

TFC - MEMORIA

App ControlStock.



TFC- Desarrollo de aplicaciones para dispositivos móviles –
Memoria, aplicación de Control de Stock.

Marcos Benito Gracia

Ingeniería Técnica en Informática de Sistemas

Consultor: Joan Vicent Orenge Serisuelo

Primavera 2013

Agradecimiento a mi familia y amigos por el apoyo que me ha dado y sobre todo a mi esposa Evelyn que me ha dado animo y me ha ayudado a superar algunos momento de desesperación, gracias y disculpa por las horas de soledad que te hice pasar.

Contenido

1. GUÍA DE TRABAJO	6
1.1 INTRODUCCIÓN.....	6
1.2 OBJETIVO Y ALCANCE DEL PROYECTO:	6
1.3 REQUISITOS.....	7
1.4 FORMACIÓN PREVIA AL ENTORNO	7
1.5 FASES	8
1.5.1 Análisis Previo.....	9
1.5.2 Análisis.....	9
1.5.3 Diseño.....	10
1.5.4 Programación.....	10
1.5.6 Prueba Test.....	11
1.6 DIAGRAMA DE GANTT	13
2. DISEÑO CENTRADO EN EL USUARIO.....	14
2.1 INTRODUCCIÓN.....	14
2.2 OBJETIVO	14
2.3 USUARIOS Y CONTEXTO DE USOS	14
2.3.1 Métodos:.....	14
2.3.2 Análisis de tareas.....	17
2.3.3 Análisis competitivo	18
2.4 DISEÑO CONCEPTUAL	19
2.4.1 Escenarios de uso	19
2.4.2 Flujos de interacción.....	21
2.5 PROTOTIPO.....	23
2.5.1 Prototipo de pantallas	23
2.5.2 Primeras ideas de diseño.....	23
2.5.3 Influencias sobre el diseño.....	26
3. ARQUITECTURA E IMPLEMENTACIÓN	27
3.1 INTRODUCCIÓN.....	27
3.2 OBJETIVO	27
3.3 PLATAFORMA IOS	27
3.4 REQUISITOS DE DESARROLLO.....	28
3.5 CONTENIDO	28
3.6 MÓDULOS DE APLICACIÓN	39
3.6.1 Navigation Controller	39
3.6.2 View Controller- Login.....	39
3.6.3 Table View Controller– ControlStock.....	39
<i>En este módulo como menú de posibilidades de la aplicación.</i>	39
<i>Aparecerá las distintas actividades a realizar, posteriormente bajo el menú de acciones aparecerá en la versión final el logotipo de la aplicación junto con la información de la versión.</i>	39
3.6.4 View Controller – Insertar Producto.....	40
3.6.5 Main Table View Controller – Listar Productos.....	40
3.6.6 View Controller – Eliminar Producto.....	40
3.6.7 View Controller – Buscar Producto.....	40
3.6.8 View Controller – Modificar Producto	40
3.7 PERSISTENCIA.....	41
3.7.1 Persistencia Local.....	41
3.7.2 Persistencia Remota.....	41
3.8 PRUEBAS Y TEST DE APLICACIÓN	42
3.8.1 Pruebas de usabilidad manuales	42
3.8.2 Pruebas de test unitarias	42
4. RESULTADOS	43
4.1 EXPERIENCIAS	43
4.2 CRÍTICAS Y MEJORAS	43
4.3 CONCLUSIONES.....	44
5. BIBLIOGRAFÍA	45

Ilustraciones

ILUSTRACIÓN 1. ARQUITECTURA FÍSICA	7
ILUSTRACIÓN 2. FASES.....	8
ILUSTRACIÓN 3. DIAGRAMA DE GANTT, FASES DE TRABAJO	13
ILUSTRACIÓN 4. INSERTAR PRODUCTO.....	21
ILUSTRACIÓN 5. CONSULTA/MODIFICACIÓN DE UN PRODUCTO.....	22
ILUSTRACIÓN 6. ORGANIGRAMA PERFIL ADMINISTRACIÓN.....	23
ILUSTRACIÓN 7. ORGANIGRAMA PERFIL ALMACÉN.....	24
ILUSTRACIÓN 8. ORGANIGRAMA PERFIL COMERCIAL	25
ILUSTRACIÓN 9. ORGANIGRAMA PLATAFORMA IOS	27
ILUSTRACIÓN 10. ARQUITECTURA DE CAPAS IOS.....	38
ILUSTRACIÓN 11. DIAGRAMA DE MÓDULOS	39
ILUSTRACIÓN 12. ESQUEMA DE PERSISTENCIA REMOTA	41
ILUSTRACIÓN 13. ESTRUCTURA DE LA TABLA USER DE PRODUCTOS	41
ILUSTRACIÓN 14. ESTRUCTURA DE LA TABLA LOGIN DE USUARIOS.....	41

Introducción

En este documento se va hacer un repaso sobre todo el trabajo realizado para el proyecto de final de carrera dando una explicación de los diferentes puntos que lo forman, justificando la realización del mismo, realizando una crítica con propuestas de mejoras y explicando experiencias y aptitudes conseguidas.

Resumen

El desarrollo de aplicaciones móviles es una de las salidas profesionales con más auge en la actualidad y tal y como se mueve el mercado y la venta de dispositivos lo será también en el futuro.

En este proyecto realizamos una aplicación para el sistema operativo IOS desarrollado por Apple, uno de los sistemas móviles más utilizados y en la actualidad el que más aplicaciones tiene.

El objetivo de este proyecto ha sido realizar una aplicación que aprovecha muchas de las ventajas que nos ofrecen las tecnologías actuales y que nos permiten desarrollar aplicaciones que faciliten la vida de los usuarios y le den un valor añadido a los dispositivos que las utilizan, así como tener un primer contacto con el desarrollo de aplicaciones para dispositivos móviles y conseguir experiencias y aptitudes.

En el proyecto valoramos el tiempo y los recursos necesarios para la realización del proyecto, realizamos un estudio de diseño muy diferente al que utilizaríamos para otro tipo de aplicación basándonos en el usuario con DCU(diseño centrado en el usuario), y finalmente realizamos la implementación del proyecto utilizando en la tecnología IOS con la intención de aprender y aprovechar todas sus ventajas teniendo siempre en mente al usuario final.

1. Guía de trabajo

1.1 Introducción

Este apartado nos ha servido de guía para el desarrollo del trabajo de final de carrera.

La idea de este apartado es que durante el desarrollo nos sirva como el guión del trabajo que tenemos que realizar.

En este apartado vamos a ver una pequeña especificación del proyecto y las diferentes fases que llevaremos a cabo durante el desarrollo, indicaremos que documentos entregaremos en cada fase y los objetivos de las mismas.

Para el desarrollo de las fases de este proyecto nos basaremos en un modelo en cascada combinado con un diseño centrado en el usuario.

1.2 Objetivo y alcance del proyecto:

El objeto de este proyecto es realizar una aplicación para dispositivos Iphone que nos permitirá realizar un control del stock de un almacén.

Esta aplicación permite adaptarse a las necesidades de su empresa, Interfaz gráfica de usuario amigable, sencilla e intuitiva, Múltiples reportes que le facilitan el control de las existencias tanto de una misma ubicación, como de otros depósitos o almacenes de su empresa, Si tiene productos con determinado número de unidades, características, precio, etc.

Funcionalidad

En la aplicación podrán acceder tres tipos diferentes de usuarios, por un lado podemos entrar como empresa que hace uso de la aplicación para gestionar los tipos de productos que se adquieren con sus características por otro lado el usuario del almacén podrá gestionar el stock, y por otro lado podemos entrar como comercial que desea realizar consultas para ver los determinados productos y sus características.

Funcionalidad para la administración del almacén

El administrativo podrá gestionar su stock para ver las adquisiciones realizadas por los usuarios y podrá realizar el control de las existencias de los productos almacenados de su empresa.

De esa manera la aplicación será su herramienta para gestionar los almacenes sin moverse de su ubicación.

1.3 Requisitos

Funcionales:

Dispositivo con sistema operativo IOS y acceso a internet.

Técnicos:

Herramientas apropiadas para el desarrollo de la aplicación, drivers, máquinas virtuales, kits de desarrollo etc.

Una base de datos donde se guardan los datos de las empresas que estén dadas de alta, y los datos de los usuarios registrados.

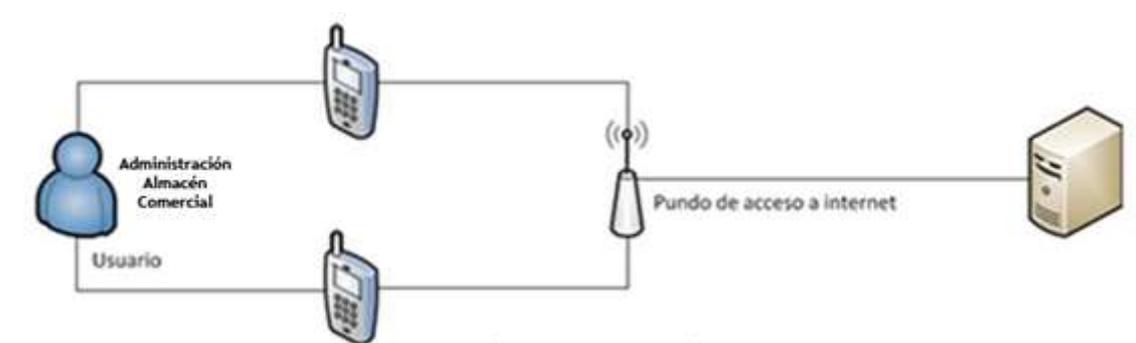


Ilustración 1. Arquitectura física

1.4 Formación previa al entorno

Al tratarse de un proyecto donde se utilizarán nuevas tecnologías antes de poder realizar las fases iniciales del proyecto será conveniente tener una formación básica de dichas tecnologías para afrontar mejor el análisis.

Objetivo:

El objetivo es conocer los kits de desarrollo (SDK, Xcode) para Iphone, como lanzar una aplicación piloto y debugar sobre un dispositivo físico, conocer las arquitecturas y el esqueleto básico de una aplicación desarrollada para un sistema operativo Iphone así como las Apis más importantes y comunes para el desarrollo de la aplicación, y conocer cómo trabaja a nivel de archivos, datos, accesos, hardware etc.

Realizar un pequeño estudio de información que podemos encontrar tanto en internet como en libros que nos puedan servir de ayuda y apoyo durante la realización del proyecto.

1.5 Fases

Para poder seguir un orden en la elaboración del proyecto seguiremos las siguientes fases:

Análisis Previo, Análisis (Planificación y especificación), Diseño, Programación (Implementación), Prueba (Test) y documentación final.

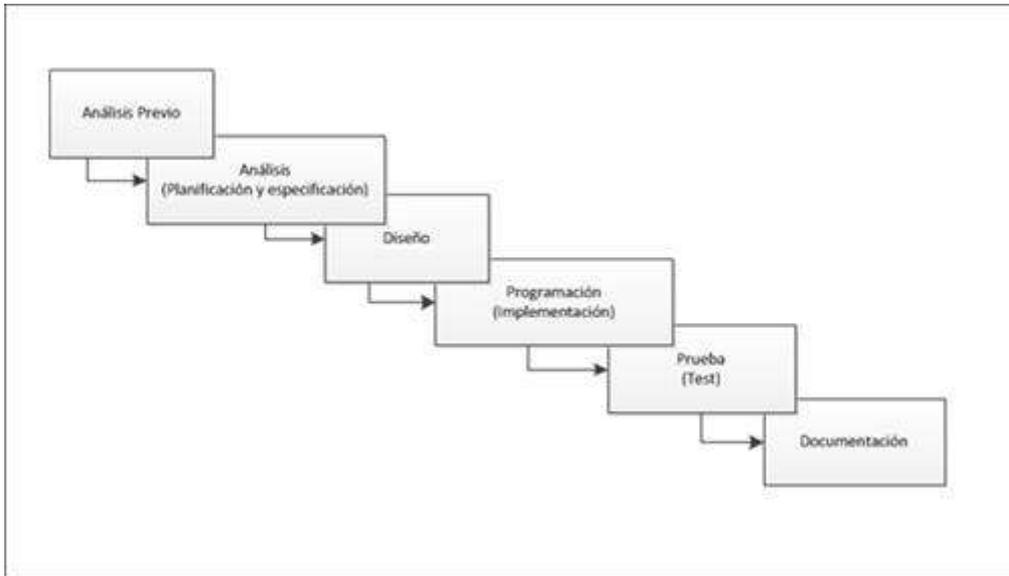


Ilustración 2. Fases

1.5.1 Análisis Previo

Objetivo:

Los objetivos de esta fase son definir la idea y el propósito de la aplicación con el fin de tener un punto claro de partida sobre el trabajo que se va a realizar y sobre que sistemas y quien lo va a utilizar.

Documentación:

La documentación de esta fase constará de un resumen donde se explicará la idea básica de la aplicación y los requisitos mínimos para el desarrollo del proyecto y las personas que participarán.

Finalidad y riesgos:

Este resumen nos tiene que servir para poder consultar cual es la idea del proyecto a grandes rasgos y tener un punto de partida sobre lo que vamos a trabajar.

1.5.2 Análisis

Objetivo:

El objetivo de esta fase es definir con detalle la funcionalidad de la aplicación, esta tiene 3 puntos claros de funcionalidad que son la parte de administración, almacén y comercial en esta fase especificaremos detalladamente la funcionalidad de cada perfil, tendremos que realizar una especificación clara de la funcionalidad detallando cada caso de uso con esquemas teniendo en cuenta las personas que utilizarán la aplicación y el propósito de la misma.

También definiremos las pantallas sobre las que interactuarán los usuarios de la aplicación, para tener una visión clara de la función de cada pantalla.

Documentación:

Resumen detallado de la funcionalidad.

Diagramas de casos de uso de los usuarios.

Imágenes donde se mostrará el diseño de las pantallas con los controles correspondientes para su utilización tanto para los casos de empresa como de usuario.

Finalidad y riesgos:

Esta documentación nos servirá para enfocar la fase de diseño y la fase de implantación teniendo en cuenta la explicación de la funcionalidad, casos de uso y pantallas funcionales.

1.5.3 Diseño

Objetivo:

Definir a partir de los datos de la fase de análisis la arquitectura que tendrá la aplicación para poder resolver la funcionalidad.

Tenemos que especificar como resolveremos funcionalmente los siguientes puntos:

Tipos de usuario y acceso de los mismo a la aplicación.

Accesos a bases de datos para altas, bajas, modificaciones, consultas de productos.

Funcionalidad de escaneo de códigos de barra, para una adecuada usabilidad y un ágil acceso a los productos.

Definición de tipo de controles utilizados para mostrar correctamente la información y que el uso funcional de la aplicación sea intuitivo.

Otro de los puntos que hay que definir en esta fase es realizar un estudio sobre Frameworks usados para el parseo de datos y posterior almacenamiento en la BBDD de internet y ver si es una buena opción para almacenar información.

Documentos:

Documentó de texto donde se especificará las diferentes funciones más importantes de la aplicación.

Escenarios de uso.

Resumen de personajes.

Diagramas de flujo.

Prototipos de pantallas.

Finalidad y riesgos:

Los datos recogidos en esta fase y los documentos elaborados nos servirán para realizar la fase de programación y poder detectar complicaciones que nos retrasen en el tiempo y evaluar la complejidad de algunas funcionalidades.

1.5.4 Programación

Objetivo:

El objetivo de esta fase es implementar el código la especificación y arquitectura definida en la fase de diseño.

En esta fase montaremos los diferentes módulos o clases que utilizaremos para la realización de la aplicación, programaremos sus funciones internas, y pondremos en práctica el uso de los Frameworks aprendidos durante la fase de formación para los diferentes puntos de la aplicación como por ejemplo el uso de los lectores de códigos de barra, los accesos a datos, control del hardware del dispositivo, etc.

Se diseñará la parte de interface gráfica para utilizar gráficos que sean intuitivos en la aplicación y que tengan resoluciones gráficas que sean fáciles de ajustar a diferentes tipos de dispositivos Iphone.

Documentos:

Archivos necesarios para realizar una instalación. Documentó de arquitectura.

Código fuente.

Manual de instalación

Finalidad y riesgos:

Durante el desarrollo de esta fase tendremos que contemplar que los tiempos pueden ser muy variables según la dificultad que nos encontremos a la hora de utilizar Apis, solucionar problemas que no se vieron en el diseño, o problemas típicos a la hora de programar, por eso hay que evaluar bien los tiempos en esta fase y contemplar que se puedan realizar cambios sobre la marcha que agilicen la entrega del producto final.

1.5.6 Prueba Test

Objetivo:

En esta fase realizaremos pruebas de test para asegurar el correcto funcionamiento de la aplicación, se realizarán pruebas de test sobre las funcionalidades más relevantes.

Acceso a bases de datos.

Formulario de alta de productos.

Consulta de información.

Pruebas de Scaneo.

Se realizarán pruebas por separado para la parte de usuario (administración), usuario (almacén) y usuario (comercial).

Finalidad y riesgos:

Hay que contemplar el tiempo para la corrección de errores encontrados y las siguientes pruebas para confirmar que estos errores se han solucionado.

Documentos:

Resumen de banco de pruebas realizado.

Resumen de errores encontrados y corregidos.

1.5.7 Documentación

Objetivo:

El objetivo de esta fase es desarrollar la memoria del proyecto junto con un manual de usabilidad de la aplicación.

Documentos:

Memoria del proyecto.

Manual de usabilidad.

1.6 Diagrama de Gantt

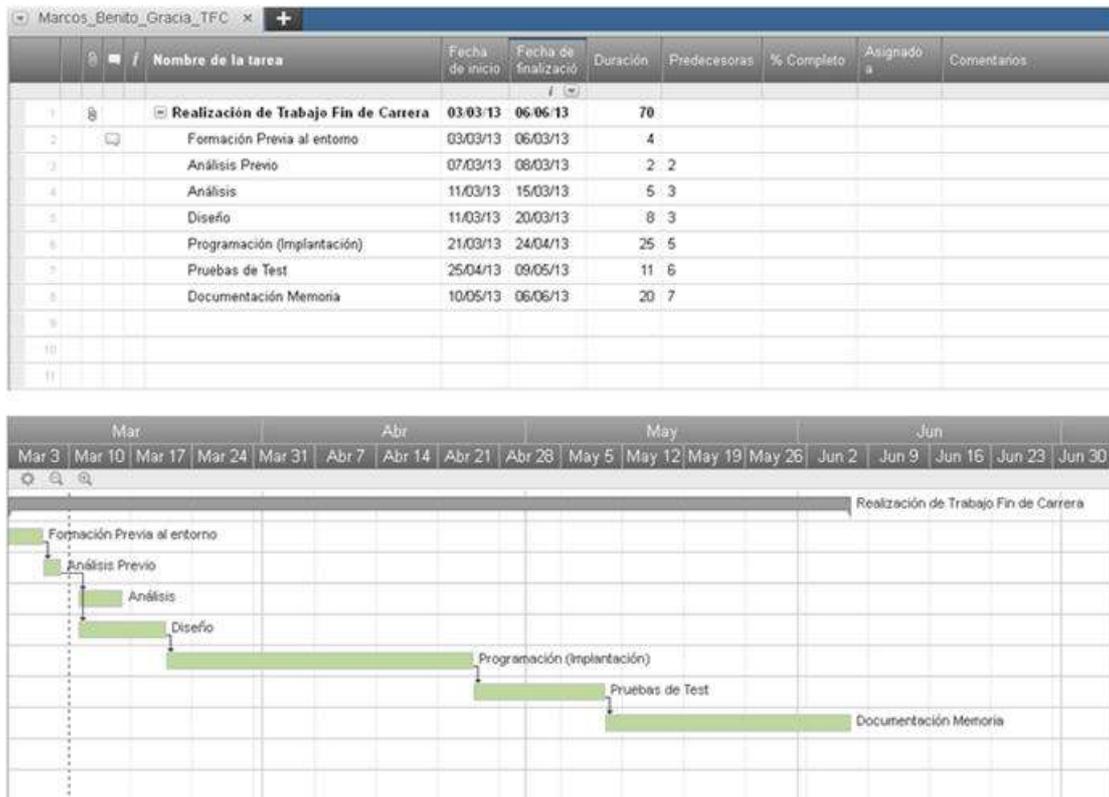


Ilustración 3. Diagrama de Gantt, fases de trabajo

2. Diseño centrado en el usuario

2.1 Introducción

En este apartado se integrará el plan de trabajo en el modelo de diseño centrado en el usuario (DCU) definiendo cuatro fases claves:

- Usuarios y contexto.
- Diseño conceptual.
- Prototipo.
- Evaluación.

Observaciones:

En el primer apartado donde definíamos el guión de trabajo se diseñó un guión con modelo en cascada combinado con diseño centrado en el usuario, después de leer y estudiar el diseño centrado en el usuario se sacan como conclusiones que el modelo en cascada no es el más apropiado para desarrollos sobre dispositivos móviles de ese modo en este apartado se integrará el modelo de trabajo del apartado de guía de trabajo con las fases estudiadas en el diseño centrado en el usuario.

2.2 Objetivo

El objetivo de este apartado es poder tener documentado el proyecto con un diseño basado en la usabilidad y la simplicidad de cara al uso del usuario final en la aplicación, este documento nos servirá para desarrollar la fase de programación de la aplicación.

2.3 Usuarios y Contexto de usos

Objetivo:

El objetivo de esta fase es involucrar a los usuarios finales de la aplicación para conocer el perfil de los usuarios, obtener información sobre sus necesidades, su manera de interactuar con la aplicación y las funcionalidades que esta tendría que tener utilizando los métodos oportunos.

2.3.1 Métodos:

Para realizar esta fase y conocer los perfiles de usuarios y su contexto de uso utilizaremos los siguientes métodos:

- Observación e investigación contextual.
- Entrevistas en profundidad.
- Análisis competitivo.

Se ha decidido hacer entrevistas en profundidad porque se tenía la posibilidad de evaluar personalmente a varios usuarios finales de la aplicación para conocer sus necesidades y ver el tipo de perfiles.

Se obtendrá información de distintos programas de control de stock y de personal relacionado con este trabajo para llevar a cabo este proyecto con la mayor información posible dentro de las posibilidades técnicas y económicas.

Perfiles de usuarios

Para el uso de la aplicación sobre la que trabajamos en este proyecto hay tres perfiles de usuarios o personajes, por un lado el perfil de administración, por otro el de comercial y el de almacén.

Perfil de administración:

Este perfil de usuario es un tipo de personal administrativo encargado de introducir los productos que la empresa encarga para luego comercializar. Introduce los datos relacionados con el producto como pueden ser características, precio, unidades, códigos, etc..

La motivación principal del tipo de perfil es el inventariar los productos para su distribución de cara a un buen funcionamiento logístico de la empresa.

Las necesidades de estos usuarios son:

- Introducción de características de productos.
- Gestión de pedidos y costes de productos.
- Compra de nuevos productos.

Perfiles de almacén:

El perfil de almacén es un usuario que dispone de un dispositivo móvil con tecnología IOS y tarifa de datos o conexión WIFI para poder ir comprobando el stock de su almacén.

Serán usuarios comprendidos entre los 18 - 60 años de edad.

Generalmente el perfil de usuario que utiliza la aplicación tiene una experiencia media en el uso de aplicaciones táctiles sobre dispositivos móviles.

Sus motivaciones son poder consultar el stock de productos en el almacén según está recorriendo el almacén para la comprobación.

Las necesidades de estos clientes son:

- Comprobación del stock en el almacén.
- Recepción y envío de pedidos.

Observaciones mostradas por este tipo de perfil en entrevista personal:

Sería interesante para el usuario la posibilidad de localizar el producto mediante un lector de código de barras que se podría incluir en la aplicación para agilizar el proceso.

Perfiles de comercial:

El perfil de comercial es un usuario que dispone de un dispositivo móvil con tecnología IOS y tarifa de datos para consultar el catálogo de productos y comprobar si hay stock o no.

Serán usuarios comprendidos entre los 18 - 60 años de edad.

Generalmente el perfil de usuario que utiliza la aplicación tiene una experiencia media en el uso de aplicaciones táctiles sobre dispositivos móviles.

Sus motivaciones son poder consultar el catalogo de productos para poder enseñar a los clientes

Las necesidades de estos clientes son:

- Disponibilidad de consulta en cualquier lugar.
- Consultar el stock en los almacenes.
- Ver las características de productos.

Observaciones mostradas por este tipo de perfil en entrevista personal:

Para el usuario sería interesante poder además de consultar los productos, poder cosultarlos sin tener que usar el teclado del dispositivo usando la cámara del dispositivo como lector de códigos de barra.

Contexto de uso

Contexto en perfil de administración:

El escenario donde los usuarios con este perfil normalmente usarán la aplicación será en su lugar de trabajo o negocio con acceso a internet, los horarios generalmente serán en función a su jornada laboral y ocasionalmente fuera del horario de trabajo y en escenarios donde tenga acceso a internet.

Contexto en perfil de almacén:

El escenario de este perfil de usuario será normalmente en su lugar de trabajo (almacén) con acceso a internet, los horarios generalmente serán en función a su jornada laboral y en escenarios donde tenga acceso a internet.

Contexto en perfil de comercial:

El escenario de este perfil de usuario puede ser cualquier lugar con acceso a internet desde donde el usuario pueda realizar la consulta, en la calle, en su puesto de trabajo, en su casa, viajando en transporte públicos, etc.

El horario podrá ser en cualquier momento del día generalmente horario de mañana y tarde durante la jornada laboral.

2.3.2 Análisis de tareas

En este apartado se va a profundizar más en las necesidades y tareas que los usuarios realizarán cuando utilicen la aplicación.

Análisis perfil de administración:

El usuario que utiliza la aplicación como empresa necesitará realizar las siguientes tareas:

Control de productos: Podrá introducir o eliminar los productos que vayan a comercializar también podrá consultar o editar los ya existentes.

Control de precios: Podrá introducir/modificar el precio de los productos.

Realizar orden de compras.

Análisis perfil de almacén:

Realización de registros: El personal de almacén se encargará de recepcionar y etiquetar (código de barras) los productos comprados para luego almacenarlos.

Control de stock: Se llevará al día el control de entrada y salida de productos.

Análisis perfil de comercial:

Gestión de cartera de clientes: El comercial puede llevar el control de su cartera de clientes y posteriormente eliminar o modificar las cuentas de clientes.

Realización de consultas de productos: Podrá realizar consultas de productos para poder ver las características y poder enseñar a los clientes.

2.3.3 Análisis competitivo

Se han analizado las siguientes aplicaciones para apoyarnos en el diseño que utiliza la competencia con servicios similares al que desarrollara nuestra aplicación.

Aplicación	Ideas
Inventario: Aplicación de gestión de inventario.	Ha aportado ideas de usabilidad y diseño.
StockControl: Aplicación con muchas funcionalidades de gestión de productos	Ha aportado ideas de características que se podrían incluir en el control de los productos.
Inventory Universal Edition	Ha aportado ideas para tener en cuenta por la falta de usabilidad, aplicación demasiado compleja de manejar lo que provoca algo de rechazo.

2.4 Diseño Conceptual.

En esta fase veremos el diseño conceptual del proyecto basándonos en los escenarios de uso y en la estructura de la aplicación.

2.4.1 Escenarios de uso

Escenario 1- Perfil de usuario administración:

Raquel tiene 34 años, trabaja por las mañanas en un centro de servicios logísticos, trabaja 8 horas de 8'00 de la mañana a 17'00 de la tarde.

Se encarga de Contratación De Transporte, Coordinar Cargas Y Descargas, Preparación De Documentación Correspondiente, Negociación Precios De Productos.

Interacción:

Raquel podrá realizar la catalogación de productos desde su dispositivo móvil o para mayor comodidad desde su aplicativo web desde su ordenador de sobremesa de la oficina mediante un navegador web se conectará al aplicativo para poder catalogar y se guardará en la BD.

Escenario 2- Perfil de usuario almacén:

José Antonio es el responsable del almacén situado en Barcelona en el barrio de Sant Andreu, tiene 52 años, entra en el almacén a las 8 de la mañana y realiza un descanso de 2 horas para comer normalmente entre las 2 y las 4 de la tarde, su jornada laboral no termina hasta las 9 o 10.

Relación con la tecnología:

Suele utilizar un pequeño ordenador en la oficina del taller para realizar gestiones y conectarse a internet. Utiliza su Smartphone para realizar llamadas y utilizar alguna aplicación de utilidad como agendas o blocs de notas.

Objetivo:

José Antonio quiere poder estar distribuyendo los productos sin tener que estar dentro de la oficina del almacén para consultar el stock determinado de otros productos además de consultarlo por código de barras in situ sin tener que depender de cables ni volcados posteriores al almacén de datos.

Interacción:

Es una persona que está acostumbrada al uso de aplicaciones de gestión, está acostumbrada a moverse por menús , enviar mails , introducir datos etc.

Escenario 3- Perfil de usuario almacén:

Persona:

Javi tiene 32 años, es uno de los comerciales de la empresa, suele ocupar las mañanas y parte de la tarde para visitar cliente y enseñar los productos tiene la empresa.

Escenario:

Vive en Madrid, en el barrio de Hortaleza, y principalmente maneja la cartera de clientes de la zona norte y este de Madrid.

Relación con la tecnología:

Siempre está en contacto con la tecnología ya sea a través de su ordenador personal su portátil o su Smartphone, lo utiliza principalmente para comunicarse con los clientes y consultar información.

Tiene acceso a internet en cualquier franja horaria y situación.

Objetivo:

Javi es una persona bastante activa y que no está mucho tiempo en su puesto de trabajo ya que las visitas le llevan gran parte de su jornada laboral. El quiere poder disponer de la mayor información de los productos para agilizar trámites en llamadas y evitar cargar con catálogos.

Interacción:

Es una persona que está acostumbrada al uso de cualquier tipo de aplicaciones, aunque le gusta que las aplicaciones tengan un entorno atractivo y fácil de utilizar.

2.4.2 Flujos de interacción

Perfil de usuario administración

Diagrama de flujo insertar producto.

Este diagrama nos muestra la estructura en el flujo de desarrollo cuando un usuario de tipo administración quiere realizar la inserción de un producto.

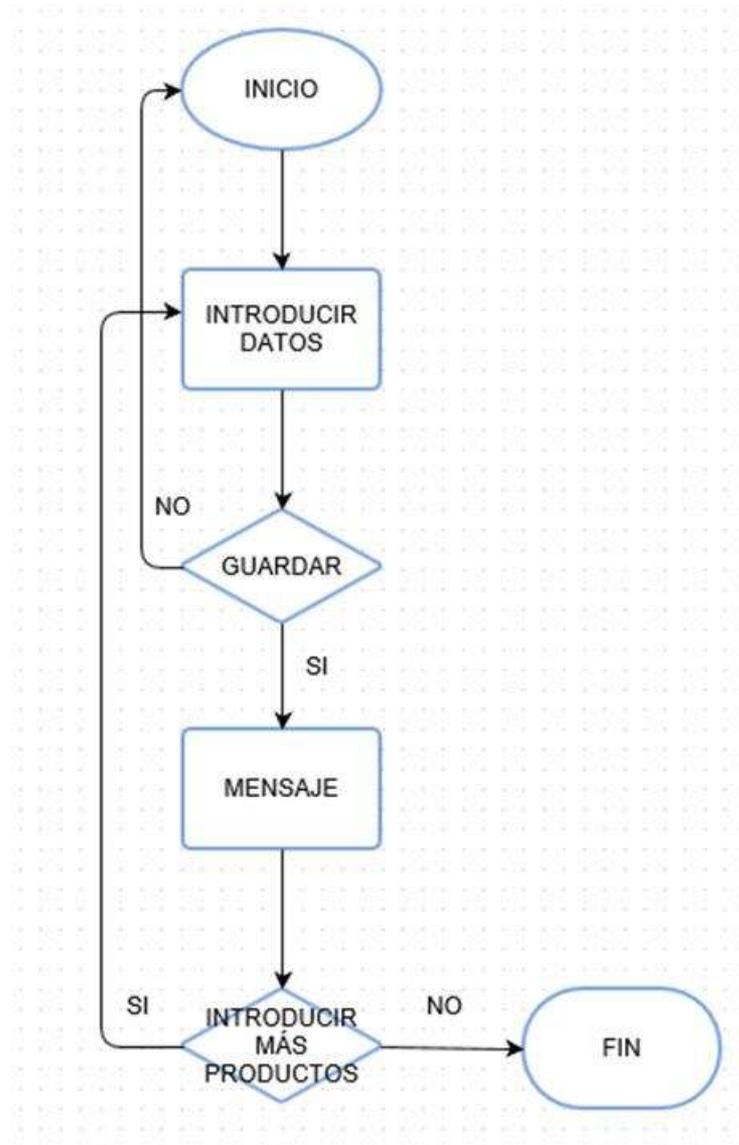


Ilustración 4. Insertar producto

Perfil de usuario administración /almacén/comercial

Diagrama de flujo para realizar consulta/modificación.

Este diagrama nos muestra la estructura en el flujo de desarrollo cuando un usuario de tipo administración/almacén/comercial que quiere realizar una consulta/modificación de un producto. Este flujo nos vale para la parte de consulta de cualquier tipo de usuario.

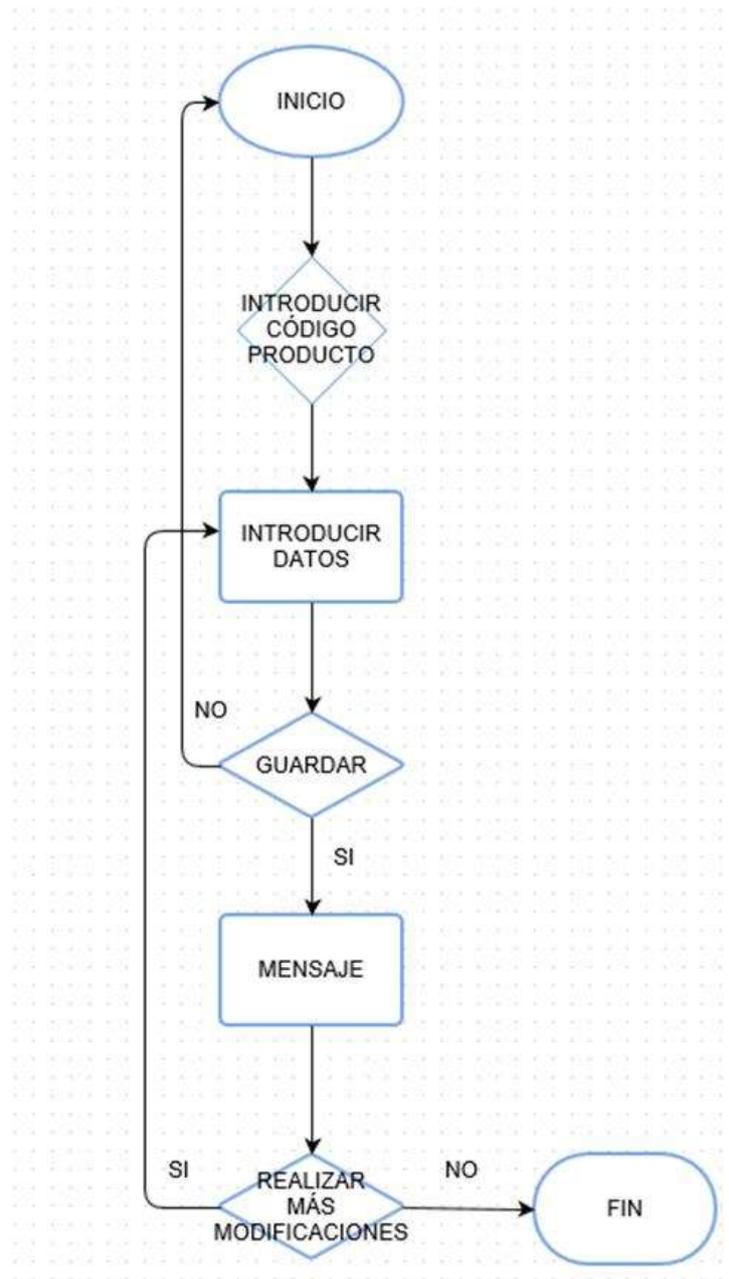


Ilustración 5. Consulta/modificación de un producto.

2.5 Prototipo

2.5.1 Prototipo de pantallas

En este apartado se van a mostrar los prototipos dibujados para la interface gráfica, tendremos tres partes diferenciadas, unas pantallas para cuando la aplicación la utilice el usuario de perfil administración, otra para el perfil almacén y otras para cuando las utilice el perfil de comercial.

2.5.2 Primeras ideas de diseño

Estas imágenes muestran las primeras ideas de la estructura y los diseños de las pantallas de la aplicación y como se interactúa a través de ellas según el tipo de usuario.

Pantallas perfil administración

En este apartado se mostrarán los prototipos de las pantallas que se utilizarán en la aplicación cuando accedamos como administración.

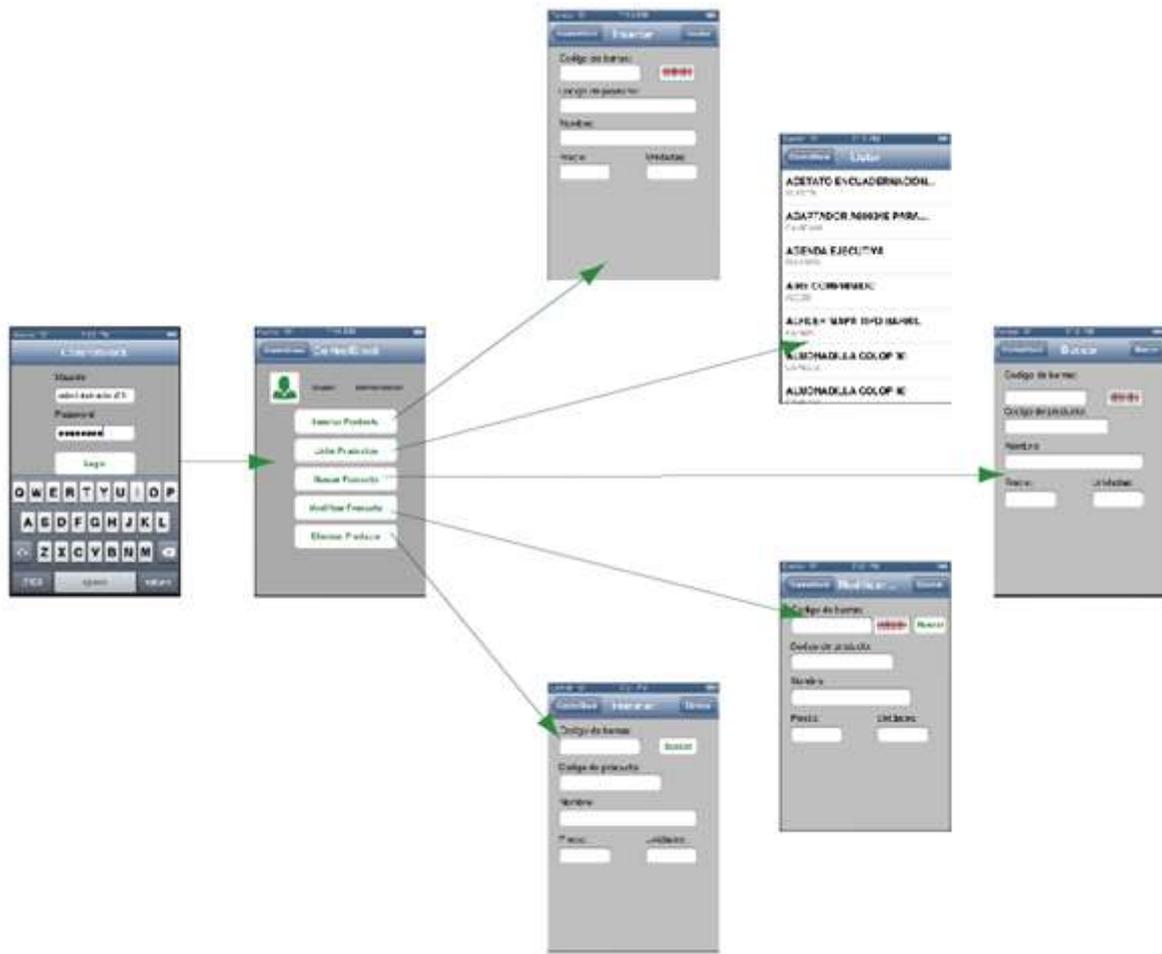


Ilustración 6. Organigrama perfil administración

Pantallas perfil almacén

En este apartado se mostrarán los prototipos de las pantallas que se utilizarán en la aplicación cuando accedamos como usuario almacén.

Organigrama de navegación:

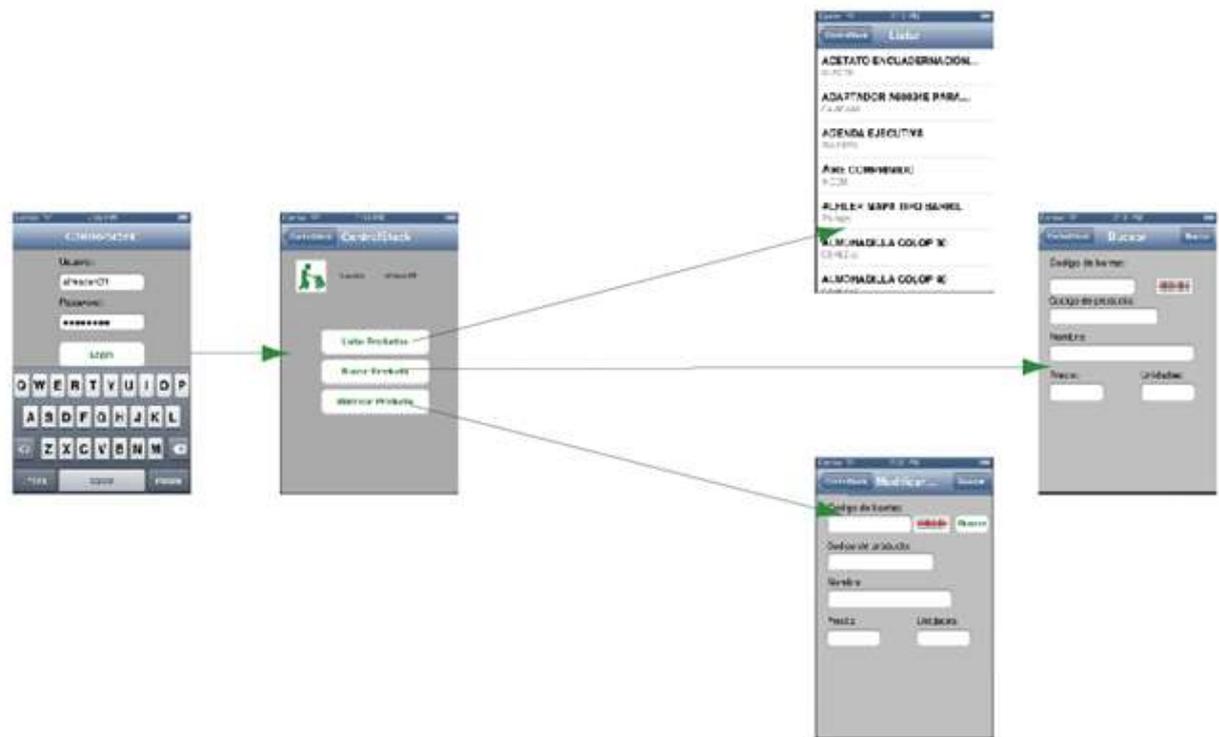


Ilustración 7. Organigrama perfil almacén

Pantallas perfil comercial

En este apartado se mostrarán los prototipos de las pantallas que se utilizarán en la aplicación cuando accedamos como usuario comercial.

Organigrama de navegación:

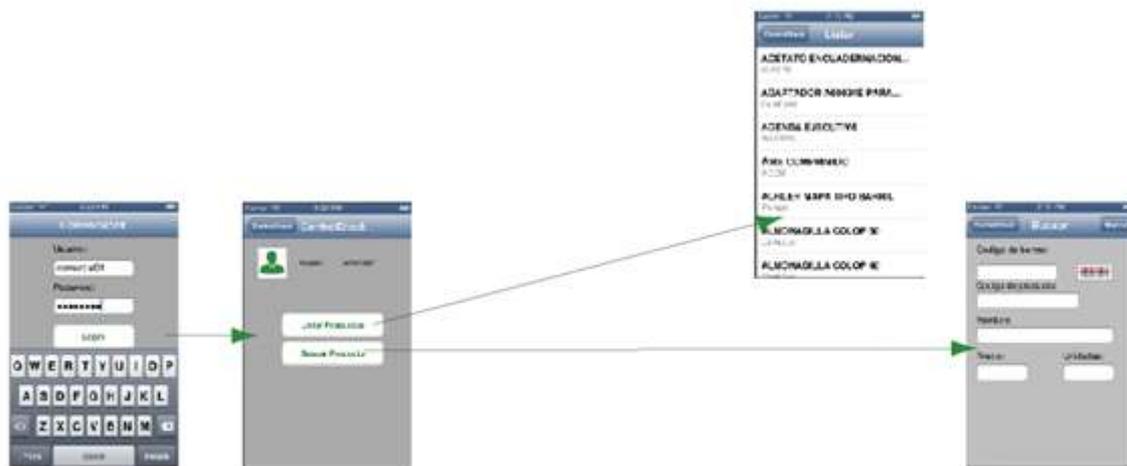


Ilustración 8. Organigrama perfil comercial

2.5.3 Influencias sobre el diseño.

En este apartado vamos a explicar cómo han afectado las entrevistas y test funcionales a los usuarios para después realizar el diseño principal.

Diseño general.

La idea inicial era una pantalla con un menú de botones para navegar por las diferentes opciones abriendo varias ventanas pero una de las conclusiones que se sacan es que para este tipo de aplicación los usuarios prefieren pantallas sencillas con pocos componentes que no se muestren muy cargadas, intuitivas y que cuando se realicen acciones sea con pocos pasos sin multitud de ventanas.

Para poder cumplir con estos objetivos se decide crear una pantalla principal donde a simple vista se pueda navegar por las opciones principales tanto en el modo de administración o de almacén o de comercial creando una barra de navegación con dos botones de atrás y guardar en casi todas las pantallas para poder avanzar o retroceder en el programa y así navegar por otras pantallas manteniendo siempre la misma estructura sencilla e intuitiva, además se consigue no cargar la pantalla ya que en cada acción de navegación se mostrará solo la información que queremos ver.

Cámara.



Se pretende utilizar la cámara  para poder usar como escáner de código de barras.

3. Arquitectura e implementación

3.1 Introducción

En este apartado se hace una pequeña introducción a la plataforma iOS y veremos la arquitectura de la aplicación especificando los diferentes módulos que la forman.

3.2 Objetivo

El objetivo de este apartado es poder tener la arquitectura del proyecto especificada para poder implementar el código de la aplicación y tener un punto de referencia para poder retomar en un futuro el proyecto con facilidad.

3.3 Plataforma iOS

La aplicación esta realizada para trabajar sobre iOS para dispositivos Iphone/Ipad.

La interfaz de usuario de iOS está basada en el concepto de manipulación directa, usando gestos multitáctiles. Los elementos de control consisten de deslizadores, interruptores y botones. La respuesta a las órdenes del usuario es inmediata y provee de una interfaz fluida. La interacción con el sistema operativo incluye gestos como deslices, toques, pellizcos, los cuales tienen definiciones diferentes dependiendo del contexto de la interfaz. Se utilizan acelerómetros internos para hacer que algunas aplicaciones respondan a sacudir el dispositivo (por ejemplo, para el comando deshacer) o rotarlo en tres dimensiones (un resultado común es cambiar de modo vertical al apaisado u horizontal).

iOS es el sistema implementado por Apple que permite **ejecutar aplicaciones nativas en los dispositivos móviles: iPhone, iPod Touch y iPad**. La arquitectura de esta plataforma toma como base el núcleo del sistema Mac OS X e incorpora una nueva capa que da **soporte a la interfaz multi-touch**.

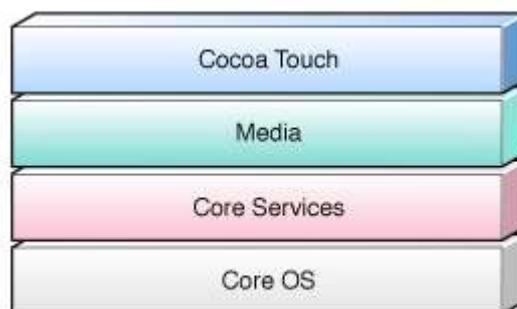


Ilustración 9. Organigrama Plataforma iOS

3.4 Requisitos de desarrollo

Para desarrollar la aplicación se han utilizado los siguientes recursos:

Herramientas de desarrollo:

XCode 4.6

Kits de desarrollo:

IOS 6 SDK

OSX 10.8 SDK

Sistema Operativo:

OSX Mountain Lion o OSX Lion

Para el entorno servidor:

Herramientas de desarrollo:

Dreamweaver CS4

Kits de desarrollo:

PHP Version 5.3

MySQL Version 5.1.58

3.5 Contenido

La aplicación está formada por los siguientes módulos distribuidos en los siguientes directorios con el código fuente de la aplicación y los archivos que la complementan.

Directorio JSON: Librerías de JSON, acrónimo de *JavaScript Object Notation*, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

Directorio ZBAR: Librerías de SouceForge capaces de leer principalmente los de código de barras tipo Code 128 que es un código de barras de alta densidad, usado ampliamente para la logística y paquetería. Puede codificar caracteres alfanuméricos o solo numéricos. Con este código es posible representar todos los caracteres de la tabla ASCII, incluyendo los caracteres de control.

Para comprender cómo se codifica este código, debemos tener en cuenta que cada ASCII se codifica mediante 11 barras.

Directorio ControlStock: Contiene los fichero fuentes del programa entregado junto con las imágenes usadas en la aplicación.

Directorio Frameworks: Contiene los frameworks usados en nuestra app.

Directorio Product: donde se genera Xcode el fichero app de la aplicación.

En cuanto a la parte servidor se han realizado distintas paginas en php que se encargan de comunicar la aplicación con la base de datos Mysql.

Los ficheros son los siguientes:

Insert.php

```
<?php
/*
Marcos Benito
TFC UOC
*/

$DB_HostName = "mysql.nixiweb.com";
$DB_Name = "u960143822_iphone";
$DB_User = "u960143822_ios";
$DB_Pass = "12345678";
$DB_Table = "user";

        $codigo = $_GET["codigo"];
        $name = $_GET["name"];
        $codbarras = $_GET["codbarras"];
        $precio = $_GET["precio"];
        $unidad = $_GET["unidad"];

//}

/*
    if (isset($_GET["codigo"])){
        $codigo = $_GET["codigo"];
    }
    else{
        $name = "error";
    }
*/

$con = mysql_connect($DB_HostName,$DB_User,$DB_Pass) or die(mysql_error());
mysql_select_db($DB_Name,$con) or die(mysql_error());

    $sql = "insert into $DB_Table (codigo, name, codbarras, precio, unidad) values
('$codigo','$name','$codbarras','$precio','$unidad)";

//($_POST[firstname],$_POST[lastname],$_POST[age]);

$res = mysql_query($sql,$con) or die(mysql_error());

mysql_close($con);

if ($res) {
    echo "correctamente";
}else{
    echo "error";
}

?>
```

Buscar.php

```
<?php
/*
Marcos Benito
TFC UOC
*/

$DB_HostName = "mysql.nixiweb.com";

$DB_Name = "u960143822_iphone";

$DB_User = "u960143822_ios";

$DB_Pass = "12345678";

$DB_Table = "user";

    $codigo = $_GET["codigo"];

    $name = $_GET["name"];

    $codbarras = $_GET["codbarras"];

    $precio = $_GET["precio"];

    $unidad = $_GET["unidad"];

//}

/*

if (isset($_GET["codigo"])){

    $codigo = $_GET["codigo"];

}

else{

    $name = "error";

}

*/

$con = mysql_connect($DB_HostName,$DB_User,$DB_Pass) or die(mysql_error());

mysql_select_db($DB_Name,$con) or die(mysql_error());

//Creamos un array para meter los resultados

$arr = array();

//Lanzamos la consulta

//$query = mysql_query("SELECT * FROM user where codigo like '%$codigo%'");

$rs = mysql_query("SELECT id, codigo, name, codbarras, precio, unidad FROM user where (codigo = '$codigo' OR codbarras = '$codbarras' OR name = '$name' OR precio = '$precio' OR unidad = '$unidad')");
```

```
//Agregamos las filas al array
while($obj = mysql_fetch_object($rs)) {
    $sarr[] = $obj;
}

//devolvemos el resultado
if (empty($sarr))
{
    echo "error";
}
else
{
    echo '{"user":'.json_encode($sarr).'}';
}

$res = mysql_query($sql,$con) or die(mysql_error());

mysql_close($con);

if ($res) {
    echo "correctamente";
}
else{
    echo "error";
}
// end else

?>
```

Actualizar.php

```
<?php
/*
Marcos Benito
TFC UOC
*/

$DB_HostName = "mysql.nixiweb.com";

$DB_Name = "u960143822_iphone";

$DB_User = "u960143822_ios";

$DB_Pass = "12345678";

$DB_Table = "user";

$id = $_GET["id"];

$codigo = $_GET["codigo"];

$name = $_GET["name"];

$codbarras = $_GET["codbarras"];

$precio = $_GET["precio"];

$unidad = $_GET["unidad"];

//}

/*

if (isset($_GET["codigo"])){

    $codigo = $_GET["codigo"];

}

else{

    $name = "error";

}

*/

$con = mysql_connect($DB_HostName,$DB_User,$DB_Pass) or die(mysql_error());

mysql_select_db($DB_Name,$con) or die(mysql_error());

//Creamos un array para meter los resultados

$arr = array();

//Lanzamos la consulta

$query = mysql_query("SELECT * FROM user where codigo like '%$codigo%'");

$rs = mysql_query("UPDATE $DB_Table SET codigo='$codigo', name='$name', codbarras='$codbarras',precio='$precio', unidad='$unidad'
WHERE id='$id'");
```

```
/*("UPDATE usuarios
SET usuarioNombre = '$usuarioNombre',
usuarioEmail = '$usuarioEmail',
usuarioDNI = '$usuarioDNI'
WHERE usuario_ID = '$usuario_ID'*/)

$res = mysql_query($sql,$con) or die(mysql_error());

mysql_close($con);

if ($res) {
    echo "correctamente";
}
else{
    echo "error";
}

?>
```

Borrar.php

```
<?php
/*
Marcos Benito
TFC UOC
*/

$DB_HostName = "mysql.nixiweb.com";
$DB_Name = "u960143822_iphone";
$DB_User = "u960143822_ios";
$DB_Pass = "12345678";
$DB_Table = "user";

$id = $_GET["id"];

$con = mysql_connect($DB_HostName,$DB_User,$DB_Pass) or die(mysql_error());
mysql_select_db($DB_Name,$con) or die(mysql_error());

$sql = "DELETE FROM `user` WHERE `id` like '%$id%'";
$res = mysql_query($sql,$con) or die(mysql_error());

$result = mysql_query($sql);

if(!$result)
{
mysql_close();
echo mysql_error();
return;
}
else{
echo "correctamentamente";
}

mysql_close();

?>
```

Login.php

```
<?php
/*
Marcos Benito
TFC UOC
*/

$DB_HostName = "mysql.nixiweb.com";

$DB_Name = "u960143822_iphone";

$DB_User = "u960143822_ios";

$DB_Pass = "12345678";

$DB_Table = "login";

$susuario = $_GET["usuario"];

$password = $_GET["password"];

//}

/*
if (isset($_GET["codigo"])){
    $codigo = $_GET["codigo"];
}
else{
    $name = "error";
}
*/

$con = mysql_connect($DB_HostName,$DB_User,$DB_Pass) or die(mysql_error());
mysql_select_db($DB_Name,$con) or die(mysql_error());

//Creamos un array para meter los resultados

$arr = array();

//Lanzamos la consulta

$query = mysql_query("SELECT * FROM user where codigo like '%$codigo%'");

$rs = mysql_query("SELECT id, usuario, password FROM login where usuario like '%$usuario%' and password like '%$password%'");

//Agregamos las filas al array
while($obj = mysql_fetch_object($rs)) {
```

```
$arr[] = $obj;
}

//devolvemos el resultado
if (empty($arr))
{
echo "error";
}
else
{
echo '{"login":'.json_encode($arr).'}';
}

$res = mysql_query($sql,$con) or die(mysql_error());

mysql_close($con);
if ($res) {
    echo "correctamente";
}
else{
    echo "error";
}
// end else

?>
```

CosultaMsql.php

```
<?php

//credenciales de la BBDD

$host='mysql.nixiweb.com';

$db='u960143822_iphone';

$id='u960143822_ios';

$pwd='12345678';

//conectar al servidor de la BBDD

$link = Mysql_connect($host, $id, $pwd) or die("No se puede conectar");

//Seleccionamos la BBDD

mysql_select_db($db) or die("No se puede seleccionar la bbdd");

//Creamos un array para meter los resultados

$arr = array();

//Lanzamos la consulta

$rs = mysql_query("SELECT codigo, name, codbarras, precio, unidad FROM user ORDER BY name");

//Agregamos las filas al array

while($obj = mysql_fetch_object($rs)) {

    $arr[] = $obj;

}

//devolvemos el resultado

echo '{"user":'.json_encode($arr).'}';

?>
```

Arquitectura de aplicación

En la arquitectura de la aplicación tenemos una primera capa de interface gráfica que es la capa con la que interactúa el usuario en esta capa se encontrarán todas las pantallas de la aplicación.

La siguiente capa está conectada directamente con la interface gráfica pues ya hemos comentado antes que la mayoría de las actividades están relacionadas con una pantalla, en esta capa es donde se gestionan los eventos producidos en la capa de interface gráfica del usuario y se comunican con la siguiente capa de componentes para realizar las acciones correspondientes, pues en esta capa también tendremos actividades que no dispongan de pantalla gráfica.

En la siguiente capa de componentes tenemos los diferentes módulos con los que trabaja la capa de actividades y a su vez estos requieren las capas inferiores de librerías y persistencia para poder ejecutar los métodos que tiene cada módulo y que se lanzan desde la capa de actividades.

La capa de API y persistencia contienen los métodos necesarios por la capa de módulos para poder utilizar las librerías de más bajo nivel que nos facilitan la utilización de las diferentes tecnologías utilizadas en la aplicación.

Y finalmente tenemos la capa servicios que incluye todos los servicios que nos proporciona el sistema operativo Android para poder utilizar los gestores que interactúan con el sistema operativo y con el hardware.

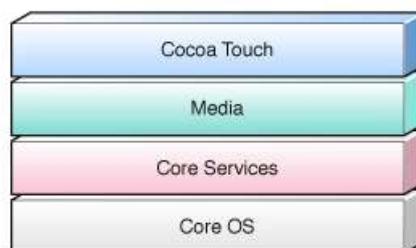


Ilustración 10. Arquitectura de capas iOS

Cocoa Touch

Cocoa Touch es la capa más importante para el desarrollo de aplicaciones iOS. Posee un conjunto de Frameworks que proporciona el API de Cocoa para desarrollar aplicaciones. Se podría decir que Cocoa Touch proviene de Cocoa, la API ya existente en la plataforma MAC.

Esta capa está formada por dos Frameworks fundamentales:

UIKit: contiene todas las clases que se necesitan para el desarrollo de una interfaz de usuario

Foundation Framework: define las clases básicas, acceso y manejo de objetos, servicios del sistema operativo

Media

Provee los servicios de gráficos y multimedia a la capa superior.

Core Services

Contiene los servicios fundamentales del sistema que usan todas las aplicaciones

Core OS

Contiene las características de bajo nivel: ficheros del sistema, manejo de memoria, seguridad, drivers del dispositivo.

3.6 Módulos de aplicación

En este apartado se comentaran los diferentes módulos que forman la aplicación. La aplicación está repartida en los siguientes módulos, *(aunque está pendiente de terminar el modulo de Eliminar y de Listar con posibilidad de modificar)*.

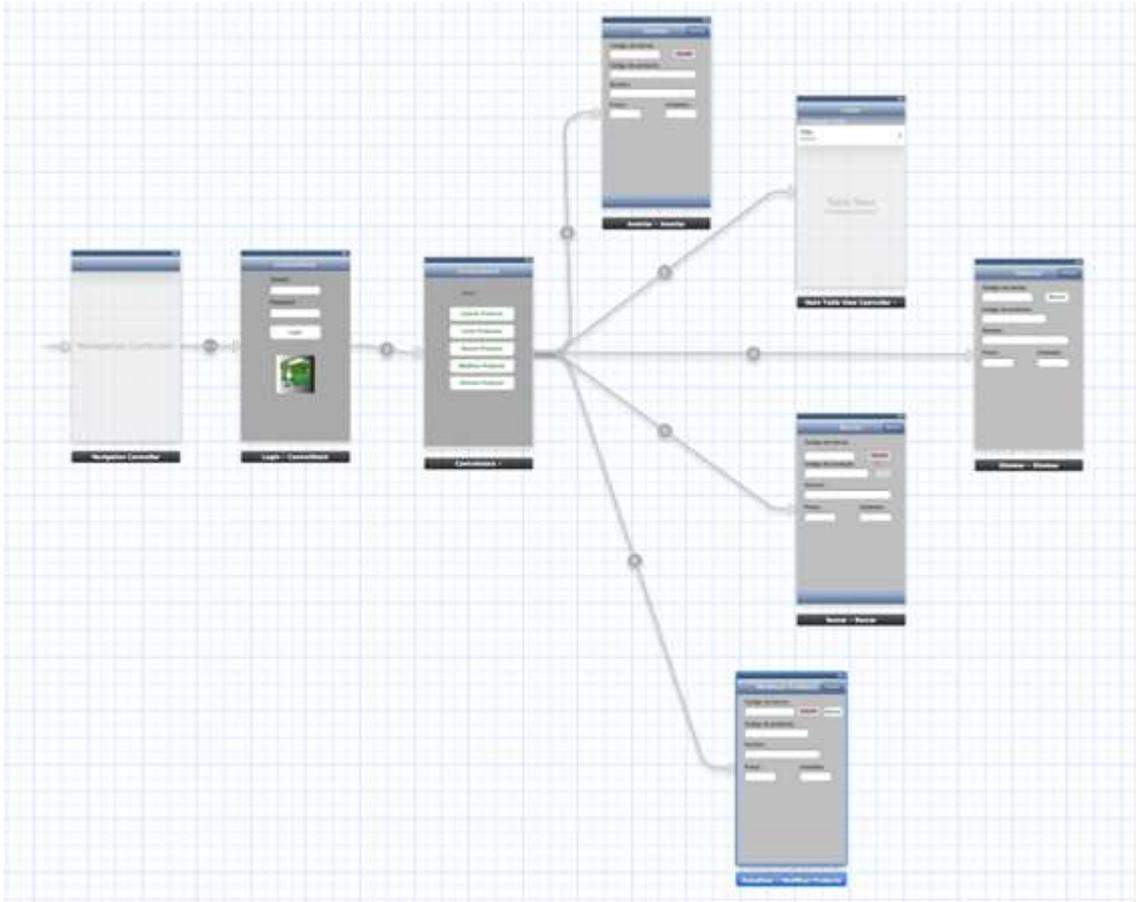


Ilustración 11. Diagrama de módulos

3.6.1 Navigation Controller

En este módulo se crea la estructura del navegador dentro de la aplicación. El Navigation Controller será la plantilla de navegación.

3.6.2 View Controller- Login

Es el módulo de acceso a la aplicación. Su función es la de controlar el acceso y gestionar diferenciar los distintos perfiles de usuario para que en la siguiente pantalla muestre las determinadas funciones dependiendo del perfil (administración, almacén o comercial).

3.6.3 Table View Controller– ControlStock

En este módulo como menú de posibilidades de la aplicación. Aparecerá las distintas actividades a realizar, posteriormente bajo el menú de acciones aparecerá en la versión final el logotipo de la aplicación junto con la información de la versión.

3.6.4 View Controller – Insertar Producto

Aparece en el menú de “*Insertar Producto*” el formulario para dar de alta un producto. En el formulario nos aparecerán los distintos campos (código de barras, código de producto, nombre, precio y unidades), que posteriormente serán almacenados en una base de datos Mysql alojada en un host de internet ([nixiweb](#)).

Además se incluye el botón Scan, que nos permitirá incluir en el campo código de barras el número obtenido escaneándolo con la cámara del dispositivo.

3.6.5 Main Table View Controller – Listar Productos

Son las funciones de “*Listar Productos*”. Realiza una consulta a la base de datos de internet y recoge los distintos productos introducidos. Mostrando en celdas por código de producto y nombre, pulsando sobre cada uno de los distintos productos nos mostrara un mensaje con los datos adicionales de precio, código de barras y unidades disponibles.

3.6.6 View Controller – Eliminar Producto

Nuestra el formulario para poder buscar un producto que deseamos eliminar. Para ello debemos primero introducir código de barras o el código o el nombre del producto, pulsar buscar y posteriormente pulsar eliminar para eliminarlo de la BBDD.

3.6.7 View Controller – Buscar Producto

Nos permitirá buscar un producto por alguno de los valores del formulario o escanear su código de barras para agilizar su búsqueda.

3.6.8 View Controller – Modificar Producto

En este caso este view controlles realiza las tareas anteriores de búsqueda para su posterior modificación. Esta acción es quizás la más utilizada por el personal del almacén para la modificación del stock. También nos permite buscar por código de barras mediante el escáner.

3.7 Persistencia

En este apartado se va a explicar la persistencia de los datos con los que trabaja la aplicación tanto localmente como remotamente.

3.7.1 Persistencia Local

En este caso no existe una persistencia local ya que no se almacenan datos en local. Lo único que se tiene en local son los propios ficheros de la app y sus imágenes.

3.7.2 Persistencia Remota

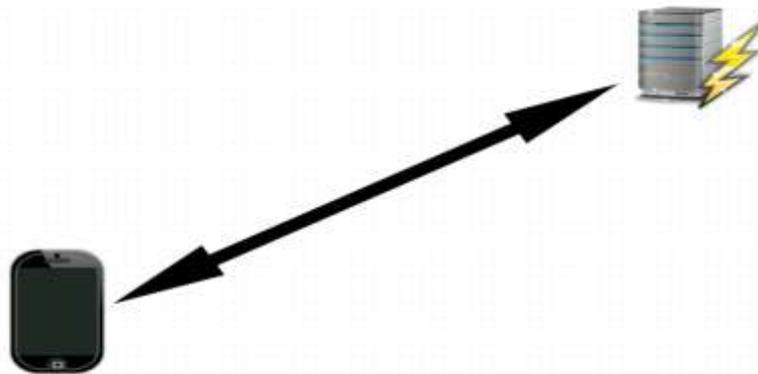


Ilustración 12. Esquema de persistencia remota

Para el sistema de persistencia remota se ha recurrido a un servicio de hosting gratuito por internet proporcionado por el proveedor Nixiweb, que nos da acceso gratuito a hosting web Apache Version: Apache/2.2.17, permite programación PHP Version:5.3 y acceso a base de datos MySQL Version:5.1.58.

Columna	Tipo	Nulo	Predeterminado	Comentarios
id	int(11)	No		
codigo	varchar(8)	No		
name	varchar(255)	No		
codbarras	varchar(25)	No		
precio	float	No		
unidad	int(11)	No		

Ilustración 13. Estructura de la tabla user de productos

Columna	Tipo	Nulo	Predeterminado	Comentarios
id	int(11)	No		
usuario	varchar(25)	No		
password	varchar(25)	No		

Ilustración 14. Estructura de la tabla login de usuarios

3.8 Pruebas y Test de aplicación

En este apartado se explican las diferentes pruebas y test llevados a cabo con el fin de verificar el correcto funcionamiento de la aplicación.

Muchos procesos de la aplicación requieren de la ejecución de las respuestas recibidas por el servidor así como la interacción del usuario eso dificulta hacer pruebas de módulos unitarias por eso se han realizado pruebas unitarias por módulos de funciones muy específicas y pruebas de usabilidad para cada proceso.

3.8.1 Pruebas de usabilidad manuales

Se han realizado pruebas manuales sobre la aplicación para confirmar el correcto funcionamiento *(como la navegación entre todas la posible pantallas y que el flujo de información sea el correcto para el uso de cualquier usuario/perfil)*.

3.8.2 Pruebas de test unitarias

Se realizan pruebas unitarias sobre funciones concretas que permiten este tipo de test a través de las funciones dentro de las clases, y se comprueba actualmente el estado de dichas funciones lo que conlleva un gran trabajo de chequeo de posibles errores a la hora de introducir datos, convertir datos o leer datos. Se confirma la funcionalidad de todos los botones para su correcto funcionamiento y se elimina las funcionalidades que se querían dejar pero no se han conseguido como la barra de búsqueda en la tabla de listar productos en la cual no he conseguido aplicar debido a incompatibilidades en la búsqueda dentro de un objeto diccionario y he tenido que descartar para que permitiera funcionar todo lo actual. Se intentó también aplicar un objeto Stepper para la modificación de unidades pero tuve también problemas con la conversión de datos de tipo string a enteros para que lo pudiera mantener desde el valor actualmente recibido de la bbdd.

4. Resultados

4.1 Experiencias

Durante el desarrollo del proyecto he adquirido una serie de aptitudes y experiencias como el aprendizaje de nuevas tecnologías, la utilización y comprensión de APIs de terceros, la comprensión de documentación externa, la improvisación y resolución de problemas sobre la metodología de programación, la correcta utilización de nomenclaturas e indentación en el código y documentación, profundizar más sobre aspectos de programación como la optimización y la seguridad, y trabajar con un enfoque totalmente pensado en la fluidez y la simplicidad de la aplicación sin perder funcionalidad de cara a que el usuario tenga una mejor experiencia de uso de la aplicación.

Y he podido conocer la arquitectura de una aplicación IOS mediante el entorno de Xcode del cual no tenía ningún conocimiento y ha sido una tarea bastante difícil debido al elevado número de horas de dedicación al conocimiento de esta tecnología junto a la del desarrollo de la parte servidor en PHP y MySQL que tampoco conocía.

Estoy adquiriendo conocimientos de parseo de datos con JSON y el funcionamiento del framework de ZBar para la lectura de códigos de barra.

La experiencia ha sido positiva aunque hay momentos de desesperación ante la falta de conocimiento y la elevada carga de trabajo a la hora de investigar la resolución de dichos problemas y la limitación de tiempo que tenemos con las fechas de entrega.

La pena es no poder tener más tiempo y de disponer de mejor material para consulta para poder abarcar el proyecto que en su inicio quería y no he podido llegar aunque reconozco el avance realizado en este entorno. También echo en falta la experiencia tanto programando como diseñando ya que me hubiera gustado saber hacer algunos efectos estéticos para una presentación más profesional.

4.2 Críticas y mejoras

La aplicación tiene varios puntos en los que se puede mejorar, en este apartado intentaré hacer un crítica sobre estos puntos y explicar cómo podríamos mejorarlos.

Se podría añadir una opción para los perfiles de administración que son los que más derechos tienen dentro de la aplicación para poder generar usuarios y asignarles los perfiles determinados.

Otro punto interesante sería poder realizar búsquedas de productos mediante la searchbar que no he conseguido hacer funcionar con el diccionario de datos devuelto en la consulta php. He conseguido realizarlas para un array estático pero no para el diccionario obtenido.

Otra de las cosas que se podían mejorar era el diseño, ya que aunque he intentado hacerlo lo mejor posible el resultado está lejos de lo que son las aplicaciones comerciales. Me hubiera gustado saber algo más de diseño gráfico para poder conseguir una apariencia más elegante.

4.3 Conclusiones

La estrategia establecida para el desarrollo del proyecto ha tenido un resultado positivo ya que se ha finalizado el proyecto y se ha logrado cumplir con casi todo el objetivo propuesto, realizando una aplicación nativa para el sistema operativo IOs cumpliendo con casi la totalidad de las funcionalidades especificadas.

Una parte importante de esta estrategia fue la formación previa para entender cómo funciona y como se trabaja sobre IOs.

Se han adquirido los conocimientos y experiencias que se esperan al finalizar este tipo de proyectos.

En resumen ha sido una tarea bonita de realizar a nivel personal aunque se ha pasado por fases muy difíciles en las que no sabía por dónde ir debido a la deficiencia de conocimientos que se ha compensado con la paciencia y la voluntad con las horas dedicadas.

5. Bibliografía

Desarrollo de Aplicaciones para Iphone & Ipad por Joe Conway y Aaron Hillegass (ANAYA).

Foro de desarrollo para xcode: www.stackoverflow.com

Mac OS X Developer Library (2011). *The Objective-C Programming Language*. [en línea]. http://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/ObjectiveC/Introduction/introObjectiveC.html://apple_ref/doc/uid/TP30001163 [data de consulta: 09/03/2012].

iOS Developer Library (2011). *iOS Technology Overview*. [en línea]. http://developer.apple.com/library/ios/documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/iPhoneOSOverview/iPhoneOSOverview.html://apple_ref/doc/uid/TP40007898-CH4-SW1 [data de consulta: 09/03/2012]

Wikipedia (2012). *Model-View-Controller pattern*. [en línea]. <http://en.wikipedia.org/wiki/Model-view-controller> [data de consulta: 01/04/2012]

Mac OS X Developer Library (2011). *Cocoa core competencies*. [en línea]. <https://developer.apple.com/library/mac/documentation/General/Conceptual/DevPedia-CocoaCore/MVC.html> [data de consulta: 01/04/2012]

iOS Developer Library (2011). *iOS App Programming Guide*. [en línea]. <http://developer.apple.com/library/ios/DOCUMENTATION/iPhone/Conceptual/iPhoneOSProgrammingGuide/Introduction/Introduction.html> [data de consulta: 03/05/2012]

<http://barcode.tec-it.com/barcode-generator.aspx?LANG=es> , web que proporciona la generación de códigos de barra de distintos tipos. En este caso el code 128.