Universidad de Alcalá
Escuela Politécnica Superior

GRADO EN SISTEMAS DE INFORMACIÓN

Trabajo Fin de Grado

DISEÑO E IMPLEMENTACIÓN DE HERRAMIENTA ON-LINE DE APRENDIZAJE SQL Y BBDD

ESCUELA POLITECNICA SUPERIOR

Autor: Javier Diaz Frias

Tutor/es: Iván González Diego

2016/2017

UNIVERSIDAD DE ALCALÁ

Escuela Politécnica Superior

Grado en Sistemas de Información

Trabajo Fin de Grado

DISEÑO E IMPLEMENTACIÓN DE HERRAMIENTA ON-LINE DE APRENDIZAJE SQL Y BBDD

Autor: Javier Diaz Frias

Tutor/es: Iván González Diego

TRIBUNAL:

Presidente:

Vocal 1º:

Vocal 2º:

FECHA:

Agradecimientos

Este trabajo ha sido posible gracias a mi tutor, Iván González, que sin él no podría presentar mi trabajo final de grado.

A mis padres y hermano, gracias por todos los ánimos que me habéis ofrecido en todo momento, preocupación e interés.

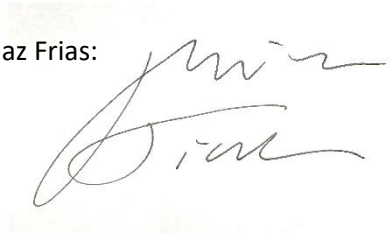
A mis amigos, gracias por esos consejos y apoyo que me habéis dado en todo momento, gracias por la ayuda y orientación.

En especial a mi pareja, a su insistencia y apoyo, gracias por creer en mí y que podía conseguirlo.

Gracias a todos, sin vosotros no lo hubiese logrado.

Gracias.

Javier Diaz Frias:



Índice

Agradecimientos 2
Tabla de ilustraciones 5
Resumen del proyecto 7
Resumen del Proyecto (inglés) 7
Palabras clave 7
Resumen extendido del proyecto 8
Acrónimos y abreviaturas 10
Memoria del Proyecto 11
1. Introducción 11
1.1. Justificación 11
1.2. Objetivos 11
1.3. Metodología seguida en el desarrollo del Trabajo Final de Grado 12
1.4. Riesgos 13
1.5. Cambios 13
1.5.1. Cambios en la planificación 13
1.5.2. Cambios en el modelo de datos 16
1.6. Fases del proyecto 17
1.7. Entorno de desarrollo 18
1.8. Productos obtenidos 19
2. Análisis 19
2.1. Requisitos 19
2.1.1. Requisitos no funcionales 19
2.1.2. Requisitos funcionales 20
2.2. Casos de uso 21
2.3. Especificación de los casos de uso 22
3. Diseño 27
3.1. Arquitectura física 27
3.2. Arquitectura de software 28
4. Implementación 33
¿Qué es APEX? 33
¿Por qué APEX? Hay otras herramientas… 34
4.1. Creación de la conexión BBDD – APEX 35
4.1.1. Creación de usuarios - esquemas en Base de Datos para APEX 36

4.1.1.1. Creación TABLESPACES 36
4.1.1.2. Creación de usuario - esquema 37
4.1.1.3. Agregar usuario – esquema en APEX 38
4.2. Realizar conexión BBDD – SQLdeveloper 39
4.3. Desarrollo de la aplicación 41
4.3.1. CURSO SQL 41
4.3.2 Mis TEST 42
4.3.3. Foro 43
4.3.4. Descargas 43
4.3.5. Gestión de usuarios 43
4.4. Carga de datos de la aplicación 44
4.5. Realizar pruebas y corrección de errores 44
5. Conclusiones 47
5.1. Conclusiones técnicas 47
5.2. Conclusiones personales 47
6. Líneas futuras 47
7. Presupuesto 48
Bibliografía 51
8. ANEXO 1: Documentación de la aplicación Capta SQL 53
8.1. Página de Logging 53
8.2. Página principal 54
8.3. Curso SQL 55
8.4. Mis test 56
8.4.1. Acceso a TEST 57
8.4.2. Resultados 59
8.5. Descargas 62
5.6. Foro 66
5.7. Gestión de usuarios 68
8. ANEXO 2: Instalación de Base de Datos 74
9. ANEXO 3: Instalación de APEX 76

Tabla de ilustraciones

Ilustración 1: Plan de trabajo 12
Ilustración 2: Planificación estimada inicialmente .....
Ilustración 3: Planificación real 14
Ilustración 4: Tabla de tiempos 15
Ilustración 5: Modelo inicial de datos 16
Ilustración 6: Modelo actualizado de datos 16
Ilustración 7: Casos de uso 21
Ilustración 8: Arquitectura física .....
Ilustración 9: Página principal aplicación CAPTA SQL 29
Ilustración 10: Imagen de los elementos de una página de APEX 30
Ilustración 11: Imagen de SQLdeveloper 31
Ilustración 12: Imagen de las tres capas a modo representativo .....
Ilustración 13: Proceso hasta llegar al directorio de instalación de APEX en consola 35
Ilustración 14: Comando habilitar puerto para APEX en consola 35
Ilustración 15: Habilitar listener APEX desde la consola 36
Ilustración 16: Crear tablespace en consola .....
Ilustración 17: Comprobación de datafile en consola 37
Ilustración 18: Crear usuario en consola .....
Ilustración 19: Comprobar usuario en consola .....
Ilustración 20: Pagina principal de administración de APEX .....
Ilustración 21: Crear Workspace-1 en APEX 38
Ilustración 22: Crear Workspace-2 en APEX 39
Ilustración 23: Crear Workspace-3 en APEX 39
Ilustración 24: Ventana conexiones SQLdeveloper 40
Ilustración 25: Configuración de conexión en SQLdeveloper 40
Ilustración 26: Comprobar instancia en consola 41
Ilustración 27: Comando establecer formato de fecha .....
Ilustración 28: PL/SQL de esquema de autenticación de ADMIN 46
Ilustración 29: Pagina de Logging de la Aplicación CAPTA SQL 53
Ilustración 30: PL/SQL de validación de usuarios 53
Ilustración 31: Página principal de aplicación CAPTA SQL con menú principal resaltado 54
Ilustración 32: Log-out y cambiar contraseña de la aplicación CAPTA SQL 54
Ilustración 33: Menú de organización de teoría de la aplicación CAPTA SQL 55
Ilustración 34: Ejemplo de submenú de teoría de la aplicación CAPTA SQL 55
Ilustración 35: Teoría de crear tabla de la aplicación CAPTA SQL-1 55
Ilustración 36: Teoría de crear tabla de la aplicación CAPTA SQL-2 56
Ilustración 37: Página principal de TEST de la aplicación CAPTA SQL 57
Ilustración 38: Ejemplo de TEST de la aplicación CAPTA SQL 57
Ilustración 39: Consulta de enunciado de test de la aplicación CAPTA SQL 57
Ilustración 40: Consulta de respuestas de la aplicación de CAPTA SQL 58
Ilustración 41: Función "INSERT" de test de la aplicación CAPTA SQL 58
Ilustración 42: Función de "UPDATE" test de la aplicación CAPTA SQL 58
Ilustración 43: Test realizados de la aplicación CAPTA SQL 59
Ilustración 44: Consulta de los test realizados de la aplicación CAPTA SQL 59

Ilustración 45: Imagen muestra para acceder a resultado de un test de la aplicación CAPTA SQL 59
Ilustración 46: Resultado de TEST de la aplicación CAPTA SQL..... 60
Ilustración 47: PL/SQL de descarga informe de resultado de TEST de la aplicación CAPTA SQL....
Ilustración 48: Informe de resultado de TEST en Excel..... 62
Ilustración 49: Página principal con los archivos compartidos de la aplicación CAPTA SQL..... 63
Ilustración 50: Consulta que muestra los archivos compartidos de la aplicación CAPTA SQL.... 63
Ilustración 51: Página para subir archivo de la aplicación CAPTA SQL..... 63
Ilustración 52: Tabla de archivos del modelo de datos empleado en la aplicación CAPTA SQL. 64
Ilustración 53: Opciones para la gestión de archivos de la aplicación CAPTA SQL 64
Ilustración 54: Procedimiento para modificar archivo de la aplicación CAPTA SQL-1.....
Ilustración 55: Procedimiento para modificar archivo de la aplicación CAPTA SQL-2.....
Ilustración 56: Procedimiento para borrar archivo de la aplicación CAPTA SQL-1 .....
Ilustración 57: Procedimiento para borrar archivo de la aplicación CAPTA SQL-2 .....
Ilustración 58: Página con todos los comentarios realizados de la aplicación CAPTA SQL..... 66
Ilustración 59: Consulta para mostrar los comentarios realizados de la aplicación CAPTA SQL 66
Ilustración 60: Página principal para crear tema nuevo de la aplicación CAPTA SQL..... 66
Ilustración 61: Tabla comentarios del modelo de datos..... 67
Ilustración 62: Procedimiento para contestar un comentario de la aplicación CAPTA SQL-1 .....
Ilustración 63: Procedimiento para contestar un comentario de la aplicación CAPTA SQL-2 67
Ilustración 64: Hilo de comentarios de un tema de la aplicación CAPTA SQL 68
Ilustración 65: Procedimiento para borrar comentario de la aplicación CAPTA SQL .....
Ilustración 66: "DELETE" para borrar comentario de la aplicación CAPTA SQL 68
Ilustración 67: Página principal de usuarios de la aplicación CAPTA SQL 69
Ilustración 68: Consulta que muestra los usuarios de la aplicación CAPTA SQL..... 69
Ilustración 69: Pagina para crear nuevo usuario de la aplicación CAPTA SQL 69
Ilustración 70: "INSERT" para crear nuevo usuario de la aplicación CAPTA SQL 70
Ilustración 71: Procedimiento para modificar un usuario de la aplicación CAPTA SQL-1..... 70
Ilustración 72: Procedimiento para modificar un usuario de la aplicación CAPTA SQL-2..... 70
Ilustración 73: "UPDATE" para modificar usuario de la aplicación CAPTA SQL 71
Ilustración 74: Opciones de estado de un usuario de la aplicación CAPTA SQL 71
Ilustración 75: Muestra de contraseña encriptada de la aplicación CAPTA SQL 72
Ilustración 76: Paso de valores al procedimiento en BBDD para cambiar contraseña a usuario de la aplicación CAPTA SQL. 72
Ilustración 77: Procedimiento para cambiar contraseña a usuario de la aplicación CAPTA SQL 72
Ilustración 78: Muestra de contraseña encriptada de un usuario de la aplicación CAPTA SQL . 72
Ilustración 79: Función que realiza encriptación de la contraseña de un usuario de la aplicación CAPTA SQL..... 73
Ilustración 80: Página de descargas de ORACLE .....
Ilustración 81: Selección de Base de Datos en el instalador 74
Ilustración 82: Página de Contraseñas de Instalación de la Base de Datos 75
Ilustración 83: Finalizar instalador de Base de Datos 75
Ilustración 84: Ruta de instalación de APEX en Windows..... 76
Ilustración 85: Conexión a la Base de Datos desde Windows.....
Ilustración 86: Sentencia instalación de APEX 76

Resumen del proyecto

El proyecto consiste en la creación de una aplicación web con la cual será posible adquirir conocimientos sobre Base de Datos y SQL, desde un nivel muy bajo hasta niveles de mayor complejidad.

Según se establece en el plan de trabajo, el proyecto evoluciona a lo largo de las distintas fases que se recogen en el plan, en primer lugar, se produce un análisis de las necesidades que plantea el proyecto y qué requisitos ha de cumplir para que en la posterior fase de implantación el desarrollo del software cumpla los requisitos mínimos y deseables que se buscan para obtener un resultado correcto.

La aplicación está construida en formato web, utilizando APEX de ORACLE® y los datos están alojados en la base de datos ORACLE DATABASEg 11 XE controlada por SQLdeveloper®.

Resumen del Proyecto (inglés)

The Project consist of how to create a web application thus could be possible learn Databases knowledge and its language, SQL, starting on low level and advance to more complex levels.

Following the workplan, the project evolves along a different phases. The first, is a project needs analysis and what requirements must cover on next project phase because the developer phase must cover all the requirements.

The application is developed on web format, it uses ORACLE's APEX and its data's is in a ORACLE's DATABASEg 11XE controlled by SQLdeveloper.

Palabras clave

- APEX
- BBDD
- SQL
- ORACLE
- ORACLE DATABASE 11G XE

Resumen extendido del proyecto

El presente trabajo se centra en el desarrollo de una aplicación que tiene como fin el estudio y comprensión del lenguaje de las Bases de Datos, el SQL.

¿Por qué SQL?

Se propone una aplicación en SQL por dos razones, el proceso de adaptación laboral está ubicado en el área de las Bases de Datos, donde el SQL está presente en todo momento y la segunda razón es porque he conocido que las herramientas que se van a utilizar a lo largo de este proyecto son muy desconocidas para muchísimos usuarios y merecen la pena que se den a conocer por la potencia que ofrece y todas las posibilidades que muestra.

Desarrollo del proyecto

El desarrollo del proyecto ha sido guiado en todo momento y ha estado marcado por un plan de trabajo definido desde el principio, que, a pesar de haber sufrido modificaciones, las fases se han seguido, fases como análisis de los requisitos o los casos de uso entre otros. Gracias a este plan de desarrollo se ha conseguido que el avance haya sido muy lineal y organizado en todo momento.

El proyecto comienza con la elaboración del anteproyecto y su depósito en la secretaria de la escuela, ese momento es el inicio donde se comienza a presentar ideas y a dar forma para posteriormente introducirlas en la aplicación final.

Estas ideas suponen realizar un análisis de viabilidad, pues es necesario conocer si el tiempo de implantación que van a tener merecen la pena. Esto ha ocurrido a lo largo del proyecto, pero gracias a un buen análisis realizado y a un correcto diseño la mayoría de las ideas que se han presentado han sido implantadas en la aplicación.

Puesto que a lo largo del proyecto se han incorporado nuevas funciones no tenidas en cuenta en la planificación inicial, esto supone que:

- La planificación cambie inmediatamente.
- El modelo de datos presentado inicialmente para las ideas iniciales sufra una transformación e incorpore nuevas tablas en la Base de Datos.
- El diseño de la aplicación cambia por completo, se produce una redistribución de ventanas.

La incorporación de los cambios no supone gran dificultad en el desarrollo de la aplicación, pues no es necesario eliminar ningún elemento, únicamente incorporar y sumar en la aplicación.

El análisis de las ideas posee su correspondiente documento, es por ello que se encuentra el documento de casos de uso donde se presentan todas las acciones que se ha de llevar a cabo para realizar las funciones y el documento de los requisitos funcionales y no funcionales el cual expone qué carencias hay que cumplir en el proyecto.

Gracias a estos documentos es posible conocer con mayor profundidad qué funciones se han descrito qué debe realizar el sistema y qué procedimiento es llevado a cabo para conseguir el resultado que se persigue.

Entrando en un punto de vista más de diseño, en cuanto a la arquitectura se refiere, encontramos que es fácil diferenciar los dos tipos de arquitectura, la arquitectura física donde ponemos en manifiesto qué elementos interactúan en el proyecto, entre los que encontramos el usuario, la Base de Datos, APEX y la aplicación; y el otro tipo de arquitectura que se refiere a el software.

La arquitectura software separa en niveles el software, entre los niveles se encuentran:

- La capa de presentación: la capa que ve el usuario, la interfaz.
- La capa de negocio: donde reside el programa que se utiliza, en nuestro caso es APEX, donde se aloja nuestra aplicación.
- La capa de datos: donde se registran los datos, en nuestro caso el sistema gestor de Base de Datos.

Cuando se conocen todos los elementos y están definidas las funciones de cada uno es el momento de comenzar con la implementación.

Gracias al proceso de análisis y diseño, el proceso de implementación de los elementos recogidos en estos procesos puede comenzar.

Dentro de la implementación de la aplicación se encuentran varias fases, entre las que se encuentran el desarrollo según el diseño realizado y el proceso de prueba y corrección de errores entre otras de las fases. Una vez ha sido probada la aplicación y se ha sometido a todo tipo de casos para probar su robustez y fiabilidad, el proceso de implementación puede darse por concluido. Pero con finalizar la implementación aún no ha finalizado el trabajo final de grado.

Como parte complementaria del trabajo, hay que poner a modo de tutorial, una documentación que explique la aplicación, que elementos recoge, que funciones realiza y de qué forma se realiza, para facilitar la comprensión de la aplicación y poder brindar la oportunidad a más desarrolladores a conocer cómo se ha realizado el desarrollo de la aplicación del trabajo final de grado.

Como conclusión hay que indicar que en este trabajo final de grado se ha puesto en manifiesto todo lo que se ha aprendido a lo largo del periodo académico. Se han puesto en práctica ciertos temas teóricos de distintas asignaturas realizadas en los años de estudio, gracias a las cuales han aportado gran valor a este trabajo final de grado.

Acrónimos y abreviaturas

- APEX: APlication Express.
- BBDD: Base de Datos.
- SQL: Structured Query Language
- INSERT: insertar
- DELETE: borrar.
- UPDATE: actualizar
- SELECT: selección
- Listener: proceso que recibe información de un cliente
- FK: foreign key, clave ajena
- PK: primary key, clave Primaria.
- PL/SQL: lenguaje procedimental SQL.

Memoria del Proyecto

1. Introducción

1.1. Justificación

A lo largo del grado y en los diferentes cursos que forman el mismo, el alumno va adquiriendo diferentes conocimientos, conceptos, herramientas, ideas etc. que debe plasmar en el proyecto fin de carrera; el cual es la parte más importante de todo ingeniero ya que en él se intenta acoplar todo lo aprendido a lo largo de esos cursos de la forma más práctica y ordenada con el fin de que una idea se convierta en producto.

Este producto va ser creado tomando como base el estudio a lo largo de los diferentes cursos, lo que permitirá realizar una planificación del mismo, así como un análisis y diseño para su posterior implementación e implantación tanto como a nivel universitario como empresarial o mismamente para desarrollar su uso en un hogar donde los conocimientos de ingeniería o informática sean básicos.

El producto está desarrollado en interfaz web mediante la herramienta APEX de ORACLE en combinación con SQLdeveloper y Oracle Database 11g XE. La elección de combinar dichas herramientas es debido a que Oracle ofrece la gama más completa e integrada de herramientas para el desarrollo de aplicaciones, desarrollo de base de datos e inteligencia de negocio, compatibles con cualquier enfoque de desarrollo, plataforma tecnológica o sistema operativo.

1.2. Objetivos

El objetivo principal es crear una herramienta que permita conocer, estudiar y poner en práctica el lenguaje de las BBDD, SQL.

En el aspecto educativo, se trata de plasmar todos los conocimientos adquiridos de forma teórica a lo largo de la carrera, así como los adquiridos de forma práctica durante mi estancia en mis practicas universitarias, donde pude acercarme más a dicho lenguaje y ampliar el conocimiento que ya poseía.

Es por eso, que esta herramienta está orientada a todo el público, pues no necesita tener una base específica para realizar consultas ya que realiza una explicación de todo el temario de SQL desde un nivel muy bajo, avanzando a medida que se avanza en el temario hasta un nivel más complejo con funciones de dificultad más elevada.

En el aspecto técnico, el objetivo del proyecto ha sido la perfecta utilización del lenguaje SQL y PL/SQL para crear e implementar funciones que permitan el funcionamiento correcto de la aplicación con el fin de obtener el objetivo principal. Dicho objetivo intentará divulgar el lenguaje SQL, lo que es, para que sirve, sus funciones… así como ponerlo en práctica de una forma sencilla.

1.3. Metodología seguida en el desarrollo del Trabajo Final de Grado

El método que se ha seguido para la creación del Trabajo Final de Grado ha sido un ciclo de vida en cascada.

Este modelo se caracteriza en que una fase no puede comenzar si no ha finalizado la anterior. En este proyecto, el ciclo de vida en cascada ciclo está compuesto por las siguientes etapas:

Ilustración 1: Plan de trabajo

- Plan de trabajo:
  Es un instrumento previo necesario para el desarrollo de un proyecto en un tiempo determinado.
  El plan de trabajo es un tipo de manual que sugiere la forma de utilizar eficientemente los recursos disponibles para planificar, ejecutar, implementar y supervisar el proyecto.
  El plan de trabajo lleva un orden lógico, donde un punto es requisito para el siguiente.

- Análisis y diseño:
  En estas fases en primer lugar se analizan las necesidades de los usuarios finales del software para cubrirlas, donde se obtiene el análisis de los requisitos.
  Posteriormente se organiza el sistema en elementos que puedan elaborarse por separado. Como resultado se obtienen los casos de uso, donde se especifica qué debe hacer cada una de las partes del sistema.

- Implementación:
  Esta fase corresponde con la implementación del código fuente, haciendo prototipos y distintas versiones, así como pruebas y corrección de errores.

- Memoria:
  Dentro del proyecto, esta fase consiste en redactar todo el proceso de desarrollo que ha tenido la aplicación, desde el primer instante que comienza el trabajo (anteproyecto) hasta la redacción de la memoria.

- Mantenimiento:
  Esta fase, aunque esta fuera del Trabajo Final de Grado, es una fase que requiere atención, pues el producto puede seguir evolucionando, incorporando nuevas funciones y herramientas que necesitan de personal de desarrollo para poder hacerlo.

1.4. Riesgos

Uno de los riesgos más importantes a la hora de utilizar las aplicaciones y programas de otros propietarios, como es el caso de APEX y SQLdeveloper, es que pueden existir restricciones dependiendo del fin que se vayan a dar a las aplicaciones finales.

Este riesgo debe ser considerado en la planificación inicial y en el producto final, pues un uso comercial de las aplicaciones de ORACLE debe de mostrar la licencia en todo momento. En nuestro caso, al tratarse de una aplicación con fin formativo, no necesitamos licencia de uso en ningún caso.

Otro riesgo a tratar está ligado al ejecutar los programas en un pc doméstico. Estos programas están diseñados para ser ejecutados en equipos totalmente optimizados para ese uso, por tanto, aunque en la experiencia no se han producido fallos ni complicaciones, no está exento de que ocurra, el mayor de los problemas que puede ocurrir es que el sistema falle y el trabajo creado se pierda.

¿Solución? En nuestro caso mantener copias regularmente como medida de contingencia.

1.5. Cambios

1.5.1. Cambios en la planificación

El proyecto sufrió desviaciones, en lo que a su planificación temporal se refiere, en el desarrollo de la aplicación.

La planificación que se estima en la entrega del anteproyecto es la siguiente:

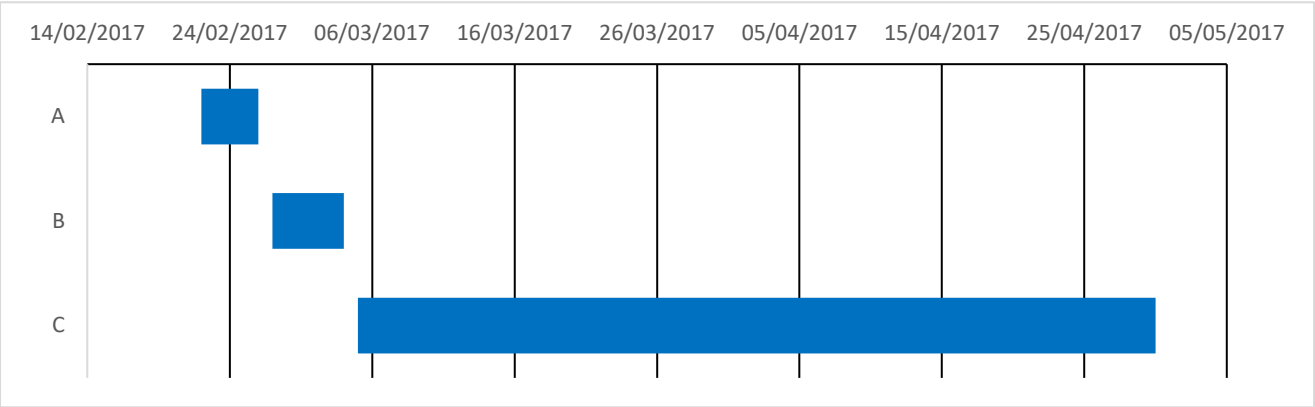


Ilustración 2 :planificación estimada inicialmente

A= Instalación del entorno y herramientas a utilizar. 4 días.
B= Creación del modelo de datos inicial. 5 días.
C= Desarrollo de la aplicación. 56 días.

Observamos los días que estimamos que durarían las fases del proyecto. La realidad cambia, pues hubo cambios en las fases lo que afecta directamente a la planificación:

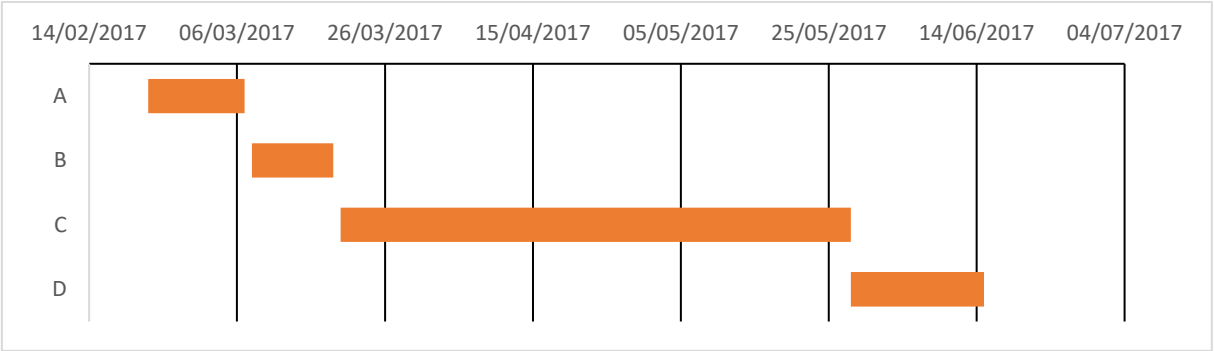


Ilustración 3 : Planificación real

Todas las fases han incrementado su tiempo en comparación a lo planificado, observemos las diferencias por fases:

- **Fase A. Plan de trabajo:** se corresponde con el momento inicial del proyecto. En esta fase se analiza y estudia el proyecto, se redacta el anteproyecto: "DISEÑO E IMPLEMENTACIÓN DE HERRAMIENTA ON-LINE DE APRENDIZAJE SQL Y BBDD" y se instala las herramientas necesarias: APEX, ORACLE DATABASE 11G XE, SQL developer. En comparación con la estimación realizada en el anteproyecto se observa gran cantidad de cambios, pues se estima que solo se hace la instalación de las herramientas y entorno

necesario y en realidad se hizo lo anterior descrito, lo que supuso un incremento en los días de la primera fase.

- **Fase B. Análisis y diseño:** esta fase comprende la adquisición de ideas, así como asentarlas y presentar un diseño inicial. Totalmente distinto a la estimación inicial. En dicha estimación esta fase únicamente se describió la realización del modelo de datos, sin embargo, realmente esta fase incorpora además del diseño del modelo de datos, un diseño visual de la distribución de la aplicación y un estudio de las tecnologías escogidas para mejorar el entendimiento y su comprensión.

  Debido a que se incorporan muchos cambios, el tiempo real aumenta bastante en comparación a el tiempo estimado.

- **Fase C. Implementación.** Tanto en la estimación, como en el proyecto real, esta fase posee los mismos efectos en el proyecto, sin embargo, en la estimación únicamente se estima el desarrollo de la aplicación y en el desarrollo real se incorpora la carga de datos, realización de pruebas y la elaboración de la documentación. Nuevos elementos que suponen un incremento del tiempo a comparación de la estimación inicial.

- **Fase D. Memoria:** última fase del proyecto la cual no se tuvo en cuenta en la elaboración del anteproyecto y supone un incremento de tiempo. Dicha fase corresponde con la realización de la memoria y presentación visual.

Como se puede observar, se posee una fase más que en la estimación inicial. Como consecuencia directa esto supone un incremento del tiempo de realización del proyecto y a esto se le suma la incorporación de más sub-fases.

Por tanto, la distribución del tiempo queda de la siguiente forma:

	TIEMPO ESTIMADO	TIEMPO REAL INVERTIDO	DIFERENCIA REAL-ESTIMADO
FASE A	4 días	13 días	9 días
FASE B	5 días	11 días	6 días
FASE C	56 días	69 días	10 días
FASE D	0 días	18 días	-18 días

Ilustración 4: Tabla de tiempos

Al no tener una “Fase D” en la estimación real, sale un total de 18 días más en esa fase.

1.5.2. Cambios en el modelo de datos

En la fase de diseño de la aplicación (Fase B) inicialmente se crea un modelo de datos que cubre las necesidades del momento. Ese modelo de datos es:

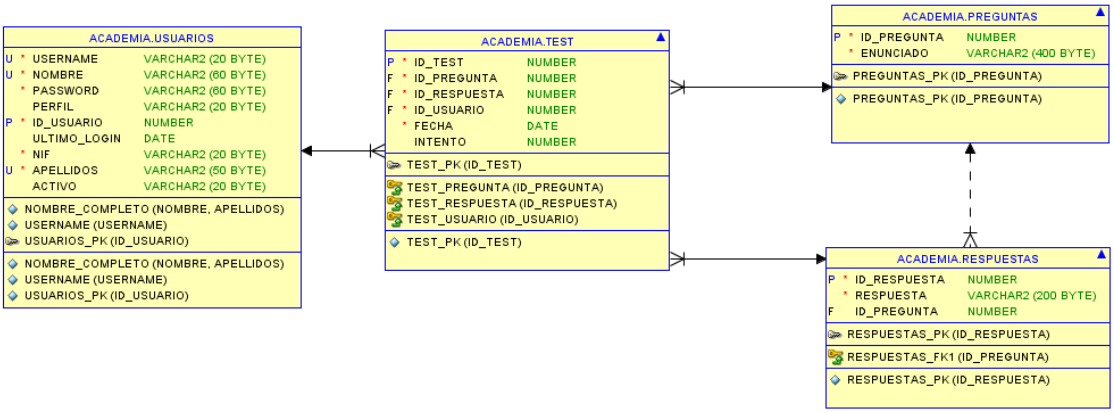


Ilustración 5: Modelo inicial de datos

Como se puede ver, en el diseño inicial se contaba con las tablas anteriores:

- Usuarios: tabla donde insertaba los usuarios de la aplicación.
- Preguntas: tabla que almacena los enunciados de las preguntas.
- Respuestas: tabla que almacena las respuestas de las preguntas. Dispone de Foreign Key a la tabla preguntas.
- Test: tabla que relaciona la pregunta y respuesta con el usuario que la realiza.

Sin embargo, según avanza el desarrollo del proyecto, se incorporaron nuevas funcionalidades y mejoras en la aplicación, lo que supuso que el modelo de datos evolucionase al siguiente:

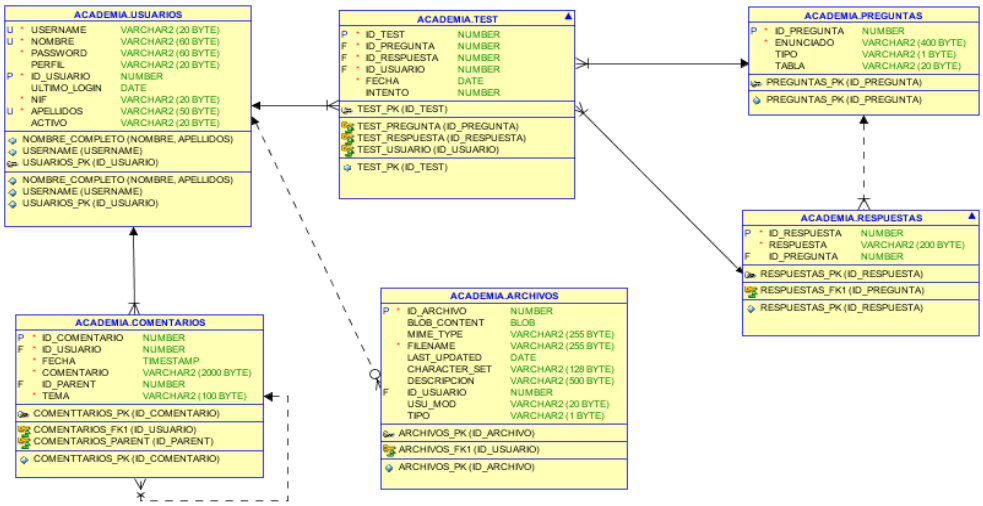


Ilustración 6: Modelo actualizado de datos

Como se puede observar, en comparación con el modelo que se creó al inicio, lo que más resalta es la incorporación de dos nuevas tablas, “Archivos” y “Comentarios”, aunque también ha sufrido modificaciones “Preguntas”. Por tanto, el modelo de datos actualizado se compone de:

- Usuarios: tabla donde insertaba los usuarios de la aplicación.
- Preguntas: tabla que almacena los enunciados de las preguntas. Incluido el campo de “TIPO” con el que se diferencia la tipología de las preguntas y crear divisiones dependiendo de su tipo y también el campo “TABLA”, éste campo permite agrupar por tablas los enunciados de las preguntas.
- Respuestas: tabla que almacena las respuestas de las preguntas. Dispone de Foreign Key a la tabla preguntas.
- Test: tabla que relaciona la pregunta y respuesta con el usuario que la realiza.
- Comentarios: tabla que almacena los comentarios proporcionados por los usuarios con un tema.
- Archivos: tabla que almacena los archivos que comparten los usuarios

El modelo de datos ha crecido según la necesidad de la aplicación y por la incorporación de nuevas características.

1.6. Fases del proyecto

El proyecto se ha dividido en cuatro fases tal y como se explica en el punto de la metodología llevada a cabo en el proyecto. Adicionalmente, las fases pueden dividirse en pequeñas fases que detallan más completamente el desarrollo.

1.6.1. Fase A – Plan de trabajo

Esta fase se corresponde con el periodo inicial del proyecto. En esta fase se han realizado las siguientes actividades:

- Analizar y estudiar el proyecto a elegir.
- Elección del proyecto a elegir.
- Redacción del Anteproyecto: “DISEÑO E IMPLEMENTACIÓN DE HERRAMIENTA ON-LINE DE APRENDIZAJE SQL Y BBDD “.
- Instalación de las herramientas necesarias: APEX, ORACLE DATABASE 11G XE, SQL developer.
- Realizar plan de trabajo.

1.6.2. Fase B – Análisis y diseño

Esta fase se corresponde con el periodo de recolección y diseño de ideas. Las actividades que se realizan son las siguientes:

- Elaboración del modelo de datos.
- Diseñar la distribución visual de la aplicación.
- Profundizar en el estudio de las tecnologías escogidas.

1.6.3. Fase C – Implementación

Esta fase se corresponde con la creación de la aplicación en sí, se crean páginas, funciones y procedimientos. Las actividades son las siguientes:

- Creación de conexión BBDD – APEX.
- Desarrollar la aplicación según el diseño previo incorporando las funciones y procedimientos.
- Carga de datos de la aplicación.
- Realizar pruebas y corrección de errores.
- Realizar documentación.
  Esta actividad comprende las distintas actividades a entregar, instalación de herramientas, manual de uso, etc.

1.6.4. Fase D – Memoria y presentación del proyecto

Esta fase corresponde con la finalización del proyecto. Las actividades que se realizan son las siguientes:

- Confeccionar la memoria del proyecto.
  Se ha realizado una síntesis de todos los datos del proyecto.
- Realizar presentación.
  Se ha creado la presentación con los puntos clave del proyecto para una posterior explicación. Se incluirá una demostración de cómo se utiliza la aplicación que se presenta.

1.7. Entorno de desarrollo

Para el desarrollo del proyecto final de grado se ha utilizado un equipo con las siguientes características:

- Procesador Intel Core i7 4720HQ.
- 12 GB de memoria RAM.
- Disco duro de 1 TB.
- Sistema operativo Windows 10 Home 64 bits.

A continuación, se detalla el software necesario para el desarrollo del Proyecto:

- Oracle Database 11G XE 64 bits.
- SQL developer.
- APEX V5.
- Microsoft Office 365 Pro Plus.
- Google Chrome.

1.8. Productos obtenidos

Durante el desarrollo del proyecto, se han generado los siguientes productos:

- Plan de trabajo
- Manual de usuario
- Manual de instalación.
- Memoria del proyecto.
- Presentación visual del proyecto.

2. Análisis

En este apartado detalla el análisis realizado sobre todas las necesidades y requisitos del entorno en este proyecto.

2.1. Requisitos

Existen dos tipos de requisitos:

- **Requisitos funcionales:** son aquellos que consisten en una característica requerida del sistema que expresa una capacidad del misma -una funcionalidad; generalmente expresada en una declaración verbal.
- **Requisitos no funcionales:** son aquellos que consisten en una característica requerida del sistema, del proceso de desarrollo, del servicio prestado o de cualquier otro aspecto del desarrollo, que señala una restricción del mismo.

2.1.1. Requisitos no funcionales

- **Sistema operativo:** ORACLE database 11G XE es compatible con distribuciones Linux y Windows, siendo incompatible con MacOS.
- **APEX V5:** el entorno de desarrollo de la aplicación está desarrollado en PL/SQL y SQL.
- **Requisito de espacio:** la aplicación precisa de espacio en disco duro para almacenar datos.
- **Usabilidad:** la aplicación debe ser fácil de utilizar.
- **Seguridad en los datos:** la aplicación encripta los datos de sesión de los usuarios en la BBDD.

2.1.2. Requisitos funcionales

- **Portabilidad:** la aplicación puede ser utilizada en PC y en interfaz móvil.
- **Gestión de identidad:** la aplicación identifica el perfil de cada usuario, permitiendo el acceso únicamente a sus datos. Cada usuario es único.
- **Seguridad:** la contraseña de cada usuario se cifra una vez ha cambiado la contraseña por defecto.
- **Gestión de los test:** la aplicación muestra las preguntas realizadas y las respuestas contestadas y correctas de los test.
- **Test ilimitados:** es posible realizar un número ilimitado de test diariamente.
- **Gestión de los archivos:** La aplicación solo permite editar comentario y archivo a su propietario.
- **Gestión de los comentarios:** la aplicación solo permite borrar un tema de conversación a su creador, al igual que las respuestas a un tema.
- **Gestión de la BBDD:** el acceso a la base de datos está restringido únicamente al desarrollador de la aplicación.

2.2. Casos de uso

Un caso de uso es una descripción de los pasos que deben realizarse para llevar a cabo un proceso. A continuación, se muestran los casos de uso del sistema:

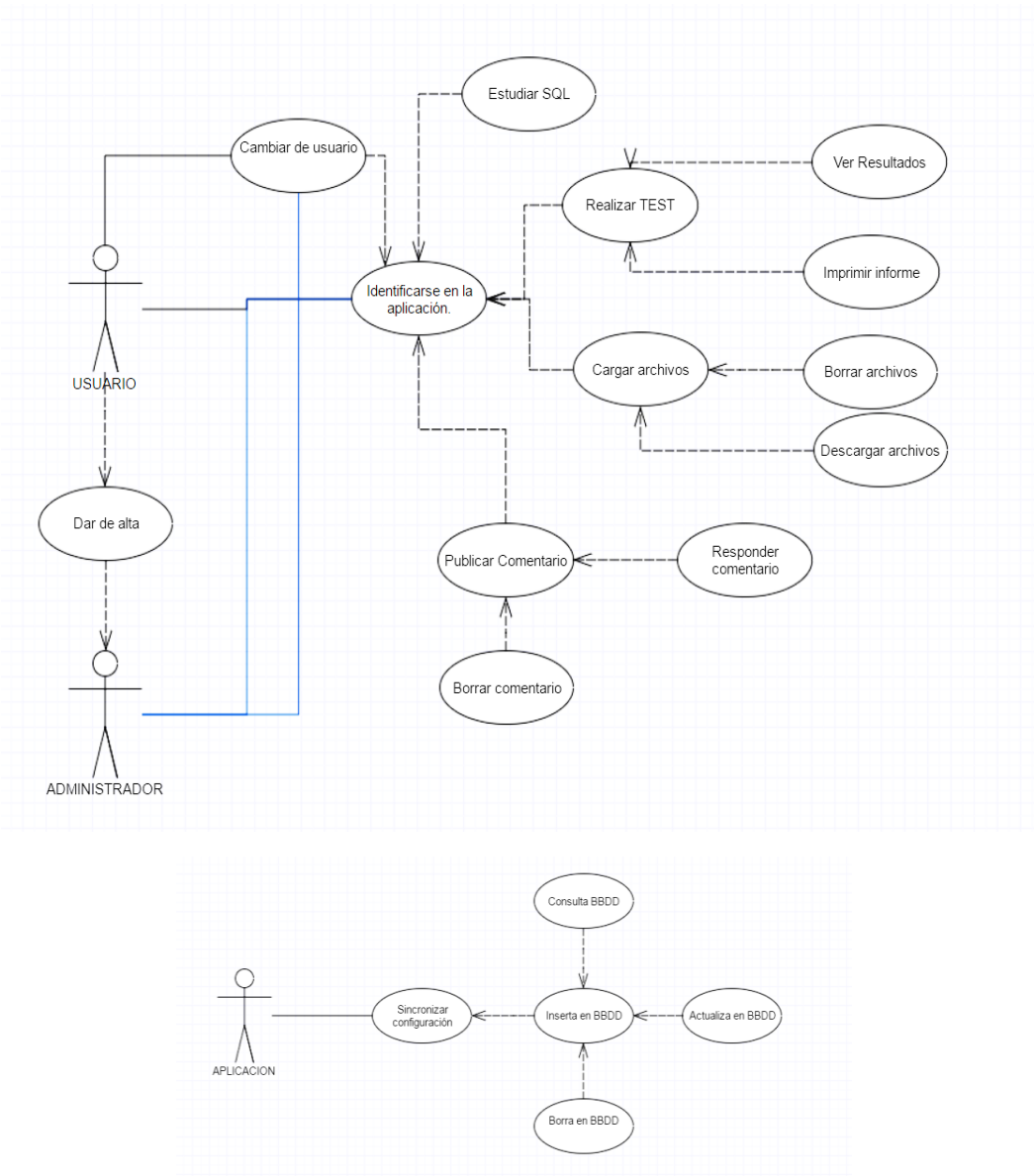


Ilustración 7: Casos de uso

2.3. Especificación de los casos de uso

A continuación, se detallan el conjunto de casos de uso que intervienen en la aplicación.

2.3.1 Caso de uso Identificación en la aplicación

Resumen de la funcionalidad: permite identificar el usuario que inicia sesión en la aplicación

Actores: Usuario, Administrador

Casos de uso relacionado: Caso de uso **Dar de alta.**

Precondición: Conocer los datos de inicio de sesión. Estar dado de alta(usuario)

Postcondición: se inicia sesión en la aplicación.

Proceso normal principal:

1. El usuario inserta sus credenciales en los correspondientes campos
2. El sistema valida las credenciales insertadas
3. El usuario obtiene acceso a la aplicación

Alternativas de proceso y excepciones:

2. El sistema no valida por error de las credenciales.

2.3.2 Caso de uso Dar de alta

Resumen de la funcionalidad: el administrador incorpora nuevos usuarios.

Actores: Administrador.

Casos de uso relacionado: Caso de uso **Identificación en la aplicación**

Precondición: debe existir un administrador.

Postcondición: un nuevo usuario ha sido creado.

Proceso normal principal:

1. El administrador elige dar de alta un nuevo usuario
2. El sistema solicita los datos necesarios.
3. El administrador inserta datos.
4. El sistema crea el usuario

Alternativas de proceso y excepciones:

4. El sistema detecta el usuario ya creado

2.3.3 Caso de uso cambiar de usuario

Resumen de la funcionalidad: permite cambiar la cuenta que tiene iniciada sesión en la aplicación.

Actores: Usuario, Administrador

Casos de uso relacionado: Caso de uso **Identificación en la aplicación.** Caso de uso **Dar de alta.**

Precondición: Conocer datos de inicio de sesión. Estar dado de alta(usuario).

Postcondición: se inicia nueva sesión.

Proceso normal principal:

1. El usuario selecciona cambiar de usuario.
2. El sistema sincroniza la configuración.
3. Se inicia nueva sesión.

Alternativas de proceso y excepciones:

3. El sistema rechaza la conexión

2.3.4 Caso de uso Estudiar SQL

Resumen de la funcionalidad: el usuario tiene disponible el temario de estudio para formarse en SQL y PL/SQL.

Actores: Usuario, Administrador.

Casos de uso relacionado: Caso de uso **Identificación en la aplicación**

Precondición: poder identificarse en la aplicación.

Postcondición:

Proceso normal principal:

1. El usuario selecciona el acceso a estudio de SQL.
2. El sistema muestra el contenido

Alternativas de proceso y excepciones:

2.3.5. Caso de uso Realizar TEST

Resumen de la funcionalidad: permite acceder a los test para comprobar el nivel de conocimientos

Actores: Usuario, Administrador

Casos de uso relacionado: Caso de uso **Identificación en la aplicación.**

Precondición: poder identificarse en la aplicación.

Postcondición: genera un test del usuario.

Proceso normal principal:

1. El usuario selecciona acceso a test.
2. El sistema muestra las preguntas.
3. El usuario responde las preguntas.
4. Finaliza el test.

Alternativas de proceso y excepciones:

4. El usuario puede finalizar el test antes de completarlo.

3.5.6. Caso de uso Ver resultados

Resumen de la funcionalidad: permite obtener resultados de los test realizados.

Actores: Usuario, administrador.

Casos de uso relacionado: Caso de uso **Realizar TEST.**

Precondición: el usuario que inicia sesión debe haber hecho al menos un test y completarlo.

Postcondición: se crea un informe con los resultados del test seleccionado.

Proceso normal principal:

1. El usuario selecciona ver resultados.
2. El sistema muestra los test realizados.
3. El usuario selecciona el test para ver el resultado.
4. El sistema recupera el test.
5. El usuario puede ver el test y los resultados.

Alternativas de proceso y excepciones:

2. El usuario no ha hecho test, por tanto, no ve ningún resultado.

2.3.7. Caso de uso Imprimir resultados.

Resumen de la funcionalidad: permite al usuario realizar un informe en formato Excel de los resultados de su test.

Actores: Usuario, Administrador.

Casos de uso relacionado: Caso de uso **Realizar TEST.**

Precondición: debe tener realizado al menos un test.

Postcondición: genera un archivo Excel con los datos del test seleccionado.

Proceso normal principal:

1. El usuario selecciona un test para imprimir.
2. El sistema genera el archivo Excel con los datos del test.

Alternativas de proceso y excepciones:

1. El usuario no tiene test realizados para seleccionar.

2.3.8. Caso de uso Cargar archivos.

Resumen de la funcionalidad: permite cargar archivos al espacio de la aplicación.

Actores: usuario, administrador.

Casos de uso relacionado: Caso de uso **Identificación en la aplicación.**

Precondición: estar logueado.

Postcondición: se ha cargado un archivo en el espacio de la aplicación.

Proceso normal principal:

1. El usuario selecciona subir archivo.
2. El sistema solicita el archivo e información.
3. El usuario inserta información.
4. El sistema verifica la duplicidad del archivo.

Alternativas de proceso y excepciones:

4. El sistema detecta archivos duplicados, rechaza subir el archivo.

2.3.9. Caso de uso Borrar archivos.

Resumen de la funcionalidad: permite borrar un archivo alojado en la aplicación.

Actores: Usuario, Administrador.

Casos de uso relacionado: Caso de uso **Cargar archivos.**

Precondición: debe existir un archivo cargado.

Postcondición: el archivo se ha borrado de la aplicación.

Proceso normal principal:

1. El usuario pulsa borrar en el archivo seleccionado.
2. El sistema borra el archivo.

Alternativas de proceso y excepciones:

1. El usuario no puede borrar el archivo, no es de su propiedad.

2.3.10. Caso de uso Descargar archivos.

Resumen de la funcionalidad: permite descargar un archivo alojado en la aplicación.

Actores: Usuario, administrador.

Casos de uso relacionado: Caso de uso **Cargar archivos.**

Precondición: debe existir un archivo alojado en la aplicación.

Postcondición: la aplicación ha generado un archivo que se descarga.

Proceso normal principal:

1. El usuario selecciona el archivo a descargar.
2. El sistema descarga el archivo.

Alternativas de proceso y excepciones:

1. El sistema debe tener archivos cargados.

2.3.11. Caso de uso Publicar comentario.

Resumen de la funcionalidad: permite publicar un comentario en la aplicación.

Actores: usuario, administrador.

Casos de uso relacionado: Caso de uso **Identificación en la aplicación.**

Precondición:

Postcondición: genera un comentario en la aplicación.

Proceso normal principal:

1. El usuario rellena los campos de comentario.
2. El sistema publica el comentario

Alternativas de proceso y excepciones:

2.3.12. Caso de uso Responder comentario.

Resumen de la funcionalidad: permite responder un comentario.

Actores: usuario, Administrador.

Casos de uso relacionado: caso de uso **Publicar comentario.**

Precondición: debe existir un comentario previo.

Postcondición: genera una respuesta a un comentario.

Proceso normal principal:

1. El usuario responde el comentario seleccionado.
2. El sistema muestra la respuesta al comentario.

Alternativas de proceso y excepciones:

1. No existen comentarios que responder.

2.3.13. Caso de uso Borrar comentario.

Resumen de la funcionalidad: permite borrar un comentario publicado en la aplicación.

Actores: usuario, administrador.

Casos de uso relacionado: caso de uso **Publicar comentario.**

Precondición: debe existir un comentario previo.

Postcondición: elimina el comentario del sistema.

Proceso normal principal:

1. El usuario elimina el comentario seleccionado.
2. El sistema elimina los datos del comentario.

Alternativas de proceso y excepciones:

4. El usuario no puede eliminar el comentario seleccionado, no es su propietario.

3. Diseño

A continuación, se verá cómo está diseñada la aplicación desde distintos puntos de vista, en lo que su arquitectura se refiere:

3.1. Arquitectura física

La arquitectura física expresa los componentes físicos del sistema (el cliente, servidor, Base de Datos, etc.) que participan en el sistema, así como la relación entre ellos.

La especificación de la arquitectura física consta de uno o más diagramas.

El producto es creado gracias a APEX, un componente de las bases de datos de ORACLE, por lo que cada base datos puede tener disponible este software instalado.

Para tener acceso a una base de datos ORACLE, no es necesario tener un servidor, un pc domestico es válido para tener la base de datos y así poder montar y ejecutar la aplicación APEX.

La arquitectura física del proyecto está compuesta por:

- Base de datos.
- APEX(Cliente).
- Usuario/s.
- Aplicación.

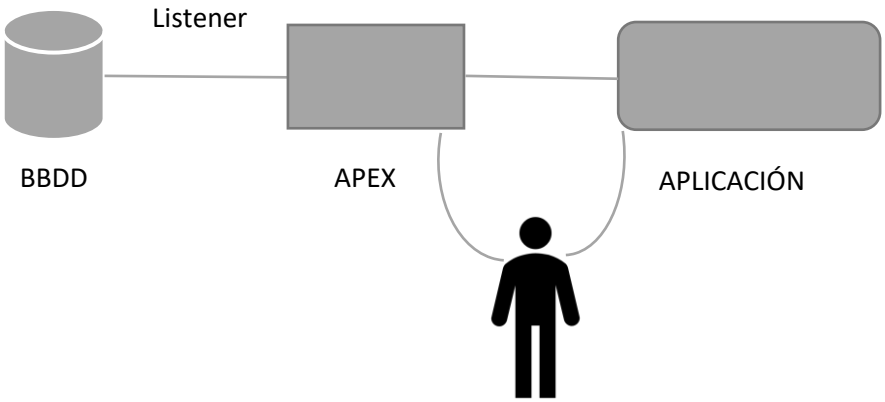


Ilustración 8: Arquitectura física

La información parte de la base de datos, la conexión está controlada hacia la aplicación APEX a través del "Listener"- este archivo de configuración indica el puerto e IP de conexión a la base de datos, es el canal por el cual APEX se comunica con la base de datos. APEX posee contacto directo con la aplicación, la cual carga datos, configuraciones, etc. Entre APEX y la aplicación aparece el usuario, que interviene en la comunicación ente ambos elementos para realizar una carga de datos de usuario.

3.2. Arquitectura de software

La arquitectura software tiene como objetivo separar las partes de un sistema software para tener de forma más sencilla un mejor mantenimiento e implementación sin tener que intervenir en la capa física.

La ventaja es que el desarrollo es posible llevarlo a través de varios niveles separados, donde cada nivel posee una misión única y así evitar problemas en el diseño de la aplicación.

El modelo que se ha aplicado en nuestro diseño es el más común, el diseño de tres niveles o diseño en tres capas:

- Capa de presentación
- Capa de negocio
- Capa de datos

3.2.1. Capa de presentación

Esta capa es la que ve el usuario y con la que hay interacción. Comúnmente se conoce como interfaz gráfica. Esta capa se comunica únicamente con la capa de negocio.

En cuanto a nuestra aplicación, la capa de presentación es la interfaz de la aplicación. Es una interfaz limpia, ordenada, sencilla, intuitiva y fácil de usar:

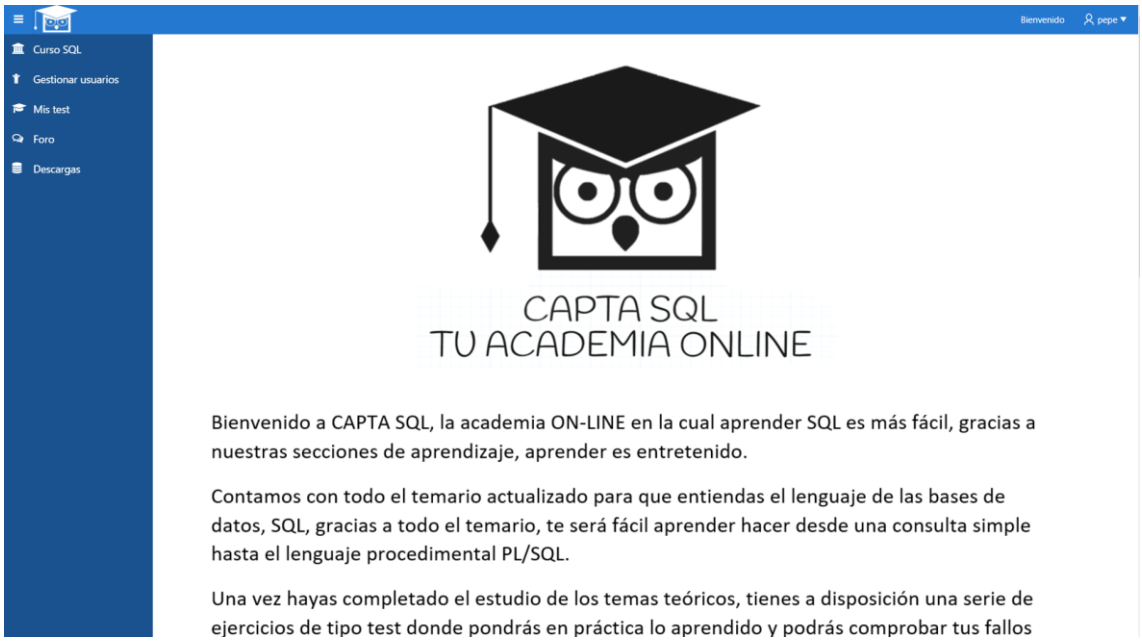


Ilustración 9: Página principal aplicación CAPTA SQL

En la anterior imagen se puede ver la página de inicio con el menú principal de la aplicación. Es una interfaz limpia que muestra toda la información. Posee un menú sencillo e intuitivo donde una simple palabra detalla la opción a la que se accede.

3.2.2. Capa de negocio

Es donde residen los programas que se ejecutan, se reciben peticiones del usuario y se envían las respuestas tras el proceso. El nombre de capa de negocio es así porque en esta capa residen el conjunto de reglas que deben cumplirse. Esta capa se sitúa como intermediaria entre la capa de presentación y la capa de datos. La capa de presentación envía solicitudes a la capa de negocio y ésta solicita los datos a la capa de datos.

En nuestro caso, la capa de negocio es APEX -recordemos que APEX es una aplicación que se ejecuta en las bases de datos ORACLE y que permite crear aplicaciones; La interfaz que es intermediaria entre la aplicación y la base de datos, es el canal por el cual la información fluye entre la BBDD y la aplicación final y es la que pone las reglas para una perfecta ejecución.

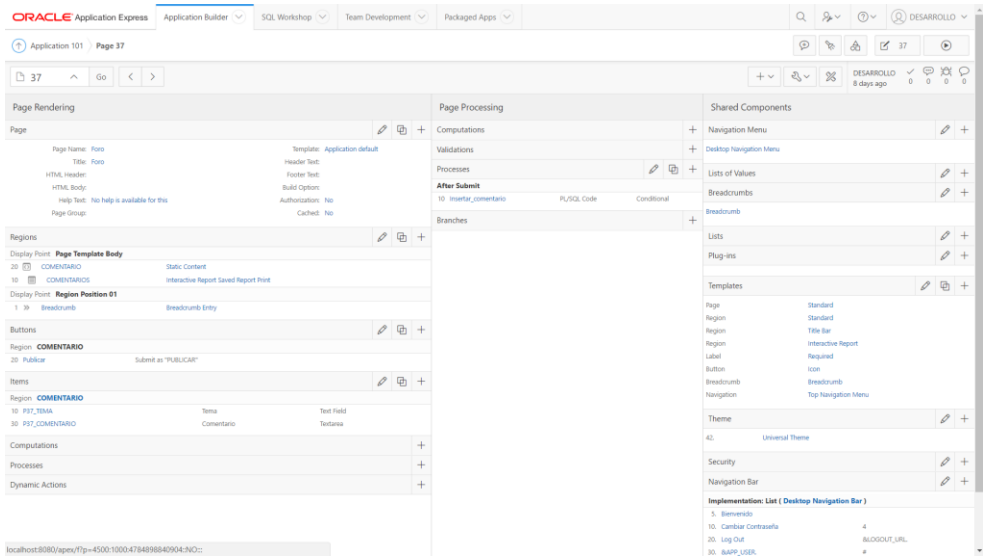


Ilustración 10: Imagen de los elementos de una página de APEX

La anterior imagen es una muestra de APEX, la aplicación que controla el flujo entre la BBDD y la aplicación final.

Se puede observar que es el espacio donde se implementan las funciones, operaciones, validaciones, etc. Estas operaciones sirven para mostrar datos en la capa de presentación que ésta misma capa solicita o bien para enviar datos desde la capa presentación. Todo este flujo es controlado por la capa de negocio.

3.2.3. Capa de datos.

La capa de datos es donde residen los datos y es la encargada de acceder a ellos mismos. Está formado por uno o más SGBD-sistemas gestores de bases de datos, que realizan todo el almacenamiento de datos, reciben solicitudes e almacenar nuevos datos y se solicitan datos desde la capa de negocio.

En nuestro caso, la capa de datos se corresponde con la Base de datos y el SGBD utilizado, en nuestro caso SQLdeveloper.

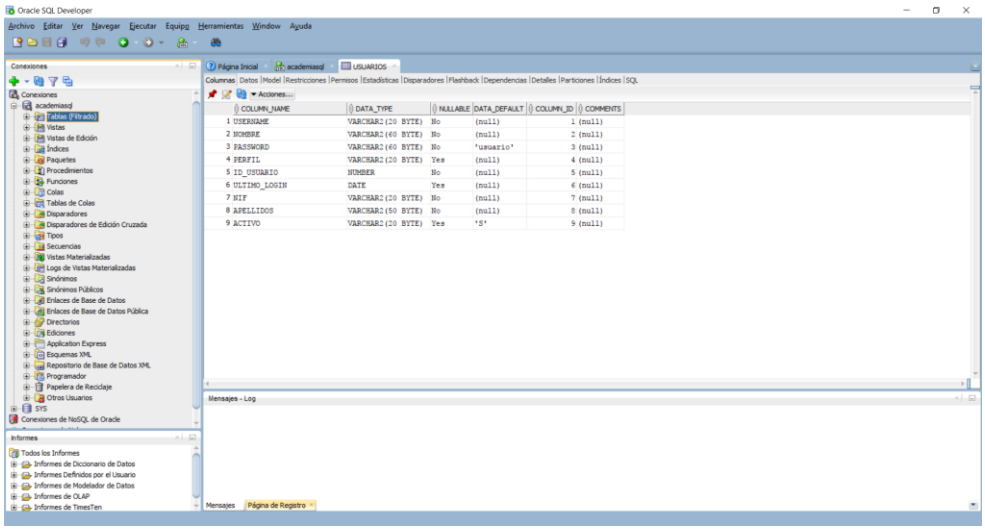


Ilustración 11: Imagen de SQLdeveloper

La anterior imagen muestra nuestro sistema gestor de base de datos, SQLdeveloper. Es la interfaz a la base de datos, lugar donde residen los datos.

SQLdeveloper nos sirve para ver el estado de la base de datos, no forma parte de la capa de datos, pero permite obtener una visión del concepto.

Todas las capas pueden residir en un único espacio, lo más común es que las capas se distribuyan en distintos ordenadores. Si es común que las capas de negocio y de datos residan en un único espacio, aunque si surge la necesidad, bien por aumento del tamaño de los datos, estas capas pueden separarse.

En este caso, al tratarse de una base de datos ejecutada en un pc doméstico, las tres capas conviven en el mismo espacio, ya que no hay gran carga de datos y no existen numerosas conexiones de usuarios.

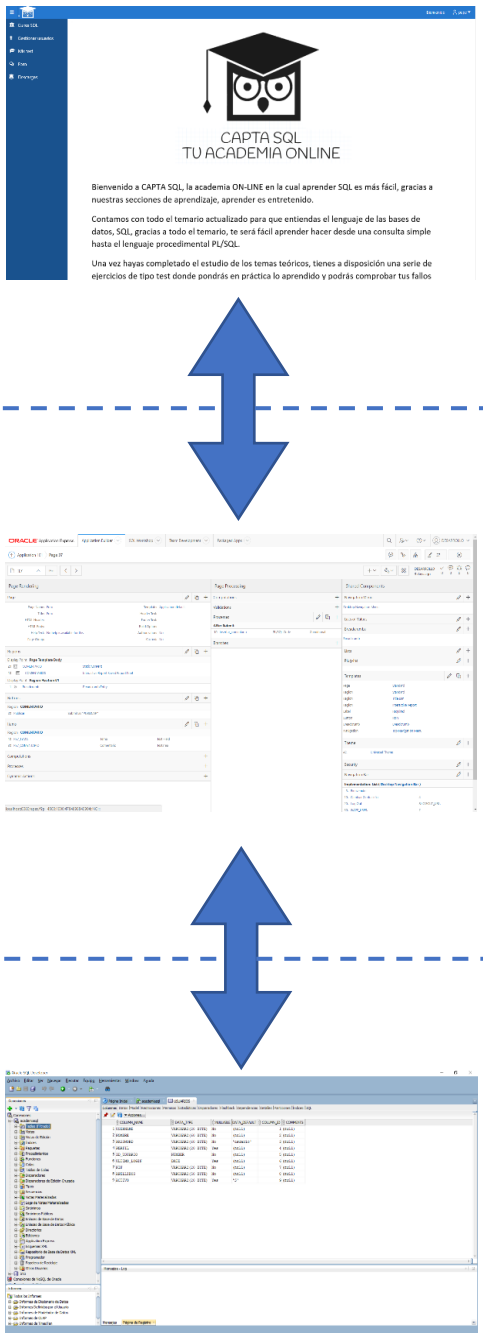


Ilustración 12: Imagen de las tres capas a modo representativo

La anterior ilustración muestra una visión de la comunicación entre capas.

4. Implementación

En este apartado se va a mostrar todo el proceso que se ha llevado a cabo para poder generar la aplicación combinando los pasos anteriores (análisis, diseño).

Los pasos que se han seguido para la implementación han sido los siguientes:

1. Creación de conexión BBDD – APEX.
2. Realizar conexión BBDD – SQL developer.
3. Desarrollar la aplicación según el diseño previo incorporando las funciones y procedimientos.
4. Carga de datos de la aplicación.
5. Realizar pruebas y corrección de errores.

Para llevar a cabo la implementación del proyecto, tal y como se ha mencionado anteriormente, se ha dado uso a las herramientas de desarrollo (APEX y SQLdeveloper) y bases de datos de ORACLE (ORACLE Database 11g xe).

¿Qué es APEX?

Aplication Express, también conocido como APEX, es la aplicación interna de las bases de datos de Oracle, permite explotar los datos de las bases de datos a través de una interfaz web.

APEX se caracteriza por ser una poderosa herramienta con la que se puede estar en producción en pocos minutos.

Desarrollo

APEX ofrece todas las herramientas necesarias para el desarrollo de la aplicación en un corto espacio de tiempo:

- Ofrece un entorno de desarrollo basado en web.
- Posee un completo editor de código con resalto de la sintaxis y sugerencias de código.
- Para desarrolladores menos experimentados, ofrece las herramientas de “arrastrar y soltar” los elementos.
- Se puede hacer cambios en tiempo real, sin tener que parar la herramienta ni entorpecer a otros usuarios.
- Soporte completo de SQL, PL/SQL y JavaScript.
- Permite integrar Plug-ings.
- Interfaz directa para usuarios de móviles: no es necesario crear una interfaz para móvil y otra para pc, es todo en uno. APEX transforma automáticamente las interfaces manteniendo el orden en los elementos.

Personalización

Personaliza la interfaz sin escribir ni una sola línea de CSS o JavaScript es una gran ventaja frente a otras herramientas de desarrollo web:

- Interfaz de usuario completamente personalizable.
- El diseño se enfoca a la accesibilidad de la interfaz gráfica.
- Componentes gráficos actualizados.
- Permite establecer colores y fuentes a través de “Theme roller”, una herramienta con la que para establecer el color de una región no es necesario escribir código.
- Soporte universal con librerías CSS.

La combinación perfecta de un desarrollo intuitivo y fácil junto con la personalización de los elementos para ofrecer una gran interfaz hacen de APEX una herramienta completa, fácil e intuitiva, con la que se pueden obtener aplicaciones completamente usables con poco tiempo.

Ejecución

APEX permite cargar las aplicaciones en un servidor propio o en la nube de Oracle:

- APEX se ejecuta como una fina capa sobre la base de datos, por lo que no requiere scripts complejos.
- Ofrece total portabilidad de las aplicaciones: puede estar en producción en pocos segundos.
- Entornos seguros, conexiones cifradas.
- Disponible generación de copias de seguridad.
- Entorno paralelo de contingencia.

Estos tres apartados resumen claramente la herramienta web que se utiliza en el desarrollo del proyecto, aunque se puede resumir en tres puntos clave:

- Desarrollo simplificado.
- Personalización sin esfuerzo
- Ejecución rápida.

Tres puntos que se basan en los tres pilares que dan fuerza a esta aplicación y con la que se concluye la explicación de la herramienta de entorno web.

¿Por qué APEX? Hay otras herramientas…

La utilización de APEX está principalmente motivada por el uso que se ha dado a esta tecnología a lo largo de la realización de las practicas universitarias.

Antes de comenzar las practicas, las herramientas de desarrollo web que conocía las conocí a través de la universidad. Son herramientas de desarrollo donde la presencia de código era prioritaria y donde se requería una implementación y desarrollo escrita en el mayor de los casos. La herramienta que más características cumplía, hasta que conocí APEX, para el desarrollo del proyecto es VISUAL STUDIO.

VISUAL STUDIO posee todas las características necesarias para poder realizar la aplicación de mi proyecto, sin embargo, no la diferencia prácticamente del resto de herramientas a la hora de crear una aplicación web, puedes arrastrar y soltar elementos web que quieras, pero para que funcione ese elemento web, hay que crear el código para dar una relación con botones, otros elementos o simplemente, alguna función.

Aquí está la diferencia con APEX. APEX no es necesario modificar código para obtener las funcionalidades de los elementos web, basta con arrastrar, soltar el elemento y añadir que función realiza ese elemento. Modificar código no es necesario para poder usar una aplicación.

Lo que la hace de APEX una herramienta rápida y sencilla para crear aplicaciones en corto espacio de tiempo.

Aun así, APEX dispone de consola de código para modificar los elementos en el caso de que se quieran obtener una personificación más avanzada y características más únicas y optimizadas.

4.1. Creación de la conexión BBDD – APEX

En este apartado se mostrará cómo se ha creado la conexión entre la base de datos y APEX.

Requisitos:

- Tener instalado en el sistema la Base de Datos. (ANEXO II)
- Tener instalado APEX junto a la Base de Datos.

Para la creación de la conexión de la Base de Datos con APEX, se necesitará que todos los requisitos anteriormente mencionados se cumplan.

La creación de la conexión se ha de realizar por medio de la ventana de comandos del sistema operativo. Los pasos son los siguientes:

1. Dirigirse hasta el directorio de instalación de APEX. En nuestro caso está en "*C:\oraclexe\apex":*

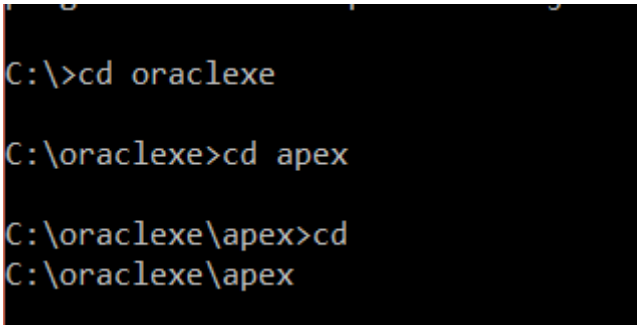


Ilustración 13: Proceso hasta llegar al directorio de instalación de APEX en consola

2. Una vez se está en la ruta de instalación de APEX, se procede a habilitar la conexión remota y habilitar el puerto por el que la base de datos "escuchará" el tráfico entrante y saliente.
   Para habilitar el puerto y la conexión remota al sistema se lanza el siguiente comando:

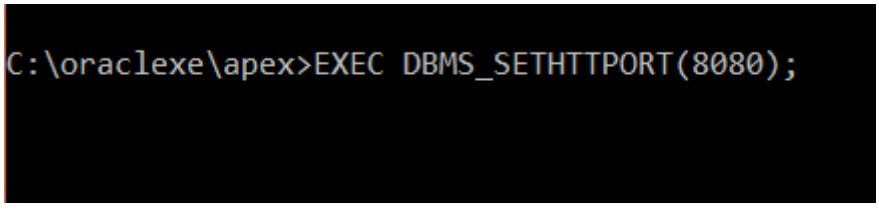


Ilustración 14: Comando habilitar puerto para APEX en consola

Teniendo el puerto y habilitado es el momento de configurar el "Listener", por donde la base de datos controlará el tráfico entrante y saliente de datos. Para ello se ejecuta el siguiente comando:

```
C:\oraclexe\apex>exec dbms_xdb.setListenerLocalAccess(l_acces=>FALSE);
```

Ilustración 15: Habilitar listener APEX desde la consola

Ya está configurado el puerto por donde se van a comunicar APEX y la Base de Datos, ahora hay que especificar quién va a realizar la conexión ente ambas entidades.

3. Para terminar de configurar la conexión entre Base de Datos y APEX es necesario indicar quien solicita esa conexión. La conexión será reclamada por la aplicación de APEX, pero esa información para la Base de Datos es irrelevante. Para que la base de datos produzca tráfico hacia APEX es necesario un usuario de base de datos.

4.1.1. Creación de usuarios - esquemas en Base de Datos para APEX

A continuación, se procede a crear un usuario - esquema de Base de datos. Este usuario - esquema será el propietario de la aplicación de APEX, por tanto, en este caso, se va a crear un usuario – esquema llamado igual que la aplicación que se desarrolla en APEX.

Para crear un usuario es necesario tener un espacio para el usuario - esquema, en Oracle, los espacios se llaman lógicamente Tablespaces (datafile físicamente). Por tanto, en primer lugar, se debe crear un tablespace para el usuario.

4.1.1.1. Creación TABLESPACES

Para crear un TABLESPACE se lanza el siguiente comando en nuestra Base de datos. Este tablespace se llamará ACADEMIA y se alojará en la ruta de instalación de la Base de Datos:

```
SQL>Create tablespace ACADEMIA datafile 'C:\Oracle\11.0.2.0\sqlplus\XE\datafile_ACADEMIA.dbf'
size 100M autoextend on next 1M;
```

Ilustración 16: Crear tablespace en consola

Se comprueba que el tablespace se ha creado correctamente:

```
SQL> select name from v$datafile;

NAME

--------------------------------------------------------------------------------
C:\Oracle\11.0.2.0\sqlplus\XE\datafile_ACADEMIA.dbf
C:\Oracle\11.0.2.0\APP\ORADATA\XE\SYSAUX.dbf
C:\Oracle\11.0.2.0\APP\ORADATA\XE\SYSTEM.dbf
C:\Oracle\11.0.2.0\APP\ORADATA\XE\USERS.dbf
C:\Oracle\11.0.2.0\APP\ORADATA\XE\UNDO.dbf
```

Ilustración 17: Comprobación de datafile en consola

Como se puede observar, el tablespace se ha creado correctamente. Ahora se procede a crear el usuario(esquema) en sí.

4.1.1.2. Creación de usuario - esquema

```
SQL>créate user ACADEMIA identified by academia default tablespace ACADEMIA;
```

Ilustración 18: Crear usuario en consola

Ahora se procede a comprobar que se ha creado correctamente:

```
SQL>select username,default_tablespace from dba_users where username like 'ACADEMIA' or
username like 'academia';


USERNAME                       DEFAULT_TABLESPACE
------------------------------------------------------------------------------------------
ACADEMIA                       ACADEMIA
```

Ilustración 19:Comprobar usuario en consola

El usuario se ha creado correctamente como se puede comprobar. Ahora se debe agregar el usuario en APEX. Para ello se debe seguir los siguientes pasos:

4.1.1.3. Agregar usuario – esquema en APEX

1. Iniciar en el modo de configuración y gestión de APEX y agregar nuevo espacio de trabajo:

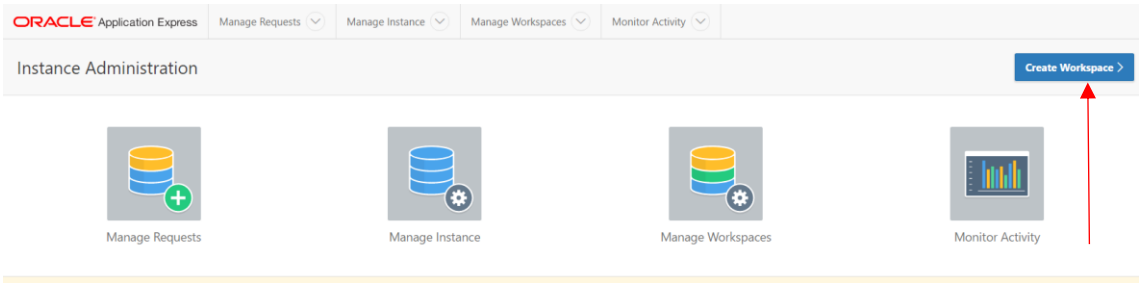


Ilustración 20: Página principal de administración de APEX

2. Indicar el nombre del espacio, en este caso **ACADEMIA:**

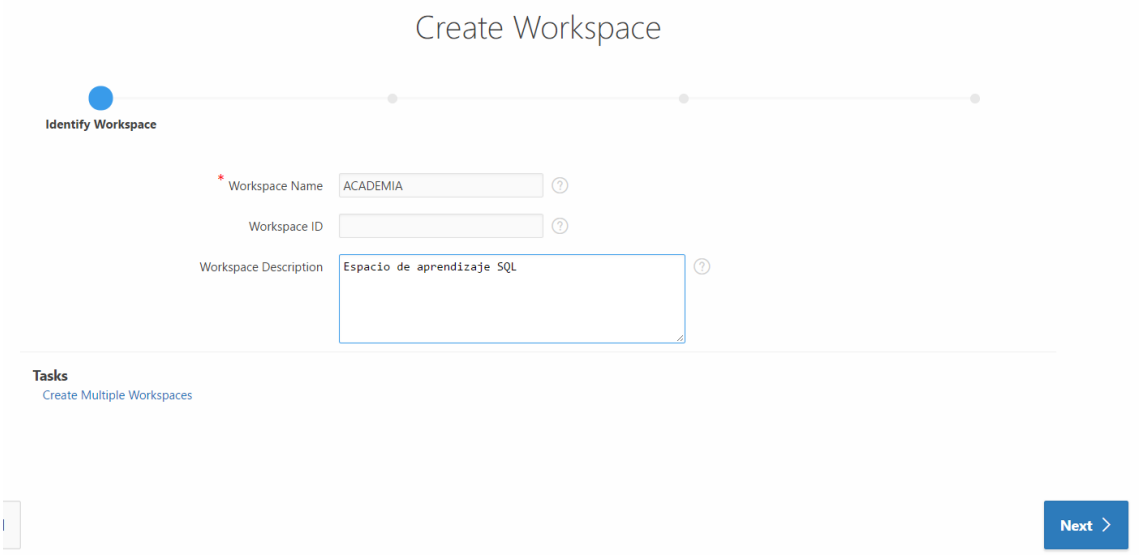


Ilustración 21: Crear Workspace-1 en APEX

3. Seleccionar “usar un esquema/usuario existente” y se selecciona el usuario que se ha creado anteriormente. Se introduce la contraseña del usuario - esquema que se creó anteriormente:

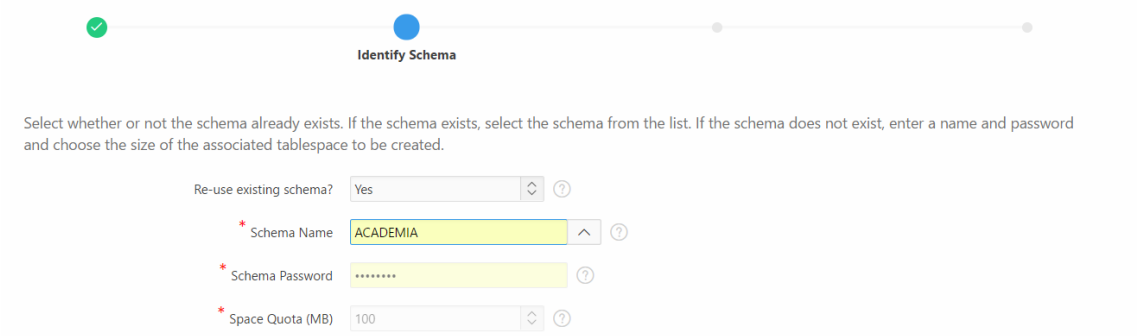


Ilustración 22: Crear Workspace-2 en APEX

4. Poner el nombre del administrador y su contraseña, **muy importante no olvidarlas,** pues no se podrá recuperar esas contraseñas.

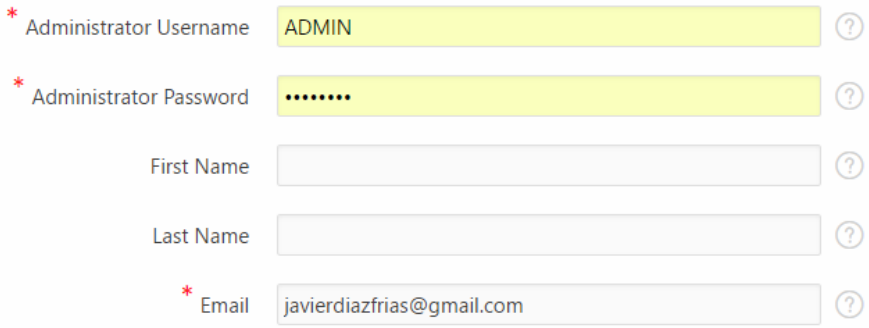


Ilustración 23: Crear Workspace-3 en APEX

Ya está lista la conexión entre APEX y la base de datos.

4.2. Realizar conexión BBDD – SQLdeveloper

SQLdeveloper es la herramienta grafica que permite acceder a la base de datos, no es la base de datos, simplemente es una aplicación que da visión y permite mejorar el trabajo con la Base de Datos. Como requisito previo, se necesita tener el usuario – esquema creado para poder agregar la conexión. (Ver crear usuario – esquema)

Una vez se tiene creado el usuario – esquema se procede a agregar la conexión con Base de Datos:

1. Se abre SQLdeveloper y se pulsa en agregar nueva conexión:

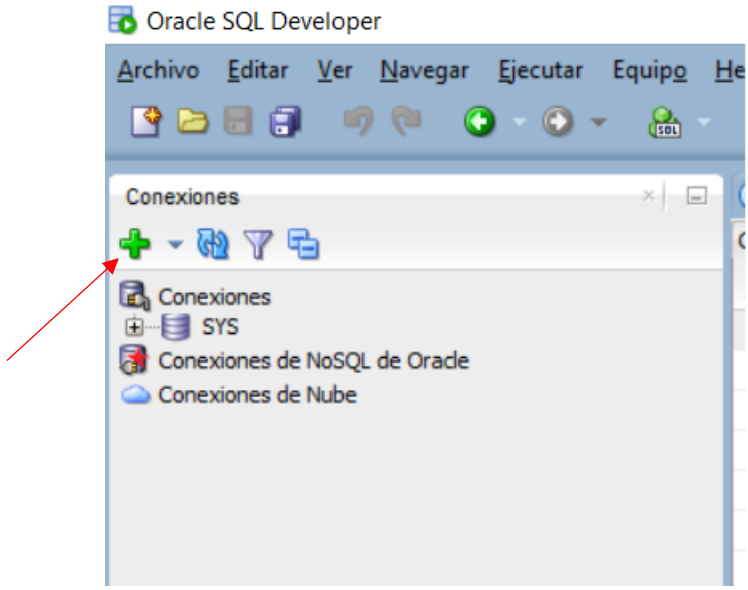


Ilustración 24: Ventana conexiones SQLdeveloper

2. Se agrega la conexión:

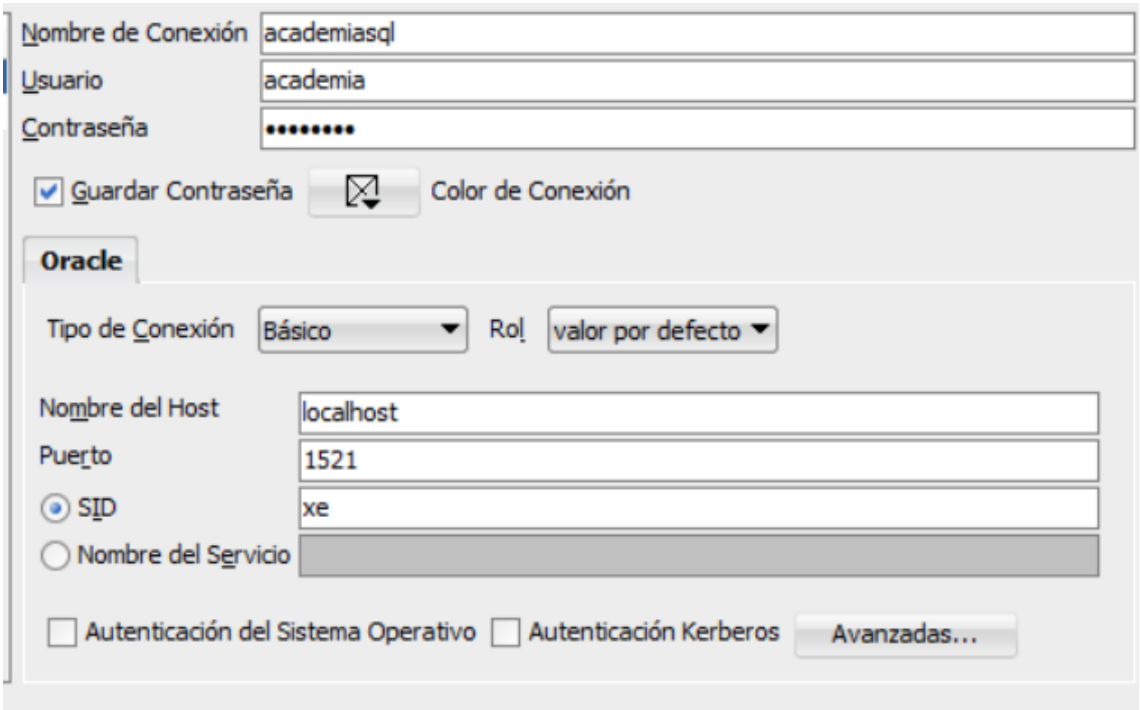


Ilustración 25: Configuración de conexión en SQLdeveloper

- Nombre de conexión: nombre de cómo se quiere llamar a la conexión, en este caso, para mantener un orden y una lógica, llamarla “academiasql”
- Usuario: nombre del usuario que se ha creado anteriormente.
- Contraseña: contraseña del usuario.
- Host: en nuestro caso al tratarse de una Base de datos local, se establece el valor de “localhost”, pero si se tratara de una Base de Datos alojada en un servidor se introduce su IP.
- Puerto: por defecto es el 1521, no es el mismo puerto que se indicó a la base de datos para la conexión con APEX.
- SID: nombre de la instancia de la base de datos:

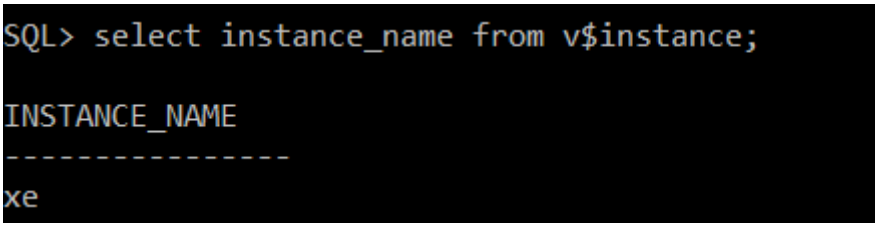


Ilustración 26: Comprobar instancia en consola

Y se tiene configurado SQLdeveloper.

4.3. Desarrollo de la aplicación

En este apartado se va a mostrar el planteamiento que se ha seguido para hacer la aplicación. Existen requisitos previos al desarrollo:

- Tener Base de datos instalada.
- Tener conexión base de Datos con APEX.
- Tener usuario con perfil administrador de APEX.

Si se tienen cumplidos los requisitos anteriores se puede comenzar con el desarrollo de la aplicación.

La aplicación va a estar desarrollada en 5 partes:

4.3.1. CURSO SQL

Este apartado contiene la base teórica para el aprendizaje del lenguaje SQL, de tal forma que la teoría adquiera mayor nivel de complejidad a medida que se avanza en el temario.

Los temas teóricos quedan ordenados de menor a mayor complejidad, quedando así:

1. Introducción a SQL:
    a. ¿Qué es SQL?
       Explicación del lenguaje SQL, cuáles son sus características y que fin tiene.
    b. Tipos de Campo.
       Define los tipos de campos que hay presentes en SQL.
    c. Tipos de datos SQL.
       Muestra todos los tipos de datos que acepta SQL.
    d. Tipos de sentencias SQL.
       Realiza una descripción de cada tipo de sentencia empleada en SQL.
2. Inserción y modificación de datos:
    a. Creación de tablas.
       Creación de las tablas de la Base de Datos.
    b. Estructuras de las tablas en SQL.
       Muestra como está diseñada una tabla.
    c. La cláusula CONSTRAINT.
       Añade restricciones a la tabla.
    d. Creación de índices.
       Muestra que es un índice y como se crea.
    e. Modificar el diseño de las tablas.
       Modifica una tabla, añadir columnas, eliminar, cambiar tipo.
    f. Añadir nuevo registro: INSERT.
    g. Borrar registros: DELETE.
    h. Actualizar un registro: UPDATE.
3. Búsqueda y selección de datos en SQL:
    a. Consultas de selección.
       Consultas básicas, devolver datos, ordenar, alias, etc.
    b. Criterios de selección en SQL.
       Filtrar registros mediante where, agrupar, máximo, mínimo, etc.
    c. Subconsultas.
       Crear consultas dentro de una consulta.
    d. Consultas SQL de unión interna.
    e. Consultas SQL de unión Externas.
4. Programación avanzada en SQL.
    a. Cursores en SQL.
       Mostrar conjunto de datos.
    b. Procedimiento en SQL.
       Lenguaje procedimental.

La teoría está comprobada y está redactada siguiendo patrones de manuales oficiales.

4.3.2 Mis TEST

En el siguiente apartado el usuario tiene la posibilidad de demostrar que ha aprendido y comprendido el temario estudiado. Para ello la aplicación pone a disposición de los usuarios la posibilidad de realizar test. Estos test se caracterizan por:

- Test con un total de 15 preguntas.

- Test de una única respuesta.
- Preguntas irrepetibles dentro de cada test.
- Capacidad de finalizar el test sin tener que contestar todas las preguntas.
- Posibilidad de reanudar el test.
- Corrección con respuestas correctas.

En los puntos anteriores se resumen las características que tienen los test.

El usuario puede acceder a ver todos los resultados de los test y puede imprimir esos resultados en un informe de Excel.

En el apartado de test existen restricciones, cada usuario puede ver sus test únicamente, al igual que acceder a sus resultados. Los test no se comparten con el resto de usuarios.

4.3.3. Foro

Este apartado está disponible para todos los usuarios. La aplicación pone a disposición de los usuarios un espacio donde es posible abrir comentarios caracterizados por un tema inicial e ir creando un hilo de conversación entre todos los usuarios.

En el apartado del foro existen restricciones, cada conversación creada posee un creador, por tanto, se reserva el borrado de la conversación al creador de ella.

4.3.4. Descargas

La aplicación dispone de un espacio de intercambio de archivos donde los usuarios pueden compartir sus documentos, fotos, programas, etc., con el resto de los usuarios.

Existe la restricción de que el archivo solo lo puede eliminar el usuario que ha compartido el archivo con la comunidad y también el *administrador*, ya que cualquier archivo subido puede contener software malicioso o no estar relacionado con el fin de la pestaña de descargas, que es compartir el conocimiento y ayudar a el resto de usuarios.

4.3.5. Gestión de usuarios

Esta pestaña, únicamente disponible a *administradores*, posee la capacidad de poder gestionar los usuarios de la aplicación. Dispone de las siguientes funciones:

- Crear usuarios.
- Dar de baja a usuarios.
- Modificar usuarios.
- Resetear contraseña olvidada a un usuario.

Esta pestaña está reservada únicamente a los *administradores*, pues contiene información de usuarios que no han de ser compartida.

La aplicación basa su funcionamiento en los anteriores apartados, teoría, test, archivos, foro y usuarios, funciones que hacen que la aplicación sea intuitiva debido a la separación que hace entre los distintos apartados.

4.4. Carga de datos de la aplicación

La aplicación tiene como fin enseñar y examinar, por tanto, basándonos en estos dos principios, la teoría debe ser fiable 100% al igual que los test de examen.

La teoría ha sido comprobada minuciosamente ofreciendo gran fiabilidad en lo descrito al igual que en los test.

Las preguntas y respuestas de los test han sido creados por el autor de la aplicación. Las preguntas planteadas en la realización de los exámenes se han planteado cuidadosamente para que no ofrezcan duda alguna, al igual que las respuestas ofrecidas.

La carga de datos ha sido muy importante a la hora de realizar el testeo de la aplicación, pues se ha podido comprobar que la aplicación funciona correctamente.

4.5. Realizar pruebas y corrección de errores

Una vez se completa la incorporación de funciones y opciones de la aplicación, llega el momento de realizar el testing.

Como requisito para este apartado esta la carga de datos, pues no podremos comprobar el funcionamiento si no hay datos en la aplicación cargados.

La inversión en el tiempo del testing obtuvo beneficios, pues gracias a ello fue posible obtener errores y su posterior corrección.

Un apartado que requirió la mayor parte del tiempo del testing fue el apartado de los test.

Este apartado, como comentamos en anteriores puntos, posee ciertas características:

- Test de 15 preguntas
- Poder finalizar antes de realizar todas las preguntas.
- Poder continuar realizando un test.
- Test donde las preguntas no se repiten
- Una única respuesta verdadera.
- Resultados generados en un Excel.

Los anteriores apartados dictan las características de los test. El mayor de los problemas se encontró en el punto 3 (Poder continuar realizando un test).
Este apartado se caracteriza en que el usuario puede realizar un test y tiene la posibilidad de

salir del apartado de los test para dirigirse a cualquier otro lugar en la aplicación y poder regresar al test que estaba realizando.

El problema que surgió fue que una vez que salía del test, cuando volvía para continuar con el test, comenzaba de nuevo, lo que hacía perder el progreso del test.

Como solución se incluye una variable que guarda el estado del test en el momento que el usuario abandona el test. Por tanto, el usuario cuando vuelve a continuar con el test, puede hacerlo siguiendo el progreso que llevaba entonces.

Otro de los problemas que surgieron en los test, fue con los resultados generados en un Excel, el punto 6.

La generación de Excel se realiza mediante un código PL/SQL. Esta función utiliza un cursor donde guardaba el resultado de una consulta. Se utiliza un cursor pues es necesario recuperar el resultado de todas las preguntas realizadas.

El problema surge en el momento que se quiere recuperar el resultado de un test de un día pasado, pues hay que indicar al procedimiento PL/SQL de qué fecha se quiere el resultado, en este momento surge el problema, el formato de fecha que se estaba utilizando era incompatible con Excel.

Como solución surge modificar el formato de fecha de las variables de APEX, a priori el problema debería estar superado, sin embargo, recordemos que APEX es únicamente la aplicación y que no dispone de los datos, los datos están disponibles en la Base de Datos, la cual ya dispone de datos y no permite modificar tablas con datos.

La solución definitiva está en cambiar el formato de fecha de APEX y de la aplicación en desarrollo mediante variables globales. Para ello se ejecuta el siguiente código:

```
ALTER SYSTEM SET NLS DATE FORMAT='DD/MON/YYYY' SCOPE=SPFILE;
```

Ilustración 27: Comando establecer formato de fecha

El anterior comando cambia el formato de entrada de datos bidireccionalmente Base de datos – APEX, por ejemplo en la Base de Datos las fechas están formateadas en 'YYYY-MM-DD' (2017-05-29) y en APEX utiliza el formato 'DD/MON/YYYY' (29/MAY/2017), por tanto a la hora de realizar búsquedas en base de datos desde APEX no se devuelve ningún resultado, sin embargo, una vez ejecutado la anterior sentencia, la conversión de datos se realiza correctamente y así se acaban los problemas con las fechas en toda la aplicación.

**Código PL/SQL se incluye en anexo. (Ver* Función descargar informe. *Página 54)*

El resto de apartados - Teoría, FORO, Descargas y gestión de usuarios, no presentaron problemas graves, únicamente hubo que controlar el acceso a las pestañas dependiendo del rol que posee cada usuario.

Para ello se utilizan esquemas de seguridad como el siguiente:

```
declare
 l_tipo_usuario VARCHAR2(20);
begin
  SELECT perfil into l_tipo_usuario
  FROM USUARIOS
  WHERE USERNAME = v('app_user');
  if l_tipo_usuario = 'ADMIN' then
    return false;
  else
    return true;
  end if;
end;
```

Ilustración 28: PL/SQL de esquema de autenticación de ADMIN

La anterior expresión PL/SQL devuelve un valor booleano verdadero o falso dependiendo del perfil del usuario que ha iniciado sesión. Si es verdadero se concede acceso a la página, ejecución de funciones, mostrar elementos, etc., por el contrario, se impide el acceso.

Se realiza un testing de todas las funciones, haciendo hincapié en las más críticas, para evitar que surjan errores que impidan el buen funcionamiento de la aplicación, por tanto, el tiempo de testing es largo, aunque se obtienen buenos resultados ya que, excepto los errores anteriormente descritos, no se han encontrado errores que imposibiliten el funcionamiento de la aplicación.

Una vez se finalizan las pruebas y corrección de errores se ha finalizado el proceso de implementación de la aplicación, lo que ya es apta para lanzarla a uso del público, aunque la aplicación ofrece un desarrollo continuo ya que ofrece muchas capacidades de explotar la plataforma.

5. Conclusiones

5.1. Conclusiones técnicas

Como en todo proyecto técnico, el principal objetivo es cubrir las necesidades de un cliente, -en el caso de este proyecto el usuario final de la aplicación. Para lograr ese objetivo se ha recurrido a una planificación marcada por importantes fases, las cuales han variado a lo largo del proyecto, pero han concluido dentro del espacio previsto.

El desarrollo de este proyecto ha implicado el aprendizaje y/o consolidación de las tecnologías de Oracle vinculadas a las Base de Datos. También se ha puesto en uso conocimientos técnicos de análisis y diseño de patrones.

El producto final ha sido una suma lógica de todas las tareas marcadas en el plan de trabajo seguido a lo largo de todo el desarrollo del proyecto.

5.2. Conclusiones personales

Desde el punto de vista personal, el desarrollo del proyecto ha constituido un gran reto para mí.

El trabajo final de grado es el final de mi etapa académica, es el resultado de sumar todos los años de esfuerzo, logrando al fin poner punto y final a esta titulación.

A pesar de que años anteriores tenía la gran incertidumbre sobre que trabajo haría llegado el momento, la suerte de haber realizado mis prácticas en un área parcialmente desconocida para mí, como es la administración de Base de Datos, me ha dado la oportunidad de poner en práctica que lo aprendido en éstas prácticas no solo me establecen una base para mi vida laboral, sino que también me han aportado conocimientos sobre plataformas desconocidas para mi hasta ahora, las cuales he visto que ofrecen un gran potencial y que me gustaría compartir con más personas.

Como comentaba, aunque tenía una base de las herramientas usadas en este trabajo que adquirí en mis prácticas, enfrentarme yo solo a una aplicación creada desde cero ha supuesto realizar un gran esfuerzo personal que ha dado como resultado ampliar mis conocimientos que tenia de las herramientas.

Es por todo esto, que el esfuerzo extra dedicado éstos últimos meses a compaginar el desarrollo de mi proyecto junto a mi vida laboral ha merecido la pena, pues, aunque ha sido duro, ha supuesto una nueva experiencia.

6. Líneas futuras

Las líneas futuras de desarrollo son las siguientes:

- Lanzar la aplicación en un servidor.
- Mejorar las funciones y actualizar procesos.
- Mejorar las interfaces dotando de una mejor experiencia para el usuario
- Continuar con una saga de aplicaciones con el nombre base de “Capta”, por ejemplo, “Capta inglés” o “Capta Mates”.

7. Presupuesto

Basándose en las líneas futuras que se pueden dar a esta aplicación, se observa que el crecimiento puede ser muy grande. Es por esto que se debe realizar un análisis.

Se debe evaluar qué impacto se tendrá cuando se estén ejecutando varias aplicaciones en la misma Base de Datos y que no se produzcan cortes en el servicio y tampoco se rechacen conexiones por exceso de usuarios.

A pesar de que el crecimiento puede ser muy alto, el impacto que realiza APEX sobre la Base de Datos es muy bajo.

Teniendo la oportunidad que brinda ORACLE de ofrecer productos sin coste para fomentar la educación, disponemos de una base de datos sin ningún coste, aunque queda exenta de soporte y no dispone de sistema de recuperación ante caídas del sistema, lo que puede dejar a usuarios sin poder usar la aplicación.

Los costes que se tienen son el servidor que se usará para instalar la Base de Datos y el entorno de APEX, el coste oscila entre 900 y 1500€, un precio muy atractivo que da un impulso extra a la expansión de más aplicaciones; y el coste del desarrollo de la aplicación/es.

Un ejemplo de un servidor es el siguiente:

- DELL PowerEdge T630.
- Intel® Xeon® E5-2600 v4 más reciente, la memoria DDR4 y siete ranuras de E/S.
- Admite hasta 32 discos duros de 2,5" o 18 de 3,5".
- Hasta cuatro unidades de estado sólido PCIe NVMe Express Flash y el controlador RAID PowerEdge 12Gb.

Una configuración básica de 4 TB y 12 GB de RAM es suficiente con 6 núcleos, aunque en el futuro, con mayor expansión será preciso incorporar más memoria.

Precios sin replicación (establecido en el proyecto)

El servidor tiene instalado Oracle Linux, una distribución Linux adaptada por Oracle a las Bases de Datos. Este sistema operativo posee un coste de cero euros, pues no es preciso soporte para la Base de Datos, ya que no se va a requerir gran carga de procesos.

El coste de la aplicación oscila entre los 4.000*€ y 6.000*€, (únicamente es el coste del desarrollo ya que los programas son gratuitos) dependiendo del contenido de la aplicación y del tamaño de esta.

El precio incluye el mantenimiento** por 6 meses desde la entrega del producto. Una vez finalizado el mantenimiento, el precio debe negociarse según las áreas de soporte a las que se quiere acceder.

	APEX	*Base de Datos*	*APEX + Base de Datos*
1 Año	1500* € / Año	900* € / Año	2350* € / Año
2 Años	1450* € / Año	850* € / Año	2300* € / Año
3 Años	1350* € / Año	750* € / Año	2000* € / Año

***Mantenimiento no incluye implementación de funciones, solo soporte de incidencias.*

Precio con replicación

Si queremos disponer de un sistema que nos aporte alta disponibilidad y capacidad de recuperación. La base de datos debe ser la versión Estándar y se debe de evaluar en primer lugar los usuarios, ya que ORACLE posee dos tipos de precios, por usuario o por núcleo del servidor.

El servidor posee 6 núcleos en dos sockets. Si es Intel, se paga el 50% del precio de los núcleos.

Precios por usuario con Base de Datos en el servidor

Los precios de licencia de Base de Datos de la versión Estándar por usuario son los siguientes:

Base de Datos en el servidor. Precio por usuarios

Para 50 usuarios	350€/usuario	77€/usuario/año	17.627 €
Servidor			2 x 1.500 €
TOTAL			**24.127* €**

Es un precio muy atractivo pero que se limita a 50 usuarios.

Precios por núcleo con Base de Datos en el servidor

Los precios de licencia de Base de Datos de la versión Estándar por núcleo son los siguientes, se recuerda que, al contar con un servidor de Intel, se pagan la mitad de los núcleos. Esta versión posee usuarios ilimitados. Los precios son los siguientes:

Base de Datos instalada en Servidor. Precio por núcleos

Para 12 núcleos (pagas la mitad)	17.500 € * 6 núcleos	3.850 €* 6 núcleos	128.100 €
Servidor			2 x 1.500 €
TOTAL			**131.100* €**

Es un precio bastante más alto pero que no supone ningún límite en cuanto al límite de los usuarios que pueden acceder.

Precios por núcleo con Máquina Virtual

Cabe la posibilidad de instalar la Base de Datos en una máquina virtual, lo que permite destinar parte de los núcleos a la base de datos y otra parte a la interfaz web. La distribución queda así:

Máquina virtual y Base de Datos Estándar

Base de Datos	3 núcleos	17.500 € * 1.5 + 3.850 € *1.5 (pagas la mitad de núcleos)	33.950 €
Base de Datos de replica	1 núcleo	17.500 € 0.5 +3.850€ *0.5	10.675 €
WebService Tomcat	2 núcleos	GRATIS (Usa Oracle XE)	
Base de Datos desarrollo	2 núcleos	GRATIS (Usa Oracle XE)	
Servidor			2 x 1.500 €
Sistema Operativo		17.500 € *1.5 +3.850€ *1.5	32.025 €
TOTAL			**79.650* €**

Se observa que el precio más atractivo, si queremos contar con replicación, con número ilimitado de usuarios y con soporte directo de Oracle, es un sistema de Máquina virtual, la última opción.

Todos los precios que se incluyen deben sumarse el precio de la aplicación mencionado en el apartado de “Precios sin replicación (establecido en el proyecto)” Página 48.

**Todos los precios finales incluyen I.V.A*

Bibliografía

- Foro, scripts y tutoriales para administración de ORACLE
  https://oracle-base.com

- Página principal de APEX
  https://apex.oracle.com/

- Web especializada en Productos ORACLE
  http://www.orasite.com

- Comunidad de programadores
  https://stackoverflow.com/

- Utilización de UML en ingeniería del software con objetos y componentes 2ª ed..
  Stevens, Perdita. S004.41UMLSTE. 2007

- Página principal de ORACLE
  http://www.oracle.com

- Blog Dimitri Gielis
  http://dgielis.blogspot.com.es/

- Wikipedia
  www.wikipedia.es

- APEX NINJAS
  www.apexninjas.com

- Desarrollo WEB
  www.desarrolloweb.com

- WEB especialista en ORACLE
  http://www.explorer-development.uk.com

- Microsoft academy
  https://social.msdn.microsoft.com

- CodeJobs
  https://www.codejobs.biz

- Solvetic
  www.solvetic.com

- DELL
  www.dell.es

8. ANEXO 1: Documentación de la aplicación Capta SQL

A continuación, se va a redactar la documentación de la aplicación Capta SQL. Esta documentación sirve como guía de utilización y comprensión de la aplicación Capta SQL.

Adicionalmente, se incluye el modelo de datos utilizado en la Base de datos.

8.1. Página de *Logging*

A continuación, se tiene la página de inicio de sesión en la aplicación, dicha página solicita el usuario y contraseña para poder acceder a usar la aplicación. En caso de no disponer usuario, no será posible el acceso.

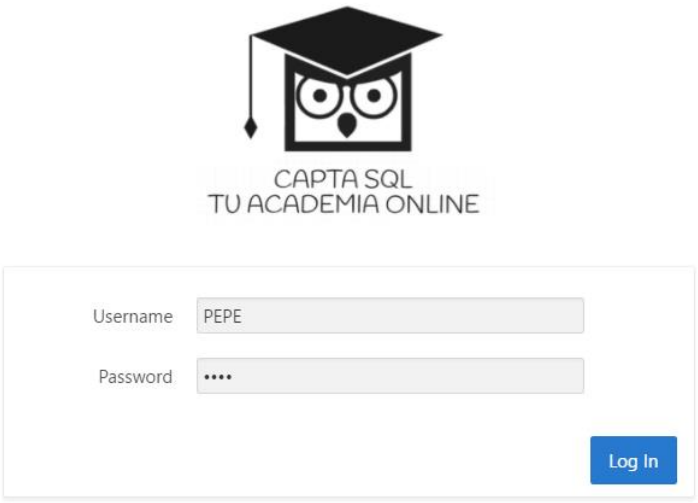


Ilustración 29: Pagina de Logging de la Aplicación CAPTA SQL

La comprobación de inicio de sesión se realiza mediante la llamada a una función PL/SQL que comprueba el usuario y contraseña en la Base de Datos y devuelve verdadero o falso dependiendo de si existe el usuario o no

```
  function f_autoriza_usuario(p_username IN VARCHAR2, p_password IN VARCHAR2)
  return BOOLEAN
is
    registros number;
    pwd_en_bd varchar2(4000);
    pwd_encriptado varchar2(4000);
begin
    SELECT count(*) into registros FROM USUARIOS WHERE username = upper(p_username);
    IF registros > 0 THEN
        SELECT password into pwd_en_bd FROM USUARIOS WHERE username = upper(p_username) and activo = 'S';
        pwd_encriptado := f_encriptar_pwd(p_password,p_username);
            IF p_password = pwd_en_bd or pwd_encriptado=pwd_en_bd THEN
                return true;
            ELSE
                return false;
            end IF;
    ELSE
        return false;
    end IF;
end f_autoriza_usuario;
```

Ilustración 30: PL/SQL de validación de usuarios

8.2. Página principal

Si el inicio de sesión es correcto, aparece la página principal donde se presenta la aplicación.

En la parte de la izquierda se encuentra el menú principal, donde se recogen las distintas secciones de la aplicación.

Ilustración 31: Página principal de aplicación CAPTA SQL con menú principal resaltado

En la parte superior izquierda se encuentra el logo de la aplicación y en la parte izquierda se observa el nombre de usuario de la persona que está en la aplicación. El nombre posee un desplegable que permite acceder a gestionar la contraseña del usuario y a salir de la aplicación.

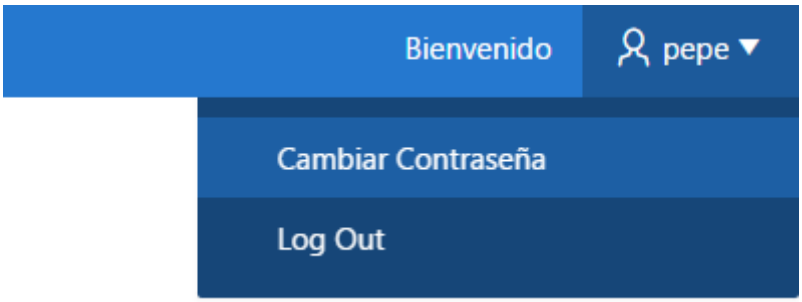


Ilustración 32: Log-out y cambiar contraseña de la aplicación CAPTA SQL

La aplicación posee 5 grandes puntos:

- Curso SQL
- Mis TEST
- Gestión de usuarios (solo visible a administradores)
- Foro
- Descargas

8.3. Curso SQL

En este apartado se encuentra disponible los temarios de SQL y PL/SQL. La teoría está organizada de menor a mayor dificultad, manteniendo siempre un control para que el aprendizaje sea continuo.

La teoría se organiza por bloques de aprendizaje. El acceso a estos bloques está distribuido por menús, por ejemplo, el menú principal de la teoría queda así:

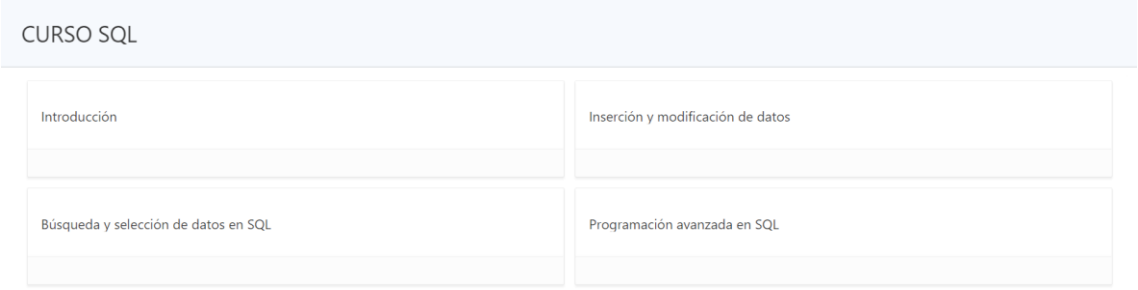


Ilustración 33: Menú de organización de teoría de la aplicación CAPTA SQL

Dentro de cada bloque se encuentra los temas relacionados con el menú principal. En el caso de "Inserción y modificación de datos" se encuentran las categorías correspondientes:

Ilustración 34: Ejemplo de submenú de teoría de la aplicación CAPTA SQL

Si se escoge, por ejemplo, el tema de "Creación de tablas" se accede a la teoría de cómo crear una tabla:

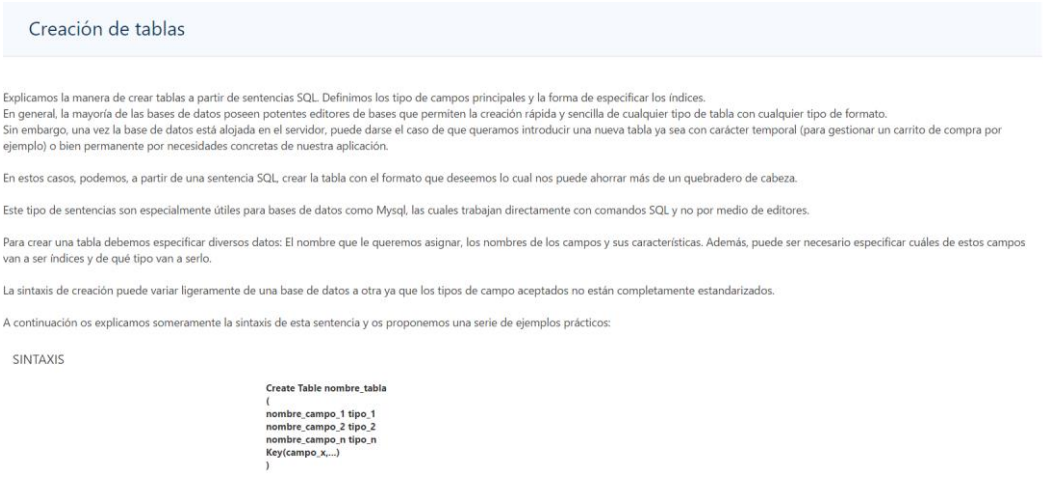


Ilustración 35: Teoría de crear tabla de la aplicación CAPTA SQL-1

La organización que se muestra en la página, (explicación, estructura, ejemplo) se cumple en todas las secciones del curso para ofrecer un mejor entendimiento y comprensión de la teoría propuesta.

En la parte baja de todas las páginas del curso se encuentran tres botones:

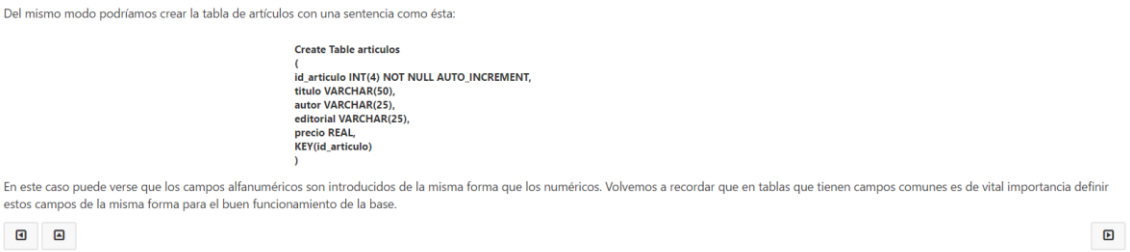


Ilustración 36: Teoría de crear tabla de la aplicación CAPTA SQL-2

Por orden, de izquierda a derecha, estos botones indican:

- Página anterior.
- Acceso a la sección en la que se encuentra (en este caso se mostraría el menú "Inserción y modificación de datos".
- Página siguiente.

La inserción de datos en todas las páginas del curso SQL es idéntica, se ha realizado por medio de texto plano, es decir, no tiene relación con la base de datos, esta información se encuentra en la propia aplicación dentro de elementos web.

Como excepción a los botones de movimientos entre páginas, en la última página del curso, en la sección "Programación avanzada SQL", el botón de avanzar dirige automáticamente a la sección de Test para poder comenzar el examen.

8.4. Mis test

La siguiente sección permite a los usuarios comprobar lo aprendido a lo largo del curso de SQL, para ello se pone a disposición de los usuarios la opción de realizar test de 15 preguntas.

En la pantalla inicial, antes de comenzar el test, está disponible un botón de acceso al test, una región que recoge los test que se han realizado y en la parte baja aparece una gráfica con los aciertos que se han obtenido en los test realizados para poder llevar control del progreso, ésta grafica solo aparece en el momento que haya test disponibles.

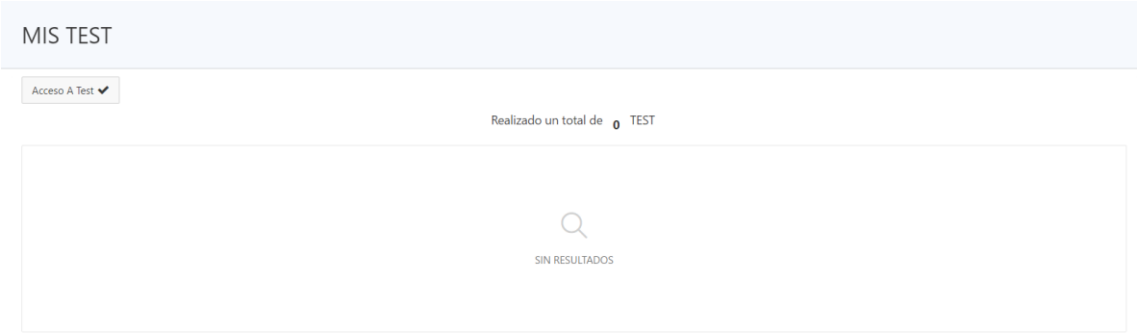


Ilustración 37: Página principal de TEST de la aplicación CAPTA SQL

8.4.1. Acceso a TEST

A través del botón es posible acceder al test que medirá el nivel de conocimiento adquirido.

Una vez se accede al test, la pantalla muestra la pregunta con 4 posibles respuestas:

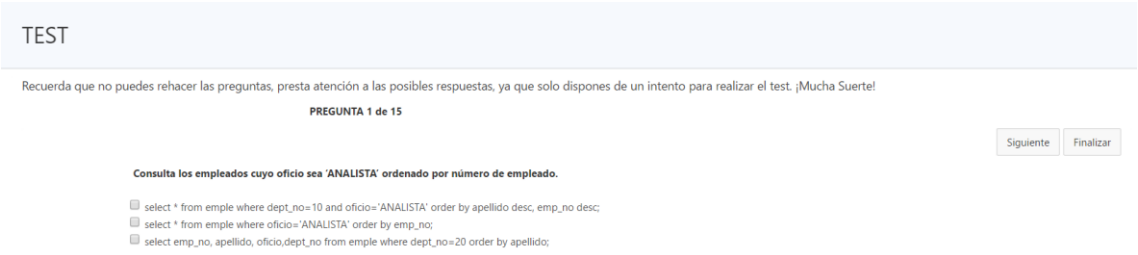


Ilustración 38: Ejemplo de TEST de la aplicación CAPTA SQL

A lo largo de todo el test estarán disponibles dos botones, “Siguiente” y “Finalizar”, salvo en la última pregunta que solo aparece “Finalizar”.

El botón de siguiente avanza a lo largo del test a la siguiente pregunta. En todo momento es posible salir del test que se está realizando para acceder a otra sección y después reanudarlo, pues no hay tiempo de realización ni finaliza hasta que no se pulsa el botón “Finalizar”, que da por finalizado el test, se hayan completado las 15 preguntas o no y establece el valor del atributo *intento* de la tabla *test* de la Base de datos como el valor más alto + 1 del número de test realizados.

Función que muestra el enunciado

```
SELECT enunciado FROM
(SELECT enunciado FROM preguntas where id_pregunta not in(select id_pregunta from test where
id_usuario=:p42_id_usuario and fecha=:p42_fecha)
ORDER BY dbms_random.value )
WHERE rownum=1 ;
```

Ilustración 39: Consulta de enunciado de test de la aplicación CAPTA SQL

La anterior función muestra un enunciado de la tabla enunciados al azar(random) y comprueba que no se haya mostrado ya esa pregunta en el mismo día y para el mismo usuario.

Función que muestra las respuestas

```
SELECT RESPUESTA AS DISPLAY_VALUE,ID_RESPUESTA AS RETURN_VALUE
FROM RESPUESTAS
WHERE ID_PREGUNTA=:P42_ID_ENUN_1

UNION

SELECT RESPUESTA AS DISPLAY_VALUE,ID_RESPUESTA AS RETURN_VALUE
FROM RESPUESTAS
WHERE tipo=(select tipo from preguntas where id_pregunta=:P42_ID_ENUN_1) and ROWNUM < 4;
```

Ilustración 40: Consulta de respuestas de la aplicación de CAPTA SQL

La función anterior obtiene la respuesta correcta comprobando con el enunciado (Primera parte) y completa hasta cuatro respuestas totales con respuestas similares que pertenezcan al mismo tipo de respuesta (después de la unión).

Función botón Siguiente

El botón “Siguiente” sirve para avanzar de pregunta en el test, sin embargo, internamente, cuando avanza de pregunta se realiza una inserción en la tabla *TEST* de la Base de datos con los datos del test, el usuario y la fecha. Se inserta el valor de 0 en el atributo de *INTENTO*.

```
begin
insert into TEST(id_pregunta,id_respuesta,ID_usuario,fecha,INTENTO) values
(:p42_id_enun_1,:p42_respuesta_1,:p42_id_usuario,to_char(sysdate,'DD/MON/YY'),0);
end;
```

Ilustración 41: Función "INSERT" de test de la aplicación CAPTA SQL

Función botón Finalizar

El botón “Finalizar” realiza dos funciones, la misma función que el botón “Siguiente” y actualizar el valor del atributo *INTENTO* igual a 0 como el valor más alto de los intentos más uno.

```
UPDATE TEST
SET INTENTO=(SELECT DISTINCT MAX(INTENTO)+1 FROM TEST WHERE intento>0 and   ID_USUARIO=:P42_ID_USUARIO)

WHERE INTENTO=0 AND ID_USUARIO=:P42_ID_USUARIO;
```

Ilustración 42: Función de "UPDATE" test de la aplicación CAPTA SQL

Una vez se pulsa el botón finalizar, dirige al usuario al menú de “TEST”.

8.4.2. Resultados

Una vez se ha finalizado el test, es el momento de comprobar las preguntas que se han contestado, si están correctas o no.

Para comprobar las respuestas, en la página principal de "TEST" aparecen los test realizados, ordenados por fecha.

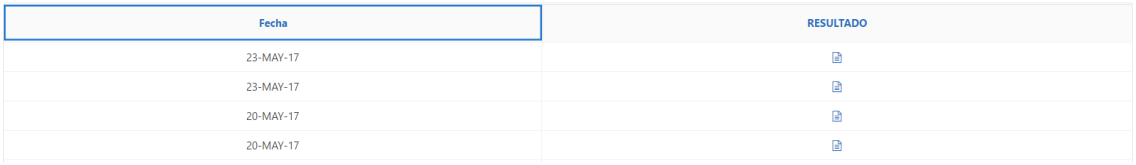


Ilustración 43: Test realizados de la aplicación CAPTA SQL

Cada usuario puede únicamente acceder a los test que él mismo ha realizado. La anterior imagen muestra los test del usuario, para ello se realiza una consulta a la Base de Datos:

```
select  fecha ,id_usuario as hidden ,intento
from test t, respuestas r
where   id_usuario=(select id_usuario from usuarios where username=v('app_user')) and intento is not null
group by id_usuario,fecha,intento order by fecha desc, intento desc
```

Ilustración 44: Consulta de los test realizados de la aplicación CAPTA SQL

La anterior consulta devuelve la *fecha* y el *id_usuario (id_usuario* como enlace) donde (where) *id_usuario* es el que inicia sesión e *intento* no es nulo. El atributo *intento* es una secuencia de valores que se va incrementando según se realizan test.

Para acceder a el resultado de un día concreto basta con pulsar el icono que está al lado de la fecha:

Ilustración 45: Imagen muestra para acceder a resultado de un test de la aplicación CAPTA SQL

Una vez pulsado el icono, la aplicación muestra el resultado del día que hemos seleccionado en la página siguiente, adicionalmente en la parte superior muestra la fecha, el nombre del usuario, las preguntas contestadas y las correctas.

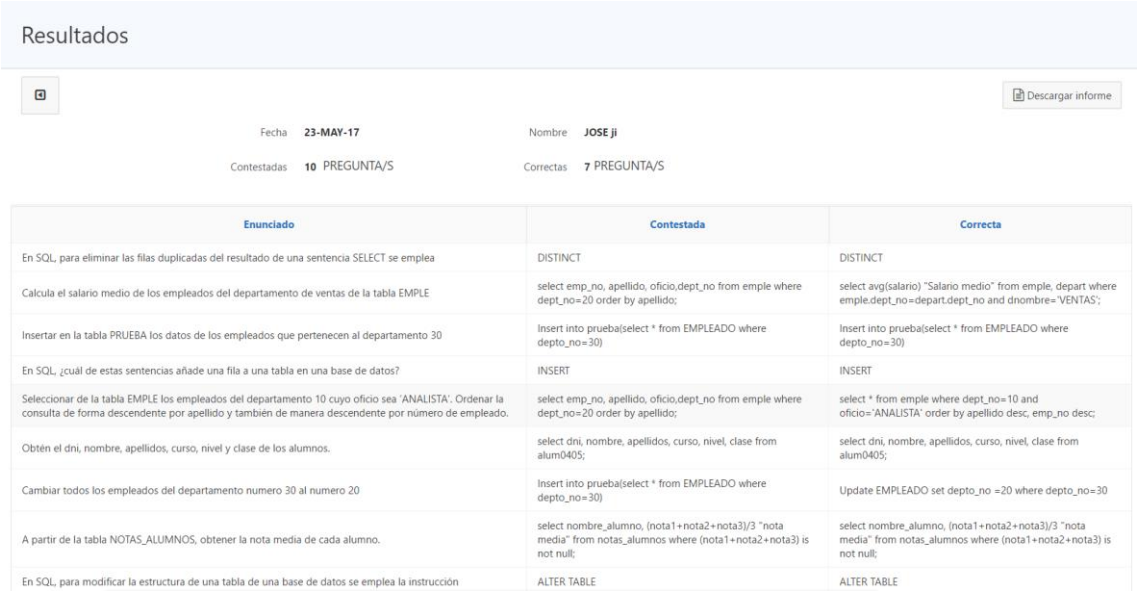


Ilustración 46: Resultado de TEST de la aplicación CAPTA SQL

La anterior pantalla muestra un informe de las preguntas y respuestas realizadas en el test a modo de corrección.

Como opción extra en los resultados de cada TEST, es posible realizar descargas de los resultados. Para ello, por medio de el botón “Descargar informe” situado en la parte superior derecha, un código PL/SQL realiza un informe en formato EXCEL para poder descargarlo.

Función Descargar Informe

```
declare
   numero number := 1;
  cursor result_cur is
     select    p.enunciado as ENUNCIADO,r.respuesta as CONTESTADA, t.respuesta as
CORRECTA
                    from preguntas p, respuestas r, test t
                    where t.id_pregunta=p.id_pregunta and t.id_respuesta=r.id_respuesta
and _usuario||'_'||:p36_fecha||'.html.xls');
  owa_util.http_header_close;
  htp.p('<table border="0" cellpadding="60" cellspacing="5" width="50%">
         <tr>
         <td width="50%" align="center" bgcolor=""></td>
          </tr>');
```

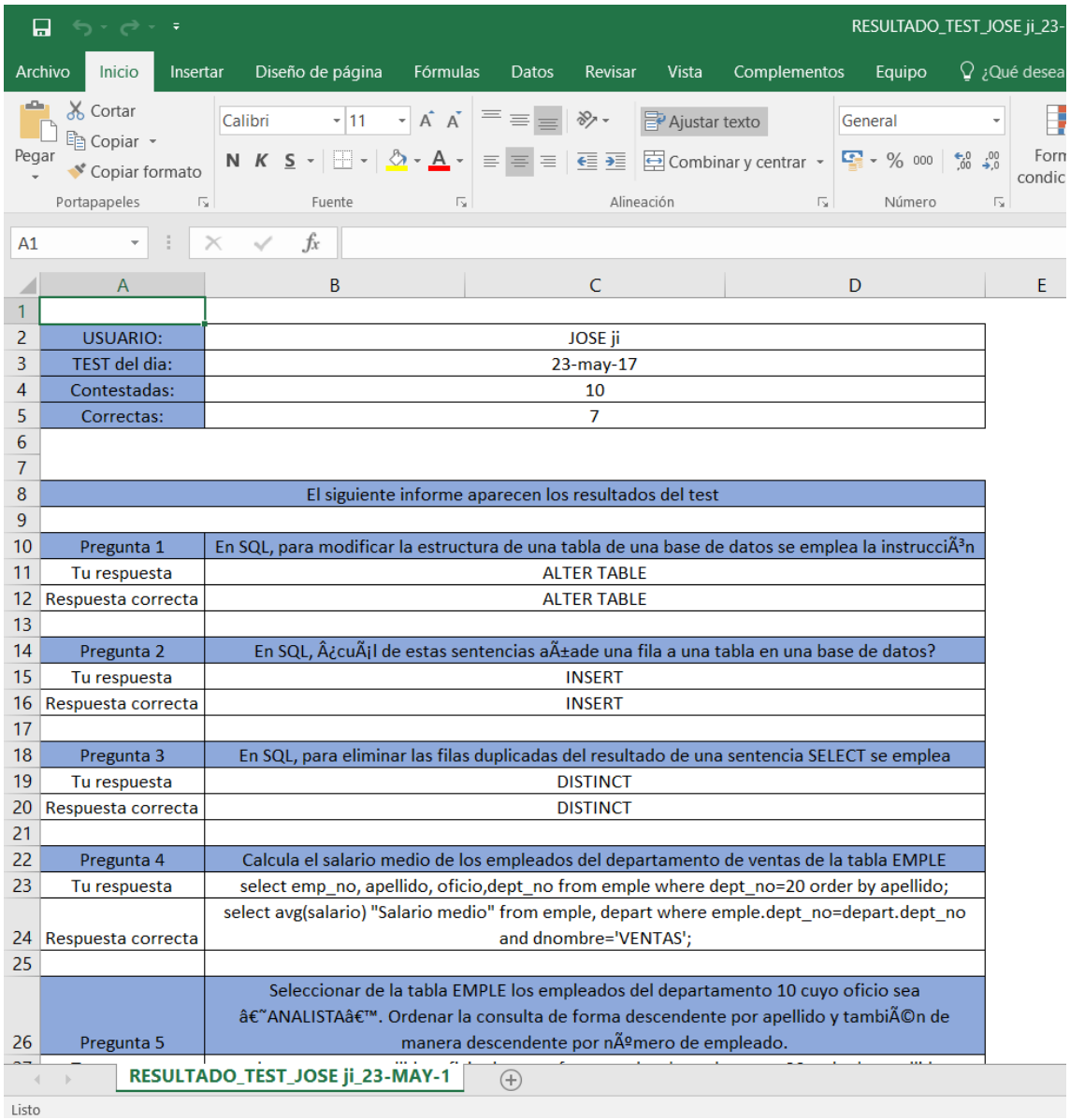
```
 htp.p('<table border="1" cellpadding="0" cellspacing="0" width="50%">');
 htp.print ( '<tr><td width="100%" align="center"
bgcolor="#8EA9DB">'||'USUARIO:'||'</td><td width="100%" align="center"
colspan="3">'||:p36_usuario||'</td></tr>');
          htp.print ( '<tr><td width="100%" align="center" bgcolor="#8EA9DB">'||'TEST del
dia:'||'</td><td width="100%" align="center" colspan="3">'||:p36_fecha||'</td></tr>');
 htp.p('</table>');
htp.p('<table border="0" cellpadding="10" cellspacing="10" width="50%"><tr> <td ="50%"
align="center" bgcolor=""></td></tr><tr>
                    <td width="50%" align="center" bgcolor=""></td>
                    </tr>');

htp.p('<table border="1" cellpadding="0" cellspacing="0" width="50%">
                    <tr>
                    <td width="50%" align="center" colspan="4" bgcolor="#8EA9DB">El
siguiente informe aparecen los resultados del test</td></tr><tr></tr>');
for it in result_cur loop
                    htp.print('<tr><td width="100%" align="center" colspan="1"
bgcolor="#8EA9DB">Pregunta '||numero||'</td><td  width="50%" align="center"
colspan="3" bgcolor="#8EA9DB">'|| it.ENUNCIADO||'</tr>
                                        <tr><td width="100%" align="center" colspan="1">Tu
respuesta</td><td  width="50%" align="center" colspan="3">'|| it.CONTESTADA||'</tr>
                                        <tr><td width="100%" align="center"
colspan="1">Respuesta correcta</td><td  width="50%" align="center" colspan="3">'||
it.CORRECTA||'</tr><td></td>');
numero :=numero+1;
end loop;
htp.p('<tr>   </tr>');
htp.p('</table>');
 htmldb_application.g_unrecoverable_error := true;

end;
```

Ilustración 47: PL/SQL de descarga informe de resultado de TEST de la aplicación CAPTA SQL

La anterior función PL/SQL devuelve los resultados del test en una hoja Excel que muestra el informe completo.



	A	B	C	D	E
1					
2	USUARIO:	JOSE ji			
3	TEST del dia:	23-may-17			
4	Contestadas:	10			
5	Correctas:	7			
6					
7					
8	El siguiente informe aparecen los resultados del test				
9					
10	Pregunta 1	En SQL, para modificar la estructura de una tabla de una base de datos se emplea la instrucciÃ³n			
11	Tu respuesta	ALTER TABLE			
12	Respuesta correcta	ALTER TABLE			
13					
14	Pregunta 2	En SQL, Â¿cuÃ¡l de estas sentencias aÃ±ade una fila a una tabla en una base de datos?			
15	Tu respuesta	INSERT			
16	Respuesta correcta	INSERT			
17					
18	Pregunta 3	En SQL, para eliminar las filas duplicadas del resultado de una sentencia SELECT se emplea			
19	Tu respuesta	DISTINCT			
20	Respuesta correcta	DISTINCT			
21					
22	Pregunta 4	Calcula el salario medio de los empleados del departamento de ventas de la tabla EMPLE			
23	Tu respuesta	select emp_no, apellido, oficio,dept_no from emple where dept_no=20 order by apellido;			
24	Respuesta correcta	select avg(salario) "Salario medio" from emple, depart where emple.dept_no=depart.dept_no and dnombre='VENTAS';			
25					
26	Pregunta 5	Seleccionar de la tabla EMPLE los empleados del departamento 10 cuyo oficio sea â€˜ANALISTAâ€™. Ordenar la consulta de forma descendente por apellido y tambiÃ©n de manera descendente por nÃºmero de empleado.			

Ilustración 48: Informe de resultado de TEST en Excel

Restricciones

- Las preguntas de los test no se repiten a lo largo del mismo.
- Los resultados de los test están accesibles únicamente al propietario de los mismos
- Una vez pulsado el botón “Finalizar” es imposible reanudar el test.

8.5. Descargas

La aplicación pone a disposición de los usuarios un espacio de alojamiento de archivos para compartir con el resto de usuarios.

La pantalla principal de archivos muestra un listado con todos los archivos que están alojados en la aplicación.

En cada archivo se indica la fecha de subida, el propietario y descripción.

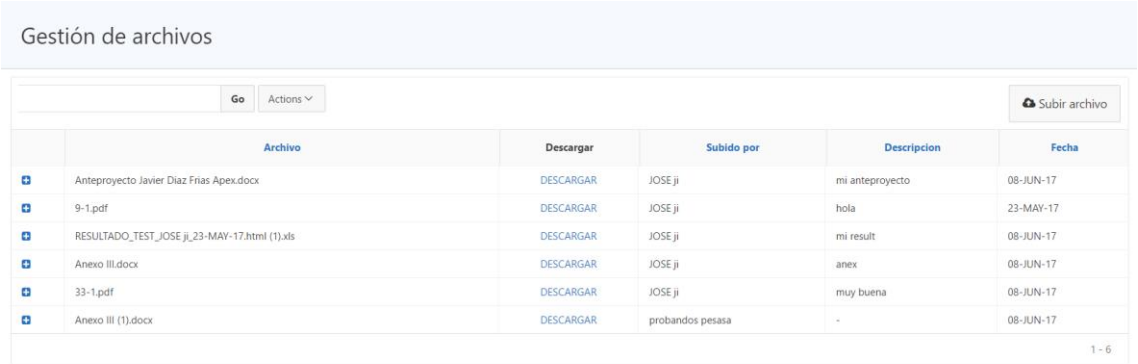


Ilustración 49: Página principal con los archivos compartidos de la aplicación CAPTA SQL

```
select a.id_archivo,dbms_lob.getlength(a.blob_content) as BLOB_CONTENT ,a.filename,
a.last_updated,a.descripcion,u.nombre||' '||u.apellidos,a.id_usuario as hidden
from archivos a, usuarios u
where a.id_usuario=u.id_usuario;
```

Ilustración 50: Consulta que muestra los archivos compartidos de la aplicación CAPTA SQL

Función subir archivo

Para subir un archivo basta con pulsar el botón “Subir archivo” situado en la esquina superior derecha y nos dirige a la página donde se selecciona el archivo, una vez seleccionado el archivo obligatoriamente a de indicarse el nombre del archivo y opcionalmente una descripción.

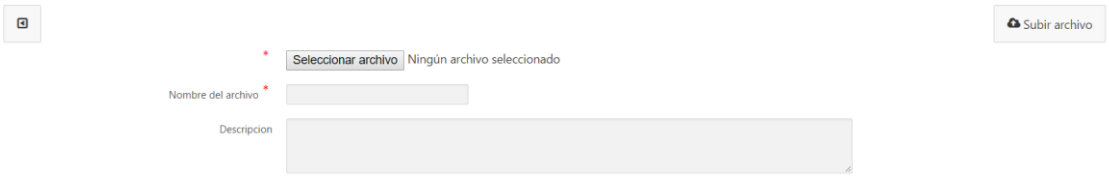


Ilustración 51: Página para subir archivo de la aplicación CAPTA SQL

Cuando se pulsa el botón “Subir archivo” se produce un “INSERT” mediante un proceso automático en la tabla “ARCHIVOS” de la Base de Datos.

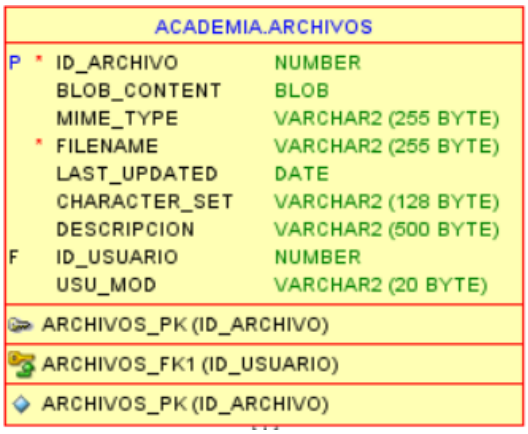


Ilustración 52: Tabla de archivos del modelo de datos empleado en la aplicación CAPTA SQL

El proceso mencionado anteriormente sirve tanto para hacer “INSERT”, “UPDATE” y “DELETE”, por lo que no es necesario crear una función por cada una de estas operaciones. Únicamente hay que indicar la PRIMARY KEY de la tabla y el ítem de APEX que contiene la PRIMARY KEY

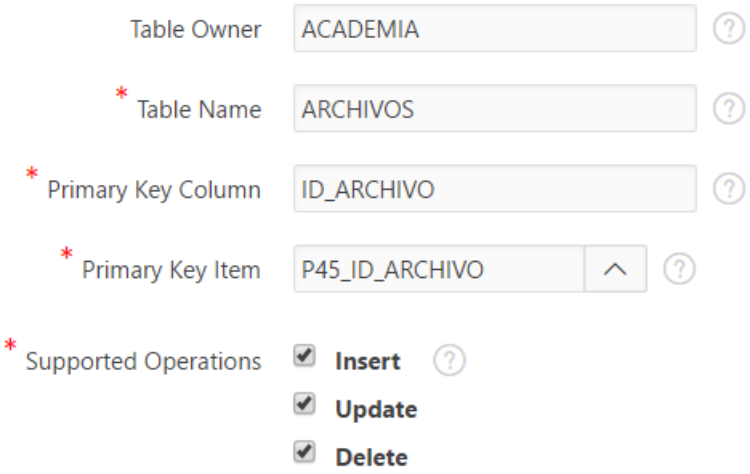


Ilustración 53: Opciones para la gestión de archivos de la aplicación CAPTA SQL

Función modificar archivo

Cuando está cargado el archivo en la Base de Datos es posible realizar modificaciones sobre el título del archivo o en el comentario, para ello, en la página principal de descargas:

1. Seleccionar el archivo deseado pulsando en el símbolo “+”.

Ilustración 54: Procedimiento para modificar archivo de la aplicación CAPTA SQL-1

2. Una vez seleccionado el archivo se muestra el nombre y comentario que está en el archivo. Basta con modificar el campo deseado y aplicar en guardar cambios.

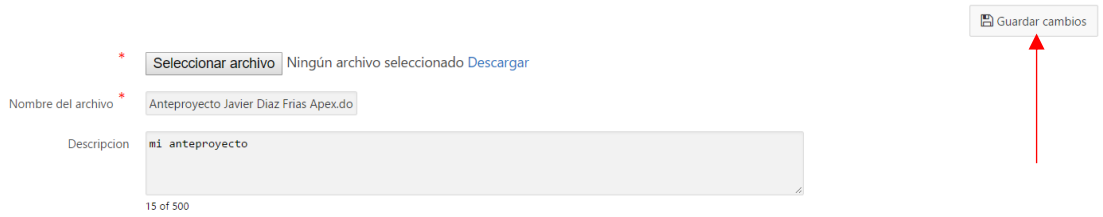


Ilustración 55: Procedimiento para modificar archivo de la aplicación CAPTA SQL-2

La edición de los archivos está restringida al propietario del archivo y a los usuarios administradores.

Función borrar archivo

Cuando está cargado el archivo en la Base de Datos es posible borrarlo, para ello, en la página principal de descargas:

1. Seleccionar el archivo deseado pulsando en el símbolo “+”.

Ilustración 56: Procedimiento para borrar archivo de la aplicación CAPTA SQL-1

2. Pulsar en el botón “Borrar archivo”.

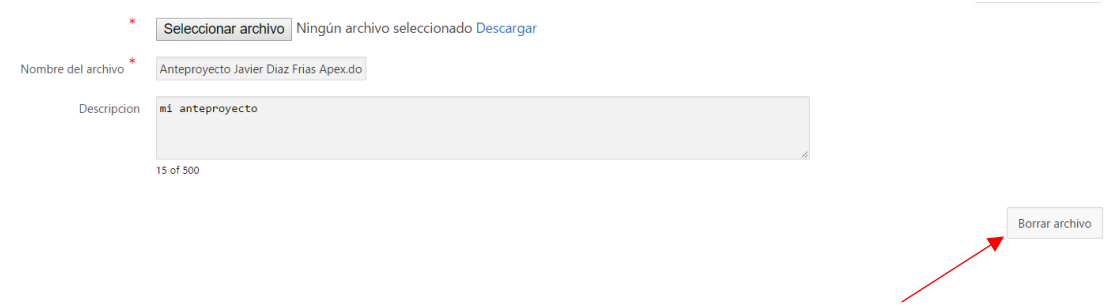


Ilustración 57:Procedimiento para borrar archivo de la aplicación CAPTA SQL-2

El borrado de archivos está restringido al propietario del archivo y a los usuarios administradores.

Restricciones

- La modificación o borrado de archivos está restringida a los propietarios de los mismos o a los administradores.
- El comentario máximo es de 500 caracteres.
- El nombre del archivo tiene como máximo 255 caracteres.

5.6. Foro

La aplicación pone a disposición de los usuarios una pantalla donde es posible intercambiar mensajes entre los distintos usuarios.

El funcionamiento es simple, basta con crear un hilo de comentario basado en un tema, a partir del primer comentario se desarrolla una conversación.

En la página principal del foro, las conversaciones se muestran por temas:

Ilustración 58: Página con todos los comentarios realizados de la aplicación CAPTA SQL

```
select c.id_comentario,c.tema,c.fecha,u.Nombre||'  '||u.apellidos "PUBLICADO POR"
from comentarios c,usuarios u
where c.id_usuario=u.id_usuario and id_parent is null
```

Ilustración 59: Consulta para mostrar los comentarios realizados de la aplicación CAPTA SQL

Función crear comentario

Para crear un comentario hay que rellenar el tema y una descripción en los campos correspondientes. El tema y comentario son campos obligatorios:

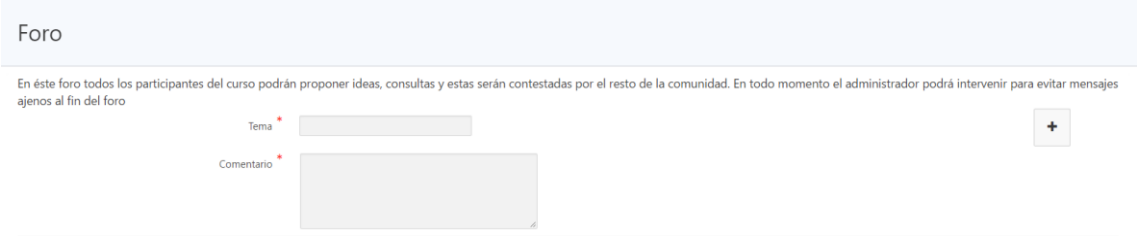


Ilustración 60: Página principal para crear tema nuevo de la aplicación CAPTA SQL

Una vez relleno ambos campos, para crear el comentario hay que pulsar el botón “+” situado en la parte derecha superior.

Este botón realiza una función de “INSERT” en la tabla comentarios:

```
insert into comentarios(id_usuario,fecha,comentario,tema)
values
((select id_usuario from usuarios where username=v('app_user')),(select sysdate from dual),:p37_comentario,:p37_tema);
```

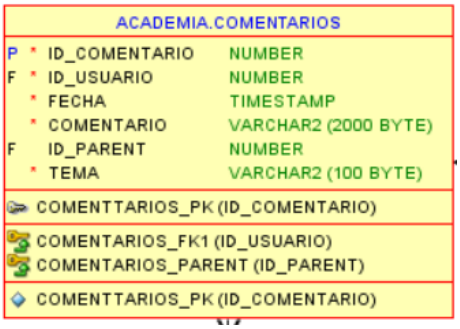


Ilustración 61: Tabla comentarios del modelo de datos

Una vez que se ha creado un comentario, aparecerá en la página principal del foro, este comentario principal establece un tema que puede ser contestado en sub-comentarios.

Contestar un tema

Para contestar un tema basta con seleccionarlo de la página principal de comentarios pulsando en el botón ‘+’ del tema deseado:

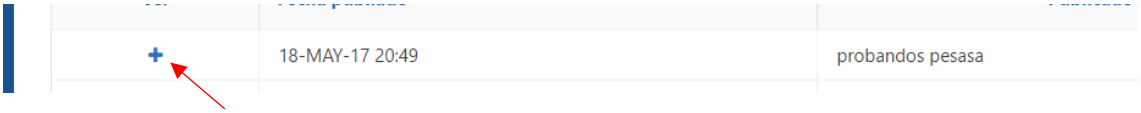


Ilustración 62: Procedimiento para contestar un comentario de la aplicación CAPTA SQL-1

Ese botón dirige a otra página donde es posible contestar el tema principal. Este

procedimiento es similar al descrito anteriormente.

Se realiza un “INSERT” en la tabla comentarios donde se relaciona el comentario con el comentario principal, siento este último el comentario padre.

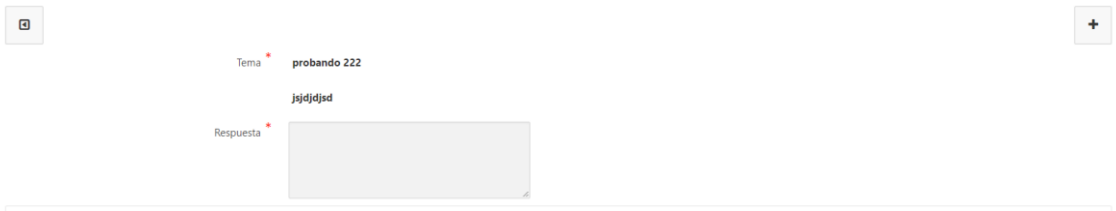


Ilustración 63: Procedimiento para contestar un comentario de la aplicación CAPTA SQL-2

Una vez creado el sub-comentario, aparecerá en la parte de abajo con todos los comentarios ordenados por fecha:

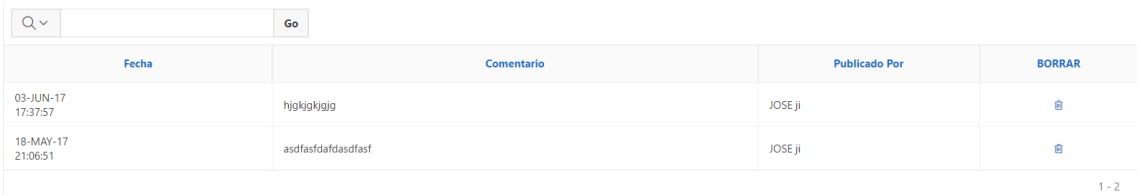


Ilustración 64: Hilo de comentarios de un tema de la aplicación CAPTA SQL

Función borrar comentarios

Cabe la posibilidad de que un comentario se haya insertado erróneamente o simplemente se desee borrar, para ello, el propietario del comentario puede borrar su comentario pulsando en el icono de borrar:

Ilustración 65: Procedimiento para borrar comentario de la aplicación CAPTA SQL

Los comentarios únicamente pueden ser borrados por sus propietarios o por el administrador.

Si se produce un borrador de un tema principal, los sub-comentarios quedan borrados inmediatamente mediante u borrado en cascada.

El proceso de borrado de comentarios se realiza mediante la función “DELETE”:

Restricciones

- Únicamente puede borrar un comentario el propietario y el administrador.
- El tema del comentario tiene una extensión total de 100 caracteres.
- El comentario tiene una extensión total de 2000 Caracteres

```
DELETE  FROM COMENTARIOS WHERE ID_comentario=:P40_ID_COMENTARIO;
```

Ilustración 66: "DELETE" para borrar comentario de la aplicación CAPTA SQL

5.7. Gestión de usuarios

La aplicación posee, como anteriormente se ha visto en el punto 5.1 Pagina de logging, un proceso de autenticación. Para llevar a cabo ese proceso, inicialmente un usuario ha de estar registrado en la aplicación.

El registro debe realizarlo un usuario con el perfil de administrador.

Todos los usuarios registrados se muestran en la página principal de gestión de usuarios:

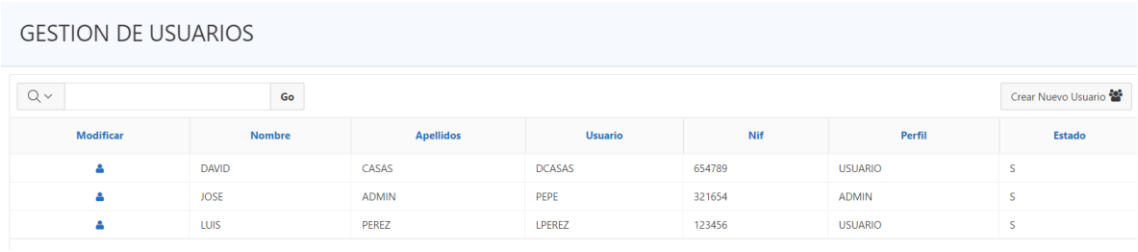


Ilustración 67: Página principal de usuarios de la aplicación CAPTA SQL

```
select id_usuario as "Modificar", nombre, apellidos, username,perfil, NIF, ACTIVO as ESTADO
from usuarios

order by nombre, apellidos
```

Ilustración 68: Consulta que muestra los usuarios de la aplicación CAPTA SQL

Los usuarios tienen dos tipos de perfiles:

- Administradores: acceso total a toda la aplicación, puede borrar comentarios y borrar archivos de otros usuarios. Puede gestionar todos los usuarios, dar de alta y dar de baja.
- Usuario: acceso únicamente a los objetos que tiene de su propiedad, no puede modificar o borrar archivos o comentarios de otros usuarios. Puede modificar su contraseña únicamente.

Función crear nuevo usuario

Para crear un nuevo usuario en la aplicación, desde la página principal de usuarios, basta con pulsar en el botón “Crear nuevo usuario” situado en la parte superior derecha. Este botón muestra la página con los campos necesarios para crear un usuario y que debe rellenar el administrador:

Ilustración 69 : Pagina para crear nuevo usuario de la aplicación CAPTA SQL

Una vez ha completado el relleno de los campos, pulsando en el botón “Aceptar” se produce un “INSERT” en la Base de Datos en la tabla de Usuarios.

```
insert into usuarios (id_usuario,nombre,apellidos,nif,username,perfil,password)
values (:P3_AUX,:P3_NOMBRE,:P3_APELLIDOS,:P3_NIF,:P3_USERNAME,:P3_PERFIL,'usuario');
```

Ilustración 70: "INSERT" para crear nuevo usuario de la aplicación CAPTA SQL

Por defecto la password que se otorga a cada usuario es “usuario”, aunque es posible cambiarla posteriormente.

Función modificar usuario

Una vez el usuario está creado, cabe la posibilidad de que algún dato esté erróneo o simplemente se desee cambiar. Para ello es posible modificar el usuario que se desee.

Basta con seleccionar el usuario que se quiera modificar de la lista de los usuarios de la página principal de gestión de usuarios y nos mostrará la información del usuario:

Ilustración 71 : Procedimiento para modificar un usuario de la aplicación CAPTA SQL-1

Ahora basta con cambiar el campo que se quiere cambiar y pulsar en el botón “Aplicar Cambios” para guardar los cambios realizados.

Ilustración 72: Procedimiento para modificar un usuario de la aplicación CAPTA SQL-2

La modificación de los datos se realiza mediante un “UPDATE” en la Base de Datos:

```
begin
    update USUARIOS
    set  USERNAME = :P5_USERNAME,
        APELLIDOS = :P5_APELLIDOS,
        NIF = :P5_NIF,
        NOMBRE = :P5_NOMBRE,
        PERFIL = :P5_PERFIL,
        ACTIVO = :P5_ACTIVO
    where ID_USUARIO = :P5_ID_USUARIO;
end;
```

Ilustración 73: "UPDATE" para modificar usuario de la aplicación CAPTA SQL

Función Dar de baja/alta usuarios

Siguiendo los mismos pasos que en apartado anterior de “Modificar un usuario”, para dar de baja o dar de alta un usuario basta con seleccionar en el campo “Activo”, S para dar de alta y N para dar de baja y posteriormente pulsar el botón “Aplicar cambios”.

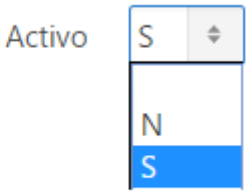


Ilustración 74: Opciones de estado de un usuario de la aplicación CAPTA SQL

Los usuarios no se borran de la Base de Datos debido a que puede poseer objetos, así como archivos o comentarios. De producir borrado de usuarios, esos objetos quedarían huérfanos en la Base de Datos y por tanto existiría inconsistencia de datos.

Función cambiar contraseña

Esta función está disponible para todos los usuarios de la aplicación, se accede desde la barra superior tal y como veíamos en el primer punto de la documentación “5.2 Página principal”.

La contraseña por defecto que se otorga a cada usuario nuevo de la aplicación es “usuario”, por motivos de seguridad es recomendable cambiar la contraseña por una más segura.

Para ello basta con introducir la contraseña deseada en los campos indicados y la contraseña actual.

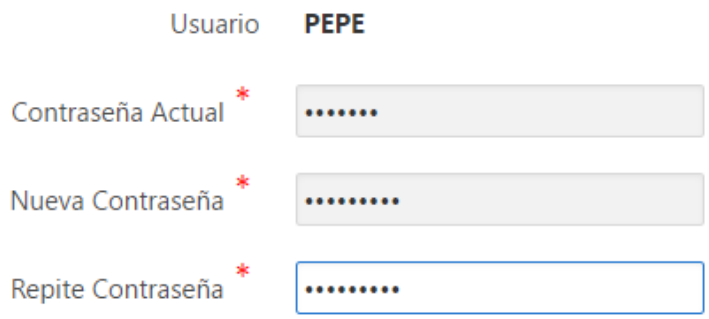


Ilustración 75: muestra de contraseña encriptada de la aplicación CAPTA SQL

La modificación de contraseña se realiza por medio de un procedimiento de Base de Datos:

```
BEGIN
#OWNER#.PKG_AUTENTICACION.P_ACTUALIZAR_PASSWORD(
P_USERNAME => :P4_USERNAME,
P_PASSWORD => :P4_PASSWORD);
END;
```

Ilustración 76 : paso de valores al procedimiento en BBDD para cambiar contraseña a usuario de la aplicación CAPTA SQL.

```
procedure p_actualizar_password(p_username in VARCHAR2, p_password IN VARCHAR2)
  as
begin
    update USUARIOS set password = f_encriptar_pwd(p_password,p_username) where username = p_username;
end p_actualizar_password;
```

Ilustración 77: Procedimiento para cambiar contraseña a usuario de la aplicación CAPTA SQL

Una vez modificada la contraseña se produce una encriptación de la misma, por lo que en Base de Datos es imposible averiguar qué contraseña se ha establecido.

Por ejemplo, el usuario “PEPE” tiene como contraseña “qwertyuio”. En Base de Datos aparece:

PEPE	JOSE	AA1E951C346D4A92E7B97CA6F713C94B	A

Ilustración 78: Muestra de contraseña encriptada de un usuario de la aplicación CAPTA SQL

La encriptación se realiza por medio de una función de Base de Datos, se ingresa la contraseña y por medio de funciones internas la contraseña se encripta:

```
function f_encriptar_pwd(p_password IN VARCHAR2, p_username IN VARCHAR2)
  return VARCHAR2
is
    salt varchar2(4000) := 'ISYmHMtSrjFmT2nEZUvEU5LA3jrV3i';
begin
    return utl_raw.cast_to_raw(
        dbms_obfuscation_toolkit.md5(
                                input_string => salt || p_password || upper(p_username)
        )
    );
end f_encriptar_pwd;
```

Ilustración 79: Función que realiza encriptación de la contraseña de un usuario de la aplicación CAPTA SQL

Hasta aquí la documentación de la aplicación CAPTA SQL desarrollada en APEX.

8. ANEXO 2: Instalación de Base de Datos

En este anexo se documenta el procedimiento de instalación de la base de datos utilizada para el desarrollo del trabajo final de grado.

En primer lugar, debemos contar con una cuenta de Oracle, de no ser así hay que crearla, es gratuita:

https://profile.oracle.com/myprofile/account/create-account

Si ya se tiene creada la cuenta en Oracle:

1. Se inicia sesión y se va al apartado de descargas.
2. Se busca la versión de base de datos que se desea, en nuestro caso Oracle Database 11g XE 64 bit.

3. Una vez concluye la descarga, se lanza la instalación.

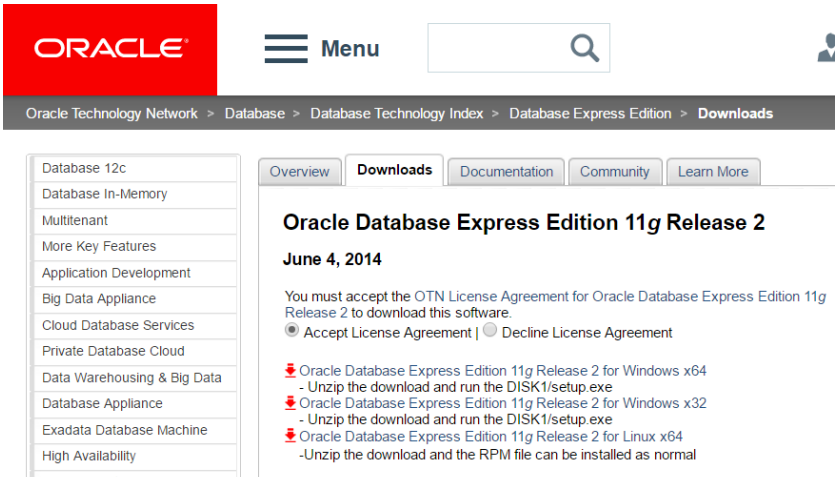


Ilustración 80: Página de descargas de ORACLE

4. Una vez se tiene cargado el administrador de la instalación, se selecciona la Base de Datos que hay que instalar.

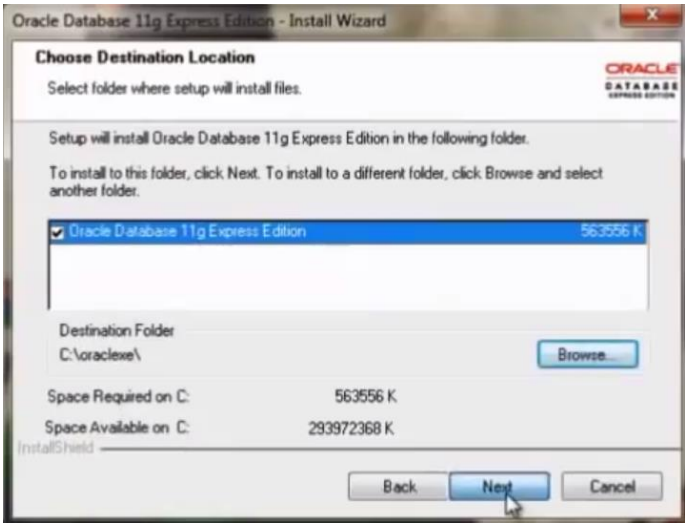


Ilustración 81: Selección de Base de Datos en el instalador

5. El instalador pedirá que se ingrese las contraseñas de los usuarios SYS y SYSTEM. Estas contraseñas se deben conservarlas siempre.

Ilustración 82: Página de Contraseñas de Instalación de la Base de Datos

6. Una vez ha terminado la instalación se sale del instalador:

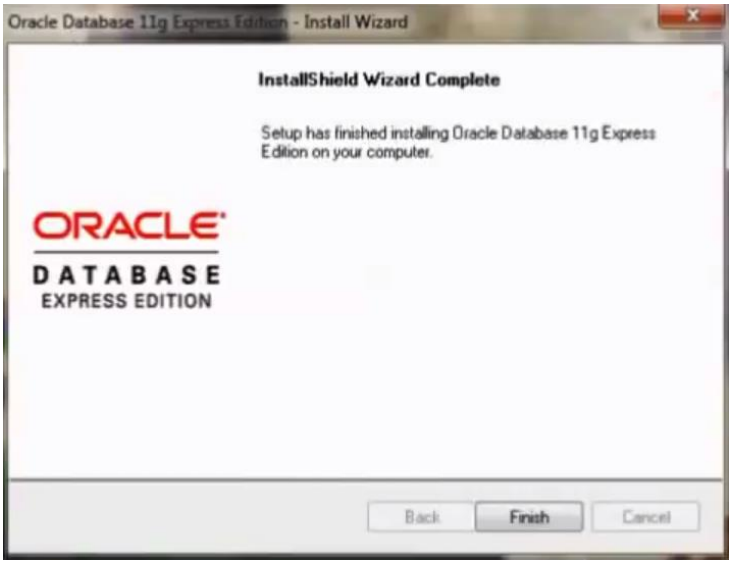


Ilustración 83: Finalizar instalador de Base de Datos

Ya ha concluido la instalación de La base de Datos.

9. ANEXO 3: Instalación de APEX

En el siguiente anexo se documenta el proceso de instalación de APEX. Como requisito debe estar instalada una Base de datos de Oracle.

1. Descargar el software de APEX de la página de Oracle:
   http://www.oracle.com/technetwork/developer-tools/apex

2. Una vez descargado y descomprimido se pega en la carpeta donde está instalada la Base de Datos:

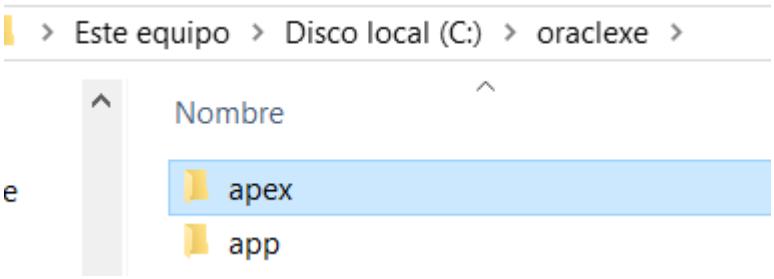


Ilustración 84: Ruta de instalación de APEX en Windows

3. A través de una ventana de comandos, se va a la ruta de APEX y se abre conexión con la Base de Datos:

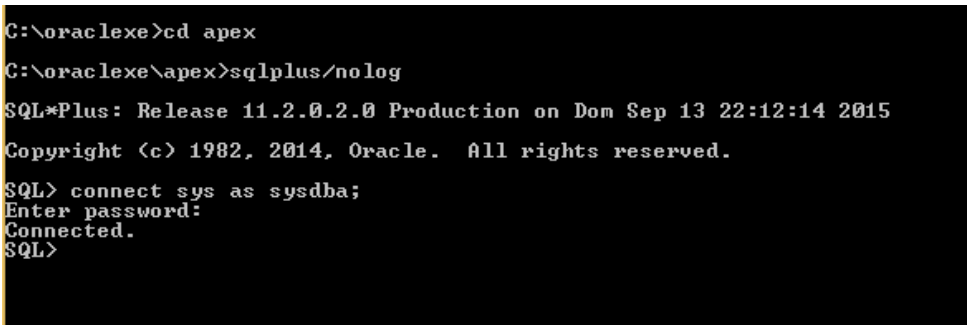


Ilustración 85: Conexión a la Base de Datos desde Windows

4. Se ejecuta el fichero de instalación de APEX: @apexins.sql USERS TEMP /i/

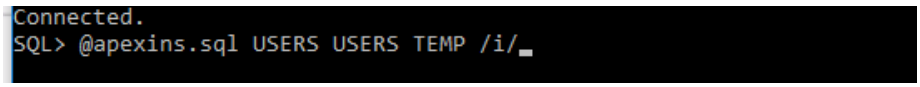


Ilustración 86: Sentencia instalación de APEX

5. Se sale de SQL plus y de las carpetas y se vuelve a entrar. Terminada la instalación se conecta nuevamente con SYS y se lanza los siguientes parámetros de configuración del entorno:

Alter user anonymous account unlock;

Alter system set job_queue_processes = 20;

Alter system set processes = 300 scope = spfile;

Alter system set transactions = 370 scope=spfile;

Alter system set sessions=345 scope=spfile;

6. Se crea un directorio para datos:

   @apxldimg.sql c:\oraclexe

Ya está instalado APEX en el equipo.

Universidad de Alcalá
Escuela Politécnica Superior

ESCUELA POLITECNICA
SUPERIOR