

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

Iago González Feroso
ETIG

Jordi Ferrer Duran

14/01/2013

Resumen

El presente trabajo fin de carrera, en adelante TFC, tiene por objeto el diseño e implementación de una base de datos que cubra la necesidad de gestionar y almacenar la información generada por la actividad de una plataforma de descarga de aplicaciones para móviles inteligentes, dentro del marco de colaboración de la UOC con la “Asociación mundial de desarrolladores de aplicaciones móviles”.

La base de datos deberá cubrir tanto la gestión de Altas, Bajas y modificaciones de los usuarios, desarrolladores y aplicaciones de la BD, como permitir almacenar toda la información generada en el proceso de descarga de aplicaciones por parte de los usuarios.

Para el desarrollo del TFC realizaremos un análisis previo del problema, una planificación temporal y a partir de ahí un análisis y diseño de la BD con sus diagramas correspondientes UML y lógico relacional. Diseñaremos los Scripts de creación de la BD para las tablas e índices correspondientes y los procedimientos almacenados y funciones para dar respuesta a los requerimientos del cliente.

A medida que vayamos avanzando iremos documentando todos los pasos en una memoria. Como último paso se entregará la memoria finalizada, conjuntamente con una presentación, con objeto de documentar el TFC.

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

A Jaione, por su comprensión y paciencia; A Marcos por su ayuda en estos años de UOC y a todos aquellos que me han apoyado, en especial a Carmen.

Índice

1. Introducción.....	5
1.1 Descripción general del proyecto.....	6
1.2 Planificación detallada del proyecto.....	7
1.3. Productos entregados en las PAC y metodología de desarrollo asociada.	10
2. Análisis de requisitos.....	12
2.1. Descripción inicial.....	12
2.2. Requisitos Funcionales.....	13
2.3. Requisitos no funcionales.....	14
3. Diseño del sistema.....	15
3.1. Diseño de la base de datos.....	15
3.2. Etapa de Diseño conceptual.....	15
3.2.1. Modelo UML.....	15
3.2.2. Identificación de Entidades.....	16
3.2.3. Identificación de relaciones.....	17
3.2.4. Atributos de las entidades.....	18
3.3. Etapa de diseño lógico.....	19
3.3.1. Diagrama Lógico Relacional.....	19
3.4 Etapa de Diseño Físico.....	20
3.4.1. Acceso a Oracle.....	20
3.4.2. Tablespaces.	20
3.4.3. Usuarios.	21
3.4.4. Tablas.....	21
3.4.5. Índices.....	29
4. Implementación.....	30
4.1. Implementación de procedimientos de Alta, baja y	

modificación (ABM) de usuarios.....	30
4.2. Implementación de procedimientos de Alta, baja y	
modificación (ABM) de desarrolladores.	33
4.3 Implementación de procedimientos de Alta, baja y	
4.4 modificación (ABM) de aplicaciones.	35
4.5 Gestión de descarga de aplicaciones.	40
4.6 Procedimiento de consultas.....	41
4.7 Módulo de estadísticas.	48
5. Plan de pruebas.....	50
5.1. Secuencia de pruebas.....	51
5.2.1. Alta, baja y modificación de usuarios, desarrolladores y	
aplicaciones.	51
5.2.2. Descargas de aplicaciones.....	52
5.2.3. Pruebas de los procedimientos de consulta.....	53
5.2.4. Módulo estadístico.....	54
5.2.5. Comprobación de Logs.....	56
5.2.6. Resultados esperados.....	56
6. Análisis de riesgos y Plan de contingencias.....	57
7. Valoración económica y recursos necesarios.....	58
7.1. Valoración económica.....	58
7.2. Recursos necesarios.....	58
8. Conclusiones.....	59
9. Bibliografía.....	60

1. INTRODUCCIÓN.

1.1. Descripción general del proyecto.

Objetivos generales del TFC.

Desde un punto de vista estrictamente académico el objetivo del trabajo final de carrera es la realización de un proyecto para dar solución a un problema inspirado en el mundo real de modo que para su resolución sea necesaria la puesta en práctica de los conocimientos adquiridos durante la titulación.

Este trabajo final de carrera se centra principalmente en la utilización de los conocimientos adquiridos en las asignaturas de Bases de datos I y II así como de Ingeniería del programari. El uso del lenguaje PL/SQL y SQL dinámico así como el manejo de un SGBD como Oracle, de amplia difusión el mercado, son también objetivos del TFC.

El proyecto a desarrollar se encuadra dentro del convenio de colaboración de la UOC con la asociación mundial de desarrolladores de aplicaciones móviles. Esta asociación se ha decidido a dotar a su nueva plataforma centralizada de descargas para dispositivos móviles de una BD para la mejor gestión de la misma, en una primera fase, y de una aplicación de gestión, en una fase posterior. La asociación mundial de desarrolladores de aplicaciones móviles, como nuestro cliente, será la entidad con la que detallaremos los requisitos concretos del proyecto software.

Objetivos específicos del TFC.

El alcance de este proyecto se centrará en poder registrar la información necesaria con objeto de gestionar las aplicaciones móviles y su subida a la plataforma de descarga de acuerdo a los requisitos expresados por el cliente.

- Esta BD contemplará el almacenamiento y la gestión de la información originada por la actividad de la plataforma.

- También contemplará el cálculo y almacenamiento de estadísticas relevantes mediante el diseño de un módulo específico.

- El modelo de BD incluirá la gestión de:

- Los datos asociados a una aplicación móvil

- Los datos asociados a las empresas desarrolladoras de las aplicaciones

- Los datos de los usuarios que realizan descargas de las mismas.

- La BD también almacenará los datos asociados a la gestión de las descargas e incluirá los datos más relevantes asociados a cada descarga.

- La BD implementará los procedimientos de alta, baja y modificación asociados a aplicaciones, desarrolladores y usuarios finales así como el procedimiento para almacenar los datos de las descargas realizadas por cada usuario.

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

- También se implementará diversos procedimientos de consulta de datos específicos de acuerdo a los requerimientos del cliente.

- Por último se implementará un módulo estadístico para dar respuesta a una serie de requerimientos solicitados por el cliente.

1.2. Planificación detallada del proyecto.

De acuerdo al enunciado inicial tenemos el siguiente calendario con estas fechas críticas:

	INICIO	FINALIZACIÓN
PAC1	20/09/2012	08/10/2012
PAC2	09/10/2012	12/11/2012
PAC3	13/11/2012	13/12/2012
ENTREGA	14/12/2012	14/01/2012

	Inicio	Entrega	Duración en horas
TFC BD	19/09/12	14/01/13	309 horas
PAC1	20/09/2012	08/10/2012	47 horas
Descrip. del sistema	20/09/2012	21/09/2012	6 horas
Diagrama de Gantt	22/09/2012	25/09/2012	8 horas
Prev. entregas PAC2	26/09/2012	28/09/2012	9 horas
RR.HH. y materiales	01/10/2012	01/10/2012	3 horas
Análisis de riesgos	02/10/2012	03/10/2012	8 horas
Valoración inicial	03/10/2012	05/10/2012	10 horas
Envío Pac1	08/10/2012	08/10/2012	3 horas
PAC2 ANÁLISIS Y DISEÑO	09/10/2012	12/11/2012	79 horas
1.Análisis	09/10/2012	05/11/2012	50 horas
1.1.Anál. de requisitos	09/10/2012	05/11/2012	50 horas
2.Diseño	05/11/2012	12/11/2012	26 horas
2.1.Diseño Conceptual	05/11/2012	08/11/2012	20 horas

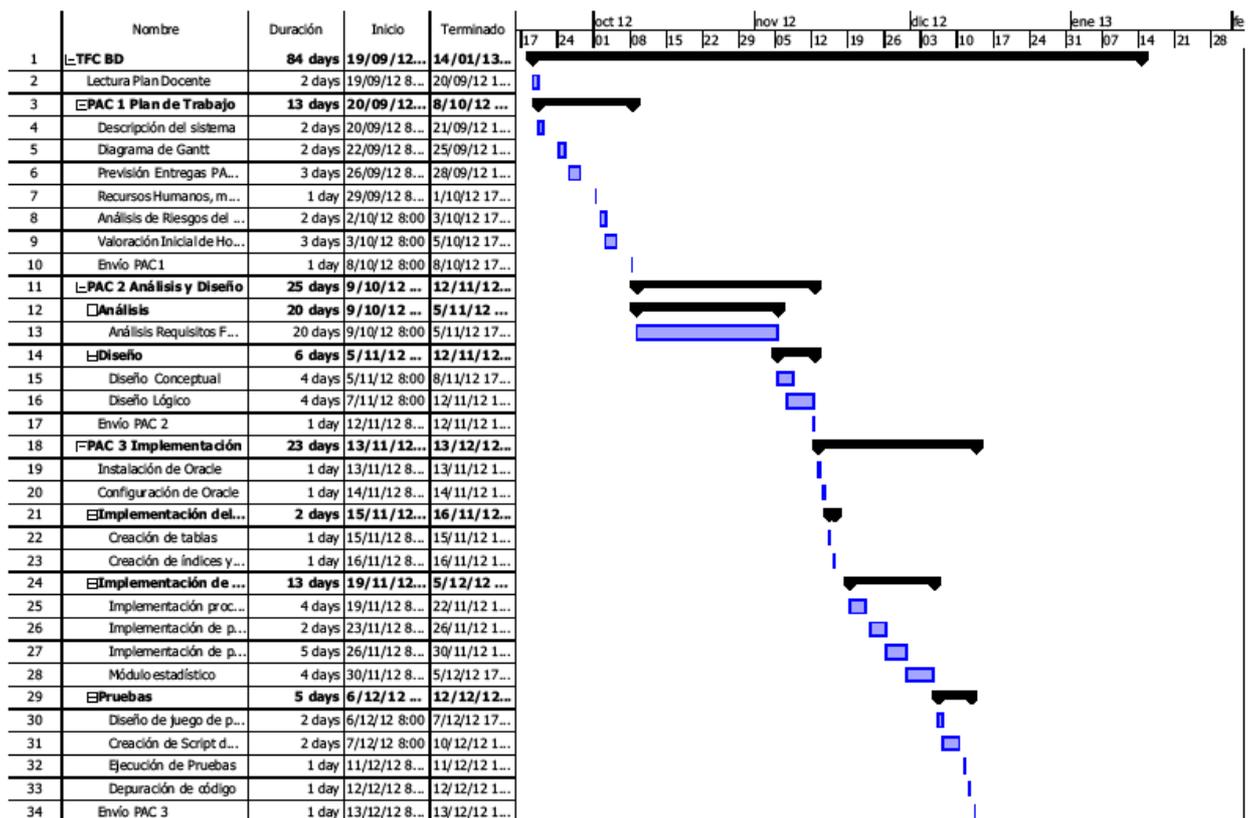
2.2.Diseño lógico	08/11/2012	12/11/2012	6 horas
Envío PAC2	12/11/2012	12/11/2012	3 horas
PAC 3 Implementación	13/11/2012	13/12/2012	90 horas
Instalación de Oracle	13/11/2012	13/11/2012	4 horas
Configuración de Oracle	14/11/2012	14/11/2012	3 horas
1.Implementación del diseño ER	15/11/2012	16/11/2012	8 horas
1.1.Creación de tablas	15/11/2012	15/11/2012	4 horas
1.2.Creación de índices y disparadores	16/11/2012	16/11/2012	4 horas
2.Implementación de funcionalidades	19/11/2012	5/12/2012	51 horas
2.1.Implementación procedimientos Alta + Baja + Mod	19/11/2012	22/11/2012	15 horas
2.2. Implementación de proc. de almacenamiento de descargas.	23/11/2012	26/11/2012	9 horas
2.3. Implementación de procedimientos de consulta	26/11/2012	30/11/2012	15 horas
2.4Módulo estadístico	30/11/2012	05/12/2012	12 horas
3. Pruebas	06/12/2012	12/12/2012	22 horas
3.1. Diseño de juego de pruebas	06/12/2012	07/12/2012	6 horas
3.2. Creación de Script de pruebas	07/12/2012	10/12/2012	8 horas
3.3.Ejecución de Pruebas	11/12/2012	11/12/2012	4 horas
3.4.Depuración de código	12/12/2012	12/12/2012	4 horas

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

Envío PAC 3	13/12/2012	13/12/2012	2 horas
Entrega Final TFC	14/12/12	14/01/13	93 Horas
1.PAC 3 Testing Exhaustivo	14/12/12	21/12/12	25 Horas
2.Redacción de la memoria	24/12/12	7/01/13	30 horas
3.Redacción presentación	8/01/13	11/01/13	15 horas
Entrega Final	14/01/13	14/01/13	3 horas

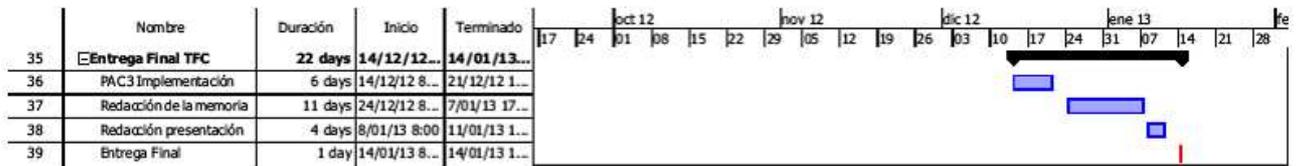
Diagrama de Gantt



Plan de trabajo - página1

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.



1.3. Productos entregados en las PAC y metodología de desarrollo asociada.

Para el desarrollo del proyecto adoptaremos el ciclo de vida clásico para el desarrollo del software. En este ciclo de vida los documentos producidos en cada fase nos servirán como punto de partida para el siguiente y consta de las siguientes fases:

Análisis previo. Se realiza una primera revisión del proyecto. En esta primera fase se determinan los recursos tanto técnicos como humanos. Se realiza también una planificación temporal de las tareas a realizar. Todo esto nos servirá para determinar la viabilidad del proyecto.

Análisis de requisitos. En esta fase se realiza de manera detallada las necesidades de requisitos expuestos por el cliente. En esta fase es necesaria una comunicación muy fluida con el cliente para determinar perfectamente lo que el cliente espera del producto a desarrollar.

Diseño. Se especifica las soluciones necesarias para los requisitos definidos en la fase anterior.

Implementación. Se lleva a cabo el proceso de codificación de la solución con ayuda de las herramientas de desarrollo necesarias. De esta manera obtendremos un primer producto software para dar respuesta a los requisitos del cliente.

Pruebas. Se realizan test del producto obtenido en la fase anterior para comprobar si satisfacen las necesidades del cliente de acuerdo a los requisitos. En caso de detectar desviaciones se realizarán las modificaciones pertinentes.

Mantenimiento. Una vez puesta a disposición del cliente se realizan labores de corrección de errores puestos de manifiesto con el uso o de implantación de nuevas funcionalidades requeridas por el cliente.

El análisis previo de este proyecto se ha realizado en la PAC1. Se determinan riesgos, medios humanos, materiales y técnicos necesarios, se hace una primera planificación temporal y se realiza una estimación del coste económico del mismo.

En la **PAC2 se ha llevado a cabo las fases de análisis de requisitos y diseño.**

En el análisis de requisitos se lleva a cabo un análisis del problema mediante entrevistas con el cliente para definir de manera exacta y concreta los requisitos y necesidades del mismo, de manera que los resultados del mismo sean de utilidad en la fase de diseño.

En la fase de diseño se procede a diseñar la base de datos de acuerdo a los pasos siguientes y distinguiendo tres fases:

- En el diseño conceptual identificamos las entidades sobre las que vamos a trabajar y como se relacionan entre ellas, así como sus atributos mediante un modelo UML y posteriormente mediante un modelo de Entidad - Relación definiendo entidades y relaciones de cardinalidad entre las mismas.
- Una vez determinadas las entidades podemos avanzar mediante el diseño lógico. Mediante un modelo Relacional avanzaremos en la definición de tablas así como de las restricciones dentro de las relaciones entre las diferentes tablas.

En la PAC 3 se han desarrollado las fases de implementación y pruebas del modelo en cascada de desarrollo del software.

Dentro de la fase de implementación se ha realizado una instalación del SGBD y mediante los lenguajes DDL y DML correspondientes se procede a la creación del esquema de tablas y a la definición de las relaciones entre las mismas con sus restricciones. Mediante el lenguaje PL/SQL y el uso de herramientas de desarrollo se implementan los procedimientos almacenados para ejecutar los procesos de alta, baja y modificación descritos en el enunciado, los procedimientos para almacenar las descargas realizadas y los procedimientos de consulta solicitados en los requisitos. Por último se ha desarrollado el módulo estadístico diseñado a partir de los procedimientos de consulta anteriormente solicitados.

Una vez terminada la fase de implementación se inicia la fase de pruebas para comprobar si se han cumplido los requisitos definidos por el cliente. Para esto se han diseñado juegos de pruebas y se han realizado las modificaciones pertinentes para corregir los errores que se han detectado.

La **Entrega final** se completa mediante una **memoria** donde se documentar las tareas realizadas en este proyecto software. Se elabora también una **presentación** para mostrar de manera resumida el trabajo realizado, así como un debate virtual sobre el proyecto.

2. ANÁLISIS DE REQUISITOS

2.1. Descripción inicial

La asociación mundial de desarrolladores de aplicaciones móviles ha establecido un marco de colaboración con la Universitat Oberta de Catalunya para el diseño de una BD que le permita a la asociación dar soporte a la plataforma centralizada de descargas de aplicaciones móviles. Esta asociación se marca como objetivo la unificación y mejora de la experiencia de descarga de las aplicaciones que sus asociados desarrollan para dispositivos móviles.

En un posterior desarrollo se implementará la aplicación de gestión de la plataforma de descargas a la que dará servicio la BD anteriormente mencionada.

La BD solicitada por nuestro cliente ha de almacenar la información más relevante relativa a los desarrolladores, las aplicaciones, los clientes de la plataforma y sobre todo la información que se genere en el proceso de las descargas de aplicaciones.

La BD deberá implementar los procedimientos de alta, baja y modificación de aplicaciones, desarrolladores y usuarios finales. Deberá asimismo implementar todo el procedimiento para almacenar los datos generados en el proceso de descargas de aplicaciones. También se contempla la programación de diversos procedimientos para la generación de los siguientes listados:

- El listado de los desarrolladores para un país dado con todos sus datos y el número de aplicaciones diferentes publicadas.

- Listar todas las aplicaciones activas ordenadas por nº. de descarga.

- Para una aplicación y un año concretos el listado de los países en que se haya descargado, así como el número de descargas por país.

- Listado con toda la actividad de descarga en la plataforma para un usuario identificado por su número de teléfono.

- Listar los 20 usuarios que más gasto han realizado en la plataforma para un año dado por orden.

Además la BD habrá de calcular y almacenar una serie de información estadística solicitada por la asociación y tenerla disponible para su consulta de acuerdo a lo especificado en el apartado del módulo estadístico. Los resultados a las consultas referentes a la mencionada se deben calcular en tiempo constante 1.

- Número total de descargas y la facturación hasta ahora mismo.

- El número medio de descargas por usuario para un año dado.

- El desarrollador con un mayor número de descargas, para un año dado y el número de las mismas.

- La aplicación que más dinero haya generado, así como su desarrollador, para un año dado.

- Para un año y un país dados, el número de usuarios diferentes que han hecho una descarga, los ingresos totales que han generado los usuarios de ese país y el número de aplicaciones descargadas por lo menos una vez.

2.2. Requisitos Funcionales

Del análisis del enunciado y de entrevistas con el cliente podemos deducir la siguiente relación de requisitos funcionales del sistema.

Referencia	Descripción
R01	El sistema debe permitir almacenar los datos mínimos asociados a una aplicación como son un identificador, versión, fecha de subida a la plataforma, desarrollador o desarrolladores de la misma, S.O. soportados por la aplicación y sus vínculos a los ficheros binarios para cada S.O, el precio y una descripción de la aplicación en función del país del cliente codificado mediante la notación ISO 3166-1 alfa-2, Un URL con un video de demostración de la aplicación, la resolución necesaria del dispositivo para la descarga y un campo binario que nos indique si la aplicación está activa para las descargas.
R02	Debe permitir guardar los datos relativos a los desarrolladores de las aplicaciones, en particular los siguientes, código identificativo del desarrollador, nombre de la empresa, nombre del representante legal de la empresa, país de registro de la empresa, dirección de las oficinas de la misma y teléfono.
R03	El sistema ha de permitir almacenar los datos referentes a un usuario activo para descarga de aplicaciones como es el número de móvil, el operador telefónico a que está abonado el usuario, el país en que está registrado el usuario, un mail del usuario. Así mismo debe poder almacenarse para cada usuario los IMEI de los dispositivos asociados al número de teléfono del usuario incluyendo el SO de cada dispositivo, su modelo y la resolución de la pantalla.
R04	El sistema ha de ser capaz de almacenar los datos asociados a las descargas de los usuarios que serán al menos la fecha de descarga, la aplicación descargada, el precio pagado por el mismo, el modo de pago, el país de registro del usuario, el IMEI en que el usuario ha descargado la aplicación y el operador telefónico del usuario en el momento de hacer la descarga.
R05	La aplicación debe de disponer, como mínimo, de las siguientes funcionalidades: <ul style="list-style-type: none"> • Procedimientos de ABM (Alta + Baja + Modificación) de las aplicaciones. • Procedimientos de ABM de los desarrolladores. • Procedimientos de ABM de los usuarios finales. • Se deben implementar los siguientes procedimientos de consulta que permitan obtener: <ol style="list-style-type: none"> a) El listado de los desarrolladores para un país concreto con todos sus datos incluyendo el número de aplicaciones diferentes publicadas. b) Listar todas las aplicaciones activas a día con sus principales datos y ordenadas por nº. de descargas hasta el día de hoy a nivel mundial. c) Para una aplicación y un año concretos el listado de los países en que se haya habido actividad de descarga de aplicaciones, así como el número de descargas para cada país. d) Dado un usuario final con un determinado número de teléfono listar toda su actividad de descargas en la plataforma reflejando los datos descargados, aplicaciones, precio pagado.. e) Listar los 20 usuarios que más gasto han realizado en la plataforma para un año dado por orden de mayor a menor. • Se valorará la implementación de otros procedimientos o funcionalidades que puedan ser de utilidad, siempre y cuando estén documentadas y consensuadas con el consultor.
R06	La aplicación deberá contemplar un módulo estadístico que se nutrirá de las anteriores implementaciones. El módulo estadístico mostrará los datos mediante un <i>SELECT</i> sobre una tabla

	<p>sin hacer uso de vistas calculadas o funciones de agregados.</p> <ul style="list-style-type: none"> a) Número total de descargas hasta el momento actual. b) La facturación acumulada por las descargas hasta el presente. c) La media de descargas por usuario para un año concreto. d) Para un año concreto, el desarrollador con un mayor número de descargas, y el número de las mismas. e) Tomando un año en particular obtener la aplicación que más dinero haya generado, así como su desarrollador. <p>Tomando un año y país concretos:</p> <ul style="list-style-type: none"> f) el número de usuarios diferentes que han realizado por lo menos una descarga. g) la facturación total generada por los usuarios de ese país. h) el núm. de aplicaciones que tienen al menos una descarga. <p>Las respuestas del módulo serán inmediatas y siempre actualizadas.</p>
--	--

Logs.

De acuerdo a los requisitos las llamadas a los procedimientos queden almacenadas en una tabla de Log donde se ha de almacenar el nombre procedimiento y sus parámetros de entrada y salida. Todos los procedimientos has de devolver un parámetro de tipo RSP. Este parámetro determinará si la ejecución ha finalizado con éxito (OK) o con un error ("Error" + tipo de error).Asimismo se implementará un sistema de tratamiento de excepciones.

2.3. Requisitos no funcionales

Los requisitos no funcionales a diferencia de los anteriores no se centran en lo que el sistema hará sino en cómo lo hará. A continuación se detallan una serie de los mismos basándonos en la documentación previa y en criterios de usabilidad comúnmente aceptados (tiempo de respuesta p.ej).

Tipo	Descripción
SGBD	El SGBD de acuerdo a la petición del cliente será Oracle. Es un SGBD de gran prestigio y amplia implantación en el mundo empresarial
Capacidad Actual y previsiones	El SGBD se ha dimensionado para una carga de trabajo mucho mayor que la real inicial de manera que podamos soportar un crecimiento del volumen de datos de la plataforma antes de tener que hacer ninguna labor de ampliación de capacidad. Hasta que se implemente la aplicación que se nutrirá de esta BD la carga de CPU y disco se mantendrá baja
Políticas de usuarios	Los usuarios tendrán inicialmente acceso total a todas las funcionalidades del sistema. No hay políticas establecidas respecto a la revocación de usuarios.
Tiempo de respuesta	El usuario podrá obtendrá las resultados de las diferentes funcionalidades en un tiempo no superior a 3 segundos.
Sistemas operativos	No se ha definido nada a este respecto por lo que la única limitación será la existencia de una versión del SGBD bajo el sistema operativo seleccionado.

3. DISEÑO DEL SISTEMA.

3.1. Diseño de la base de datos.

De acuerdo con la metodología de desarrollo comúnmente aceptada ¹ procederemos a descomponer el diseño de base de datos en tres etapas: diseño conceptual, diseño lógico y diseño físico. Antes de comenzar la implementación llevaremos a cabo la tarea del diseño físico creando en Oracle la Base de datos como tal, tablespaces, usuarios y las tablas con sus correspondientes campos y restricciones.

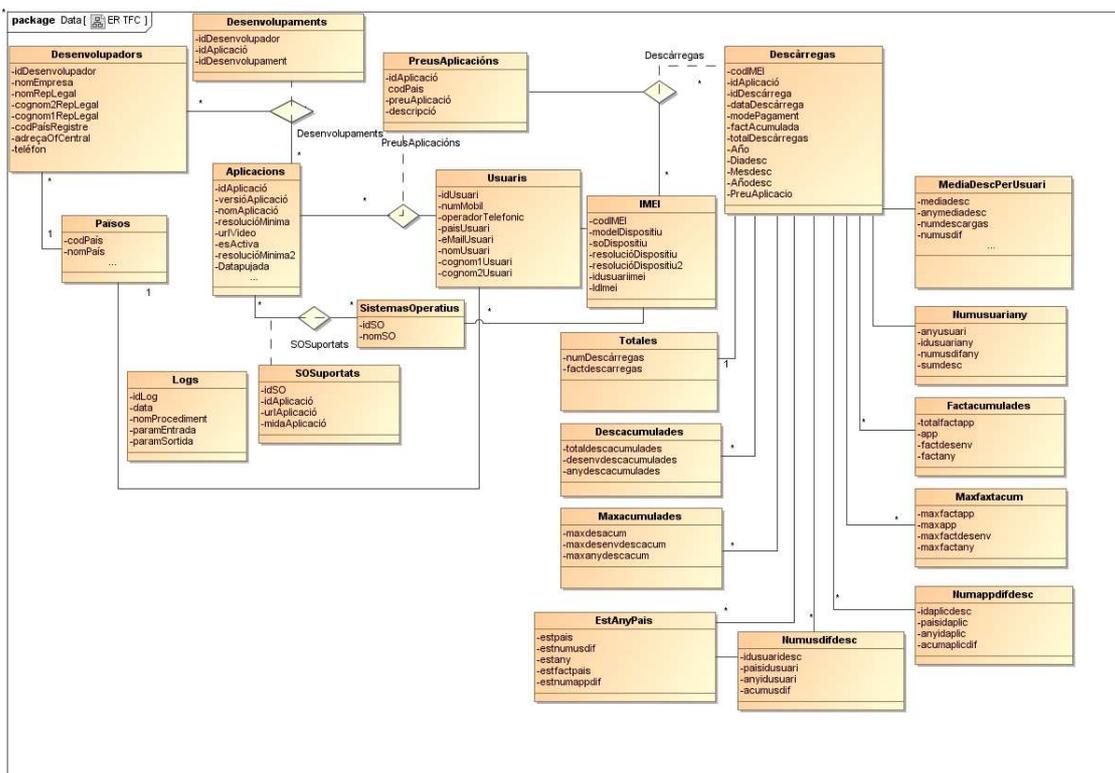
3.2. Etapa de Diseño conceptual.

En esta etapa se obtiene la estructura de la información de la base de datos sin condicionarnos por la tecnología que se adoptará en el futuro. Es decir aunque no hayamos decidido ni el tipo de base datos (que en este caso será relacional) ni el SGBD y lenguaje de implantación podemos modelar la estructura de la información. Como resultado de esta etapa obtendremos un modelo de datos de alto nivel que estará reflejado mediante un modelo UML

Para llevar a cabo el diseño conceptual del sistema tenemos que proceder a identificar las entidades que formarán parte del sistema y a definir las relaciones entre ellas.

Una vez definidas las relaciones entre las entidades con sus correspondientes cardinalidades definiremos los atributos de cada entidad.

3.2.1. Modelo UML



1 - PRESSMAN, R.S., *Ingeniería del Software. Un enfoque práctico*. 6ª ed. 2006: Mc Graw Hill, Interamericana de España S.A.U.

3.2.2. Identificación de Entidades

Las entidades, entendidas como objetos del mundo real sobre el cual queremos almacenar cierta información, las clasificaremos en entidades principales, intermedias y auxiliares.

Las entidades principales son las de mayor importancia para el sistema. Contemplan la información básica del sistema y el resto de tipos de entidades se estructura a su alrededor.

Las entidades intermedias aparecerán como resultado de relaciones N-M entre entidades principales. Aquí obtendremos una nueva tabla para reflejar esta entidad intermedia y su clave primaria será de tipo compuesta y a partir de las claves primarias pertenecientes a las entidades principales que forman esta relación N-M.

Las entidades auxiliares son necesarias por los datos que iremos almacenando en ellos y su existencia responde a algún tipo de requerimiento del sistema. En este caso usaremos entidades auxiliares para los datos que conformarán el módulo estadístico de modo que las consultas en este módulo cumplan la condición de tiempo constante 1 impuesta en el enunciado. También incluiremos aquí la tabla de Logs.

Entidades principales	Entidades intermedias:
Aplicacions	Descàrregas
Usuaris	PreusAplicacions
Desenvolupadors	SOSuportats
SistemesOperatius	Desenvolupaments
Països	

Entidades Auxiliares.

Totales Número de descargas y la facturación totales hasta ahora mismo.

MediaDescPerUsuari. Recoge para cada año el número media de descargas por usuario.

Numusuariany: El número de usuarios diferentes que han descargado por año.

Descacumulades: Recoge el número de descargas acumuladas para cada desarrollador en un año.

Maxacumulades: Recoge el número de descargas del desarrollador con más descargas por año.

Factacumulades: Guarda la el total de la facturación de una aplicación por año.

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

Maxfactacumulades: Guarda la aplicación que más ha facturado por año.

Numusidfdesc: Registra el número de usuarios diferentes por país y año que han descargado.

Munappdifdesc: Guarda el número de aplicaciones que se han descargado por año y país.

EstAnyPais. Entidad que recoge los ingresos totales generados por los usuarios registrados por país y año.

Logs

3.2.3. Identificación de relaciones.

Aplicacions –Sistemas Operatius: Una aplicación está disponible en varios Sistemas Operatius y un S.O. tiene múltiples aplicaciones. N-M

Aplicacions- Desenvolupadors: Una Aplicación puede estar desarrollado por uno o varios desarrolladores y un desarrollador puede diseñar múltiples aplicaciones. N-M

Desenvolupadors-Països: Un País puede tener muchos desarrolladores y un desarrollador sólo puede pertenecer a un país. N-1

Usuaris-Països: Un usuario pertenece a un país y un país tiene varios usuarios. N-1

Aplicacions-Usuaris: Una aplicación tiene varios precios para cada usuario en función de su país y un usuario en función de su país tiene diferentes precios para diferentes aplicaciones N-M

Usuaris – IMEI: Un usuario tiene varios *IMEI* (1 por terminal) y un *IMEI* sólo puede pertenecer a un terminal 1-N

PreusAplicación-IMEI: Un precio de una aplicación se aplica para muchos *IMEI* (que dependerá del país del usuario del *IMEI*) y un *IMEI* tiene varios precios de descarga de aplicaciones compatibles con el terminal asociado al *IMEI*. Esta asociación también comprenderá las descripciones de las aplicaciones. N-M

Relaciones del módulo estadístico

Descàrregas-Totales: El número total de descargas es único para las descargas efectuadas (será la suma de las filas de la tabla descàrregas) 1-1

Descàrregas-MediaDescPerUsuariAny: Habrá N años de descarga, una por cada año. N-1

Descarregas-numusuariany. Hay N cantidades de usuarios para cada año. N-1

EstAnyPais -Descàrregas: Habrá N países y años de descarga.N-1

Descarregas – descacumulades Hay N descargas por año acumuladas. N-1

Descarregas – maxacumulades. N máx. de descarga (uno por año) acumuladas N-1

TFC – Memoria Bases de dades relacionals

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

Descarregas – factcumulades. Hay N descargas facturadas por año N-1

Descarregas – maxfactacum. Hay N máx. de facturación (uno por año) acumulada N-1

Descarregas – numusdifdesc Tenemos N descargas para cada núm. de usuarios diferentes que han descargado por año.

Descarregas – numappdifdesc. Existen N descargas paracada núm. de aplicaciones diferentes que se han descargado por año.

3.2.4. Atributos de las entidades.

A continuación se relacionan los atributos de las entidades identificadas. Las claves primarias de las relaciones se especifican subrayadas.

APLICACIONES (idAplicació, nomAplicació, versióAplicació, urlVideo, resolucióMínima, resolucioMinima2, dataPujada, esActiva)

USUARIS (idUsuari, númMobil, nomUsuari, cognom1Usuari, cognom2Usuari, paísUsuari, operadorTelèfonic, eMailUsuari)

DESENVOLUPADORS (idDesenvolupador, nomEmpresa, nomRepLegal, cognom1RepLegal, cognom2RepLegal, codPaísRegistre,adreçaOficina,telèfon)

SISTEMASOPERATIUS (idSO, nomSO)

PAÏSOS (codPaís, nomPaís)

IMEI (codIMEI, modDispositiu, soDispositiu, resolucióDispositiu)

DESCÀRREGAS (codIMEI, idAplicació, idDescarrega, preuAplicació, dataDescarrega, modePagament,factAcumulada,totaldescarregas,Año,anodesc,diadesc,mesdesc)

PREUSAPLICACIONS (idAplicació, codPaís, preuAplicació, descripció)

SOSUPORTATS (idAplicació,idSO,urlAplicació,midaAplicació)

DESENVOLUPAMENTS (idDesenvolupador, idAplicació, idDesenvolupament)

LOGS (idLog, dataLog,nomProcediment, paramsEntrada, paramsSortida)

TOTALES (numdescarregas, factdescarregas)

MEDIADESCPERUSUARI (anymediadesc, mediadesc, numdescargas, numusdif)

NUMUSUARIANY (anyusuari, idusuariany, numusdifany, sumdesc)

DESCACUMULADES (totaldescacumulades, desenvdescacumulades, anydescacumulades)

MAXACUMULADES (maxdescacum, maxdesenvdescacum, maxanydescacum)

FACTACUMULADES (totalfactapp,app, factdesenv,factany)

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

MAXFAXTAXUM (maxfactapp,maxapp, maxfactdesenv,maxfactory)

NUMUSDIFDESC (idusuaridesc, paisidusuari, anyidusuari, acumusdif)

NUMAPPDIFDESC (idaplicdesc, paisidaplic, anyidaplic, acumaplicdif)

ESTANYPAIS (estpais,estany, estnumusdif, estfactpais, estnumappdif)

3.3. Etapa de diseño lógico

En esta etapa nos basamos en los resultados obtenidos en el diseño conceptual y los adaptamos a la tecnología que deseamos emplear.

Como vamos a utilizar un SGBD relacional en esta etapa se definirán un conjunto de relaciones. En este conjunto de relaciones estarán definidos sus atributos, claves primarias y foráneas.

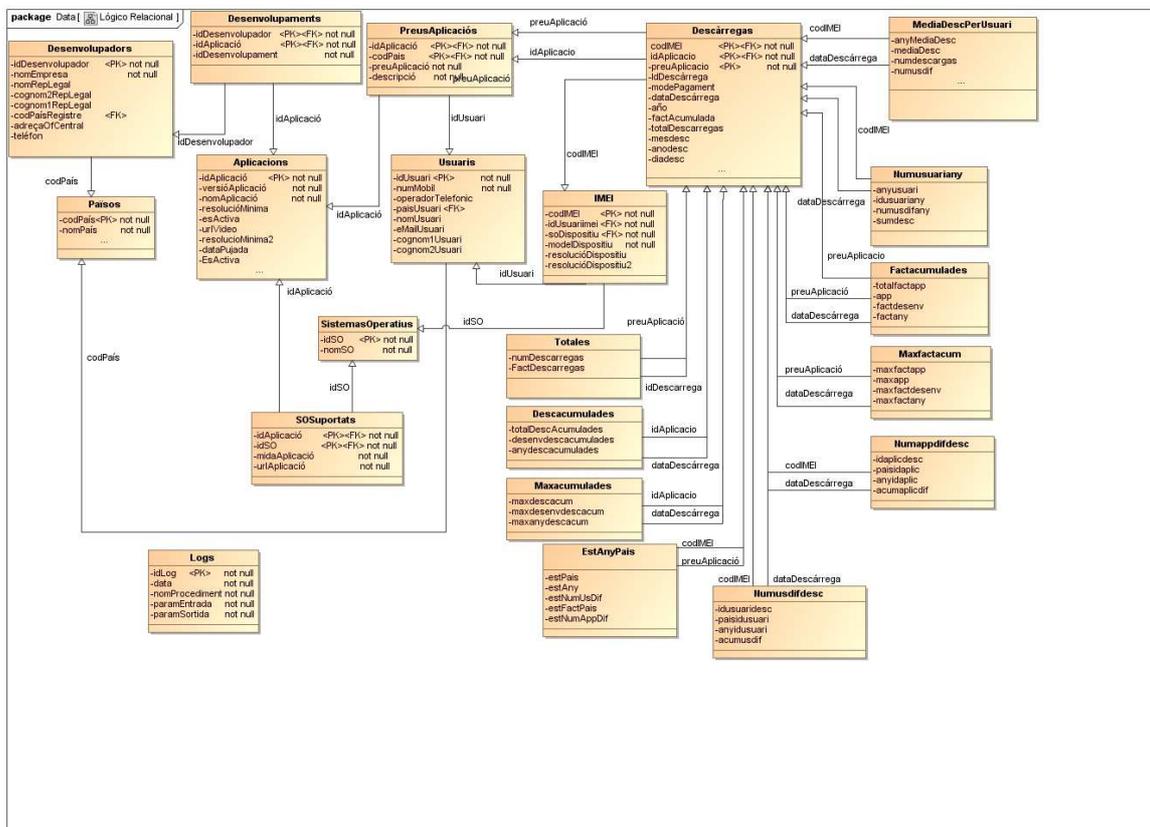
-Las entidades definidas en el diseño lógico originan relaciones

- Las interrelaciones binarias 1:1 y 1:N dan lugar a la aparición de claves foráneas

- Las interrelaciones binarias de tipo M:N generan nuevas relaciones.

En este diagrama se muestran las restricciones de clave primaria (PK) y foránea (FK).Las relaciones para las intermedias son a la vez foráneas entendidas una a una y primarias entendidas como una clave compuesta primaria.

3.3.1. Diagrama Lógico Relacional



3.4 Etapa de Diseño Físico

Para trasladar el diseño que hemos realizado a un SGBD debemos crear una estructura de tablas de base de datos. De acuerdo con los requisitos del cliente utilizaremos el SGBD Oracle.

3.4.1. Acceso a Oracle.

Accedemos mediante la herramienta SQL Developer y dentro de la conexión local System-XE vamos a definir tablespaces, usuarios y tablas de la BD. Para la conexión local nos pedirán definir una contraseña, en este caso "hola". Consideraremos que los clientes ya tienen una plataforma Oracle perfectamente instalada por lo que a partir de una conexión local con privilegios de administrador en la máquina del cliente definiremos usuario, tablespaces y tablas.

3.4.2. Tablespaces.

Se van a crear dos tablespaces, el primero para los datos con nombre ts_taulles y un segundo para los índices, de nombre ts_indexs. Las tablas y los índices estarán almacenados en sus correspondientes tablespaces. Estos tablespaces serán locales y con la característica de AUTOALLOCATE de manera que será el propio SGBD Oracle el que gestionará el tamaño necesario para la base de datos de manera que en caso de necesitar más espacio el propio SGBD se lo asignará. Esta opción tiene el inconveniente de que se puede generar cierta fragmentación de los datos. Por defecto vamos a asignar con 50 Mb a cada tablespace, que a priori parece espacio suficiente para el tipo de datos que vamos a manejar.

En caso de tablas muy grandes y con objeto de mejorar el rendimiento se puede optar por diseñar tablespaces de característica de UNIFORM.

Aquí se muestra el *script* de creación de las tablespaces. El *path* del Datafile se tiene que ajustar a nuestra instalación particular de Oracle. El *script* se encuentra en el directorio Producto "0-tablespaces.sql".

```
--tablespaces
--tablespace per taulles
CREATE TABLESPACE ts_taulles
DATAFILE 'C:\oracle\app\oracle\oradata\XE\taules.dbf' SIZE 50M
EXTENT MANAGEMENT LOCAL AUTOALLOCATE;
--tablespace per indexs
CREATE TABLESPACE ts_indexs
DATAFILE 'C:\oracle\app\oracle\oradata\XE\indexs.dbf' SIZE 50M
EXTENT MANAGEMENT LOCAL AUTOALLOCATE;
```

3.4.3. Usuarios.

Se definirá un usuario desde donde vamos a acceder al SGBD para crear las tablas, las relaciones entre ellas, las funciones y procedimientos así como los disparadores y demás elementos que necesitaremos para implementar todas las funcionalidades de la base de datos. Este usuario local se llamara TFC con roles CONNECT para poder crear tablas y vistas y RESOURCE para poder crear procedimientos y triggers. Se asignará cuota ilimitada para este usuario respecto a los tablespaces anteriormente creados.

El *script* de creación es “1-Creacion_usuario.sql” del directorio Producto. Las instrucciones a ejecutar serán las siguientes creando un usuario TFC con contraseña “hola”.

```
DROP USER TFC CASCADE;

CREATE USER TFC
  IDENTIFIED BY "hola"
  DEFAULT TABLESPACE TS_TAULES
  TEMPORARY TABLESPACE TEMP
  PROFILE DEFAULT
  ACCOUNT UNLOCK;

GRANT RESOURCE TO TFC;
GRANT CONNECT TO TFC;
GRANT CREATE VIEW TO TFC;
ALTER USER TFC DEFAULT ROLE ALL;

REVOKE UNLIMITED TABLESPACE FROM TFC;

ALTER USER TFC QUOTA UNLIMITED ON TS_TAULES;
ALTER USER TFC QUOTA UNLIMITED ON TS_INDEXES;
```

3.4.4. Tablas

Basándonos en el diseño lógico definido anteriormente y mediante las sentencias de Oracle correspondiente generamos la estructura de tablas de la base de datos. En este apartado se van a detallar los campos de cada tabla, los campos de las tablas que son NOT NULL (es decir han de estar siempre cumplimentados) y se van a definir sus claves primarias y foráneas.

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

El *script* de creación de las tablas es “2-creaciontablas.sql” y “8-CreacionTablasEst.sql” del directorio Producto. Además se crean los correspondientes ficheros de secuencias, triggers para insertarlas e índices en los ficheros “3-Secuencias.sql”, “4-Indices.sql” y “6-triggers.sql”.

APLICACIONES			
Campos	Tipo	Nulo	Descripción
IDAPLICACION	NUMBER	NOT NULL	Identificador de la aplicación
NOMAPLICACION	VARCHAR2(40)	NOT NULL	Nombre de la aplicación
VERSIONOAPLICACION	NUMBER	NOT NULL	Versión de la aplicación
URLVIDEO	VARCHAR2(100)		Dirección del video descriptivo de la aplicación
RESOLUCIONMINIMA	VARCHAR2(20)		Resolución horizontal de la aplicación
RESOLUCIONMINIMA2	VARCHAR2(20)		Resolución vertical de la aplicación
DATAPUJADA	DATE		Fecha de subida a la plataforma
ESACTIVA	VARCHAR(2)		Aplicación activa para la descarga
Constraint	Tipo	Tabla Referenciada	Campos
PK_IDAPLICACION	Primary key		IDAPLICACION

USUARIOS			
Campos	Tipo	Nulo	Descripción
IDUSUARIO	NUMBER	NOT NULL	Identificador del usuario
NUMMOBIL	NUMBER	NOT NULL	Número de tlf. del usuario
NOMUSUARIO	VARCHAR2(20)		Nombre del usuario
COGNOM1USUARIO	VARCHAR2(20)		Primer apellido del usuario
COGNOM2USUARIO	VARCHAR2(20)		Segundo apellido del usuario
PAISUSUARIO	VARCHAR2(2)		País del usuario
OPERADORTELEFONIC	VARCHAR2(20)		Operador de telefonía del usuario
EMAILUSUARIO	VARCHAR(40)		Dirección de correo del usuario
Constraint	Tipo	Tabla Referenciada	Campos

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

PK_IDUSUARI	Primary key		IDUSUARI
FK_USUARIPAISOS	Foreign key	PAISOS	PAISUSUARI

DESENVOLUPADORS			
Campos	Tipo	Nulo	Descripción
IDDESENVOLUPADOR	NUMBER	NOT NULL	Identificador del desarrollador
NOMEMPRESA	VARCHAR2(20)	NOT NULL	Nombre de empresa
NOMREPLEGAL	VARCHAR2(20)		Nombre del representante legal de la empresa
COGNOM1REPLEGAL	VARCHAR2(20)		Primer apellido del representante legal
COGNOM2REPLEGAL	VARCHAR2(20)		Segundo apellido del representante legal
CODPAISREGISTRE	VARCHAR2(2)		País del desarrollador
ADREÇAOFICINA	VARCHAR2(30)		Dirección de la oficina
TELEFON	NUMBER		Teléfono de la oficina del desarrollador
Constraint	Tipo	Tabla Referenciada	Campos
PK_IDDESENVOLUPADOR	Primary key		IDDESENVOLUPADOR
FK_DESENVOLUPADORPAISOS	Foreign key	PAISOS	PAISUSUARI

SISTEMASOPERATIUS			
Campos	Tipo	Nulo	Descripción
IDSO	NUMBER	NOT NULL	Código del Sistema Operativo
NOMSO	VARCHAR2(20)	NOT NULL	Nombre del Sistema Operativo
Constraint	Tipo	Tabla Referenciada	Campos
PK_CODPAIS	Primary key		IDSO

PAISOS			
Campos	Tipo	Nulo	Descripción
CODPAIS	VARCHAR(2)	NOT NULL	Código del país ISO 3166-1 alfa-2

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

NOMPAIS	VARCHAR2(180)	NOT NULL	Nombre del país
Constraint	Tipo	Tabla Referenciada	Campos
PK_CODPAIS	Primary key		CODPAIS

IMEI			
Campos	Tipo	Nulo	Descripción
IDIMEI	NUMBER	NOT NULL	Identificador del IMEI
CODIMEI	VARCHAR2(15)	NOT NULL	Número IMEI
MODDISPOSITIU	VARCHAR2(30)	NOT NULL	Modelo del dispositivo
SODISPOSITIU	NUMBER		Cód. de SO soportado por el dispositivo
RESOLUCIODISPOSITIU	NUMBER		Res. Horiz. Del dispositivo.
RESOLUCIODISPOSITIU2	NUMBER		Res. Vert. Del dispositivo.
IDUSUARIIMEI	NUMBER		ID del usuario asociado
Constraint	Tipo	Tabla Referenciada	Campos
PK_CODIMEI	Primary key		CODIMEI
FK_SODISPOSITIU	Foreign key	SISTEMASOPERATIUS	SODISPOSITIU
FK_IDUSUARI	Foreign key	USUARIS	IDUSUARIIMEI

DESCARREGAS			
Campos	Tipo	Nulo	Descripción
CODIMEI	VARCHAR2(15)	NOT NULL	Número IMEI
IDAPLICACIO	NUMBER		Id. De la aplicación
IDDESCARREGA	NUMBER		Número de descarga
PREUAPLICACIO	NUMBER		Precio de la aplicación
DATADESCARREGA	DATA		Fecha de descarga
AÑO	NUMBER		Año de descarga
MODEPAGAMENT	VARCHAR2(10)		Modo de pago
DIDESC	VARCHAR(2)		Día del mes de la descarga
MESDESC	VARCHAR(2)		Mes de la descarga

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

AÑODESC	VARCHAR(2)		Año de la descarga
Constraint	Tipo	Tabla Referenciada	Campos
PK_CODESCARREGAS	Primary key		CODIMEI, IDAPLICACIO, IDDESCARREGA

PREUSAPLICACION			
Campos	Tipo	Nulo	Descripción
IDAPLICACIO	NUMBER	NOT NULL	Identificador de la aplicación
CODPAIS	VARCHAR2(2)	NOT NULL	Código del País
PREUAPLICACIO	NUMBER	NOT NULL	Precio de la aplicación en el país de CODPAIS
DESCRIPCIO	VARCHAR2(100)	NOT NULL	Descripción de la aplicación
Constraint	Tipo	Tabla Referenciada	Campos
PK_PREUSAPLICACIONES	Primary key		IDAPLICACIO, CODPAIS
FK_PREUS_PAISOS	Foreign Key		CODPAIS
FK_IDAPLICACIO	Foreign Key		IDAPLICACIO

SOSUPORTATS			
Campos	Tipo	Nulo	Descripción
IDAPLICACIO	NUMBER	NOT NULL	Identificador de la aplicación
IDSO	NUMBER	NOT NULL	Código del SO para el que está diseñada la aplicación
URLAPLICACIO	VARCHAR2(200)	NOT NULL	Dirección del código fuente
MIDAAPLICACIO	NUMBER	NOT NULL	Tamaño en Bytes de la aplicación.
Constraint	Tipo	Tabla Referenciada	Campos
PK_SOSUPORTATS	Primary key		IDAPLICACIO, IDSO
FK_IDSO	Foreign Key		IDSO
FK_IDAPLICACIO	Foreign Key		IDAPLICACIO

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

DESENVOLUPAMENTS			
Campos	Tipo	Nulo	Descripción
IDDESENVOLUPADOR	NUMBER	NOT NULL	Identificador del desarrollador
IDAPLICACIO	NUMBER	NOT NULL	Identificador de la aplicación
IDDESENVOLUPAMENT	NUMBER	NOT NULL	Identificador del desarrollo
Constraint	Tipo	Tabla Referenciada	Campos
PK_SODESENVOLUPAMENTS	Primary key		IDDESENVOLUPADOR, IDAPLICACIO
FK_IDIDDESENVOLUPADOR	Foreign Key		IDDESENVOLUPADOR
FK_IDAPLICACIODESENVOLUPAMENT	Foreign Key		IDAPLICACIO

Tabla de Log:

La tabla de Log almacenará todos los procedimientos ejecutados en la base datos. Almacenará el nombre del procedimiento ejecutado, los parámetros de entrada, de salida y un campo datalog de fecha de creación.

LOGS			
Campos	Tipo	Nulo	Descripción
IDLOG	NUMBER		Identificador de la línea de Log
DATALOG	DATE		Fecha de ejecución
PROCEDIMENT	VARCHAR2(50)		Nombre del proc. Ejecutado
PARAMSENTRADA	VARCHAR2(1000)		Parámetros de entrada
PARAMSORTIDA	VARCHAR2(200)		Parámetros de salida
Constraint	Tipo	Tabla Referenciada	Campos
PK_IDLOG	Primary key		IDLOG

Tablas del módulo estadísticas.

Con objeto de responder en tiempo constante uno a una serie de consultas planteadas en los requerimientos se han diseñado tres tablas diferentes. En estas tres tablas se

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

almacenarán los resultados de estas consultas a medida que se vayan produciendo descargas en la plataforma mediante la ejecución de los disparadores correspondientes.

TOTALES			
Campos	Tipo	Nulo	Descripción
NUMDESCARREGAS	NUMBER		Número total de descargas.
FACTDESCARREGAS	NUMBER		Fact. Acumulada

MEDIADDESCPERUSUARI			
Campos	Tipo	Nulo	Descripción
ANYMEDIADDESC	NUMBER		Año para las medias
MEDIADDESC	NUMBER		Nº medio de descargas por usuario
NUMDESCARGAS	NUMBER		Nº de descargas por año
NUMUSDIF	NUMBER		Nº de usuarios diferentes que han descargado

NUMUSUARIANY			
Campos	Tipo	Nulo	Descripción
ANYUSUARI	NUMBER		Año de descarga
IDUSUARIANY	NUMBER		Id Usuario
NUMUSDIFANY	NUMBER		Num. Usuarios diferentes por año
SUMDESC	NUMBER		Total de desc. Por año

DESCACUMULADES			
Campos	Tipo	Nulo	Descripción
TOTALDESCACUMULADES	NUMBER		Número total de descargas
DESENVDESCACUMULADES	VARCHAR2(20)		Desarrollador
ANYDESCACUMULADES	NUMBER		Año de la descarga

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

MAXACUMULADES			
Campos	Tipo	Nulo	Descripción
MAXDESCACUM	NUMBER		Nº max de descargas
MAXDESENVDESCACUM	VARCHAR2(20)		Desarrollador
MAXANYDESCACUM	NUMBER		Año de la descarga

FACTACUMULADES			
Campos	Tipo	Nulo	Descripción
TOTALFACTAPP	NUMBER		Fact. Total para un año por aplic.
APP	VARCHAR2(40)		Aplicación
FACTDESENV	VARCHAR2(20)		Desarrollador
FACTANY	NUMBER		Año de facturación

MAXFACTACUM			
Campos	Tipo	Nulo	Descripción
MAXFACTAPP	NUMBER		Fact. Máxima para un año por aplic.
MAXAPP	VARCHAR2(40)		Aplicación
MAXFACTDESENV	VARCHAR2(20)		Desarrollador
MAXFACTANY	NUMBER		Año de facturación

ESTANYPAIS			
Campos	Tipo	Nulo	Descripción
ESTPAIS	VARCHAR2(2)		País de consulta
ESTANY	NUMBER		Año de consulta
ESTNUMSDIF	NUMBER		Núm. De usuarios diferentes que han realizado una descarga
ESTFACTPAIS	NUMBER		Facturación total por usuarios registrados en un país

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

ESTNUMAPPDIF	NUMBER		Número de aplicaciones diferentes descargadas una vez
--------------	--------	--	---

NUMUSDIFDESC			
Campos	Tipo	Nulo	Descripción
IDUSUARIDESC	NUMBER		Código de usuario
PAISIDUSUARI	VARCHAR2(2)		Código de país
ANYIDUSUARI	NUMBER		Año de la descarga
ACUMUSDIF	NUMBER		Suma de usuarios diferentes

NUMAPPDIFDESC			
Campos	Tipo	Nulo	Descripción
IDAPLICDESC	NUMBER		Cód. de aplicación
PAISIDAPLIC	VARCHAR2(2)		Código de país
ANYIDAPLIC	NUMBER		Año de la descarga
ACUMAPLICDIF	NUMBER		Suma de aplic. diferentes

3.4.5. Índices

Tomando las claves primarias de cada tabla Oracle crea directamente un índice único y no nulo que estará asociado a cada clave primaria. También se crean índices asociados a las claves foráneas.

4. IMPLEMENTACIÓN.

Una vez hemos definido las tablas que forman parte de la base de datos pasamos a implementar los procedimientos que nos permiten introducir datos en las tablas de acuerdo a los requerimientos de nuestro cliente.

Cada procedimiento tendrá una serie de parámetros de entrada en función de los datos a introducir y un único parámetro de salida que será OK si el procedimiento se ha ejecutado correctamente o un error con su descripción en caso de producirse. Los *scripts* de carga de estos procedimientos están en: "9-SP_ALTA_USUARI.sql", "10-SP_ANADIR_IMEI.sql", "11-SP_QUITAR_IMEI.sql", "12-SP_MODIF_USUARI.sql" y "13-SP_BAJA_USUARI.sql" dentro del directorio BD del producto entregado.

4.1. Implementación de procedimientos de Alta, baja y modificación (ABM) de usuarios.

PROCEDIMIENTOS DE ABM DE USUARIOS	
SP_ALTA_USUARI	
Nombre:	SP_ALTA_USUARI
Propósito:	Introduce un usuario a la tabla USUARIS. A la vez que los datos del usuario se ha de introducir un IMEI válido de un dispositivo que será el primer dispositivo asociado al número de móvil del usuario.
Parámetros entrada:	<ul style="list-style-type: none"> - p_nomusuari VARCHAR2(20) , nombre del usuario - p_cognom1usuari VARCHAR2(20) , primer apellido usuario - p_cognom2usuari VARCHAR2(20) , segundo apellido usuario - p_paisusuari VARCHAR2(2), Código país del usuario - p_operadortelefonic VARCHAR2(20) , operador telefónico - p_emailusuari VARCHAR2(40) , email usuario - p_codimei VARCHAR2(15), primer imei usuario,
Parámetros salida:	RSP
Precondición:	No existe el usuario en el sistema.
Postcondición:	Un usuario con un IMEI y núm. De teléfono válidos en el sistema
Retorna:	'OK' si la ejecución ha finalizado con éxito o un error dependiendo del mismo. ERROR: EL USUARIO YA ESTÁ DADO DE ALTA EN EL SISTEMA

<p>ERROR: EL CAMPO DEL NÚMERO DE MÓVIL ESTÁ EN BLANCO</p> <p>ERROR: EL PAÍS NO EXISTE</p> <p>ERROR: EL IMEI NO EXISTE</p> <p>ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)</p>	
<p>SP_ANADIR_IMEI</p>	
Nombre:	SP_ANADIR_IMEI
Propósito:	Asocia un IMEI con un usuario. Se han de introducir un número móvil dado de alta y un IMEI válidos y no asociado a ningún móvil.
Parámetros entrada:	<ul style="list-style-type: none"> - p_nummobil VARCHAR2(20) , nombre del usuario - p_codimei VARCHAR2(15), primer imei usuario,
Parámetros salida:	RSP
Precondición:	Un usuario válido
Postcondición:	Añade otro dispositivo con su correspondiente IMEI a un usuario
Retorna:	<p>'OK' si la ejecución ha finalizado con éxito</p> <p>ERROR: EL USUARIO NO ESTÁ DADO DE ALTA EN EL SISTEMA</p> <p>ERROR: UNO DE LOS CAMPOS ESTÁ EN BLANCO</p> <p>ERROR: EL IMEI YA ESTÁ ASOCIADO. NO DISPONIBLE</p> <p>ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)</p>
<p>SP_QUITAR_IMEI</p>	
Nombre:	SP_QUITAR_IMEI
Propósito:	Elimina la asociación de un IMEI con un usuario. Se ha de introducir un IMEI y un usuario válidos y asociados entre ellos
Parámetros entrada:	<ul style="list-style-type: none"> - p_nummobil VARCHAR2(20) , nombre del usuario - p_codimei VARCHAR2(15), primer imei usuario,
Parámetros salida:	RSP
Precondición:	Un usuario válido con un usuario asociado
Postcondición:	Elimina el dispositivo con su correspondiente IMEI al usuario

Retorna: 'OK' si la ejecución ha finalizado con éxito
ERROR: EL USUARIO NO ESTÁ DADO DE ALTA EN EL SISTEMA
ERROR: UNO DE LOS CAMPOS ESTÁ EN BLANCO
ERROR: EL IMEI NO EXISTE
ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

SP_MODIF_USUARI

Nombre: SP_MODIF_USUARI
Propósito: Modifica los datos de un usuario a la tabla USUARIS como nombre, apellidos, operador o email
Parámetros entrada:
- p_nummobil NUMBER , número de teléfono del usuario
- p_nomusuari VARCHAR2(20) , nombre del usuario
- p_cognom1usuari VARCHAR2(20) , primer apellido usuario
- p_cognom2usuari VARCHAR2(20) , segundo apellido usuario
- p_operadortelefonic VARCHAR2(20) , operador telefónico
- p_emailusuari VARCHAR2(40) , email usuario
Parámetros salida: RSP
Precondición: Un usuario válido
Postcondición: Un usuario válido con datos modificados
Retorna: 'OK' si la ejecución ha finalizado con éxito
ERROR: EL USUARIO NO ESTÁ DADO DE ALTA EN EL SISTEMA
ERROR: EL CAMPO DEL NÚMERO DE MÓVIL ESTÁ EN BLANCO
ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

SP_BAJA_USUARI

Nombre: SP_BAJA_USUARI
Propósito: Elimina un usuario a la tabla USUARIS. Elimina la vinculación de los IMEIS del usuario de manera que se puedan asignar a otro
Parámetros entrada:
- p_nummobil NUMBER , Número de teléfono del usuario

Parámetros salida:	RSP
Precondición:	Un usuario válido
Postcondición:	Un usuario borrado de la base datos. Elimina la vinculación de los IMEI
Retorna:	'OK' si la ejecución ha finalizado con éxito ERROR:EL USUARIO NO ESTÁ DE ALTA EN EL SISTEMA ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

4.2. Implementación de procedimientos de Alta, baja y modificación (ABM) de desarrolladores.

Los *script* de carga de estos procedimientos son: "14-SP_ALTA_DESENVOLUPADOR.sql", "15-SP_MODIF_DESENVOLUPADOR.sql" y "16-BAJA_DESENVOLUPADOR.sql" dentro del directorio BD del producto entregado.

PROCEDIMIENTOS DE ABM DE DESARROLLADORES	
SP_ALTA_DESENVOLUPADOR	
Nombre:	SP_ALTA_DESENVOLUPADOR
Propósito:	introduce un desarrollador a la tabla DESENVOLUPADORS
Parámetros entrada:	<ul style="list-style-type: none"> - p_nomempresa VARCHAR2(20): Nombre de la empresa desarrolladora. - p_nomreplegal VARCHAR2(20): Nombre del representante legal. - p_cognom1replegal VARCHAR2(20): Primer apellido del representante legal. - p_cognom2replegal VARCHAR2(20): Segundo apellido del representante legal. - p_codpaisregistre VARCHAR(2): Código del país de la empresa. - p_adreçaoficina VARCHAR2(30): Dirección de la empresa. - p_telefon NUMBER: Número de teléfono de

la empresa	
Parámetros salida:	RSP
Precondición:	No existe el desarrollador en el sistema.
Postcondición:	Un desarrollador dado de alta en el sistema.
Retorna:	'OK' si la ejecución ha finalizado con éxito ERROR: EL DESARROLLADOR YA ESTÁ DE ALTA ERROR: EL CAMPO NOMBRE DEL DESARROLLADOR ESTÁ VACIO ERROR: EL PAÍS NO EXISTE ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)
SP_MODIF_DESENVOLUPADOR	
Nombre:	SP_MODIF_DESENVOLUPADOR
Propósito:	Modifica los datos de un desarrollador en la tabla DESENVOLUPADORS
Parámetros entrada:	<ul style="list-style-type: none"> - p_nomempresa VARCHAR2(20): Nombre de la empresa desarrolladora. - p_nomreplegal VARCHAR2(20): Nombre del representante legal. - p_cognom1replegal VARCHAR2(20): Primer apellido del representante legal. - p_cognom2replegal VARCHAR2(20): Segundo apellido del representante legal. - p_adreçaoficina VARCHAR2(30): Dirección de la empresa. - p_telefon NUMBER: Número de teléfono de la empresa
Parámetros salida:	RSP
Precondición:	Exista el desarrollador en el sistema para modificar datos
Postcondición:	Un desarrollador con sus datos no esenciales modificados
Retorna:	'OK' si la ejecución ha finalizado con éxito

ERROR: EL DESARROLLADOR NO ESTÁ DADO DE ALTA EN LA PLATAFORMA ERROR: EL CAMPO DEL DESARROLLADOR ESTÁ EN BLANCO ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)	
SP_BAJA_DESENVOLUPADOR	
Nombre:	SP_BAJA_DESENVOLUPADOR
Propósito:	Da de baja a un desarrollador a la tabla DESENVOLUPADORS
Parámetros entrada:	- p_nomempresa VARCHAR2(20): Nombre de la empresa desarrolladora.
Parámetros salida:	RSP
Precondición:	Exista el desarrollador en el sistema
Postcondición:	Un desarrollador dado de baja de la plataforma
Retorna:	'OK' si la ejecución ha finalizado con éxito ERROR: EL DESARROLLADOR NO ESTÁ DADO DE ALTA EN LA PLATAFORMA ERROR: EL CAMPO DEL DESARROLLADOR ESTÁ EN BLANCO ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

4.3. Implementación de procedimientos de Alta, baja y modificación (ABM) de aplicaciones.

Los *script* de carga de estos procedimientos son: “17-SP_ALTA_APLICACION.sql”, “18-SP_ANADIR_SO.sql”, “19-SP_ANADIR_PRECIO.sql”, “20-SP_MODIF_PRECIO.sql”, “21-SP_MODIF_APLICACION.sql” y “22-SP_ACTIVAR.sql” dentro del directorio BD del producto entregado.

PROCEDIMIENTOS DE ABM DE APLICACIONES	
SP_ALTA_APLICACION	
Nombre:	SP_ALTA_APLICACION
Propósito:	Da de alta una aplicación en el sistema. En la tabla Aplicacions añade los datos que serán únicos para cada aplicación. En la tabla sosupportats daremos de alta los SO soportados con su tamaño y dirección URL. En la tabla preusAplicacions irá el precio para cada país.

Parámetros entrada:	
-	p_nomaplicacio VARCHAR2(40) , Identificador de la aplicación
-	p_versioAplicacio NUMBER , Versión de la aplicación
-	p_desenvolupadors VARCHAR2(20), Nombre del desarrollador
-	p_urlvideo VARCHAR2(100) , URL con una demo en video
-	p_resolucioMinima VARCHAR2(20), Resolución mínima
-	p_esActiva VARCHAR2(2) , La aplicación está activa(SI/NO)
-	p_idso NUMBER , código del S.O. de la aplicación
-	p_urlAplicacio VARCHAR2(200), Descarga de la aplicación
-	p_midaAplicacio NUMBER, tamaño de la aplicación en Bytes
-	p_codPais VARCHAR2(2), Código del País donde está disponible
-	p_preuAplicacio NUMBER , precio de la aplicación en ese País
-	p_descripcio VARCHAR2(100), Descripción de la aplicación
Parámetros salida: RSP	
Precondición:	Aplicación si dar de alta en el sistema
Postcondición:	Una aplicación con un precio y un SO soportado dada de alta en el sistema
Retorna:	'OK' si la ejecución ha finalizado con éxito ERROR: LA APLICACIÓN YA ESTÁ DADA DE ALTA EN LA PLATAFORMA ERROR: ALGUNO DE LOS CAMPOS OBLIGATORIOS ESTÁ VACÍO ERROR: EL PAÍS NO EXISTE ERROR: EL SO NO ESTÁ DADO DE ALTA EN LA PLATAFORMA ERROR: EL DESARROLLADOR NO EXISTE ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)
SP_ANADIR_SO	
Nombre:	SP_ANADIR_SO
Propósito:	Añade un sistema operativo soportado para una aplicación existente. Introduce en el sistema los datos de SO, url de la aplicación y el tamaño de la versión para ese S.O.
Parámetros entrada:	

-	p_nomaplicacio	VARCHAR2(40) , Identificador de la aplicación
-	p_idso	NUMBER , código del S.O. de la aplicación
-	p_urlAplicacio	VARCHAR2(200), Descarga de la aplicación
-	p_midaAplicacio	NUMBER, tamaño de la aplicación en Bytes
Parámetros salida: RSP		
Precondición:	Aplicación válida dada de alta en el sistema	
Postcondición:	Una aplicación con un SO soportado más, un tamaño en Bytes y una URL para su descarga	
Retorna:	'OK' si la ejecución ha finalizado con éxito	
	ERROR: LA APLICACIÓN NO ESTÁ DE ALTA EN LA PLATAFORMA	
	ERROR: ALGUNO DE LOS CAMPOS OBLIGATORIOS ES NULO	
	ERROR: EL SO NO ESTÁ SOPORTADO EN LA PLATAFORMA	
	ERROR: LA APLICACIÓN YA ESTÁ SOPORTADO POR ESE S.O.	
	ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)	

SP_ANADIR_PRECIO

Nombre:	SP_ANADIR_PRECIO	
Propósito:	Permite añadir países para los que la aplicación estará Disponible con sus precios y descripciones correspondientes	
Parámetros entrada:		
-	p_nomaplicacio	VARCHAR2(40) , Identificador de la aplicación
-	p_codPaís	VARCHAR2(2), Código del País donde está disponible
-	p_preuAplicacio	NUMBER , precio de la aplicación en ese País
-	p_descripcio	VARCHAR2(100), Descripción de la aplicación
Parámetros salida: RSP		
Precondición:	Aplicación válida dada de alta en el sistema	
Postcondición:	Una aplicación disponible en un país más, con su precio y descripción correspondientes	
Retorna:	'OK' si la ejecución ha finalizado con éxito	
	ERROR: LA APLICACIÓN NO ESTÁ DE ALTA EN EL SISTEMA	

<p>ERROR: ALGUNO DE LOS CAMPOS OBLIGATORIOS ES NULO</p> <p>ERROR: EL PAÍS NO EXISTE</p> <p>ERROR: YA HAY PRECIO ASIGNADO PARA ESE PAÍS</p> <p>ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)</p>	
<p>SP_MODIF_PRECIO</p>	
Nombre:	SP_MODIF_PRECIO
Propósito:	Permite modificar precios de aplicaciones disponibles
Parámetros entrada:	<ul style="list-style-type: none"> - p_nomaplicacio VARCHAR2(40) , Identificador de la aplicación - p_codPais VARCHAR2(2), Código del País donde está disponible - p_preuAplicacio NUMBER , precio de la aplicación en ese País - p_descripcio VARCHAR2(100), Descripción de la aplicación
Parámetros salida:	RSP
Precondición:	Aplicación válida dada de alta en el sistema
Postcondición:	Una aplicación con el precio o la descripción modificadas.
Retorna:	<p>'OK' si la ejecución ha finalizado con éxito</p> <p>ERROR: LA APLICACIÓN NO ESTÁ DE ALTA EN EL SISTEMA</p> <p>ERROR: ALGUNO DE LOS CAMPOS OBLIGATORIOS ES NULO</p> <p>ERROR: EL PAÍS NO EXISTE</p> <p>ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)</p>
<p>SP_MODIF_APLICACION</p>	
Nombre:	SP_MODIF_APLICACION
Propósito:	Permite modificar datos no esenciales de una aplicación como pueden ser la dirección con el video o la resolución.
Parámetros entrada:	<ul style="list-style-type: none"> - p_nomaplicacio VARCHAR2(40) , Identificador de la aplicación - p_urlvideo VARCHAR2(100) , URL con una demo en video - p_resolucioMinima VARCHAR2(20), Resolución mínima Ancho - p_resolucioMinima2 VARCHAR2(20), Resolución mínima Alto

Parámetros salida:	RSP
Precondición:	Aplicación válida dada de alta en el sistema
Postcondición:	Una aplicación con los datos no esenciales (resolución o url video) modificados
Retorna:	'OK' si la ejecución ha finalizado con éxito ERROR: LA APLICACIÓN NO ESTÁ DE ALTA EN LA PLATAFORMA ERROR: ALGUNO DE LOS CAMPOS OBLIGATORIOS ES NULO ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

SP_ACTIVAR

Nombre:	SP_ACTIVAR
Propósito:	Cambia el estado de la aplicación (Activa o no mediante un SI o NO)
Parámetros entrada:	<ul style="list-style-type: none">- p_nomaplicacio VARCHAR2(40) , Identificador de la aplicación- p_esActiva VARCHAR2(2) , Identificador de la aplicación
Parámetros salida:	RSP
Precondición:	Aplicación válida dada de alta en el sistema
Postcondición:	Cambia el estado de la aplicación a Activa (SI/NO) en función del parámetro de entrada para poder ser descargada
Retorna:	'OK' si la ejecución ha finalizado con éxito ERROR: LA APLICACIÓN NO ESTÁ DADA DE ALTA EN EL SISTEMA ERROR: UN CAMPO OBLIGATORIO ESTÁ VACÍO ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

4.4. Gestión de descarga de aplicaciones.

El *script* de carga de este procedimiento es: "23-DESCARGAR_APLICACION.sql" dentro del directorio BD del producto entregado.

PROCEDIMIENTOS DE DESCARGA DE APLICACIONES	
SP_DESCARGAR_APLICACION	
Nombre:	SP_DESCARGAR_APLICACION
Propósito:	Guarda los datos asociados a la descarga de una aplicación a un dispositivo de un usuario válido. Para ello se necesita introducir un nombre de aplicación un imei válidos
Parámetros entrada:	<ul style="list-style-type: none">- p_nomaplicacion VARCHAR2(40) , Identificador de la aplicación- p_codimei VARCHAR2(2), Código del Imei donde se va a descargar- p_modepagament VARCHAR2(10), Modo de pago- p_diadesc VARCHAR(2), Día de descarga- p_mesdesc VARCHAR(2), Mes de descarga- p_añodesc VARCHAR(2), Año de descarga
Parámetros salida:	RSP
Precondición:	Aplicación e imei de dispositivo con usuario asociado válidos y dado de alta en el sistema
Postcondición:	Registra la descarga en la tabla descarga
Retorna:	'OK' si la ejecución ha finalizado con éxito ERROR: ALGUNO DE LOS CAMPOS OBLIGATORIOS ES NULO ERROR: EL DISPOSITIVO ASOCIADO A ESTE IMEI NO EXISTE O NO TIENE USUARIO ERROR: LA APLICACIÓN NO ESTÁ DE ALTA O NO ESTÁ ACTIVA EN EL SISTEMA ERROR: NO EXISTE ESA APLICACIÓN PARA EL PAÍS DEL USUARIO ERROR: APLICACIÓN NO DISPONIBLE PARA EL SO DEL DISPOSITIVO ERROR: RESOLUCIÓN DE LA APLICACIÓN NO SOPORTADA POR EL DISPOSITIVO

ERROR: FECHA O FORMATO DE FECHA DE DESCARGA NO VALIDA. FORMATO DD MM AAAA ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

4.5. Procedimiento de consultas

De acuerdo a los requerimientos del cliente se han implementado cinco funciones para devolver los datos solicitados relativos a las descargas realizadas por los usuarios finales. Para diseñar estas funciones se ha utilizado un tipo TYPE donde vamos a almacenar la información devuelta por cada registro. Estos registros devueltos los mostraremos posteriormente mediante un objeto tipo tabla.

1. Listado de desarrolladores de un país dado incluyendo todos sus datos.
"24-FU_LIST_DESARROLLADORES_PAIS.sql"
2. Listar todas las aplicaciones activas ordenadas por nº. de descarga.
"25-FU_LIST_APLICACIONES_DESC.sql"
3. Listados de descargas de aplicaciones para un país dados un año y aplicación concretos.
"26-FU_LIST_APLIC_ANO_PAIS.sql"
4. Listado de toda la actividad de descarga de un usuario identificado por número de teléfono
"27-FU_LIST_ACTIVIDAD_USR.sql"
5. Listar los 20 usuarios que más gasto han realizado en la plataforma para un año dado por orden de gasto de más a menos gasto.
"28-FU_TOP20_USR.sql"

Las cinco funciones se encuentran dentro del directorio BD del producto entregado.

LISTADO DE DESARROLLADORES DE UN PAÍS	
FU_LIST_DESARROLLADORES_PAIS	
NOMBRE:	TFC.FU_LIST_DESARROLLADORES_PAIS
PROPÓSITO:	Función que devuelve un listado con todos los desarrolladores de un país que se le pasa como parámetro -Nombre de la empresa

```

-Nombre del representante legal
-Primer apellido representante legal
-Segundo apellido representante legal
-Dirección de la oficina
-Teléfono

```

Una vez finalizado inserta en la tabla de Log la el resultado de la ejecución, OK si ha ido bien o el Error en caso contrario

PARAMETROS:

INPUT: p_codpais: código del país

OUTPUT: RSP

RETORNA:

Si ha ido bien retorna un Listado con todos los desarrolladores del país pasado como parámetro

Si no ha ido bien alguno de los mensajes siguientes, dependiendo del error:

ERROR: EL PAIS ES OBLIGATORIO

EJECUTAR 'FU_LIST_DESARROLLADORES_PAÍS(CODPAIS)

ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

EJECUCION:

```
SELECT * FROM TABLE(FU_LIST_DESARROLLADORES_PAIS('ES'));
```

Tipos Asociados

```
--EN ESTE TIPO SE VAN A INSERTAR LOS REGISTROS CON LOS DATOS DE LOS
```

```
--DESARROLLADORES DE UN PAÍS DADO
```

```
drop type DESARROLLADORES_PAIS_OBJ force;
```

```
CREATE OR REPLACE TYPE DESARROLLADORES_PAIS_OBJ as object (
```

```
  nomempresa          VARCHAR2(20),
```

```
  numaplicacions      NUMBER,
```

```
  nomreplegal         VARCHAR2(20),
```

```
  cognom1replegal     VARCHAR2(20),
```

```
  cognom2replegal     VARCHAR2(20),
```

```
  adreçaoficina       VARCHAR2(30),
```

```
  telefon             NUMBER
```

```
);
```

LISTADO DE TODAS LAS APLICACIONES ACTIVAS POR Nº DE DESCARGA

FU_LIST_APLICACIONES_DESC

NOMBRE: TFC.FU_LIST_APLICACIONES_DESC

PROPÓSITO: Función que devuelve un listado con todas las aplicaciones descargadas con sus datos principales ordenadas por número de descarga. En caso de que la aplicación tenga varias versiones o esté disponible en varios países se mostrarán los datos por versión y país pero con el total absoluto de descargas.

- Número de descargas
- Nombre de la aplicación
- SO soportado por la aplicación
- Versión de la aplicación
- País donde se oferta la aplicación
- url del video
- resolución mínima ancho
- resolución mínima alto
- Si está activa
- Fecha de subida al sistema
- precio de la aplicación (por País)
- Descripción de la aplicación (Por país)
- url de la Aplicación (por SO Soportado)
- Tamaño de la aplicación (por SO Soportado)

Una vez finalizado inserta en la tabla de Log el resultado de la ejecución, OK si ha ido bien o el Error en caso contrario

PARAMETROS:

OUTPUT: RSP

DEVUELVE:

Si ha ido bien devuelve una lista de las aplicaciones activas ordenadas por número de descargas.

Si no ha ido bien alguno de los mensajes siguientes, dependiendo del error:

ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

EJECUCION:

```
SELECT * FROM TABLE(FU_LIST_APLICACIONES_DESC);
```

Tipos Asociados

----LISTADO DE LOS APLICACIONES ACTIVAS Y DATOS PRINCIPALES

--EN ESTE TIPO SE VAN A INSERTAR LOS REGISTROS CON LOS DATOS DE LAS

--APLICACIONES ACTIVAS ORDENADAS POR NÚMERO DE DESCARGA

```
drop type LIST_APLICACIONES_OBJ force;
```

```
CREATE OR REPLACE TYPE LIST_APLICACIONES_OBJ as object (
```

```
    numdescarregas      NUMBER,  
    nomaplicacio        VARCHAR2(40),  
    nomso                VARCHAR2(20),  
    codpais              VARCHAR2(2),  
    preuAplicacio       NUMBER,  
    descripcio          VARCHAR2(100)
```

```
);
```

LISTADO DE PAISES Y DESCARGAS PARA UNA APLICACIÓN Y AÑO CONCRETOS

FU_LIST_APLIC_ANO_PAIS

NOMBRE: TFC.FU_LIST_APLIC_ANO_PAIS

PROPÓSITO: Función que devuelve un listado con el número de descargas por país para una aplicación y año dados

-Número de descargas

-País de la descarga

Una vez finalizado inserta en la tabla de Log la el resultado de la ejecución, OK si ha ido bien o el Error en caso contrario

PARAMETROS:

INPUT: p_nomaplicacio: nombre de la aplicación, p_ano: año para el que

consultamos

OUTPUT: RSP

RETORNA:

Si ha ido bien retorna un Listado con el número de descargas por país para esa aplicación y año

Si no ha ido bien alguno de los mensajes siguientes, dependiendo del error:

'ERROR: EL NOMBRE O EL AÑO SON OBLIGATORIOS EJECUTAR:

FU_LIST_APLIC_ANO_PAIS(NOMAPLICACION,AÑO)'

ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

EJECUCION:

```
SELECT * FROM TABLE(FU_LIST_APLIC_ANO_PAIS('WORD', '2012'));
```

Tipos Asociados

----LISTADO DE LOS PAISES Y Nº DE DESCARGAS

--EN ESTE TIPO SE VAN A INSERTAR LOS REGISTROS CON LOS DATOS DE LOS

--PAISES Y Nº DE DESCARGAS

```
DROP TYPE APLIC_ANO_PAIS_OBJ FORCE;
```

```
CREATE OR REPLACE TYPE APLIC_ANO_PAIS_OBJ as object (
```

```
codpais          VARCHAR2(2),
```

```
numdescargas     NUMBER
```

```
);
```

LISTADO DE ACTIVIDAD DE DESCARGA POR USUARIO FINAL

FU_LIST_ACTIVIDAD_USR

NOMBRE: TFC.FU_LIST_ACTIVIDAD_USR

PROPÓSITO: Función que devuelve un listado la actividad de descargas por usuario en la plataforma por fecha y aplicación de descarga

-Fecha de descarga

-Nombre de aplicación de descarga

-Precio de la aplicación

-Modo de pago

Una vez finalizado inserta en la tabla de Log el resultado de la ejecución, OK si ha ido bien o el Error en caso contrario

PARAMETROS:

INPUT: p_nummobil: número de móvil del usuario para el que queremos ver su actividad

OUTPUT: RSP

RETORNA:

Si ha ido bien retorna un Listado con el histórico de actividades en la plataforma de descargas

Si no ha ido bien alguno de los mensajes siguientes, dependiendo del error:

ERROR: EL NÚMERO DE TELÉFONO ES OBLIGATORIO

EJECUTAR: FU_LIST_ACTIVIDAD_USR(TELÉFONO)

ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

EJECUCION:

```
SELECT * FROM TABLE(FU_LIST_ACTIVIDAD_USR('616634669'));
```

Tipos Asociados

----LISTADO DE LA ACTIVIDAD DE DESCARGA DE UN USUARIO

--EN ESTE TIPO SE VAN A INSERTAR LOS REGISTROS CON LOS DATOS DE LA

--ACTIVIDAD DE DESCARGA PARA UN USUARIO DADO

```
DROP TYPE ACTIVIDAD_USR_OBJ FORCE;
```

```
CREATE OR REPLACE TYPE ACTIVIDAD_USR_OBJ as object (
```

```
    datadescarrega      DATE,
    nomaplicacio         VARCHAR2(40),
    preuaplicacio        NUMBER,
    modepagament         VARCHAR2(10)
);
```

LISTADO DE LOS 20 USUARIOS CON MAYOR GASTO ACUMULADO PARA UN AÑO DADO

FU_TOP20_USR

NOMBRE: TFC.FU_TOP20_USR

```
PROPOSITO: Función que devuelve un listado de los 20 usuarios que más han
gastado en aplicaciones para una año dado

    -telefono
    -Nombre del usuario
    -Primer apellido del usuario
    -Segundo apellido del usuario
    -Facturación

Una vez finalizado inserta en la tabla de Log el resultado de la
ejecución, OK si ha ido bien o el Error en caso contrario

PARAMETROS:

    INPUT: p_ano: año para el que queremos hacer la consulta

    OUTPUT: RSP

RETORNA:

    Si ha ido bien retorna un Listado un listado de los 20 usuarios que más han
gastado en aplicaciones para una año dado

    Si no ha ido bien alguno de los mensajes siguientes, dependiendo del error:

        ERROR: EL AÑO ES OBLIGATORIO EJECUTAR: FU_TOP20_USR(AÑO)

        ERROR: TIPO DE ERROR (EN CUALQUIER OTRO CASO)

EJECUCION:

    SELECT * FROM TABLE(FU_TOP20_USR('2012'));
```

Tipos Asociados

```
----LISTADO DE LOS 20 USUARIOS CON MAYOR FACTURACIÓN DE DESCARGA PARA UN AÑO DADO
-----

--EN ESTE TIPO SE VAN A INSERTAR LOS REGISTROS CON LOS DATOS DE LOS 20
--USUARIOS CON MAYOR FACTURACIÓN DE DESCARGAS PARA UN AÑO DADO

DROP TYPE TOP20_USR_OBJ FORCE;

CREATE OR REPLACE TYPE TOP20_USR_OBJ as object (
    nummobil          NUMBER,
    nomusuari         VARCHAR2(20),
    cognom1usuari     VARCHAR2(20),
    cognom2usuari     VARCHAR2(20),
    facturacion       NUMBER
);
```

4.6. Módulo de estadísticas.

Para el desarrollo de las funcionalidades solicitadas por el cliente y que conforman el módulo de estadísticas se han diseñado una serie de disparadores de manera que una vez se realice una descarga en la aplicación el código del disparador actualice las tablas diseñadas para realizar las consultas solicitadas.

Así pues tendremos un trigger que actualizará la tabla totales con los datos de facturación y el número de descargas total de la plataforma cada vez que haya una descarga. (Consultas 1 y 2).

Otro trigger actualizará la tabla diseñada por años, mediadescperusuari, y para cada descarga de un año irá actualizando el número medio de descargas por usuario para ese año. Tendremos otro disparador que, también para cada año, actualizará al desarrollador con el mayor número de descargas así como el número de las mismas en la tabla maxacumulades. Por último un último trigger irá almacenando o actualizando el nombre de la aplicación que más ha facturado en descargas y el nombre de su desarrollador en la tabla maxfactacum. (Consultas 3,4 y 5)

La última tabla sobre la que actualizaremos es estanypais. Aquí un trigger actualizará la tabla para cualquier cambio en la combinación de Año y País de descarga almacenando el número de usuario que ha realizado al menos una descarga, los ingresos generados por usuarios para un mismo año y país y el número de aplicaciones descargadas al menos una vez para un mismo año y país. (Consultas 6, 7 y 8). El *script* de creación de los trigger se halla en los ficheros :

“29-TR_AFT_INS_DESC_EST1_2_TC1.sql”,

“30-TR_AFT_INS_DESC_EST3_TC1.sql”,

“31-TR_AFT_INS_DESC_EST4_TC1.sql”,

“32-TR_AFT_INS_DESC_EST5_TC1.sql”

y “33-TR_AFT_INS_DESC_EST6_TC1.sql”.

Se encuentran dentro del directorio BD del producto entregado.

Pasamos a detallar el nombre y funcionamiento de cada trigger implementado:

Trigger	Se activa cuando...	Acción
TR_AFT_INS_DESCARREGAS_EST1_2	Después de cada inserción en la tabla descarregas	Actualiza los totales de descargas y de facturación de la tabla totales
TR_AFT_INS_DESCARREGAS_EST3	Después de cada inserción en la tabla descarregas	Actualiza o inserta para el año de la descarga el número medio de descargas por usuario en la tabla mediadescperusuari
TR_AFT_INS_DESCARREGAS_EST4	Después de cada inserción en la tabla descarregas	Actualiza o inserta para cada año el nombre del desarrollador con mayor

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

		número de descargas y el número de las descargas del mismo en la tabla maxacumulades.
TR_AFT_INS_DESCARREGAS_EST5	Después de cada inserción en la tabla descarregas	Actualiza o inserta para cada año el nombre de la aplicación que más dinero ha recaudado en descargas, así como la cantidad recaudada y el nombre de su desarrollador en la tabla maxfactacum.
TR_AFT_INS_DESCARREGAS_EST6	Después de cada inserción en la tabla descarregas	Actualiza o inserta después de cada descarga en un año y país dado los datos de: -Número de usuarios diferentes que han descargado al menos una aplicación. -Ingresos totales generados por usuarios registrados -Número de aplicaciones descargadas al menos una vez. Estos datos se almacenarán en la tabla estanypais

5. PLAN DE PRUEBAS.

Con objeto de comprobar si la base de datos diseñada e implementada responde a los requerimientos expuestos por el cliente se van a diseñar y ejecutar una serie de sentencias SQL. Mediante la correcta ejecución de las sentencias y la salida de nuestro fichero de LOG podremos determinar que las funcionalidades que han sido implementadas responden a estos requerimientos.

5.1. Carga inicial de datos.

El primer paso es la creación de la BD mediante la creación de los tablespaces de la BD mediante la ejecución del *script* "0-tablespaces.sql" en SQLPLUS por un usuario con privilegios de administrador (por. Ej. SYSTEM) y la posterior ejecución en sqlplus del fichero CARGA_BD del directorio producto. En caso de que el usuario SYSTEM tenga otra contraseña hay que editar la línea *conn system/hola* sustituyendo "hola" por la contraseña correcta.

El fichero "CARGA_BD.sql" cargará todas las tablas y demás elementos de la base de datos diseñados hasta ahora. El último fichero al que llama "34-CARGAR_SO_IMEI_PAIS" realiza una carga mediante Instrucciones Insert de 4 sistemas operativos, 9 *imei* asociados a un dispositivo y 4 países codificados de acuerdo a las especificaciones del cliente. Para simplificar el plan de pruebas hemos permitido *imei* de menos de 15 dígitos, dejando esta comprobación como una posible mejora posterior del producto. Esta carga puede generar algún error inicial debido a las órdenes de borrado de tablas o demás elementos que inicialmente no existen y que hemos introducido para que podamos repetir el proceso de carga las veces que sea necesario.

Una vez cargada la BD con los datos mínimos vamos a ejecutar el *script* "CARGA_PROC.sql" que realizará la carga de usuarios asociándoles un primer *imei* de un dispositivo ("P2-INSERTAR_USUARIOS.sql") y la carga de desarrolladores y aplicaciones ("P7-ALTA_DESENV-APLIC.sql") mediante el uso de los procedimientos diseñados con tal fin.

A continuación el *script* añadirá *imei* a usuarios ("P3-ANADIR_IMEI_A_USUARIO.sql") y dará de alta aplicaciones en diferentes países con sus correspondientes precios y descripciones en el lenguaje de cada país ("P13-AÑADIR_PRECIO.sql").

También dará de alta a diferentes SS.OO. soportados para cada aplicación ("P12-AÑADIR_SO.sql") de modo que la aplicación se pueda bajar desde móviles con diferentes sistemas operativos.

Activará alguna aplicación en caso de haberse dado de alta sin activar ("P10-ACTIVAR_APLICACION.sql") y por último realizará varias descargas de prueba ("P15-DESCARGAR_APLICACIONES.sql") La llamada al procedimiento de las descargas pide una fecha de descarga para realizar estadísticas de descargas por año, cargando descargas de años diferentes.

Comprobaremos que todos los procedimientos se han ejecutado sin fallos y que los datos se han cargado correctamente realizando una consulta sobre la tabla de log y comprobando que el campo “paramssortida” siempre tiene el valor OK:

```
“SELECT DATALOG,PROCEDIMENT, PARAMSSORTIDA FROM LOGS ORDER BY DATALOG;”
```

5.1. Secuencia de pruebas

La primera fase de pruebas va a comprobar que los procedimientos de Alta, baja y modificación respondan correctamente a lo que se les pide. Esto es, que en caso de que los datos introducidos sean correctos, realicen el procedimiento correspondiente y en caso contrario notifiquen el error producido para que el usuario sea consciente del problema.

Se van a probar los procedimientos de Alta, baja y modificación tanto introduciendo datos correctos como erróneos de manera que cada procedimiento nos informará del resultado de la ejecución.

En lo respecto al procedimiento de descargas se ha probado con datos correctos y con todos los tipos de fallos que generan cada una de las excepciones registradas.

Estas pruebas han tratado de cubrir los casos más representativos que se pueden dar en la realidad.

Las pruebas que se enumeran se pueden ejecutar mediante el fichero “P-16-PRUEBAS_ABM_DESCARGAS.sql”, que genera el correspondiente “P-16logPRUEBAS_ABM_DESCARGAS.log” donde podemos comprobar el resultado de cada prueba.

5.2.1. Alta, baja y modificación de usuarios, desarrolladores y aplicaciones.

1. ALTA USUARIO ESTANDAR.
2. MODIFICAR USUARIO EXISTENTE.
3. MODIFICAR USUARIO QUE NO EXISTE.
4. AÑADIR IMEI A USUARIO.
5. QUITAR IMEI A USUARIO.
6. DAR DE BAJA UN USUARIO QUE NO EXISTE.
7. DAR DE BAJA UN USUARIO QUE EXISTE.
8. ALTA USUARIO ESTANDAR BORRADO.
9. ALTA USUARIO ESTANDAR SIN PAIS VALIDO.
- 10 .ALTA USUARIO ESTANDAR SIN IMEI VALIDO.

11. ALTA USUARIO ESTANDAR SIN TELEFONO.
12. AÑADIR IMEI A USUARIO QUE NO EXISTE.
13. QUITAR IMEI A USUARIO SIN ESE IMEI ASIGNADO.
14. ALTA DESARROLLADOR ESTANDAR.
15. ALTA DESARROLLADOR QUE EXISTE.
16. MODIFICAR DESARROLLADOR QUE NO EXISTE.
17. BAJA DESARROLLADOR QUE NO EXISTE.
18. BAJA DESARROLLADOR QUE EXISTE.
19. ALTA APLICACIÓN ESTANDAR QUE EXISTE.
20. ALTA APLICACIÓN ESTANDAR.
21. AÑADIR SO A APLICACION QUE NO EXISTE.
22. AÑADIR PRECIO A APLICACION QUE NO EXISTE.
23. AÑADIR SO A APLICACION QUE EXISTE.
24. AÑADIR PRECIO A APLICACION QUE EXISTE.
25. ACTIVAR APLICACIÓN QUE NO EXISTE.
26. DESACTIVAR APLICACIÓN QUE EXISTE.
27. ACTIVAR APLICACIÓN QUE EXISTE.
28. MODIFICAR PRECIO DE APLICACIÓN.
29. MODIFICAR PRECIO DE APLICACION QUE NO EXISTE.
30. MODIFICAR PRECIO DE APLICACION SIN DATOS.

5.2.2. Descargas de aplicaciones.

31. ALTA DESCARGA APLICACION QUE NO EXISTE.
32. ALTA DESCARGA IMEI QUE NO EXISTE.
33. ALTA DESCARGA CORRECTA.
34. ALTA DESCARGA CON FECHA ERRÓNEA.
35. ALTA DESCARGA NO DISPONIBLE EN EL SO DEL DISPOSITIVO.

36. ALTA DESCARGA CON RESOLUCION NO SOPORTADO POR EL DISPOSITIVO.

37. ALTA DESCARGA NO DISPONIBLE EN EL PAIS DEL CLIENTE.

38. DESACTIVAR APLICACIÓN.

39. ALTA DESCARGA CON APLICACIÓN NO ACTIVA PARA DESCARGAS.

40. ACTIVAR APLICACIÓN.

5.2.3. Pruebas de los procedimientos de consulta

A continuación se prueba las funciones definidas para realizar las consultas solicitadas en el apartado R5 de los requerimientos del cliente *Script* ("P20-SELECTSR5.sql"). Calcula los resultados a partir de una carga de datos sencilla con 9 desarrolladores, 9 aplicaciones, 9 usuarios. Se han cargado inicialmente para las pruebas 4 países, 10 imeis (1 por usuario excepto a uno que le asignamos 2) y 4 sistemas operativos para móviles. Para el país ES y para el sistema operativo IOS se han dado de alta todas las aplicaciones. Para el resto de países sólo una selección de las mismas.

```
-- Consulta 1: listado con todos los desarrolladores de un país
```

```
-----  
SELECT NOMEMPRESA AS NOMBRE, NUMAPLICACIONES AS NUM,  
NOMREPLEGAL AS NOMBRE, COGNOM1REPLEGAL AS APELLIDO,COGNOM2REPLEGAL AS APELLIDO,  
ADREÇAOFICINA AS DIRECCIÓN,  
TELEFON AS TLF FROM TABLE(FU_LIST_DESARROLLADORES_PAIS('ES'));
```

```
---- Consulta 2: Función que devuelve un listado con todos las aplicaciones descargadas  
-- con sus datos principales ordenadas por número de descarga. En caso  
-- de que la aplicación tenga varias versiones o esté disponible en varios  
-- países se mostraran los datos por versión y país pero con el total  
-- absoluto de descargas.
```

```
-----  
SELECT NUMDESCARREGAS AS NÚMERO, NOMAPLICACION AS NOMBRE,  
NOMSO AS SO, CODPAIS AS PAIS, PREUAPLICACION AS PRECIO, DESCRIPCION AS DESCRIPCION FROM  
TABLE(FU_LIST_APLICACIONES_DESC);
```

```
-- Consulta 3: Función que devuelve un listado con el número de descargas por país
```

```

-- para una aplicación y año dados
-----
SELECT      CODPAIS      AS      PAIS,NUMDESCARGAS      AS      NÚMERO      FROM
TABLE(FU_LIST_APLIC_ANO_PAIS('GRADIUS','2011'));

-- Consulta 4: Función que devuelve un listado la actividad de descargas por usuario
-- en la plataforma por fecha y aplicación de descarga
-----

SELECT DATADESCARREGA AS FECHA,
NOMAPLICACIO AS NOMBRE,
PREUAPLICACIO AS PRECIO,
MODEPAGAMENT AS PAGO
FROM TABLE(FU_LIST_ACTIVIDAD_USR('616634669'));

--Consulta 5: Función que devuelve un listado de los 20 usuarios que más han
--gastado en aplicaciones para una año dado
-----

SELECT NUMMOBIL AS NUMERO,
NOMUSUARI AS NOMBRE,
COGNOM1USUARI AS APELLIDO,
COGNOM2USUARI AS APELLIDO,
FACTURACION
FROM TABLE(FU_TOP20_USR('2011'));

```

5.2.4. Módulo estadístico

En este apartado comprobamos el correcto funcionamiento del módulo de estadísticas. Se comprueba mediante el *script* “P21-ESTADISTICAS.sql” y se puede ver el resultado de la ejecución en el correspondiente log “P21log-ESTADISTICAS.log”

```

---CONSULTA 1. Número total de descargas de la plataforma hasta ahora mismo
-----

SELECT NUMDESCARREGAS AS N°DESCARGAS FROM TOTALES;

---CONSULTA 2. Número total de € generados en descargas en la plataforma

```

```
-----  
SELECT FACTDESCARREGAS AS FACTURACIÓN FROM TOTALES;
```

```
---CONSULTA 3. Para un año concreto nº medio de descargas realizadas por usuario  
-----
```

```
SELECT MEDIADESC AS MEDIA FROM MEDIADESCPERUSUARI  
WHERE ANYMEDIADESC = 2011;
```

```
---CONSULTA 4. Para un año concreto desarrollador con mayor num de descargas  
-- y número de las mismas  
-----
```

```
SELECT MAXDESCACUM AS NºDESCARGAS, MAXDESENVDESCACUM AS NOMBRE FROM MAXACUMULADES  
WHERE MAXANYDESCACUM = 2010;
```

```
---CONSULTA 5. Para un año concreto aplicación que más dinero ha recaudado  
--- y su desarrollador  
-----
```

```
SELECT MAXFACTDESENV AS DESARROLLADOR, MAXAPP AS APLICACIÓN FROM MAXFACTACUM  
WHERE MAXFACTANY = 2004;
```

```
---CONSULTA 6. Para un año y un país concretos número de usuarios que han tenido  
-- al menos una descarga  
-----
```

```
SELECT ESTNUMUSDIF AS USUARIOS FROM ESTANYPAIS  
WHERE ESTPAIS= 'ES' AND ESTANY = 2011;
```

```
---CONSULTA 7. Para un año y un país concretos los ingresos totales que han  
-- generado en descargas los usuarios de ese país.  
-----
```

```
SELECT ESTFACTPAIS AS FACTURACIÓN FROM ESTANYPAIS  
WHERE ESTPAIS= 'ES' AND ESTANY = 2011;
```

```
---CONSULTA 8. Para un año y un país concretos número de aplicaciones descargadas  
-- al menos una vez.
```

```
-----  
SELECT ESTNUMAPPDIF AS NUMERO FROM ESTANYPAIS  
WHERE ESTPAIS= 'ES' AND ESTANY = 2011;
```

5.2.5. Comprobación de Logs.

Podemos comprobar que la tabla de logs se ha ido generando a medida que invocamos los procedimientos mediante: *SELECT * FROM LOGS ORDER BY IDLOG;*

5.2.6. Resultados esperados.

En la carga inicial de datos todas las RSP del fichero de LOG son OK por lo que la carga se ha realizado correctamente.

En las pruebas de procedimientos ABM los resultados del fichero de LOG se corresponden con los datos introducidos en los procedimientos correspondientes, generando excepciones en los casos que el procedimiento no puede tratar.

Observando los resultados de las consultas extraemos las siguientes conclusiones:

- Existen dos desarrolladores afincados en España
- Dado que tanto *whatsapp* como *Gradius* se han bajado 8 veces se muestran ambos. Se muestran tantas veces como combinaciones de SO y países se han publicado. Se podría optar por no repetir los datos pero perderíamos información acerca de los precios y SO en que están disponibles para cada aplicación.
- En 2011 se descargó 3 veces *Gradius* en España y una en Francia.
- El usuario del teléfono 616634676 ha bajado el *Whatsapp* en 4 años diferentes.
- Los tres usuarios con más facturación en 2011 son Aroa López, Carmen Díaz y Julio Carballada.

Respecto al módulo estadístico podemos ver que:

- Hemos realizado 16 descargas.
- La plataforma ha facturado 47,84€
- El número medio de descargas por usuario para el año 2011 es 2
- En el año 2010 el desarrollador con más descargas es *SCEE* con un total de 1
- Para el año 2004 la aplicación que más ha recaudado ha sido *Whatsapp* del desarrollador *SCEE*.

- Para el año 2011 en España sólo ha habido un usuario que haya descargado aplicaciones.
- En el año 2011 los ingresos generados por usuarios en España han sido de 14,97€ (el *imei* 1 ha bajado 3 veces el juego *Gradius*, 2 en el *script* de carga de datos y otra vez más en las pruebas de ABM y descargas a 4,99€ la descarga)
- En el año 2011 en España sólo ha habido una aplicación descargada al menos una vez (*Gradius*).

6. ANÁLISIS DE RIESGOS Y PLAN DE CONTINGENCIAS.

El riesgo en un proyecto software se debe gestionar mediante el análisis de los riesgos para lo cual primero hay que identificar los posibles factores de riesgo. Los factores de riesgo se pueden encontrar en el ámbito del proyecto, en los requerimientos funcionales, en el diseño del sistema, etc.

Así si seguimos afinando, por ejemplo, en lo respectivo a los requerimientos, debemos preguntar si estos son estables, están claramente definidos, cual es la criticidad del sistema para el cliente, etc. En tanto en cuanto estos requerimientos estén definidos de una manera clara y unívoca estaremos minimizando los riesgos asociados.

Debido a la naturaleza académica del proyecto debemos poner especial atención a los riesgos derivados del personal técnico, ¿Son los técnicos suficientemente experimentados?, ¿dominan las herramientas?, ¿La disponibilidad horaria es suficiente? A esto se le suman los imponderables debidos a problemas de disponibilidad debido a enfermedades o a incompatibilidades. De acuerdo con estos últimos podemos identificar riesgos concretos y posibles modos de atenuarlos

Riesgos debido a necesidades laborales. Para el caso de que exigencias laborales dificulten el desarrollo del proyecto se decidiría trabajar también los fines de semana (que se encuentran inicialmente fuera de la aplicación temporal) y hacer uso de las vacaciones que queden en este año natural. Durante la elaboración del proyecto hemos podido compaginar las obligaciones laborales con las académicas.

Riesgos por problemas de salud grave. En este caso se discutirían los requerimientos esenciales para llegar a un acuerdo con el cliente debido a la restricción temporal que provocan. No han sido relevantes en el desarrollo del proyecto.

Riesgos asociados a discrepancias en la interpretación de los requisitos con el cliente. En esta situación estableceríamos más reuniones que las inicialmente programadas de manera que los requisitos del cliente quedasen claros y el producto software se ajustase a las especificaciones iniciales. Entendemos que no se han producido y hemos llevado en todo momento una comunicación fluida con el cliente.

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

7. VALORACIÓN ECONÓMICA Y RECURSOS NECESARIOS.

7.1 Valoración económica.

Necesitamos tres perfiles diferenciados para el desarrollo del proyecto con tres costes asociados.

-Desarrollador	10€/hora
-Analista	12€/hora
-Administrador de BBDD	15 €/hora

Si llevamos esto a los perfiles necesarios en la planificación temporal tenemos el siguiente coste para el proyecto:

Admon. de BBDD 7 Horas - reacción y admón. de BBDD - 70 €

Analista 126 Horas – Planificación, análisis y diseño del producto software. – 1512€

Desarrollador 176 Horas – Labores de desarrollo y documentación. – 2640€

Coste final estimado: 4.222€

A estos costes hay que añadir los costes de licencias que entendemos serán asumidos por el cliente.

7.2. Recursos necesarios

Hardware

2 Equipos PC Compatible

Equipo principal: Acer Aspire 5720G con 2Gb de Ram y 160 Gb de HD – SO Windows Vista

Equipo de backup: HP mini 210 con 1Gb de Ram y 250 Gb de HD – SO Windows 7

Software

Sistema gestor de bases de datos: Oracle Express Edition 11g Release 2

Entorno de desarrollo: Oracle SQL Developer

Diseño de diagramas de Gantt y planificación temporal: OpenProj.

Diseño de esquemas UML y E-R: MagicDraw 16.8.

Redacción de documentos: MS Word 2007.

Creación de presentaciones: MS PowerPoint 2007.

8. CONCLUSIONES

El trabajo desarrollado en este proyecto ha sido muy útil para poner en práctica los diversos conocimientos adquiridos a lo largo de la titulación.

Para la elaboración del proyecto debemos ponernos en el lugar de todos los roles de un desarrollador de un proyecto software en su relación con el cliente final. Desde el inicio del proyecto hasta la entrega del producto final debemos mantener una comunicación fluida con el cliente para asegurarnos de que cumplimos los requisitos del cliente. Esta comunicación fluida, que en este TFC se traduce en el uso del foro y en las entregas parciales, es fundamental para minimizar cualquier desviación entre los productos esperados por el cliente y el desarrollo realizado.

Ponemos de manifiesto la importancia de realizar una buena planificación con plazos de entrega realistas. Esta buena planificación redundará en que el tiempo para el desarrollo será suficiente y no habrá partes del proyecto de menor calidad por estar excesivamente restringidas en plazo respecto a otras.

Centrándonos en el producto desarrollado entiendo que se han cumplido todos los requisitos del cliente en cuanto a las funcionalidades pedidas. Se ha realizado una adecuada planificación que ha redundado en que se ha dispuesto del tiempo suficiente para realizar todas las entregas parciales de manera correcta en tiempo y forma.

Se han utilizado procedimientos almacenados para el desarrollo de todas las Altas, Bajas y modificaciones así como para la gestión de descargas. Se han diseñado funciones para cumplir las consultas solicitadas por el cliente. Para el módulo de estadísticas se han desarrollado disparadores que actualizan tablas auxiliares, de manera que almacenemos los resultados que buscamos a medida que se realiza la actividad de descargas. Todos los procedimientos incluyen control de errores y gestión de excepciones. La actividad de ABM y descargas de la plataforma se puede revisar en el correspondiente fichero de Log. Todo el código desarrollado está documentado y comentado, funcionando de manera correcta.

El desarrollo del proyecto ha implicado que tengamos que adaptarnos a un SGBD determinado y de gran difusión como es Oracle y a un entorno de desarrollo nuevos como es el Oracle SQL Developer. Así también hemos hecho uso de herramientas de gestión de proyectos como Open Proj. También se pone de manifiesto la importancia de una buena documentación asociada al producto software ya que es fundamental que el cliente disponga de la mejor documentación tanto para aprovechar las funcionalidades de la aplicación como para las tareas de mantenimiento y mejoras del producto software. Así también hemos realizado una valoración de los riesgos y de los costes en términos económicos del proyecto tal y como se haría en un proyecto de la vida real.

Concluyendo podemos afirmar que en este TFC se han alcanzado los objetivos propuestos al inicio del mismo y ha servido para poner en práctica los conocimientos adquiridos a lo largo de titulación. El desarrollo de un proyecto equiparable a uno que nos podemos encontrar en el mundo real es de gran utilidad para que el estudiante tenga una aproximación al desempeño profesional de su titulación.

TFC – Memoria

Diseño e implementación de la base de datos de un sistema de descarga de aplicaciones para móviles inteligentes.

9. BIBLIOGRAFÍA

Materiales didácticos de las asignaturas de Bases de datos I, Bases de datos II y Estructura de la información

PRESSMAN, R.S., *Ingeniería del Software. Un enfoque práctico*. 7ª ed. 2010: Mc Graw Hill, Interamericana de España S.A.U. ISBN: 978-60-715-0314-5.

TEACHING SOFTGROUP. *Programación en Oracle11G SQL, SQL*PLUS Y PL/SQL* RA-MA, 2011. ISBN: 978-84-996-4118-8.

URMAN, Scott. *Oracle 9i: Programación PL/SQL*. Oracle Press Osborne, McGraw Hill, 2002. ISBN 978-84-481-3707-6.

ORACLE. *Oracle Database Express Edition Documentation*. Disponible en la web http://docs.oracle.com/cd/E17781_01/index.htm

ORACLE. *SQL, PL/SQL, and SQL*Plus Syntax and Examples*. Disponible en la web http://www.oracle.com/pls/db92/db92.sql_keywords

VV.AA. *Ayuda para Oracle*. Disponible en la web <http://ora.u440.com/>