

**UNIVERSIDAD DE COLIMA**

FACULTAD DE TELEMÁTICA

Desarrollo de una aplicación móvil de Facturación con Código Bidimensional (QR) para Microempresas

TESIS

QUE PARA OBTENER LA TITULACIÓN DE INGENIERO EN SOFTWARE

PRESENTA

RICARDO CIBRIÁN REYNOSO

ASESOR

M. EN C. ARMANDO ROMÁN GALLARDO

Colima, Col. septiembre de 2013

Contenido

[Introducción 4](#_Toc367229106)

[Capítulo 1. Introducción 7](#_Toc367229107)

[1.1 Antecedentes 8](#_Toc367229108)

[1.2 Planteamiento del problema 9](#_Toc367229109)

[1.3 Justificación 9](#_Toc367229110)

[1.4 Objetivos 10](#_Toc367229111)

[1.4.1 Objetivo General 10](#_Toc367229112)

[1.4.2. Objetivos Específicos 11](#_Toc367229113)

[1.5. Importancia de la investigación 11](#_Toc367229114)

[1.6. Metodología 12](#_Toc367229115)

[1.6.1. Fases del ciclo de vida de Programación Extrema: 13](#_Toc367229116)

[Capítulo 2. Estado del arte 15](#_Toc367229117)

[2.1. Smartphones (Dispositivos móviles inteligentes) 16](#_Toc367229118)

[2.1.1. Antecedentes. 16](#_Toc367229119)

[2.1.2. Características 17](#_Toc367229120)

[2.1.3. Participación de fabricantes en el mercado. 18](#_Toc367229121)

[2.1.4. Operadores mexicanos. 19](#_Toc367229122)

[2.1.5. Redes móviles 19](#_Toc367229123)

[2.1.6. Usabilidad 22](#_Toc367229124)

[2.2. Sistemas operativos para dispositivos móviles. 22](#_Toc367229125)

[2.2.1. iOS (Apple) 22](#_Toc367229126)

[2.2.2. Android (Google) 26](#_Toc367229127)

[2.2.3. Windows Phone (Microsoft) 29](#_Toc367229128)

[2.3. Facturación con Código de barras bidimensional. 32](#_Toc367229129)

[2.3.1. Antecedentes 32](#_Toc367229130)

[2.3.2. Características 32](#_Toc367229131)

[2.4. Microempresas en México 34](#_Toc367229132)

[2.4.1. Descripción 34](#_Toc367229133)

[2.4.2. Características 34](#_Toc367229134)

[2.5. Metodologías ágiles. 35](#_Toc367229135)

[2.5.1. Definición. 35](#_Toc367229136)

[2.5.2. Tipos de Metodologías ágiles. 37](#_Toc367229137)

[2.5.3. Programación Extrema. 38](#_Toc367229138)

[2.6. Phonegap y jQuery Mobile 42](#_Toc367229139)

[2.6.1. Phonegap 42](#_Toc367229140)

[2.6.2. jQuery Mobile 44](#_Toc367229141)

[Capítulo 3. Desarrollo de la aplicación móvil 46](#_Toc367229142)

[3.1. Introducción 47](#_Toc367229143)

[3.2. Planeación Inicial 48](#_Toc367229144)

[3.2.1. Historias de usuario. 48](#_Toc367229145)

[3.3. Iteraciones 52](#_Toc367229146)

[3.3.1. Iteración 1. 52](#_Toc367229147)

[Bibliografía 62](#_Toc367229148)

# Introducción

Con la cotidianidad del uso de las computadoras muchos aspectos de nuestras vidas han migrado a un entorno digital. Particularmente lo podemos apreciar cuando realizamos una transferencia bancaria o cuando se solicita una beca a una institución, anteriormente estas actividades se realizaban de forma presencial. Esto creó una gran problemática en muchos aspectos administrativos, dado a que manejar inmensas cantidades de papeleo complicaba tener un control de lo que se hacía y además de que se perdía mucho tiempo.

Esta problemática la empezaron a detectar muchos sectores empresariales que necesitaban restructurar muchos de sus procesos administrativos. La solución fue el uso de herramientas como las computadoras e internet.

En esta etapa de digitalización de los procesos entró el área fiscal recientemente, a consecuencia de los cambios en reforma fiscal mexicana que estableció la obligatoriedad en el proceso de la emisión de comprobantes fiscales mediante el uso medios electrónicos. (SAT, 2010)

Esta reforma fiscal otorga muchos beneficios a los contribuyentes al momento querer generar rápidamente los comprobantes fiscales y un ahorro de papel debido a que todo se realizará de forma electrónica. El contribuyente tendrá la garantía de que la información que contenga su comprobante no puede ser manipulada o modificada.

Existen tres esquemas diferentes, que los contribuyentes pueden optar o deben utilizar, de acuerdo a su actividad económica e ingresos que tenga éste:

• Comprobantes Fiscales Impresos a través de establecimiento autorizado.

• Comprobantes Fiscales Impresos con Código de Barras Bidimensional.

• Factura Electrónica.

Basándose en información otorgada por el Instituto Nacional de Estadísticas y Geografía, se puede estimar que el esquema más utilizado en México son los Comprobantes Fiscales Impresos con Código de Barras Bidimensional ya que este esquema es utilizado por las microempresas las cuales representan el 95.2% de empresas en el país. (INEGI, 2010).

Las microempresas son empresas pequeñas y que tienen ventas anuales hasta por 4 millones de pesos. Si se dedica al comercio se suelen tener entre 11 y 30 trabajadores. Este tipo de organizaciones no se destacan por hacer grandes tipos de inversiones en tecnología, más que los más básico que lo requiera su actividad económico. Esto puede ser atractivo para muchas empresas, en especial a las enfocadas al desarrollo de software, ya que pueden ofrecer productos que pueden ser de gran utilidad a un microempresario que sienta la necesidad de modernizar su negocio a un precio asequible.

Un área que puede resultar atractiva a las microempresas es la fiscal. Con la poca inversión en tecnología, algunos microempresarios pueden tener cierto temor al tener la necesidad de emitir sus comprobantes fiscales de manera digital por su desconocimiento en el uso de la computadora y del Internet. Con esta problemática, podría ser útil tener una herramienta portable que facilitara el proceso de emisión de comprobantes fiscales de manera que sea práctica, sencilla y que no requiera una gran inversión.

Los dispositivos móviles se han convertido en unas herramientas muy importantes en nuestra vida cotidiana. Los teléfonos celulares o los reproductores de música se han convertido en herramientas que realizan más allá de hacer llamadas o reproducir contenido multimedia. Actualmente la mayoría de los dispositivos móviles en el mercado ofrecen una gran cantidad de funciones que permiten mejorar nuestra productividad en la escuela o trabajo. Una de las ventajas de estos dispositivos es que podemos agregarle funcionalidades a nuestro gusto por medio de aplicaciones ya que cuenta con sistema operativo tal como lo utilizan las computadoras.

Se puede notar en el área laboral cómo los dispositivos móviles han ayudado a realizar una gran cantidad de tareas que antes no eran posibles más que con una computadora. Gracias a la popularidad de estos dispositivos, las empresas desarrolladoras de sistemas operativos para móviles han ofrecido herramientas que permiten a los desarrolladores de software para crear aplicaciones que beneficien a infinidad de áreas. Desde aplicaciones que permiten consultar el correo, llevar el control de un inventario, visualizar documentos, entre otras.

En esta investigación se pretende aprovechar los dispositivos electrónicos como una potencial herramienta para el apoyo de la inteligencia de negocios y control administrativo. Utilizando algunas metodologías de desarrollo de software se pretende realizar una aplicación para dispositivos móviles para la emisión de comprobantes fiscales bajo el esquema de Código de Barras Bidimensional, la cual optimice y facilite este proceso en cualquier tipo microempresa.

# Introducción

1. Antecedentes
2. Planteamiento del problema
3. Justificación
4. Objetivos
5. Importancia de la investigación
6. Metodología

## Antecedentes

Los inicios de la telefonía celulares se remontan en la época de la segunda guerra mundial. El fabricante de celulares Motorola había desarrollado un aparato que usaba ondas de radio para comunicarse entre tropas del ejército durante la guerra.

Esta tecnología dio paso a una gran cantidad de herramientas de comunicación, aunque no eran propiamente teléfonos móviles el desarrollo de estos aparatos estableció las bases para la creación de los primeros teléfonos celulares.

El considerado como el padre de la telefonía celular, Martin Cooper, fabricó la primera radio teléfono entre 1970 y 1973, en Estados Unidos. (Basterretche, 2007)

Los smartphones han existido realmente desde 1993. La diferencia entre ese entonces y ahora, es que los primeros smartphones sólo estaban disponibles para altos ejecutivos, ya que su precio resultaba prohibitivo para la mayoría de las personas. (Rodríguez J. )

IBM Simon fue el primer smartphone de la historia. El Simon poseía una interfaz basada en una pantalla táctil, sin botones físicos, con texto predictivo, agenda, funciones de pager y fax, y otras características que hoy consideramos básicas en cualquier celular. (ABC.es, 2012)

Uno de los dispositivos que aumentó la popularidad de los smartphones y cambió totalmente la forma de utilizarlos, fue el Smartphone de Apple, iPhone. Este dispositivo salió al mercado en el 2007 y ofrecía un teclado QWERTY táctil y una excelente experiencia en internet, lo que provocó que tuviera una gran aceptación y en poco tiempo vendió millones de unidades. Actualmente es de los dispositivos móviles de alta gama más comprados en el mundo.

En el mismo año que salió iPhone, la compañía Google presentó su sistema operativo Android, el cual no llamó mucho la atención en su momento, pero 6 años después se convirtió en el principal competidor de iPhone.

Antes un celular estaba limitado a tener funcionalidades prestablecidas y no podíamos agregarle más, como lo haríamos con una computadora. Los smartphones empezaron a ofrecer características adicionales o servicios para que de forma personalizada eligiéramos las que más nos resultaran útiles. Esto derivó a que poco a poco se fueron desarrollando aplicaciones para estos dispositivos, que nos permitieran hacer diferentes tareas aparte de poder mandar mensajes o realizar llamadas.

Enfocándose en otra área importante en esta investigación, como lo es la Facturación electrónica encontramos que sus inicios son a partir del 2005, como respuesta a las necesidades comerciales de los contribuyentes (Conexión a la contabilidad, eliminación del almacenamiento y administración del papel) y como estrategia para reducir esquemas de evasión con documentación apócrifa. (SHCP, 2011)

A partir del primer día del 2011, entraron en vigor las reformas en el Código Fiscal de la Federación, publicadas a principios de diciembre de 2009 en el Diario Oficial. En esta reforma se modificaron los esquemas para la emisión de comprobantes fiscales. La emisión de comprobantes fiscales impresos con código de barras bidimensional, es uno de los esquemas que se contempló en esta reforma. (SAT, 2011)

## Planteamiento del problema

Se desconoce el impacto y beneficios de utilizar aplicaciones móviles para la emisión de comprobantes fiscales digitales en México, que se apeguen a los lineamientos establecidos por el Servicio de Administración Tributaria. Lo cual hace necesario que se realice un estudio que permita conocer los nuevos mecanismos disponibles en el mercado mexicano que agilicen el proceso de emisión de comprobantes fiscales digitales y otras herramientas que enfocadas a las microempresas mexicanas.

En base a la problemática que se ha seleccionado, han surgido una serie de preguntas de investigación.

* ¿Cuáles aplicaciones existen en el mercado para la emisión de comprobantes fiscales digitales?
* ¿Cuál metodología es la más óptima para el desarrollo de una aplicación empresarial móvil?
* ¿Qué beneficios ofrece implementar una herramienta de emisión de comprobantes fiscales para dispositivos móviles en una microempresa?
* ¿Qué impacto tiene la utilización de dispositivos móviles para la emisión de comprobantes fiscales digitales?
* ¿Qué características debe tener una herramienta para la emisión de comprobantes fiscales digitales?

## Justificación

Como lo establece la Ley del Impuesto Sobre la Renta, dispone obligaciones para tanto personas morales como físicas, expedir comprobantes por las actividades que realicen y conservar una copia de los mismos a disposición de la Secretaría de Hacienda y Crédito Público.

Los contribuyentes son sancionados por no entregar o no enviar los comprobantes fiscales por sus actividades o cuando se expidan sin cumplir los requisitos señalados en el Código Fiscal de la Federación, por lo tanto los contribuyentes deben entregar o enviar a sus clientes un comprobante fiscal digital en un plazo de tres días posteriores cuando se realice la operación y de ser necesario entregar una representación impresa del comprobante fiscal digital. (SAT, 2011)

Aparentemente realizar la emisión de los comprobantes fiscales puede resultar sencillo, sin embargo, esta tarea puede ser complicada para algunos contribuyentes que no se sientan muy familiarizados con el uso de una computadora.

Se pretende desarrollar una herramienta que permita emitir comprobantes fiscales para contribuyentes que cumplan los requisitos para utilizar el esquema de Código de Barras Bidimensional. Esta herramienta facilitaría este proceso, para que el contribuyente de forma inmediata pueda emitir la factura para su cliente.

Esta herramienta puede tener un gran impacto ya que puede ser utilizada por contribuyentes que emiten comprobantes fiscales bajo el esquema CBB, los cuales representan un 95% del total de empresas en México. Uno de los beneficios este sistema de facturación, es que no se necesitará invertir en alguna computadora o equipo en especial, sólo con adquirir un dispositivo móvil que cumpla con los requisitos necesarios para soportar esta aplicación. El costo de un dispositivo móvil es bajo respecto a la mayoría de las computadoras que existen en el mercado.

Es posible que por la popularidad de este tipo de dispositivos, el contribuyente haya adquirido para su uso personal algún dispositivo que cumpla las características necesarias para utilizar esta aplicación, por lo que no tendría que invertir en prácticamente nada.

## Objetivos

### Objetivo General

* Agilizar el mecanismo de emisión de comprobantes fiscales bajo el esquema de Código de Barras Bidimensional enfocada a las microempresas mexicanas, a través de una herramienta sencilla y fácil de utilizar mediante un dispositivo móvil.

### Objetivos Específicos

* Identificar las aplicaciones para dispositivos móviles que existan en el mercado para la emisión de comprobantes fiscales con código de barras bidimensional.
* Establecer las características deseables que debe tener una aplicación móvil para la emisión de comprobantes fiscales con código de barras bidimensional.
* Determinar el tipo de metodología más óptima para el desarrollo de la aplicación móvil.
* Desarrollar un prototipo basándose en la metodología de desarrollo de software seleccionada.

## Importancia de la investigación

Durante el año 2010, el uso de la facturación electrónica fue opcional para cualquier contribuyente, por lo cual se le permitía cambiar al esquema de facturación electrónica cuando fuese más conveniente. (SAT, 2013)

A partir del año 2011, se estableció la obligatoriedad del uso de la facturación electrónica, de acuerdo con las reformas establecidas al Código Fiscal de la Federación. (SAT, 2013)

Sin embargo, a pesar de ser una herramienta tan útil, cuando recién se estableció la obligatoriedad en este esquema, provocó una gran cantidad de dudas. En el documental “Facturación Electrónica. ¿Realidad o mito?” (CONTPAQi1, 2010), se puede apreciar de la incomodidad que provocó a los contribuyentes el cambiar un esquema que habían manejado años atrás, aunado con la poca familiarización del uso de las tecnologías actuales.

A octubre de 2012, alrededor de 690 mil contribuyentes han optado por el esquema de facturación electrónica y se han emitido más de 5,000 millones de facturas electrónicas. (SAT, 2012)

La Factura Electrónica está transformando la forma de hacer negocios en México. Quienes la utilizan han visto sus beneficios en materia de seguridad, disminución de costos, optimización de controles internos, impulso de procesos tecnológicos y cambio de prácticas, por lo que han incrementado su uso. (SAT, 2012)

Gracias a la reciente cotidianidad del uso de los dispositivos móviles, podemos realizar una gran cantidad de tareas, que sólo podíamos realizar con una computadora. De acuerdo con The Competitive Intelligence Unit (CIU) en el primer trimestre de 2012, el mercado móvil en México alcanzó 95.1 millones de líneas móviles, de las cuales el 15.5% corresponden a teléfonos inteligentes, un total de 14.7 millones. (Competitive Intelligence Unit, 2012)

Lo que se pretende en este trabajo de investigación-desarrollo es aprovechar el uso cotidiano de estas tecnologías para facilitar la emisión de comprobantes fiscales, utilizando el esquema de CBB (Código de Barras Bidimensional). Se espera que las contribuyentes que sientan temor o incomodad usando una computadora para la emisión de sus comprobantes digitales, tengan una alternativa sencilla e intuitiva para realizar esta tarea con su Smartphone. En el caso de existir aplicaciones para la emisión de comprobantes fiscales en plataformas móviles, se propondrá una metodología para el desarrollo de una aplicación que mejore las ya existentes en el mercado.

## Metodología

La metodología que se utilizará para lograr los objetivos establecidos es la de Programación Extrema.

Programación Extrema (XP, siglas en inglés de eXtreme Programming), es una de las primeras metodologías ágiles que existe (Wells, 2009). Fue introducida por Kent Beck a finales de los años 90’s. Este esquema ha tenido una gran aceptación y nos podemos dar cuenta de ello por la gran cantidad de seguidores que trabajan con este enfoque. (Wells, 2009)

Las metodologías ágiles ofrecen a los desarrolladores flexibilidad en el proceso de desarrollo. XP anima a trabajar sin tener miedo a los constantes cambios de los clientes, aunque sean en fases avanzadas.

La metodología de Programación extrema tiene un ciclo de vida que lo hace similar a algunas metodologías tradicionales, como lo es analizar lo que el cliente necesita, diseñar una solución, desarrollar la solución, realizar pruebas y entregar el producto al nuestro cliente. La diferencia significativa entre Programación Extrema y otras metodologías, es el dinamismo que encontramos en el ciclo de vida, permitiendo cambiar especificaciones sin afectar el producto final.

El ciclo de vida de Programación Extrema está basando en iteraciones. Cada iteración es un periodo de tiempo donde se realizarán todas las fases de desarrollo. Estas iteraciones suelen ser cortas para que el cliente observe si realmente los requerimientos que especificó resuelven su problema, de forma que existe una retroalimentación para buscar detalles a mejorar del desarrollo para volver a realizar una nueva iteración esperando mejorar la calidad del producto después de varias iteraciones. Las iteraciones están compuestas por fases como diseño, codificación y pruebas.

El ciclo de vida ideal de XP consiste de siete fases: Exploración, Planificación de la Entrega, Diseño, Producción, Pruebas, Mantenimiento y Muerte del Proyecto. (Gordillo, 2010)

### Fases del ciclo de vida de Programación Extrema:

#### Fase de Planeación:

En esta fase, se iniciará a elaborar las historias de usuario, las cuales permiten diseñar y desarrollar la solución al problema. De esta forma se identificarán cada una de las tareas que debe realizar el producto final. A partir de esto, se puede ir empezando el desarrollo de los primeros prototipos funcionales. La fase de exploración puede tomar pocas semanas o meses de acuerdo a la complejidad de la tecnología que se implementará.

Las historias de usuario conocer cada una de las necesidades que se tienen que resolver para cumplir el objetivo. Cada historia de usuario viene en forma de ficha que permite recordar detalles importantes. Las historias no deben ser muy extensas para que en pocos días o semanas de trabajo se tenga listo prototipos listos para aplicarles las primeras pruebas funcionales.

#### Fase de Diseño

La metodología de Programación Extrema requiere diseños simples y claros. Un diseño simple se implementa más rápidamente que uno complejo. Un diseño simple se puede implementar en poco tiempo pero que sea funcional. No hay que realizar diseños complejos o adelantarse a diseños que no corresponden a la iteración entorpecería las posteriores fases.

Siempre se buscará en esta fase realizar recodificación, buscando hacer cada vez más sencillo y claro un fragmento de código ya escrito.

#### Fase de Producción

En esta fase se realiza una comprobación del funcionamiento del prototipo desarrollado antes de que sea liberado. En esta fase, los nuevos cambios pueden todavía ser encontrados y debe tomarse la decisión de si se incluyen o no en esta iteración.

#### Fase de pruebas

En esta fase se aplicarán pruebas de unidad y pruebas de aceptación al prototipo.

#### Fase de muerte:

Es cuando ya no se cuentan con más historias para ser incluidas en el producto. Para ello se han cumplido las necesidades y se ha desarrollado el producto final. En esta fase se genera la documentación final.

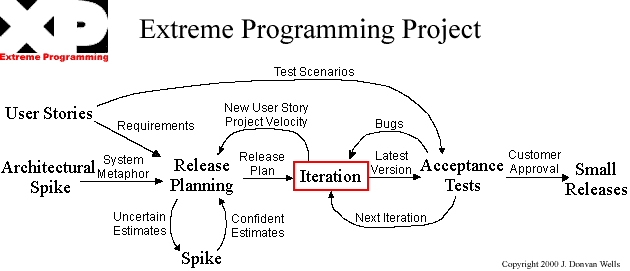


Ilustración . Fases de la metodología Programación Extrema. Ilustración obtenida de: http://www.extremeprogramming.org/map/project.html

# Estado del arte

1. Smartphones.
2. Sistemas operativos.
3. Facturación con código de barras bidimensional.
4. Microempresas en México.
5. Metodologías Ágiles.
6. Phonegap y jQuery Mobile.

## Smartphones (Dispositivos móviles inteligentes)

### Antecedentes.

Los inicios de la telefonía celulares se remontan en la época de la segunda guerra mundial. El fabricante de celulares Motorola había desarrollado un aparato que usaba ondas de radio para comunicarse entre tropas del ejército durante la guerra.

Debido a su alto costo, su tamaño y peso, estos dispositivos, no fueron tan populares como lo son ahora.

Esta tecnología dio paso a una gran cantidad de herramientas de comunicación, aunque no eran propiamente teléfonos móviles el desarrollo de estos aparatos estableció las bases para la creación de los primeros teléfonos celulares.

El considerado como el padre de la telefonía celular, Martin Cooper, fabricó la primera radio teléfono entre 1970 y 1973, en Estados Unidos. (Basterretche, 2007)

Los primeros celulares tenían características que los hacían poco apropiados para su transporte, por lo que sólo se utilizaba a bordo de automóviles. Se solía instalar en un maletero y se pasaba un cable con el teléfono hasta el tablero.

Las primeras generaciones de teléfonos celulares funcionaban de manera analógica, por lo que sólo se permitían realizar transmisiones de voz.

Los smartphones han existido realmente desde 1993. La diferencia entre ese entonces y ahora, es que los primeros smartphones sólo estaban disponibles para altos ejecutivos, ya que su precio resultaba prohibitivo para la mayoría de las personas. (Rodríguez J. )

IBM Simon fue el primer smartphone de la historia. El Simon poseía una interfaz basada en una pantalla táctil, sin botones físicos, con texto predictivo, agenda, funciones de pager y fax, y otras características que hoy consideramos básicas en cualquier celular. (ABC.es, 2012)

Entre los primeros smartphones también podemos encontrar el celular Nokia 9110 Communicator. Fue un dispositivo bastante completo para su época ya que cuenta con características de un Smartphone. Salió a la venta en 1998. Tenía un teclado QWERTY que sirvió para modelo en celulares como Blackberry que popularizó este tipo de teclados en sus dispositivos.

Uno de los dispositivos que aumentó la popularidad de los smartphones y cambio totalmente la forma de utilizarlos, fue el Smartphone de Apple, iPhone. Este dispositivo salió al mercado en el 2007 y ofrecía un teclado QWERTY táctil y una excelente experiencia en internet, lo que provocó que tuviera una gran aceptación y en poco tiempo vendió millones de unidades. Actualmente es de los dispositivos móviles de alta gama más comprados en el mundo.

En el mismo año que salió iPhone, la compañía Google presento tu sistema operativo Android, el cual no llamó mucho la atención en su momento, pero 6 años después se convirtió en el principal competidor de iPhone.

Antes un celular estaba limitado a tener funcionalidades prestablecidas y no podíamos agregarle más, como lo haríamos con una computadora. Los smartphones empezaron a ofrecer características adicionales o servicios para que de forma personalizada eligiéramos las que más nos resultaran útiles. Esto derivó a que poco a poco se fueron desarrollando aplicaciones para estos dispositivos, que nos permitieran hacer diferentes tareas aparte de poder mandar mensajes o realizar llamadas.

### Características

Pantalla

Si se realiza una comparación entre los actuales smartphones con los antiguos celulares, la primera gran diferencia que encontramos es la pantalla, la cual su complejidad y funcionalidad de la pantalla táctil de un smartphone es incomparable. No se ha definido algún estándar para definir un tamaño óptimo para un Smartphone. Algunos prefieren una pantalla contenida de 3,5 pulgadas, y otros prefieren una pantalla más generosa como la que incorpora el nuevo Samsung Galaxy SIII con su pantalla de 4,8 pulgadas. Aunque el Galaxy Note copa lo alto de la clasificación con su pantalla de 5,8 pulgadas, y que lo sitúa entre un smartphone y una tablet. (Ramón, 2012)

Teclado

Existen dos clases de teclados que se imponen en el mercado de los móviles. Uno de ellos es el QWERTY, que se asemeja al de una computadora, pero en miniatura. La otra clase de teclado son muy similares a los QWERTY, sin embargo, éstos son virtuales y forman parte de la pantalla, llamados QWETY táctil.

Almacenamiento de información

El almacenamiento de información en dispositivos móviles puede ser interno o externo. Es importante para la descarga de video y audio, así como el almacenamiento de fotografías. (Pérez P. & Verbakel V., 2009)

Sistema operativo

Existen una gran variedad de sistemas operativos para Smartphones. Android continúa creciendo y en este segundo trimestre se han vendido 102,4 millones de androides, su cuota: un 66,9%. Por detrás aparece iOS, aunque sus 26 millones le sirven para mantener una distancia respecto al tercer sistema operativo su cuota ha bajado un 7%. Dos de cada tres teléfonos tienen el sistema operativo de Google y éste se ha convertido en el socio ideal para muchos fabricantes. El 91% de los smartphones vendidos por Samsung lo usan. Si hablamos de Android en conjunto el reparto por fabricante queda así: 45% para Samsung, 8% para HTC, 8% para ZTE, Sony 7%, Huawei 7%, LG 6%, Motorola 6% y el resto un 13%. En tercer y cuarto lugar Blackberry y Symbian luchan por frenar la caída y no perder sus posiciones. (González, 2012).

Soporte de lenguajes de programación

Los navegadores móviles se conectan generalmente a través de redes de telefonía móvil y muestra contenido escrito en XHTML (WAP 2.0), o WML (que evolucionó del HDML). WML y HDML son de formato reducido e ideales para la transmisión con ancho de banda limitado. Los micronavegadores más recientes pueden manejar HTML, WML, i-mode HTML, cHTML, plus CSS, ECMAScript, y los plug-ins tales como Adobe Flash. Las tecnologías como WAP, plataforma del i-mode de NTTDocomo y plataforma de HDML de Openwave han iniciado la primera avanzada en servicios inalámbricos de datos. (Micronavegador, 2013)

### Participación de fabricantes en el mercado.

Cada trimestre las agencias presentan informes relacionados con el mercado de la telefonía móvil, a nivel regional e internacional. Normalmente esta información tiende a variar ligeramente entre los diferentes informes, por lo que recopilar las estadísticas de las diferentes agencias y mezclarlas no cambia de manera significativa lo que se concluyó en cada estudio. (González, 2012)

La agencia Gartner adelantó a principios de 2012 que Nokia había soltado la corona para que Samsung liderara el mercado en ventas totales de smartphones. Este nuevo trimestre se mantiene la tendencia y son los coreanos quienes lideran aumentando su cuota de mercado un 2,3%. (González, 2012)

Samsung ha vendido 50,4 millones de smartphones en el segundo trimestre. Esto es debido al lanzamiento del Samsung Galaxy SIII, que lleva más de 10 millones vendidos, y a la renovación de otros terminales de gama baja con el Galaxy Ace II. (González, 2012)

Detrás tenemos a Apple con una cuota del 17% y un crecimiento del 1,8% respecto al primer trimestre del año 2012. 26 millones de iPhones vendidos son una cifra notable teniendo en cuenta el corto portafolio de Cupertino. (González, 2012)

En tercer lugar tenemos a Nokia. Su cuota desciende un 1,5% y sus ventas se sitúan en 10,2 millones. (González, 2012)

Detrás, en cuarta posición, HTC crece un 0,4% vendiendo un total de 8,8 millones de teléfonos este trimestre. (González, 2012).

### Operadores mexicanos.

Telcel

Compañía líder de telefonía celular en México. Telcel es una marca propiedad de la empresa mexicana Radiomovil DIPSA. Esta compañía opera bajo TDMA (Análogo y Digital), GSM (2G), GPRS (2.5G) y UMTS (3G).

Su red 4G LTE inicialmente cuenta con una cobertura en más de 30 zonas de alta demanda en 9 ciudades del país y a lo largo del 2013 se espera abarcar más de 26 ciudades cubriendo el 65% de la población. (Telcel)

Movistar

Movistar es la segunda empresa (por cobertura y número de usuarios) proveedora de servicios de telefonía móvil en México. Las tecnologías que utiliza son CDMA (Digital), GSM (2G), GPRS (2.5G), EDGE (2.75G), UMTS (3G), HSDPA (3.5G) y HSUPA (3.75G).

Iusacell

Iusacell es una compañía de telefonía celular propiedad de Grupo Salinas. Opera bajo CDMA (Digital) y EV-DO (3G sobre CDMA).

Unefón

Es una compañía mexicana de telefonía móvil propiedad de Iusacell. Opera bajo CDMA.

Nextel México

Nextel México es una empresa de telecomunicaciones y es una filial de la multinacional NII Holdings. Opera bajo iDEN.

### Redes móviles

* Primera generación (1G)

En las época de los 70’s y principios de los 80’s se empezó a introducir la Primera Generación de redes móviles (1G). La tecnología 1G se caracterizó por ser analógica y sólo por voz, implementó el uso de varias celdas y su capacidad de transferir llamadas de un lugar a otro mientras el usuario, la capacidad de transferir llamadas de un lugar a otro mientras el usuario viajaba durante la conversación, para lo cual la torre de cobertura se enlazaba con los sitios de células cercanas para mantener la comunicación. Sin embargo, la transmisión de estas celdas poco precisa y una deficiente calidad de sonido. (Conde)

Existieron varias tecnologías en esta generación, las cuales son:

* + AMPS (Sistema Telefónico Móvil Avanzado). Esta tecnología evitaba las interferencias entre llamadas gracias a su capacidad de dividir el espacio físico entre dos móviles por las llamadas celdas, que separan las frecuencias. Fue usado en EEUU, America Latina, Nueva Zelanda, Australia y en zonas de Rusia y Asia-Pacífico. (GSMspain).
  + TACS (Sistema de Comunicaciones de Acceso Total). Es conocida como la versión europea del modelo AMPS. Este sistema fue muy usado en Inglaterra y posteriormente en países asiáticos (Hong-Kong y Japón). Utilizaba la banda de frecuencia de 900 MHz. (Kioskea, 2013)
  + NMT (Telefonía Móvil Nórdica): fue introducida en Suecia y posteriormente en Noruega, Finlandia y Dinamarca por la compañía Televerket. Fue la primera red analógica automatizada, del estándar abierto y compatible con roaming. (Alex, 2011)
* Segunda Generación (2G)

En la segunda generación, la tecnología móvil cambió todo su proceso análogo a digital. Debido a al aumento de la demanda se fueron presentando teléfonos celulares más ligeros, avanzados y que tenían una conexión rápida a la señal de las redes. La tecnología utilizada en esta generación fue GSM (Sistema Global para las comunicaciones móviles).

La tecnología GSM mejoraba la calidad de voz y permitía la comunicación mediante mensajes de texto SMS, los cuales aumentaron la popularidad de los celulares. Posteriormente se empezaron a ofrecer servicios como fax y buzón de voz.

Una fase 2.5G fue introducida a finales de los 90’s. Esta tecnología utiliza el estándar GPRS, el cual permite a los usuarios poder enviar datos con gráficos. La importancia de este servicio creció conjuntamente con el desarrollo del Internet y los Protocolos de Internet (IP). Esta tecnología dio paso a los primeros Smartphones.

* GRPS: Es una técnica de conmutación de paquetes, que es integrable con la estructura actual de las redes GSM. Esta tecnología permitirá una velocidad de datos de 115 kbps. Sus ventajas son múltiples, y se aplican fundamentalmente a las transmisiones de datos que produzcan tráfico "a ráfagas", es decir, discontinuo. Por ejemplo, Internet y mensajería. (GSMspain)

El estándar EDGE (Velocidades de datos mejoradas para la evolución global) anunciado como G2.75, cuadriplica las mejoras en el rendimiento de GPRS con la tasa de datos teóricos anunciados de 384 Kbps, por lo tanto, admite aplicaciones de multimedia. En realidad, el estándar EDGE permite velocidades de datos teóricas de 473 Kbits/s pero ha sido limitado para cumplir con las especificaciones IMT-2000 (Telecomunicaciones móviles internacionales-2000) de la ITU (Unión internacional de telecomunicaciones). (Kioskea, 2013)

* Tercera Generación (3G)

La tecnología 3G incorporó nuevas funcionalidades como el uso de aplicaciones con contenidos multimedia, ver videos con rapidez (streaming) y uso de video llamadas. Ofrece velocidades de hasta 2Mbps

Uno de los objetivos más importantes de la tecnología 3G era estandarizar las redes en un único protocolo de red global, en vez de utilizar los diferentes estándares que fueron adoptados anteriormente en Europa, Estados Unidos, y otras regiones.

Además, la tecnología 3G tiene voz y data de forma simultanea, excepto si se utilizan en redes CDMA.

Esta tecnología han sufrido algunas modificaciones, una de las más significativas fue la actualización de la tecnología UMTS, haciendo llegar la misma a velocidades de hasta 14Mbps, en el mejor escenario posible.

En esta generación de redes móviles, aprovechando las grandes velocidades de conexión, podemos utilizar nuestro Smartphone como un módem 3g que nos permita navegar por internet en nuestras laptops o computadoras, sin necesidad de tener contratado algún servicio de banda ancha convencional.

* Cuarta Generación (4G)

La tecnología 4G, técnicamente se le denomina LTE, siglas de “Long Term Evolution” (evolución a largo plazo. El aspecto principal es que se tratará de una red 100% IP, es decir, que podrá interoperar de manera integrada con el resto de redes IP, que configuran Internet. (Galea, 2009)

Esta tecnología tendrá un incremento importante en el ancho de banda, en comparación a las redes 3.5 G, donde se incluyan velocidades mínimas de 100 Mbps. Esto permitirá la visualización de contenidos multimedia en alta definición en tiempo real.

Sin embargo, uno de los principales inconvenientes es que tardará algunos años para implantarse de forma global. En algunos países podría generalizarse en 2020 o más. También dependerá qué tan dispuestas están las operadoras en invertir en infraestructura para esta tecnología. Estas desventajas disminuyendo con el tiempo, como lo fue en algún momento con la tecnología 3G.

4G Americas, organización comercial de la industria compuesta por proveedores de servicios y fabricantes de telecomunicaciones líderes, informó que 150 operadores de todo el mundo ya han lanzado servicios LTE comerciales en 67 países, 50 de ellos durante los últimos 5 meses. (Tecnologia Hecha Palabra, 2013)

### Usabilidad

Diseño de interfaces para dispositivos móviles

Al momento de diseñar interfaces para dispositivos móviles podemos creer que sólo nos tenemos que enfocar los smartphones, es necesario tomar en cuenta a otros dispositivos como pueden ser las tablets o netbooks, incluso los nuevos dispositivos móviles que entraron al mercado llamados “smartwatches” (infobae, 2013) o relojes inteligentes.

Actualmente, podemos encontrar en el mercado una gran variedad de dispositivos móviles de distintas marcas, resolución de pantalla, funcionalidades y tamaños. Por lo que cuando decidamos realizar un diseño para una aplicación, tenemos que tomar en cuenta la diversidad de smartphones para realizar que sea funcional para cualquier modelo sin importar sus características.

No es necesario tener que hacer pruebas con todos los modelos de smartphones, sólo con enfocarse en el diseño de las resoluciones más populares o usadas para crear un diseño que se adapte a cualquier tipo de pantalla que haya en el mercado.

En el proceso de creación de contenidos para dispositivos hay algunas características de contenido y procedimiento que debemos tomar en cuenta.

Los dispositivos son una de ellas, ya que determinan de forma considerable de cómo se consume la información, entre sus característica podemos apreciar el tamaño de su pantalla (la cual actualmente cubre el 90% en la mayoría de los smartphones), su teclado (el teclado se incluye de forma digital a las pantallas táctiles) y el soporte de ciertas tecnologías. (Pérez P. & Verbakel V., 2009)

Un factor que es inherente en la elección de un Smartphone es su operadora móvil, que tiene una función primordial en el acceso a internet. (Pérez P. & Verbakel V., 2009)

El tercer factor son los navegadores web, los cuales nos permiten navegar por internet y realizar muchas actividades que sólo eran posibles con una computadora. Actualmente existen técnicas de diseño y desarrollo web que me permiten adaptar un sitio web al entorno del usuario, esta metodología se conoce como Responsive Design.

Otro factor que es importante es la usabilidad en el diseño de los contenidos, los cuales deben de aprovechar la necesidad de colocar pocos elementos en una pantalla reducida, para que la experiencia del usuario sea satisfactoria al navegar en un sitio web o utilizar una aplicación móvil.

## Sistemas operativos para dispositivos móviles.

### iOS (Apple)

* Antecedentes

Apple tenía una cierta incertidumbre para ir al mercado de la telefonía móvil, lo que Apple quería, era un teléfono con un buen diseño, características únicas, atractivo y que funcionara bajo un sistema nunca antes visto, es por eso, que en 2005 fue la primera vez que Apple habló de la posibilidad de que hubiera un teléfono en su compañía. (Rodríguez E. F., 2011)

Corría el año 2007, realmente acababa de empezar. Ubicados en la Macworld Conference & Expo del 9 de enero,

A principios del año 2007, en la Macworld Conference & Expo fue presentada la primera generación de iPhone y de iOS (antes iPhone OS).

La necesidad comercial de vender el primer modelo de iPhone (que sólo soportaba conexión a redes de GSM con tecnología EDGE) fuera de EEUU, aun siendo ilegal según los términos de venta del dispositivo, fue uno de los factores que crearon el mundo paralelo del iPhone que ha determinado en parte el devenir del mismo en muchos aspectos a lo largo de los años: el universo jailbreak. De hecho, a los pocos días del lanzamiento del iPhone, el 10 de julio de 2007, ya existía un primer método para hacer jailbreak al dispositivo. Su función era tan sencilla como permitir usar canciones de la biblioteca musical como tonos de llamada o de mensaje. Poco después, el 6 de agosto, aparecía el primer juego nativo no oficial para iOS. Steve Jobs, cofundador de Apple, se dio cuenta del potencial que tenía el dispositivo, el cual parecía que no había visto hasta entonces. (Muñoz, 2011)

* Descripción

iOS es un sistema operativo para dispositivos móviles de la empresa Apple Inc. El único dispositivo que contaba con este sistema operativo fue la primera generación de iPhone, pero posteriormente fue incluido en dispositivos como iPod Touch, iPad y Apple TV.

Es una variante del Mac OS X, por lo cual usa un kernel XNU basado en Mach, hereda parte de las tecnologías desarrolladas por NeXT y utiliza frameworks Cocoa. (Morata, 2011)

En la primera versión de iOS se incluyeron aplicaciones para navegar por internet, acceso a la tienda musical iTunes de Apple. Además tenía utilidades para poder revisar correo electrónico, ver contenido multimedia y también ofrecía características de reproductor de música. En las versiones siguientes se implementó posibilidad de cortar, copiar y pegar. Sin embargo la más importante fue poder utilizar aplicaciones de terceros.

En la versión 4 de iOS, introdujo FaceTime, iBooks, GameCenter y multitareas.

En iOS 5 hizo su aparición el asistente personal Siri, que mediante comandos de voz podemos realizar diferentes actividades como mandar SMS, realizar búsquedas en internet y redactar correos, entre otras cosas.

En la más reciente versión de iOS 6, Apple realizó varias modificaciones como presentar un sistema de mapas propio y la herramienta Passbook intenta sustituir los tickets y pases de vuelo. En esta nueva versión el asistente Siri soporta nuevos idiomas como español e italiano.

Una de las prioridades de Apple es dar seguridad a los usuarios al momento de utilizar iOS por lo que constantemente está mejorando la seguridad de este sistema y con ello incluyen funciones para cumplir este fin. Algunas de las funciones que Apple busca para garantizar la seguridad de los usuarios son Ubicación geográfica, Borrado automático y Cifrado de datos. (Rojas Alcocer, Durán Hernández, & Castro Koschny, 2012).

La arquitectura iOS está basada en capas (Cocoa Touch, Media, Core Services y Core OS), donde las capas más altas contienen los servicios y tecnologías más importantes para el desarrollo de aplicaciones, y las capas más bajas controlan los servicios básicos. (Rojas Alcocer, Durán Hernández, & Castro Koschny, 2012).

* ***Capa del Núcleo del Sistema Operativo (Core OS):*** se encarga de realizar la gestión de controladores, memoria virtual, sistema de ficheros, TCP/IP, sockets, seguridad, gestión de memoria y comunicación entre procesos entre otras funciones. (Morata, 2011)
* ***Capa de Servicios principales (Core Services):*** esta capa se encarga de los servicios fundamentales del sistema, que todas las aplicaciones van a usar (directa o indirectamente). Permite realzar conexiones a la red, acceso a ficheros, acceso a la agenda, usar la base de datos SQLite, ubicación del dispositivo y gestión de threads entre otras funciones. (Morata, 2011)
* ***Capa de Medios de Comunicación (Media):*** el conjunto de Frameworks y librerías que forman esta capa permiten construir aplicaciones con gráficos avanzados, reproducción de video, audio, animaciones o imágenes. (Morata, 2011)
* ***Capa de Cocoa (para Mac OS X) - Capa Cocoa Touch (para iOS***): El conjunto básico de herramientas que permiten crear y acceder a los objetos y estructuras de datos básicos, creación de interfaces de usuario, conectar la interfaz con controladores para manejar eventos, etc. (Morata, 2011)
* Desarrollo en iOS

Apple entendía que ciertas compañías de programación deseaban incorporar cierto software al gadget, por lo que empezaron a realizar sus desarrollos sobre el navegador, en Safari, aprovechando la potencia del mismo. Claro está, esta solución no era fue la más idónea. (Muñoz, 2011)

El 27 de octubre Apple presentó el primer kit de desarrollo que se hizo disponible a los ingenieros de software para el año siguiente, en marzo.

A partir de entonces se permitiría el desarrollo de aplicaciones de terceros, habría que establecer un lugar donde alojarlas y ofertarlas al usuario, y así es como nació el App Store, el 11 de julio. Esta tienda para acceder aplicaciones de terceros se incluyó en la segunda versión de iOS.

Para el desarrollo de aplicaciones para dispositivos móviles de Apple en necesario, utilizar el iOS SDK (Software Development Kit) junto con el entorno integrado de desarrollo Xcode.

* Xcode

Con Xcode podemos obtener todo lo necesario para crear aplicaciones (Incluye el SDK). También Incluye un editor de código fuente y de interfaz gráfica de usuario, entre otras características, tales como una única ventana de trabajo en la que se almacena toda la información necesaria para escribir el código, luego compilarlo y diseñar la interfaz de usuario en una misma ventana. Ofrece todo lo que los desarrolladores pueden necesitan para el desarrollo de aplicaciones para Mac, iPhone y iPad.

Las versiones más recientes de XCode han sido optimizadas con un diseño de interfaz de usuario unificado, codificación, pruebas y depuración de todo ello en una sola ventana.

A pesar de ser una herramienta bastante potente y gratuita, Xcode funciona únicamente en computadoras o laptops que tengan Mac OS X instalado.

* Objective C

Objective-C fue desarrollado por BradCox en 1980. Es una extensión del lenguaje C, así como lo es C++. Si se compara con C++, Objetive-C es un lenguaje más limpio, pequeño y rápido de aprender que C++.

El lenguaje que se usa para el desarrollo en Cocoa, la API orientada a objetos de Mac OS X, como Cocoa Touch, la API orientada a objetos de iOS, es Objective-C.

A diferencia de otros lenguajes de las GCC, Objective-C no ha sido estandarizado por ningún organismo internacional, sino que fue NeXTSTEP, y ahora Mac OS X y iOS, quienes han contribuido a crear este lenguaje. (Hernández, 2012)

Objective-C es una extensión de C compatible hacia atrás; es decir, toda la sintaxis C se puede usar en Objective-C. Además, muchas reglas sintácticas de C han sido heredadas por Objective-C, entre ellas:

* + Sentencias de control de flujo.
  + Los tipos de datos fundamentales, estructuras y punteros.
  + Las conversiones implícitas y explícitas entre tipos.
  + Los ámbitos de las variables: globales, estáticas y locales.
  + Las funciones y su sintaxis.
  + Las directivas del preprocesador.
  + Las directivas del compilador.

Si se tiene que seleccionar alguna característica que diferencia a Objetive-C de otros lenguajes de programación, es que este es un lenguaje muy dinámico. Dicho de otra manera, las decisiones que toma este lenguaje los pospone en tiempo de ejecución, mientras que otros lenguajes lo harían en tiempo de compilación.

Objective-C está implementado como una capa situada por encima de C, sigue siendo completamente compatible con código totalmente escrito en C. (Morata, 2011)

### Android (Google)

* Antecedentes

La existencia de Android parece reciente dado a que su popularidad empezó entre el 2008 y 2009, pero en realidad su diseño y construcción inició poco antes del año 2005. Android es una plataforma móvil creada, inicialmente, por una compañía homónima Android Inc., pero su desarrollo fue hecho por Google. Google adquirió Android Inc., a mediados del 2005.

Esta plataforma móvil se inició oficialmente el 22 de agosto de 2008, cuando fue lanzada en un modelo de T-Mobile, llamada G1.

Medio año más tarde (Marzo 2009) es presentada la versión 1.1 de Android con algunas modificaciones estéticas y nuevas posibilidades, cómo la búsqueda por voz, nuevas aplicaciones en el Android Market, etc.

Meses después, se presentaron algunas actualizaciones, en su versión 1.1, con varias modificaciones, entre las que destacan nuevas aplicaciones en la Android Market. Android Market, es una tienda de aplicaciones bastante similar a la App Store de Apple.

A mediados del 2009, Google lanzó las versiones 1.5 y 1.6, las cuales se incluyeron en varios dispositivos de diferentes fabricantes. Incluso la versión 1.6, es de las más usadas, ya que bastantes dispositivos tienen esta versión.

La evolución de Android se fue tal grado que superó a las expectativas de Google en el mercado de la telefonía móvil, sin ser un fabricante de dispositivos consiguió tener a uno de los sistemas operativos más usados en el mundo y directo competidor de Apple.

* Descripción

Android es un sistema operativo desarrollado pensado para dispositivos móviles, pero con el desarrollo y crecimiento de esta plataforma, se ha implementado diferentes dispositivos que en casi nada se relación con un Smarphone. Así lo podemos ver en televisiones, en radios, en vehículos, en lavadoras, en robots industriales, microondas, etc. (antocara, 2012)

Android está basado en Linux, disponiendo de un Kernel en este sistema y utilizando una máquina virtual sobre este kernel que es la responsable de convertir el código escrito en Java de las aplicaciones a código capaz de comprender el Kernel. (antocara, 2012)

Google ha publicado la mayoría del código fuente de Android bajo la licencia de Software Apache, una licencia de software libre y de código abierto a cualquier desarrollador. (Vilchez, 2009 )

#### La arquitectura interna de la plataforma Android, está integrada por las siguientes capas:

* + - Kernel

El núcleo de Android está basado en el kernel de Linux en su versión 2.6, similar a la que pudiera tener cualquier distribución de Linux, como Ubuntu.

Este núcleo funciona como una capa de abstracción entre el hardware y las demás capas de esta arquitectura. Esto impide que al desarrollar accedamos a esta capa de forma directo, sino que tendremos que usar las librerías disponibles en las capas superiores. Esto nos ayuda a no tener la necesidad de conocer las necesidades específicas de cada Smartphone.

Si necesitamos tener acceso a algún elemento del Smartphone, como la cámara, el sistema operativo se encarga de utilizar la que incluya el teléfono, sea cual sea. Para cada elemento de hardware del teléfono existe un controlador dentro del kernel que permite utilizarlo desde el software. (Condesa, 2011)

* Librerías:

Incluye un conjunto de librerías en C/C++ usadas en varios componentes de Android. Están compiladas en código nativo del procesador. Muchas de las librerías utilizan proyectos de código abierto.

En esta capa tenemos un grupo de librerías en C/C++, las cuales proporcionan a Android la mayor parte de sus capacidades más características. Entre las librerías incluidas en Android encontramos una librería de sistema C (libc), una librería de multimedia (formatos de audio, imagen y video), una librería de red, una librería de base de datos (SQLite), una librería de 3D (OpenGL) y una librería de fuente (FreeType).

* + Runtime de Android

El Runtime de Android utiliza el concepto de máquina virtual que se utiliza en Java. Las limitaciones existentes en algunos de los dispositivos Android con poca memoria o procesador limitado, no fue posible utilizar una máquina virtual Java estándar.

Dalvik es la máquina virtual de Android. Está especialmente mejorada para sistemas embebidos, y sus programas ocupan menor espacio de memoria; al mismo tiempo que es capaz de ejecutar varios programas simultáneamente. (Rivas, 2012)

* + Aplicaciones

Aplicaciones. En la última capa se incluyen todas las aplicaciones del dispositivo, tanto las que tienen interfaz de usuario como las que no, las nativas (programadas en C o C++) y las administradas (programadas en Java), las que vienen preinstaladas en el dispositivo y aquellas que el usuario ha instalado. (Condesa, 2011)

* + Entorno de aplicación:

Proporciona una plataforma de desarrollo libre para aplicaciones con gran riqueza e innovaciones (sensores, localización, servicios, barra de notificaciones,). Esta capa ha sido diseñada para simplificar la reutilización de componentes. Las aplicaciones pueden publicar sus capacidades y otras pueden hacer uso de ellas (sujetas a las restricciones de seguridad). Este mismo mecanismo permite a los usuarios reemplazar componentes.

* Herramientas de desarrollo

Para el desarrollo de aplicaciones y juegos para Android, es necesario contar con el SDK Android. SDK Android, contiene las librerías y atributos necesarios para crear aplicaciones en Android. Es necesario tener instalado una plataforma de desarrollo, la más utilizada y recomendada es Eclipse.

* + - * Eclipse

Eclipse es un buen IDE para desarrollar en Android, ya que Java (el lenguaje nativo utilizado para el desarrollo en Android) lo maneja perfectamente. Entre las ventajas de utilizar Eclipse como IDE es que se puede programar en cualquier plataforma Windows, MAC o Linux y además no necesita ser instalado.

Eclipse emplea módulos (plugins) para proporcionar toda su funcionalidad al frente de la Plataforma de Cliente rico, a diferencia de otros entornos monolíticos donde las funcionalidades están todas incluidas, las necesite el usuario o no. (Eclipse)

* + - * Java

Para desarrollar aplicaciones para Android es indispensable tener conocimientos de Java así como de Programación Orientada a Objetos. Java es un lenguaje sencillo y rápido de aprender. Java es compatible con los más variados entornos de red, lo que nos permite desarrollo bajo los sistemas Windows, MAC y Linux sin tener ningún inconveniente.

La programación en java está orientada a objeto, en su mayoría, ya que al estar agrupados en estructuras encapsuladas que hace más fácil su manipulación.

Java es un lenguaje robusto, ya que realiza verificaciones en busca de problemas tanto en tiempo de compilación como en tiempo de ejecución. La comprobación de tipos en Java ayuda a detectar errores, lo antes posible, en el ciclo de desarrollo. Java obliga a la declaración explícita de métodos, reduciendo así las posibilidades de error. Maneja la memoria para eliminar las preocupaciones por parte del programador de la liberación o corrupción de memoria. (Características de Java)

* + - * XML

En el desarrollo para Android se utiliza XML para la parte visual. Es lenguaje descriptivo muy fácil de utilizar y que proporciona sencillez en el desarrollo para aplicaciones Android.

### Windows Phone (Microsoft)

* + Antecedentes

Antes de Windows Phone, existió un sistema operativo llamado Windows Mobile, enfocado para dispositivos móviles. Su lanzamiento fue el 19 de Abril del año 2000 y dejo de actualizarse el 2 de Febrero del 2010. Este sistema operativo ha sido descontinuado y por consiguiente, ya no habrá equipos reciente con este sistema operativo.

Windows Mobile tuvo uno de sus mayores éxitos con la versión de Windows Mobile 2003 y Windows Mobile 5 del 2005. (JCANEZ, 2012)

Windows Phone inicio su desarrollo desde 2008, lo que debía ser un replanteamiento de su sistema operativo, ya que en el mercado existían nuevos sistema operativos que empezaban a acaparar el mercado.

En el año 2010, Microsoft presentó su nuevo sistema operativo Windows Phone en el Mobile World Congress 2010. (JCANEZ, 2012)

En Octubre de 2010, el sistema operativo Windows Phone 7 sale a la venta con el apoyo de 4 fabricantes que crearon 10 modelos que se pusieron a la venta en 30 países a través de 60 operadores. (JCANEZ, 2012)

El 29 de octubre de 2013, se presentó de forma oficial Windows Phone 8, el nuevo sistema operativo de Microsoft con nuevas características y bastantes mejoras con respecto a Windows Phone 7. Los fabricantes que han incluido este sistema operativo en sus dispositivos móviles han sido Nokia, Samsung, HTC y Huawei.

* + Descripción.

Windows Phone es un sistema operativo móvil desarrollado por Microsoft, buscando competir contra sistemas operativos como iOS y Android.

Windows pone incorpora una nueva interfaz llamada “Metro”. Esta interfaz cuenta con una pantalla de inicio que se compone de “Live Tiles” o baldosas vivas. Estas “Live Tiles” son enlaces a aplicaciones, opciones, funciones y elementos fundamentales. Estos son modificables, se pueden borrar, mover y añadir nuevos. Windows Phone utiliza tecnología multitouch y temas oscuros para ahorrar batería ya que los pixeles oscuros no emiten luz. (Microsoft)

Windows 8 también ofrecerá soporte para la tecnología de pagos móviles NFC y gráficos avanzados.

Este sistema operativo ofrece muchas características y herramientas para los administradores de TI y profesionales le fueron agregados. (World, 2012)

* + - * Arquitectura.

En la arquitectura de Windows Phone se cuenta con un framework para el desarrollo de aplicaciones para el sistema y controladores para los diferentes tipos de hardware que tenga el dispositivo.

La gestión de memoria del nuevo sistema operativo es sustancialmente diferente. El espacio de memoria de 32 bits se divide en dos, una parte superior de 2GB dedicada al núcleo y una parte inferior de 2GB dedicado a código fuente que no es del núcleo. La parte superior se divide en dos partes de 1 GB. La parte superior de 1GB se utiliza para diversas estructuras del sistema como las bibliotecas del sistema y la parte inferior de 1GB es para el uso de programas o lo que necesiten. (Fernández)

Windows Phone 7 utiliza sistemas dos sistemas de archivos: IMGFS y TextFAT. El primero esta diseñado para almacenar imágenes actualizables en el sistema. El último es una versión del sistema de ficheros exFAT que se utiliza para el almacenamiento del usuario. Los archivos de usuario se organizan mediante un sistema de almacenamiento unificado que proporcionan aplicaciones que muestra una vista al usuario unificada de los archivos independientemente de su ubicación. Para los gráficos en 3D, Windows Phone 7 utiliza Direct3D 11, basado en DirectX 10. Los frabricantes de dispositivos tendrán que escribir sus propios controladores en 2D y 3D. (Fernández)

* Herramientas de desarrollo

El SDK de Windows Phone 8.0 es un entorno de desarrollo completo que se usa para crear aplicaciones y juegos para Windows Phone 8.0 y Windows Phone 7.5. El SDK de Windows Phone ofrece una versión independiente de Visual Studio Express 2012 para Windows Phone. También puede funcionar como complemento de las ediciones Professional, Premium o Ultimate de Visual Studio 2012. (Microsoft, 2012)

Para desarrollar aplicaciones para la nueva versión del sistema operativo móvil de Windows Phone es necesario tener actualizada el equipo donde se trabajará. Windows Phone 8 SDK sólo permite trabajar en equipos que tengan instalada la versión de 64 bits de Windows 8 Pro.

* + - * .NET

Para el desarrollo de aplicaciones para Windows Phone podemos utilizar los lenguajes de programación C# y Visual Basic .NET. Gracias a que ambos son entendidos por el Framework .NET.

Visual Basic .NET es la evolución de Visual Basic 6.0. Se volvió a desarrollar por completo para la plataforma a fin de hacerlo totalmente orientado a objetos, y puede hacer uso de todos los elementos del framework, como cualquier otro lenguaje. (RedUSERS)

C # es un nuevo lenguaje que fue diseñado, específicamente, para la plataforma .NET. Tiene una sintaxis similar a las de C y Java. (RedUSERS)

Ambos lenguajes son de primer nivel en la plataforma y no presentan diferencias de rendimiento en ella. (RedUSERS)

* XAML

Por la parte de diseño se utiliza el lenguaje Silverlight también conocido como XAML. Silverlight es la alternativa que ofrece Microsoft a Adobe Flash.

XAML es un lenguaje declarativo basado en XML que define los objetos y sus propiedades usando esquemas XML. La sintaxis XAML se centra en la definición de IU (Interfaces de Usuario) para WINDOWS PRESENTATION FOUNDATION (WPF) y por consiguiente esta separado del código oculto de la aplicación. (Paz, 2006)

* XNA

En el caso de las aplicaciones de tipo video juegos, utilizan XNA para generar gráficas de tipo 2D y 3D e igualmente pueden hacer uso de Silverlight al mismo tiempo.

* Visual Studio 2012

Visual Studio 2012 es un conjunto de herramientas y tecnologías que podemos utilizar para desarrollar aplicaciones de alto rendimiento. Visual Studio 2012 nos provee herramientas para diseñar, desarrollar, depurar, e implementar aplicaciones para sistemas operativos Windows. El entorno de desarrollo de Microsoft Visual Studio puede cambiar de acuerdo al lenguaje de programación deseado.

Si se tiene experiencia desarrollando página ASP.NET o aplicaciones .NET, no costará demasiado esfuerzo utilizar esta herramienta tan potente.

## Facturación con Código de barras bidimensional.

### Antecedentes

Los inicios de la facturación electrónica son a partir del 2005, como respuesta a las necesidades comerciales de los contribuyentes (Conexión a la contabilidad, eliminación del almacenamiento y administración del papel) y como estrategia para reducir esquemas de evasión con documentación apócrifa. (SHCP, 2011)

A partir del primer día del 2011, entraron en vigor las reformas en el Código Fiscal de la Federación, publicadas a principios de diciembre de 2009 en el Diario Oficial. En esta reforma se modificaron los esquemas para la emisión de comprobantes fiscales. La emisión de comprobantes fiscales impresos con código de barras bidimensional, es uno de los esquemas que se contempló en esta reforma. (SAT)

### Características

Un contribuyente es toda aquella persona que tiene una actividad económica y que por lo tanto está obligado a contribuir para el financiamiento del gasto público, de acuerdo con las leyes fiscales. Dicho individuo puede ser nacional o extranjero, persona física o moral o bien ser una entidad pública o privada. (SAT, 2013)

* Contribuyentes que pueden utilizar este esquema:

En el esquema de facturación electrónica se incluye, para los contribuyentes con ingresos iguales o menores a 4 millones de pesos al año, la opción de expedir comprobantes fiscales impresos con código de barras bidimensional desde su casa, negocio o café internet.

Dicho en otras palabras, todas aquellas personas físicas y morales que:

a) En el último ejercicio declarado hubieran tenido ingresos menores a 4 millones de pesos.

b) inicien sus operaciones y calculen que durante el año, sus ingresos no superarán los 4 millones de pesos. (SAT, 2013,)

* Requerimientos:

1. Clave CIEC.
2. Fiel vigente
3. Solicitar aprobación de folios vía internet
4. Incorporar Código de Barras Bidimensional (CBB)
5. Impresión de los comprobantes por medios propios del contribuyente o bien a través de un tercero.

* Requisitos de los comprobantes impresos con CBB:
  + De acuerdo al artículo 29-A del CFF:

1. Clave del Registro Federal de Contribuyentes del emisor.
2. Régimen Fiscal en que tributen conforme a la Ley del ISR.
3. Si se tiene más de un local o establecimiento, se deberá señalar el domicilio del local o establecimiento en el que se expidan.
4. Lugar y fecha de expedición.
5. Clave del Registro Federal de Contribuyente de la persona a favor de quien se expida.
6. Cantidad, unidad de medida y clase de los bienes o mercancías o descripción del servicio o del uso o goce que amparen.
7. Valor unitario consignado en número.
8. Importe total consignado en número o letra.
9. Señalamiento expreso cuando la contraprestación se pague en una sola exhibición o en parcialidades.
10. Cuando proceda, se indicará el monto de los impuestos trasladados desglosados por tasa de impuesto y, en su caso, el monto de los impuestos retenidos.
11. Forma en que se realizó el pago (efectivo, transferencia electrónica, cheque nominativo o tarjeta de débito, de crédito, de servicio o la denominada monedero electrónico, indicando al menos los últimos cuatro dígitos del número de cuenta o de la tarjeta correspondiente.
12. Número y fecha del documento aduanero, tratándose de ventas de primera mano de mercancías de importación.
    * De acuerdo a la RMF para 2012 (Regla II.2.6.1.1)
13. El CBB proporcionado por el SAT al contribuyente al momento de la asignación de los folios, mismo que deberá reproducirse con un mínimo 200/200 dpi en un área de impresión no menor a 2.75 cm. por 2.75 cm.
14. El número de aprobación del folio asignado por SICOFI.
15. El número de folio asignado por el SAT y en su caso la serie.
16. La leyenda “La reproducción apócrifa de este comprobante constituye un delito en los términos de las disposiciones fiscales”, misma que deberá ser impresa con letra no menor de 5 puntos.
17. La leyenda “Este comprobante tendrá una vigencia de dos años contados a partir de la fecha de aprobación de la asignación de folios, la cual es: dd/mm/aaaa”, misma que deberá ser impresa con letra no menor de 5 puntos.

* Código de barras bidimensional

Es una imagen que el SAT proporciona al momento de autorizar los folios. (SHCP, 2011)

Contiene los datos del emisor y de la aprobación de la factura:

* RFC del emisor
* Número de aprobación
* Rango de folios y serie
* Fecha de asignación de folios
* Fecha de vigencia

## Microempresas en México

Microempresas son empresas pequeñas y que tienen ventas anuales hasta por 4 millones de pesos. Si se dedica al comercio deberá tener entre 11 y 30 trabajadores, para industria y servicios el tope se sube hasta los 50 empleados.

### Descripción

La mayoría de las empresas que existen en México se componen por micros y pequeñas empresas que conforman más del 95% del total de la industria, por lo que este tipo de empresas son muy importantes en la economía y en el empleo a nivel nacional y regional.

Según Alfredo Rico Bravo, presidente de la Cámara Nacional de Comercio de Ensenada mencionda que: “En México, las Micro, Pequeñas y Medianas Empresas (Mipymes) constituyen el segmento más importante para conseguir el desarrollo del país; representan el 99.8 por ciento del total de las empresas, crean 7 de cada 10 empleos y generan más de la mitad de la riqueza -52 por ciento- del Producto Interno Bruto (PIB)”. (el-mexicano, 2012)

Las microempresas surgen en su mayoría sin planeación y sin financiamiento, nacen al quedar el jefe de familia desempleado, o cuando el salario deja de alcanzar por sostener a la familia. Aunque para ellos se gestan y operan las micro y pequeñas empresas juegan un papel fundamental en el desarrollo y sostenimiento de un país. (Gutiérrez, 2012)

Las microempresas han logrado ser la opción para muchos individuos que después de perder su trabajo, se enfrentan a la necesidad de mantener a su familia. No hay diferenciación de case social o posición laboral, en este grupo de empresarios encontramos desde directivos desempleados de grandes empresas hasta ayudantes de algún oficial técnico. (Gutiérrez, 2012)

### **Características**

En México la microempresa representa el 95.6% del total de todas las empresas es decir de cada 100 empresas mexicanas 96 son micro y contribuyen con el 40.6% del empleo, aportando el 15% del PIB. (Gutiérrez, 2012)

La microempresa es la que ocupa entre 1 y 10 empleados, suyo valor de sus ventas anuales sea el que se determine la secretaría de comercio y fomento industrial. Cuenta con propietarios y administración independiente, no domina el sector de la actividad en que opera y tiene una estructura organizacional muy sencilla. (Gutiérrez, 2012)

Las microempresas son un motor de crecimiento económico y de empleo fundamental para el país ya que, de acuerdo a resultados del último censo económico del INEGI:

* De cada 100 empresas mexicanas, 96 son microempresas
* Contribuyen con el 40.6% del empleo
* Aportan el 15% del PIB

´

* + 1. Tecnología

Hasta hace poco tiempo el sector de las Micro y pequeñas empresas se mantuvo al margen de la revolución tecnológica, liderada por la innovadora Red de redes (Internet). Esto se debió principalmente a la falta de recursos y a la existencia de ciertas trabas culturales, manifiesta Ernesto Haberkorn, vicepresidente de Microsiga Software en un artículo publicado por El Financiero. (Gutiérrez, 2012)

Según datos de la Secretaría de Economía las MPyMEs, invirtieron mil 228 millones de dólares en tecnología de información, lo cual equivale al 21 por ciento de la inversión total en tal adopción. (Gutiérrez, 2012)

Hasta el momento menos del 20% de ellas están automatizadas, lo cual convierte a México en uno de los mercados más grandes y atractivos para compañías proveedoras de hardware y software de cómputo. Los principales servicios que las Micro y pequeñas empresas buscan están destinados a sus procesos básicos, tales como la nómina, la administración, la contabilidad y las compras. (Gutiérrez, 2012)

## Metodologías ágiles.

### Definición.

Durante el desarrollo de un proyecto de software debemos de utilizar metodologías que nos aseguren lograr los objetivos, teniendo una especie de “guía” o planificación que nos indique la forma de hacer las cosas durante todo un proyecto.

Los esquemas tradicionales para el desarrollo de un software han resultado efectivos y necesarios en proyecto complejos. Para tener un control de estos procesos se documenta de manera rigurosa todas las actividades, los recursos necesarios, roles y todos aquellos elementos esenciales para desarrollar el proyecto. En ocasiones usar estos esquemas no pueden ser lo más indicado, ya que los proyecto actuales son muy cambiantes y es necesario realizar constante modificaciones a lo planeado, pero que esto no afecte la calidad del producto final. Los metodologías tradicionales difícilmente aceptar muchas restricciones de tiempo y flexibilidad.

Se buscaron crear enfoques alternativas a las ya existentes, las cuales eliminaran complejos procedimientos y reglas que se enfocan más en cumplir ciertos procesos, que descuidan de cierta forma el desarrollo de un producto.

Las metodologías ágiles emergen como una posible respuesta para llenar ese vacío metodológico. Por estar especialmente orientadas para proyectos pequeños, las Metodologías Ágiles constituyen una solución a medida para ese entorno, aportando una elevada simplificación que a pesar de ello no renuncia a las prácticas esenciales para asegurar la calidad del producto.

Las metodologías ágiles surgen como una respuesta a las necesidades de nuestra época. Son ideales para proyectos pequeños, con la finalidad de simplificar muchos procesos tradicionales pero sin perder la calidad del producto que se pretende realizar.

Estos nuevos esquemas son más flexibles al cambio, por lo que son enfoques que están preparadas para todo momento al cambio. Uno de los objetivos de las metodologías ágiles es adaptarse a cualquier tipo de proyecto, donde los procesos acepten cualquier tipo de restricción y se va entorpecido el desarrollo del proyecto.

En febrero de 2001, se creó una organización, sin ánimo de lucro, enfocada a promover aspectos relacionados con el desarrollo ágil de software. Esta organización se llamó The Agile Alliance.

Esta organización estableció una filosofía, para el desarrollo ágil de software, en el cual establece que:

* ***Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas***. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.
* ***Desarrollar software que funciona más que conseguir una buena documentación.*** La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.
* ***La colaboración con el cliente más que la negociación de un contrato.*** Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
* ***Responder a los cambios más que seguir estrictamente un plan.*** La habilidad de responder a los cambios que puedan surgir a los largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

### Tipos de Metodologías ágiles.

* SCRUM

Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos. El desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. (Schwaber K., 2001)

* Crystal Methodologies

Se trata de un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo y la reducción al máximo del número de artefactos producidos. Han sido desarrolladas por Alistair Cockburn. El desarrollo de software se considera un juego cooperativo de invención y comunicación, limitado por los recursos a utilizar. El equipo de desarrollo es un factor clave, por lo que se deben invertir esfuerzos en mejorar sus habilidades y destrezas, así como tener políticas de trabajo en equipo definidas. (Cockbun, 2001)

* Dynamic Systems Development Method (DSDM).

Define el marco para desarrollar un proceso de producción de software. Nace en 1994 con el objetivo de crear una metodología RAD unificada. Sus principales características son: es un proceso iterativo e incremental y el equipo de desarrollo y el usuario trabajan juntos. Propone cinco fases: estudio viabilidad, estudio del negocio, modelado funcional, diseño y construcción, y finalmente implementación. Las tres últimas son iterativas, además de existir realimentación a todas las fases. (Stapleton, 1997)

* Adaptive Software Development (ASD).

Su impulsor es Jim Highsmith. Sus principales características son: iterativo, orientado a los componentes software más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda desarrollan las características y finalmente en la tercera se revisa su calidad, y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo. (Highsmith & Orr, 2000)

* Feature-Driven Development (FDD).

Define un proceso iterativo que consta de 5 pasos. Las iteraciones son cortas (hasta 2 semanas). Se centra en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software. Sus impulsores son Jeff De Luca y Peter Coad. (Coad, Lefebvre, & De Luca, 1999)

* Lean Development (LD).

Definida por Bob Charettes a partir de su experiencia en proyectos con la industria japonesa del automóvil en los años 80 y utilizada en numerosos proyectos de telecomunicaciones en Europa. En LD, los cambios se consideran riesgos, pero si se manejan adecuadamente se pueden convertir en oportunidades que mejoren la productividad del cliente. (Poppendieck & Poppendieck, 2003)

### Programación Extrema.

Programación Extrema (XP, siglas en inglés de eXtreme Programming), es una de las primeras metodologías ágiles que existe. Fue introducida por Kent Beck a finales de los años 90’s. Este esquema ha tenido una gran aceptación y nos podemos dar cuenta de ello por la gran cantidad de seguidores que trabajan con este enfoque.

Como se explicó anteriormente, las metodologías ágiles ofrecen a los desarrolladores flexibilidad en el proceso de desarrollo. XP anima a trabajar sin tener miedo a los constantes cambios de los clientes, aunque sean en fases avanzadas.

En esta metodología considera que la calidad del software es responsabilidad del gerente, clientes y desarrolladores del proyecto, por lo que es necesario el trabajo en equipo para obtener un producto con las especificaciones deseadas.

* + - Características de la programación extrema:
      * Aceptar la tolerancia a los cambios y requisitos nuevos de forma dinámica
      * Es ideal para proyectos poco complejos o pequeños, aunque también sirve en proyectos con alto riesgo.
      * Este esquema maneja un ciclo de vida iterativo, las iteraciones duran menos de un mes.
      * El proceso de desarrollo se puede ver afectado si no se tiene comunicación constante con el cliente, ya que es la persona conoce perfectamente lo que se quiere.
      * La documentación acerca del diseño o la planificación no es muy extensa.

Es posible que al escuchar el nombre de programación extrema y conocer detalles generales de esta metodología nos sintamos confundido de por qué recibió esta denominación, pero si analizamos los principios que manera esta metodología podemos darnos cuenta que algunos procesos son algo atípicos y que parecen ser muy exagerados, ya que ciertas tareas se manejan a un nivel extremo, por ejemplo:

Analizar nuestro código es una buena actividad para detectar errores que no hayamos visto antes, pero bajo la programación extrema, la revisión del código será una tarea de todos los días.

Las pruebas de cualquier tipo nos ayudan a detectar errores que sólo podemos detectarlos en diferentes escenarios, sin embargo, si lo hacemos sólo en una fase del desarrollo puede que queden incompletas. En programación extrema las pruebas unitarias y las pruebas funcionales serán una constante durante el desarrollo del proyecto.

Otro aspecto importante bajo este esquema, es que la simplicidad será una meta que se buscará lograr de forma constante, logrando simplificar un diseño sin perder en lo absoluto su funcionalidad.

* Principios básicos de la programación extrema:

La Programación Extrema se basa en 12 principios básicos agrupados en cuatro categorías:

* + Retroalimentación a escala fina.

1. El principio de pruebas: se tiene que establecer un período de pruebas de aceptación del programa (llamado también período de caja negra) donde se definirán las entradas al sistema y los resultados esperados de estas entradas. Es muy recomendable automatizar estas pruebas para poder hacer varias simulaciones del sistema en funcionamiento. Para hacer estas simulaciones automatizadas. (Universidad del Valle, 2011)
2. Proceso de planificación: en esta fase, el usuario tendrá que escribir sus necesidades, definiendo las actividades que realizará el sistema. Se creará un documento llamado Historias del usuario (User Stories). Entre 20 y 80 historias (todo dependiendo de la complejidad del problema) se consideran suficientes para formar el llamado Plan de Liberación, el cual define de forma específica los tiempos de entrega de la aplicación para recibir retroalimentación por parte del usuario. Por regla general, cada una de les Historias del usuario suelen necesitar de una a tres semanas de desarrollo. Son muy importantes y tienen que ser una constante las reuniones periódicas durante esta fase de planificación. Estas pueden ser a diario, con todo el equipo de desarrollo para identificar problemas, proponer soluciones y señalar aquellos puntos a los que se les ha de dar más importancia por su dificultad o por su punto crítico. (Universidad del Valle, 2011)
3. El cliente en el sitio: se le dará poder para determinar los requerimientos, definir la funcionalidad, señalar las prioridades y responder las preguntas de los programadores. Esta fuerte interacción cara a cara con el programador disminuye el tiempo de comunicación y la cantidad de documentación, junto con los altos costes de su creación y mantenimiento. Este representante del cliente estará con el equipo de trabajo durante toda la realización del proyecto. (Universidad del Valle, 2011)
4. Programación en parejas: uno de los principios más radicales y en el que la mayoría de gerentes de desarrollo pone sus dudas. Requiere que todos los programadores XP escriban su código en parejas, compartiendo una sola máquina. De acuerdo con los experimentos, este principio puede producir aplicaciones más buenas, de manera consistente, a iguales o menores costes. (Universidad del Valle, 2011)
   * Proceso continuo en lugar de por lotes.
5. Integración continua: permite al equipo hacer un rápido progreso implementando las nuevas características del software. En lugar de crear builds (o versiones) estables de acuerdo a un cronograma establecido, los equipos de programadores XP pueden reunir su código y reconstruir el sistema varias veces al día. Esto reduce los problemas de integración comunes en proyectos largos y estilo cascada. (Universidad del Valle, 2011)
6. Refactorización: permite a los equipos de programadores XP mejorar el diseño del sistema a través de todo el proceso de desarrollo. Los programadores evalúan continuamente el diseño y recodifican lo necesario. La finalidad es mantener un sistema enfocado a proveer el valor de negocio mediante la minimización del código duplicado y/o ineficiente. (Universidad del Valle, 2011)
7. Entregas pequeñas: colocan un sistema sencillo en producción rápidamente que se actualiza de forma rápida y constante permitiendo que el verdadero valor de negocio del producto sea evaluado en un ambiente real. Estas entregas no pueden pasar las 2 o 3 semanas como máximo. (Universidad del Valle, 2011)
   * *Entendimiento compartido.*
8. Diseño simple: se basa en la filosofía de que el mayor valor de negocio es entregado por el programa más sencillo que cumpla los requerimientos. Simple Design se enfoca en proporcionar un sistema que cubra las necesidades inmediatas del cliente, ni más ni menos. Este proceso permite eliminar redundancias y rejuvenecer los diseños obsoletos de forma sencilla. (Universidad del Valle, 2011)
9. Metáfora: desarrollada por los programadores al inicio del proyecto, define una historia de como funciona el sistema completo. XP estimula historias, que son breves descripciones de un trabajo de un sistema en lugar de los tradicionales diagramas y modelos UML (Unified Modeling Language). La metáfora expresa la visión evolutiva del proyecto que define el alcance y propósito del sistema. Las tarjetas CRC (Clase, Responsabilidad y Colaboración) también ayudarán al equipo a definir actividades durante el diseño del sistema. Cada tarjeta representa una clase en la programación orientada a objetos y define sus responsabilidades (lo que ha de hacer) y las colaboraciones con las otras clases (cómo se comunica con ellas). (Universidad del Valle, 2011)
10. Propiedad colectiva del código: un código con propiedad compartida. Nadie es el propietario de nada, todos son el propietario de todo. Este método difiere en mucho a los métodos tradicionales en los que un simple programador posee un conjunto de código. Los defensores de XP argumentan que mientras haya más gente trabajando en una pieza, menos errores aparecerán. (Universidad del Valle, 2011)
11. Estándar de codificación: define la propiedad del código compartido así como las reglas para escribir y documentar el código y la comunicación entre diferentes piezas de código desarrolladas por diferentes equipos. Los programadores las han de seguir de tal manera que el código en el sistema se vea como si hubiera estado escrito por una sola persona. (Universidad del Valle, 2011)
    * Bienestar del programador.

1. La semana de 40 horas: la programación extrema sostiene que los programadores cansados escriben código de menor calidad. Minimizar las horas extras y mantener los programadores frescos, generará código de mayor calidad. Como dice Beck, está bien trabajar tiempos extra cuando es necesario, pero no se ha de hacer durante dos semanas seguidas. (Universidad del Valle, 2011)

## Phonegap y jQuery Mobile

### Phonegap

Phonegap es una framework de código abierto para la creación de aplicaciones nativas multiplataforma usando tecnologías como “HyperText Markup Language”, “Cascading Style Sheets” y Javascript. El tipo de aplicaciones desarrolladas bajo esta tecnología son conocidas como aplicaciones híbridas. La finalidad principal de este framework es facilitar el desarrollo de aplicaciones móviles y poder distribuirlas rápidamente en los diferentes mercados que existen. Ya que hacerlo de forma convencional requiere costos y tiempo destinado en únicamente la capacitación.

PhoneGap también es compatible con frameworks de desarrollo web móvil como jQuery Mobile o Sencha Touch.

Phonegap actualmente soporta los siguientes sistemas operativos móviles:

* Apple iOS.
* Google Android.
* HP/Palm webOS.
* Microsoft Windows Phone 8.
* Nokia Symbian.
* RIM Blackberry.

#### Historia.

Fue desarrollado por Nitobi. Sin embargo, en Octubre de 2011 Adobe anunció la adquisición de Nitobi, de forma que PhoneGap ahora pertenece a la empresa Adobe. Existió la duda si Adobe convertiría Phonegap en una tecnología propiertaria, pero Adobe donó Phonegap a la fundación Apache, conservando su característica OpenSource. Actualmente la fundación Apache lo ha nombrado como Apache Cordova, aunque todavía se utiliza el nombre de Phonegap.

#### ¿Cómo trabaja este Framework?

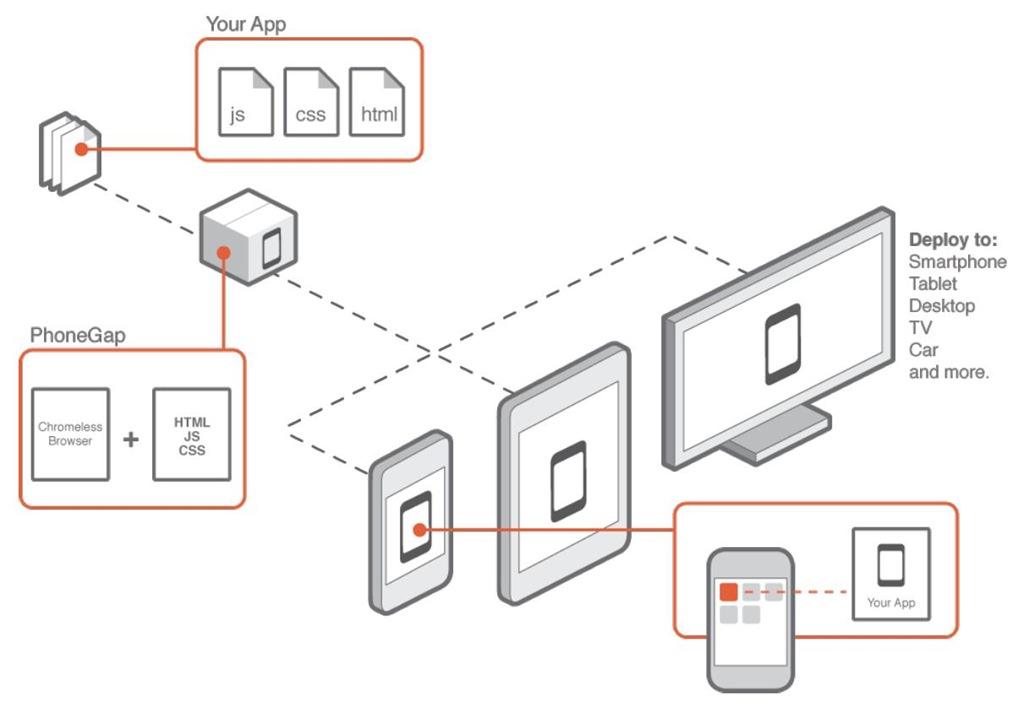


Ilustración 2. Arquitectura de Phonegap

PhoneGap permite a los desarrolladores construir aplicaciones nativas para dispositivos móviles (tanto para smartphones y tablets) utilizando tecnologías web como HTML, CSS y JavaScript. Phonegap provee un paquete de herramientas que permiten que tecnologías web puedan acceder a funciones específicas del teléfono (actualmente no es posible sin este tipo de frameworks) como acelerómetro, cámara, GPS, gestión de archivos, entre otras características. Sí analizamos la aplicación nativa cómo funciona nos daremos cuenta que es simplemente una pantalla que contiene una vista web, de forma que ahí dentro se mostrará nuestra aplicación móvil utilizando tecnologías web. Se podría decir que la aplicación es una especie de contenedor en la cual se incluye nuestra página web móvil con las tecnologías anteriormente mencionadas y la cual tiene acceso a funciones que únicamente las aplicaciones nativas tendrían acceso.



Ilustración . Interacción de una aplicación Phonegap

La siguiente tabla muestra las características que soporta en cada uno de los sistemas operativos.

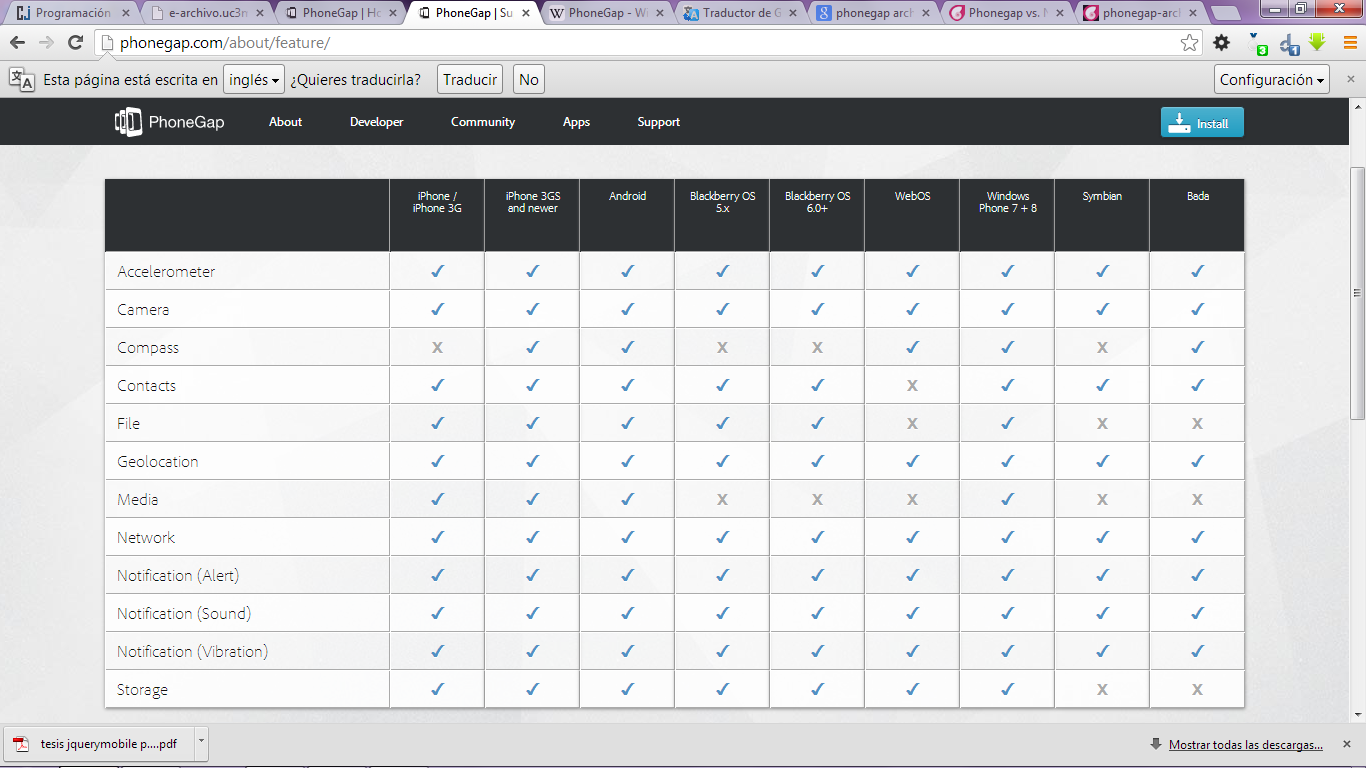


Ilustración . Características disponibles para cada Sistema Operativo

### jQuery Mobile

jQuery Mobile es un framework multiplataforma diseñado para simplificar y mejorar el desarrollo de aplicaciones web para móviles mediante la integración de HTML5, CSS3, JavaScript y jQuery.

El framework es compatible con la mayoría de aplicaciones para PC y móviles, incluyendo iOS, Android, Blackberry, Palm WebOS, Nokia/Symbian, Windows Mobile, Opera Mobile, Firefox Mobile y todos los navegadores modernos como Mozilla Firefox y Google Chrome.

La documentación disponible para jQuery Mobile es muy completa y además incluye algunos ejemplos que permiten familiarizarse rápido con esta herramienta.

jQuery mobile provee archivos CSS que permiten crear la interfaz de usuario de forma muy sencilla y rápida, prácticamente el desarrollador se debe de enfocar en el funcionamiento de la aplicación. jQuery provee botones, barras de navegación, lista, encabezados de página que son muy similares a lo que proveen las herramientas de desarrollo de aplicaciones móviles como Xcode o Microsoft Visual Studio.

Adobe Dreamweaver provee algunas facilidades para desarrollar aplicaciones jQuery Mobile de forma sencilla, en los cuales puedes agregar elementos como cajas de texto o botones a la aplicación mediante un simple click, sin necesidad de escribir el código.



Ilustración . Aplicaciones móviles desarrolladas en jQuery Mobile

# Desarrollo de la aplicación móvil

1. Introducción
2. Planificación Inicial
3. Iteraciones:
   1. Iteración 1.
      1. Planificación
      2. Diseño
      3. Pruebas
   2. Iteración 2.
      1. Planificación
      2. Diseño
      3. Pruebas
   3. Iteración 3.
      1. Planificación
      2. Diseño
      3. Pruebas

## Introducción

Para el desarrollo de esta aplicación móvil se ha elegido la metodología ágil de Programación Extrema. Como se muestra en el capítulo anterior, Programación Extrema es una metodología de carácter iterativo por lo que se ha decidido establecer 3 iteraciones para el desarrollo de esta aplicación. En cada una de estas iteraciones se irá desarrollando y probando algunas de las funcionalidades que permitan realizar facturas electrónicas, una vez concluidas estas iteraciones se obtendrá el producto final.

El proyecto consiste en desarrollar una aplicación móvil que permita generar comprobantes fiscales con Código de Barras Bidimensional (CBB). Esta aplicación permitirá la gestión de los clientes y productos (o servicios) que sean ofrezca un usuario. De esta forma al realizar la factura se podrá realizar rápidamente seleccionando al cliente que se le hará la factura.

Tras analizar las diferentes herramientas que existen para el desarrollo de la aplicación móvil se ha descartado realizarlo en lenguajes como Java y Objective-C, ya que requieren de un tiempo de capacitación para dominarlos y poder realizar la aplicación, por lo que se eligió desarrollarlo bajo el framework PhoneGap (Apache Cordova) y jQuery Mobile, ya que estas herramientas no requieren mucho tiempo de capacitación, además que se podrá generar un producto para las plataformas de iOS y Android permitiendo cubrir gran parte del mercado de las aplicaciones móviles. La aplicación utilizará las tecnologías HTML5, CSS y Javascript y para la gestión de los archivos se utilizará SQLite.

En el apartado de Planificación inicial se muestran las historias que inicialmente estaban previstas realizar en esa fase de desarrollo de este proyecto.

Posteriormente se incluyen cada una de las iteraciones a realizar y los que se hizo en cada una de ellas, con las respectivas tareas a realizar, el diseño y el resultado de las pruebas de aceptación realizadas al usuario.

En la planificación que corresponde a cada iteración se seleccionaron algunas tareas a realizar respecto a las historias de usuario en la planificación inicial. En la fase de diseño se establecieron las soluciones y algunos prototipos de mediana fidelidad de lo que se realizó para saber lo que se codificará.

En la fase de pruebas se realizarán pruebas de aceptación con el usuario esperando detectar posibles errores en el funcionamiento o diseño de la aplicación en cada iteración y corregirlas en la iteración siguiente durante la fase de codificación.

## Planeación Inicial

### Historias de usuario.

En las siguientes historias representan las diferentes funcionalidades que deberá tener la aplicación en cada una se incluye la prioridad durante el desarrollo de la aplicación, el riesgo que existe que afecte el funcionamiento de otras características de la aplicación, el nivel de esfuerzo representado por números (1 = bajo, 2 = medio, 3 = alto), la iteración en la cual será desarrollada y una breve descripción en lo que consistirá esa función.

|  |  |  |
| --- | --- | --- |
| Historia de Usuario | | |
| Número: 1 | Usuario: Cliente | |
| Nombre historia: Gestión de datos de clientes | | |
| Prioridad en negocio:  / Media | | Riesgo en desarrollo:  Baja |
| Puntos estimados: 2 | | Iteración asignada: 1 |
| Programador responsable: Ricardo Cibrián | | |
| Descripción: Se podrá gestionar los datos de clientes a los cuales se realizará la factura (alta, baja y modificación). | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| Historia de Usuario | | |
| Número: 2 | Usuario: Cliente | |
| Nombre historia: Gestión de los productos y servicios | | |
| Prioridad en negocio:  / Media | | Riesgo en desarrollo:  Baja |
| Puntos estimados: 2 | | Iteración asignada: 1 |
| Programador responsable: Ricardo Cibrián | | |
| Descripción: Se podrá gestionar información de los productos o servicios que se ofrezcan (alta, baja y modificación). | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| Historia de Usuario | | |
| Número: 3 | Usuario: Cliente | |
| Nombre historia: Generación de la factura | | |
| Prioridad en negocio:  Alta | | Riesgo en desarrollo:  Bajo |
| Puntos estimados: 2 | | Iteración asignada: 2 |
| Programador responsable: Ricardo Cibrián | | |
| Descripción: Se generan las facturas mediante un formulario. Se seleccionará al cliente y posteriormente se agregará los productos o servicios que ha recibido. | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| Historia de Usuario | | |
| Número: 4 | Usuario: Cliente | |
| Nombre historia: Enviar factura por Email. | | |
| Prioridad en negocio:  Alta | | Riesgo en desarrollo:  Bajo |
| Puntos estimados: 3 | | Iteración asignada: 3 |
| Programador responsable: Ricardo Cibrián | | |
| Descripción: Una vez creada la factura se podrá adjuntar en un correo electrónico para enviarla a un cliente. | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| Historia de Usuario | | |
| Número: 5 | Usuario: Cliente | |
| Nombre historia: Validación del código CBB (Código de Barras Bidimensional). | | |
| Prioridad en negocio:  Alta | | Riesgo en desarrollo:  Bajo |
| Puntos estimados: 3 | | Iteración asignada: 3 |
| Programador responsable: Ricardo Cibrián | | |
| Descripción: Se validará el código CBB del usuario, para evitar generar facturas que no tengan validez ante SAT. | | |
|  | | |
| |  |  |  | | --- | --- | --- | | Historia de Usuario | | | | Número: 6 | Usuario: Cliente | | | Nombre historia: Control de acceso de usuarios. | | | | Prioridad en negocio:  Baja | | Riesgo en desarrollo:  Bajo | | Puntos estimados: 1 | | Iteración asignada: 2 | | Programador responsable: Ricardo Cibrián | | | | Descripción: Antes de utilizar la aplicación será necesario introducir un nuevo de usuario y su respectiva contraseña. | | | | |  |  |  | | --- | --- | --- | | Historia de Usuario | | | | Número: 7 | Usuario: Cliente | | | Nombre historia: Gestión de los datos personales del usuario. | | | | Prioridad en negocio:  Baja | | Riesgo en desarrollo:  Bajo | | Puntos estimados: 2 | | Iteración asignada: 2 | | Programador responsable: Ricardo Cibrián | | | | Descripción: Gestionar la información del usuario que expide los comprobantes. Es necesario para generar las facturas. | | | | |  |  |  | | --- | --- | --- | | Historia de Usuario | | | | Número: 8 | Usuario: Cliente | | | Nombre historia: Gestión el valor del IVA | | | | Prioridad en negocio:  Baja | | Riesgo en desarrollo:  Bajo | | Puntos estimados: 1 | | Iteración asignada: 2 | | Programador responsable: Ricardo Cibrián | | | | Descripción: Gestionar la información del usuario que expide los comprobantes. Es necesario para generar las facturas. | | | | |  |  |  | | --- | --- | --- | | Historia de Usuario | | | | Número: 9 | Usuario: Cliente | | | Nombre historia: Manual de usuario | | | | Prioridad en negocio:  Baja | | Riesgo en desarrollo:  Bajo | | Puntos estimados: 1 | | Iteración asignada: 2 | | Programador responsable: Ricardo Cibrián | | | | Descripción: Para no requerir algún tipo de capacitación se incluirán unos videotutoriales para la realización de las tareas en la aplicación. | | | | IteracionesIteración 1.Planificación. De acuerdo a la planificación inicial se determinó realizar las tareas correspondientes a las dos primeras historias de usuario (Gestión de los clientes y gestión de los usuarios).  Se han establecido las siguientes tareas para la historia de usuario 1.   |  |  | | --- | --- | | Tarea | | | Número tarea: 1 | Número historia: 1 | | Descripción: Interfaz de usuario para agregar clientes. Se agregará una barra de navegación, botones para agregar el usuario y cajas de texto para introducir los datos del cliente que se agregará. | | | Tipo de tarea :  Desarrollo | Puntos estimados: 1 | | Fecha inicio: 12/09/2013 | Fecha fin: 12/09/2013 | | |  |  | | --- | --- | | Tarea | | | Número tarea: 2 | Número historia: 1 | | Descripción: Realizar interfaz para el listado de los usuarios. | | | Tipo de tarea :  Desarrollo | Puntos estimados: 1 | | Fecha inicio: 12/09/2013 | Fecha fin: 12/09/2013 | |  | | | |  |  |  | | --- | --- | | Tarea | | | Número tarea: 3 | Número historia: 1 | | Descripción: Realizar interfaz para la modificación del usuario. Cajas de texto que permitirán modificar los datos del usuario seleccionado, además contará con un botón para eliminarlo. | | | Tipo de tarea :  Desarrollo | Puntos estimados: 1 | | Fecha inicio: 12/09/2013 | Fecha fin: 12/09/2013 | | |  |  | | --- | --- | | Tarea | | | Número tarea: 4 | Número historia: 1 | | Descripción de la tarea: Generar el script para la generación de la base de datos de clientes y realizar los scripts correspondientes para insertar, modificar, eliminar y mostrar datos de los clientes. | | | Tipo de tarea :  Desarrollo | Puntos estimados: 1 | | Fecha inicio: 13/09/2013 | Fecha fin: 13/09/2013 | | Tareas definidas para la historia de usuario 2.   |  |  | | --- | --- | | Tarea | | | Número tarea: 1 | Número historia: 2 | | Descripción: Interfaz de usuario para agregar productos. Se agregará una barra de navegación, botones para agregar el producto y cajas de texto para introducir los datos del producto que se agregará. | | | Tipo de tarea :  Desarrollo | Puntos estimados: 1 | | Fecha inicio: 14/09/2013 | Fecha fin: 14/09/2013 | | |  |  | | --- | --- | | Tarea | | | Número tarea: 2 | Número historia: 2 | | Descripción: Realizar interfaz para el listado de los productos | | | Tipo de tarea :  Desarrollo | Puntos estimados: 1 | | Fecha inicio: 14/09/2013 | Fecha fin: 14/09/2013 | | |  |  | | --- | --- | | Tarea | | | Número tarea: 3 | Número historia: 2 | | Descripción: Realizar interfaz para la modificación de un producto. Se incluyen cajas de texto que permitirán modificar los datos del producto seleccionado, además contará con un botón para eliminarlo o actualizarlo. | | | Tipo de tarea :  Desarrollo | Puntos estimados: 1 | | Fecha inicio: 14/09/2013 | Fecha fin: 42/09/2013 | | |  |  | | --- | --- | | Tarea | | | Número tarea: 4 | Número historia: 2 | | Descripción de la tarea: Generar el script para la generación de la base de datos de productos y realizar los scripts correspondientes para insertar, modificar, eliminar y mostrar datos de los productos. | | | Tipo de tarea :  Desarrollo | Puntos estimados: 1 | | Fecha inicio: 13/09/2013 | Fecha fin: 13/09/2013 | |  | | | | | | | | | | | | | | | | | |  Diseño | | |   De acuerdo a las tareas determinadas para cada una de las historias de usuario presentadas se han desarrollado los siguientes prototipos de mediana fidelidad de las funcionalidades de la aplicación.    Ilustración . Listado de productos  C:\Users\Ricardo\Documents\Tesis v2\Mockups Tesis\clientesMockup.png  Ilustración . Listado de clientes    Ilustración . Formulario para insertar clientes | | | | | |



Ilustración . Formulario para insertar productos

#### Pruebas

Una vez finalizada la fase de codificación se realizarán las pruebas funcionales relacionadas con las tareas se llevaron acabo en cada historia de usuario (historia 1 e historia 2)

***Esta sección se documenta el conjunto de pruebas funcionales relacionadas con la historia de usuario: Gestión de datos de cliente [Historia 1].***

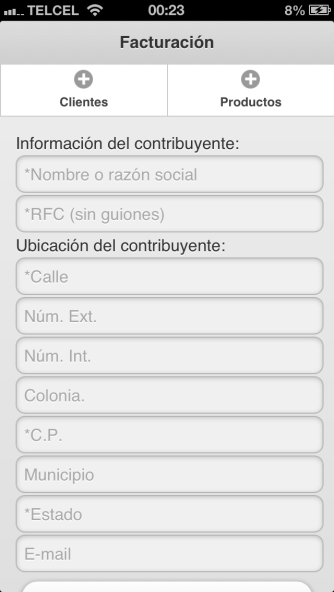


Ilustración . Módulo "Agregar cliente"

Se comprobará si se agrega correctamente el registro del nuevo cliente en la base de datos.

Descripción.

Una vez que el usuario entre a la aplicación se le mostrará el formulario para agregar un nuevo cliente. El usuario ingresará los datos en los campos obligatorios y oprimirá el botón de agregar. Si el registro se ha agregado correctamente mostrará un mensaje que indicando que se ha creado un nuevo cliente.

Condiciones de ejecución.

Ninguna.

Entrada

El usuario introducirá los datos del cliente que desea agregar.

Oprimirá el botón de agregar.

Resultado esperado

Tras haber introducido los datos en los campos de texto y haber agregado los datos. Se habrán limpiado los campos de texto y se habrá agregado el nuevo registro en la base de datos. El nuevo registro será desplegado en la sección de *Listado de clientes.*

Evaluación de la prueba

Prueba satisfactoria.

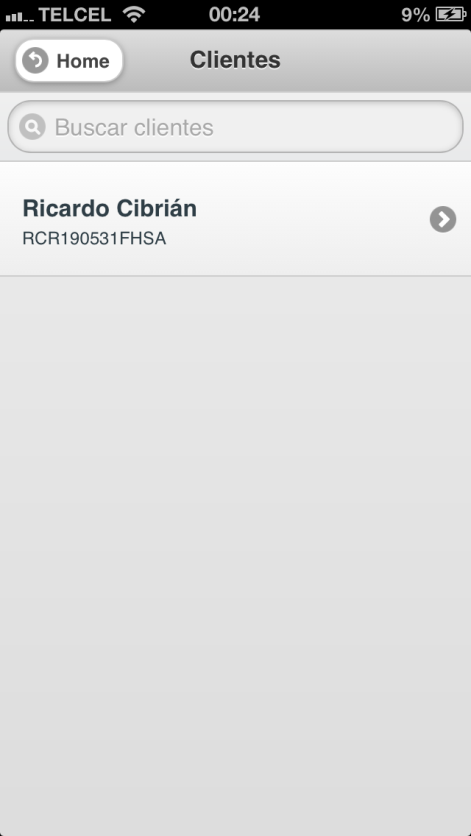


Ilustración . Listado de clientes

***Esta sección se documenta el conjunto de pruebas funcionales relacionadas con la historia de usuario: Gestión de los productos y servicios [Historia 2].***



Ilustración . Módulo de "Agregar producto"

Se comprobará si se agrega correctamente el registro de un nuevo producto en la base de datos.

Descripción.

Una vez que el usuario entre a la aplicación se le mostrará el formulario para agregar un nuevo cliente. Seleccionará el botón de productos que se presenta en el encabezado. Será desplegado al usuario el formulario donde ingresará los datos de un producto y posteriormente oprimirá el botón de agregar. Si el registro se ha agregado correctamente mostrará un mensaje que indicando que se ha creado un nuevo producto.

Condiciones de ejecución.

Ninguna.

Entrada

El usuario introducirá los datos del producto que desea agregar.

Oprimirá el botón de agregar.

Resultado esperado

Tras haber introducido los datos del producto en los campos de texto y haber presionado el botón. Se habrán limpiado los campos de texto y se habrá agregado el nuevo registro en la base de datos. El nuevo registro será desplegado en la sección de *Listado de productos*

Evaluación de la prueba

Prueba satisfactoria.

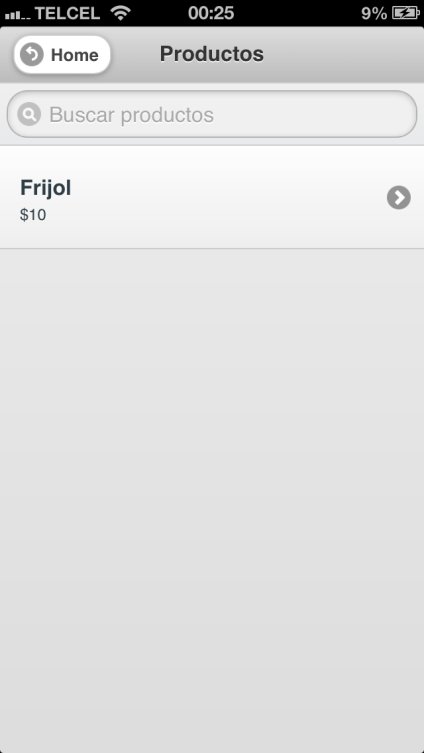


Ilustración . Listado de productos

# Bibliografía

*Micronavegador*. (9 de Marzo de 2013). Recuperado el 11 de Marzo de 2013, de Wikipedia, la enciclopedia libre: http://es.wikipedia.org/wiki/Micronavegador

ABC.es. (22 de Febrero de 2012). *IBM Simon, el primer smartphone de la historia.* Recuperado el 13 de Marzo de 2013, de ABC.es Tecnología: http://www.abc.es/20120222/tecnologia/abci-simon-primer-smartphone-historia-201202221308.html

Alex. (1 de Octubre de 2011). *La primera red móvil comercial del mundo NMT cumple 30 años*. Recuperado el 12 de Marzo de 2013, de movilToday: http://moviltoday.com/la-primera-red-movil-comercial-del-mundo-nmt-cumple-30-anos/

antocara. (19 de Octubre de 2012). *¿Qué es Android?, historia, conceptos y evolución del sistema operativo para dispositivos móviles*. Recuperado el 13 de Marzo de 2013, de ActualidadG: http://actualidadg.com/2012/10/que-es-android-historia-conceptos-y-evolucion-del-sistema-operativo-para-dispositivos-moviles/

Basterretche, J. F. (2007). *Dispositivos Móviles.* Recuperado el 12 de Marzo de 2013, de Facultad de Ciencias Exactas y Naturales y Agrimensura: http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/tfbasterretche.pdf

*Características de Java*. (s.f.). Recuperado el 14 de Marzo de 2013, de Web Taller: http://www.webtaller.com/manual-java/caracteristicas-java.php

Coad, P., Lefebvre, E., & De Luca, J. (1999). *Java Modeling In Color With UML: Enterprise Components and Process.* Prentice Hall.

Cockbun, A. (2001). *Agile Software Development.* Addison-Wesley.

Competitive Intelligence Unit. (2012). *Competencia en Mercado de Smartphones*. Recuperado el 25 de Febrero de 2013, de CIU: http://www.the-ciu.net/nwsltr/029\_1Smartphones.html

Conde, R. (s.f.). *¿Qué significan 1G, 2G, 3G y 4G?* Recuperado el 10 de Marzo de 2013, de About en Español: http://celulares.about.com/od/Preguntas\_frecuentes/a/Que-Significan-1g-2g-3g-Y-4g.htm

Condesa. (4 de Julio de 2011). *Arquitectura de Android*. Recuperado el 13 de Marzo de 2013, de Androideity: http://androideity.com/2011/07/04/arquitectura-de-android/

*Eclipse*. (s.f.). Recuperado el 14 de Marzo de 2013, de Ecured: http://www.ecured.cu/index.php/Eclipse,\_entorno\_de\_desarrollo\_integrado

el-mexicano. (Marzo de 2012). *Microempresas son base de la economía mexicana.* Recuperado el 14 de Marzo de 2013, de Pepe y Toño: http://pepeytono.com.mx/novedades/2012/03/microempresas\_son\_base\_de\_la\_economia\_mexicana

Fernández, N. C. (s.f.). *VoCSDroid.* Recuperado el 14 de Marzo de 2013, de Universidad Carlos III de Madrid: http://e-archivo.uc3m.es/bitstream/10016/16337/1/TFG\_Nicolas\_Cordero\_Fernandez.pdf

Galea, A. L. (Septiembre de 2009). *Evolución de las redes de telefonía móvil.* Recuperado el 12 de Marzo de 2013, de Consulta Teleco: www.consultateleco.es/periodistas/descargar.php?id\_media=83

González, J. C. (15 de Agosto de 2012). *Samsung y Apple consolidan su dominio: así está el mercado de telefonía*. Recuperado el 6 de Agosto de 2013, de Xataka: http://www.xataka.com/moviles/un-repaso-al-estado-del-mercado-de-telefonia-movil-en-el-segundo-trimestre-del-2012

Gordillo, P. M. (Julio de 2010). USO DE PATRONES DE PRODUCTO EN METODOLOGÍAS ÁGILES. Madrid, España.

GSMspain. (s.f.). *Glosario*. Recuperado el 12 de Marzo de 2013, de GSMspain: http://www.gsmspain.com/glosario/?palabra=AMPS

Gutiérrez, L. M. (Agosto de 2012). Nuevo Esquema de Facturación en las Microempresas. Xalapa-Enríquez, Veracruz, México.

Hernández, F. L. (2012). *Objective-C - Curso Práctico Para Programadores De Mac OS X, Iphone Y Ipad.* Alfaomega.

Highsmith, J., & Orr, K. (2000). *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems.* Dorset House.

INEGI. (21 de Septiembre de 2010). *INEGI PRESENTA RESULTADOS DEFINITIVOS DE LOS CENSOS ECONÓMICOS 2009*. Recuperado el 17 de Abril de 2013, de INSTITUTO NACIONAL DE ESTADÍSTICA Y GEOGRAFÍA: http://www.inegi.org.mx/inegi/contenidos/espanol/prensa/comunicados/censoseconomicos09.asp

JCANEZ. (12 de Enero de 2012). *El Windows Phone no es lo mismo que el Windows Mobile*. Recuperado el 14 de Marzo de 2013, de Nokia Conversaciones: http://conversaciones.nokia.com/2012/01/12/el-windows-phone-no-es-lo-mismo-que-el-windows-mobile/

Kioskea. (Marzo de 2013). *Telefonía móvil*. Recuperado el 12 de Marzo de 2013, de Kioskea.net: http://es.kioskea.net/contents/telephonie-mobile/reseaux-mobiles.php3

Microsoft. (30 de Octubre de 2012). *SDK de Windows Phone 8.0.* Recuperado el 14 de Marzo de 2013, de Download Center: https://www.microsoft.com/es-mx/download/details.aspx?id=35471

Microsoft. (s.f.). *Características*. Recuperado el 14 de Marzo de 2013, de Windows Phone: http://www.windowsphone.com/es-es/features

Morata, J. Q. (Diciembre de 2011). *Desarrollo de una aplicación distribuida para dispositivos iOS.* Recuperado el 13 de Marzo de 2013, de Universidad Politécnica de Valencia: http://riunet.upv.es/bitstream/handle/10251/15598/Tesis%20Javier%20Querol.pdf?sequence=1

Muñoz, D. (21 de Octubre de 2011). *Historia del software: iOS*. Recuperado el 13 de Marzo de 2013, de Bitelia: http://bitelia.com/2011/10/historia-del-software-ios

Paz, C. A. (15 de Septiembre de 2006). *Primeros pasos con XAML*. Recuperado el 14 de Marzo de 2013, de el Guille: http://www.elguille.info/colabora/NET2006/SnowMk\_Hola\_Mundo\_XAML.htm

Pérez P., C., & Verbakel V., G. (Junio de 2009). *Desarrollo de interfases.* Recuperado el 6 de Marzo de 2013, de Usando.info: http://www.usando.info/docs/udd\_informe\_dispositivosmoviles\_2009.pdf

Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit for Software Development Managers.* Addison Wesley.

Ramón, J. (13 de Noviembre de 2012). *Pantalla smartphone*. Recuperado el 6 de Marzo de 2013, de Aplicaciones para móviles: http://blog-apps.euroresidentes.es/2012/11/pantalla-smartphone.html

RedUSERS. (s.f.). *El lenguaje .Net.* Recuperado el 14 de Marzo de 2013, de RedUSERS: http://img.redusers.com/imagenes/libros/lpcu143/capitulogratis.pdf

Rivas, J. R. (18 de Octubre de 2012). *La Arquitectura de Android*. Recuperado el 13 de Marzo de 2013, de Catédras Teléfonica: http://catedratelefonica.ulpgc.es/blog/arucas/2012/10/18/la-arquitectura-de-android/

Rodríguez, E. F. (3 de Junio de 2011). *iOS.* Recuperado el 13 de Marzo de 2012, de Steve Jobs: http://luxstevejobs.comxa.com/obra1.html

Rodríguez, J. (s.f.). *Breve historia de los Smartphones.* Recuperado el 13 de Marzo de 2013, de Somosgeeks.net: https://somosgeeks.wordpress.com/2011/01/17/breve-historia-de-los-smartphones/

Rojas Alcocer, A., Durán Hernández, A., & Castro Koschny, D. (30 de Marzo de 2012). *Características y evolución del iOS.* Recuperado el 13 de Marzo de 2013, de TEC Digital: http://www.tec-digital.itcr.ac.cr/file/3688258/iOS.pdf

SAT. (8 de octubre de 2010). *ETAPA DE TRANSICIÓN*. Recuperado el 24 de Abril de 2013, de SAT: http://www.sat.gob.mx/sitio\_internet/asistencia\_contribuyente/principiantes/comprobantes\_fiscales/66\_18807.html

SAT. (1 de Julio de 2011). *Guía para la emisión de Comprobantes Fiscales Impresos con CBB.* Recuperado el 14 de Marzo de 2013, de Servicio de Administración Tributaria: ftp://ftp2.sat.gob.mx/asistencia\_ftp/publicaciones/.../comprobantesfiscalesCBB/GuiaEmisionCFCBB.pdf

SAT. (20 de Diciembre de 2011). *INFRACCIÓN Y SANCIÓN POR NO ENTREGAR O NO ENVIAR LOS COMPROBANTES FISCALES*. Recuperado el 19 de Abril de 2013, de SAT: http://www.sat.gob.mx/fiscal/rf/157\_22053.html

SAT. (22 de Noviembre de 2012). *SOLO QUEDAN 5 SEMANAS PARA MIGRAR A FACTURA .* Recuperado el 2013 de Febrero de 20, de SAT: ftp://ftp2.sat.gob.mx/asistencia\_servicio\_ftp/publicaciones/boletines/com2012\_120.pdf

SAT. (10 de enero de 2013). *Factura Electrónica*. Recuperado el 22 de Febrero de 2013, de Secretaría de Administración Tributaria: http://www.sat.gob.mx/sitio\_internet/asistencia\_contribuyente/principiantes/comprobantes\_fiscales/66\_20014.html

SAT. (28 de Enero de 2013). *Glosario*. Recuperado el 14 de Marzo de 2013, de Servicio de Administración Tributaria: http://www.sat.gob.mx/sitio\_internet/21\_6485.html

SAT. (10 de Enero de 2013,). *INFORMACIÓN GENERAL DE LAS FACTURAS IMPRESAS CON CBB*. Recuperado el 14 de Marzo de 2013, de Servicio de Administración Tributaria: http://www.sat.gob.mx/sitio\_internet/asistencia\_contribuyente/principiantes/comprobantes\_fiscales/66\_19204.html

SAT. (s.f.). *Guía para la emisión de Comprobantes Fiscales Impresos con CBB.* Recuperado el 14 de Marzo de 2013, de Servicio de Administración Tributaria: ftp://ftp2.sat.gob.mx/asistencia\_ftp/publicaciones/.../comprobantesfiscalesCBB/GuiaEmisionCFCBB.pdf

Schwaber K., B. M. (2001). *Agile Software Development with SCRUM.* Prentice Hall.

SHCP. (9 de Marzo de 2011). *Factura Electrónica.* Recuperado el 14 de Marzo de 2013, de Info-Finanzas: http://www.info-finanzas.com/upload/CyT/SAT-2011.pdf

Stapleton, J. (1997). *Dsdm Dynamic Systems Development Method: The Method in Practice.* Addison-Wesley.

Tecnologia Hecha Palabra. (25 de Febrero de 2013). *Hay 150 redes LTE comerciales en 67 países; 250+ redes LTE previstas para fin de 2013.* Recuperado el 12 de Marzo de 2013, de Tecnologia Hecha Palabra.

Telcel. (s.f.). *Cobertura.* Recuperado el 12 de Marzo de 2013, de 4GLTE: http://www.telcel.com/4GLTE/cobertura.html

Universidad del Valle. (16 de Diciembre de 2011). *Programación Extrema.* Recuperado el 14 de Marzo de 2012, de Universidad del Valle: http://eisc.univalle.edu.co/materias/WWW/material/lecturas/

Vilchez, A. (2 de Abril de 2009 ). *Que es Android: Características y Aplicaciones*. Recuperado el 13 de Marzo de 2013, de Configurar Equipos: http://www.configurarequipos.com/doc1107.html

Wells, D. (Septiembre de 2009). *Extreme Programming: A gentle introduction*. Recuperado el 30 de Marzo de 2013, de Extreme Programming: http://www.extremeprogramming.org/

World, P. (21 de Junio de 2012). *Las mejores nuevas características de Windows Phone 8* . Recuperado el 14 de Marzo de 2013, de PC World.