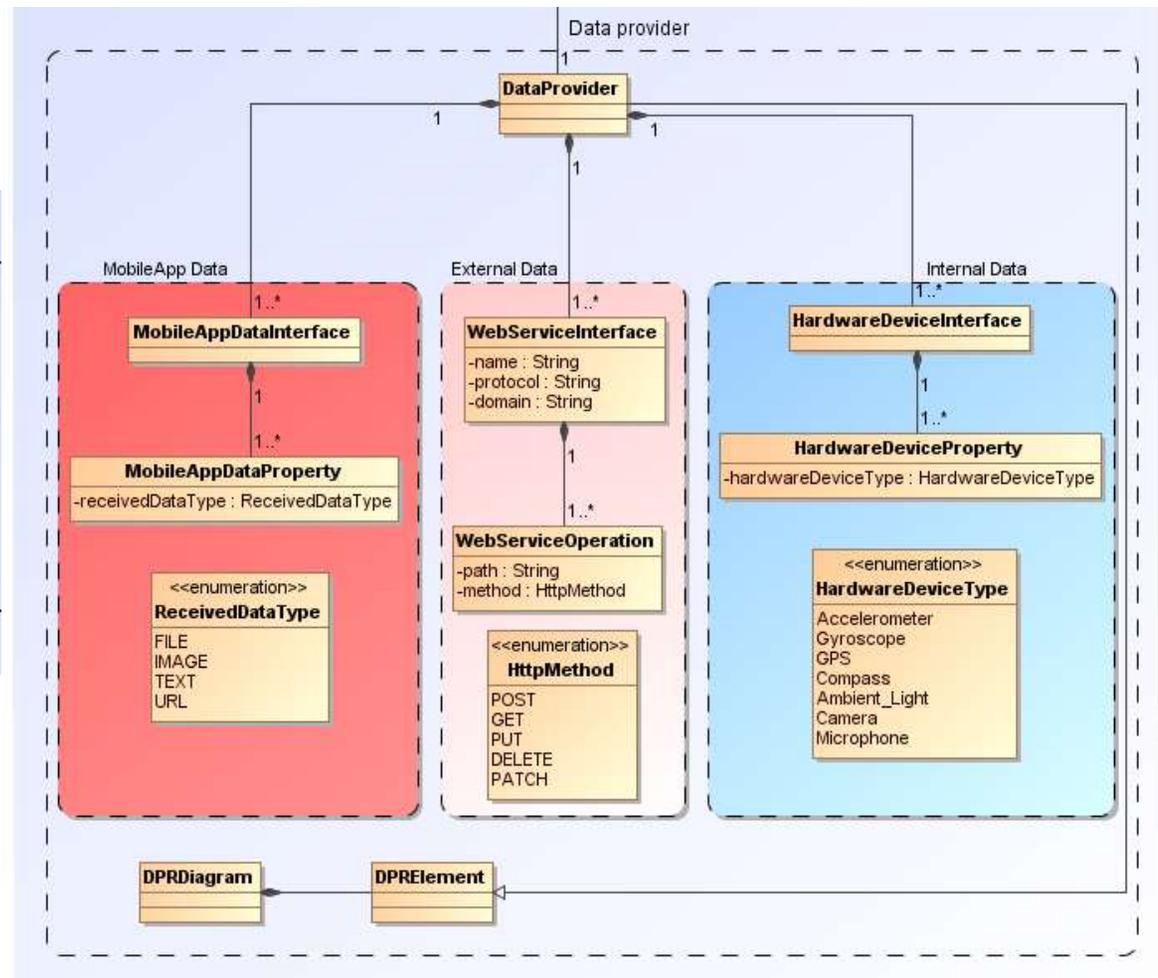
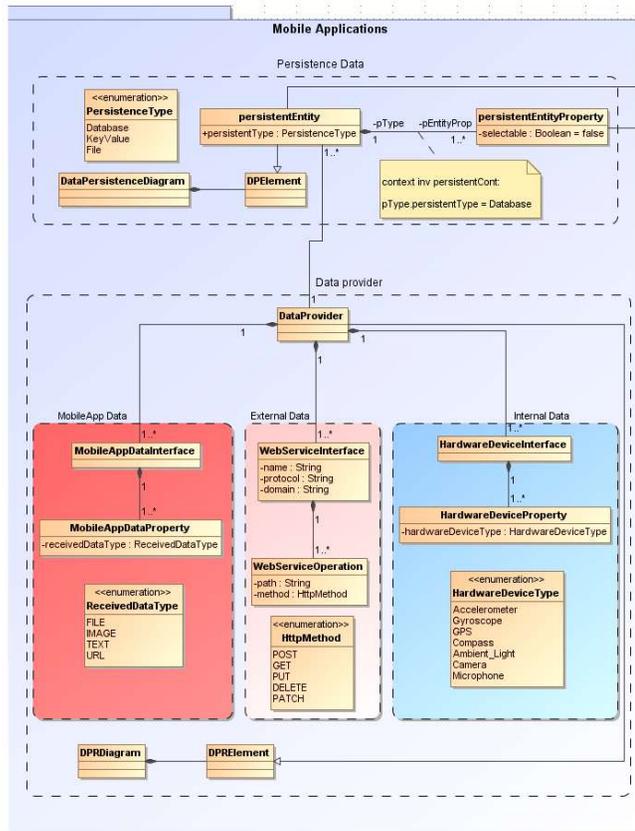
# Definición del ASM móvil

- Metamodelo extendido del PIM de Entidades



# Definición del ASM móvil

## - Metamodelo MoWebA Mobile

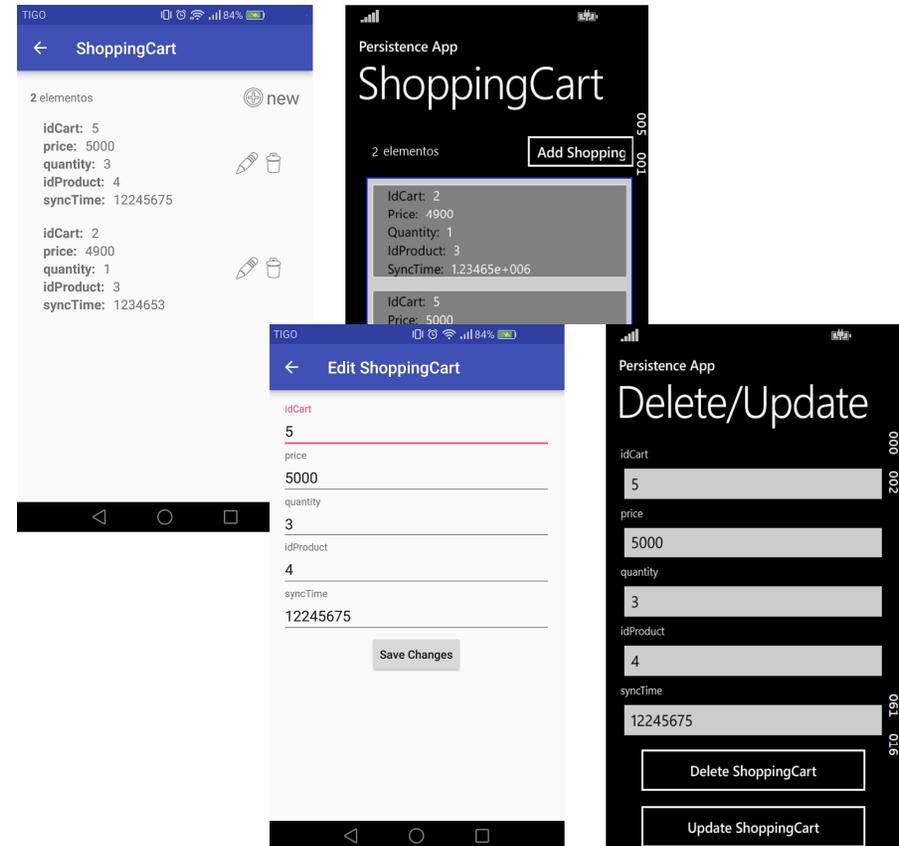


# Ejemplo

- Modelado de una aplicación

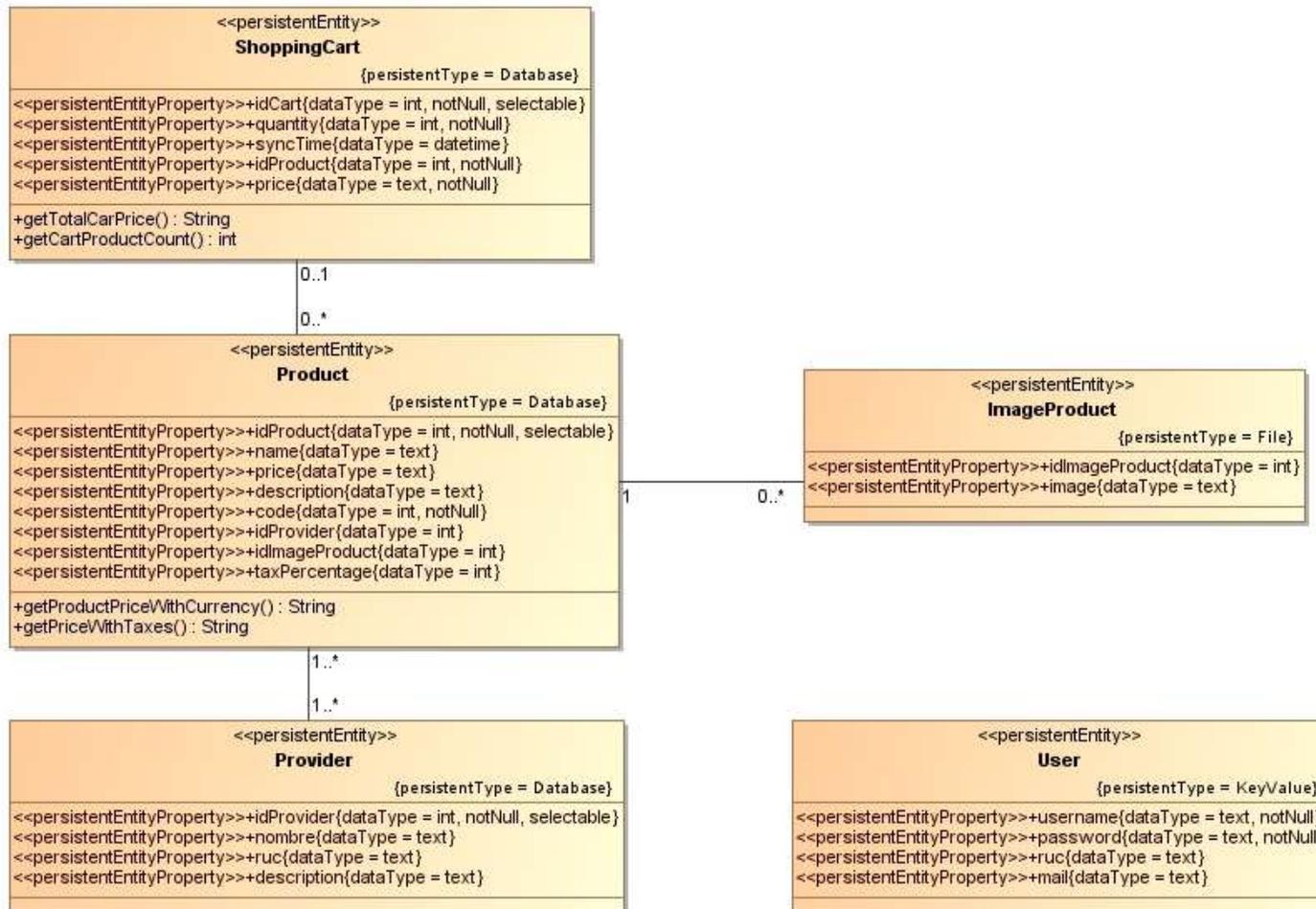
## E-market: compras online

- Agregar productos a tu carrito
- Disponible sin conexión a red
- Entrega a domicilio (Geolocalización)
- Maneja URLs compartidas por otras aplicaciones



# Ejemplo

## - Modelo de persistencia



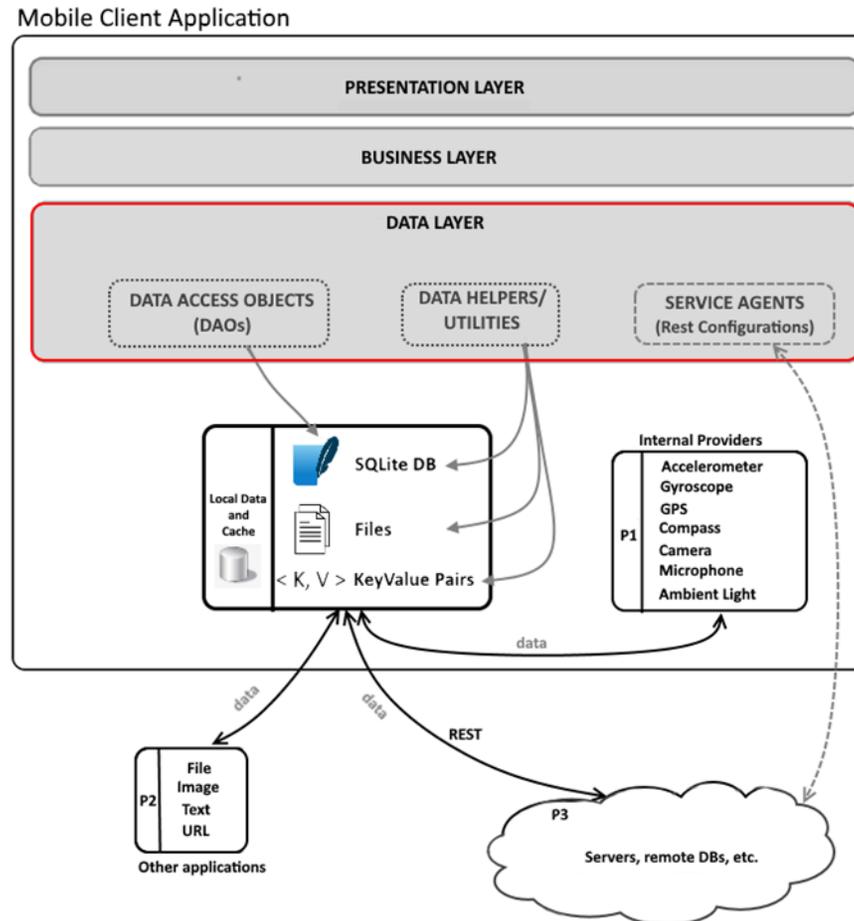
## Ejemplo

### - Modelo de proveedores de datos



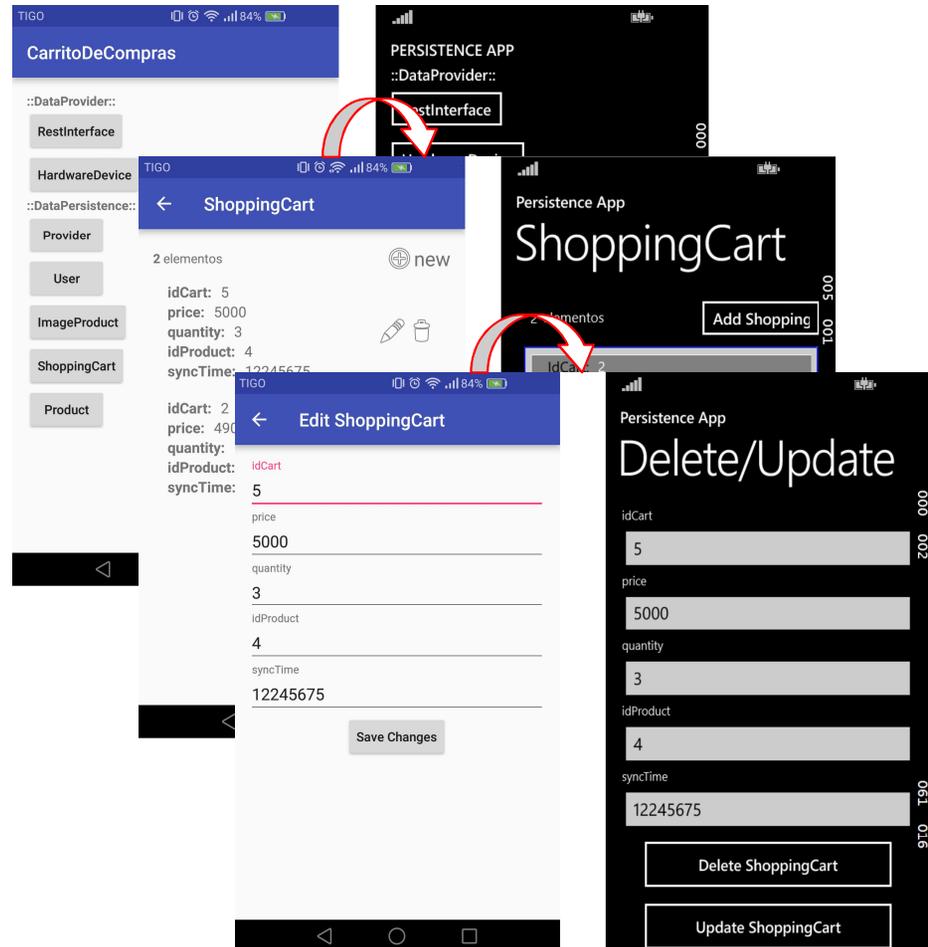
# Reglas de transformación

- Aspectos considerados para la aplicación generada



# Reglas de transformación

```
60 [template public generateElement(model: Model)]
61 [comment @main/]
62 [let aPackages: Sequence(Package) = model.eAllContents(Package) ]
63
64 [generateGeneralAndroidClasses(model)/]
65 [generateGeneralWindowsClasses(model)/]
66
67 [for (aPackage : Package | aPackages)]
68 [let aClasses: Set(Class) = aPackage.ownedElement->filter(Class) ]
69 [let p : Package = aPackage.ancestors(Package)->first()]
70 [comment]Recorremos los paquetes existentes en el modelo. Solo nos interesa dos paquetes:
71 DataPersistence y DataProvider[/comment]
72
73 [comment]Si existe el paquete DataPersistence[/comment]
74 [if (aPackage.hasStereotype('DataPersistence'))]
75
76 [comment]Genera los beans o modelos de la aplicacion[/comment]
77 [beansGenAndroid(aPackage, p.name.toLower()/)]
78 [beansGenWindows(aPackage)/]
79
80 [comment]Solo creamos este archivo Si existe por lo menos una entidad tipo Database[/comment]
81 [if (aPackage.isPackageHasThisPropertyStereotype('persistentEntity', 'persistentType', 'Database'))]
82 [generateDBForAndroid(aPackage, p.name.toLower()/)]
83 [generateDBForWindows(aPackage, p.name.toupperFirst()/)]
84 [endif]
85
86 [comment]Solo creamos este archivo Si existe por lo menos una entidad tipo File[/comment]
87 [if (aPackage.isPackageHasThisPropertyStereotype('persistentEntity', 'persistentType', 'File'))]
88 [generateFilesForAndroid(aPackage, p.name.toLower()/)]
89 [generateFilesForWindows(aPackage, p.name.toupperFirst()/)]
90 [endif]
91
92 [comment]Solo creamos este archivo Si existe por lo menos una entidad tipo KeyValue[/comment]
93 [if (aPackage.isPackageHasThisPropertyStereotype('persistentEntity', 'persistentType', 'KeyValue'))]
94 [generateKVForAndroid(aPackage, p.name.toLower()/)]
95 [generateKVForWindows(aPackage, p.name.toupperFirst()/)]
96 [endif]
97 [endif]
98
99 [comment]Si existe el paquete DataProvider[/comment]
100 [if (aPackage.hasStereotype('DataProvider'))]
101 [for (aClass : Class | aClasses)]
102 [comment]Si la clase tiene el paquete webserviceInterface[/comment]
103 [if (aClass.hasStereotype('webserviceInterface'))]
104 [generateRestAndroid(aClass, p.name.toLower()/)]
105 [generateRestWindows(aClass, p.name.toupperFirst()/)]
106 [endif]
107
108 [comment]Si la clase tiene el paquete HardwareDeviceInterface[/comment]
109 [if (aClass.hasStereotype('HardwareDeviceInterface'))]
110 [generateSensorsAndroid(aClass, p.name.toLower()/)]
111 [generateSensorsWindows(aClass, p.name.toupperFirst()/)]
112 [endif]
113 [endif]
114 [endif]
115 [endif]
116 [endif]
117 [endif]
118 [endif]
119 [/template]
```



## Outline

- Aplicaciones móviles y algunas de sus problemáticas
- El desarrollo de aplicaciones móviles
- Objetivos
- MoWebA y su extensión para móviles
- **Experiencia de validación**
- Conclusiones y Trabajos Futuros

## Experiencia de validación

### - Actividades

---

- Diseño
- Planificación
- Preparación y recolección de datos
- Análisis e interpretación de resultados

## Experiencia de validación

- Diseño

### Objetivos de la experiencia

- Perfiles:

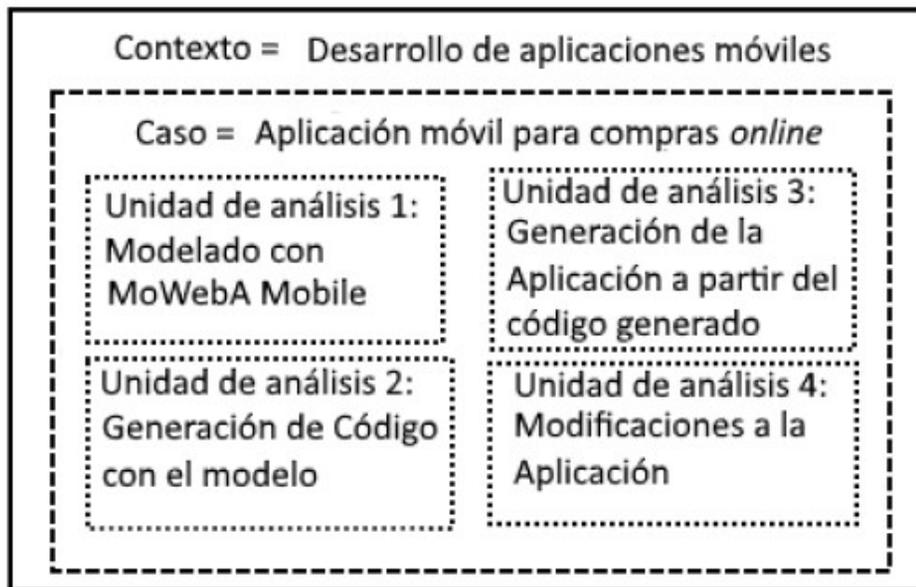
- Modeladores
- Desarrolladores

O1.- **Analizar** el enfoque MDD para el desarrollo de aplicaciones móviles nativas enfocadas en la capa de datos: MoWebA Mobile, **con el propósito de** una mejor comprensión **con respecto a** la usabilidad **desde el punto de vista** del modelador **en el contexto de** desarrollo de aplicaciones móviles.

O2.- **Analizar** el enfoque MDD para el desarrollo de aplicaciones móviles nativas enfocadas en la capa de datos: MoWebA Mobile, **con el propósito de** una mejor comprensión **con respecto a** la usabilidad y portabilidad **desde el punto de vista** del desarrollador **en el contexto de** desarrollo de aplicaciones móviles.

## Experiencia de validación

- Diseño



### ■ Participantes:

- Modeladores: alumnos
- Desarrolladores: desarrolladores móviles

## Preguntas de investigación

01.-

- ¿Qué percepción de usabilidad presenta el proceso de:
  - modelado del enfoque propuesto?
  - generación de código del enfoque propuesto?
- ¿Qué percepción de satisfacción presenta el enfoque MDD propuesto?

02.-

- ¿Qué percepción de usabilidad presenta el proceso de:
  - generación de la aplicación a partir del código generado del enfoque propuesto?
  - realizar una modificación a la aplicación generada?
- ¿Qué percepción de satisfacción presenta el enfoque MDD propuesto?
- ¿Qué percepción de portabilidad presenta el enfoque MDD propuesto?

## Experiencia de validación

### - Planificación

#### ▪ Cuestionarios utilizados:

- ❑ Para medir la usabilidad: ASQ y SUS
- ❑ Primera aproximación a la portabilidad: preguntas abiertas

#### ▪ Sesiones:

- ❑ #1: Exposición del enfoque
  - Desarrollo de una aplicación *ejemplo*.
- ❑ #2: Modelado y generación de código de la aplicación e-market
  - Modelado
  - Generación de código
- ❑ #3: Generación de la aplicación e-market a partir del código generado.
  - Generación de la aplicación a partir del código generado
  - Portabilidad
  - Modificaciones a la aplicación generada: manual
  - Modificaciones a la aplicación generada: modelado

## Experiencia de validación

- Preparación y recolección de datos

- **Fuentes de información:**
  - ❑ Documentación del proyecto
  - ❑ Planillas de medición de tiempos
  - ❑ Cuestionarios: datos cualitativos (por medio de comentarios y opiniones personales) y cuantitativos (puntaje obtenido del cuestionario).
- **Cómo medimos la usabilidad**
  - ❑ Efectividad (*documentación del proyecto - tasa de éxito*)
  - ❑ Eficiencia (*planillas de medición de tiempos - tiempo medido*)
  - ❑ Satisfacción (*cuestionarios - puntaje obtenido*)
- **Cómo nos aproximamos a la portabilidad**
  - ❑ Carga de datos (CRUD) en distintas plataformas
  - ❑ Verificación del código generado

## Experiencia de validación

### - Preparación y recolección de datos

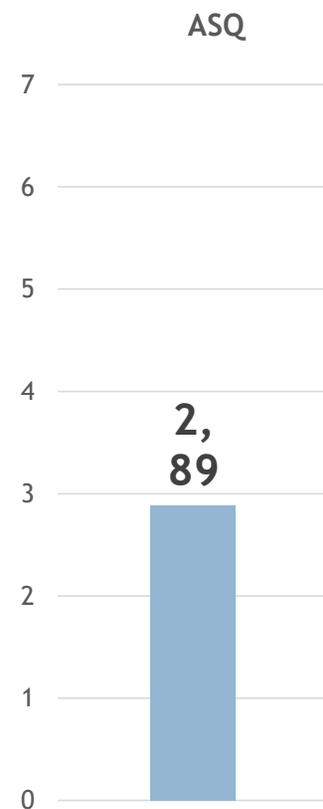
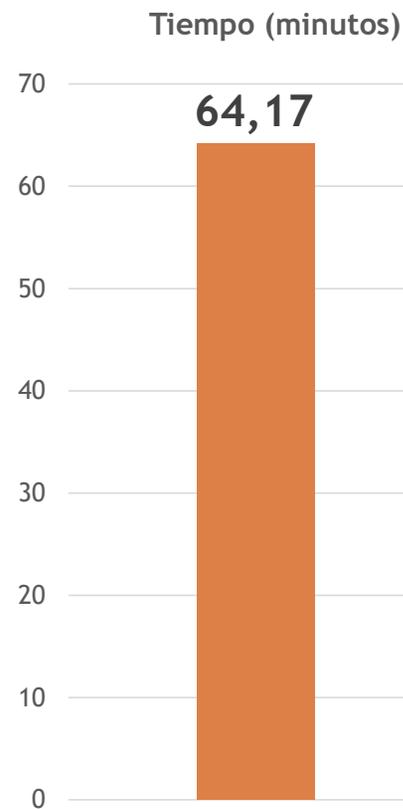
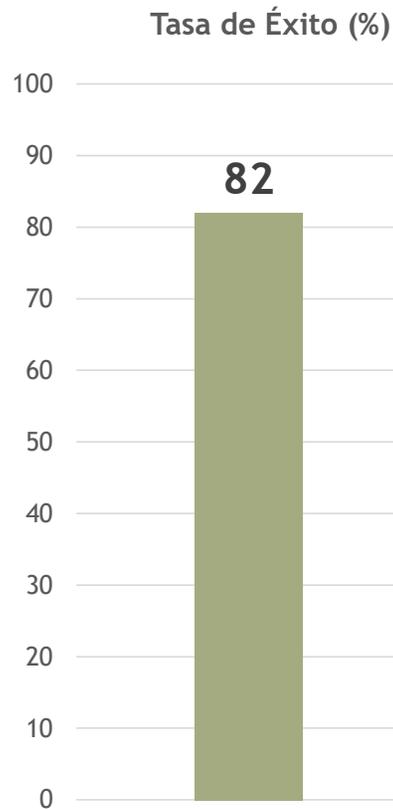
Validez Interna	Control
Nivel de experiencia desbalanceado	Perfiles divididos según la actividad
	Experiencia alumnos y desarrolladores
Falta de participación y ausentismo	Experiencia parte de la actividad de la materia (alumnos)
	Coordinación constante y seguimiento (desarrolladores)
Plagios	Trabajo individual por participante
	Supervisión y control durante la experiencia

Validez Externa	Control
Reducida cantidad de participantes	Permite emitir primeras aproximaciones y juicios iniciales
Caso reducido	Caso de desarrollo simple y funcional, pero con funcionalidades cercanas a la industria; y contempla todos los aspectos propuestos por el enfoque.

# Experiencia de validación

- Modelado

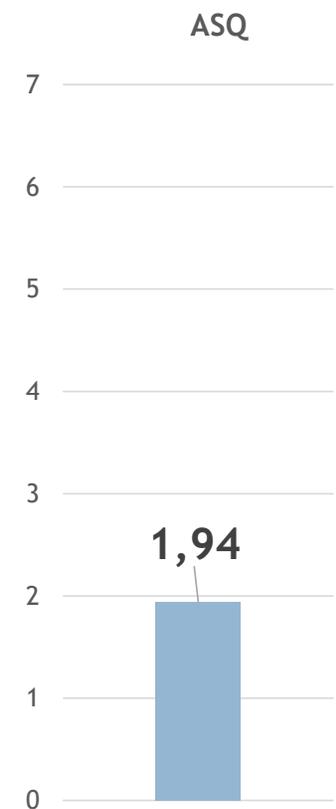
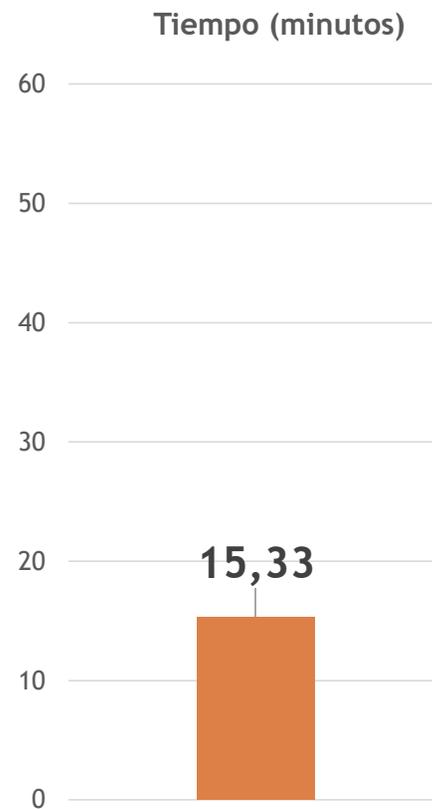
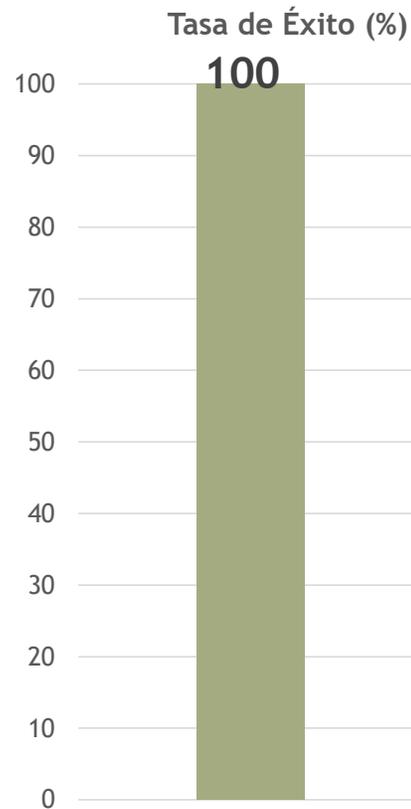
- **Efectividad:** todos completaron la actividad.
- **Eficiencia:** buen tiempo.
- **Satisfacción:** bueno.



# Experiencia de validación

## - Generación de código

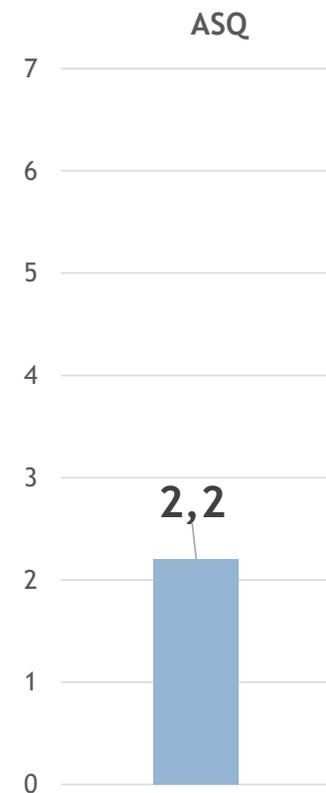
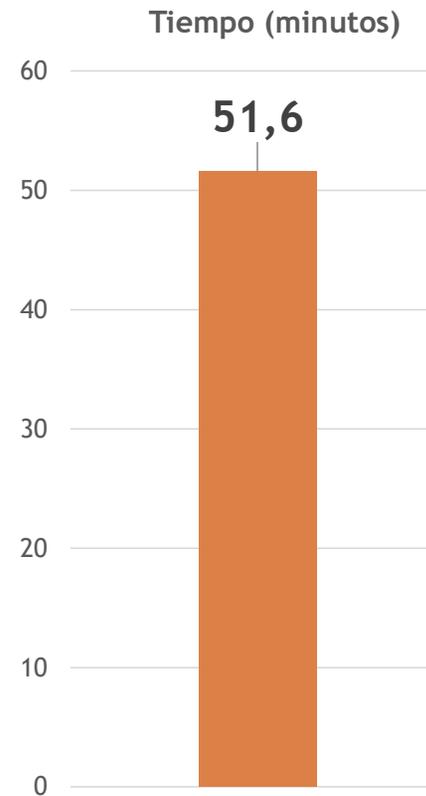
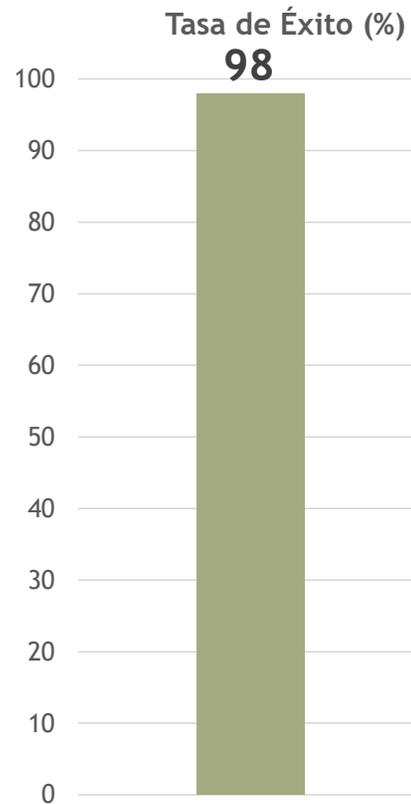
- **Efectividad:** todos completaron la actividad.
- **Eficiencia:** tiempo elevado.
- **Satisfacción:** muy bueno.



## Experiencia de validación

- Generación de la aplicación a partir del código generado

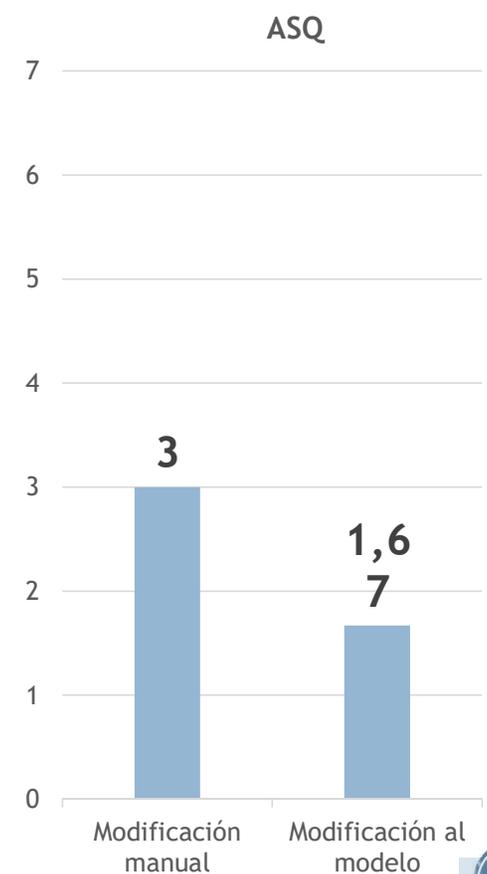
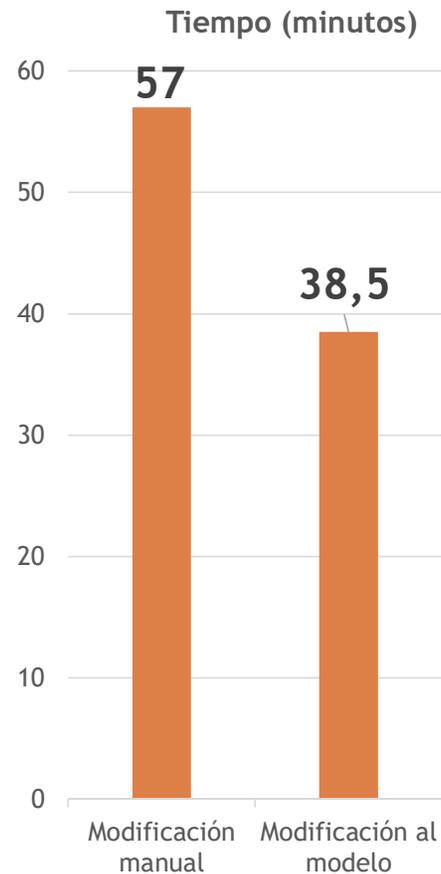
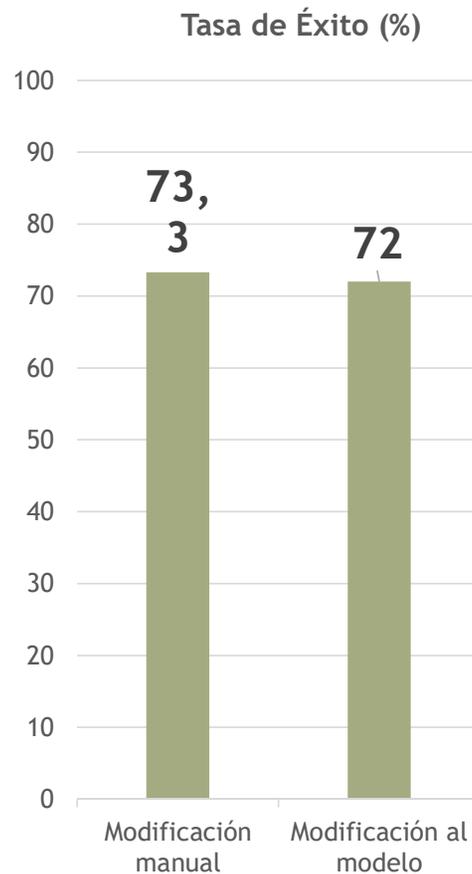
- **Efectividad:** todos completaron la actividad.
- **Eficiencia:** tiempo elevado.
- **Satisfacción:** bueno.



# Experiencia de validación

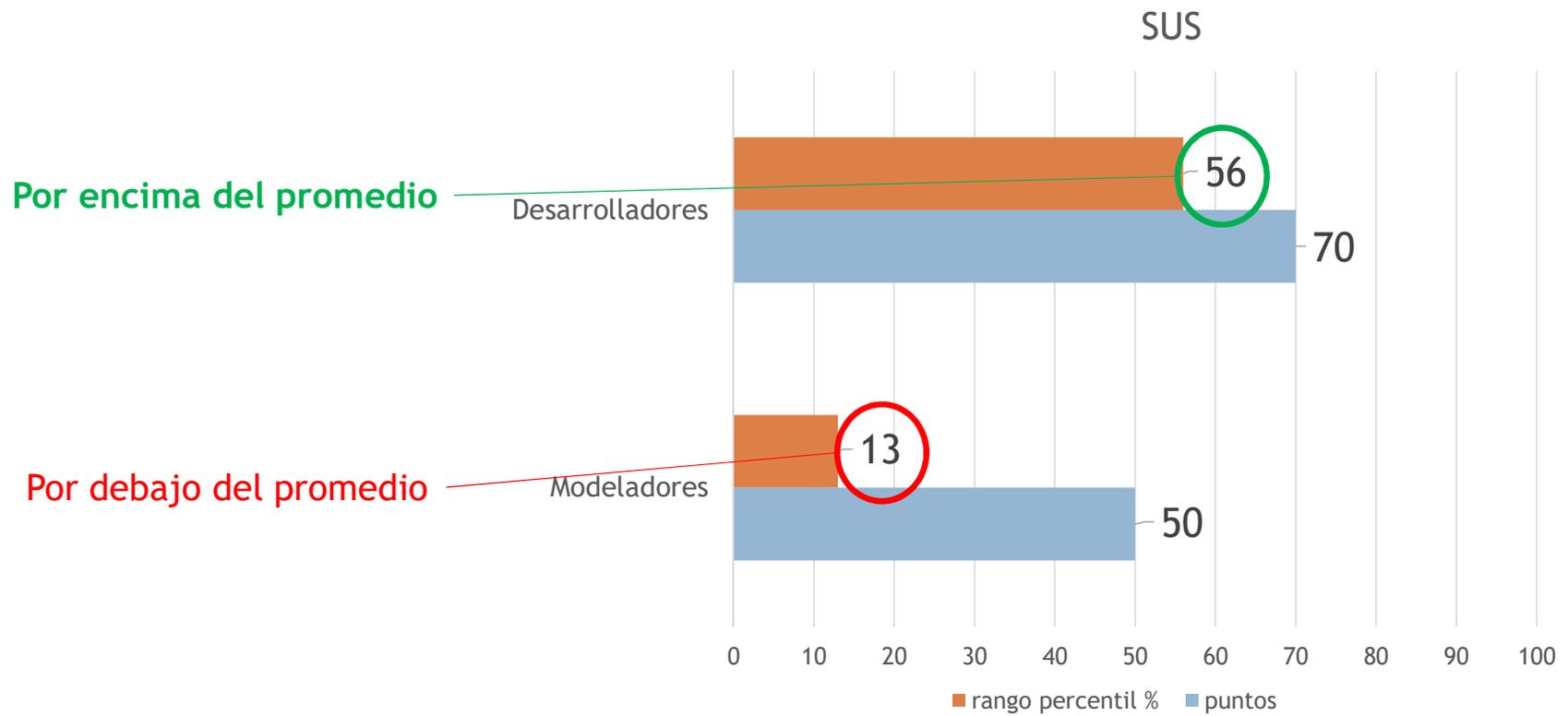
## - Modificación de la aplicación

- **Efectividad:** no todos completaron la actividad.
- **Eficiencia:** enfoque automatico 2.85x más rápido.
- **Satisfacción:** mejor nuestro enfoque.



# Experiencia de validación

- Resultados (Satisfacción general)



## Experiencia de validación

### - Resultados (portabilidad)

Cuestionario para obtener una primera aproximación a la portabilidad, respuestas recabadas:

- *Notaste alguna diferencia con respecto a las funcionalidades?*
  - Buen funcionamiento de los mecanismos de persistencia en las distintas plataformas.
- *Te parece útil MoWebA Mobile?*
  - Sí, notable ahorro de esfuerzo y tiempo para este tipo de desarrollo.
- *Los problemas en este escenario*
  - Errores identificados en la configuración de sensores y algunos detalles de interfaz.

## Experiencia de validación

- Análisis del desarrollo manual vs desarrollo automático

### Desarrollo completo de una aplicación móvil.

	MoWebA Mobile					Desarrollo manual		
	Capacitación	Modelado	Generación de código	Generación de la aplicación	Tiempo Total	Aplicación para Android	Aplicación para Windows Phone	Tiempo Total
Tiempo (minutos)	150	64.17	15.33	51.6	281.1	1500	3660	5160

- El desarrollo manual involucra una curva de aprendizaje por plataforma. Con MoWebA Mobile, curva de aprendizaje en cuanto al modelado.
- MoWebA Mobile, 18X más rápido que el desarrollo manual.
- MoWebA Mobile, menos pasos de desarrollo.

## Experiencia de validación

- Análisis del desarrollo manual vs desarrollo automático

---

### Modificación de una aplicación

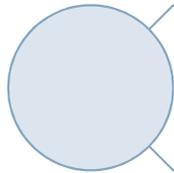
- *Tiempo y ASQ (mencionados previamente)*
- Líneas de código agregadas de las modificaciones solicitadas:
  - ❑ Con MoWebA Mobile se agregaron menos líneas de código.
- Expresividad del modelo (enfoque):
  - ❑ Las modificaciones al modelo no necesitaron ninguna modificación manual extra.

## Outline

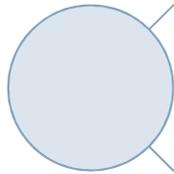
- Aplicaciones móviles y algunas de sus problemáticas
- El desarrollo de aplicaciones móviles
- Objetivos
- MoWebA y su extensión para móviles
- Experiencia de validación
- Conclusiones y Trabajos Futuros

## Conclusiones

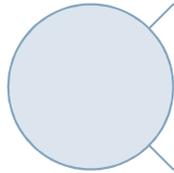
### - Aspectos destacados



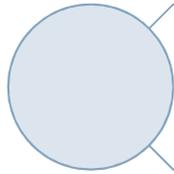
El enfoque contribuye en la solución de los problemas presentados.



Los objetivos establecidos fueron alcanzados.



Resultados positivos y alentadores en la obtención de los primeros juicios arrojados por la experiencia de validación realizada, en cuanto a la evaluación de la usabilidad y portabilidad, y a la comparación con el desarrollo manual.



Potencial del enfoque para expandirse a otras capas y generar aplicaciones más completas.

## Conclusiones

### - Principales contribuciones

- Un estudio detallado acerca de las aplicaciones móviles, de la persistencia en los móviles, MDD y del enfoque MoWebA
- Un estudio de las propuestas MDD para el desarrollo de aplicaciones móviles, observando los aspectos de persistencia utilizados
- Una extensión de MoWebA para el desarrollo de aplicaciones móviles, considerando la capa de datos mediante metamodelos y perfiles ASM
- Reglas de transformación para la generación de una aplicación móvil funcional tanto para Android como Windows Phone, que permite evidenciar las funcionalidades definidas en el enfoque propuesto
- Una validación de la propuesta a través de una experiencia de desarrollo de una aplicación móvil

## Conclusiones

### - Trabajos futuros

- Generación automática de la aplicación
- Diseñar y evaluar aplicaciones de la industria
- Generación para otras plataformas y arquitecturas
- Expandir la generación de los servicios web
- Realizar validaciones más estrictas de la propuesta
- Comparar las medidas de usabilidad
- Realizar validaciones considerando más aspectos de calidad del software

**Gracias por su atención!**

Preguntas?



# Contacto

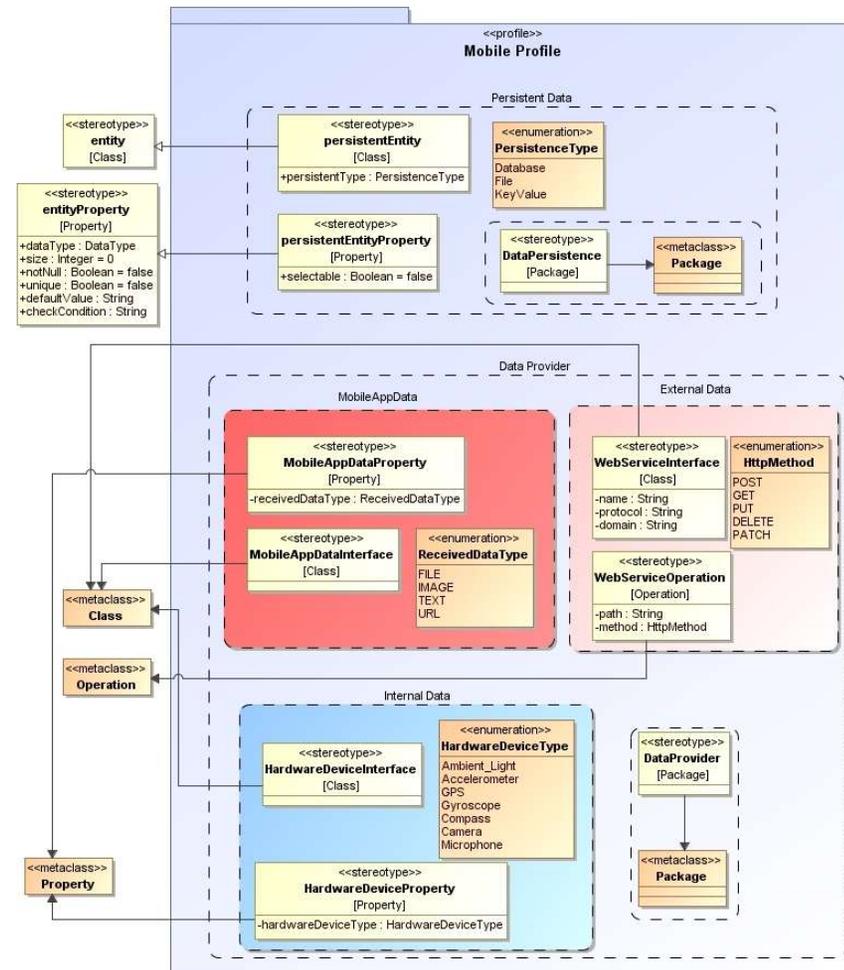
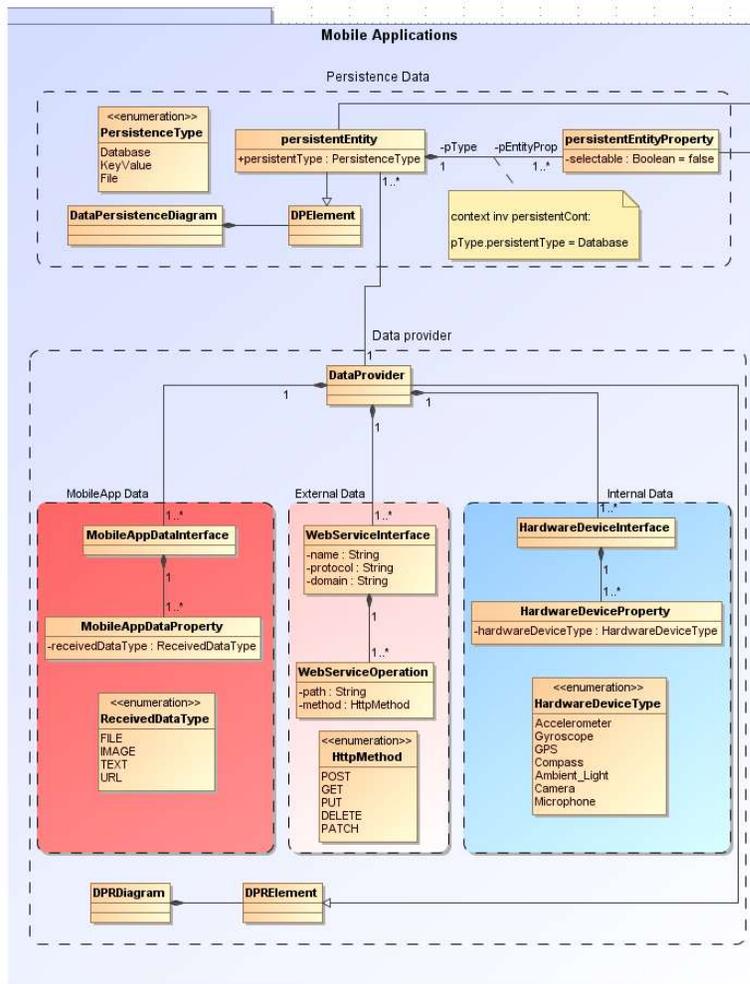
Manuel A. Núñez G.

[manuel.nunez@uc.edu.py](mailto:manuel.nunez@uc.edu.py)

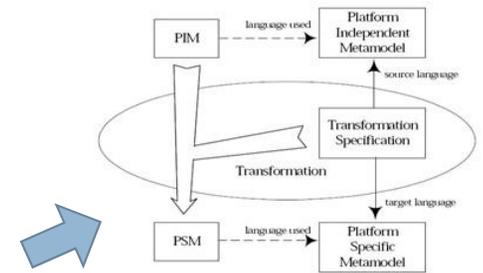


# Definición del ASM móvil

## - Metamodelo y Perfil MoWebA Mobile



# Contexto de la propuesta



# Desarrollo Móvil Dirigido por Modelos

## - Propuestas MDD para el desarrollo de aplicaciones móviles

### Discusión de los puntos más resaltantes:

#### Persistencia

- Muy poco nivel de detalle. Encontramos algunas implementaciones realizadas por las propuestas.

#### Plataformas destino

- Android y iOS constituyen las plataformas más elegidas para la evaluación y validación de las propuestas. Windows Phone figura como tercera opción.

#### Tipo de aplicaciones generadas

- En su gran mayoría son nativas y orientadas a datos.

#### Generación

- A excepción de MD2, la gran mayoría utiliza un IDE y adapta el código generado para tal, para la compilación y generación de la aplicación.

#### Evaluación

- El tipo de validación más utilizado es el de la ilustración o un caso de estudio de la propuesta.

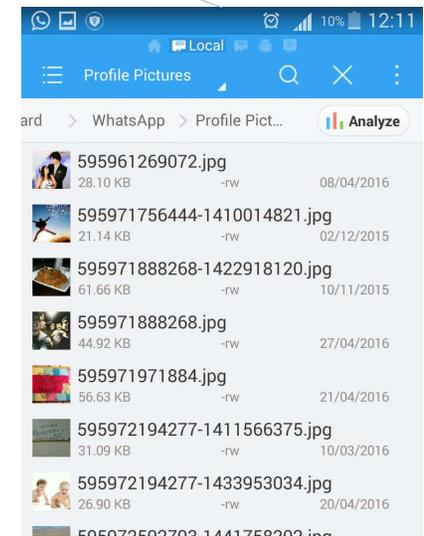
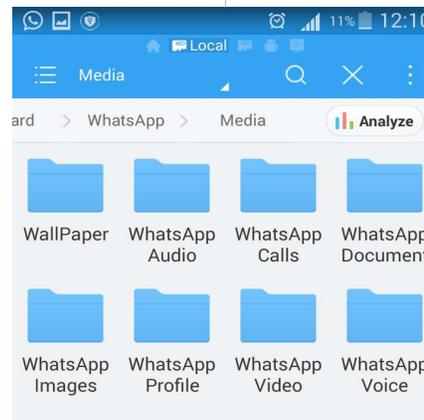
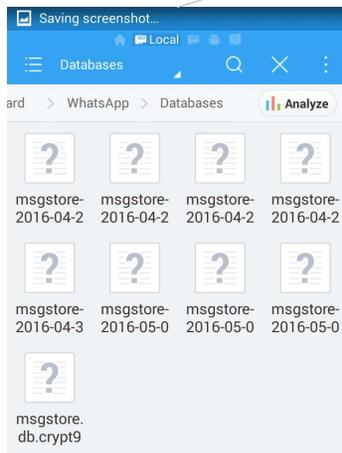
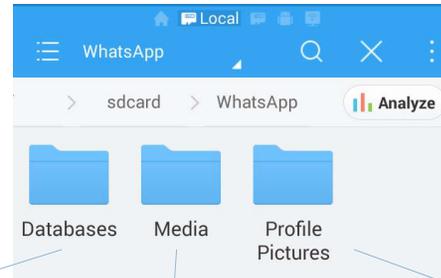
#### Estándar MDA

- Poca adopción de MDA (en especial del estándar MOF).



# Problemáticas

## - Persistencia y la Naturaleza transitoria



# Alternativas para el desarrollo de aplicaciones móviles

## Resultados obtenidos:

13 alternativas

- 6 frameworks
- 7 soluciones dirigidas por modelos

Desarrollo de aplicaciones nativas

- 5 DSL
- 4 frameworks

Desarrollo de aplicaciones híbridas.

- Los frameworks se enfocan más en este tipo de desarrollo

Acceso a API nativo

- Todos los DSL
- 4 frameworks
  - 3 mediante Javascript
  - 1 mediante C#



# Alternativas para el desarrollo de aplicaciones móviles

## Resultados obtenidos:

Los frameworks no requieren de aprendizaje de lenguaje extra

- Los frameworks utilizan C# y .NET, y tecnologías web como HTML5, CSS y Javascript.
- 3 usan HTML5

Los DSL sí requieren un esfuerzo de aprendizaje.

- Proponen nuevos lenguajes específicos del dominio.
- 2 DSL extienden un lenguaje de modelado.

Plataformas de desarrollo

- Las más utilizadas son Android y iOS

Disponibilidad de la herramienta

- La mayoría de los frameworks son herramientas disponibles, con documentación.
- La mayoría de los DSL, son trabajos y/o propuestas en desarrollo.
- 2 DSL permiten acceder a su herramienta



## Desarrollo Dirigido por Modelos

- Aspectos de persistencia a incorporar en los modelos

Posibilidad de utilizar las distintas opciones y mecanismos de persistencia analizados

Posibilidad de modelar la base de datos local para el teléfono móvil

Configuración del *datasource* de conexión a una base de datos externa

Brindar la posibilidad de utilizar la aplicación en modo “sin conexión”

# Problemáticas

## - La fragmentación de la persistencia

### Cómo afecta la fragmentación a la persistencia?

Apple Inc. [US] <https://developer.apple.com/library/ios/referencelibrary/GettingStarted/DevelopiOSAppsSwift/Lesson10.html>

iOS Developer Library

Start Developing iOS Apps (Swift)

Getting Started

- Jump Right In
- Learn the Essentials of Swift

Building the UI

- Build a Basic UI
- Connect the UI to Code
- Work with View Controllers
- Implement a Custom Control
- Define Your Data Model

### Persist Data

This lesson is focused on saving a meal list across FoodTracker app sessions. Data persistence is one of the most important and common problems in iOS app development. iOS has many persistent data storage solutions; in this lesson, you'll use NSCoder as the data persistence mechanism in the FoodTracker app. NSCoder is a protocol that enables a lightweight solution for archiving objects and other structures. Archived objects can be stored on disk and retrieved at a later time.

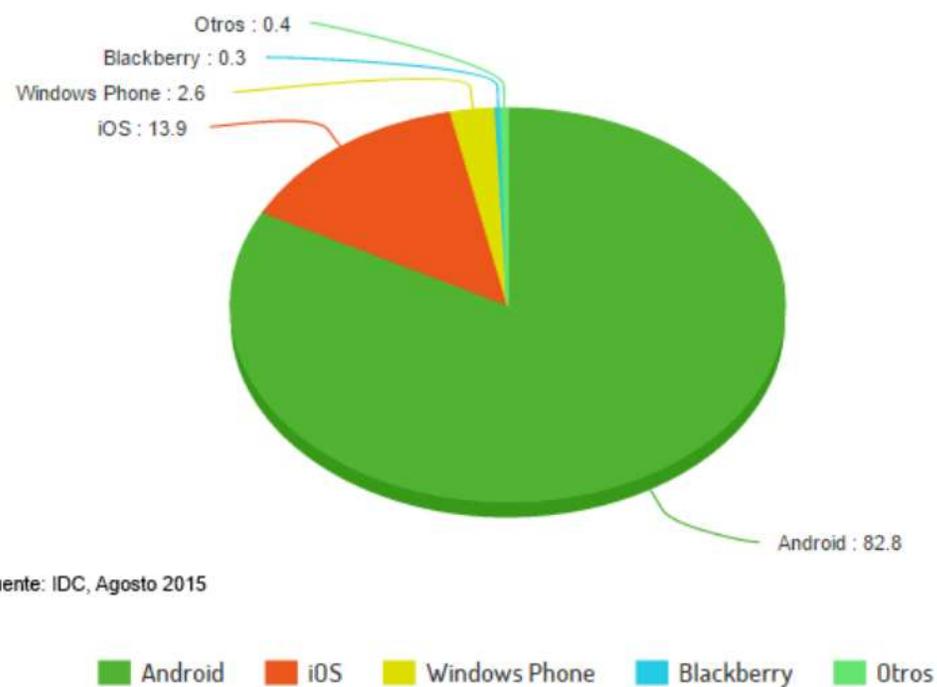
#### Learning Objectives

At the end of the lesson, you'll be able to:

“La persistencia de datos es uno de los problemas más importantes y comunes en el desarrollo de aplicaciones iOS. iOS tiene muchas soluciones de almacenamiento de datos persistente...”

# Sistemas Operativos móviles

## MERCADO ACTUAL DE LOS PRINCIPALES SISTEMAS OPERATIVOS MÓVILES



Fuente: IDC, Agosto 2015

Fig. 2.1 Mercado actual de los sistemas operativos móviles

# Base de datos móviles

Base de datos	Tipo	Tipo de datos almacenados	Licencia	Plataformas soportadas
<b>BerkeleyDB</b>	relacional / NoSQL	objetos, pares clave - valor, documentos	AGPL 3.0	Android, iOS
<b>Couchbase Lite</b>	NoSQL	documentos	Apache 2.0	Android, iOS
<b>LevelDB</b>	NoSQL	pares clave - valor	New BSD	Android, iOS
<b>SQLite</b>	relacional	NULL, Real, Entero, Texto, Blob	Dominio público	Android, iOS, Windows Phone, Blackberry
<b>UnQLite</b>	NoSQL	pares clave - valor, documentos	BSD 2-Clause	Android, iOS, Windows Phone

# Alternativas para el desarrollo de aplicaciones móviles

	FRAMEWORKS						MDE						
	Rhodes	Titanium	Sencha	Apache Cordova (Phonegap)	IBM Mobile first	Xamarin	Mobia	mobl	md2	xmob dsl	Axiom DSL	XIS-Mobile DSL	WebRatio
<b>Licencia</b>	MIT, Comercial	Apache 2.0	GPLv3 or comercial	Apache 2.0	IBM	Comercial	Comercial	MIT	Apache License 2.0	No especificado	No especificado	No especificado	Versión comunidad, gratuito; versión profesional, pago
<b>Código abierto</b>	Si	Si	No	Si	No	No	No	No especificado	No especificado	No especificado	No especificado	No especificado	No especificado
<b>Plataforma soportada</b>	iOS, Android Blackberry, Windows Phone 8	iOs, Android Blackberry y web con HTML 5	Android, iOS, BlackBerry Kindle, Windows Phone, Tizen	Android, iOS, Blackberry, Symbian, WebOS, Windows Phone, FireOS, Ubuntu OS, Firefox OS	iOS, Web con HTML, integración con <i>frameworks</i> PhoneGap, Sencha Touch, jQuery	Android, iOS y Windows Phone	iOS y Android	Navegadores basados en Web-kit, iOS, Android, WebOS, Safari y Chrome	Android y iOS	Android, iOS y .NET	Android y iOS	Android, iOS y Windows Phone	Android y iOS
<b>MVC</b>	Sí	Sí	Sí	No	No especificado	Sí	No	Sí	Sí	Sí	No	No	No especificado
<b>Estado de la herramienta</b>	Herramienta disponible	Herramienta disponible	Herramienta disponible	Herramienta disponible	Herramienta disponible	Herramienta disponible	Herramienta disponible	Herramienta disponible	Trabajo en desarrollo	Trabajo en desarrollo	Trabajo en desarrollo	Trabajo en desarrollo	Herramienta disponible
<b>Modelo</b>	No aplica	No aplica	No aplica	No aplica	No especificado	No aplica	Mobia modeler	DSL	DSL	DSL	DSL	Perfil UML	modelo ER ó modelo IFML
<b>Tipo de modelado permitido</b>	No aplica	No aplica	No aplica	No aplica	No especificado	No aplica	Gráfico	Textual	Textual	Textual	Textual	Gráfico	Gráfico
<b>Lenguajes utilizados</b>	HTML5/ Ruby	Javascript	HTML5, CSS y JavaScript	HTML5, CSS y JavaScript	HTML5	C# y .NET	HTML5, CSS y JavaScript	HTML5, CSS y JavaScript	MD2	Xmob	Axiom	UML	IFML
<b>Tipo de aplicación generada</b>	Nativo	Nativo	Híbrido	Híbrido	Web, Nativo y Híbrido	Nativo	Nativo	Web	Nativo	Nativo	Nativo	Nativo	Híbrido
<b>Acceso a API nativo</b>	Sí, con Javascript/ Ruby	No	Sí, con apache Cordova	Sí, con Javascript	No especificado	Sí	No especificado	Sí, con Javascript	Sí	Sí	Sí	Sí	Sí

# Propuestas MDD

Discusión:

## Persistencia

1

- Mencionan un *Tipo de Persistencia*. Pero no da detalles de cuáles

2

- Utilizan mecanismos específicos de la plataforma (*Isolate Storage* - Windows y *Content Provider* - Android)

1

- Indica si una entidad debe ser persistente

3

- Indican el tipo de conexión para la obtención de los datos, diferenciando entre *local* y *remoto*.

5

- Contemplan la conexión con una base de datos (*datasource*)

2

- Mencionan que cuentan con la modalidad “sin conexión”

# Propuestas MDD

Discusión:

## Estándares de la OMG

- 6 • Utilizan estándares MDA

## Plataformas más utilizadas

- 16 • Generan para Android
- 8 • Generan para Android y iOS
- 4 • Generan para Windows Phone

# Propuestas MDD

Discusión:

## Generación de la aplicación

- 9 • Generan aplicaciones nativas
- 3 • Generan aplicaciones híbridas
- 2 • Generan aplicaciones sin la necesidad de compilar en un IDE
- 6 • Generan aplicaciones dirigidas por datos

## Evaluaciones

- 4 • Generación de una misma aplicación para diferentes plataformas (*cross platform test*)
- 11 • Simple caso de estudio para demostrar el uso de la herramienta
- 3 • Llevaron a cabo pruebas con un grupo de personas (profesionales y personas)

## Planilla de modelado

	Elementos	Puntos	Total puntos
Modelo de persistencia	Estereotipo	30	100
	<<DataPersistence>> (1)	10	
	<<persistentEntity>> (5)	10	
	<<persistentEntityProperty>> (23)	10	
	Tag Value	20	
	persistentType (5)	10	
	selectable (3)	5	
	dataType (23)	5	
Modelo de proveedores	Estereotipo	30	
	<<DataProvider>> (1)	5	
	<<WebServiceInterface>> (1)	4	
	<<HardwareDeviceInterface>> (1)	4	
	<<MobileAppDataInterface>> (1)	4	
	<<WebServiceOperation>> (3)	5	
	<<HardwareDeviceProperty>> (1)	4	
	<<MobileAppDataProperty>> (1)	4	
	Tag Value	10	
	domain = www.api2.cart.com.py (1)	2	
	protocol = http (1)	1	
	method (3)	2	
	path (3)	2	
	hardwareDeviceType = GPS (1)	1	
	receivedDataType = URL (1)	2	
	Servicios	10	
	Modelado de servicios (3)	10	

## Generación de la aplicación

ESCALA	Elementos	Puntos	Total puntos
Generación de la aplicación	Importar al IDE	30	100
	Nombre del proyecto	10	
	Importar código según estructura de carpetas	20	
	Realizar ajustes previos	40	
	Importar librerías	20	
	Corregir beans	10	
	Verificar "Imports"	10	
	Ejecutar compilador	30	

## Modificación

ESCALA	Elementos	Puntos	Total puntos
Modificación manual	Crear tabla nueva	20	100
	Crear DAO para tabla (funciones para agregar, editar y eliminar)	20	
	Mostrar datos provenientes del DAO (Interfaz)	10	
	Generación de la aplicación y correcto funcionamiento de los cambios realizados	50	
Modificación modelo	Cambiar tag value	15	100
	Agregar "selectable"	15	
	Exportar a Acceleo	5	
	Generar código	5	
	Importar al IDE	10	
	Generación de la aplicación y correcto funcionamiento de los cambios realizados	50	

Módulos	Lista de actividades	Android (horas)	Windows (horas)
Base de datos	Trabajo con SQLite en la plataforma (Aprendizaje de la herramienta y/o configuraciones iniciales)	0	5
	Diseño e implementación de pantallas	1	3
	Creación de la BD y Tablas	1	1.5
	Generación de los DAOs para cada tabla	2	3
	Generación de beans	0.5	1
	Testing	0.5	0.5
Files	Trabajo con Files en la plataforma (Aprendizaje de la herramienta y/o configuraciones iniciales)	0	3
	Diseño e implementación de pantallas	1	1
	Diseño de los Helpers	0.5	2
	Testing	0.25	0.5
Pares clave - valor	Trabajo con pares clave valor en la plataforma (Aprendizaje de la herramienta y/o configuraciones iniciales)	0	6
	Diseño e implementación de pantallas	1	1
	Diseño de los Helpers	0.5	2
	Testing	0.25	1
REST	Trabajo con la librería (Aprendizaje de la herramienta y/o configuraciones iniciales)	0	6.5
	Diseño de la interfaz de API y servicios	0.5	0.5
	Diseño e implementación de pantallas	0.5	1
	Testing (diseño de servicios de prueba)	2	2
Sensores	Trabajo con sensores (Aprendizaje de la herramienta y/o configuraciones iniciales)	5	7
	Diseño e implementación de pantallas	3	3.5
	Testing	0.5	2
Interoperabilidad con otras aplicaciones	Trabajo con el método (Aprendizaje de la herramienta y/o configuraciones iniciales)	4	5
	Diseño e implementación de pantallas	0	1
	Testing	1	2
<b>Total</b>		<b>25</b>	<b>61</b>